STUDY OF PREDICTIVE CONTROL ALGORITHMS FOR PARALLEL ROBOT STRUCTURES

BELDA KVĚTOSLAV, BÖHM JOSEF, VALÁŠEK MICHAEL
Department of Adaptive Systems, Institute of Information Theory and Automation,
Academy of Sciences of the Czech Republic,
Pod vodárenskou věží 4, 182 08 Praha 8 - Libeň,
fax: +420-266052068, e-mails: belda@utia.cas.cz, bohm@utia.cas.cz.

Abstract: The parallel robots are promising way to improve the most of mechanical properties, accuracy and speed of machine tools used in industrial production. The parallel robots can be simply understood as movable truss constructions or as movable work platforms supported by a set of parallel arms. This paper focuses on several possible variants of discrete predictive algorithms for design of control actions applied to the robot structures; i.e. absolute, incremental, incremental with nonlinear simulative prediction and predictive algorithm, used for the planning of quadratically-optimal trajectories. The control design is based on mathematical models of the mechanical structures. The models are derived from Lagrange's equations of mixed type (leading to DAEs) and transformed for control design to the classical differential equations (ODEs). The models will be represented in several forms according to requirements of individual approaches. All algorithms will be illustrated by simulation in MATLAB - SIMULINK environment.

Keywords: Generalized Predictive Control, Parallel robot structures, Nonlinear prediction.

1 INTRODUCTION

Nowadays, the future development of industrial robots – machine tools – requires change in their control; i.e. replacement of traditional control represented e.g. by NC systems, which cannot fully utilize mechanical properties of the machine. They are only general approach providing control of the tool drives as separate units, but not an approach solving the control from view of the whole machine system. The decentralization is suitable from view of production line, however not from view of one workplace – of machine tool or machine center.

NC program represented by setting of the sequence of coordinates and technological conditions does not take into account particularity of mechanical structure of the machine. One possible way, relatively accurate, but more sophisticated, is an utilization of model-based control approaches, which could be initially implemented directly to the NC programs as a user-defined functions or later, after demonstration of their functionality, implemented in new adequately designed control systems. At present, such systems are only in laboratories without any suitable interface: user – control system - machine tool. They wait for proper time, when the customers will require them. Till that time, they will be developed locally, as individual applications.

One good example of the tools, which require mentioned high-level control are parallel robots. However, that the parallelism is appeared in robotics in the sixties (Stewart platform, 1965; [1]), their wider development started only in the nineties. The parallel robots are promising way to improve the most of the mechanical properties, accuracy and speed of machine tools. They can be simply understood as movable truss constructions or as movable work platforms supported by a set of parallel arms [2].

This paper will discuss several possible variants of discrete predictive algorithms for design of control actions applied to the parallel robot structures. Specifically, absolute, incremental, incremental with nonlinear simulative prediction and predictive algorithm used for the planning of quadratically-optimal trajectories will be explained. The described control design uses mathematical models of the mechanical structures. The models are derived from Lagrange's equations of mixed type - differential algebraic equations (DAEs), transformed to the ordinary differential equations (ODEs).

2 MODEL-BASED APPROACHES TO CONTROL

The model-based approaches use the model as prior information (feed-forward). It enables to predict future behavior of system. Such a way, the input energy can be optimized considering future requirements.

In this paper, the state-space model is considered. It more clearly expresses the relations in multi-input multi-output structures (MIMO systems). Thus, let us assume for next explanation, that some model of controlled system is given [3], [4], that is linear or it can be linearized [5] and can be transformed to the state-space formulation.

In case of mechanical systems (parallel robots), the initial model is given in a form of system of nonlinear differential equations

$$\ddot{\mathbf{y}} = \mathbf{f}(\mathbf{y}, \dot{\mathbf{y}}) + \mathbf{g}(\mathbf{y})\mathbf{u} \tag{1}$$

written in state-space formulation

$$\begin{aligned} \dot{\mathbf{X}} &= \mathbf{f}(\mathbf{X}) + \mathbf{g}(\mathbf{X})\,\mathbf{u}\,, \quad \mathbf{X} = [\mathbf{y}, \dot{\mathbf{y}}]^T \\ \mathbf{y} &= \mathbf{h}\,\mathbf{X} \end{aligned} \tag{2}$$

