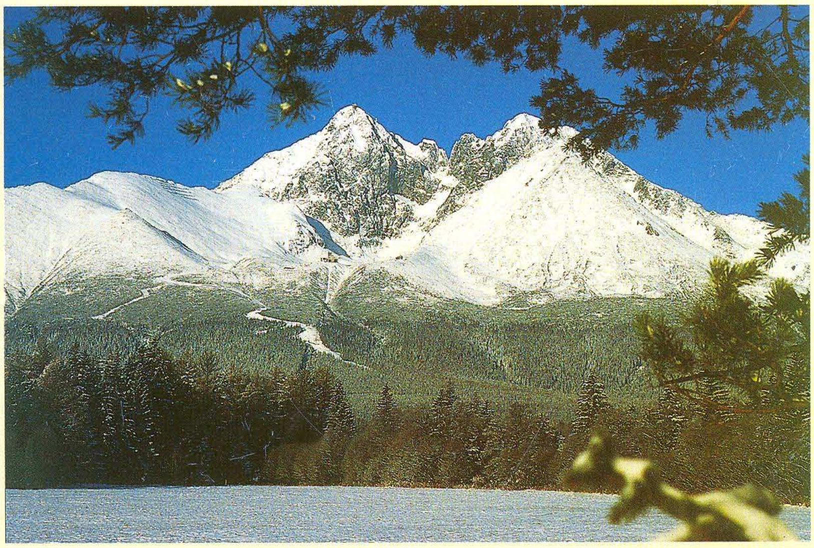
INES´99

1999 IEEE International Conference on Intelligent Engineering Systems

PROCEEDINGS

November 1 - 3, 1999
Poprad, High Tatras, Stará Lesná, Slovakia



Sponsored by
IEEE Industrial Electronics Society
Technical University of Košice, Slovakia
Bánki Donát Polytechnic, Hungary
IEEE Hungary Section

Advanced Control of Robot in Technological Operation

József K. Tar
Bánki Donát Polytechnic
Dept. of Information Technology
H-1081 Budapest, Népszínház utca 8.Hungary
E-mail: jktar@zeus.banki.hu

J.A. Tenreiro Machado
Instituto Superior de Engenharia do Porto
Dep. de Eng. Electrotécnica
Portugal
E-mail: jtm@dee.isep.ipp.pt

Imre J. Rudas
Bánki Donát Polytechnic
Dept. of Information Technology
H-1081 Budapest, Népszínház utca 8.Hungary
E-mail: rudas@zeus.banki.hu

László Horváth
Bánki Donát Polytechnic
Department of Manufacturing Engineering
H-1081 Budapest, Népszínház utca 8.Hungary
E-mail: lhorvath@zeus.banki.hu

Abstract — A novel adaptive robot control tackling the problem of the approximately known robot dynamics and the unknown external dynamic interactions is presented in this paper. By applying *uniform structures* derived from the Euler-Lagrange equations in the most general and formal level it differs from and overcomes the limitations of classical feedforward neural network-based approaches as far as the *a priori unknown* number of the necessary nodes and the scaling ranges of free parameters are concerned. Using a relatively simple structure of reduced number of parameters real time tuning can be carried out in the control. From the point of view of the possible local optimums resulting in improper control the structure here used seems to have the possible least complexity and coupling for a given degree of freedom robot. Several task-independent ancillary procedures also support the control. The method is illustrated via simulation in the case of a 3 active and one passive DOF SCARA arm used for polishing the surface of a bell-shaped work-piece.

I. INTRODUCTION

Strong non-linear dynamic interaction between the robot and its environment in technological processes means a hard problem in robot control not completely solved even in our days in tasks as *grinding* and *polishing*. Such tasks are normally completed by humans in the case of grinding ship propellers, etc. Insufficient and inaccurate knowledge about the dynamic properties of the robot arm itself make this problem even more complicated. The need for developing a universally useful controller for the robot makes it undesirable to include some particular model of the robot-work-piece interaction in the control software. Instead of this a more intelligent control being able to "learn" the main features specific to the technological operation under consideration would be much more expedient. This indefinite or undetermined nature of the task makes it unlikely to find a closed form analytical problem-formulation for which an elegant proof of convergence or bounded error etc. could be found. Even in the case of "rigid body approximation" for the robot arm during the motion normally no satisfactory information is available for real-time system-identification in the classical "complete and analytical" sense [1]. It is more likely to achieve results via combining simple linguistic rules, saturated functions and some learning techniques as normally is done in the case of the traditional Soft Computing (SC) approaches.

