A Neural Network-Based Classification of Environment Dynamics Models for Compliant Control of Manipulation Robots

Dusko Katic and Miomir Vukobratovic

***Abstract*— In this paper, a new method for selecting the appropriate compliance control parameters for robot machining tasks based on connectionist classification of unknown dynamic environments, is proposed. The method classifies the type of environment by using multilayer perceptron, and then, determines the control parameters for compliance control using the estimated characteristics. An important feature is that the process of pattern association can work in an on-line mode as a part of selected compliance control algorithm. Convergence process is improved by using evolutionary approach (genetic algorithms) in order to choose the optimal topology of the proposed multilayer perceptron. Compliant motion simulation experiments with robotic arm placed in contact with dynamic environment, described by the stiffness model and by the general impedance model, have been performed in order to verify the proposed approach.**

I. Introduction

MANY manipulation robots, especially in industrial practice, are required to operate in uncertain environments. Thus, the characteristics of the environment can be assumed to be unknown and to significantly change according to the task to be done. Beside environment uncertainties, for some type of robots, uncertainties of robot dynamic model can have a strong influence on the quality of robot performance. In this case, one of the most delicate problems in compliant motion control of robots interacting with the dynamic environment is the stability of both desired motion and interaction forces. A multitude of various control approaches such as hybrid control, stiffness control, impedance control, damping control, etc. [1], point to the stability of control task as a problem which is not yet satisfactorily solved, both from the theoretical and the practical standpoint. Namely, when considering specific contact tasks, simplifications in the modeling of robot and environment dynamics are introduced in almost all approaches, in order to obtain simpler control algorithms. Very popular approach is to describe the environment by a set of algebraic equations, assuming that the robot motion in contact is kinematically constrained [2], [3]. Colgate and Hogan consider the environment to be a linear time invariant dynamic system [4]. In both cases experimental verification [5] led to the discovery of instability caused by the environment dynamics.

In [6], McClamroch and Wang emphasized the important role of the constraints in the compliance control, especially with relation to the stabilization problem. They presented global conditions for tracking, based on a modified computed torque controller, and local conditions for feedback stabilization using a linear controller. The closed loop properties in the case of force disturbances, dynamics in the force feedback loops, or uncertainty in the constraint functions were also investigated. In [7], Eppinger and Seering studied the influence of unmodeled dynamics on contact task stability, introducing additional (elastic) degrees of freedom of both the robot and the environment. Nevertheless, it was concluded that environment dynamics cannot cause instability. In order to extend the problem solving to the more general case when the environment exhibits a dynamic behavior [8], Vukobratovic and Ekalo have established a unified approach to control simultaneously position and force in an environment with completely dynamic reactions [9]. Vukobratovic and Ekalo [9] and Vukobratovic [10] especially focused their attention to the role of dynamic environment and the stabilization of the position, when asymptotic stability of the contact force was ensured. This task is the basic problem of controlling the robot interacting with dynamic environment. However, without knowing a sufficiently accurate environment model it is not possible to determine, for instance, nominal (desired) contact force. Besides, insufficiently accurate environment dynamics model can significantly influence the contact task performance. Also, in the case of unknown environment, it is very difficult to determine the maximum boundary of the feedback gains.

Hence, there still remain two critical problems in contact task research: 1) how to determine control parameters under the uncertain characteristics of robot and environment and 2) how to deal with nonlinear characteristics of robot and dynamic environment. One excellent possible solution is to use the learning concept for contact tasks, since we can significantly enhance robotic performance by learning capabilities which use an a priori low level of information about model of manipulation robot and environment. The second important characteristic of contact tasks is their repetitive nature which is very important for process of learning by trial-and-error procedure.

With recent extensive research in the area of robot position/force control, various learning algorithms for constrained manipulation have been proposed such as iterative-analytical, tabular and connectionist (neural networks) methods [11]. Neural networks are able to compensate for a wide range

Manuscript received March 17, 1996; revised November 3, 1996.
The authors are with the Robotics Centre, Mihailo Pupin Institute, 11000 Belgrade, Yugoslavia (e-mail: d.katic@ieee.org; vuk@robot.imp.bg.ac.yu).
Publisher Item Identifier S 1083-4419(98)00214-3.

of robot uncertainties and to perform excellent association and knowledge generalization. In the application of neural networks in robot contact tasks two essentially different approaches can be distinguished: one, the aim of which is the transfer of human manipulation skills to robot controllers, and the other in which manipulation robot is treated as an independent dynamic system in which learning is achieved through repetition of the working task. The principle of transfer of human manipulation skill is developed in the papers of Asada and coworkers [12], [13]. The approach is based on acquisition of manipulation skills and strategies from human experts and their transfer to the robot controller by learning connectionist structures. The second group of learning methods, based on autonomous on-line learning procedures with the repetition of the working task, is also evaluated through several algorithms [14]±[20]. The main distinction between these algorithms is in the aim of learning which is in rst case the on-line modi cation of the control signal, and in the second the building of internal model of robotic system.

Previous research, however, did not specially consider the problem of environment uncertainties. Without adequate knowledge about environment dynamics it is not even possible to determine consistent values of nominal trajectory and force, as well as nominal control, not to mention achieving asymptotic stability. In this case, algorithms that identify the type of environment models on-line, could signi cantly improve the performance of contact task control schemes. As one solution, off-line identi cation of environment parameters based on experimental measuring [21] may also result in good system performance with approximate modeling of robot dynamic environment that would be suf ciently exact. But in the case of nonlinear complex models of environment or uncertain structure of environment model, conventional parameter identi cation method is not a solution for compliance control synthesis.

As another solution for the expressed problem, some researchers [22]±[24] used the intelligent techniques (neural networks and fuzzy logic) for dynamic environment identi cation. In paper [22], direct method of environment parameter identi cation using recurrent neural networks with terminal attractors for space robotic applications is proposed. Results indicate good performance in learning and generalization processes. Very interesting approach is using intelligent techniques for dynamic environment classi cation, instead using intelligent techniques for parameter identi cation. Cha *et al*. [23] used the indirect method with neural networks for telerobotic purposes, in order to classify the dynamic environment, and fuzzy logic to select force re ection gain based on estimated characteristic of the environment.

In this paper, a new method for selecting the appropriate control parameters and parameters of dynamic robot environment for robot machining tasks, based on connectionist classi cation of unknown dynamic environments, is proposed. This method classi es the type of robot environment using multilayer perceptron through off-line training process and through process of on-line pattern association. It is assumed that for classi ed dynamic environment, the control parameters and parameters of environment models (structure of environment model is known) are de ned in advance, or that they can be obtained by the process of linear interpolation. It is important that the process of pattern association by proposed multilayer perceptron can work in an on-line mode as an integral part of selected compliance control algorithm [9]. Based on classi cation and generalized features of the proposed neural network, acquired in off-line training process, the control algorithm can select the appropriate control parameters which achieve the satisfying system performance. In the proposed off-line training algorithm, convergence process is improved by using evolutionary approach (genetic algorithms) in order to choose the appropriate topology of the proposed multilayer perceptron. It is important to notice that it is assumed that the manipulation robot is a deterministic system without uncertainties. There are some other papers about connectionist approach to robot contact tasks based on compensation of dynamic robot model uncertainties [19], [25]±[27].

The paper is organized as follows: In Section II, factors affecting contact task performances in stabilizing position/force control algorithms are analyzed. In Section III, the basic principles of connectionist approach utilized for environment classi cation purposes and selection of appropriate control parameters, including applied learning rules and evolutionary approach, are introduced. In Section IV, overall structure of the proposed connectionist control algorithms is presented. In Section V, the proposed approach is veri ed through simulation experiments. Section VI concludes the paper with a discussion on ongoing and future work.

II. Factors Affecting Task Performance and Stability in Robotic Compliance Control

In this section we introduce the speci c nonlinear and linear models of robot and environment that are considered for classi cation and control purposes, as well as special control algorithms for stabilizing position and force based on quality of transient processes [9]. These stabilizing control laws ensure exponential stability of the closed loop systems. In order to connect these control algorithms with connectionist classi cation, factors affecting task performance and stability in control algorithms based on classi cation of unknown dynamic environment are specially analyzed. The main idea of using neural networks for classi cation of unknown robot dynamic environment can be ef ciently applied to other types of robot contact control algorithms, too. Since this paper primarily considers contact with unknown environments, problems related to gross motion control and impact control are neglected.

The dynamic model of the robot interacting with the environment is described by a vector differential equation in the form

$$\mathbf{H}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{J}^T(\mathbf{q})\mathbf{F} = \tau \qquad (1)$$

where, $\mathbf{q} = \mathbf{q}(t)$ is an n-dimensional vector of robot generalized coordinates; $\mathbf{H}(\mathbf{q})$ is an $n \times n$ positive de nite matrix of inertia moments of the manipulation mechanism; $\mathbf{h}(\mathbf{q}, \dot{\mathbf{q}})$ is an n-dimensional nonlinear function of centrifugal, Coriolis, and gravitational moments; $\tau = \tau(t)$ is an n-dimensional vector of input control; $\mathbf{J}^T(q)$ is an $n \times m$ Jacobian matrix connecting the velocities of robot end-effector and the velocities of robot generalized coordinates; and $\mathbf{F} = \mathbf{F}(t)$ is an m-dimensional vector of generalized forces or of generalized forces and moments from the environment acting on the end-effector. Presently, it will be adopted that $n = m$ (in general $n \geq m$), where m is number of contact force components.

