Control Principles of Stationary Articulated Robots Used in Cyber-Physical Factories

Květoslav Belda
dept. of Adaptive Systems, The Czech Academy of Sciences
Institute of Information Theory and Automation
Pod Vodárenskou věží 4, 182 00 Prague 8, Czech Republic
belda@utia.cas.cz, https://orcid.org/0000-0002-1299-7704

Lukáš Venkrbec
dept. of Technical Studies
The College of Polytechnics Jihlava
Tolstého 16, 586 01 Jihlava, Czech Republic
venkrbec@vspj.cz, https://orcid.org/0000-0002-7823-294X

***Abstract*—This paper deals with control principles and means, used for research and development purposes supporting and improving motion control of industrial stationary articulated robots. These robots represent the main automation elements on all levels of today's industrial productions considering all technologies such as manipulation, welding, assembling, painting and others. In conjunction with conveyor belts, they are essential for cyber-physical factories built on concepts of Industry 4.0. The related modelling principles are proposed and explained with respect to used control approaches conventional and advanced including software tools supporting the programming, rapid prototyping, verification and commissioning.**

***Index Terms*—industrial robotics, real-time experiments, motion control, model predictive control**

I. Introduction

Articulated robots belong to a specific class of robots widely used in today's industrial production [1], [2]. Considering their mass employment, the control principles used in their control systems represent continuously a subject for testing, investigation and improvement [3], [4]. It follows from a permanent effort to reduce an accumulation of infinitesimal redundant energy inputs that mean needless additional costs [5], [6].

This paper focuses on modelling tools to improve model-based control design. In Section II., the considered representative robot structure is described. Section III. summarises mathematical model synthesis. Then, Section IV. introduces control concepts both conventional cascade configurations and advanced model predictive procedures in hierarchical arrangement combining global centralised and local distributed design [8]. The following Sections V. and VI. focus on current software tools and their use for research covering both simulation and real-run of motion control of usual industrial robots.

II. Considered Robot Structure

In this paper, an industrial robot Mitsubishi RV-4FL-D (Fig. 1) is considered for model synthesis, simulations and real experiments [7] in a cyber-physical factory, Fig. 2. It represents a stationary articulated structure with 6 degrees of freedom (DOF) that enables user to set and control x, y and z positions and A_x, B_y and C_z orientations of tool centre point (TCP).

This work was supported by The Czech Academy of Sciences, Institute of Information Theory and Automation and The College of Polytechnics Jihlava under project No. 23-04676J of the Czech Science Foundation: Controllable gripping mechanics: Modelling, control and experiments.

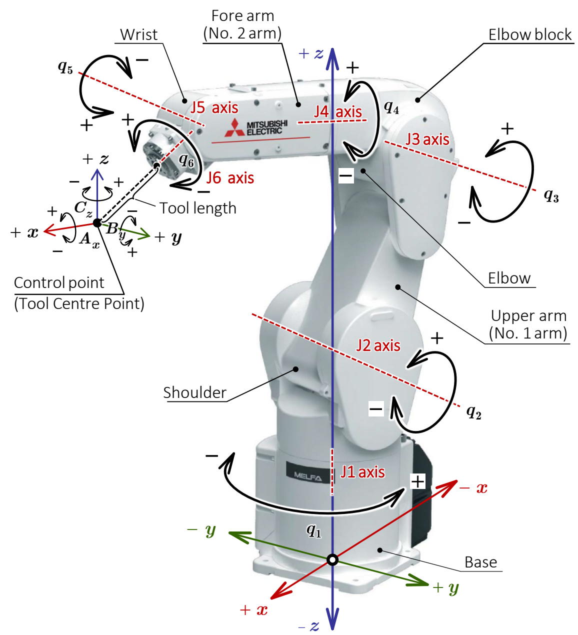

Fig. 1. Robot RV-4FL-D with 6 DOF and its coordinates [7].

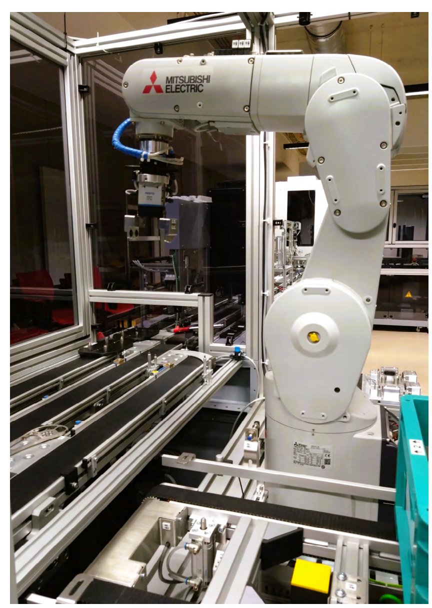

Fig. 2. Robot RV-4FL-D in cyber-physical factory.