The system (2) can be decomposed according to [5] to linear form

$$\begin{aligned} \dot{\mathbf{X}} &= \mathbf{A}(\mathbf{X})\,\mathbf{X} + \mathbf{g}(\mathbf{X})\,\mathbf{u} \\ \mathbf{y} &= \quad \mathbf{h} \quad \mathbf{X} \end{aligned} \tag{3}$$

and discretized for discrete algorithms

$$\begin{aligned} \mathbf{X}(k+1) &= \mathbf{A}\,\mathbf{X}(k) + \mathbf{B}\,\mathbf{u}(k) \\ \mathbf{y}(k) &= \mathbf{C}\,\mathbf{X}(k) \end{aligned} \tag{4}$$

The model (4) is fundamental form for control design based on generalized predictive algorithms.

3 GENERALIZED PREDICTIVE CONTROL (GPC)

Generalized Predictive Control is a multi-step control [6] based on local optimization of quadratic cost function (quadratic criterion)

$$J_k = \sum_{j=No+1}^{N} \left\{ \left(y(k+j) - w(k+j) \right)^T Q_y \left(y(k+j) - w(k+j) \right) \right\} + $$

$$+ \sum_{j=1}^{Nu} \left\{ u(k+j-1)^T Q_u \; u(k+j-1) \right\} \tag{5}$$

The criterion is expressed in step k. N is a horizon of optimization, No is a horizon of initial insensitivity and Nu is a control horizon. Qy and Qu are output and input penalizations and $y\,(k+j)$ and $u\,(k+j-1)$ are input and output values.

The approach combines both feed-forward and feed-back parts. The feed-back, closed from measured outputs, compensates some inaccuracies of mathematical model and certain bounded disturbances. The base of the control is a prediction (computation, estimation) of future output values. They are compared with desired vales in the criterion. The following subsection introduces three ways how to predict future outputs and subsection after deals with minimization of quadratic criterion.

3.1 Predictions of future outputs

The most usual, absolute predictive algorithm arises directly from state-space model (4)

$$
\begin{array}{lllll}
\hat{\mathbf{X}}(k+1) = & \mathbf{A}\ \mathbf{X}(k) + & \mathbf{Bu}(k) & & \\
\hat{\mathbf{y}}(k+1) = & \mathbf{CA}\ \mathbf{X}(k) + \mathbf{C} & \mathbf{Bu}(k) & & \\
\hat{\mathbf{X}}(k+2) = & \mathbf{X}(k) + \mathbf{A} & \mathbf{Bu}(k) & & \\
\hat{\mathbf{y}}(k+2) = & \mathbf{CA}^2\ \mathbf{X}(k) + \mathbf{CA} & \mathbf{Bu}(k) & & \\
\vdots & \vdots & \vdots & \ddots & \\
\hat{\mathbf{X}}(k+N) = & \mathbf{A}^N\ \mathbf{X}(k) + & \mathbf{A}^{N-1}\mathbf{Bu}(k) + \cdots + & \mathbf{Bu}(k+N-1) & \\
\hat{\mathbf{y}}(k+N) = & \mathbf{CA}^N\ \mathbf{X}(k) + & \mathbf{CA}^{N-1}\mathbf{Bu}(k) + \cdots + & \mathbf{CBu}(k+N-1) &
\end{array}
\tag{6}
$$

written in matrix notation is

$$
\hat{\mathbf{y}} = \mathbf{f} + \mathbf{Gu}\Big|_{\substack{\hat{\mathbf{y}}=[\hat{\mathbf{y}}(k+1),\,\hat{\mathbf{y}}(k+2),\,\cdots,\,\hat{\mathbf{y}}(k+N)]^T \\ \mathbf{u}=[\mathbf{u}(k),\,\mathbf{u}(k+1),\,\cdots,\,\mathbf{u}(k+N-1)]^T}}, \quad \mathbf{f} = \begin{bmatrix} \mathbf{CA} \\ \vdots \\ \mathbf{CA}^N \end{bmatrix} \mathbf{X}(k), \quad \mathbf{G} = \begin{bmatrix} \mathbf{CB} & \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{CA}^{N-1}\mathbf{B} & \cdots & \mathbf{CB} \end{bmatrix}
\tag{7}
$$

For second possible way of prediction, let us consider the following incremental algorithm, the model (4) is modified to a form (8), transformed to (9).

