Range-space predictive control for optimal robot motion

Květoslav Belda, Josef Böhm

***Abstract*—**The paper deals with a simple modification of predictive control for specifically-optimal robot motion. The task of optimal robot motion is solved in many different industrial applications including accurate manipulation and positioning. The modification consists in different definition of requirements for a robot motion. Usually, the robot motion is determined by known trajectory. The range-space modification investigated herein takes into account only range (limits) of required robot movement and its end point. Such approach can just solve manipulation issues, where the accurate achievement of some trajectory is not important, but the robot has to move through known corridor described by appropriate output range (limits) and has to reach some defined end point. The modification generates optimal control actions, which meet mentioned requirements. The explanation is documented by several examples of described control process applied to one advanced robot structure based on parallel kinematical concept.

***Keywords*—**Accurate Manipulation, Industrial Robotics, Predictive Control, Range-Space Control.

I. Introduction

INDUSTRIAL applications especially robotic applications include a lot of operations, in which amount of different displacements of materials, semi or final products, tools or active elements as sensors and cameras have to be provided. These operations have a manipulation character, which does not require the accurate achievement of some predetermined trajectory. Thus, in most of cases, the trajectory is not important, but on the other hand a fulfillment of some permitted output range is required. The range or set of ranges are given by definite limits following from space configuration of individual robot, manipulated object and other possible obstacles occurred in the robot workspace.

Solution of manipulation issues is important particularly at parallel robots [1], where the workspace is more limited and achievement of optimal robot movement is more challenging than in case of conventional open-loop robots [2].

Manuscript received December 10, 2007; revised form received March 7, 2008. This work was supported by Grant Agency of the Czech Republic under Grant 102/06/P275 (2006-08) 'Model-based control of mechatronic systems for robotics'.

K. Belda is with the 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, Czech Republic (corresponding author to provide phone: +420-266-052-310; fax: +420-266-052-068; e-mail: belda@utia.cas.cz).

J. Böhm is with the Department of Adaptive Systems, Institute of Information Theory and Automation, Academy of Sciences of the Czech Republic, Prague, Czech Republic (e-mail: bohm@utia.cas.cz).

This paper specifically concerns with one solution of described task based on predictive control [3], which takes into account robot dynamics. Proposed solution represents simple way managing both trajectory planning with time parameterization and real control. This way can replace special CAD software intended for trajectory planning or can save time spending in this software.

At the paper beginning, the manipulation problem is formulated and general model of robot dynamics is defined. In next sections, the predictive control and its range-space modification [10] is explained. Finally, the paper concludes by several illustrative application examples of proposed solution.

II. Problem Formulation

As mentioned, the manipulation belongs to the most frequently occurred operations in many different robotic applications. From control point of view, it can be formulated as a controlled movement of robot gripper (robot end-effector) from start to end point through defined free range Fig. 1.

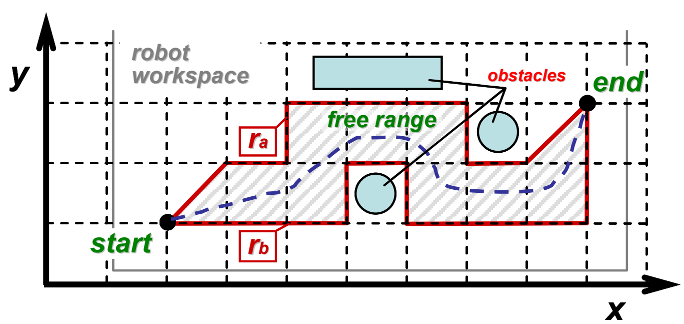


Fig. 1 Free range for possible robot movement

Since, there is no further (exact) specification for the robot movement, the real robot trajectory (trace of realized robot movement) should begin at start point and achieve required end point taking into account ranges of workspace during the robot displacement. It is possible to generate infinite number of such trajectories, but they need not be optimal in view of considered robot.

Optimal robot motion can be obtained by specific modification of predictive control design, which considers required points, free (permitted) ranges and model of robot dynamics. The control actions provide smooth motion in spite of sharp ranges.

III. Model of Robot Dynamics

The model of robot dynamics expresses time relations of external (drive actions) and internal (reactions, inertias) forces and torques, all usually relating to the position of robot gripper (end-effector). The model makes possible to predict future robot behavior and to optimize its further movement.

The robots themselves (not only parallel configurations) generally represent multi-body systems usually with multi-input multi-output character. Their elements are determined by weights and inertia moments (mass parameters) of individual bodies and a set of lengths and physical coordinates (geometrical parameters). By coordinates, the dynamics can be straightforwardly described by Lagrange's equations [4]. The equation type depends on number of used coordinates in relation to the number of degrees of freedom (DOF). In general, in case of using physical coordinates, which number is higher than DOF, then Lagrange's equations are mixed type and lead to the system of differential algebraic equations – DAE (1) and (2)

$$\mathbf{M}\ddot{\mathbf{s}} - \mathbf{\Phi}_s^T \boldsymbol{\lambda} = \mathbf{g} + \mathbf{T}\mathbf{u} \tag{1}$$

$$\mathbf{f}(\mathbf{s}) = \mathbf{0} \tag{2}$$

where $\mathbf{M}$ is a mass matrix, $\mathbf{s}$ is a vector of physical coordinates, $\mathbf{\Phi}_s$ is a Jacobian, $\boldsymbol{\lambda}$ is a vector of Lagrange's multipliers, $\mathbf{g}$ is a vector of other internal relations, matrix $\mathbf{T}$ connects inputs $\mathbf{u}$ to appropriate differential equations and $\mathbf{f}(\mathbf{s}) = \mathbf{0}$ represents algebraic equations of geometrical constraints.

