DOI 10.2478/ama-2025-0064

A COMPREHENSIVE MODEL OF THE ABB IRB 2400 FOR SIMULATION AND CONTROL APPLICATIONS

Paweł OBAL*, Piotr GIERLAK*

*The Faculty of Mechanical Engineering and Aeronautics, Dept of Applied Mechanics and Robotics, Rzeszow University of Technology, al. Powstańców Warszawy 12, 35-959 Rzeszów

p.obal@prz.edu.pl, pgierlak@prz.edu.pl

received 01 April 2025, revised 28 August 2025, accepted 28 September 2025

Abstract: This paper presents a comprehensive mathematical model of the ABB IRB 2400 industrial robot, developed to support simulation, control design, and digital twin applications. The model includes both kinematic and dynamic descriptions of the manipulator. Kinematic modelling is based on a modified Denavit-Hartenberg convention and includes transformation matrices and the Jacobian matrix. The dynamic model was derived using both the Euler-Lagrange and Newton-Euler formalisms, enabling validation through independent formulations. Physical parameters such as link masses, centres of mass, and moments of inertia, were estimated through CAD analysis. Friction coefficients were determined by experimental testing. Model validation was performed by comparing simulated joint torques with measurements on the real robot for representative trajectories. The results show agreement: for Joints 2–3 the RMSE relative to the average actuator torque is ≈ 3.4–3.7%, while for Joints 4–6 it remains below 10% (Joint 1 reaches 11.6%). Compared with typical kinematics-only simulation in offline-programming tools, the proposed model captures dynamic effects. The equation set is computationally light and amenable to real-time use within standard control cycles, facilitating integration into digital-twin workflows. The approach is also transferable to other six-axis manipulators of comparable architecture by updating link inertias and friction coefficients. Limitations include the rigid-body assumption (no link/joint compliance or backlash) and reliance on controller-reported actuator torques whose proprietary accuracy is not disclosed; these aspects motivate future extensions with elastic joints, external-force observers and uncertainty tracking.

Key words: dynamics modelling, industrial robot, kinematic modelling, Euler-Lagrange formalism, Newton-Euler formalism

1. INTRODUCTION

The deployment of industrial robots in manufacturing processes has exhibited steady growth in recent years. This trend is largely attributable to the inherent advantages of automated production systems, including increased efficiency, repeatability, and adaptability. According to the International Federation of Robotics [1], the automotive sector remains the predominant consumer of industrial robotics. Nevertheless, its relative share has diminished as other industries increasingly invest in automation technologies to enhance their production capabilities.

Industrial robots are mainly used for operations such as material handling, welding, assembly, and dispensing. Conversely, their application in robotic machining remains comparatively limited. This limitation stems primarily from the insufficient structural rigidity and relatively low precision of robotic manipulators when compared to conventional CNC (Computer Numerical Control) machine tools. However, robots offer distinct advantages over CNC machines in terms of extended workspace, higher degrees of freedom, and operational flexibility [2–5]. Consequently, robotic systems are being adopted for machining processes where high structural stiffness is not critical – for instance, in milling soft materials such as wood. In scenarios that demand greater motion accuracy, various compensation strategies are implemented to correct the robot's positional deviations relative to the workpiece.

A critical requirement for improving robotic accuracy and integrating industrial robots into more demanding manufacturing applications is the development of a high-fidelity dynamic model. Such a model facilitates development of simulation environments and the implementation of *digital twins* – virtual representations of physical systems – which enable predictive analysis and optimisation of production workflows [6–8]. Digital twins constitute a contemporary paradigm in manufacturing systems engineering, allowing for the virtual planning of machine trajectories, estimation of cycle times, and in-depth experimentation with process parameters without physical hardware. Furthermore, digital twins are instrumental in simulating control system behaviour and validating inter-device communication during virtual commissioning. Modern industrial networks support integration with virtual environments [9,10], enabling real-time interaction between physical controllers (e.g., PLCs, robot controllers) and virtualised machinery. This capability allows for pre-deployment software verification, early detection of control logic errors, and flexible reconfiguration of existing systems.

Most industrial robot manufacturers provide proprietary software environments for offline programming and simulation of robotic work cells based on their own products. Additionally, third-party platforms exist that support the integration of equipment from multiple vendors [11,12] . Regardless of origin, the majority of these tools rely predominantly on the robot's kinematic model [11,13], which only describes geometric relations between joints and links. Such a simplified description cannot predict phenomena that dominate robotic machining, including regenerative chatter, tool-tip deflections caused by structural compliance, velocity-dependent friction with pronounced Stribeck hysteresis, or thermally induced drift.