The model of working environment represents one of the most complex and least investigated problems in the robot contact tasks. In the case when the environment does not possess the displacements that are independent from the robot motion extra degrees of freedom, mathematical model of the environment can be described by nonlinear differential equations [8]

$$\mathbf{M}(\mathbf{s})\ddot{\mathbf{s}} + \mathbf{L}(\mathbf{s}, \dot{\mathbf{s}}) = \mathbf{F}, \quad \mathbf{s} = \varphi(\mathbf{q}) \tag{2}$$

where $\mathbf{s}$ is a vector of environment coordinates (displacements) and $\varphi(\mathbf{q})$ is a vector function connecting two coordinate frames. In the frame of robot joint coordinates, the model of environment dynamics can be presented in the form

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{L}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{S}^T(\mathbf{q})\mathbf{F} \tag{3}$$

where $\mathbf{M}(\mathbf{q}) \in R^{n\times n}$ is a nonsingular matrix; $\mathbf{L}(\mathbf{q}, \dot{\mathbf{q}}) \in R^n$ is a nonlinear vector function; and $\mathbf{S}^T(\mathbf{q}) \in R^{n\times n}$ is the matrix with $\mathrm{rank}(\mathbf{S}) = n$.

The end-effector of the manipulator is constrained on static geometric surfaces

$$\Phi(\mathbf{q}) = 0 \tag{4}$$

where $\Phi(\mathbf{q}) \in R^m$ is the holonomic constraint function.

The presented forms of the robot dynamics model and the model of the dynamic environment which can be used for learning control synthesis have important features that are given in nonlinear form of generalized coordinates, although the commonly used mathematical models for contact tasks are based on linearized models and external coordinates [1]. Hence, in practice it is convenient to adopt a simpli ed model of the environment, taking into account the dominant effects, such as stiffness

$$\mathbf{F} = \mathbf{K}'(\mathbf{x} - \mathbf{x_0}) \tag{5}$$

or an environment damping during the tool motion

$$\mathbf{F} = \mathbf{B}'\dot{\mathbf{x}} \tag{6}$$

where $\mathbf{K}' \in R^{n\times n}$, $\mathbf{B}' \in R^{n\times n}$ are semide nite matrices describing the environment stiffness and a damping, respectively, and $\mathbf{x_0} \in R^n$ denotes the coordinate vector in Cartesian coordinates of the point of contact between the end-effector (tool) and a constraint surface. However, it is more appropriate to adopt the relationship de ned by speci cation of the target impedance [28]

$$\mathbf{F} = \mathbf{M}'\Delta\ddot{\mathbf{x}} + \mathbf{B}'\Delta\dot{\mathbf{x}} + \mathbf{K}'\Delta\mathbf{x} \tag{7}$$

where

$$\Delta\mathbf{x} = \mathbf{x} - \mathbf{x_0} \tag{8}$$

and $\mathbf{M}'$ is a positive de nite inertia matrix. The matrices $\mathbf{M}', \mathbf{B}', \mathbf{K}'$ de ne the target impedance which can be selected to correspond to various objectives of the given manipulation task. The application of this linear model may be limited to the class of contact tasks, with which either the environment dynamics can be described suf ciently exactly by a linear equation, or to the tasks where the application of linearized equations of the environment dynamics is admissible. The latter is possible, e.g. in the case of solving the assembly task, when jamming of parts during the assembly should be avoided.

In the case of contact with the environment, the robot control task can be described as robot motion along a programmed trajectory $\mathbf{q_P}(t)$ representing a twice continuously differentiable function, when a desired force of interaction $\mathbf{F_P}(t)$ acts between the robot and the environment. Thus, the programmed motion $\mathbf{q_P}(t)$ and desired interaction force $\mathbf{F_P}(t)$ cannot be arbitrary. These two functions must satisfy the following relation:

$$\mathbf{F_P}(t) \equiv \mathbf{f}(\mathbf{q_P}(t), \dot{\mathbf{q}}_\mathbf{P}(t), \ddot{\mathbf{q}}_\mathbf{P}(t)). \tag{9}$$

The control goal of robot interacting with dynamic environment can be formulated in the following way.

Let us de ne the control $\tau(t)$ for $t \geq t_0$, that is to satisfy the target conditions

$$\begin{aligned} \mathbf{q}(t) &\to \mathbf{q_P}(t) \quad \text{as } t \to \infty \\ \mathbf{F}(t) &\to \mathbf{F_P}(t) \quad \text{as } t \to \infty \end{aligned}. \tag{10}$$

In this paper, we generally adopted the principle of control laws synthesis on the basis of preset quality of the transient responses [9]. Because of the relation de ned by (9), simultaneous stabilization of perturbed robot motion and perturbed interaction force with independent requirements for a desired quality of their transient responses is not possible.

As a rst example, the control algorithm based on stabilization of the robot motion with a preset quality of transient responses is considered, which has the following form [9]:

$$\tau = \mathbf{H}(\mathbf{q})[\ddot{\mathbf{q}}_\mathbf{P} - \mathbf{KP}\eta - \mathbf{KD}\dot{\eta}] + \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{J}^T(\mathbf{q})\mathbf{F}. \tag{11}$$

The family of desired transient responses is speci ed by the vector differential equation

$$\ddot{\eta} = -\mathbf{KP}\eta - \mathbf{KD}\dot{\eta} \tag{12}$$

$$\eta(t) = \mathbf{q}(t) - \mathbf{q_P}(t) \tag{13}$$

where $\mathbf{KP} \in R^{n\times n}$ is the diagonal matrix of position feedback gains and $\mathbf{KD} \in R^{n\times n}$ is the diagonal matrix of velocity feedback gains. The right side of (12), i.e., PD-regulator is chosen such that the system de ned by (12) is asymptotically stable in the whole. The values of matrices $\mathbf{KP}$ and $\mathbf{KD}$ can be chosen according to algebraic stability conditions.

The proposed control law represents a version of the well-known computed torque method including force term which uses dynamic robot model and the available on-line information from the position, velocity and force sensors. The important characteristic of this control algorithm is that model of robot environment does not have any in uence on the performance of control algorithm. Hence, in uence of different robot environments is expressed through different values of initial force at robot tip, i.e., through to different parameters of environment model. In initial contact there are different values of initial force for various robot environments, while all other model and control parameters are equal. These different initial conditions cause different force transient responses. Also, the inclusion of noises from robot sensors cause different force steady-state responses for various robot environments. But, if our aim is the to achieve same quality of force steady-state responses for different environments, the same force performance can be achieved only with different values of PD gains. Hence, in this case PD local gains based on classification of robot environment affect the task performance.

As the second example, control algorithm based on stabilization of the interaction force with a preset quality of transient responses is considered, which has the following form [9]:

$$\begin{aligned}\tau = {} & \mathbf{H}(\mathbf{q})\mathbf{M}^{-1}(\mathbf{q})[-\mathbf{L}(\mathbf{q},\dot{\mathbf{q}}) + \mathbf{S}^T(\mathbf{q})\mathbf{F}] + \mathbf{h}(\mathbf{q},\dot{\mathbf{q}}) \\ & + \mathbf{J}^T(\mathbf{q})\left\{\mathbf{F_p} - \int_{t_0}^{t}\left[\mathbf{KFP}\mu(\omega) + \mathbf{KFI}\int_{t_0}^{t}\mu(\omega)\,dt\right]d\omega\right\}\end{aligned} \tag{14}$$

where $\mu(t) = \mathbf{F}(t) - \mathbf{F_p}(t)$; $\mathbf{KFP} \in R^{n\times n}$ is the matrix of proportional force feedback gains; and $\mathbf{KFI} \in R^{n\times n}$ is the matrix of integral force feedback gains. Here, it has been assumed that the interaction force in transient process should behave according to the following differential equation:

$$\dot{\mu}(t) = \mathbf{Q}(\mu) \tag{15}$$

$$\mathbf{Q}(\mu) = -\mathbf{KFP}\mu - \mathbf{KFI}\int_{t_0}^{t}\mu\,dt. \tag{16}$$

PI force regulator (continuous vector function $\mathbf{Q}$) is chosen such that the system defined by (15) is asymptotically stable in the whole.

In this case, environment dynamics model has explicit influence on the performance of contact control algorithm, also having influence on PI force local gains. It is clear that without knowing a sufficiently accurate environment model (parameters of matrices $\mathbf{M}(\mathbf{q}), \mathbf{L}(\mathbf{q},\dot{\mathbf{q}}), \mathbf{S}(\mathbf{q})$) it is not possible to determine the nominal contact force $\mathbf{F_p}(\mathbf{t})$. Besides that, inexact model of environment dynamics can significantly influence the contact task performances. Hence, in our analysis, if the aim is to obtain the same quality of force steady-state processes for different environments, the same force performances can be achieved only by parameter identification of robot environment models, and with equal fixed PI force local gains. The unknown parameters of robot environment model have a greater significance for system performance in comparison with PI force local gains. Hence, in this case, parameters of robot environment models based on classification of robot environment affect the task performance.