As mentioned, the model (1) is a DAE system, since physical coordinates generally need not represent independent coordinate system. The number of coordinates is usually greater than number of DOF. However, the model (1) can be transformed to a system of ordinary differential equations, i.e. transformed to independent coordinate space corresponding to the number of DOF. The transformation reduces the number of differential equations and mainly removes redundant Lagrange's multipliers, connected with structural forces and structural relations; i.e. the transformation provides zeros of term $\mathbf{\Phi}_s^T \boldsymbol{\lambda}$ and reduces physical coordinates $\mathbf{s}$ only to independent coordinates $\mathbf{y}$ agreeing with number of DOF.

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

$$\mathbf{\Phi}_s \mathbf{R} = \mathbf{R}^T \mathbf{\Phi}_s^T = \mathbf{0} \tag{3}$$

and arising also from coordinate transformation (4):

$$\mathbf{s} = \mathbf{s}(\mathbf{y}) \;\Rightarrow\; \dot{\mathbf{s}} = \frac{\partial \mathbf{s}}{\partial \mathbf{y}}\dot{\mathbf{y}} = \mathbf{R}\dot{\mathbf{y}} \;\Rightarrow\; \ddot{\mathbf{s}} = \dot{\mathbf{R}}\dot{\mathbf{y}} + \mathbf{R}\ddot{\mathbf{y}} \tag{4}$$

Now, if expression (4) is inserted in (1) and obtained equation is multiplied by transposed Jacobian matrix $\mathbf{R}^T$, then the resultant system of equations – pure equations of motion have the following form:

$$\mathbf{R}^T\mathbf{M}\mathbf{R}\ddot{\mathbf{y}} + \mathbf{R}^T\mathbf{M}\dot{\mathbf{R}}\dot{\mathbf{y}} = \mathbf{R}^T\mathbf{g} + \mathbf{R}^T\mathbf{T}\mathbf{u} \tag{5}$$

They can be transformed and simplified to a form with isolated second derivative

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

where $\mathbf{f}(\mathbf{y},\dot{\mathbf{y}})$ represents robot dynamics and $\mathbf{g}(\mathbf{y})$ is an input matrix of control actions. The model (6) can be rewritten in the state-space formula:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}) + \mathbf{G}(\mathbf{x})\mathbf{u} \\ \mathbf{y} &= \mathbf{H}\mathbf{x} \end{aligned} \tag{7}$$

where $\mathbf{x}$ is a state vector defined as $\mathbf{x} = [\mathbf{y}, \dot{\mathbf{y}}]^T$. This formula is simpler and more transparent relating to multi-input multi-output character of robot. To have finite computation time of the control and to provide uniformly distributed points of robot trajectory, the model (7) is discretized. However, before model discretization, it is necessary to linearize nonlinear vector function $\mathbf{f}(\mathbf{x})$, in order that resultant model corresponds to usual state-space formulation. Matrix $\mathbf{G}(\mathbf{x})$, although is also nonlinear in relation to the robot state $\mathbf{x}$, does not need to be linearized. The nonlinear function $\mathbf{f}(\mathbf{x})$ in (7) can be decomposed according to [12], to have linear form. Then, the model is written as follows

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}(\mathbf{x})\mathbf{x} + \mathbf{G}(\mathbf{x})\mathbf{u} \\ \mathbf{y} &= \;\; \mathbf{H} \;\; \mathbf{x} \end{aligned} \tag{8}$$

The model (8) including decomposition $\mathbf{A}(\mathbf{x})\mathbf{x}$ for $\mathbf{f}(\mathbf{x})$ describes the robot dynamics identically as model (6) or (7). The individual elements of matrices $\mathbf{A}(\mathbf{x})$ and $\mathbf{G}(\mathbf{x})$ have to be recomputed on-line for appropriate robot state $\mathbf{x}$. The use of decomposition described in [12] is shown e.g. in [8] or [9].

Then, the discretization of (8) can be done by usual way via expansion of exponential functions [11]. The resultant model has following standard form:

$$\begin{aligned} \mathbf{x}_{(k+1)} &= \mathbf{A}_{(k)}\mathbf{x}_{(k)} + \mathbf{B}_{(k)}\mathbf{u}_{(k)} \\ \mathbf{y}_{(k)} &= \;\; \mathbf{C} \;\; \mathbf{x}_{(k)} \end{aligned}, \quad \mathbf{x}_{(k)} = \begin{bmatrix} \mathbf{y}_{(k)} \\ \dot{\mathbf{y}}_{(k)} \end{bmatrix} \tag{9}$$

which represents initial model for control design.

