DIGITAL CONTROL OF OVER-ACTUATED PARALLEL ROBOTS

KVĚTOSLAV BELDA, JOSEF BÖHM, MICHAEL VALÁŠEK*

Department of Adaptive Systems, Institute of Information Theory and Automation, Academy of Sciences of the Czech Republic, Pod vodárenskou věží 4, 182 08 Prague 8, fax: +420 - 26605 2068 and e-mails: belda@utia.cas.cz; bohm@utia.cas.cz
*Department of Mechanics, Faculty of Mechanical Engineering, Czech Technical University in Prague, Karlovo nám. 13, 121 35 Prague 2, e-mail: valasek@fsik.cvut.cz.

Abstract. Parallel robots, especially over-actuated, which are suitably controlled, represent promising way to improve dynamics, stiffness, accuracy and productivity of modern machine tools and their centers. The over-actuation means redundantly actuated structure i.e. movable platform supported by several links includes more drives than DOF. This layout solves existence of singular positions in a workspace and furthermore improves stiffness and dynamics. This paper deals with two possible ways of model-based control for such structures: Sliding Mode Control (SMC) and Generalized Predictive Control (GPC). The both use discrete linear state-space model. Since the robot (mechanical structure) cannot be described by linear model, the paper introduces suitable methods – direct pseudo-discretization based on Taylor series and exact linearization-decomposition that enable us to use even nonlinear models of the robots.

Key Words. Linearization-discretization, sliding mode control, generalized predictive control.

1. INTRODUCTION

Nowadays, the further development in industrial area is constrained by deficit of powerful machines with adequate dynamics and stiffness. Utilization of parallel robots, especially over-actuated (in other words redundant), which are controlled by suitable control algorithms, represents promising way to improve dynamics, stiffness, accuracy and also productivity of machine tools and their centers. One their example is shown in Fig. 1.

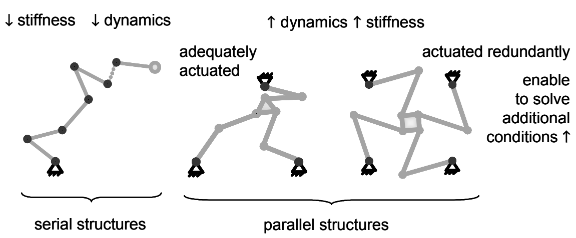

Fig. 1. Comparison of structures of robots.

The robots based on redundant parallel kinematics [5] can be understood as well as adequately-actuated structures – movable truss constructions or the movable platform (place of tool or gripper) supported by several links – but they include more actuated links than degrees of freedom /DOF/. This option reduces existence of singular positions of adequate structures and furthermore improves stiffness and dynamics.

The important and not fully solved issue is their control. This paper deals with two possible ways based on model approach: Sliding Mode Control (SMC) [3] and Generalized Predictive Control (GPC) [2]. These strategies use for design linear state-space model of the controlled system. Since the robot (mechanical structure) cannot be described by linear model, the paper introduces two suitable methods – – direct pseudo-discretization by Taylor series and exact linearization-decomposition [4] that enable us to use even nonlinear models of robot structures. The digital (discrete) cases of control strategies are applied due to practical reasons.

2. NONLINEAR MODEL OF THE ROBOTS

Model, which is used for control design, describes the robot dynamics and it is represented by equations of motion. They give integral information about relations of kinematic variables of separate bodies, external force effects (external forces, torques generated e.g. by drives), force effects in joints (reactions) and other internal force effects. Robot structure generally represents multibody system. For composition of equations of motion in robotics, the Lagrange's equations of second or mixed type, based on expression of kinetic and potential energy are mostly used. Difference between the types of Lagrange's equations is given by a choice of so-called generalized coordinates, which, in the first case, must be independent. In the most of cases, the expressions in independent coordinates are too difficult, therefore let us assume the latter case, mixed type that is more general. According the theory described e.g. in [1] the model (DAE form) is given partly by structural relations

$$\mathbf{f}(\mathbf{s}(t)) = \mathbf{0}\,,\ \left(f_k(\mathbf{s}(t) = 0;\ k = 1,\cdots,r\ \ (r = n - i)\right) \quad (1)$$

and partly by a system of n second-order differential equations (equations of motion)