In practice these effects translate into trajectory errors of several millimetres and torque oscillations, ultimately degrading surface quality and tool life. Capturing them requires a comprehensive dynamic model that couples inertia, joint friction, link flexibility with high-frequency excitation from the cutting process [2,14–16] . Recent research on wire-arc additive manufacturing (WAAM) confirms a similar trend: accurate thermo-mechanical models are indispensable for predicting distortion and residual stresses in large-scale deposits [17–22].

Due to intellectual-property constraints, robot manufacturers rarely disclose detailed dynamic models or component-level parameters of their products. Although constructing a custom research platform is possible [23], it is often prohibitively expensive and may not yield insights applicable to existing commercial robots. A more pragmatic approach is to build transferable, data-driven models of widely deployed manipulators. The formulation proposed here based on Euler-Lagrange and Newton-Euler formalisms can be ported to other six-axis arms of comparable architecture. Only link inertias and drive-train friction coefficients need to be updated. Parameter identification was therefore carried out in two stages: CAD-derived mass and inertia data were refined using torque measurements obtained via the External Guided Motion interface and a six-axis force/torque sensor, and a constant-velocity procedure was employed to isolate friction hysteresis despite the absence of direct joint-torque sensors. The resulting parameter set was validated against time-series data of measured torques for various trajectories. The results for single-member movements were presented to demonstrate the modelled phenomena. This foundation enables high-fidelity digital twins, supports simulation-driven experimentation and paves the way for advanced control algorithms aimed at improving task accuracy in robotic machining applications.

This article presents the development of a comprehensive dynamic model of the ABB IRB 2400 industrial robot, which serves as a research platform for investigating robotic machining processes. The IRB 2400 is widely used in academic and industrial research centres worldwide [24–27]. However, to the best of the authors' knowledge, a complete dynamic model of this manipulator has not yet been published. The presented work introduces a full rigid-body dynamic model, including the robot's kinematic equations, Jacobian matrix, and dynamic equations of motion. The methodology for estimating the model parameters and validating the model's accuracy is also described. The estimated parameter values are made publicly available, thus contributing to the existing body of knowledge and enabling further research. The proposed model facilitates the development of high-fidelity digital twins, supports simulation-driven experimentation, and provides a foundation for advanced control algorithm design aimed at improving task accuracy in robotic machining applications.

2. OBJECT DESCRIPTION

In order to conduct research on the robotic automation of machining processes, a dedicated experimental setup was developed, as illustrated in Fig. 1 and Fig. 2. The core of the research station is the ABB IRB 2400 industrial robot, which is equipped with a six-axis force/torque sensor and a high-speed spindle unit. The station also includes a two-axis positioner for workpiece manipulation. The Tool Centre Point (TCP) of the robot is defined relative to the machining tool mounted in the spindle holder. Its spatial position is determined with respect to a coordinate frame attached to the robot's flange surface. The robot has been equipped with the Absolute Accuracy option, which means that it has been additionally calibrated to improve the TCP's absolute positioning accuracy. The manufacturer provides a calibration ("Birth") certificate specifying unit-specific positioning-accuracy metrics; the values for the robot used in this study are listed in Tab. 1. The ABB IRB 2400 used in this study is additionally equipped with the External Guided Motion (EGM) interface, which streams the full robot state vector (joint positions, velocities and controller-estimated actuator torques) at 250 Hz directly from the IRC5 controller. While positional accuracy is certified by the manufacturer, the actuator torques are not measured; they are computed online by the controller's internal dynamic model and the associated uncertainty is not disclosed. Accordingly, torque values are treated as estimates.

Tab. 1. Accuracy parameters from the Robot's "Birth certificate"

Parameter	Value (mm)
Average Absolute Error	0.18
Maximum Absolute Error	0.38
Standard Deviation	0.07

The mechanical structure of the manipulator comprises six revolute joints forming an open kinematic chain. Notably, the third link is actuated by a parallelogram linkage that maintains its spatial orientation constant during the motion of the second joint. This kinematic arrangement ensures stable tool orientation, which is advantageous for precision tasks such as machining.