IV. Predictive Control

The choice of suitable control depends on character of given control process and used means of control. In robotics, in the most of cases, the control process (controlled robot movement) is required to be accurate, robust and effective both in respect time and energy consumption [1].

The robots were and are usually intended for flexible production, therefore their control algorithms have to cope with energy optimization and saving in working time. These algorithms should optimize control actions in relation to future robot behavior. To managing mentioned conditions, predictive control is offered [3] (Fig. 2).

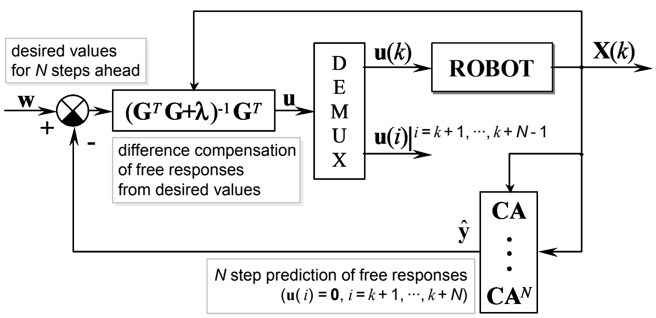


Fig. 2 Principal scheme of predictive control

Predictive control, as well as linear quadratic control [5], combines feedback and feedforward. It represents a multi-step strategy, which optimizes control actions using quadratic criterion within some finite horizon.

In the criterion, future expected robot behavior like states with appropriate user requirements are compared. The future states are expressed by equations of predictions within predefined finite horizon. The equations arise from robot model. The following subsection will show their composition.

A. Equations of Predictions

Equations of predictions represent from control point of view the expression of feed-forward for some given prediction horizon N. They form the main base determining dominant part of control actions. Using discrete state-space form (9), the equations of predictions have the following form [3]:

$$\begin{aligned}
\hat{\mathbf{x}}_{(k+1)} &= \mathbf{A}\,\mathbf{x}_{(k)} + \mathbf{B}\,\mathbf{u}_{(k)} \\
\hat{\mathbf{y}}_{(k+1)} &= \mathbf{C}\,\mathbf{A}\,\mathbf{x}_{(k)} + \mathbf{C}\,\mathbf{B}\,\mathbf{u}_{(k)} \\
&\vdots \qquad \vdots \qquad \ddots \\
\hat{\mathbf{x}}_{(k+1)} &= \mathbf{A}^{N}\,\mathbf{x}_{(k)} + \mathbf{A}^{N-1}\,\mathbf{B}\,\mathbf{u}_{(k)} + \cdots + \mathbf{B}\,\mathbf{u}_{(k+N-1)} \\
\hat{\mathbf{y}}_{(k+1)} &= \mathbf{C}\,\mathbf{A}^{N}\,\mathbf{x}_{(k)} + \mathbf{C}\,\mathbf{A}^{N-1}\,\mathbf{B}\,\mathbf{u}_{(k)} + \cdots + \mathbf{C}\,\mathbf{B}\,\mathbf{u}_{(k+N-1)}
\end{aligned} \tag{10}$$

The system (10) can be expressed by matrix notation

$$\hat{\mathbf{y}} = \mathbf{f} + \mathbf{G}\,\mathbf{u} \tag{11}$$

where
$$\mathbf{f} = \begin{bmatrix} \mathbf{CA} \\ \vdots \\ \mathbf{CA}^{N} \end{bmatrix} \mathbf{x}_{(k)}$$

and
$$\mathbf{G} = \begin{bmatrix} \mathbf{C} & \mathbf{B} \cdots & \mathbf{0} \\ \vdots & \ddots & \vdots \\ \mathbf{C}\,\mathbf{A}^{N-1} & \mathbf{B} \cdots & \mathbf{CB} \end{bmatrix}$$

which simplifies the expressions in the quadratic criterion.

The equations are composed by repetitive substitution of unknown future states for states determined from one topical state and state-space model (9). The state-space model is considered to be constant for composition of the equations for considered time step and corresponding horizon N, called horizon of predictions.

Control actions $\mathbf{u}$, which are occurred in (10) and (11), represent unknown parameters. Thus, (10) or its condensed notation (11) means functional expression of dependency of future states $\hat{\mathbf{y}}$ on control actions $\mathbf{u}$ for whole horizon N. Unknown actions are computed by minimization of quadratic criterion.