The traditional PID controllers also are designed for controlling rather the free traveling motion of the robot arm. In the "Hybrid Position-Force" control it is supposed that the motion of the robot arm is so slow that its dynamics in the Euler-Lagrange equations can completely be ignored and the problem can be formulated as a "static one". Furthermore, it also is supposed that in the operational space complement sub-spaces can be set along which —in the sense of the *exclusive or*— either the translation/rotation, or the force/torque components can be freely prescribed by the control. However, such a restriction is not always valid in the practice: in a polishing task the technology defines the necessary torque for holding back the tool's casing from rotation, but its rotational pose at which it must be kept "fixed" can also be prescribed freely. Both prescriptions concern the same axis. The *traditional application* of components of passive compliance is rather used for compensating the uncertainty of position control in insertion tasks during assembly. The controlled active compliance of the SCARA arms is rather a more intelligent, improved version of the same basic idea of compensation.).

An alternative approach to similar problems is the use of modern, highly parallel Soft Computing (SC) methods completely *abandoning any analytical description* of the system's dynamics. These methods use simple and uniform architectures not strictly tailored to the particular properties of the task to be solved by them. Instead, they contain a considerable number of free parameters by the appropriate variation or "tuning" of which some adaptivity can be achieved. This process frequently is referred to as "learning".

For instance, multilayer perceptrons used for non-linear mapping and realized by feed-forward Artificial Neural Networks (ANNs) contain typical sigmoidal activation functions, connection weights and threshold values to be determined "experimentally".

Fuzzy Controllers (FCs) normally contain typical membership functions having their typical parameters, some fuzzy relations describing the approximately and linguistically already known "rules" to determine the actions to be carried out by the controller.

Fuzzy Controllers (FCs) normally contain typical membership functions having their typical parameters, some fuzzy relations describing the approximately and linguistically already known "rules" to determine the actions to be carried out by the controller.

Normally, *determination of the number of the necessary free parameters or elements* connected to each other in a uniform structure may mean a serious problem. ANNs constructed on the basis of Kolmogorov's approximation theorem may contain a huge parameter-space for tuning because this structure is fit to develope models for the very wide class of continuous functions.

In this context various methods were proposed for finding the proper "size" of the network to be used. A typical class of solutions starts from a quite big initial network and applies dynamic pruning for getting rid of the unimportant nodes and connections [2]. Alternative approaches starting from quite small network in which the number of nodes is increased step by step also correspond to typical solution [3-4]. The appropriate "range" of the parameters in the sigmoids to be tuned cannot be known in advance, too ("scaling").

For learning a great variety of methods ranging from several variants of the *steepest descent method* [5], application of activation functions of tunable shape [6], combination of standard backpropagation with Genetic Algorithms [7], and semi-stochastic Complex Algorithm [8] can be mentioned. In general, if the necessary "size" of the uniform structure, and as a consequence, the number of the parameters to be tuned, is too big, fast and efficient learning algorithm for real-time control can scarcely be constructed. Therefore any reduction in the size of the problem as well as in the possible range of the unknown parameters is a significant advantage.

From *control technical point of view* the here proposed method combines an improved PID/ST adaptivity with the application of a simple uniform structure containing appropriate tunable parameters. These are adjusted by the Simplex Algorithm. This tuning is a rough analogy of controlling a bowl rolling on the surface of a plane in a gravitational field: small variation in tilting the plane can keep the bowl on the appropriate trajectory. The number of the free parameters was found to be large enough to cope with the problem of environmental interactions. The need for small and fast changes in the directly tuned parameters made it possible to use a fast, dynamic, "incomplete or partial system identification" with time-varying "identified" parameters. A further ancillary tool to back learning is regression analysis was applied in our approach.