The previously selected compliance control algorithm is a very complex control algorithm, but in order to enhance robustness of compliance control algorithms due to high uncertainties in modeling of robot environment, more relaxed practical stability conditions can be used [29]. The use of practical stability tests enable the study of dynamic effects that have to be compensated for given control algorithm under given conditions (manipulation robot structure, environment dynamics, required performances, i.e., speed, accuracy, etc.).

III. The Connectionist Classification of Unknown Robot Environments

The two control algorithms presented in the previous section do not work in a satisfactory way if there is no sufficiently accurate information about the type of robot environment and the parameters of their models. Hence, our idea is to use the connectionist approach instead of parameter identification, which is capable to assure a sufficiently exact classification of type of robot dynamic environment and determination of environment model parameters through the off-line learning process. A neural network classifier based on four-layer perceptron is chosen for the purpose of classification due to its good generalization properties. Its objective is to classify the type of the environment in an on-line manner. Hence, the application of the connectionist approach to this type of problems is divided into two phases: *first*, related to the acquisition process and off-line training of the proposed neural network, and *second* association phase, where on-line control algorithms based on excellent generalization properties of neural network classifier must assure the necessary quality of the system performances.

A. Acquisition Process of Neural Classifier—The First Phase

In the acquisition process of the first phase, based on the real-time realization of two proposed contact control algorithms and using a previously chosen set of different working environments, force data from force sensors are collected. In the case of the first simple example (compliance control algorithm with stabilizing robot motion), for each chosen robot environment and for the chosen contact control algorithm, values of normal force $\mathbf{F_n}(t)$ in time instants $(t), (t-1), (t-2)$, and $(t-3)$ are measured, calculated, and stored as special input patterns for training of neural network. In the case of the second complex example with compliance control algorithm with stabilizing interaction force, for each chosen robot environment and for the chosen contact control algorithm, values of normal force $\mathbf{F_n}(t)$ and error of normal force ($\mathbf{\Delta F_n} = \mathbf{F_n} - \mathbf{F_{np}}$, where F_{np} is desired normal force) in time instants $(t), (t-1), (t-2)$ and $(t-3)$ are measured, calculated, and stored as special input patterns for training of neural network. Generally, six contact force sensor components can be gathered during the task realization, but we will focus our attention only to normal force, as one of the most interesting components which is sufficient to classify the unknown environment characteristics. It is assumed that normal force component can be obtained from force sensor, because in considering machining operations, normal and tangential directions of force components are defined. The choice of error of normal force in previous time instants as input variables is determined because the nonlinear mapping depends on the previous inputs and outputs of the system. On the other side, the acquisition process must be accomplished using various robot environments, starting with the environment with a low level of system characteristic

TABLE I
INPUTS AND TARGET OUTPUTS OF NEURAL CLASSFIER

Input data for classifier	Target outputs for classifier
$F_n(t)$	Styrofoam 0.00
$\Delta F_n(t)$	Silicon 0.25
$\Delta F_n(t-1)$	Rubber 0.50
$\Delta F_n(t-2)$	Plastic 0.75
$\Delta F_n(t-3)$	Steel 1.00

(for example, with a low level of environment stiffness) and ending with the environment with a high level of system characteristic (with high level of environment stiffness). This approach represents good foundation in order for encircling the wide range of unknown robot environment characteristics. It is important to note that the main idea is classi cation of environment type, not environment parameter identi cation. Hence, using this approach, it is possible in similar way to include one extension in classi cation process which is connected for recognition of environment types with different structure of environment models.

After that, during the extensive off-line training process, neural network receives a set of input-output patterns, where input variables form a previously collected set of force data (in each learning iteration normal force $\mathbf{F_n}(t)$ in time instants $(t), (t-1), (t-2)$ and $(t-3)$ or normal force $\mathbf{F_n}(t)$ and error of normal force $\mathbf{\Delta F_n}$ in time instants $(t), (t-1), (t-2)$ and $(t-3)$ during force transient process). As desired output, neural network has a value between zero and unity which exactly de nes the type of training robot environment. The aim of connectionist training is that the real output of neural network for given inputs can be exact or very close to the desired output value determined for appropriate training robot environment. In our example, training of neural network is accomplished with ve different working environments (similar to [23]). The input variables and target outputs for neural classi er are shown in Table I.

The target outputs of neural classi ers for various environments are chosen by equidistant numerical values between zero and 1, in order to drive the process of linear interpolation in nal procedure of on-line control parameters selection.

It is generally assumed that training examples (training patterns) represent common and most used robot-environment con guration models. Hence, the success of the classi cation is determined by the ªrichnessº of the training patterns.

B. Learning Process for Neural Classi erÐThe First Phase

After acquisition process in the rst phase, it is very important to choose an ef cient learning algorithm for off-line training process in order to assure the best convergence of learning process. Hence, in this paper, learning algorithms for adjusting the network weights based on application of recursive least-square (RLS) method with gradient approximation [30] is considered. Using these methods with time-varying learning rates yields bene ts for learning speed and generalization as compared to those available with the standard back propagation algorithm.

The main relations in process of training for forward-pass in four-layer networks are described according to the following expressions:

$$\mathbf{s^2}(k) = \mathbf{W^{12}}(k)^T \mathbf{i^1}(k) \tag{17}$$

$$o_a^2(k) = 1/\left(1+\exp\left(-s_a^2(k)\right)\right) \qquad a = 1, \ldots, L_1 \quad o_0^2(k) = 1 \tag{18}$$

$$\mathbf{s^3}(k) = \mathbf{W^{23}}(k)^T \mathbf{o^2}(k) \tag{19}$$

$$o_b^3(k) = 1/\left(1+\exp\left(-s_b^3(k)\right)\right) \qquad b = 1, \ldots, L_2 \quad o_0^3(k) = 1 \tag{20}$$

$$\mathbf{s^4}(k) = \mathbf{W^{34}}(k)^T \mathbf{o^3}(k) \tag{21}$$

$$y_c(k) = s_c^4(k) \quad c = 1. \tag{22}$$

where $\mathbf{s^2}(k)$, $\mathbf{s^3}(k)$, and $\mathbf{s^4}(k)$ are the output vectors for linear parts of layers in time instant k; $\mathbf{o^2}(k)$ and $\mathbf{o^3}(k)$ are the output vectors of the hidden layers; $\mathbf{W^{12}} = [w_{m\times L1}^{12}]$, $\mathbf{W^{23}} = [w_{L1+1\times L2}^{23}]$, and $\mathbf{W^{34}} = [w_{L2+1\times n}^{34}]$ are the weighting factors of the layers; $\mathbf{i^1}(k)$ are inputs to the network (force data $m = 5+$ bias member $= 6$); and $y(k)$ is output of network in time instant k.

The general idea is that multilayer perceptrons can be observed as a set of sequential linear decomposed subsystems which are connected by nonlinear connections. As it is well-known, recursive estimators for linear deterministic systems show faster convergence properties than gradient estimators (BP algorithm). Hence, the aim of estimation is to de ne the optimal values for matrices $\mathbf{W^{12}}$, $\mathbf{W^{23}}$, and $\mathbf{W^{34}}$. In application of this method, for speci cation of desired values for linear parts of layers $\mathbf{s^{id}}(i = 2, 3)$ gradient approximation is used. The basic equations which describe the proposed learning rules based on RLS gradient approximation method are given according to the following formulae [30]:

for $h = 4$

$$s_c^{hd}(k) = y_c^d(k) \quad c = 1 \tag{23}$$

for $h = 3, 2 \quad g = 4, 3$

$$s_t^{hd}(k) = s_t^h(k) + \alpha_t \delta^{\mathbf{g}}(k)^T \mathbf{W^{hg}}(k)^T \mathbf{o^h}(k)\left[1 - o_t^h(k)\right] \qquad t = 1, \ldots, L_2 \quad \text{or} \quad t = 1, \ldots, L_1 \tag{24}$$

for $h = 3, 2, 1$

$$\mathbf{C^h}(k) = \mathbf{C^h}(k-1) - \frac{\mathbf{C^h}(k-1)\mathbf{o^h}(k)\mathbf{o^h}(k)^T\mathbf{C^h}(k-1)}{1 + \mathbf{o^h}(k)^T\mathbf{C^h}(k-1)\mathbf{o^h}(k)} \tag{25}$$

for $h = 3, 2, 1 \quad g = 4, 3, 2$

$$\mathbf{W^{hg}}(k) = \mathbf{W^{hg}}(k-1) + \mathbf{C^h}(k)\mathbf{o^h}(k)[\mathbf{s^{gd}}(k) - \mathbf{W^{hg}}(k-1)^T\mathbf{o^h}(k)]^T \tag{26}$$

for $h = 4, 3 \quad g = 3, 2$

$$\delta_t^h(k) = s_t^{hd}(k) - \sum_j w_{tj}^{gh}(k) o_j^g(k) \qquad t = 1, \ldots, n \quad j = 1, \ldots, L_2 + 1 \quad \text{or} \quad t = 1, \ldots, L_2 \quad j = 1, \ldots, L_1 + 1 \tag{27}$$

where α_t is the appropriate learning rate; $\mathbf{C^h}$ is the appropriate covariance matrix; and y_c^d is the desired output of the network.