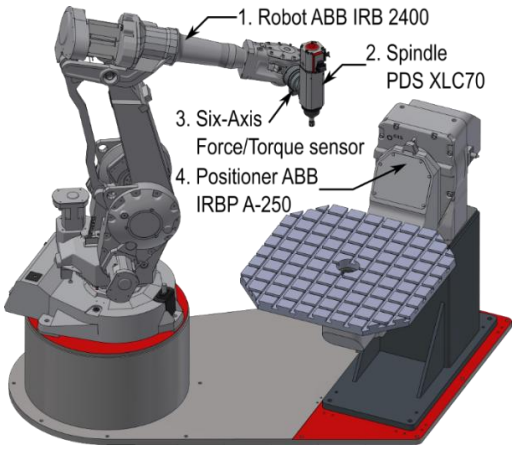


Fig. 1. Schematic diagram of the robotic machining test stand

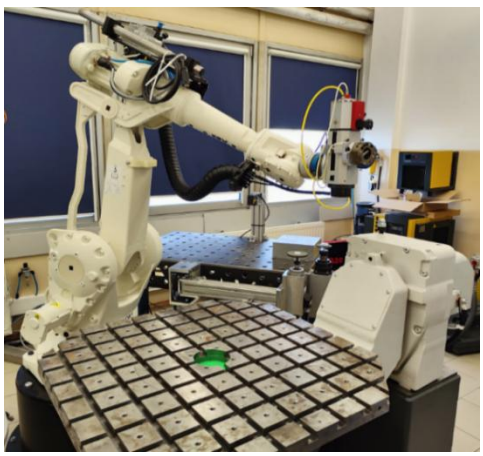


Fig. 2. Physical implementation of the robotic machining test stand

DOI 10.2478/ama-2025-0064

The mathematical model of the robot will be utilised for the synthesis of control algorithms and for performing simulation studies. The development of the model was based on the following simplifying assumptions:

- The manipulator links and the tool are modelled as rigid bodies, neglecting structural deformations;
- The dynamic behaviour of actuators, joint compliance, and transmission backlash are disregarded;
- The parallelogram linkage is not modelled explicitly, its effect is absorbed into the link inertias;
- The centre of mass of each link is assumed to lie within its plane of symmetry.

These simplifications reduce model complexity without unduly compromising the fidelity required for control design and dynamic simulation. They are dictated primarily by limitations of the measurement chain: with access only to motor-side position signals, the effects of link flexibility, joint elasticity and gearbox backlash cannot be isolated and are therefore neglected. The resulting model underpins subsequent studies on robotic machining performance, trajectory optimisation and accuracy-enhancement strategies.

3. KINEMATIC EQUATIONS OF THE ABB IRB 2400 ROBOT

The kinematic model of the ABB IRB 2400 manipulator was formulated using the Modified Denavit–Hartenberg (MDH) convention, which is widely adopted for the systematic modelling of serial robotic chains. The MDH convention was selected because its link frames coincide with the physical joint axes, eliminating the half-link offset of the classical formulation and thereby simplifying gravity compensation and inertia characterisation. Using MDH also guarantees direct compatibility with the kinematic data provided by ABB for the IRB 2400, keeping the analytical model numerically consistent with the controller during calibration, collision checking and offline programming. In addition, the MDH arrangement aligns with modern spatial-vector libraries (for example, Pinocchio and RBDL), enabling seamless reuse of CAD-derived inertia tensors and efficient evaluation of dynamic terms. Taken together, these aspects make MDH a transparent and computationally economical choice relative to the classical DH and Product-of-Exponentials descriptions, and it is therefore adopted throughout this study. According to the MDH convention, a Cartesian coordinate frame is assigned to each i link of the manipulator such that:

- The z-axis z_i is aligned with the axis of rotation of joint i;
- The axis x_i intersects z_{i-1};
- The axis y_i completes the right-handed coordinate system.

The transformation of coordinate frame i with respect to frame $i-1$ is described by a homogeneous transformation matrix $\mathbf{T}_i^{i-1}$. In the MDH convention homogeneous transformation is expressed as the product of four elementary transformations:

$$\mathbf{T}_i^{i-1} = \mathbf{Rot}_{x,\alpha_{i-1}}\mathbf{Trans}_{x,a_{i-1}}\mathbf{Rot}_{z,\theta_i}\mathbf{Trans}_{z,d_i}, \tag{1}$$

where: $\mathbf{Rot}_{x,\alpha_{i-1}}$ – rotation around the x-axis by the twist angle α_{i-1}, $\mathbf{Trans}_{x,a_{i-1}}$ – translation along the x-axis by the link length a_{i-1}, $\mathbf{Rot}_{z,\theta_i}$ – rotation around the z-axis by the joint angle θ_i, $\mathbf{Trans}_{z,d_i}$ – translation along the z-axis by the offset d_i. The names of the trigonometric functions have been abbreviated using the following notation $\mathrm{s}_{q_i} = \sin(q_i)$, $\mathrm{c}_{q_i} = \cos(q_i)$.