In its philosophy this approach is similar to the traditional Soft Computing but the structures it uses are better fit to the requirements of Classical Mechanics, therefore considerable reduction in the number of the tuned parameters was achieved in comparison with a general approach. The new method also is free from scaling problems. The synthesis of the individually quite limited methods led to an efficient control in which the significance of the different components changes according to the task to be executed. The method is based on simple considerations, does not require *a priori* information on the dynamic model of the robot and its environment.

Regarding technology needs, in the last section of the robot arm a single, uncontrolled degree of freedom was built up in the form of a spring of known stiffness. This makes it possible to transform the required contact force of the technological operation into a deformation of the "nominal" trajectory. Accurate tracking of the nominal trajectory automatically leads to the required technological contact force. The good adaptive properties of the control made it possible to guarantee good tracking of the nominal trajectory for the controller without modeling the details of the environmental interactions.

For simulation tests a SCARA arm was considered. It was required to polish the surface of a Gaussian bell-shaped surface with a disc spinning on the end of a cardan shaft.

II. DETAILS OF THE METHOD APPLIED

The uniform structure applied originate from the Euler-Lagrange equation of motion considered at the following level of abstraction:

$$\sum_j M_{ij}(\mathbf{q})\ddot{q}_j + \sum_{js}\frac{\partial M_{ij}}{\partial q_s}\dot{q}_s\dot{q}_j - \sum_{sj}\frac{\partial M_{sj}}{\partial q_i}\dot{q}_s\dot{q}_j + \\ +\frac{\partial V(\mathbf{q})}{\partial q_i} = Q_i \quad (1)$$

In our previous approaches the symmetric positive definite inertia matrix "M" was constructed of a diagonal and an orthogonal matrix (**D** and **O** respectively) according to the "*Singular Value Decomposition*" as --in a 3DOF case--

$$\mathbf{M}(\mathbf{q}) = \mathbf{O}(\mathbf{q})\mathbf{D}(\mathbf{q})\mathbf{O}^T(\mathbf{q}), \quad (2)$$

$$\mathbf{O} = \mathbf{O}^{(1,2)}(\mathbf{q})\mathbf{O}^{(2,3)}(\mathbf{q})\mathbf{O}^{(3,1)}(\mathbf{q}), \\ \mathbf{O}^{(i,j)}(\mathbf{q}) \equiv \mathbf{O}^{(i,j)}(\xi_{ij}(\mathbf{q})), \quad D_{ii}(\mathbf{q}) = \exp(\xi_{ii}(\mathbf{q})) \quad (3)$$

The directly tuned parameters were the $g_{ijk}=\partial\xi_{ij}/\partial q_k$ partial derivatives and the estimated inertia was integrated according to these ever varying coefficients. This decomposition can describe the Coriolis forces and all the other terms quadratic in the angular velocities in (1).

However, the structures described by Eqs.(2,3) have the disadvantage of some computational complexity in calculating the quadratic terms in (1).

To avoid this inconvenience in the present paper the simplified model in (4) was used for **M**. (In this structure the *symmetric positive definite* **M** is constructed as the sum of *symmetric positive semi-definite* matrices.

$$\begin{bmatrix}\cos\xi_{12} & -\sin\xi_{12} & 0\\ \sin\xi_{12} & \cos\xi_{12} & 0\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}\exp(\xi_{11}) & 0 & 0\\ 0 & \exp(\xi_{22}) & 0\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}\cos\xi_{12} & \sin\xi_{12} & 0\\ -\sin\xi_{12} & \cos\xi_{12} & 0\\ 0 & 0 & 0\end{bmatrix}+$$
$$+\begin{bmatrix}\cos\xi_{13} & 0 & -\sin\xi_{13}\\ 0 & 0 & 0\\ \sin\xi_{13} & 0 & \cos\xi_{13}\end{bmatrix}\begin{bmatrix}\exp(\xi_{11}) & 0 & 0\\ 0 & 0 & 0\\ 0 & 0 & \exp(\xi_{33})\end{bmatrix}\begin{bmatrix}\cos\xi_{13} & 0 & \sin\xi_{13}\\ 0 & 0 & 0\\ -\sin\xi_{13} & 0 & \cos\xi_{13}\end{bmatrix}+$$
$$+\begin{bmatrix}0 & 0 & 0\\ 0 & \cos\xi_{23} & -\sin\xi_{23}\\ 0 & \sin\xi_{23} & \cos\xi_{23}\end{bmatrix}\begin{bmatrix}0 & 0 & 0\\ 0 & \exp(\xi_{22}) & 0\\ 0 & 0 & \exp(\xi_{33})\end{bmatrix}\begin{bmatrix}0 & 0 & 0\\ 0 & \cos\xi_{23} & \sin\xi_{23}\\ 0 & -\sin\xi_{23} & \cos\xi_{23}\end{bmatrix} \quad (4)$$