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

$$
\begin{aligned}
\begin{bmatrix} \mathbf{X}(k+1) \\ \mathbf{u}(k+1) \end{bmatrix} &= \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{0} & \mathbf{1} \end{bmatrix} \begin{bmatrix} \mathbf{X}(k) \\ \mathbf{u}(k) \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \mathbf{1} \end{bmatrix} \Delta\mathbf{u}(k) \\
\mathbf{y}(k) &= \begin{bmatrix} \mathbf{C} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{X}(k) \\ \mathbf{u}(k) \end{bmatrix}
\end{aligned}
\tag{8}
$$

$$
\begin{aligned}
\overline{\mathbf{X}}(k+1) &= \overline{\mathbf{A}}\ \overline{\mathbf{X}}(k) + \overline{\mathbf{B}}\ \Delta\mathbf{u}(k) \\
\mathbf{y}(k) &= \overline{\mathbf{C}}\ \overline{\mathbf{X}}(k)
\end{aligned}
\tag{9}
$$

The model (9) (= (8), respectively) is extended for discrete integrator (summator) of control actions $\mathbf{u}(k+1) = \mathbf{u}(k) + \Delta\mathbf{u}(k)$. The summator represents one-step delay. This model can be used for control design in similar way as a model for absolute algorithm with exception that resultant control actions are realized in next time instant (k + 1). Expression (9) together with equation $\mathbf{u}(k+1) = \mathbf{u}(k) + \Delta\mathbf{u}(k)$ itself represents full inclusion of integrator, both to model (9) used for control design and also to the real control loop by inserting the same expression $\mathbf{u}(k+1) = \mathbf{u}(k) + \Delta\mathbf{u}(k)$.

The third, incremental algorithm with simulative substitution consider modified version - (4):

$$\begin{aligned}\mathbf{X}(k+1) &= \underbrace{\mathbf{A}(\mathbf{X}(k))\,\mathbf{X}(k)+\mathbf{B}(\mathbf{X}(k))\,\mathbf{u}(k)_{sim}}+\mathbf{B}\Delta\mathbf{u}(k)\\ \mathbf{X}(k+1) &= \qquad\mathbf{X}(k+1)_{sim}\qquad +\mathbf{B}\Delta\mathbf{u}(k)\,,\quad \mathbf{u}(k)=\mathbf{u}(k)_{sim}+\Delta\mathbf{u}(k)\\ \mathbf{y}(k) &= \mathbf{C}\,\mathbf{X}(k)\end{aligned} \tag{10}$$

where $\mathbf{X}(k+1)_{sim}$, $\mathbf{u}(k)_{sim}$ (etc. $\mathbf{X}(k+i)_{sim}$, $\mathbf{u}(k+i-1)_{sim}$; $i = 1,\cdots, N$) is a state and control action given from pre-simulation in considered horizon of some comparable discrete control. The derivation is similar to previous cases – i.e. suitable repetitive substitution for predicted future states and outputs respectively.

$$\begin{bmatrix}\hat{\mathbf{y}}(k+1)\\ \hat{\mathbf{y}}(k+2)\\ \vdots\\ \hat{\mathbf{y}}(k+N)\end{bmatrix}=\begin{bmatrix}\mathbf{C}\,\mathbf{X}(k+1)_{sim}\\ \mathbf{C}\,\mathbf{X}(k+2)_{sim}\\ \\ \mathbf{C}\,\mathbf{X}(k+N)_{sim}\end{bmatrix}+$$

$$+\begin{bmatrix}\mathbf{CB}(\mathbf{X}(k)) & & \cdots & \mathbf{0}\\ \mathbf{C}(\mathbf{A}(\mathbf{X}(k+1)_{sim})+\mathbf{I})\mathbf{B}(\mathbf{X}(k)) & \mathbf{CB}(\mathbf{X}(k+1)_{sim}) & & \vdots\\ \vdots & & \ddots & \\ \mathbf{C}(\mathbf{A}(\mathbf{X}(k+N-1)_{sim})+\cdots+\mathbf{A}(\mathbf{X}(k+1)_{sim})+\mathbf{I})\mathbf{B}(\mathbf{X}(k)) & & \cdots & \mathbf{CB}(\mathbf{X}(k+N-1)_{sim})\end{bmatrix}$$

$$\times\begin{bmatrix}\Delta\mathbf{u}(k)\\ \Delta\mathbf{u}(k+1)\\ \\ \Delta\mathbf{u}(k+N-1)\end{bmatrix} \tag{11}$$