B. Quadratic Criterion

As was mentioned, quadratic criterion is fundamental for determination of control actions. Usually, it is expressed in the following form

$$J_k = \sum_{j=1}^{N} \{ \| (\hat{\mathbf{y}}_{(k+j)} - \mathbf{w}_{(k+j)})\, \mathbf{Q}_{\mathbf{y}} \|^2 + \| \mathbf{u}_{(k+j-1)}\, \mathbf{Q}_{\mathbf{u}} \|^2 \} \tag{12}$$

where $\mathbf{Q}_y$ and $\mathbf{Q}_u$ are penalizations for control errors and actions, respectively. They balancing terms of the criterion; and $\mathbf{w}_{(k+j)}$ are desired values. The criterion (12) can be profitably expressed in condensed matrix notation as well:

$$J_k = [(\hat{\mathbf{y}} - \mathbf{w})^T \;\; \mathbf{u}^T] \begin{bmatrix} \mathbf{Q_y} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q_u} \end{bmatrix}^T \begin{bmatrix} \mathbf{Q_y} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q_u} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{y}} - \mathbf{w} \\ \mathbf{u} \end{bmatrix} \tag{13}$$

The future outputs $\hat{\mathbf{y}}$ in (13) are substituted by matrix notation (11). The criterion form (13) is, at first, more transparent for minimization process, and in the second place it represents suitable initial form for stable mathematical solution. The control actions are determined by minimization of the criterion as it is shown in next section.

C. Minimization of the Criterion

Used algorithm for minimization of the criterion is crucial for considering of different (additional) control requirements. The simple algorithm based on determination of control actions as a local minimum search [3], [11] leads to the following expression

$$\mathbf{u} = (\mathbf{G}^T\mathbf{G} + \boldsymbol{\lambda})^{-1}\,\mathbf{G}^T\,(\mathbf{w} - \mathbf{f}) \tag{14}$$

However, this algorithm is limited by matrix inversion. Different, more general way is algorithm based on square-root minimization. It arises from condensed notation (13) represented as follows:

$$J_k = \mathbf{J}\,\mathbf{J} \tag{15}$$

from which only one part (square-root) is sufficient to be minimized:

$$\mathbf{J} = \begin{bmatrix} \mathbf{Q_y} & \mathbf{0} \\ \mathbf{0} & \mathbf{Q_u} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{y}} - \mathbf{w} \\ \mathbf{u} \end{bmatrix} \tag{16}$$

After substitution of $\hat{\mathbf{y}}$, the square-root is expressed

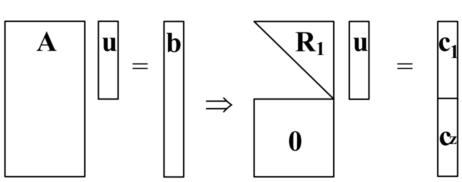
$$\mathbf{J} = \begin{bmatrix} \mathbf{Q_y\,G} \\ \mathbf{Q_u} \end{bmatrix} \mathbf{u} - \begin{bmatrix} \mathbf{Q_y}(\mathbf{w} - \mathbf{f}) \\ \mathbf{0} \end{bmatrix} \tag{17}$$

Its minimization leads to the solution of algebraic equations for unknown control actions

$$\begin{aligned} \begin{bmatrix} \mathbf{Q_y\,G} \\ \mathbf{Q_u} \end{bmatrix} \mathbf{u} - \begin{bmatrix} \mathbf{Q_y}(\mathbf{w} - \mathbf{f}) \\ \mathbf{0} \end{bmatrix} &= \mathbf{0} \\ \mathbf{A}\quad \mathbf{u} - \quad \mathbf{b} \quad &= \mathbf{0} \end{aligned} \tag{18}$$

Suitable method of solving set of algebraic equations is QR decomposition [6] based on Householder algorithm [7]. It reduces excess rows of matrix $\mathbf{A}$ and elements of vector $\mathbf{b}$ into upper triangular matrix $\mathbf{R}$ and vector $\mathbf{c}$, or matrix $\mathbf{R}_1$ and vector $\mathbf{c}_1$, respectively, according to the following computational scheme:

$$\begin{aligned} \mathbf{A\,u} &= \mathbf{b} \qquad \times \mathbf{Q}^T \\ \mathbf{Q}^T\mathbf{A\,u} &= \mathbf{Q}^T\mathbf{b} \\ \mathbf{R\,u} &= \mathbf{c} \end{aligned} \tag{19}$$



(20)

To obtain unknown control actions $\mathbf{u}$, then only upper part of the system (20) is need

$$\begin{aligned} \mathbf{R}_1\,\mathbf{u} &= \mathbf{c}_1 \\ \mathbf{u} &= (\mathbf{R}_1)^T\,\mathbf{c}_1 \end{aligned} \tag{21}$$

Since a matrix $\mathbf{R}_1$ is upper triangle, then the control $\mathbf{u}$ is given directly by backward calculation procedure.

Obtained vector $\mathbf{u}$ represents control actions for whole horizon N. However, only first appropriate actions are really applied to the robot.

The criterion (12), respectively the form (13) can be simply extended for additional terms representing additional requirements on control. This property will be considered in range-space modification.

The described process of minimization is repeated in every time step for appropriately updated model within defined horizon of prediction.

V. Range-Space Modification

Range-space modification follows from the demand to simplify coding (planning) the robot trajectory in cases, when the trajectory is not strictly determined by some accurate geometrical shape or path. In such cases, there is only information on location of obstacles in a robot workspace and furthermore knowledge of start and required end points [10], altogether define rough free corridor for safe robot motion, i.e. free range-space (Fig. 3).