The resulting matrix has the following explicit form:

$$\mathbf{T}_i^{i-1} = \begin{bmatrix} \mathrm{c}_{\theta_i} & -\mathrm{s}_{\theta_i} & 0 & a_{i-1} \\ \mathrm{s}_{\theta_i}\mathrm{c}_{\alpha_{i-1}} & \mathrm{c}_{\theta_i}\mathrm{c}_{\alpha_{i-1}} & -\mathrm{s}_{\alpha_{i-1}} & -d_i\mathrm{s}_{\alpha_{i-1}} \\ \mathrm{s}_{\theta_i}\mathrm{s}_{\alpha_{i-1}} & \mathrm{c}_{\theta_i}\mathrm{s}_{\alpha_{i-1}} & \mathrm{c}_{\alpha_{i-1}} & d_i\mathrm{c}_{\alpha_{i-1}} \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{2}$$

The parameters α_{i-1}, a_{i-1}, θ_i, and d_i represent the geometric characteristics of each link, as defined by the modified DH convention.

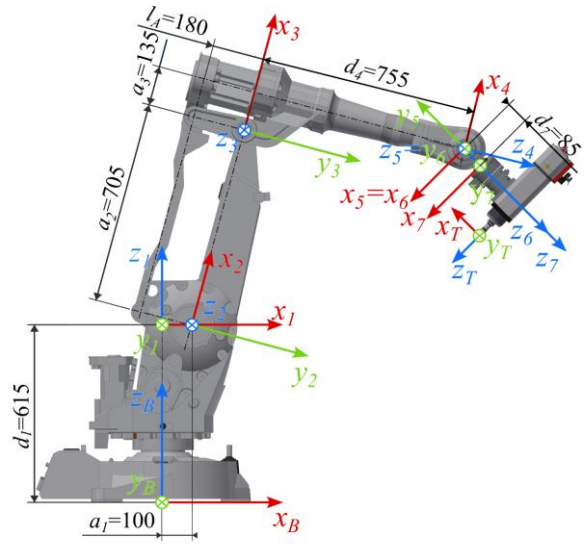


Fig. 3. Schematic diagram of the ABB IRB 2400 manipulator with coordinate frames assigned in accordance with the modified Denavit–Hartenberg convention

The coordinate frames used for the kinematic description of the ABB IRB 2400 manipulator, as well as the corresponding distances between them, are presented in Fig. 3. The dimensional and structural data were extracted from the manufacturer's documentation [28].

In this study, coordinate frames permanently attached to the robot's physical structure are labelled according to the link number in the kinematic chain. The links are numbered starting from the robot base (Link 1) up to Frame 7, which is attached to the flange and corresponds to the tool coordinate system, commonly referred to as TCP of tool 0 in industrial robot programming manual [29].

Additional coordinate frames, which can be defined arbitrarily (e.g., for external tools or mounting), are denoted using alphabetic symbols. In Fig. 3, the base frame is labelled ***B***, while the frame associated with the tool mounted on the robot flange is denoted as ***T***, representing the TCP of the spindle.

Because the origin of a coordinate frame is a point in three-dimensional space, the homogeneous transformations described above enable the determination of the position and orientation of any frame with respect to another.

An important kinematic characteristic of the ABB IRB 2400 is that the third link is driven via a parallelogram linkage mechanically aligned with the second link. Joint 3 is actuated via a parallelogram. The motor axis is coaxial with Joint 2, while Joint 3's rotation axis is distinct. The rotation of each joint is expressed by a generalised joint variable q_i, representing the angular displacement measured along the respective actuation axis.

The Denavit–Hartenberg parameters of the ABB IRB 2400 manipulator, based on the modified convention, are summarized in Tab. 2.

Tab. 2. Geometric parameters of ABB IRB 2400 manipulator according to modified Denavit-Hartenberg notation

	a_i (m)	α_i (rad)	d_i (m)	θ_i (rad)
Link 1	0	0	$d_1 = 0.615$	q_1
Link 2	$a_1 = 0.1$	$-\frac{\pi}{2}$	0	$q_2 - \frac{\pi}{2}$
Link 3	$a_2 = 0.705$	0	0	$q_3 - q_2$
Link 4	$a_3 = 0.135$	$-\frac{\pi}{2}$	$d_4 = 0.755$	q_4
Link 5	0	$\frac{\pi}{2}$	0	$q_5 + \pi$
Link 6	0	$\frac{\pi}{2}$	0	q_6
Tool 0	0	0	$d_7 = 0.085$	0