$$\mathbf{M}\ddot{\mathbf{s}} = \mathbf{g} + \mathbf{T}\mathbf{u} + \mathbf{\Phi}^T\boldsymbol{\lambda}\,,\ (\mathbf{\Phi}:\ \phi_{k,j} = \frac{\partial f_k}{\partial s_j};\ \begin{matrix} k = 1,\cdots,r \\ j = 1,\cdots,n \end{matrix}) \quad (2)$$

where $\mathbf{s}$ are physical coordinates, n is their number, i is a number of degrees of freedom, $\mathbf{M}$ is a regular symmetric square matrix of an order n, $\mathbf{g}$ is an n dimensional vector, $\mathbf{T}$ is a rectangular matrix of type $n \times m$, u is an m dimensional input vector, $\mathbf{\Phi}^T$ is a transpose of overall Jacobian of type $n \times r$ and $\boldsymbol{\lambda}$ is a vector of Lagrange's multipliers with dimension $r = n - i$.

To design control more simply, we search for the most compact notation of system equations (1)-(2) with as small as possible number of unknown parameters.

According to [1] the equations can be transformed to independent coordinates. Used transformation reduces the number of differential equations and mainly it removes redundant Lagrange's multipliers, connected with structural forces and structural relations. The transformation searches for such solution, which annuls element $\mathbf{\Phi}^T\ \boldsymbol{\lambda}$ and reduces physical coordinates only to independent coordinates equaled DOF.

From mathematical point of view, the suitable solution is a null space of the whole Jacobian $\mathbf{\Phi}$, which is simultaneously interpreted as a Jacobian matrix $\mathbf{R}$, fitting the following equality, arising from properties of the null space

$$\mathbf{\Phi}_s\,\mathbf{R} = \mathbf{R}^T\mathbf{\Phi}_s^T = \mathbf{0} \quad (3)$$

and arising also from recomputation of independent to physical coordinates

$$\mathbf{s} = \mathbf{s}(\mathbf{x}) \ \Rightarrow\ \dot{\mathbf{s}} = \frac{\partial \mathbf{s}}{\partial \mathbf{x}}\dot{\mathbf{x}} = \mathbf{R}\dot{\mathbf{x}} \ \Rightarrow\ \ddot{\mathbf{s}} = \dot{\mathbf{R}}\dot{\mathbf{x}} + \mathbf{R}\ddot{\mathbf{x}} \quad (4)$$

Now, if we insert eq. (4) for eq. (2) and perform its multiplication by transposition of Jacobian matrix $\mathbf{R}^T$, we obtain resultant system of equations - pure equations of motion

$$\mathbf{R}^T\mathbf{M}\mathbf{R}\ddot{\mathbf{x}} + \mathbf{R}^T\mathbf{M}\dot{\mathbf{R}}\dot{\mathbf{x}} = \mathbf{R}^T\mathbf{g} + \mathbf{R}^T\mathbf{T}\mathbf{u} \quad (5)$$

transformed and simplified to a form with isolated second derivatives

$$\begin{aligned} \ddot{\mathbf{x}} &= (\mathbf{R}^T\mathbf{M}\mathbf{R})^{-1}\mathbf{R}^T(\mathbf{g} - \mathbf{M}\dot{\mathbf{R}}\dot{\mathbf{x}}) + (\mathbf{R}^T\mathbf{M}\mathbf{R})^{-1}\mathbf{R}^T\mathbf{T}\mathbf{u} \\ \ddot{\mathbf{x}} &= \qquad\qquad \mathsf{f}(\mathbf{x},\dot{\mathbf{x}}) \qquad\qquad + \qquad \mathsf{g}(\mathbf{x}) \qquad \mathbf{u} \end{aligned} \quad (6)$$

Then, the equations (6) can be simply normalized into state-space formulation (7)

$$\dot{\mathbf{X}} = \mathbf{f}(\mathbf{X}) + \mathbf{B}_C(\mathbf{X})\,\mathbf{u}\ \Big|_{\mathbf{f}(\mathbf{X}) = \begin{bmatrix}\mathbf{x} \\ \mathsf{f}(\mathbf{x},\dot{\mathbf{x}})\end{bmatrix},\ \mathbf{B}_C(\mathbf{X}) = \begin{bmatrix}\mathbf{0} \\ \mathsf{g}(\mathbf{x})\end{bmatrix}} \quad (7)$$

where $\mathbf{X} = [\mathbf{x},\dot{\mathbf{x}}]^T$, $\dot{\mathbf{X}} = [\dot{\mathbf{x}},\ddot{\mathbf{x}} = \mathsf{f}(\mathbf{x},\dot{\mathbf{x}})]^T$ respectively.