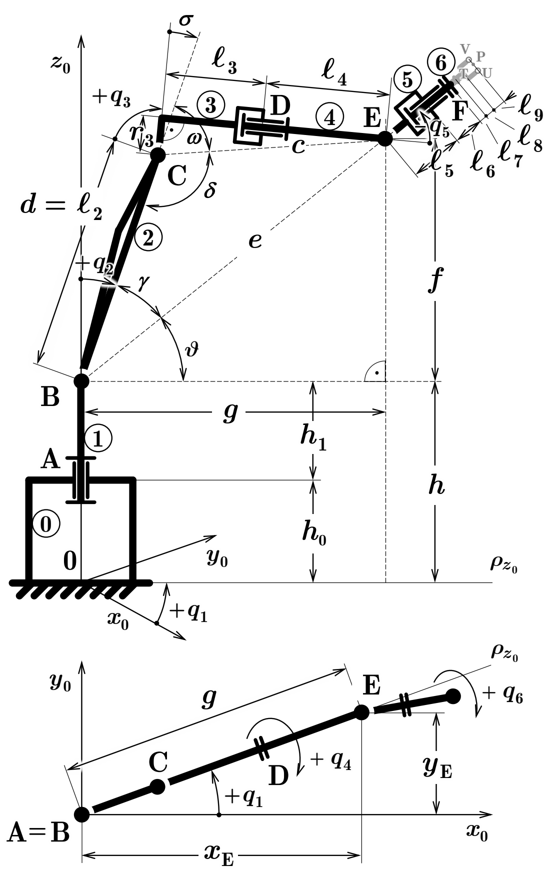

Fig. 3. 2D views of robot structure for kinematic analysis.

III. Mathematical Model Synthesis

This section shows the main points of synthesis of a suitable mathematical model for substitution of reality (simulation of real robot) and as well as a model for control design. It includes the synthesis of both forward and inverse problems for the robot kinematics and dynamics.

A. Model of Robot Kinematics

The kinematic model arises from Denavit-Hartenberg (DH) concept that defines the transformations between joint coordinate space (drive space) and operational coordinate space (tool centre point (TCP), robot end effector) in both directions: forward ($q \to y$) and inverse ($y \to q$), where joint and operational coordinates q and y are defined as follows [1]

$$q = [\,q_1,\, q_2,\, q_3,\, q_4,\, q_5,\, q_6\,]^T,\;\; y = [\,x,\, y,\, z,\, A_x,\, B_y,\, C_z\,]^T \quad (1)$$

1) Forward Problem:

For the articulated robot with 6 DOF, the forward problem represents a specific combination of three basic motions and three basic rotations as follows:

$$\begin{aligned} F = {} & T_0^1\;\, T_1^2\,(q_1)\;\, T_2^3\,(-q_2)\;\, T_3^{\bar{3}}\,(-q_3)\;\, T_{\bar{3}}^4(\,90^\circ) \\ & T_4^5(q_4)\;\, T_5^6(q_5)\;\, T_6^F(q_6)\;\, r_0 \end{aligned} \quad (2)$$

where $F = [\,x_F,\, y_F,\, z_F,\, 1]^T$ and $r_0 = [0,\, 0,\, 0,\, 1]^T$.

The individual motions and rotations are realised by transformation matrices expressed by one general matrix as follows

$$\begin{bmatrix} cos\, q_i & -cos\,\alpha_i\, sin\, q_i & sin\,\alpha_i\, sin\, q_i & a_i\, cos\, q_i \\ sin\, q_i & cos\,\alpha_i\, cos\, q_i & -sin\,\alpha_i\, cos\, q_i & a_i\, sin\, q_i \\ 0 & sin\,\alpha_i & cos\,\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

The matrices contain constant and/or variable parameters q_i, d_i, a_i and α_i - so called D-H parameters relating to a kinematic chain of the robot [1]. Their values are listed in the TABLE I. The number of variable rotations q_i is 6 according to DOF of the considered robot. Other indicated constants h_0, h_1, ℓ_2, r_3, ℓ_3, ℓ_4, ℓ_5 and ℓ_{6-9} are length parameters such as vertical distances, axis offsets and lengths of robot arms. Furthermore, the joint between 'Elbow ver. (= vertical)' and 'Elbow block hor. (= horizontal)' is fixed as immovable right-angle joint $\bar{3}$.