And corresponding matrix notation is expressed as

$$\hat{\mathbf{y}} = \hat{\mathbf{y}}_{sim} + \mathbf{G} \times \Delta\mathbf{u} \qquad \left|\begin{aligned}\hat{\mathbf{y}} &= [\hat{\mathbf{y}}(k+1),\hat{\mathbf{y}}(k+2),\cdots,\hat{\mathbf{y}}(k+N)]^T\\ \Delta\mathbf{u} &= [\Delta\mathbf{u}(k),\Delta\mathbf{u}(k+1),\cdots,\Delta\mathbf{u}(k+N-1)]^T\end{aligned}\right. \tag{12}$$

All introduced predictive algorithms of $\hat{\mathbf{y}}$, arising from the notations (7) and (12), can be generalized via one expression

$$\hat{\mathbf{y}} = \mathbf{f} + \mathbf{G}\mathbf{u} \tag{13}$$

in which the vector **f** and matrix **G** are changed according to type of prediction. For next derivation - - control design, the differences in (13) are not significant and they do not change the derivation.

The implication of described prediction consists in the use. The first, absolute form is the simplest one. The second, incremental form represents integration that can solve steady state error. And third, the most difficult, incremental algorithm with simulative substitution uses prior predicted states from numerical simulation of comparable local discrete control (i.e. control + numerical pre-simulation within considered horizon N). Thus, the new prediction (12) within horizon N is not based only on one model given by $\mathbf{X}(k)$, but it is based on a set of models changing with each step of prediction $(1,\cdots, N)$. The prediction (12) respects by such way the nonlinearities of controlled system.

3.2 Computation of control actions

The control actions are obtained by minimization of quadratic criterion (5). It can be simply rewritten to the following matrix product

$$J_k = [(\hat{\mathbf{y}} - \mathbf{w})^T, \mathbf{u}^T] \begin{bmatrix} \mathbf{Q_y} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q_u} \end{bmatrix} \begin{bmatrix} \mathbf{Q_y} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q_u} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{y}} - \mathbf{w} \\ \mathbf{u} \end{bmatrix} = \mathbf{J}^T \mathbf{J} \tag{14}$$

where $\hat{\mathbf{y}}$ is a vector composed according to (16) (time step k+1, · · ·, k+N), **w** is a vector of desired values, corresponding to vector $\hat{\mathbf{y}}$ and **u** is a vector of designed future inputs, again in discrete time instants for the whole horizon (k, · · ·, N -1). The product (14) is more suitable form that can be decomposed in two parts so-called square roots of the criterion. From mathematical point of view the minimization of square root is more suitable [7]. Let us select square root on the right side (without transpositions) and use the expression for prediction (13)

$$\mathbf{J} = \begin{bmatrix} \mathbf{Q_y} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q_u} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{y}} - \mathbf{w} \\ \mathbf{u} \end{bmatrix} = \begin{bmatrix} \mathbf{Q_y G} \\ \mathbf{Q_u} \end{bmatrix} \mathbf{u} - \begin{bmatrix} \mathbf{Q_y (w - f)} \\ \mathbf{0} \end{bmatrix} \tag{15}$$

J is a column vector, of which Euclidean norm is a cost of the square root of the criterion.

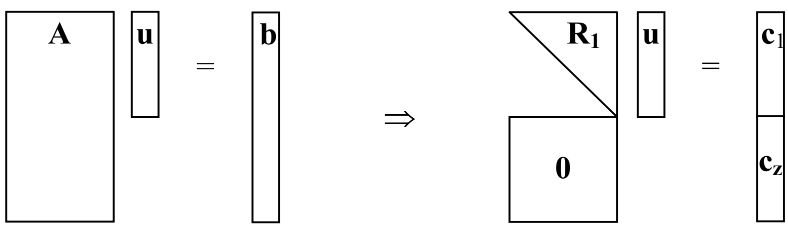
We search for such **u**, which minimizes the square root (15) i.e. the control **u** should minimize the norm |**J**| of criterion. It is fulfilled, if the right side of expression (15) (system with more rows than columns, over-determined system) is annulled.