This form can be already linearized and discretized for future control design.

3. LINEARIZATION – DISCRETIZATION

The model-based control strategies use the model as prior information on future behavior of controlled system.

From mathematical point of view, the strategies (algorithms) solve specific equations for unknown control actions. The equations include information on desired values, current state of controlled system (output) with relation to designed control (input).

This relation is represented just by model of controlled system, which should be linear, at least in view of unknown control. This condition is sufficient for one-step strategies. Multi-step strategies need moreover linear relation for state (output) of the system.

The subsection 3.1 introduces method based on Taylor series providing direct pseudo-discretization, suitable only for one-step strategies. The subsequent subsection 3.2 explains the method of exact linearization - decomposition meant mainly for multi-step strategies. It is based on differences among working points and one appropriately selected reference point.

3.1 Direct Pseudo-Discretization via Taylor Series

Let us consider the system of nonlinear differential equations (6) and Taylor series with first two terms

$$\mathbf{x}(k+1)=\mathbf{x}(k)+\frac{\dot{\mathbf{x}}}{1!}\delta+\frac{\ddot{\mathbf{x}}}{2!}\delta^2+(Oh^3)\Big|_{\ddot{\mathbf{x}}=\mathbf{f}(\mathbf{X}(k)=[\mathbf{x}(k),\dot{\mathbf{x}}(k)]^T)+\mathbf{g}(\mathbf{x}(k))\mathbf{u}} \quad (8)$$

where δ is sampling and Oh^3 is remainder term. Then the equations (6) can be directly transformed to the following pseudo-discrete form:

$$\mathbf{X}(k+1)=\mathbf{A}(\mathbf{X}(k))+\mathbf{B}(\mathbf{X}(k))\mathbf{u}(k) \quad (9)$$

where the matrixes $\mathbf{A}(\mathbf{X}(k)$ and $\mathbf{B}(\mathbf{X}(k))$, considering Taylor series (8) consist of the elements:

$$\mathbf{A}(\mathbf{X}(k))\Big|_{\mathbf{X}(k)=\begin{bmatrix}\mathbf{x}\\ \dot{\mathbf{x}}\end{bmatrix}}=\begin{bmatrix}\mathbf{x}(k)+\delta\ \dot{\mathbf{x}}(k)+\frac{\delta^2}{2}\mathbf{f}(\mathbf{X}(k))\\ \dot{\mathbf{x}}(k)+\delta\,\mathbf{f}(\mathbf{X}(k))\end{bmatrix},$$

$$\mathbf{B}(\mathbf{X}(k))\Big|_{\mathbf{X}(k)=\begin{bmatrix}\mathbf{x}\\ \dot{\mathbf{x}}\end{bmatrix}}=\begin{bmatrix}\frac{\delta^2}{2}\mathbf{g}(\mathbf{X}(k))\\ \delta\,\mathbf{g}(\mathbf{X}(k))\end{bmatrix} \quad (10)$$

and state vector $\mathbf{X}(k)$ is composed of $[\mathbf{x},\dot{\mathbf{x}}]^T$. Such simple reformulation (9) with specification (10) is sufficient for one-step strategies. It is linear only to control actions (system input) $\mathbf{u}$ and nonlinearity for state stays.

3.2 Exact Linearization – Decomposition

Almost all model-based control strategies use the model in some linear form. The following derivation introduces one simple specific algorithm of linearization, which gives exact solution indicated by (11)

$$\dot{\mathbf{X}}=\mathbf{f}(\mathbf{X})=\mathbf{A}_C(\mathbf{X})\mathbf{X} \quad (11)$$

Let us assume that the nonlinear function $\mathbf{f}(\mathbf{X})$ (autonomous system, $\mathbf{u}=\mathbf{0}$) and point $\mathbf{X}$ be given, $\mathbf{X}$ belongs to range of definition of the function and in a case of zero elements in $\mathbf{X}$, is possible to substitute them by suitable nonzero number $\kappa\to 0$, to avoid zero division. Furthermore, let us assume two types of state variables: generally outputs $\mathbf{x}$ and their time derivatives $\dot{\mathbf{x}}$; in detail

$$\mathbf{X}=\begin{bmatrix}\mathbf{x}\\ \dot{\mathbf{x}}\end{bmatrix}=\begin{bmatrix}\mathbf{x}_1\\ \mathbf{x}_2\end{bmatrix}=\begin{bmatrix}\mathbf{x}_1=[x_{11},x_{12},x_{13}]\\ \mathbf{x}_2=[x_{21},x_{22},x_{23}]\end{bmatrix} \quad (12)$$