TABLE I
D-H Parameters

Robot body names	D-H Parameters			
	q_i	d_i	a_i	α_i
Base (0 - 1)	0°	h_0	0	0°
Shoulder (1 - 2)	q_1	h_1	0	90°
Upper arm (2 - 3)	$90^\circ - q_2$	0	ℓ_2	0°
Elbow ver. (3 - $\bar{3}$)	$-q_3$	0	r_3	90°
Elbow block hor. ($\bar{3}$ - 4)	90°	ℓ_3	0	0
Fore arm (4 - 5)	$-90^\circ - q_4$	ℓ_4	0	-90°
Wrist (5 - 6)	q_5	ℓ_5	0	90°
Flange (6 - T)	q_6	ℓ_6	0	0°

2) Inverse Problem:

The inverse problem can be solved either numerically as a solution of the transformations expressed by (2) or analytically. Here, the analytical solution is demonstrated using geometrical relations depicted in Fig. 3

$$q_1 = \operatorname{atan2}(y_E,\, x_E) \quad (4)$$

$$q_2 = \frac{\pi}{2} - (\gamma - \vartheta) \quad (5)$$

$$q_3 = \frac{3}{2}\,\pi\, - \beta - \gamma \quad (6)$$

$$f = z_E - h = z_E - (h_0 + h_1) \quad (7)$$

$$g = \sqrt{x_E^2 + y_E^2},\;\; e = \sqrt{f^{\,2} + g^{\,2}} \quad (8)$$

$$c = \sqrt{r_3^{\,2} + (\ell_3 + \ell_4)^2} \quad (9)$$

$$\gamma = \arccos \frac{d^{\,2} + e^{\,2} - c^{\,2}}{2\, d\, e} \quad (10)$$

$$\delta = \arccos \frac{c^{\,2} + d^{\,2} - e^{\,2}}{2\, c\, d} \quad (11)$$

$$\vartheta = \operatorname{atan2}(f,\, g),\;\; \omega = \operatorname{atan2}(\ell_3 + \ell_4, r_3) \quad (12)$$

To obtain coordinates of point E for specific cases, let point E be determined using appropriate transformation chains based on orientation angles Ax, By, Cz from end points especially F (Flange) or tool points T, U, V and P - see [9].

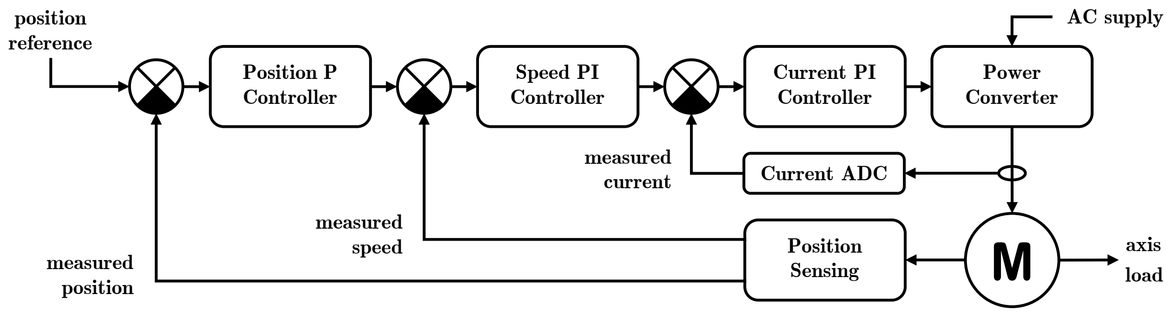

Fig. 4. Conventional independent PID-based cascade control for individual joints/axes/drives.

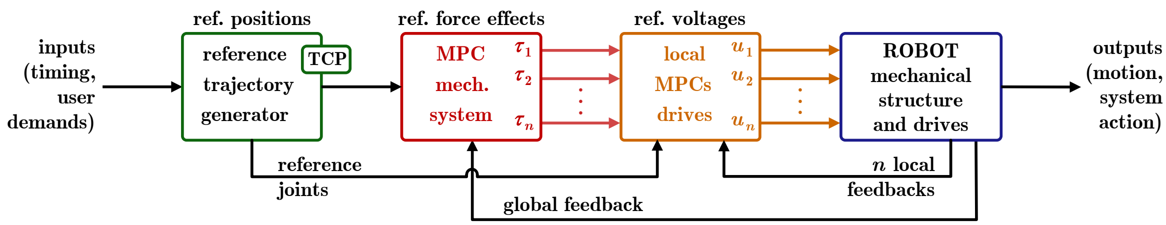

Fig. 5. Novel hierarchical concept of model predictive control for robot dynamics and robot drives.