C. Improvement of Learning Process for Neural Classi er by Evolutionary ApproachÐThe First Phase

One of the main design parameters related to network topology is the number of neurons on each hidden layer. In order to avoid heuristic selection of number of neurons based on long-time simulation experiments, a new approach to network topology selection based on evolutionary neural networks (ENN) is proposed. ENN represents a combination of the connectionist approach and evolutionary search procedures like genetic algorithms (GA) [31]. GA has been proposed to tackle different kind of problems in neural network research area. One type of problems is determination of connection weights, and overcoming local minimum (referred as training of network) [32]. Another applications of GA is searching the topology of neural network (referred as designing of a network) [33].

For the proposed approach, i.e., for the problem of determination of network topology, the neural network and the genetic algorithms works together in a collaborative fashion. The rst step in application of genetic algorithms is to set a generation of initial population of possible network topologies in a random way. In this case, it is a previously determined number of pairs which de ne the number of neurons in the rst and the second hidden layer. For the second step, it is necessary to convert the numeric values of number of neurons in hidden layers to a binary representation (two 8-bit strings). The crucial point in GA algorithm is the choice of tness function. Our aim is to choose a topology of neural network with the minimum approximation error, i.e., we can use the value of well-known mean square error criterion at the end of previously de ned learning epoch as a quality information for search

$$E^p(k) = 0.5 \sum_{i=1}^{k} |\hat{y}^p(k) - y^p(k)|^2 \tag{28}$$

where $\hat{y}^p(k)$ is the target output of neural network in learning epoch k; $y^p(k)$ is the real value of network output in learning epoch k; $E^p(k)$ is the value of the mean-square criterion for one input±output pattern p $(p \in P)$ in learning epoch k; and P is the set of input±output pairs.

Now, after neural network training, all strings in initial population have their own tness function. Hence, according to the basic idea of ªsurvival of the ttestº, the *selection* genetic operator is applied. There are many selection procedures, but in this case the roulette wheel selection [31] that chooses individuals for reproduction according to their tness function values is chosen. Due to the experience in training of multilayer perceptrons, one limitation in selection procedure is included, i.e., only pairs of strings where number of neurons in the rst hidden layer is greater than the number of neurons in the second hidden layer are ready for reproduction purposes. In order to improve the search process, the following two genetic operators (*crossover* and *mutation*) are applied with some limitations. *Uniform crossover*, which swaps each column in chromosome representation having the same probability is chosen. In order to avoid great changes in numerical representation of the proposed problem and the proper nature of the search problem, the second operator mutation is limited only to ve lower bits of each string. Now the complete new population is generated, which is converted into numerical representation after decoding process, and which is ready for evaluation of its tness function through neural network training process with a new network topology. The process is stopped when the desired value of tness function is achieved.

Using the proposed approach and choosing the optimal node size in hidden layers of network, it is possible to assure a fast learning process and better classi cation properties of the neural classi er.

IV. ON-LINE COMPLIANCE CONTROL ALGORITHMS FOR CONTACT TASKS WITH ENVIRONMENT CLASSIFICATIONÐT HE SECOND PHASE

Based on the rst phase, related to the acquisition process and off-line training process of neural classi er, it is possible to determine the whole structure of compliance control algorithms including the xed neural classi er.

In the rst example considered, the following stiffness model of robot environment is chosen for the control algorithm based on the stabilization of the robot motion with a preset quality of transient process

$$\mathbf{F} = \mathbf{K}'(\mathbf{x} - \mathbf{x_0}). \tag{29}$$

After the off-line training process with different working environments (different environment stiffness), neural classi er with xed weighting factors is included in on-line version of control algorithm (11) to produce some value y at the output of network, between zero and 1, based on on-line force inputs de ned in the previous section

$$\tau = \mathbf{H}(\mathbf{q})[\ddot{\mathbf{q}}_\mathbf{p} - \hat{\mathbf{KP}}\eta - \hat{\mathbf{KD}}\dot{\eta}] + \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{J}^T(\mathbf{q})\mathbf{F} \tag{30}$$

$$\hat{\mathbf{KP}} = f_{kp}(y) \tag{31}$$

$$\hat{\mathbf{KD}} = f_{kd}(y) \tag{32}$$

where f_{kp} and f_{kd} are linear interpolation functions for positional and velocity feedback gains; and y is the output of the neural classi er.

If we adopt as a performance criterion the same force steady-state process for all different robot environments, then we can *a priori* choose, using algebraic stability conditions, the set of PD local gains for previously de ned set of known robot environments (in our case, there are ve different environments) which will satisfy this requirement. Hence, in the case of unknown environment type, the information from neural classi er output can be ef ciently utilized for calculation of necessary PD local gains by linear interpolation procedures. It is also assumed that output of neural network for the given environment varies in small ranges. In this way, local PD gains are relatively xed during the operations. They are chosen for preset stability conditions for each environment type. Fig. 1 shows the overall structure of the proposed algorithm.

In the second example, for the control algorithm based on stabilization of the interaction force with a preset quality

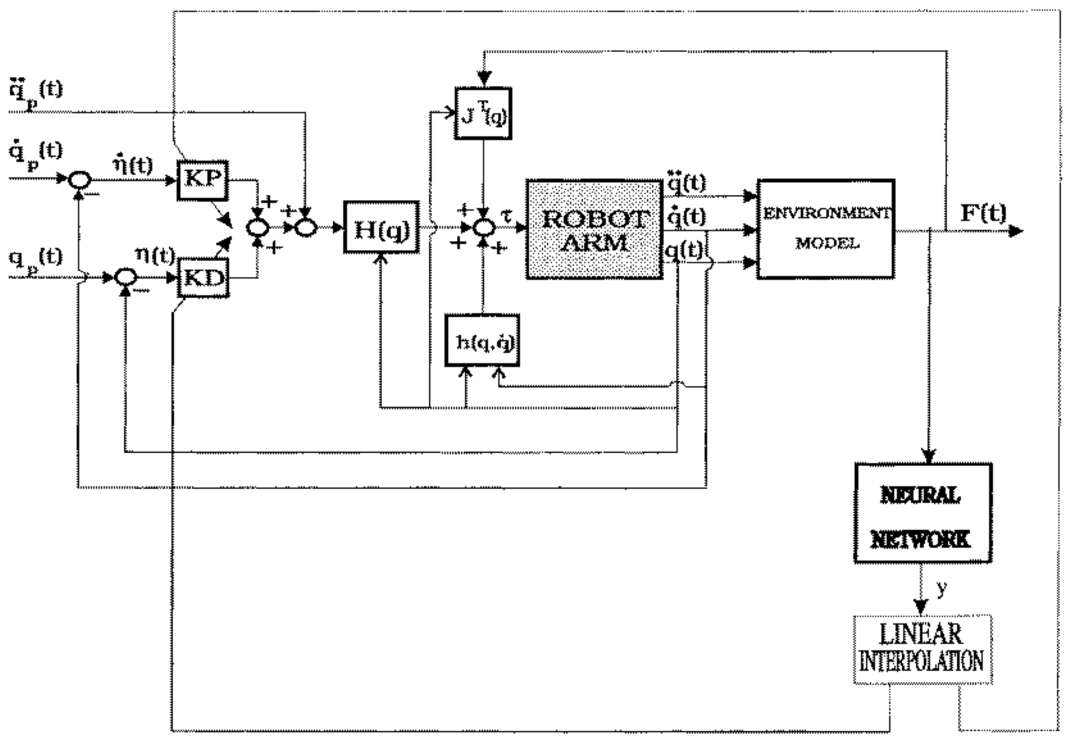


Fig. 1. Scheme of control law stabilizing robot motion with neural classi er.

of transient process, the general impedance model of robot environment is chosen

$$\mathbf{F} = \mathbf{M}'\boldsymbol{\Delta}\ddot{\mathbf{x}} + \mathbf{B}'\boldsymbol{\Delta}\dot{\mathbf{x}} + \mathbf{K}'\boldsymbol{\Delta}\mathbf{x}. \tag{33}$$

Hence, after the off-line training process, on-line version of compliance control algorithm with neural classi er with xed weighting factors based on on-line force and force errors inputs is given by the following relations for speci ed environment model (7)

$$\begin{aligned}\tau = &-\mathbf{H}(\mathbf{q})\hat{\mathbf{M}}'^{-1}(\mathbf{q})[\hat{\mathbf{B}}'\dot{\mathbf{q}} + \hat{\mathbf{K}}'\mathbf{q}] + \mathbf{h}(\mathbf{q},\dot{\mathbf{q}})\\ &+ (\mathbf{J}^T(\mathbf{q}) - \mathbf{H}(\mathbf{q})\hat{\mathbf{M}}'^{-1})\left\{\mathbf{F_p} - \int_{t_0}^{t}\left[\mathbf{KFP}\mu(\omega)\right.\right.\\ &\left.\left.+\ \mathbf{KFI}\int_{t_0}^{t}\mu(\omega)\,dt\right]d\omega\right\}\end{aligned} \tag{34}$$

$$\hat{\mathbf{M}}' = f_{\mathbf{M}'}(y) \tag{35}$$

$$\hat{\mathbf{B}}' = f_{\mathbf{B}'}(y) \tag{36}$$

$$\hat{\mathbf{K}}' = f_{\mathbf{K}'}(y) \tag{37}$$

where $f_{\mathbf{M}'}, f_{\mathbf{B}'}$, and $f_{\mathbf{K}'}$ are linear interpolation functions for parameters of matrices $\mathbf{M}', \mathbf{B}'$, and $\mathbf{K}'$.