Finally, let us assume that

$$\mathbf{f}(\mathbf{X})=\mathbf{0}\Big|_{\mathbf{X}=\mathbf{X}_r=[\mathbf{x}_{r1}\text{arbitrary},\ \mathbf{x}_{r2}=\mathbf{0}]} \quad (13)$$

Then, we can obtain the decomposition indicated in eq. (11). It starts from second group of variables $\mathbf{x}_2$ (i.e. according to eq. (12): order is $\{x_{21}, x_{22}, x_{23}, x_{11}, x_{12}, x_{13}\}$ i.e. {4, 5, 6, 1, 2, 3}). The order is given by amount of the information included in the function $\mathbf{f}(\mathbf{X})$. Its first elements are only copies of $\mathbf{x}_2$ but its second elements represent its own nonlinear relation expressed by $\ddot{\mathbf{x}}=\mathbf{f}(\mathbf{x},\dot{\mathbf{x}})+\mathbf{g}(\mathbf{x})\mathbf{u}|_{\mathbf{u}=\mathbf{0}}$. For these assumptions the exact linearization – decomposition can be based on differences

$$\mathbf{f}(\mathbf{X})=\frac{\Delta\mathbf{f}(\circ)}{.\Delta x_{11}}\Delta x_{11}+\frac{\Delta\mathbf{f}(\circ)}{.\Delta x_{12}}\Delta x_{12}+\frac{\Delta\mathbf{f}(\circ)}{.\Delta x_{13}}\Delta x_{13}+$$
$$+\frac{\Delta\mathbf{f}(\circ)}{.(x_{21}-0)}(x_{21}-0)+\frac{\Delta\mathbf{f}(\circ)}{.(x_{22}-0)}(x_{22}-0)+\frac{\Delta\mathbf{f}(\circ)}{.(x_{23}-0)}(x_{23}-0) \quad (14)$$

(Note: The dots in denominators mark division 'element by element'; i.e. elements of differences by scalar Δx_{ij}) i.e.

$$\mathbf{f}(\mathbf{X})=\frac{\mathbf{f}([x_{11},x_{12},x_{13},0,0,0]^T)-\mathbf{f}([x_{r11},x_{12},x_{13},0,0,0]^T)}{.(x_{11}-x_{r11})}(x_{11}-x_{r11})+\cdots$$
$$+\frac{\mathbf{f}([x_{11},x_{12},x_{13},x_{21},x_{22},x_{23}]^T)-\mathbf{f}([x_{11},x_{12},x_{13},0,x_{22},x_{23}]^T)}{.x_{21}}x_{21}+\cdots \quad (15)$$

Let us analyze expression (15). The individual fractions are columns of matrix $\mathbf{A}_C(\mathbf{X})$. The internal structure of the matrix is the following (in our example, the matrix is sixth order, according to number of state variables)

$$\mathbf{A}_C(\mathbf{X})=\left[\begin{array}{ccc|ccc}0&0&0&1&0&0\\0&0&0&0&1&0\\0&0&0&0&0&1\\ \hline 0&0&0&a_{44}&a_{45}&a_{46}\\0&0&0&a_{54}&a_{55}&a_{56}\\0&0&0&a_{64}&a_{65}&a_{66}\end{array}\right] \quad (16)$$

the first three columns contain only zeros because are composed of differences being also zeros - the vector function $\mathbf{f}(\mathbf{X})$ equals zeros for zero time derivatives; see equation (17) - numerator of the first column of $\mathbf{A}_C(\mathbf{X})$.

$$\mathbf{f}([x_{11},x_{12},x_{13},0,0,0]^T)-\mathbf{f}([x_{r11},x_{12},x_{13},0,0,0]^T)=\mathbf{0} \quad (17)$$

Now, we can define whole linearized continuous state-space formulation with time-varying elements

$$\dot{\mathbf{X}}=\mathbf{f}(\mathbf{X})+\mathbf{B}_C(\mathbf{X})\,\mathbf{u}\ \rightarrow\ \dot{\mathbf{X}}=\mathbf{A}_C(\mathbf{X})\mathbf{X}+\mathbf{B}_C(\mathbf{X})\,\mathbf{u} \quad (18)$$

The formulation (18) can be already discretized by standard discretization to a form

$$\begin{aligned}\mathbf{X}(k+1)&=\mathbf{A}\,\mathbf{X}(k)+\mathbf{B}\,\mathbf{u}(k)\\ \mathbf{y}(k)&=\mathbf{C}\,\mathbf{X}(k)\end{aligned} \quad (19)$$

Now the models – pseudo-discrete form (9) or ordinary state-space formulation (19) are prepared to the use in appropriate control strategies.