B. Model of Robot Dynamics

The dynamic model can be given by Lagrange mechanics using Lagrange equations of second type:

$$\frac{d}{dt}\left(\frac{\partial\mathcal{L}}{\partial\dot q}\right)^T-\left(\frac{\partial\mathcal{L}}{\partial q}\right)^T=\xi\,,\quad \mathcal{L}=V(q,\dot q)-P(q) \tag{13}$$

$$\text{i.e.}\quad \frac{d}{dt}\left(\frac{\partial V}{\partial\dot q}\right)^T-\left(\frac{\partial V}{\partial q}\right)^T+\left(\frac{\partial P}{\partial q}\right)^T=\xi \tag{14}$$

where ξ are generalised force effects, V is kinematic energy:

$$V=\sum_{i=1}^{n}V_i=\frac{1}{2}\sum_{i=1}^{n}\left(v_i^T\,m_i\,v_i+\omega_i^T\,\mathcal{I}_i\,\omega_i\right) \tag{15}$$

The kinetic energy V in (15) for translation and rotation along the principal axes can be simply expressed as follows

$$V=\frac{1}{2}\sum_{i=1}^{n}\left(m_i\,v_i^2+I_i\,\omega_i^2\right) \tag{16}$$

where n is a number of robot bodies and quantities m_i, I_i, v_i and ω_i are weights, moments of inertia, translation and angular speeds, respectively. P is potential energy:

$$P=\sum_{i=1}^{n}P_i=\sum_{i=1}^{n}m_i\,g\,z_i \tag{17}$$

where quantities z_i represent appropriate vertical distances of centres of gravity of individual bodies and g is gravitational acceleration. Expressing the terms of (14) leads to the dynamic equations of motion as follows (inverse dynamics) [1]:

$$M(q,\,\dot q)\,\ddot q+N(q,\,\dot q)\,\dot q+g(q)=\tau \tag{18}$$

or (forward dynamics):

$$\ddot q=-M^{-1}\,N\,\dot q-M^{-1}\,g+M^{-1}\,\tau \tag{19}$$

$$\ddot q=-M^{-1}\,N\,\dot q+u \tag{20}$$

where modification $u=-M^{-1}g+M^{-1}\tau$ removes effects of gravitation from the model and simplifies control design. The expression (19) can be done in continuous or discrete state-space forms

$$\left.\begin{array}{lcl}\dot x=A(x)\,x+B\,u & \rightarrow & x_{k+1}=A(x_k)\,x_k+u_k\\ y=C\,x & \rightarrow & y_k=C\,x_k\\ \tau=M\,u+g & \rightarrow & \tau_k=M\,u_k+g\end{array}\right\} \tag{21}$$

The equations in (21) are prepared for simulation and for control design as well.

IV. Control Concepts and Principles

This section deals with specific representatives of non-model and model-based control approaches used and usable for application to industrial stationary (not only articulated) robots. Note that the success and accuracy of the robot motion depend on the used robot model and the reference trajectory.

A. Conventional Cascade Control

Continuous or discrete PID controllers belong to non-model-based control. Their usual basic form is the following

$$u(t)=K_p\left(e(t)+\frac{1}{T_i}\int_0^t e(t)\,\mathrm{d}t+T_d\,\dot e(t)\right) \tag{22}$$

and its discrete incremental form is:

$$\begin{aligned}u_k=u_{k-1}+K_p\Bigg(&(e_k-e_{k-1})\\&+\frac{1}{T_i}\,e_k\,T_s+T_d\,\frac{e_k-2\,e_{k-1}+e_{k-2}}{T_s}\Bigg)\end{aligned} \tag{23}$$

This form for drive control, i.e. actuation of individual robot joints, is modified to the cascade configuration [8]. For motion control of TCP, the cascade control includes positional, speed and current loops, see Fig. 4.