According to the similar principle, the same condition for control law and all different robot environments is using the same local PI force gains. In our case, parameters of dynamic models of different chosen environments M', B', and K' are stored as an information necessary for calculating the basic control algorithm. In the case of the unknown environment, information from neural classi er output can be ef ciently utilized for calculation of necessary environment parameters M', B', and K' by linear interpolation procedures. Fig. 2 shows the overall structure of the proposed algorithm.

V. Case Study

A. General Reference Task Description

For demonstrating the performance of contact control schemes with neural elements, compliance control implementations are simulated using robot MANUTEC r3 (Fig. 3) [34] and various models of robot environment.

Technological working demands for reference working operation are de ned by the following statements. a) The working tool of the robot was realized in the form of a rotational-milling tool, performing surface processing in the plane which is parallel to the X-Y plane. b) Tool trajectory is 100 mm-long. c) The task of the robot is to carry out the machining process of the work surface along the prescribed trajectory with a desired contact force $F_N^0 = 5$ N and a prescribed velocity of 25 mm/s.

The following initial conditions were used in the simulation: the robot gripper starts with zero initial velocity. The settling time of the desired contact force is given as $t = 0.5$ s.

For the rst example of stabilizing motion control algorithm, the stiffness model of environment is adopted, while in the case of stabilizing interaction force algorithm, a general model of impedance is chosen. The parameters of the environment model in the form of diagonal members of appropriate matrices for all different chosen environments and for both control algorithms are given in Tables II±IV.

B. The Stabilizing Motion Control AlgorithmÐChoice of Local Gains for Network Training

To investigate the effect of different chosen PD local gains for stabilizing motion control law and various robot environments, some simulation experiments were conducted

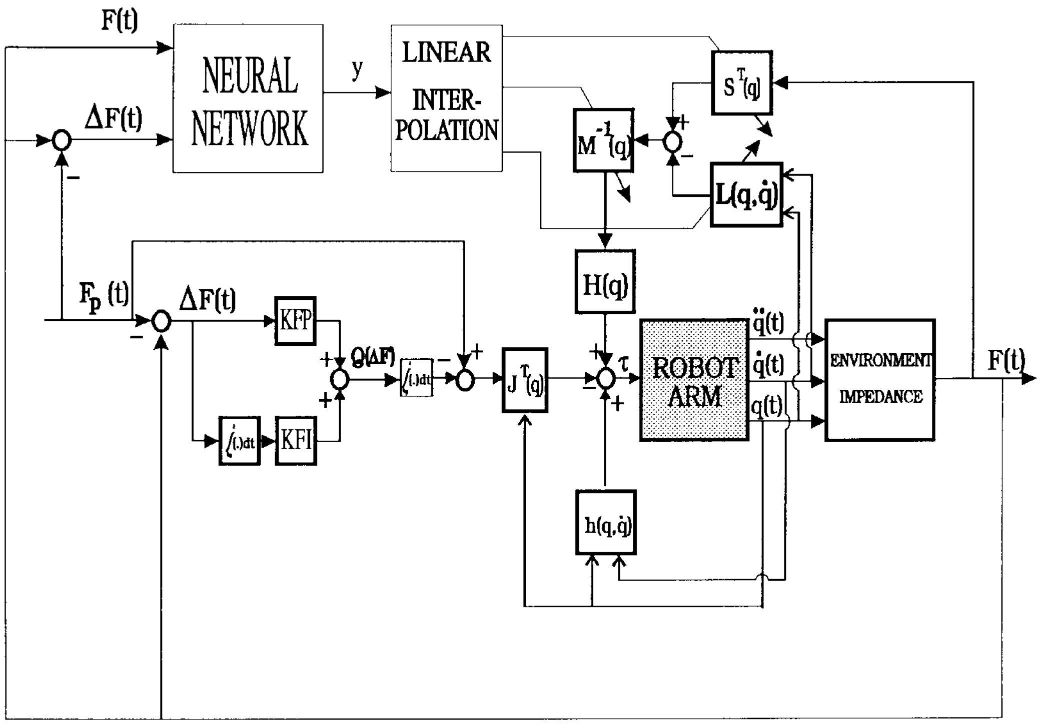


Fig. 2. Scheme of control law stabilizing interaction force with neural classi er.

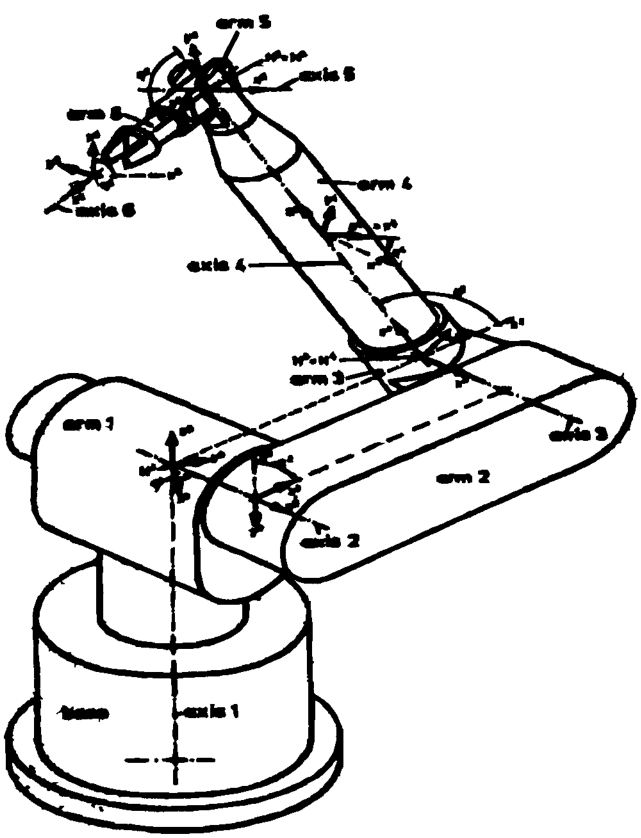


Fig. 3. Industrial robot MANUTEC r3.

TABLE II
THE STIFFNESS PARAMETERS OF ROBOT ENVIRONMENT MODELS

Environment	K'_{11}	K'_{22}	K'_{33}	K'_{44}	K'_{55}	K'_{66}
Styrofoam	660.	6600.	660.	0.0007	0.0007	0.0007
Silicon	2000.	20000.	2000.	0.002	0.002	0.002
Rubber	3300.	33000	3300.	0.003	0.003	0.003
Plastic	6000.	60000.	6000.	0.006	0.006	0.006
Steel	24000.	240000.	24000.	0.02	0.02	0.02

TABLE III
THE DAMPING PARAMETERS OF ROBOT ENVIRONMENT MODELS

Environment	B'_{11}	B'_{22}	B'_{33}	B'_{44}	B'_{55}	B'_{66}
Styrofoam	69.	277.	69.	0.007	0.007	0.007
Silicon	210.	839.	210.	0.02	0.02	0.02
Rubber	346.	1385	346.	0.035	0.035	0.035
Plastic	629.	2517.	629.	0.06	0.06	0.06
Steel	2638.	10704.	2638.	0.27	0.27	0.27

TABLE IV
THE INERTIA PARAMETERS OF ROBOT ENVIRONMENT MODELS

Environment	M'_{11}	M'_{22}	M'_{33}	M'_{44}	M'_{55}	M'_{66}
Styrofoam	1.88	6.7	1.88	0.007	0.007	0.007
Silicon	5.7	20.3	5.7	0.02	0.02	0.02
Rubber	9.4	33.5	9.4	0.03	0.03	0.03
Plastic	17.	60.9	17.	0.06	0.06	0.06
Steel	68.15	243.41	68.15	0.2	0.2	0.2

with two set of PD local gains. One set of gains is chosen for low stability degree, while the other is chosen for high stability degree. Internal coordinates error and force error in this case are presented in Figs. 4±7. The feedback gains for stabilizing control laws with required quality of position transient response have been chosen in the form of diagonal matrices for the rst set of gains (low-stability degree)

$$\mathbf{KP} = \mathrm{diag}\{kp^i\}, \quad i = 1, \ldots, n, \quad kp^i = 1. \tag{38}$$

TABLE V
PD LOCAL GAINS FOR SATISFACTION OF PERFORMANCE CRITERION

Environment	KP	KD
Styrofoam	100.	20.
Silicon	484.	44.
Rubber	900.	60.
Plastic	2209.	94.
Steel	10000.	200.

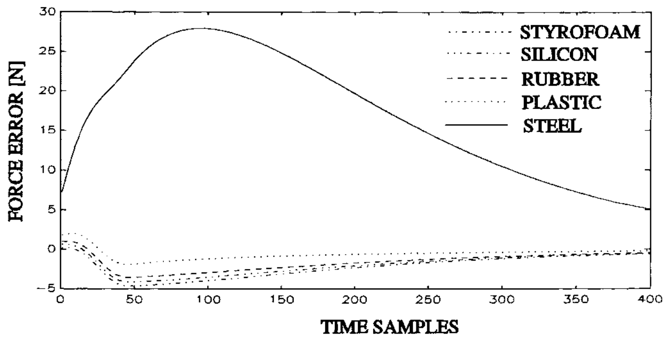


Fig. 4. Force error for control law stabilizing robot motion with rst set of feedback gains.