However, if none of the exponentials in the main diagonal approaches zero the sum is *positive definite* since the null space of each matrix is "covered" by the other matrices.)

For more than 3 DOF this structure can easily be generalized by using pairwisely chosen combinations of the exponentials $\{\exp(\xi_{ii})|i=1,...,DOF\}$. It is easy to calculate the matrix elements of **M** in this representation in simple closed analytic form. Furthermore, the partial derivatives $\frac{\partial M_{ij}}{\partial \xi_{kl}}$ used in the terms of (1) can simply be expressed in closed form, too. Therefore, in developing the model in (1) any matrix product can be evaded: only linear combinations of special DOF×DOF matrices must be computed. The directly tuned parameters are the $g_{ijk} := \frac{\partial \xi_{ij}}{\partial q_k}$ partial derivatives.

This form has the following advantages with respect to the "traditional" and Artificial Neural Network -based descriptions:

- The tedious work of constructing a dynamic model on the basis of the Denavit-Hartenberg conventions can be avoided.
- The number and the proper role of each tuned parameter is clearly set and determined from starting the tuning; This number is quite limited in comparison with the possibly required number of neurons in the case of a multilayer perceptron.
- The characteristic ranges of the tuned parameters were set almost completely independently from the particular dynamic properties of the robot modeled by this structure (from 0 to 2π for the rotational ones, and for the exponential terms it is trivial that *exp*(-10) is very small and *exp*(10) is very big, that is in the practice when at least an order of magnitude estimation is available for the dynamic data of the robot this scaling can be regarded practically problem-independent).

The initial model is a pure diagonal matrix proportional to the identity operator. This model is improved step by step by tuning the "g_{ijk}" parameters according to the *Simplex Algorithm* in which a function of the difference between the desired ("D") and the achieved (R) joint accelerations is minimized:

$$Cost=\frac{\left|\ddot{\mathbf{q}}^D-\ddot{\mathbf{q}}^R\right|}{1+\frac{\left|\ddot{\mathbf{q}}^D\right|}{\mathrm{KRel}}}\cong\begin{cases}\left|\ddot{\mathbf{q}}^D-\ddot{\mathbf{q}}^R\right| \; if \; \left|\ddot{\mathbf{q}}^D\right|<<\mathrm{KRel}\\ \mathrm{KRel}\frac{\left|\ddot{\mathbf{q}}^D-\ddot{\mathbf{q}}^R\right|}{\left|\ddot{\mathbf{q}}^D\right|} \; if \; \left|\ddot{\mathbf{q}}^D\right|>>\mathrm{KRel}\end{cases} \quad (5)$$

This cost function is proportional to the *relative error* in the acceleration for "*large*" desired acceleration, and it approximates the *absolute error* for "*small*" desired ones (terms *large* and *small* are to be understood in comparison with a constant *KRel*).

To support this process further ancillary tools were applied:

- a $sigmoid(x)=x/(1+|x|)\in(-1,1)$ function used in stabilization against the effect of extreme noises in the terms
  $\sin(\xi_{ij})\to\sin(\pi\times sigmoid(\xi_{ij}/\pi))$
  $\cos(\xi_{ij})\to\cos(\pi\times sigmoid(\xi_{ij}/\pi))$
  $\exp(\xi_{ii})\to\exp(2.3\times sigmoid(\xi_{ii}/2.3))$
  (for reducing computational complexity this saturated nature is not taken into account in the calculation of the partial derivatives of **M**);
- an "*Additional Generalized Force*" term based on a simple version of regression analysis in which the prediction is "qualified" and suppressed according to the noisiness of the environment it originates from [11];
- a PID term in which the coefficients of the proportional, derivative and integrated term are tuned as the function of the integrated error in order to keep a prescribed pole-structure in the *desired damping* of the coordinate errors fixed (described in details e.g. in [9]); in the present version this approach is improved by allowing this feedback increase if the overall torques of the drives are smooth functions of time, and it is decreased in the more "noisy" phases of the motion; here "noisiness" is determined by the *forgetting sum* $c_{int}(t+1)=\alpha\times c_{int}(t)+|\mathbf{Q}(t)-\mathbf{Q}(t-1)|$ and a *fuzzy membership function* describing the "*smoothness*" of the torque signal in comparison with a reference value *cCoeff*.