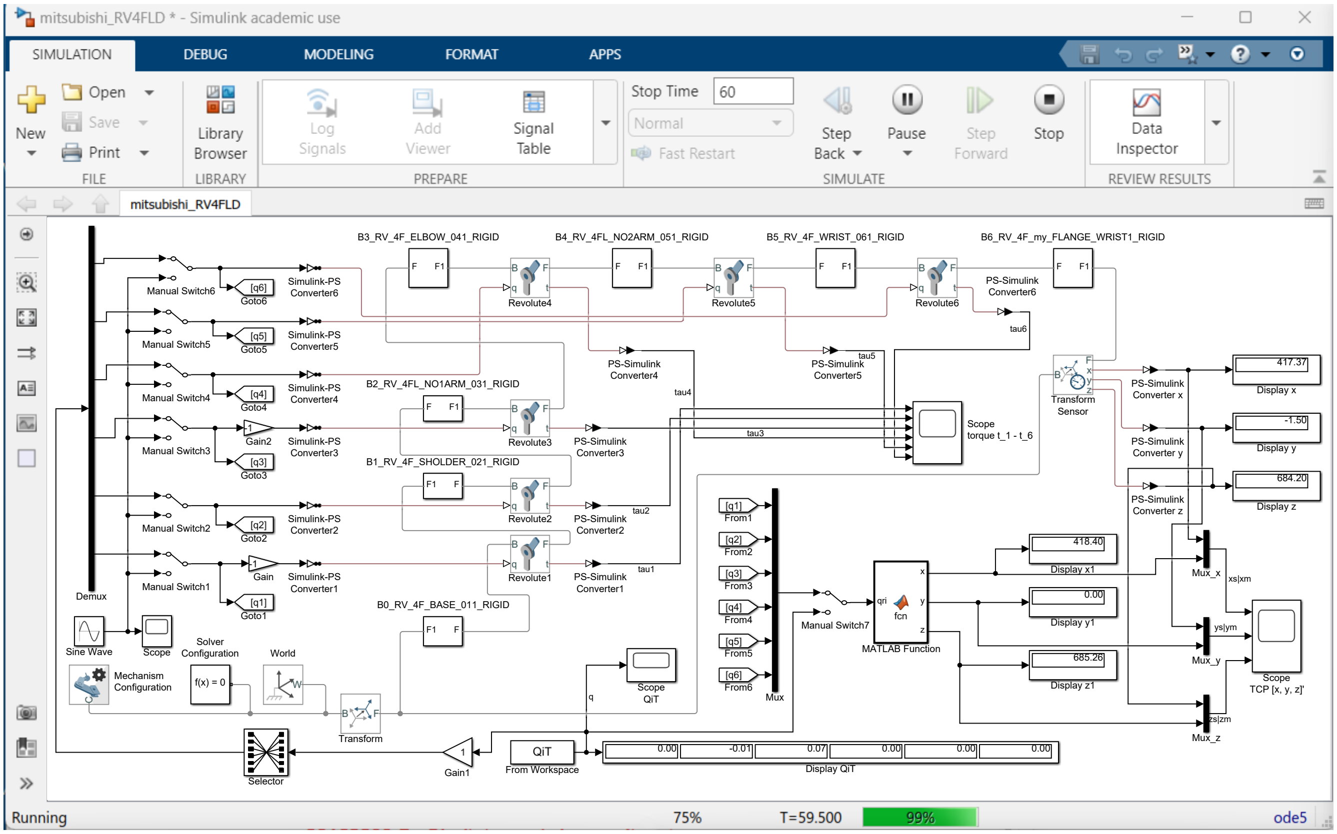

Fig. 6. Simulink model of Mitsubishi robot RV-4FL-D based on Simscape Multibody Library.

The mentioned concept represents still the independent (distributed, local) control approach from the robot point of view. All internal interactions given by a robot dynamic behaviour are considered as external disturbances.

B. Advanced Model Predictive Control

The discrete model predictive control is the representative of model-based control with a specific optimality criterion:

$$\min_{\Delta U_k} J_k\,(\hat{Y}_{k+1},\, W_{k+1},\, \Delta U_k) \tag{24}$$

subject to the state-space model (21) and constraints: $y_{min} \leq y_{j+1} \leq y_{max}$ and $u_{min} \leq u_j \leq u_{max}$, where $\hat{Y}_{k+1}$, W_{k+1} and ΔU_k are vectors of output predictions, reference values and increments of searched control actions $u_k = u_{k-1} + \Delta u_k$, respectively; $j = k, \cdots, k+N$ and N is a prediction horizon. The cost behaviour can be expressed by quadratic function such as:

$$J_k = ||Q_{YW}\,(\hat{Y}_{k+1} - W_{k+1})||_2^2 + ||Q_{\Delta U}\,\Delta U_k||_2^2 \tag{25}$$

Fig. 5 shows the novel concept of the application of predictive control. One is a global slower form with optimisation running online [10] and next is a local fast form employing pre-computed control laws [8]. However, both types arise from the same computation principle.

V. Software Tools

This section introduces suitable software tools, combination of which enables a user to obtain detailed robot models, to simulate models of robot, to realise real-run control and data acquisition from the physical robot motion. Outside of the mentioned tools, other modelling software such as NX or Inventor and others should be considered. All these tools serve for determination of realistic (ideal) physical parameters such as moments of inertia, weights, lengths and positions of centres of gravity that all are important for the construction of an adequate mathematical model.

A. MATLAB/Simulink Simscape Multibody Lib - MathWorks

Simscape Libraries, especially Simscape Multibody library [11] included in Simulink, serve for simulations to verify derived models and 3D CAD models. In this paper, the Multibody library was used for simulations of dynamic behaviour of the robot; see Fig. 6 – Simulink model based on blocks of Multibody library and Fig. 7 – 'Mechanics Explorer' involving imported 3D CAD model.

B. MATLAB Robotics System Toolbox - MathWorks

Similar to the previous section, the plain MATLAB environment with the Robotics System Toolbox [12] is suitable for simulations with a wider function set as well. It enables users to have all procedures in hands, i.e. from trajectory planning, control design and its verification, see Fig. 8.