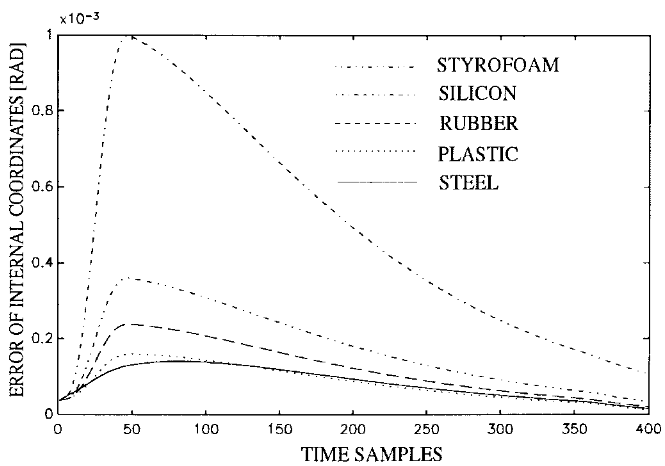


Fig. 5. Internal coordinate error for control law stabilizing robot motion with rst set of feedback gains.

$$\mathbf{KD} = \mathrm{diag}\{kd^i\}, \quad i = 1, \ldots, n, \quad kd^i = 2 \tag{39}$$

and for the second set of gains (high-stability degree)

$$\mathbf{KP} = \mathrm{diag}\{kp^i\}, \quad i = 1, \ldots, n, \quad kp^i = 100. \tag{40}$$

$$\mathbf{KD} = \mathrm{diag}\{kd^i\}, \quad i = 1, \ldots, n, \quad kd^i = 20. \tag{41}$$

We can observe the dependance of transient processes on the type of robot environment and chosen set of PD-gains. In the case of high gain feedback, there are differences only at the beginning of transient process. In the case when force noise is included, different force steady-state processes for different working environments are presented (Fig. 8).

Hence, for our reference case, the following performance criterion is chosen: The sum of force error during the task cannot be greater than 11. To achieve this performance criterion, different local PD gains for different environments must be synthesized based on simulation experiments (Table V).

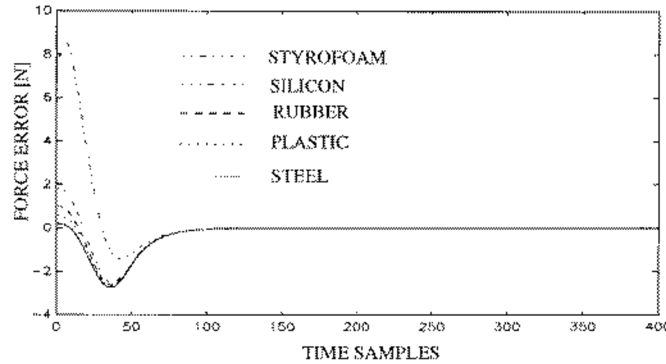


Fig. 6. Force error for control law stabilizing robot motion with second set of feedback gains.

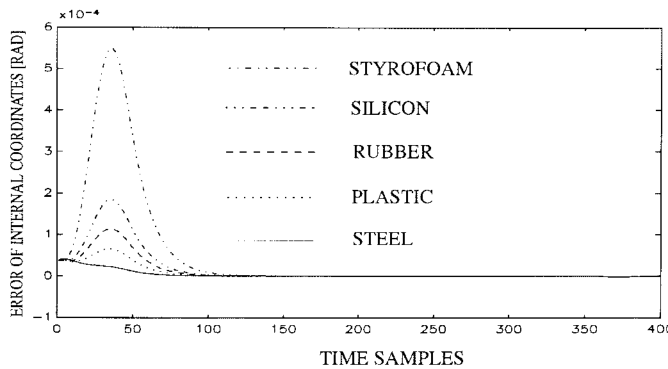


Fig. 7. Internal coordinate error for control law stabilizing robot motion with second set of feedback gains.

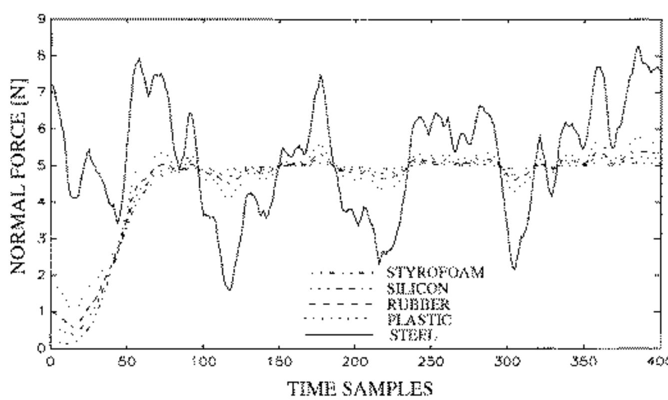


Fig. 8. Steady-state process for normal force control law stabilizing robot motion with second set of feedback gains.

C. The Stabilizing Motion Control AlgorithmÐOff-Line Training Process

In the process of neural network training, 500 force training patterns are used (for all ve different environments with the same control law and different previously chosen PD local gains there are 100 input±output patterns). After intensive simulation experiments, the following network topology is chosen: 5-62-41-1 (represents a number of neurons in network layers). According to the proposed learning rules, initial values of covariance matrix are $\mathbf{C_{init}} = 100\,000 I$ and gradient factor $\alpha_t = 0.02$. The training results are shown in Figs. 9 and 10, which represent square criterion during training and the comparison of desired and real outputs of network after training.

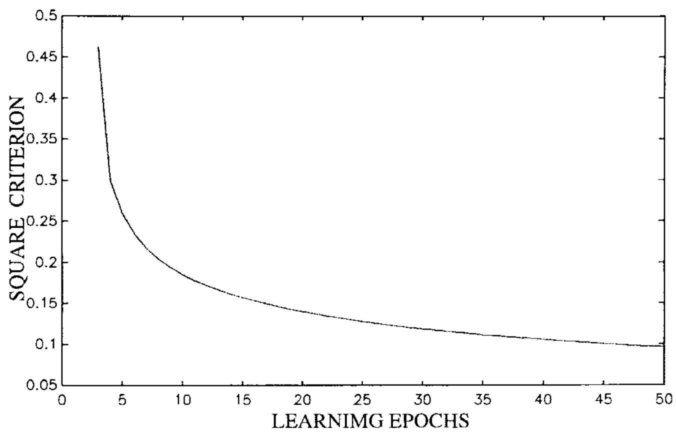


Fig. 9. Square criterion during learning epochs.

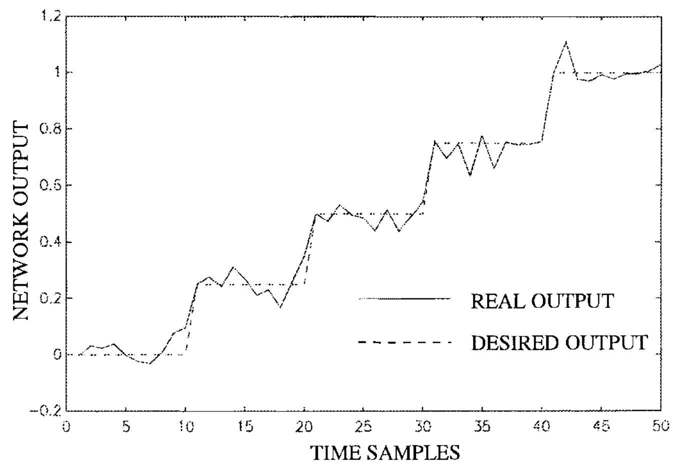


Fig. 10. Comparison of desired and real output of network.

D. The Stabilizing Motion Control Algorithm—On-Line Generalization Process

In the generalization test, the “learned” neural classifier with fixed weighting factors is included in control algorithm for the recognition of unknown robot environment. In this case, the robot environment with dominant stiffness $\mathbf{K}'_{22}$ is selected. The goal is to achieve the same quality of force steady-state process. The neural classifier based on input force data generates output of network having numeric values of 0.62. This information is necessary for calculating the local PD gains by linear interpolation that can satisfy the desired performance criterion (Fig. 11). For comparison, the example of application of nonlearning control laws with inexact (user assumed) information of environment stiffness is given in Fig. 12. It is clear that in the case when there are no exact information about robot environment, the quality of performance is very poor. Hence, inclusion of neural classifier is very important in order to improve the capabilities of control algorithms in working environment with significant level of uncertainties.

E. The Stabilizing Force Control Algorithm—Influence of Different Working Environments

In the second case, for the application of stabilizing force interaction control algorithm, the performance criterion based on selection of the same force PI gains is chosen. These PI force gains are synthesized using the same system frequencies

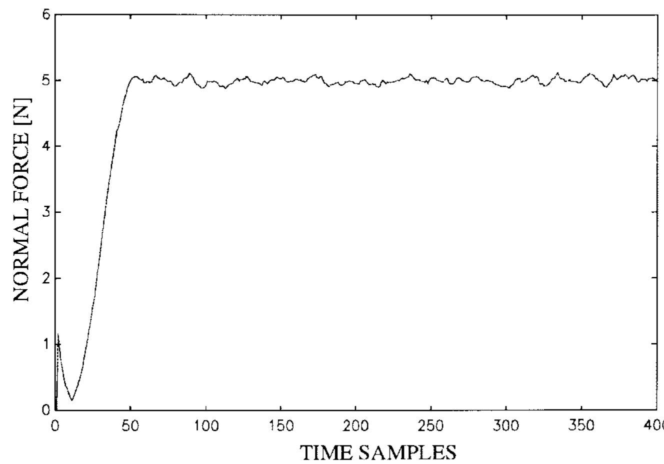


Fig. 11. Normal force with neural classifier—stabilizing robot motion.