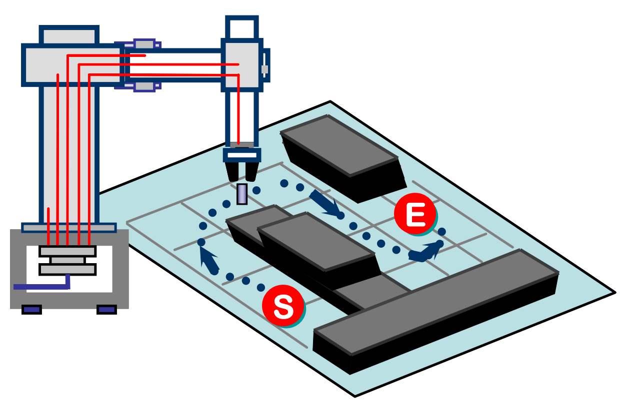


Fig. 3 Example of free corridor for safe robot motion

The range-space modification of predictive control can serve either as on-line control for slower movements or as a fast off-line simulative planner of smooth trajectories. To design adequate control actions via range-space modification, it is necessary to make the following steps:

- define free movement ranges: splitting the robot workspace or robot neighborhood within view in small simple elementary areas inclusive obstacle highlighting;
- select suitable penalty matrices: balancing of importance of the corridors in given time for robot motion.

Then, it is possible to minimize modified quadratic criterion, which provides mentioned assignment.

A. Modified Quadratic Criterion

The quadratic criterion used in predictive control design can cope with different control requirements due to its flexibility. It considers robot dynamics included in predictions for defined horizon. By this, the optimal distribution of input energy is provided. The quadratic criterion for range-space modification is modified relating to the new required ranges [10], which substitute usual desired values

$$J_k = \sum_{j=1}^{N} \left(\left\| \overline{\mathbf{Q}}_{ra} (\hat{\mathbf{y}}_{(k+j)} - \mathbf{r}_{a(k+j)}) \right\|^2 + \left\| \overline{\mathbf{Q}}_{rb} (\hat{\mathbf{y}}_{(k+j)} - \mathbf{r}_{b(k+j)}) \right\|^2 + \left\| \overline{\mathbf{Q}}_{\mathbf{u}} \mathbf{u}_{(k+j-1)} \right\|^2 \right) \tag{22}$$

The criterion can be again expressed in advantageous matrix notation

$$J_k = \mathbf{J}\,\mathbf{J} = \|\mathbf{J}\|^2 = \left\| \begin{bmatrix} \mathbf{Q}_{ra} & \cdots & \mathbf{0} \\ \vdots & \mathbf{Q}_{rb} & \vdots \\ \mathbf{0} & \cdots & \mathbf{Q}_{\mathbf{u}} \end{bmatrix} \begin{bmatrix} \hat{\mathbf{y}} - \mathbf{r}_a \\ \hat{\mathbf{y}} - \mathbf{r}_b \\ \mathbf{u} \end{bmatrix} \right\|^2 \tag{23}$$

The modified criterion leads to the same procedure of its minimization indicated in subsection IV.*C*, only matrices have different types.

In the criterion, there occur two new terms, which correspond to differences from the limits of free (permissible) ranges. Furthermore, there are also two new output penalizations $\overline{\mathbf{Q}}_{ra}$ and $\overline{\mathbf{Q}}_{rb}$.

In usual control process with standard criterion, the only one appropriate penalization $\mathbf{Q}_{\mathbf{y}}$ is selected as identity matrix. However, the new penalizations need not be constant and equal identity matrices. The selection of penalization will be outlined thereinafter in subsection V.*C*.

B. Definition of Free Movement Ranges

Definition of free movement ranges arises from specific simple envelope of possible free space for robot movement, which excludes all obstacles and other limits of given robot workspace. This envelope is divided into uniform square or triangle segments.

The density of segmentation follows from rough distance of the start and end points and selected sampling period. For small periods, the segment density should be higher and vice versa.

Thus, the free ranges are determined for individual segments. They represent upper and lower limits of the segments. There are important two segments: start and end. At the beginning, the free range spreads out from start point (triangle segment) and at the end it narrows to end point as it is indicated in Fig. 4.

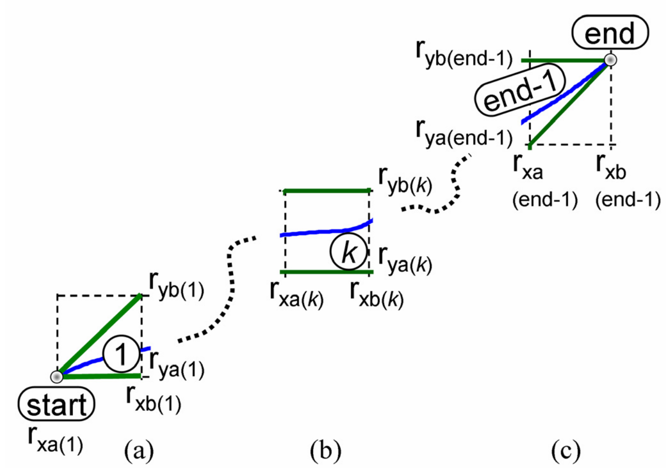


Fig. 4 Free ranges: (a) acceleration period (start), (b) general state (process), (c) braking period (end)

Example of one short definition of free ranges is illustrated in Fig. 5.