Based on the general transformation formulation introduced in Section 3, the individual homogeneous transformation matrices were derived using the Denavit–Hartenberg parameters listed in Tab. 2. The transformations from one link frame to the next are given by the following matrices:

$$\mathbf{T}_1^B = \begin{bmatrix} \mathrm{c}_{q_1} & -\mathrm{s}_{q_1} & 0 & 0 \\ \mathrm{s}_{q_1} & \mathrm{c}_{q_1} & 0 & 0 \\ 0 & 0 & 1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{3}$$

$$\mathbf{T}_2^1 = \begin{bmatrix} \mathrm{s}_{q_2} & \mathrm{c}_{q_2} & 0 & a_1 \\ 0 & 0 & 1 & 0 \\ \mathrm{c}_{q_2} & -\mathrm{s}_{q_2} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{4}$$

$$\mathbf{T}_3^2 = \begin{bmatrix} \mathrm{c}_{q_2-q_3} & \mathrm{s}_{q_2-q_3} & 0 & a_2 \\ -\mathrm{s}_{q_2-q_3} & \mathrm{c}_{q_2-q_3} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{5}$$

$$\mathbf{T}_4^3 = \begin{bmatrix} \mathrm{c}_{q_4} & \mathrm{s}_{q_4} & 0 & a_3 \\ 0 & 0 & 1 & d_4 \\ -\mathrm{s}_{q_4} & -\mathrm{c}_{q_4} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{6}$$

$$\mathbf{T}_5^4 = \begin{bmatrix} -\mathrm{c}_{q_5} & \mathrm{s}_{q_5} & 0 & 0 \\ 0 & 0 & -1 & 0 \\ -\mathrm{s}_{q_5} & -\mathrm{c}_{q_5} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{7}$$

$$\mathbf{T}_6^5 = \begin{bmatrix} \mathrm{c}_{q_6} & -\mathrm{s}_{q_6} & 0 & 0 \\ 0 & 0 & -1 & 0 \\ \mathrm{s}_{q_6} & \mathrm{c}_{q_6} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{8}$$

$$\mathbf{T}_7^6 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_7 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{9}$$

In this work, vector and matrix notations adopt a convention where the lower index indicates the frame in which the vector or transformation is expressed, while the upper index refers to the reference frame. For example, $\mathbf{o}_T^7$ represents the position vector of point T with respect to frame 7, while $\mathbf{o}_7^T$ denotes the position of frame 7's origin expressed in the TCP frame.

To account for the tool mounted on the robot's flange, an additional coordinate frame associated with the TCP of the spindle was defined. Its position and orientation with respect to frame 7 (flange) are defined by the translation vector $\mathbf{o}_T^7$ and the rotation matrix $\mathbf{R}_T^7$, resulting in the homogeneous transformation:

$$\mathbf{T}_T^7 = \begin{bmatrix} & \mathbf{R}_T^7 & & \mathbf{o}_T^7 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{10}$$

The rotation matrix $\mathbf{R}_T^7$ corresponds to a rotation of $\pi/2$ around the y-axis:

$$\mathbf{R}_T^7 = \mathbf{Rot}_{y,\frac{\pi}{2}} = \begin{bmatrix} \mathrm{c}_{\frac{\pi}{2}} & 0 & \mathrm{s}_{\frac{\pi}{2}} \\ 0 & 1 & 0 \\ -\mathrm{s}_{\frac{\pi}{2}} & 0 & \mathrm{c}_{\frac{\pi}{2}} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{bmatrix}. \tag{11}$$

Hence, the complete homogeneous transformation matrix becomes:

$$\mathbf{T}_T^7 = \begin{bmatrix} 0 & 0 & 1 & x_T \\ 0 & 1 & 0 & y_T \\ -1 & 0 & 0 & z_T \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{12}$$

Using the defined transformation matrices, the complete transformation from any intermediate frame j to the base frame $\boldsymbol{B}$ determined via successive matrix multiplication:

$$\mathbf{T}_j^B = \prod_{k=1}^{j} \mathbf{T}_k^{k-1}. \tag{13}$$

Accordingly, the position and orientation of the TCP with respect to the base frame is expressed by the overall homogeneous transformation matrix:

$$\mathbf{T}_T^B = \begin{bmatrix} R_{11} & R_{12} & R_{13} & o_{T_x} \\ R_{21} & R_{22} & R_{23} & o_{T_y} \\ R_{31} & R_{32} & R_{33} & o_{T_z} \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{14}$$