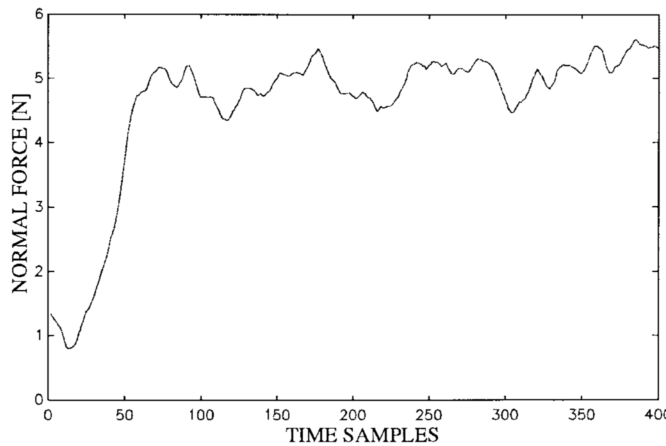


Fig. 12. Normal force without neural classifier—stabilizing robot motion.

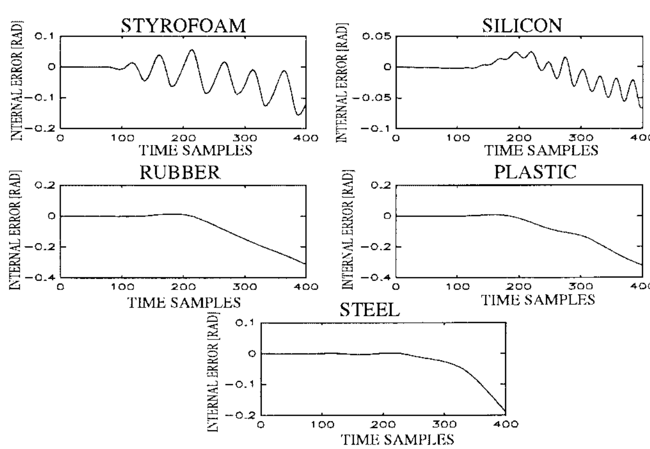


Fig. 13. Internal coordinate error for stabilizing interaction force control algorithm.

for all different working environments ($\omega_n = 2$ Hz). The transient processes of internal coordinates error and force error are given in Figs. 13 and 14. We can notice the influence of different working environments.

F. The Stabilizing Force Control Algorithm—Off-Line Training Process with Evolutionary Approach

In the phase of connectionist training, the efficient genetic algorithm is used in order to select the optimal topology of

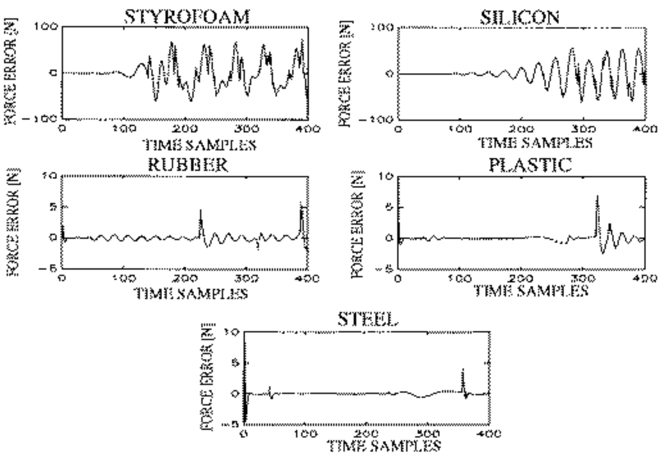


Fig. 14. Force error for stabilizing interaction force control algorithm.

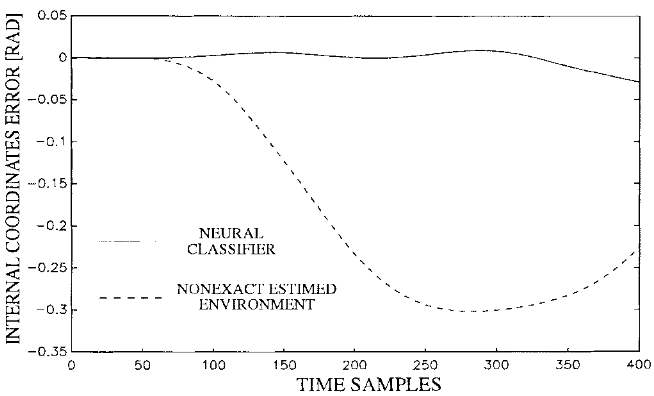


Fig. 15. Internal coordinate errorÐcomparison with and without neural classi er.

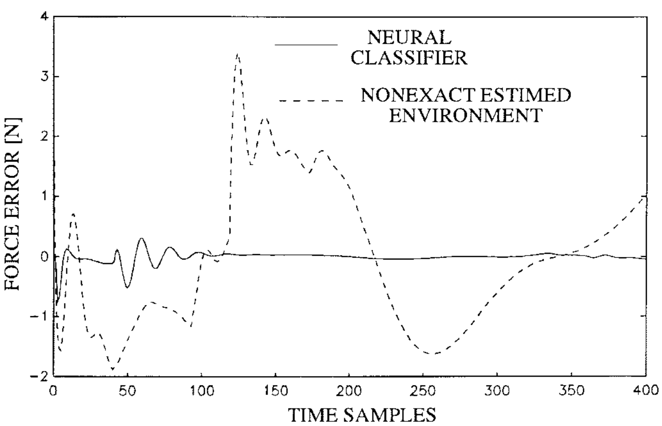


Fig. 16. Force errorÐcomparison with and without neural classi er.

neural network. The initial population of 50 pairs of possible topology solutions is given and three successive generations are simulated. The following genetic parameters are chosen: crossover probability $pcros = 0.3$ and mutation probability $pmut = 0.03$. Using this procedure, the following optimal network topology is selected: 6-32-21-1.

Using adopted network topology and the same learning rules and learning parameters, training process is achieved with stored weighting factors.

G. The Stabilizing Force Control AlgorithmÐOff-Line Training Process with Evolutionary Approach

In similar way as in the previous case, generalization test with unknown environments (dominant stiffness $\mathbf{K}'_{22}$ and output of neural classi er 0.70) using approaches with and without neural classi er is performed. The results are given in Figs. 15 and 16. The conclusions are the same as in the case of stabilizing robot motion control algorithms, but in this case the in uence of unknown environment is very signi cant, because of the implicit inclusion of environment parameters in the control law. Hence, the neural classi er signi cantly improves the system performance.

VI. Conclusion

This paper presents a new method for selecting the appropriate compliance control parameters for robot machining tasks based on connectionist classi cation of unknown dynamic environment type. The method classi es the type of environment by using in the rst phase, acquisition process of force sensor data and off-line training process by multilayer perceptrons. The off-line process is signi cantly improved by special genetic algorithms with some limitations on genetic operators in order to choose the optimal number of neurons in hidden layers for fast learning. In the second on-line phase, based on inclusion of neural classi er, the compliance control algorithm determines the control parameters based on network output and previously determined set of environment model characteristics by linear interpolation procedure. The important feature is that the process of pattern association can work in an on-line mode as a part of selected compliance control algorithm. Simulation experiments show that well-trained neural classi er in on-line mode can identify characteristics of robot environment in various machining tasks.

Further work includes one comprehensive connectionist approach for compensation of the in uence of uncertainties of the robot and robot environment, consideration of connectionist identi cation of nonlinear environment models, and the development of other similar approaches for explicit estimation of robot environment by neural networks.

References

[1] M. Vukobratovic and A. Tuneski, ªContact control concepts in manipulation robotics, an overiew,º *IEEE Trans. Ind. Electron.*, vol. 41, pp. 12±24, Feb. 1994.
[2] M. H. Raibert and J. J. Craig, ªHybrid position/force control of manipulators,º *ASME J. Dyn. Syst., Meas., Contr.*, vol. 103, pp. 126±133, June 1981.
[3] T. M. Mason, ªCompliance and force control for computer controlled manipulators,º *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-11, no. 6, pp. 418±432, 1981.
[4] J. Colgate and N. Hogan, ªRobust control of dynamically interacting systems,º *Int. J. Contr.*, vol. 48, pp. 65±88, 1988.
[5] C. An and J. Hollerbach, ªKinematic stability issues on force control of manipulators,º in *Proc. IEEE Int. Conf. Robotics Automation*, Raleigh, NC, May 1987, pp. 847±903.
[6] N. H. McClamroch and D. Wang, ªFeedback stabilization and tracking in constrained robots,º *IEEE Trans. Automat. Contr.*, vol. 33, no. 5, pp. 419±426, 1988.
[7] S. Eppinger and W. Seering, ªIntroduction to dynamic models for robot force control,º *IEEE Contr. Syst. Mag.*, vol. 7, no. 2, pp. 48±52, 1987.

[8] A. DeLuca and C. Manes, ªOn the modeling of robots in contact with dynamic environments,º in *Proc. 5th Int. Conf. Advanced Robotics*, Pisa, Italy, 1991, pp. 568±574.