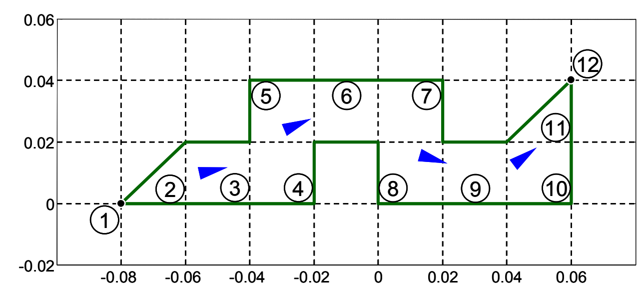


Fig. 5 Example of definition of free ranges

Corresponding values of upper and lower limits of individual segments in Fig. 5 are enumerated in Table I. The all values are repeated in number equaled length of horizon N .

TABLE I
DEFINITION OF FREE RANGES:
$\mathbf{r}_a = [\mathrm{xa}, \mathrm{ya}, \psi]$ AND $\mathbf{r}_b = [\mathrm{xb}, \mathrm{yb}, \psi]$.

%	1	2	3	4	5	6	…
xa=[	-0.08	-0.08	-0.06	-0.04	-0.04	-0.02	…
xb=[	-0.08	-0.06	-0.04	-0.02	-0.02	0.00	…
ya=[	0.00	0.00	0.00	0.00	0.02	0.02	…
yb=[	0.00	0.02	0.02	0.02	0.04	0.04	…

…	7	8	9	10	11	12
…	0.00	0.00	0.02	0.04	0.04	0.06];
…	0.02	0.02	0.04	0.06	0.06	0.06];
…	0.02	0.00	0.00	0.00	0.02	0.04];
…	0.04	0.02	0.02	0.02	0.04	0.04];

Horizon length and rate of elementary areas determine time needed for robot displacement from start to end point through the free ranges.

C. Selection of Penalty Matrices

Selection of penalty matrices for outputs $\overline{\mathbf{Q}}_{ra}$ and $\overline{\mathbf{Q}}_{rb}$, and for inputs $\overline{\mathbf{Q}}_{u}$ is depended on considered sampling period of the control (speed of control process), selected ranges of segments and horizon of predictions. They have the following structures

$$\overline{\mathbf{Q}}_{ra} = \begin{bmatrix} Qx_{ra} & 0 & 0 \\ 0 & Qy_{ra} & 0 \\ 0 & 0 & Q\psi_{ra} \end{bmatrix}, \quad \overline{\mathbf{Q}}_{rb} = \begin{bmatrix} Qx_{rb} & 0 & 0 \\ 0 & Qy_{rb} & 0 \\ 0 & 0 & Q\psi_{rb} \end{bmatrix},$$
$$\overline{\mathbf{Q}}_{u} = \begin{bmatrix} Qu_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & Qu_m \end{bmatrix} \tag{24}$$

The structures show details, but it is enough to set only multiplicative constants. They follow from dynamic relations of given robot. Usual adequate selection is $\overline{\mathbf{Q}}_{ra} = \overline{\mathbf{Q}}_{rb} = k_r\ \mathbf{I}$ and $\overline{\mathbf{Q}}_{u} = k_u\ \mathbf{I}$, where $\mathbf{I}$ is identity matrix. The lower values k_r and k_u still keeping control stability lead to the best results.

VI. Illustrative Examples

In this section, several simulative examples will demonstrate application of range-space control on one given planar parallel robot 'Moving Slide' considered as basis for simple top milling machine.

A. Considered Robot Structure

Considered robot, illustrated in Fig. 6, represents horizontal planar parallel mechanism, which includes 4 x Rotational + Prismatic + Rotational joints. Moreover, it is redundantly actuated. There are four drives for only three degrees of freedom. Note, this feature furthermore improves robot stiffness.

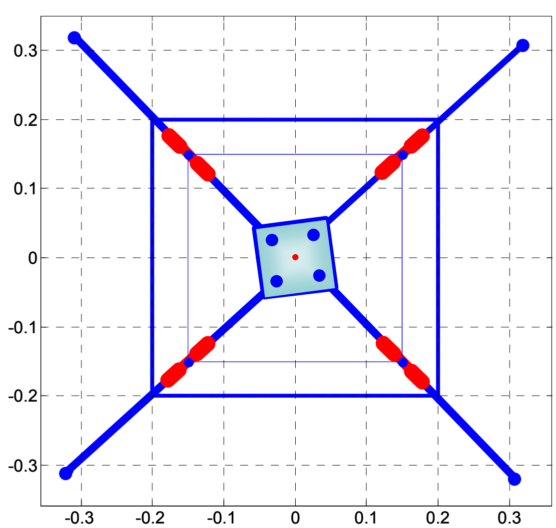


Fig. 6 Scheme: 4RPR parallel robot 'Moving Slide'

B. Experiments

The aim of experiments was to control robot through given free ranges and to achieve given end points. The following figures demonstrate such process.