Where the terms R_{ij} and o_{Ti} represent the elements of rotation submatrix and translation vector, respectively. The analytical expressions for these components are detailed in equations (15) through (26):

$$R_{11} = \mathrm{s}_{q_1}\mathrm{s}_{q_4}\mathrm{s}_{q_5} - \mathrm{c}_{q_1}\mathrm{c}_{q_3}\mathrm{c}_{q_5} + \mathrm{c}_{q_1}\mathrm{c}_{q_4}\mathrm{s}_{q_3}\mathrm{s}_{q_5}, \tag{15}$$

$$R_{12} = \mathrm{c}_{q_1}\mathrm{c}_{q_6}\mathrm{s}_{q_3}\mathrm{s}_{q_4} - \mathrm{c}_{q_4}\mathrm{c}_{q_6}\mathrm{s}_{q_1} + \mathrm{c}_{q_1}\mathrm{c}_{q_3}\mathrm{s}_{q_5}\mathrm{s}_{q_6} + \mathrm{c}_{q_5}\mathrm{s}_{q_1}\mathrm{s}_{q_4}\mathrm{s}_{q_6} + \mathrm{c}_{q_1}\mathrm{c}_{q_4}\mathrm{c}_{q_5}\mathrm{s}_{q_3}\mathrm{s}_{q_6}, \tag{16}$$

$$R_{13} = \mathrm{c}_{q_1}\mathrm{s}_{q_3}\mathrm{s}_{q_4}\mathrm{s}_{q_6} - \mathrm{c}_{q_1}\mathrm{c}_{q_3}\mathrm{c}_{q_6}\mathrm{s}_{q_5} - \mathrm{c}_{q_5}\mathrm{c}_{q_6}\mathrm{s}_{q_1}\mathrm{s}_{q_4} - \mathrm{c}_{q_4}\mathrm{s}_{q_1}\mathrm{s}_{q_6} - \mathrm{c}_{q_1}\mathrm{c}_{q_4}\mathrm{c}_{q_5}\mathrm{c}_{q_6}\mathrm{s}_{q_3}, \tag{17}$$

$$R_{21} = \mathrm{c}_{q_4}\mathrm{s}_{q_1}\mathrm{s}_{q_3}\mathrm{s}_{q_5} - \mathrm{c}_{q_3}\mathrm{c}_{q_5}\mathrm{s}_{q_1} - \mathrm{c}_{q_1}\mathrm{s}_{q_4}\mathrm{s}_{q_5}, \tag{18}$$

$$R_{22} = \mathrm{c}_{q_1}\mathrm{c}_{q_4}\mathrm{c}_{q_6} - \mathrm{c}_{q_1}\mathrm{c}_{q_5}\mathrm{s}_{q_4}\mathrm{s}_{q_6} + \mathrm{c}_{q_6}\mathrm{s}_{q_1}\mathrm{s}_{q_3}\mathrm{s}_{q_1} + \mathrm{c}_{q_3}\mathrm{s}_{q_1}\mathrm{s}_{q_5}\mathrm{s}_{q_6} + \mathrm{c}_{q_4}\mathrm{c}_{q_5}\mathrm{s}_{q_1}\mathrm{s}_{q_3}\mathrm{s}_{q_6}, \tag{19}$$

$$R_{23} = \mathrm{c}_{q_1}\mathrm{c}_{q_4}\mathrm{s}_{q_6} + \mathrm{c}_{q_1}\mathrm{c}_{q_5}\mathrm{c}_{q_6}\mathrm{s}_{q_4} - \mathrm{c}_{q_3}\mathrm{c}_{q_6}\mathrm{s}_{q_1}\mathrm{s}_{q_5} + \mathrm{s}_{q_1}\mathrm{s}_{q_3}\mathrm{s}_{q_4}\mathrm{s}_{q_6} - \mathrm{c}_{q_4}\mathrm{c}_{q_5}\mathrm{c}_{q_6}\mathrm{s}_{q_1}\mathrm{s}_{q_3}, \tag{20}$$

$$R_{31} = \mathrm{c}_{q_5}\mathrm{s}_{q_3} + \mathrm{c}_{q_3}\mathrm{c}_{q_4}\mathrm{s}_{q_5}, \tag{21}$$

$$R_{32} = \mathrm{c}_{q_3}\mathrm{c}_{q_6}\mathrm{s}_{q_6} - \mathrm{s}_{q_3}\mathrm{s}_{q_5}\mathrm{s}_{q_6} + \mathrm{c}_{q_3}\mathrm{c}_{q_4}\mathrm{c}_{q_5}\mathrm{s}_{q_6}, \tag{22}$$

$$R_{33} = \mathrm{c}_{q_3}\mathrm{s}_{q_4}\mathrm{s}_{q_6} + \mathrm{c}_{q_6}\mathrm{s}_{q_3}\mathrm{s}_{q_5} - \mathrm{c}_{q_3}\mathrm{c}_{q_4}\mathrm{c}_{q_5}\mathrm{c}_{q_6}, \tag{23}$$

$$o_{T_x} = a_1\mathrm{c}_{q_1} + d_4\mathrm{c}_{q_1}\mathrm{c}_{q_3} + a_2\mathrm{c}_{q_1}\mathrm{s}_{q_2} + a_3\mathrm{c}_{q_1}\mathrm{s}_{q_3} + (d_7 + z_T)\big(\mathrm{c}_{q_1}\mathrm{c}_{q_3}\mathrm{c}_{q_5} - \mathrm{s}_{q_1}\mathrm{s}_{q_4}\mathrm{s}_{q_5} -$$