$$\mathbf{J} = \begin{bmatrix} \mathbf{Q_y G} \\ \mathbf{Q_u} \end{bmatrix} \mathbf{u} - \begin{bmatrix} \mathbf{Q_y (w - f)} \\ \mathbf{0} \end{bmatrix} \overset{!}{=} \mathbf{0}$$

$$\mathbf{A} \quad \mathbf{u} - \quad \mathbf{b} \quad = \mathbf{0} \tag{16}$$

In computation, the orthogonal triangular decomposition [8] is used. It reduces excess rows of matrix **A** [(2·N·i)×(N·i)] and elements of vector **b** [2·N·i] (i is a number of DOF) into upper triangular matrix **R** and vector **c** according to the following scheme:

$$\begin{aligned} \mathbf{A u} &= \mathbf{b} \qquad / \mathbf{Q}^T \quad \text{(multiplication by orthogonal matrix)} \\ \mathbf{Q}^T \mathbf{A u} &= \mathbf{Q}^T \mathbf{b} \\ \mathbf{R u} &= \mathbf{c} \end{aligned} \tag{17}$$



(18)

Vector $\mathbf{c}_z$ is a lost vector, whose Euclidean norm $|\mathbf{c}_z|$ is equal to value of square root $\sqrt{J}$ (i.e. $J = \mathbf{c}_z^T \mathbf{c}_z$). To obtain unknown control **u**, we need only upper part of the system (18)

$$\begin{aligned} \mathbf{R}_1 \mathbf{u} &= \mathbf{c}_1 \\ \mathbf{u} &= (\mathbf{R}_1)^T \mathbf{c}_1 \end{aligned} \tag{19}$$

Since a matrix $\mathbf{R}_1$ is upper triangle, then the control **u** is given directly by back-run procedure.

3.3 Planning of quadratically-optimal trajectories

As one interesting possibility, the predictive control offers, due to its several horizons (N, No, Nu), planning trajectories by record of outputs from simulation. The task is defined as follows: let us have two points – start and end, and at the same time, a path (trajectory) is not conditioned, only end-point must be achieved. In such case, we can use predictive control with specific setting of the output horizons N and No. If we set, that the horizon $N = Nmax$ and $No = N - k$, where k is order of the controlled system, then the quadratic criterion will consider only last k differences among predicted end-point and its reference value. Thus, the matrix **G** and corresponding differences (**w** - **f**) in the criterion (14), are reduced only on their last k rows and elements respectively

$$\left[\begin{array}{cc:c} \multicolumn{2}{c:}{\overline{\mathbf{G}}} & \mathbf{w}-\mathbf{f} \\ \hdashline \mathbf{1} & \mathbf{0} & \mathbf{0} \\ & \ddots & \vdots \\ \mathbf{0} & \mathbf{1} & \mathbf{0} \end{array}\right] = \left[\begin{array}{cccccc:c} \mathbf{CA}^{N-k}\mathbf{B} & \cdots & \mathbf{CB} & \cdots & & \mathbf{0} & (\mathbf{w}-\mathbf{f})_{N-k} \\ \vdots & & & \ddots & & \vdots & \vdots \\ \mathbf{CA}^{N-1}\mathbf{B} & \mathbf{CA}^{N-2}\mathbf{B} & \cdots & \mathbf{CAB} & & \mathbf{CB} & (\mathbf{w}-\mathbf{f})_{N} \\ \hdashline \mathbf{1} & \cdots & & \cdots & & \mathbf{0} & \mathbf{0} \\ \vdots & & \mathbf{1} & & & \vdots & \vdots \\ \mathbf{0} & \cdots & & \cdots & & \mathbf{1} & \mathbf{0} \end{array}\right] \tag{20}$$

Lower unit matrix in (20) corresponds to dimension of input penalization.

Such form, specifically last k rows of matrix **G** and corresponding differences (**w** - **f**), causes quadratic distribution of energy to individual inputs (control actions) within whole horizon $Nmax$. If indicated procedure would be applied, then the control process has no information, in which step should stop. Difference of horizons N and No is still the same. Information on stopping the control process is given by horizon No.

Described sequence represents specific dead-bead control spread within time. Thus, it is not necessary to achieve end point during minimal number of steps (= order of system) in control process, but on the other hand (from reasons of feasibility by drives) it is better to distribute the input energy uniformly without rapid turns in some wider horizon. Its length should arise from technological requirements.