The first, Fig. 7, shows realization of free ranges from Fig. 5, defined by Table I; the points mark the appropriate times.

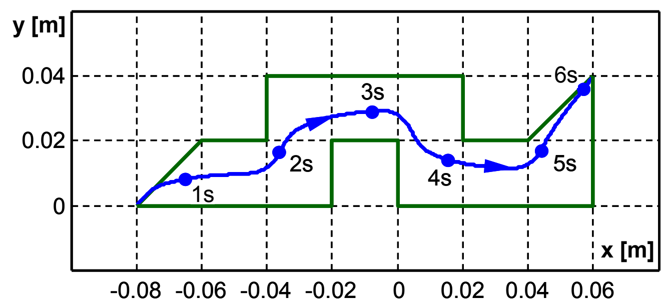


Fig. 7 2D plan of realized control process

The time histories of the corresponding control actions are drawn in Fig. 8. In it, the braking at the end of movement is well perceptible. In some cases, it can be a problem. The robot by its own inertial energy can cross (or miss) the required end point and can continue a little bit over and thereafter stop near the required point. To damp this undesired property, it is necessary to spread several last segments in their higher number or to adapt the penalization matrices.

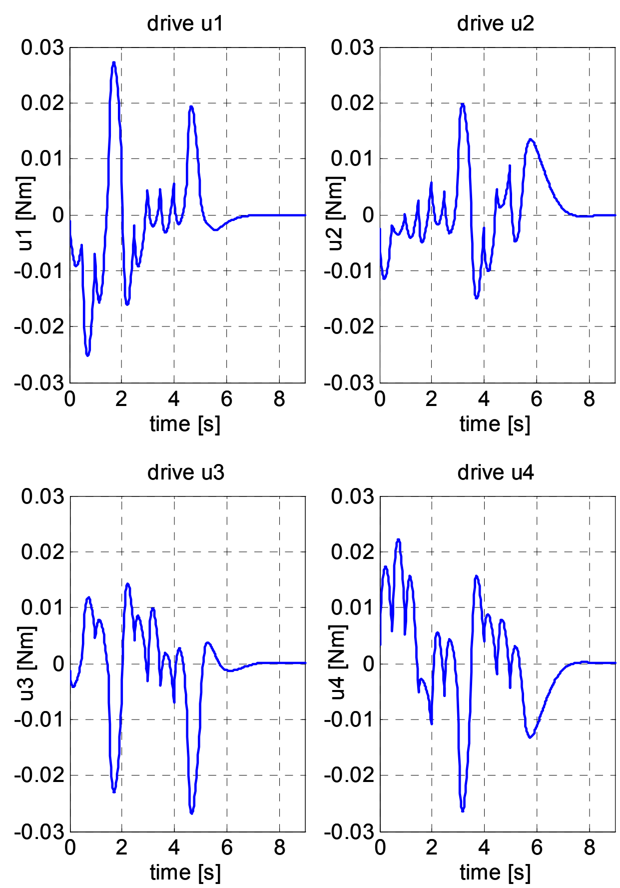


Fig. 8 Time histories of control actions for Fig. 7

The further figures (Fig. 9 and Fig. 10) show other different examples of robot movement.