DOI 10.2478/ama-2025-0064

$$c_{q_1}c_{q_4}s_{q_3}s_{q_5}) + x_T(c_{q_1}s_{q_3}s_{q_4}s_{q_6} - c_{q_1}c_{q_3}c_{q_6}s_{q_5} - c_{q_5}c_{q_6}s_{q_1}s_{q_4} - c_{q_4}s_{q_1}s_{q_6} - c_{q_1}c_{q_4}c_{q_5}c_{q_6}s_{q_3}) + y_T(c_{q_1}c_{q_6}s_{q_3}s_{q_4} - c_{q_4}c_{q_6}s_{q_1} + c_{q_1}c_{q_3}s_{q_5}s_{q_6} + c_{q_5}s_{q_1}s_{q_4}s_{q_6} + c_{q_1}c_{q_4}c_{q_5}s_{q_3}s_{q_6}), \quad (24)$$

$$o_{T_y} = a_1s_{q_1} + d_4c_{q_3}s_{q_1} + a_2s_{q_1}s_{q_2} + a_3s_{q_1}s_{q_3} + (d_7 + z_T)(c_{q_3}c_{q_5}s_{q_1} + c_{q_1}s_{q_4}s_{q_5} - c_{q_4}s_{q_1}s_{q_3}s_{q_5}) + x_T(c_{q_1}c_{q_4}s_{q_6} + c_{q_1}c_{q_5}c_{q_6}s_{q_4} - c_{q_3}c_{q_6}s_{q_1}s_{q_5} + s_{q_1}s_{q_3}s_{q_4}s_{q_6} - c_{q_4}c_{q_5}c_{q_6}s_{q_1}s_{q_3}) + y_T(c_{q_1}c_{q_4}c_{q_6} - c_{q_1}c_{q_5}s_{q_4}s_{q_6} + c_{q_6}s_{q_1}s_{q_3}s_{q_4} + c_{q_3}s_{q_1}s_{q_5}s_{q_6} + c_{q_4}c_{q_5}c_{q_1}s_{q_3}s_{q_6}), \quad (25)$$

$$o_{T_z} = d_1 + a_2c_{q_2} + a_3c_{q_3} - (d_7 + z_T)(c_{q_5}s_{q_3} + c_{q_3}c_{q_4}s_{q_5}) + x_T(c_{q_3}s_{q_4}s_{q_6} + c_{q_6}s_{q_3}s_{q_5} - c_{q_3}c_{q_4}c_{q_5}c_{q_6}) + y_T(c_{q_3}c_{q_6}s_{q_4} - s_{q_3}s_{q_5}s_{q_6} + c_{q_3}c_{q_4}c_{q_5}s_{q_6}). \quad (26)$$

The resulting expressions account for the full kinematic chain, including the tool offset, and can be used in further analysis for simulation, path planning, or control algorithm development.

4. JACOBIAN OF THE MANIPULATOR

The Jacobian matrix is a key analytical tool in modelling robotic manipulators. It enables the analysis of how different phenomena translate between the configuration space and the task space. It is commonly used for:

- mapping joint velocities to task space velocities in forward and inverse kinematics,
- analysing the impact of external forces acting on the end-effector on the joint torques,
- detecting and avoiding singularities,
- planning robotic motion,
- and designing control systems.

The transformation of angular joint velocities $\dot{\mathbf{q}} \in \mathbb{R}^n$ into the spatial velocity $\mathbf{v}_i^B$ of i^{th} link using the Jacobian can be expressed as:

$$\mathbf{v}_i^B = \begin{bmatrix} \mathbf{v}_i^B \\ \boldsymbol{\omega}_i^B \end{bmatrix} = \mathbf{J}\dot{\mathbf{q}}, \quad (27)$$

where: $\mathbf{v}_i^B \in \mathbb{R}^3$ – linear velocity of the origin of the coordinate frame attached to the i^{th} link with respect to the base frame, $\boldsymbol{\omega}_i^B \in \mathbb{R}^3$ – angular velocity of the i^{th} frame with respect to the base frame, $\mathbf{J} \in \mathbb{R}^{6\times n}$ – the manipulator Jacobian, $\dot{\mathbf{q}} \in \mathbb{R}^n$ – joint velocity vector, $\mathbf{v}_i^B \in \mathbb{R}^6$ – spatial velocity of the i^{th} link with respect to the base frame.