[9] M. Vukobratovic and Y. Ekalo, ªNew approach to the control of manipulation robots interacting with dynamic environment,º *Robotica*, vol. 14, pp. 31±39, 1996.

[10] M. Vukobratovic, ªThe role of environment dynamics in contact force control of manipulation robots,º submitted for publication.

[11] M. Vukobratovic and D. Katic, ªRobot control structures for high-quality learning in exible-manufacturing tasks,º in *Logistic and Learning for Quality Management and Manufacturing*, B. Socek, Ed. New York: Wiley, 1994.

[12] H. Asada, ªTeaching and learning of compliance using neural nets: Representation and generalization of nonlinear compliance,º in *Proc. IEEE Int. Conf. Robotics Automation*, Cincinnati, OH, May 1990, pp. 1237±1244.

[13] H. Asada and S. Liu, ªTransfer of human skills to neural net robot controllers,º in *Proc. IEEE Int. Conf. Robotics Automation*, Sacramento, CA, Apr. 1991, pp. 2442±2448.

[14] V. Gullapali, R. Grupen, and A. Barto, ªLearning reactive admittance control,º in *Proc. IEEE Int. Conf. Robotics Automation*, Nice, France, May 1992, pp. 1475±1481.

[15] V. Gullapali, A. Barto, and R. Grupen, ªLearning admittance mappings for force-guided assembly,º in *Proc. IEEE Int. Conf. Robotics Automation*, San Diego, CA, May 1994, pp. 2633±2638.

[16] D. Jeon and M. Tomizuka, ªLearning hybrid force and position control of robot manipulators,º in *Proc. IEEE Int. Conf. Robotics Automation*, Nice, France, May 1992, pp. 1455±1460.

[17] S. Arimoto and T. Naniwa, ªLearning control for robot tasks under geometric endpoint constraints,º in *Proc. IEEE Int. Conf. Robotics Automation*, Nice, France, May 1992, pp. 1914±1919.

[18] T. Fukuda, T. Shibata, M. Tokita, and T. Mitsuoka, ªAdaptation and learning by neural network for robotic manipulator,º in *Proc. IMACS Int. Symp. Mathematical Intelligent Models in System Simulation*, Brussels, Belgium, Sept. 1990.

[19] J. M. Tao and J. Y. S. Luh, ªApplication of neural network with real-time training to robust position/force control of multiple robots,º in *Proc. IEEE Int. Conf. Robotics Automation*, Atlanta, GA, May 1993, pp. 142±148.

[20] S. Jung and T. C. Hsia, ªOn neural network application to robust impedance control of robot manipulators,º in *Proc. IEEE Int. Conf. Robotics Automation*, Nagoya, Japan, May 1995, pp. 869±874.

[21] D. Seslija and M. Vukobratovic, ªEnvironment parameters identi cation for the control of robotized machining,º in *Proc. 1st ECPD Int. Conf. Advanced Robotics Intelligent Automation*, Athens, Greece, Sept. 1995, pp. 632±637.

[22] S. T. Venkataraman, S. Gullati, J. Barhen, and N. Toomarian, ªA neural network based identi cation of environment models for compliant control of space robots,º *IEEE Trans. Robot. Automat.*, vol. 9, pp. 685±697, Oct. 1993.

[23] D. H. Cha and H. S. Cho, ªA neurofuzzy model-based compliance controller with application to a telerobot system,º *Contr. Eng. Practice*, vol. 4, pp. 319±330, May 1996.

[24] M. Tarokh and S. Bailey, ªForce tracking with unknown environment parameters using adaptive fuzzy controllers,º in *Proc. IEEE Int. Conf. Robotics Automation*, Minneapolis, MN, Apr. 1996, pp. 270±275.

[25] M. Vukobratovic and D. Katic, ªRobust stabilizing position/force control of robots interacting with dynamic environment by learning connectionist structures,º *Automatica*, vol. 33, pp. 1733±1739, Dec. 1996.

[26] D. Katic and M. Vukobratovic, ªThe application of connectionist structures for learning impedance control in robotic contact tasks,º *Appl. Intell.*, vol. 7, pp. 315±326, Nov. 1997.

[27] H.-L. Pei, Q.-J. Zhou, and T. P. Leung, ªA neural network robot force controller,º in *Proc. 1992 IEEE/RSJ Int. Conf. Intelligent Robots Systems*, Raleigh, NC, July 1992, pp. 1974±1979.

[28] N. Hogan, ªImpedance control: An approach to manipulation, part 1Ðtheory, part 2Ðimplementation, part 3Ðapplication,º *ASME J. Dyn. Syst., Meas., Contr.*, vol. 107, pp. 1±24, 1985.

[29] D. Stokic and M. Vukobratovic, ªIs dynamic control needed for robots interacting with dynamic environment?,º in *Proc. 2nd ECPD Int. Conf. Advanced Robotics, Intelligent Automation Active Systems*, Vienna, Austria, Sept. 1996.

[30] D. Katic and M. Vukobratovic, ªDecomposed connectionist architecture for fast and robust learning of robot dynamics,º in *Proc. IEEE Int. Conf. Robotics Automation*, Nice, France, May 1992, pp. 2064±2069.

[31] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Reading, MA: Addison-Wesley, 1989.

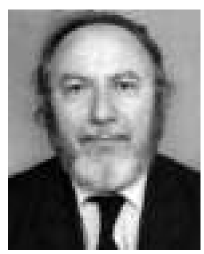
[32] D. J. Montana and L. Davis, ªTraining feedforward neural networks using genetic algorithms,º in *Proc. 11th Joint Conf. Arti cial Intelligence*. San Mateo, CA: Morgan Kaufmann, 1989, pp. 762±767.

[33] Q. Wang, H. H. Shao, and Z. J. Zhang, ªGenetic evolved neural network for process modeling and optimization,º in *Proc. 13th Triennial IFAC World Congr.*, San Francisco, CA, July 1996, vol. M, pp. 283±288.

[34] S. Turk and M. Otter, ªDas DFVLR Modell Nr 1 des industrieroboters Manutec r3,º *Robotersysteme*, no. 3, pp. 101±106, 1987 (in German).

Dusko Katic received the B.S. and M.S. degrees in mechanical engineering and the Ph.D. degree in electrical engineering from the University of Belgrade, Belgrade, Yugoslavia, in 1982, 1987, and 1994, respectively.

Since 1983, he has been a Researcher with the Robotics Centre, Mihajlo Pupin Institute, Belgrade, where he is involved in the modeling and design of advanced control systems for manipulation robots and other large-scale dynamic systems. He is presently Higher Counselor at the Robotics Centre and a Member of the Scienti c Council. He is the author or coauthor of about 50 scienti c papers in the eld of robotics which have been published in international journals, monographs, and proceedings of international conferences and congresses. His current scienti c interests include theoretic and experimental work in the areas of intelligent autonomous systems, intelligent control, robotics, adaptive and learning systems, neural networks, evolutionary computing, and signal processing.



Miomir Vukobratovic was born in Zrenjanin, Yugoslavia, 1931. He received the B.Sc. and Ph.D. degrees in mechanical engineering from the University of Belgrade, Belgrade, Yugoslavia, in 1957 and 1964, respectively, and the D.Sc. degree from the Institute Mashinovedenya, Soviet (now Russian) Academy of Sciences, Moscow, in 1972.

From 1968 to 1996, he was Head of the Biodynamics Department, then Director of the Laboratory for Robotics and Flexible Automation and Director of the Robotics Laboratory at the Mihailo Pupin Institute, Belgrade. He is presently Advisor for Science for the General Director of the Institute. He is also Visiting Professor teaching graduate courses in robotics at several universities in Yugoslavia and abroad. His main interest is in the development of computer-aided modeling of robotic systems dynamics. His special interest is dynamic nonadaptive and adaptive control of noncontact and contact tasks in manipulation robotics as well as dynamic modeling and control in locomotion robotics. He is author or coauthor of 180 scienti c papers in the eld of robotics published in leading international journals, as well as author or coauthor of more than 280 papers in proceedings of international conferences and congresses. He is author or coauthor of 12 research monographs published in English, Japanese, Russian, Chinese, and Serbian, and two advanced textbooks in robotics. He is the Scienti c Leader of the Serbian National Program in Robotics, as well as the Principal Investigator of three international robotics projects (EC, NSF, and UNIDO).

Dr. Vukobratovic is a present and past member of 50 international committees of IFAC, IFAC/IFIP, and IFToMM symposia and congresses. He is also a member of editorial boards of several leading scienti c journals in robotics, manufacturing, and arti cial intelligence. He is a member and vice president of the Scienti c Society of Serbia, ordinary member of Serbian Academy of Sciences and Arts, foreign member of Soviet (now Russian) Academy of Sciences and ordinary member of International Academy of Engineering. He was the winner of the highest Yugoslav state award, AVNOJ, in 1982, for his outstanding achievements in robotics and technical cybernetics, winner of ª7 Julyº state award of Serbia, 1976, and ª20 Octoberº prize of Belgrade, 1978. He is also, with his coauthors, the winner of the highest scienti c Yugoslav award ªNikola Tesla,º in 1986, for their world-recognized seven volume research monograph series published by Springer-Verlag. He was also the recipient of the Joseph Engelberger Award, given by Robotic Industries Association in 1996.