The sequence can be used only once under condition, that the system is linear and horizon No is a little bit lower than horizon N; i.e. value No gives the length of horizon, on which the system should stop after previous $(Nmax - No)$ steps. In case of nonlinear systems, the sequence has to be repeated with progressively shortened horizon N

$$N := Nmax, Nmax - 1, \cdots, Nmin + 1, Nmin \tag{21}$$

where the value $Nmin$ is suitable selected, not exceed number c. 20 ($k < Nmin < 20$). The higher numbers do not improve the process. The repetition provides the changes of model during planning of the trajectory according to real state of the controlled system i.e. it respects nonlinearity by changing of models in compliance with real positions and velocities of the robot.

Illustration will be given in the following section of examples and results.

4 EXAMPLES, RESULTS

For examples and tests, the robotic structures shown in Figure 1 were used.

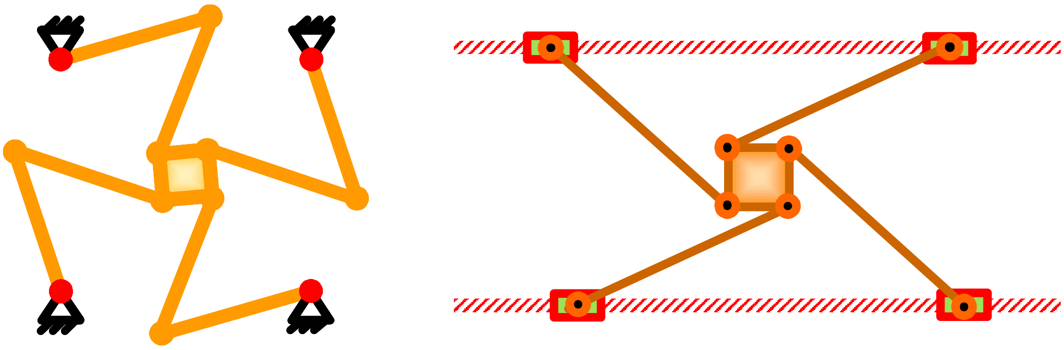


Figure 1 – Examples of parallel planar robotic structures.

The following figures (Figure 2 – Figure 4) illustrate theory of previous sections. Firstly, absolute, incremental and incremental nonlinear algorithms are compared. The robot structure moved along *S*-shaped trajectory. The profiles of velocities are at the bottom parts of Figure 2.

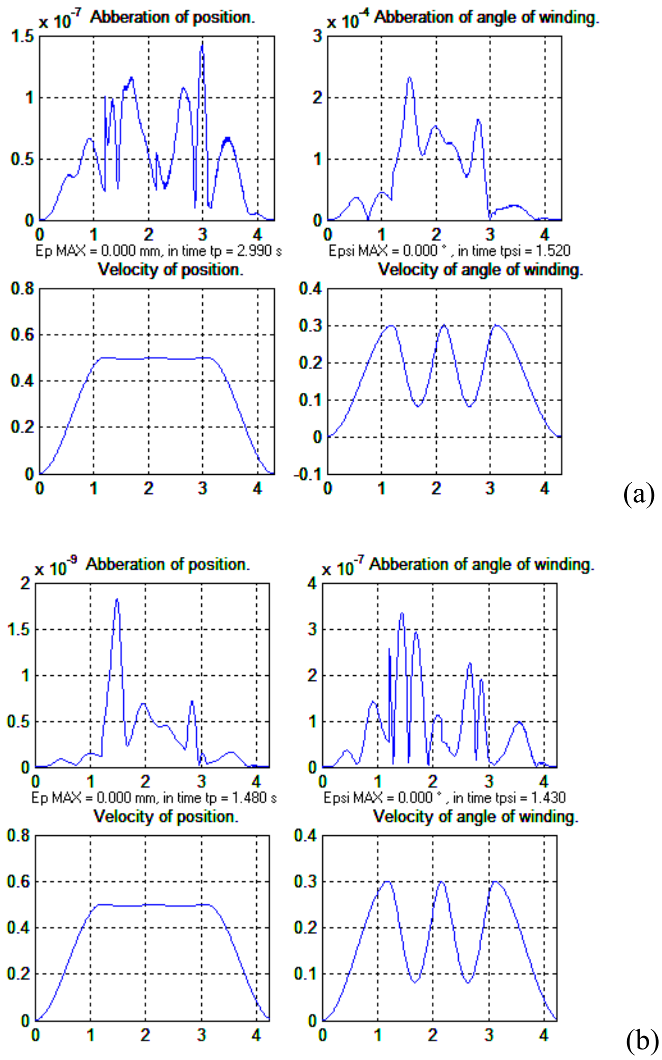


Figure 2 – Comparison of results of absolute (a) and incremental nonlinear (b) algorithms ($Q_y = 1$, $Q_u = 0$).