4. SLIDING MODE CONTROL

Discrete type of Sliding Mode Control [3] is derived by analogy to the theory of stability in a continuous domain. It is based on 'switching' control action and performance of Lyapunov stability theorem. The state is driven towards a desired switching (sliding) hyperplane. The 'switching' maintains the state on this hyperplane once it is reached, in spite of perturbations. This method offers an advantage of accuracy for the cost of control dithering, which ensues from the 'switching' part.

Now we can start with derivation of control law. Firstly let us define a discrete Lyapunov function:

$$V(\mathbf{X}(k)) = \mathbf{s}^2(\mathbf{X}(k)) > 0$$

$$\Delta V(\mathbf{X}(k)) = \mathbf{s}^2(\mathbf{X}(k+1)) - \mathbf{s}^2(\mathbf{X}(k)) \le 0 \qquad (20)$$

Eq. (20) represents the discrete convergence and attractivity condition. According to them the **s**-dynamics and sliding hyperplane may be chosen as

$$\mathbf{s}(k+1) = e^{-P\delta}\mathbf{s}(k) - \mathbf{K}sign(\mathbf{s}(k)) \text{ and } \mathbf{s}(k) = \mathbf{Ce}\,(k) \qquad (21)$$

where P is positive scalar, δ is sampling and $\mathbf{K}$ is positive diagonal matrix and $\mathbf{s}(k) = [s_1(k)\, s_2(k) \cdots s_m(k)]^T$, $\mathbf{e}(k) = \mathbf{X}(k) - \mathbf{X}_d(k)$. $\mathbf{X}_d(k)$ is a vector of desired values with the same size as a vector $\mathbf{X}(k)$, $\mathbf{C} = diag(\mathbf{C}^i), \mathbf{C}^i = [c_1^i \quad c_2^i \quad \cdots \quad c_n^i]$. $\mathbf{C}_i$ is chosen in order to satisfy Jury's stability condition of discrete systems, i is state variable index and n is order of system. At this moment **s**-dynamics at time $t = (k+1)\,\delta$ is

$$\mathbf{s}(k+1) = \mathbf{Ce}(k+1) = \mathbf{C}[\mathbf{A}(k) + \mathbf{B}(k)\mathbf{u}(k) + \mathbf{\Psi}(k) - \mathbf{X}_d(k+1)] \qquad (22)$$

Using (21), the discrete control law is obtained like this:

$$\mathbf{u}(k) = -(\mathbf{CB}(k))^{-1}\left\{\mathbf{C}[\mathbf{A}(k) + \mathbf{\Psi}(k) - \mathbf{X}_d(k+1)] - e^{-P\delta}\mathbf{s}(k) + \mathbf{K}sign(s(k))\right\} \qquad (23)$$

$\mathbf{\Psi}(k)$ represents unknown perturbation in time $k\ \delta$, which can be estimated by $\Psi(k-1)$

$$\Psi(k-1) = \mathbf{x}_{\substack{actual\\topical}}(k) - \mathbf{A}(k-1) - \mathbf{B}(k-1)\mathbf{u}(k-1) \qquad (24)$$

This estimation is correct on the assumption that dynamics of perturbation is significantly slower than sampling and moreover an order of perturbation magnitude is much smaller.

When (23) is inserted to (22), assumption of (20) and the fact that variations in perturbation are slow against sampling frequency, the diagonal elements of matrix **K** can be selected as:

$$[k_1\ k_2 \cdots k_m]^T = \eta\,\mathbf{C}|\Psi(k-1)| > \mathbf{C}|\Psi(k) - \Psi(k-1)|, \quad \text{for } \eta > 0 \qquad (25)$$

In the text above we consider that the product of matrices $\mathbf{CB}(k)$ is regular and may be inverted, but the product in our case has deficient rank due to redundancy. We can use e.g. orthogonal-triangular decomposition (QR), least square method for deficient rank problem [7]. This result gives anew possibility to compute control law, let us say actuators, with minimum energy. Furthermore, this result can be varied by free arbitrary remainder from QR decomposition. It can serve to perform some additional requirements to control e.g. anti-backlash control, without any violation of correct energy distribution to inputs of controlled system.