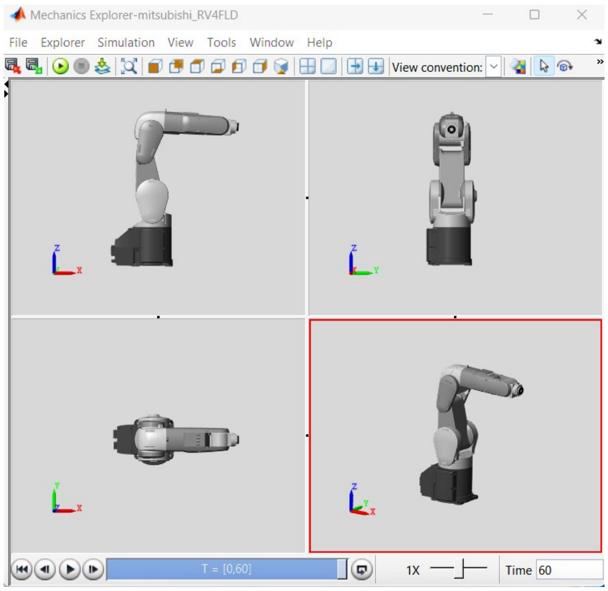

Fig. 7. Mechanics Explorer with 3D model of Mitsubishi robot RV-4FL-D.

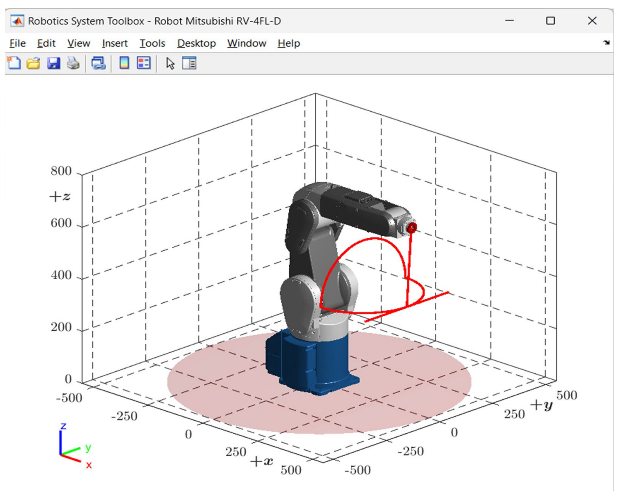

Fig. 8. Figure with 'rigidBodyTree' class – Robotics System Toolbox.

C. RT Toolbox - Mitsubishi Electric

RT Toolbox represents a suitable software tool that enables user to program robot motion, to verify it including 3D visualisation and to execute real control of the specific class of industrial robots including data acquisition options [13]. In this paper, this tool was used for verification of designed motion trajectory and for real experiments on the described robot Mitsubishi RV-4FL-D.

In this paper, MATLAB was used for the import of 3D CAD model of the robot [14], for the implementation of derived models and for processing of measured data by RT Toolbox (Mitsubishi) see the figures in the next section.

VI. Simulation and Real Experiment

Modelling includes both mathematical and 3D modelling and planning of motion trajectories, i.e. it represents a number of complex procedures. This section focuses on the results

TABLE II

G-code of testing trajectory (mm)

```
N010  G19
N020  G00  X400  Y0    Z400
N030  G01  X400  Y200  Z400
N040  G01  X400  Y0    Z400
N050  G02  X200  Y-200 Z400  I-200  J0    K0
N060  G02  X200  Y200  Z400  I0     J200  K0
N070  G02  X400  Y0    Z400  I0     J-200 K0
N080  G01  X400  Y-200 Z400
N090  G01  X400  Y200  Z400
N100  G01  X400  Y0    Z400
N110  G00  X420  Y0    Z710
% HOME position of the robot RV-4FL-D:
% Joint Jog mode [deg]:
% J1 =   0; J2 = 0; J3 =  90; J4 =    0; J5 =   0; J6 =   0;
% XYZ  Jog mode   [mm]:
%   X = 420;  Y = 0;  Z = 710; Ax = -180; By =  90; Cz = 180;
```