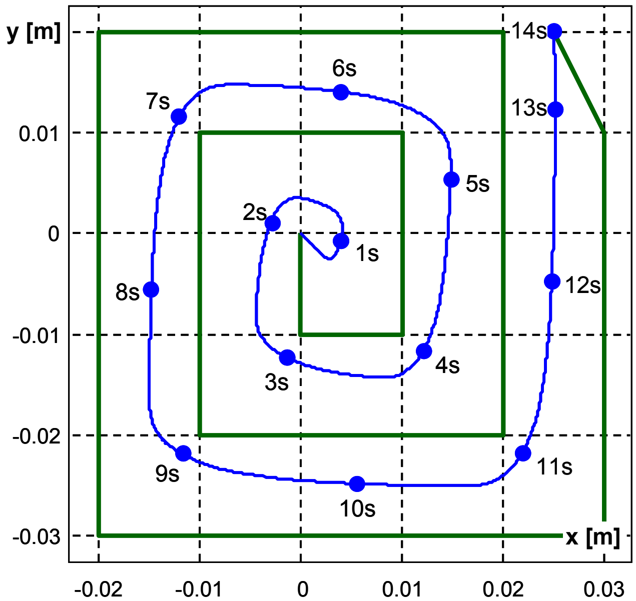


Fig. 9 Example of simple spiral trajectory

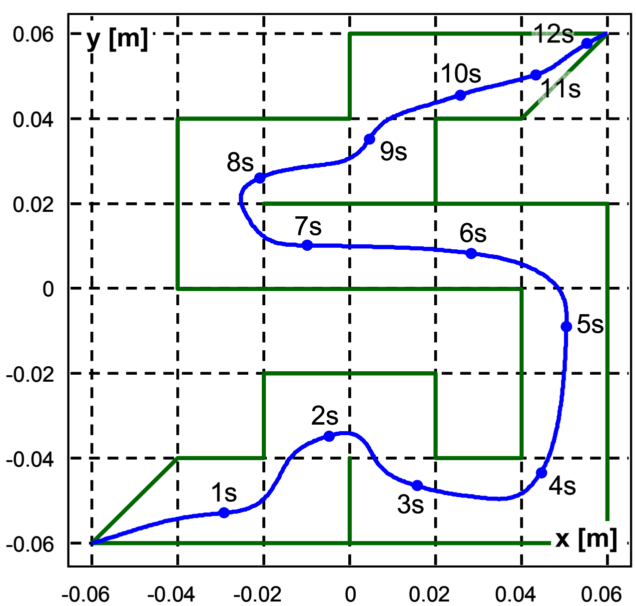


Fig. 10 Example of more complicated trajectory

As was mentioned in subsection V.*B*, by the selection of control horizon and adjustment of elementary area rates, the speed of the robot movement may be regulated.

In figures above, which represent *xy*-graphs, it is perceptible that the movement speed is more or less uniform expect for robot accelerating, braking in sharp turns and braking before the end points. The acceleration or braking stages in figures are indicated by distance lengthening or shortening, whereas the time difference stays the same.

VII. Conclusion

In the paper, the range-space modification of predictive control was introduced. It can solve not only mentioned manipulation issues, but also some camera scan of some scene given by ranges (corridors) or can serve after some adjustments for on-line optimal path control of mobile robots.

In general, the predictive control approach belongs to the powerful and flexible control strategies, which are promising to solve different requirements of industrial applications.

References

[1] L.-W. Tsai, Robot Analysis, *The Mechanics of Serial and Parallel Manipulators*, John Wiley and sons, Inc., New York, 1999.

[2] L. Sciavicco, B. Siciliano, *Modeling and control of robot manipulators*, The McGraw-Hill Companies, Inc. New York 1996.

[3] A. Ordys, D. Clarke, A State-Space Description for GPC Controllers, *Int. J. Systems SCI*. Vol. 24, No. 9, 1993, pp. 1727-1744.

[4] V. Stejskal, M. Valášek, *Kinematics and Dynamics of Machinery*, Marcel Dekker, New York, 1996.

[5] V. Bobál, J. Böhm, J. Fessl, J. Macáček, *Digital Self-tuning Controllers*, Springer, London, 2005.

[6] Ch. Lawson, R. Hanson, *Solving Least Square Problems*, Prentice - Hall, Inc., 1974.

[7] H. G. Golub, Ch. F. L. Van, *Matrix Computations*, The Johns Hopkins Univ. Press, 1989.

[8] K. Belda, J. Böhm, M. Valášek, Predictive Control Applied to Planar Parallel Robots. *Preprints of the 16th World Congress of the International Federation of Automatic Control. 16th IFAC World Congress*, Prague 2005, 6 pp.

[9] K. Belda, J. Böhm, P. Píša, Predictive control for modern industrial robots. *Proc. of Int. Conf. on Informatics in Control, Automation and Robotics*, INSTICC Press, 2005, pp. 1-8.

[10] K. Belda, J. Böhm, Predictive Control of redundant Parallel Robots and Trajectory Planning. *Proceedings of the Parallel Kinematic Machines in Research and Practice*. PKS 2006. Fraunhofer IWU, Chemnitz, Germany 2006, pp. 497-513.

[11] K. Belda, *Control of Parallel Robotic Structures Driven by Electromotors*. Ph.D. dissertation, CTU-FEE, Prague 2005, 114 pp.

[12] M. Valášek, P. Steinbauer, Nonlinear Control of Multibody Systems, *Euromech 1999*, Lisabon, pp. 437-444.

Květoslav Belda received M.Sc. degree in automatic control and engineering informatics in 1999 and Ph.D. degree in technical cybernetics in 2002 and Ph.D. in electrotechnology and materials in 2005 from Czech Technical University in Prague.

He is a research associate in the Department of Adaptive Systems at the Institute of Information Theory and Automation, Academy of Sciences of the Czech Republic. His main interest is research and development of model-based control strategies for mechatronic systems especially in robotic applications.

Josef Böhm received M.Sc. degree in 1964 and Ph.D. degree in 1972 from Faculty of Electrical Engineering of the Czech Technical University in Prague.

He is a research fellow in the Department of Adaptive Systems at the Institute of Information Theory and Automation, Academy of Sciences of the Czech Republic. In the long term, he specializes in problems of adaptive control, especially in synthesis of control based on minimization of quadratic criterion (LQ). He engages both in theoretical fundamentals and in algorithmic issues with respect to realization of practical requirements and given constraints.
Dr. Böhm is an outside academic pedagogue at the faculties of the Czech Technical University in Prague and a member of IFAC committee for control.

For more information, please visit our website: http://as.utia.cas.cz/asc.