$$c=c_0\left(1+2\frac{cCoeff}{cCoeff+(1-\alpha)c_{int}}\right), \quad (6)$$

$$\ddot{\varepsilon}=-b'\varepsilon-c'\dot{\varepsilon}-k\int_{-\infty}^{t}\varepsilon(t')dt' \quad (7)$$

$$\kappa=\frac{c}{2}\left[1+sigmoid\left(50\left|\int_{-\infty}^{t}\varepsilon(t')dt'\right|\right)\right] \quad (8)$$

$$c'=c+\kappa,\; b'=\frac{c^2}{4}+c\kappa,\; k=\frac{c^2}{4}\kappa \quad (9)$$

The fraction in '*c*' can be interpreted as a fuzzy set describing the "smoothness" of the control: for small torque derivatives it approaches 1, while for too fast changes in the momentum it converges to zero; this rigid rule means that for strongly varying momentum it is not reasonable to require too strong feedback in order to avoid instabilities and overshots, but in the "stable phase" of the control an increase in the feedback may improve accuracy.

- a truncation in the angular velocities at a lower limit when calculating the inertia matrix according to (4) to achieve good adaptivity for slow motion, too (detailed in [10]);
- *"external loop parameters"* of slow tuning used as *reference values* --built in certain *fuzzy membership functions*-- in the *"assessment"* of several properties of the control; their appropriate value can be set roughly *"experimentally"*; further slow real-time tuning can help in finding their *optimum* value; since the optimum setting can change in time, it is expedient to keep them adjusted in real-time;

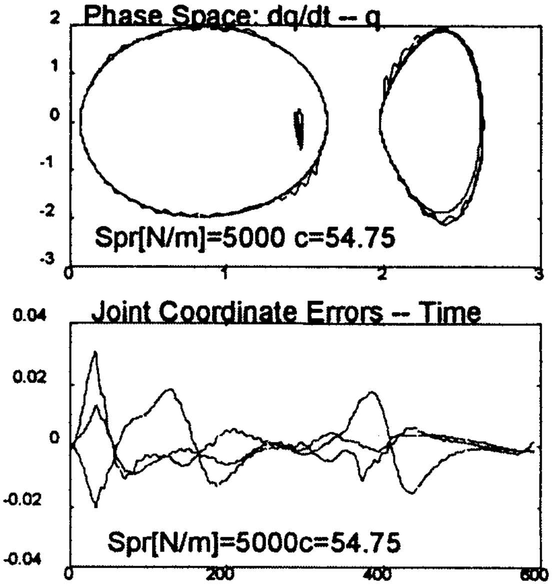
In the sequel simulation examples are presented to illustrate the operation of the method.