TABLE III

Code in Melfa Language for RT Toolbox

```
N010  Servo On
N020  M1=1
N030  While M1<=10
N040  Mvs  P1
N050  Mvs  P2
N060  Mvr  P2, Pr23, P3
N070  Mvr  P1,  Pr2, P4
N080  Mvr  P4,  Pr3, P5
N090  Mvr  P5,  Pr1, P6
N100  Mvs  P7
N110  Mvs  P8
N120  Mvs  P9
N130  Mvs P10
N140  M1 = M1 + 1
N150  WEnd
N140  Hlt
Definition of used points Pi (mm):
  Pi   =(   x,    y,    z,   Ax,   By,   Cz)(Fl1,Fl2)
  P1   =(+400,   +0,+400,+180,   +0,+180)( 7 , 0 )
  P2   =(+400,+200,+400,+180,   +0,+180)( 7 , 0 )
  Pr23=(+400,+100,+400,+180,   +0,+180)( 7 , 0 )
  P3   =(+400,   +0,+400,+180,   +0,+180)( 7 , 0 )
  Pr2  =(+341,-141,+400,+180,   +0,+180)( 7 , 0 )
  P4   =(+200,-200,+400,+180,   +0,+180)( 7 , 0 )
  Pr3  =(+200,   +0,+600,+180,   +0,+180)( 7 , 0 )
  P5   =(+200,+200,+400,+180,   +0,+180)( 7 , 0 )
  Pr1  =(+341,+141,+400,+180,   +0,+180)( 7 , 0 )
  P6   =(+400,   +0,+400,+180,   +0,+180)( 7 , 0 )
  P7   =(+400,-200,+400,+180,   +0,+180)( 7 , 0 )
  P8   =(+400,+200,+400,+180,   +0,+180)( 7 , 0 )
  P9   =(+400,   +0,+400,+180,   +0,+180)( 7 , 0 )
  P10  =(+420,   +0,+710,+180,   +0,+180)( 7 , 0 )
  P0   =(+420,   +0,+710,+180,   +0,+180)( 7 , 0 )
```

Note: $\underline{341} = 200 + 141 (= 200 \sin 45^\circ)$ and $\underline{141} = 200 \sin 45^\circ$

in the form of time histories of simulated and measured physical quantities characterising robot motion. Specifically, let us first consider the definition of reference motion in TABLE II. It shows the universal description of the motion geometry in G-code for motion trajectory generation, i.e. path interpolation and time parameterisation. The equivalent code in Melfa Language (RT Toolbox) is shown in Table III.

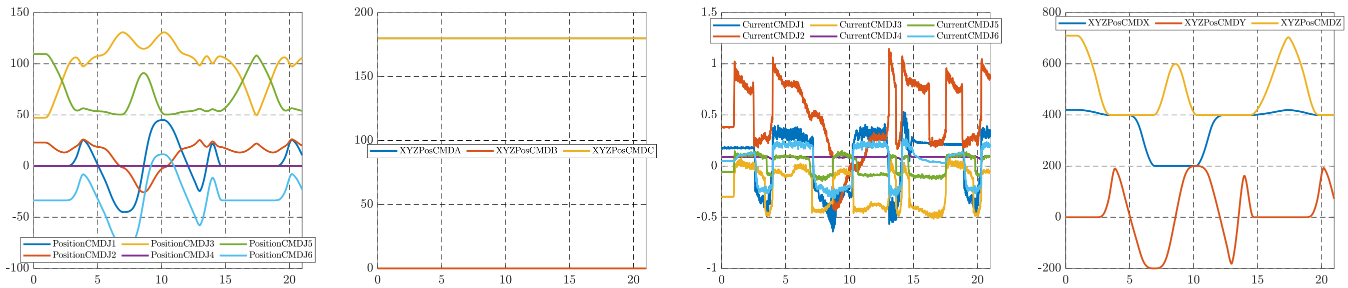

Fig. 9. Time histories: (a) joint position command 1-6, [deg]; (b) Ax,By,Cz command 1-6 [deg]; (c) current command 1-6, [Arms], (d) x,y,z command [mm].

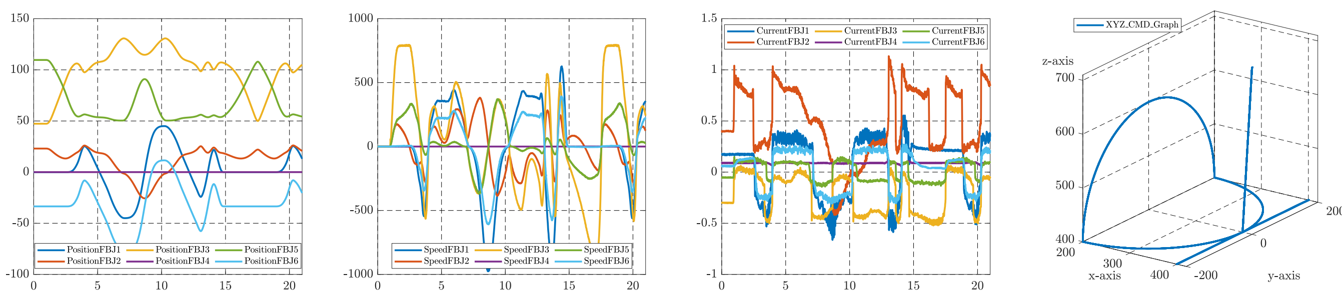

Fig. 10. Time histories: (a) joint position feedback 1-6, [deg]; (b) speed feedback 1-6, [deg/s]; (c) current feedback 1-6, [A]; (d) XYZ graph [mm].