Figure 3 shows trajectory planning - substitution point to point control manner by specific quadratically-optimal trajectory. The Cartesian coordinates of start-point **A** is $\mathbf{A}_s = [x_s, y_s, \psi_s] = [0.2\text{m}, 0.2\text{m}, -20°]$ and end-point **B** is $\mathbf{B}_e = [x_e, y_e, \psi_e] = [0.8\text{m}, 0.8\text{m}, 20°]$.

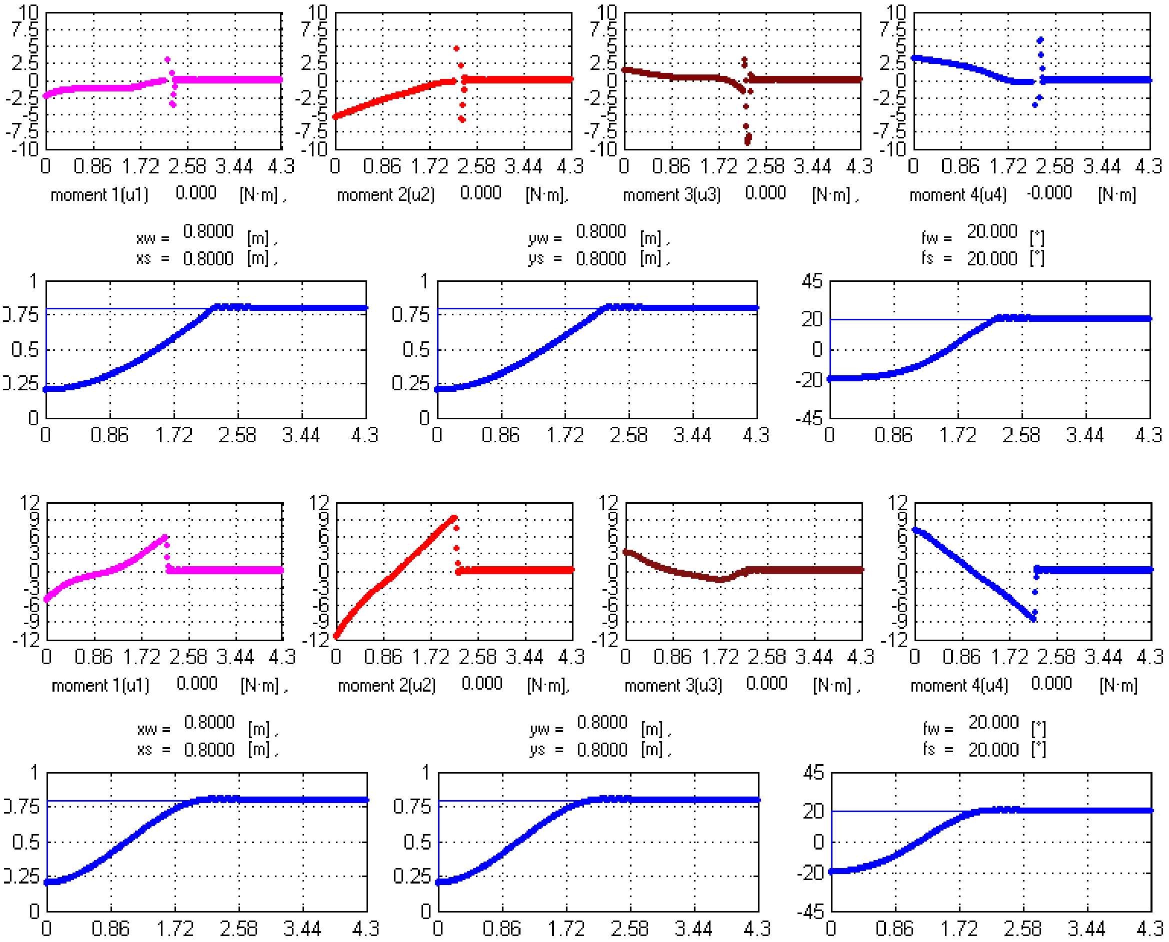
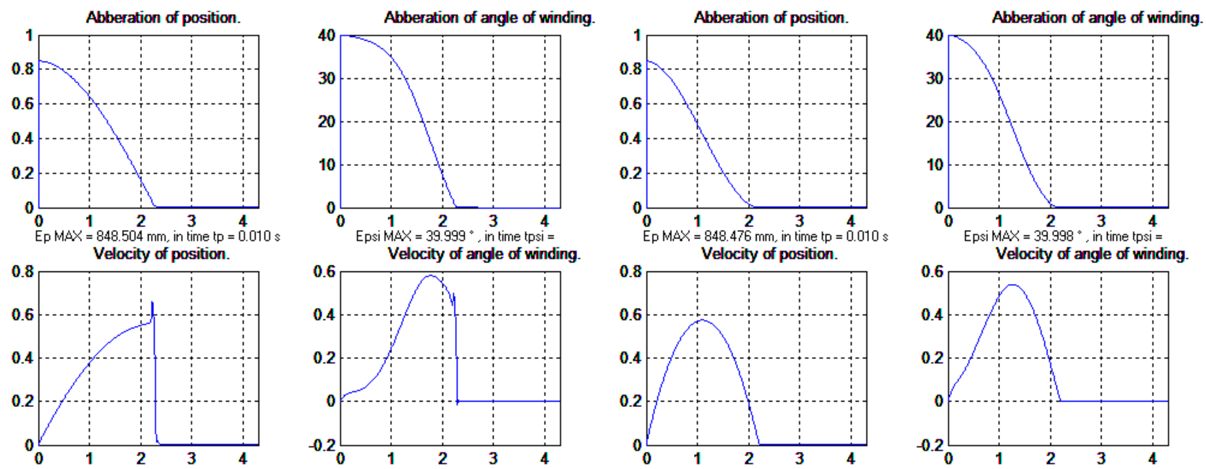