III. SIMULATION INVESTIGATIONS

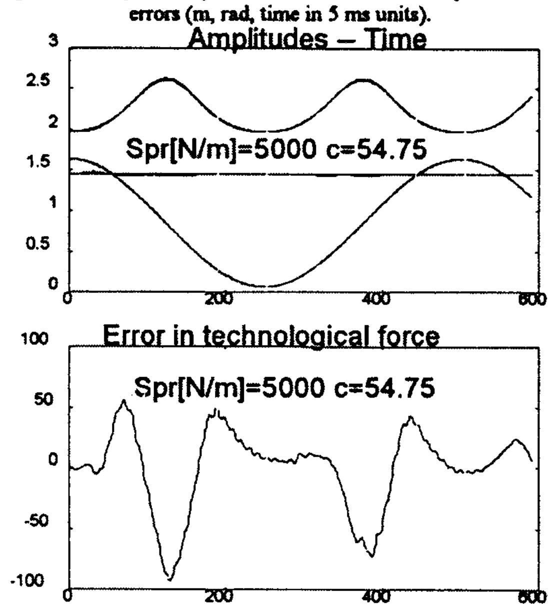
In the simulation test a SCARA arm was required to polish the surface of a Gaussian bell-shaped surface with a disc spinning on the end of a cardan shaft. Due to this construction the polishing disk automatically takes the orientation of the tangent of the surface to be polished. The spring was able to move in the vertical direction. The robot's linear shaft was oriented vertically, the two rotary axes were parallel with it allowing horizontal motion of the two arms of 2 m length each. In the simulation the spring constant *Spr* was equal to 5000 N/m, the required contact force in the direction of the external normal vector of the surface to be polished was 1000 N. Due to the fast spinning of the disc frictional forces along the tangent of the surface were neglected, but the contact torque parallel to it was taken into account —only in the simulation but not in the program of the controller—. It was supposed that the polishing disk has a fast damped motion without oscillation resulting in much faster motion than that of the full robot arm.

In Figs. 1-2 the phase-trajectories and the joint coordinate errors are given for the nominal and the simulated motion.

In Figs. 3-4 the desired and the simulated joint coordinate trajectories and the error of the required 1000 N contact force are described. It can well be seen that the relative force error is under 10%. The necessary torque components to be exerted by the drives are described in Figs. 5-6. It is clear that only a small portion of the force originates from the regression-based term, the other contribution is the result of the more intelligent structure-based model and learning.



Figs. 1:-2 The phase trajectories for motion and the joint coordinate errors (m, rad, time in 5 ms units).



Figs. 3-4: The nominal and the simulated joint coordinates (m, rad), and the error in the prescribed contact force (N)

To reveal the background processes Figs. 7-8 describe the six independent components of the symmetric inertia matrix as estimated by the control.

Figs. 9-10 describing certain directly tuned parameters support the idea that only very small and fast adjustment is necessary for keeping the motion near the nominal path. It can also be seen that the value of the parameter "c" considerably changes depending on the noisiness of the torque signal. This adaptivity plays an important role in the stability of the control.

To illustrate the effect of the desired speed of motion prescribed for the end-effector Figs. 11-12 describe the phase trajectories and the joint coordinate errors for an increased nominal velocity along the same polishing path (extremely high velocity). The errors increased in comparison with that of the slower desired motion. However, the control remained stable. In Figs. 13-14 the full torques and the error in the contact force also increased (this latter to max. 25%).

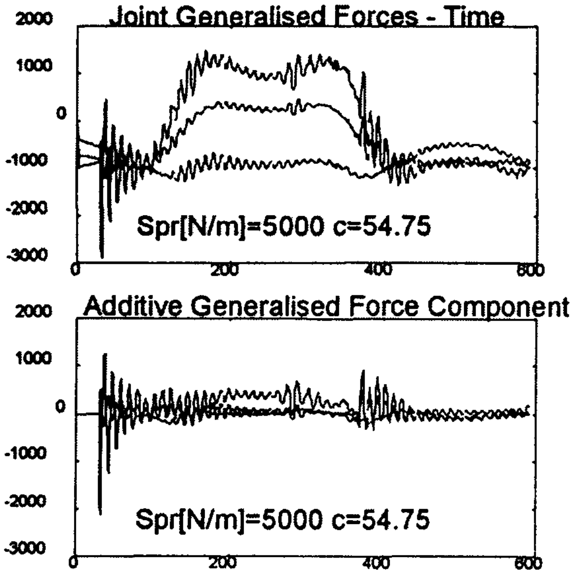
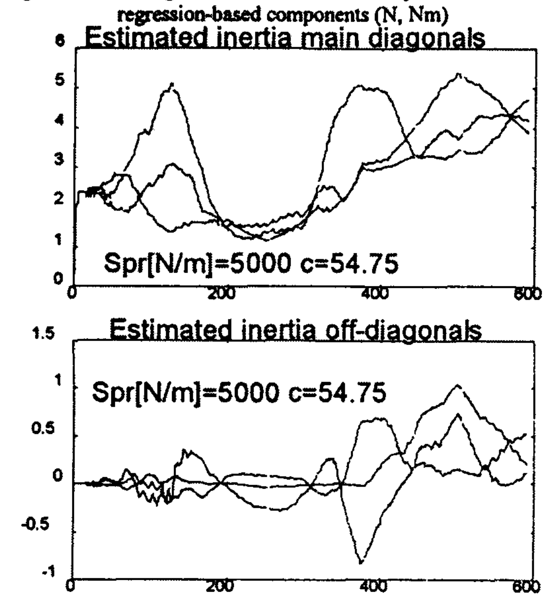