Using both codes, G-code (in MATLAB user functions) and Code in Melfa Language (in RT Toolbox) the simulations and real-time measurements on the physical robot were realised. The figures, Fig. 9 (a)-(c) and Fig. 10 (a)-(c), show measured state quantities using RT Toolbox specifically: angular position, angular speed, current (torque currents) for individual joints – appropriate drives along defined trajectory, i.e. each containing 6 time histories corresponding to 6 driven joints/axes. The presented time histories were captured by RT Toolbox for sampling period $T_s = 7.1$ ms, [13]. Current profiles indicate load from motion and the robot arm itself from joint 1 to joint 6. The biggest load has to be covered by joint 2, which produces the main motion of the robot access system (the first three axes). On the other side, the orientation system (wrist, the last three axes) represents only a particular motion. It was reduced in the presented case by the character of the reference trajectory. Fig. 9 (d) and Fig. 10 (d) depict x, y and z Cartesian coordinates as individual time histories and resultant 3D trajectory. The used testing trajectory consists of line and arc segments (linear and circular interpolation).

VII. Conclusion

The paper deals with mathematical modelling adapted to a specific industrial robot kinematics and dynamics, conventional control and outline of novel advanced hierarchical control concept. There are representatives of possible software tools for modelling [15], [2]. They provide data (both measured and simulated) for identifying and verifying model parameters of a particular robot. Each industrial robot considered in the research requires an analysis of the manufacturer's configuration. Control design optimisation can then be designed and achieved. All pictured real data were captured on the assembly module (Fig. 2) involved in the cyber-physical factory at the College of Polytechnics Jihlava.

References

[1] R. Jazar, *Theory of Applied Robotics*, 2nd ed. Springer, 2010.

[2] M. Raza, P. M. Kumar, D. V. Hung, W. Davis, H. Nguyen, and R. Trestian, "A digital twin framework for I4.0 enabling next-gen manufacturing," in *Indus. Tech. Management Conf.*, 2020, pp. 73–77.

[3] K. Belda and J. Jirsa, "Control principles of autonomous mobile robots used in cyber-physical factories," in *Proc. 23rd Int. Conf. Process Control*. IEEE, 2021, pp. 96–101.

[4] Z. Hendzel and M. Trojnacki, "Adaptive fuzzy control of a four-wheeled mobile robot subject to wheel slip," *WSEAS Trans Systems*, vol. 22, pp. 602–612, 2023.

[5] H. Jiang, Z. Wang, Y. Jin, X. Chen, P. Li, Y. Gan, S. Lin, and X. Chen, "Hierarchical control of soft manipulators towards unstructured interactions," *Int. J. Robotics Resch.*, vol. 40, no. 1, pp. 411–434, 2021.

[6] D. Docimo, "A design framework with embedded hierarchical control architecture optimization," in *ACC conf.*, 2022, pp. 3184–3191.

[7] *RV-FR Series, Instruction manual, Robot Arm Setup & Maitenance, BFP-A3474-N*, Mitsubishi Electric, 2022.

[8] K. Belda and P. Píša, "Explicit model predictive control of pmsm drives," in *Proc. IEEE 30th Int. Symposium on Industrial Electronics*. JTB Communication Design, Inc., 2021, pp. 1–6.

[9] K. Belda and O. Rovný, "Predictive control of 5 DOF robot arm of autonomous mobile robotic system motion control employing mathematical model of the robot arm dynamics," in *Proc. 21th Int. Conf. Process Control*, 2017, pp. 339–344.

[10] K. Belda, "Nonlinear model predictive control algorithms for industrial articulated robots," in *Informatics in Control, Automation and Robotics: ICINCO, Revised Selected Papers*. Springer, 2019, pp. 230–251.

[11] *Simscape^{TM} Multibody^{TM}, User's Guide*, MathWorks, 2022.

[12] *Robotics System Toolbox^{TM}, User's Guide*, MathWorks, 2022.

[13] *RT ToolBox3/RT ToolBox3 mini, User's Manual, BFP-A3495-L*, Mitsubishi Electric, 2020.

[14] K. Belda and K. Dvořák, "Path modeling and 3D robot visualization for model-based control of articulated robots," in *Preprints 15th European Workshop on Advanced Control and Diagnosis*. IFAC, 2019, pp. 1–16.

[15] M. Ryalat, H. ElMoaqet, and M. AlFaouri, "Design of a smart factory based on cyber-physical systems and internet of things towards industry 4.0," *Applied Sciences*, vol. 13, no. 4, pp. 1–19, 2023.