5. GENERALIZED PREDICTIVE CONTROL

Generalized Predictive Control (GPC) combines feedback ~ feedforward as well as similar approach – – Linear Quadratic Control (LQC). It is a multi-step control, based on local optimization of quadratic criterion

$$J = \sum_{i=t+1}^{t+N}(\ y(i)^T Q_y\ y(i) + u(i-1)^T Q_u\ u(i-1)) \qquad (26)$$

where N is a horizon of optimization, Q_y and Q_u are output and input penalizations (here normalized as $Q_y = 1$ and $Q_u = \lambda$) and $y(i)$ and $u(i\text{-}1)$ are outputs and inputs respectively. The base of GPC control is a prediction (computation, estimation) of future values of outputs. It uses e.g. state-space model (19). The most usual, absolute algorithm of prediction is given by

$$\begin{aligned}
\hat{\mathbf{X}}(k+1) &= \mathbf{A}\ \mathbf{X}(k) + \mathbf{Bu}(k)\\
\hat{\mathbf{y}}(k+1) &= \mathbf{CA}\ \mathbf{X}(k) + \mathbf{C}\ \mathbf{Bu}(k)\\
&\vdots\\
\hat{\mathbf{X}}(k+N) &= \mathbf{A}^N\ \mathbf{X}(k) + \mathbf{A}^{N-1}\mathbf{Bu}(k) + \cdots\ \mathbf{Bu}(k+N-1)\\
\hat{\mathbf{y}}(k+N) &= \mathbf{CA}^N\ \mathbf{X}(k) + \mathbf{CA}^{N-1}\mathbf{Bu}(k) + \cdots \mathbf{CBu}(k+N-1)
\end{aligned} \qquad (27)$$

The prediction can be suitably rewritten also in matrix notation

$$\hat{\mathbf{y}} = \mathbf{f} + \mathbf{Gu}, \quad \mathbf{f} = \begin{bmatrix}\mathbf{CA}\\ \vdots\\ \mathbf{CA}^N\end{bmatrix}\mathbf{X}(k),$$

$$\mathbf{G} = \begin{bmatrix}\mathbf{C}\ \mathbf{B} & \cdots & \mathbf{0}\\ \vdots & \ddots & \vdots\\ \mathbf{CA}^{N-1}\mathbf{B} & \cdots & \mathbf{C}\ \mathbf{B}\end{bmatrix} \qquad (28)$$

To compute control actions **u**, the criterion (26) can be simply modified to the following matrix product

$$J_k = \left[[\hat{\mathbf{y}} - \mathbf{w}]^T, \mathbf{u}^T \right] \begin{bmatrix} \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\lambda} \end{bmatrix} \begin{bmatrix} \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\lambda} \end{bmatrix} \begin{bmatrix} [\hat{\mathbf{y}} - \mathbf{w}] \\ \mathbf{u} \end{bmatrix} = \mathbf{J}^T \mathbf{J} \quad (29)$$

from which only its one part (so-called square root) is necessary

$$\mathbf{J} = \begin{bmatrix} \mathbf{1} & \mathbf{0} \\ \mathbf{0} & \boldsymbol{\lambda} \end{bmatrix} \begin{bmatrix} [\hat{\mathbf{y}} - \mathbf{w}] \\ \mathbf{u} \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{y}} \\ \boldsymbol{\lambda}\mathbf{u} \end{bmatrix} - \begin{bmatrix} \mathbf{w} \\ \mathbf{0} \end{bmatrix} \quad (30)$$

In criterion the prediction (28) is substituted for $\hat{\mathbf{y}}$ and obtained system is solved by QR decomposition

$$\begin{aligned} \mathbf{A}\mathbf{u} &= \mathbf{b} \qquad /\mathbf{Q}^T \\ \mathbf{Q}^T\mathbf{A}\mathbf{u} &= \mathbf{Q}^T\mathbf{b} \\ \mathbf{R}\mathbf{u} &= \mathbf{c} \;\Rightarrow\; \mathbf{u} \end{aligned} \quad (31)$$

The result **u** represents actions for whole horizon *N*, but only first term **u**(*k*) is applied to the robot.

6. PRACTICAL RESULTS

The figures in section 6 show differences among presented control approaches – Sliding Mode Control and Generalized Predictive Control – and conventional feedback PID/PSD control. The results were obtained during real experiments on laboratory models of planar over-actuated parallel robots – Crosshead (Fig. 2a) and Sliding Star (Fig. 2b).