Fig. 5-6: The full generalized forces exerted by the drives and their regression-based components (N, Nm)



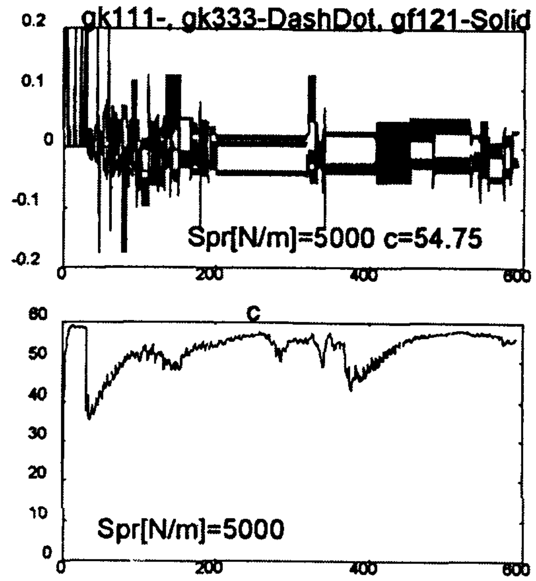
Figs. 7-8: The estimated inertia matrix

The -1000 N limit in the transient phase indicates that for a short time the polishing disc left the surface of the work-piece. However, after some learning this error is considerably reduced even in this *extreme* case, too.

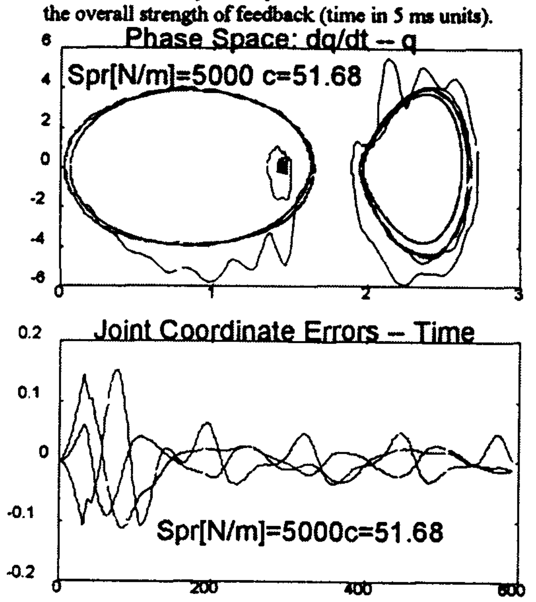
IV. CONLUSIONS

It was illustrated via simulations that the proposed method based on uniform structures with free parameters of reduced complexity and coupling and supported by several simple *ancillary methods* can serve as an efficient adaptive control for robots used in grinding or polishing operations.

The control corresponds to a synthesis of certain, individually quite limited methods the elements of which are present in the conventional *soft computing* approaches, too. These are partly *"rigid"* computational rules applied in the PID/ST part of the control, *fuzzy membership functions* qualifying certain *observed properties of the control*, and saturated sigmoid functions typical in



Figs. 9-10: Certain directly tuned parameters and factor "*c*" determining the overall strength of feedback (time in 5 ms units).