Figure 3 – Planning of quadratically-optimal trajectories – without information on stopping (a), with stopping (b).



(a) (b)

Figure 4 – Time histories during trajectory planning (Figure 3) – without information on stopping (a) and with stopping (b).

5 CONCLUSIONS

The conclusions of the paper, which presents absolute, incremental and incremental nonlinear predictive control algorithms applied to the parallel robotic structures [9], are summarized in the following points.

Presented predictive algorithms allow and solve:

- redundancy of drives in redundant cases, where the actions are not uniquely defined
- possible mutual drive fighting
- run in real time (linear absolute and linear incremental algorithms)
- control of substantially nonlinear systems (nonlinear incremental algorithm)
- integrating character (linear incremental algorithm)
- quadratically-optimal trajectory planning.

References

[1] TSAI, L.-W. *Robot Analysis, The Mechanics of Serial and Parallel Manipulators*, New York: John Wiley & Sons, Inc. 1999, ISBN 0-471-32593-7.

[2] BELDA, K.; BÖHM, J.; VALÁŠEK, M. State-Space Generalized Predictive Control for Redundant Parallel Robots. *Mechanics Based Design of Structures and Machines*, 2003, Vol. 31, No. 3, pp. 413-432, ISSN 1539-7734.

[3] STEJSKAL, V.; VALÁŠEK, M. *Kinematics and dynamics of machinery*. New York: Marcel Dekker, Inc., 1996, 494 pp., ISBN 0-8247-9731-0.

[4] BELDA, K. *Control of Redundant Parallel Structures of Robotic Systems*. Prague 2002, dissertation.

[5] VALÁŠEK, M.; STEINBAUER, P. Nonlinear Control of Multibody Systems, *Euromech 99*, Lisabon, 1999, pp. 437 - 444.

[6] ORDYS, A.; CLARKE, D. A state-Space Description for GPC Controllers. *INT. J. Systems SCI.*, 1993, Vol. 24, No. 9, pp. 1727-1744.

[7] BOBÁL, V.; BÖHM, J., PROKOP, R., FESSL, J. *Praktické aspekty samočinně se nastavujících regulátorů: algoritmy a implementace*, VUT v Brně, 1999, 242 pp., ISBN 80-214-1299-2.

[8] LAWSON, Ch. J.; HANSON, R. J. *Solving Least Square Problems*, Prentice - Hall, Inc. 1974.

[9] NEUGEBAUER, R. (editor): *Development Methods and Application Experience of Parallel Kineamatics.* Reports from the IWU, Vol. 16, IWU Chemnitz 2002, ISBN 3-928921-76-2.

Acknowledgments

This research has been supported by GA ČR (101/03/0620, 2003-2005)
"Redundant drives and measurement for hybrid machine tools".