Crosshead represents horizontal planar structure – one level of potential energy. On the other hand, the second structure – Sliding Star represents vertical planar structure with different levels of potential energies. In that case, the gravitational forces are taken into account.

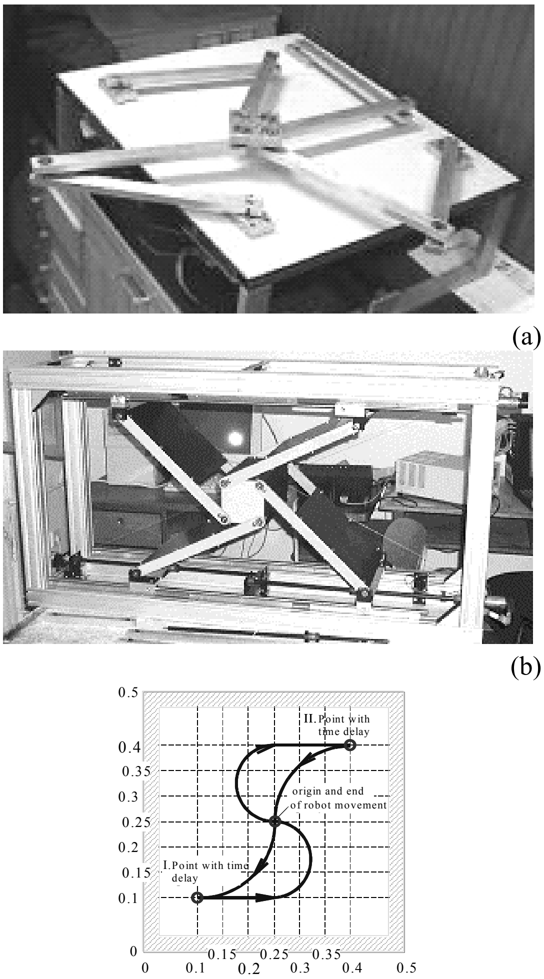

Fig. 2. Laboratory models and one exemplary trajectory.

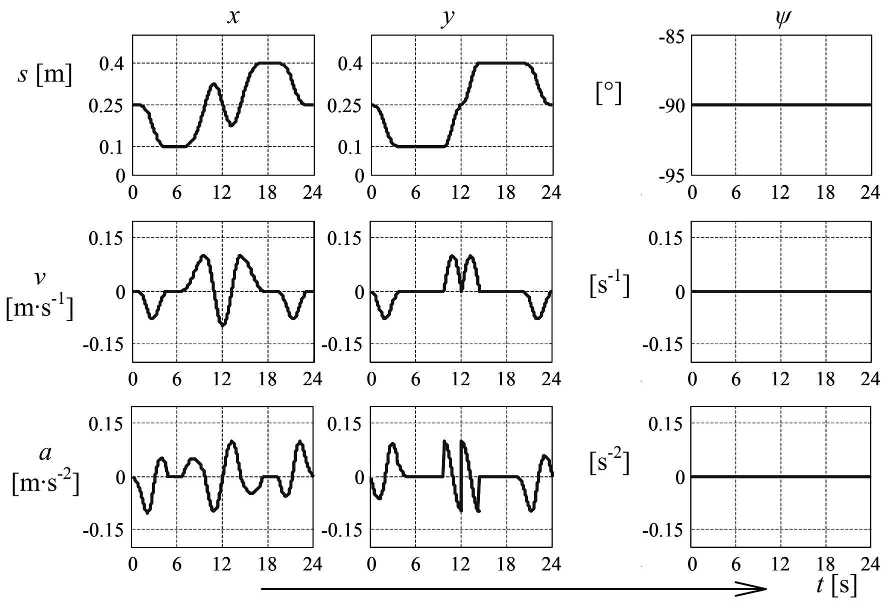

Fig. 3. Kinematical characterizations (s, v, a) of testing “*S*-shaped” trajectory – used for real experiments.