Figs. 11-12: the phase trajectories and the joint coordinate errors for an increased nominal velocity along the same polishing path

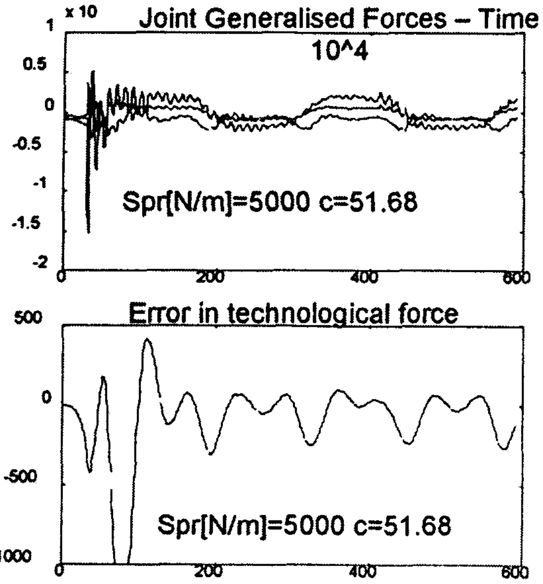
artificial neural networks (ANNs), too. Being free of the formal requirements of the pure ANN or fuzzy based methods the computational structure can be more simplified and tightly tailored to the requirements of the control of mechanical systems.

The new method also is free from scaling problems. It can be regarded as a compromise between the traditional Soft Computing and Hard Computing.

In the control the significance of the different components remains comparable and changes according to the task to be executed.

The introduction of the passive compliant element makes if possible to apply the method for technological operations transforming force needs into the deformation of the nominal trajectory .

Further investigations are needed for different robot arm structures and technological operations, too.



Figs. 13-14: The full torques and the error in the contact error also increased

V. ACKNOWLEDGMENT

The authors gratefully acknowledge the support by the FANUC Fund and that of the Hungarian-Portuguese Bilateral Scientific and Technology Co-operation Fund TéT the authors obtained for 1998-1999.

VI. REFERENCES

[1] Lantos, B., "Identification and Adaptive Control of Robots", International Journal Mechatronics, Vol. 2, No. 3, 1993, pp. 149-166.

[2] R. Reed: "Pruning Algorithms - A Survey", IEEE Transactions on Neural Networks, 4., pp.- 740-747, 1993.

[3] S. Fahlmann, C. Lebiere: "The Cascade-Correlation Learning Architecture", Advances in Neural Information Processing Systems, 2, pp. 524-532, 1990.

[4] T. Nabhan, A. Zomaya: "Toward Generating Neural Network Structures for Function Approximation", Neural Networks, 7, pp. 89-9, 1994.

[5] G. Magoulas, N. Vrahatis, G. Androulakis: "Effective Backpropagation Training with Variable Stepsize" Neural Networks, 10, pp. 69-82, 1997.

[6] C. Chen, W. Chang: "A Feedforward Neural Network with Function Shape Autotuning", Neural Networks, 9, pp. 627-641, 1996.

[7] W. Kinnenbrock: "Accelerating the Standard Backpropagation Method Using a Genetic Approach", Neurocomputing, 6, pp. 583-588, 1994.

[8] A. Kanarachos, K. Geramanis: "Semi-Stochastic Complex Neural Networks", IFAC-CAEA '98 Control Applications and Ergonomics in Agriculture, pp. 47-52, 1998.

[9] J.K. Tar, K. Koz³owski, I.J. Rudas, J.F. Bitó: "Combined Application of Regression Analysis and Approximate Modeling in an Adaptive Control for Mechanical Devices", in the Proc. of the 5th International Symposium on Methods and Models in Automation and Robotics -- MMAR'98, 25-29 August 1998, Miedzyzdroje, Poland, pp. 807-812, (ISBN 83-87423-81-5).

[10] J.K. Tar, O.M. Kaynak, I.J. Rudas, J.F. Bitó: "The Use of Partially Decoupled Uniform Structures and Procedures for the Robust and Adaptive Control of Mechanical Devices", in the Proc. of the 2^{nd} International Conference on Recent Advances in Mechatronics, 24-26 May 1999, Istanbul, Turkey (also included amongst the selected papers issued by Springer-Verlag ISBN: 981-4021-34-2).

[11] J.K. Tar, L. Horváth, J.A. Tenreiro Machado, I.J. Rudas: "Selection of Appropriate Uniform Structures for an Adaptive Control of Robot Under Environmental Interaction", in Proc. of the "Jubilee Scientific Conference at Bánki Donát Polytechnic", 1-2 September 1999, Budapest, Hungary.