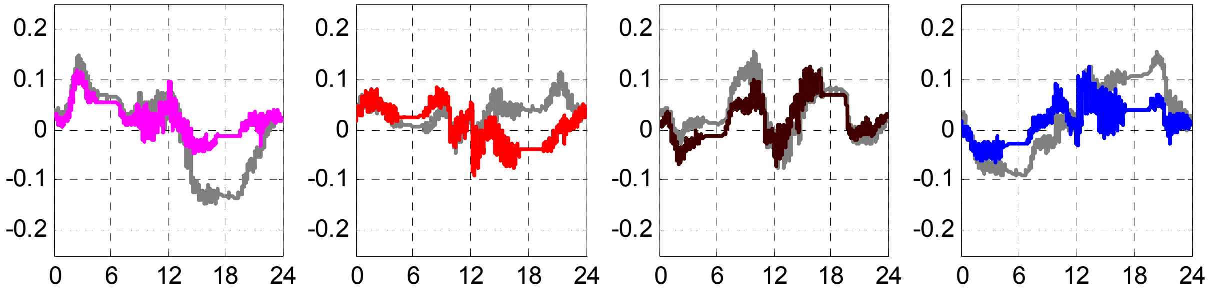

Fig. 4. Comparison of time histories of Sliding Mode Control and PSD control.
(Time histories in the same color mark conventional PSD control).

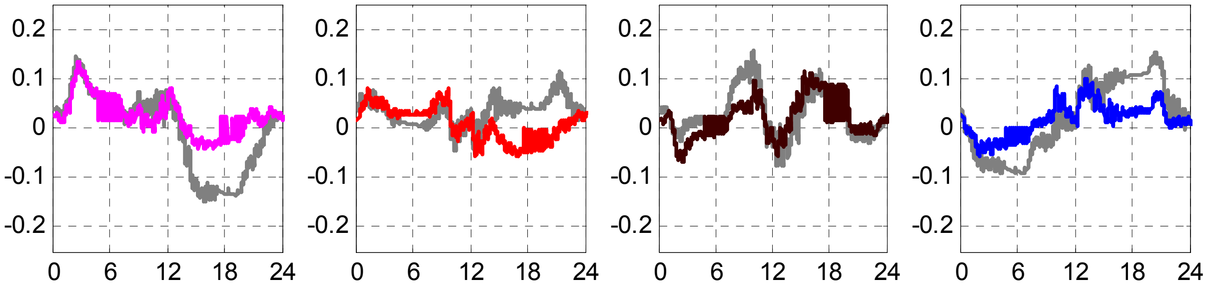

Fig. 5. Comparison of time histories of torques with Predictive Control and PSD control.
(Time histories in the same color mark decentralized PSD control)

The time histories (Fig. 4 and 5) show robot motion along "S - shaped" trajectory in Fig. 3 a Fig. 4. Vertical axes (range -0.2 to 0.2) represent range of torques on motors in [Nm], horizontal axes are time axes scaled in [s].

7. CONCLUSION

The Figures Fig. 4 and 5 compare the properties of conventional PID/PSD control deducing the actions only from the feedback, which cannot be energetically effective. The figures show the advantages of the combination of feedforward action modulation with feedback inaccuracy and disturbance compensation – more effective control strategies SMC and GPC, way with lower energetic demands. It supports the reasons why we should try to replace the classical energetically consuming approaches by model-based strategies e.g. SMC or GPC control. They consider most of the available information on given controlled object, and thus can react more effective in given time.

8. REFERENCES

1. Stejskal V., Valášek M.: Kinematics and Dynamics of Machinery, Marcel Dekker, Inc. 1996.
2. Ordys A., Clarke D.: A State - Space Description for GPC Controllers, Int. J. Systems Sci., vol.24, no.9, 1993, pp. 1727 - 1744.
3. Elmali H., Olgac N.: Sliding mode control with perturbation estimation (SMCPE): a new approach, INT. J. CONTROL, vol.56, no.4, 1992, pp. 923 - 941.
4. Belda K., Böhm J., Valášek M.: State-Space Generalized Predictive Control for Redundant Parallel Robots, Mechanics Based Design of Structures and Machines, Vol. 31, No. 3, 2003, pp. 413-432.
5. Belda K.: Control of Redundant Parallel Structures of Robotic Systems, dissertation, 2002, Czech Technical University in Prague.
6. Valášek M., Steinbauer P.: Nonlinear Control of Multibody Systems, In: Euromech 99, Lisbon 1999, pp. 437 - 444.
7. Lawson Ch., Hanson R.: Solving Least Square Problems, Prentice - Hall, Inc., 1974.

ACKNOWLEDGEMENT

The authors appreciate the kind support

by the grant GAČR (101/03/0620, 2003/2005) – – "Redundant drives and measurement for hybrid machine tools"

and

by the grant GAČR (102/02/0204, 2002/2004) – – "Design of adaptive intelligent controllers".