Separating the Jacobian into translational and rotational components gives the following relations:

$$\mathbf{J} = \begin{bmatrix} \mathbf{J}_v \\ \mathbf{J}_\omega \end{bmatrix}, \quad (28)$$

$$\mathbf{v}_i^B = \mathbf{J}_v\dot{\mathbf{q}}, \quad (29)$$

$$\boldsymbol{\omega}_i^B = \mathbf{J}_\omega\dot{\mathbf{q}}. \quad (30)$$

The dimension of the Jacobian matrix depends on the number of generalised coordinates describing the configuration and the number of task-space variables. In this study, a Cartesian task space is assumed, characterized by six variables (three translations and three rotations). For a manipulator with n degrees of freedom, the Jacobian has the form:

$$\mathbf{J} = [\mathbf{J}_1 \quad \mathbf{J}_2 \quad \dots \quad \mathbf{J}_n] \in \mathbb{R}^{6\times n}, \quad (31)$$

According to [30], the Jacobian for a manipulator composed of revolute joints can be constructed based on the geometry of the kinematic chain using:

$$\mathbf{J}_i = \begin{bmatrix} \mathbf{J}_{vi} \\ \mathbf{J}_{\omega i} \end{bmatrix} = \begin{bmatrix} \mathbf{z}_i^B \times (\mathbf{o}_n^B - \mathbf{o}_i^B) \\ \mathbf{z}_i^B \end{bmatrix}, \quad (32)$$

where $\boldsymbol{z}_i^B$ denotes the third column of the rotation matrix of the i^{th} link with respect to the base frame.

The ABB IRB 2400 robot consists of six revolute joints, with the second joint forming a parallelogram mechanism that preserves the orientation of subsequent links. Taking this into account, the Jacobian is determined as follows:

$$\mathbf{J} = \begin{bmatrix} \mathbf{z}_1^B \times (\mathbf{o}_\mathrm{T}^B - \mathbf{o}_1^B) & \mathbf{z}_2^B \times (\mathbf{o}_3^B - \mathbf{o}_2^B) & \dots \\ \mathbf{z}_1^B & [0 \quad 0 \quad 0]^\mathrm{T} & \\ \dots & \mathbf{z}_3^B \times (\mathbf{o}_T^B - \mathbf{o}_3^B) & \mathbf{z}_4^B \times (\mathbf{o}_T^B - \mathbf{o}_4^B) & \dots \\ & \mathbf{z}_3^B & \mathbf{z}_4^B & \\ \dots & \mathbf{z}_5^B \times (\mathbf{o}_T^B - \mathbf{o}_5^B) & \mathbf{z}_6^B \times (\mathbf{o}_T^B - \mathbf{o}_6^B) & \\ & \mathbf{z}_5^B & \mathbf{z}_6^B & \end{bmatrix}. \quad (33)$$

This Jacobian enables transformation of velocities and forces from the joint space to the task space. Additionally, the Jacobian allows to determine singular positions, which is useful information when generating trajectories. The full manipulator Jacobian (33) enables a purely algebraic search for poses in which controllability is lost. A configuration is singular when

$$\mathbf{det}(\mathbf{J}(\mathbf{q})) = 0. \quad (34)$$

For the ABB IRB 2400 condition (34) yields exactly two independent singular configurations. The first results from the equation:

$$\sin(\mathrm{q}_5) = 0 \Rightarrow q_5 = 0 \quad (35)$$

because the axes z_4, z_5, z_6 become collinear and the orientation sub-Jacobian loses rank. Since the range of movement of joint 5 is from -120 to 120, $q_5 = 0$ is the only solution. The second singularity occurs when the centre point of the robot's wrist intersects the z_B axis, which results from the equation:

$$(c_{q_2}s_{q_3})a_1a_3 + (c_{q_2}c_{q_3} + s_{q_2}s_{q_3})a_1d_4 + (c_{q_3}{c_{q_2}}^2 + s_{q_2}s_{q_3}c_{q_2})a_2a_3 + (s_{q_3} + c_{q_2}c_{q_3}s_{q_2})a_2d_4 + c_{q_2}{a_3}^2 + (s_{q_2} + 2c_{q_2}c_{q_3}s_{q_3})a_3d_4 + (c_{q_2}{c_{q_3}}^2 + s_{q_2}s_{q_3}c_{q_3})({d_4}^2 - {a_3}^2) - (c_{q_3}s_{q_2})a_1a_3 - c_{q_3}a_2a_3 - ({c_{q_2}}^2 s_{q_3})a_2d_4 - (2{c_{q_3}}^2 s_{q_2})a_3d_4 = 0, \quad (36)$$

satisfying condition (34). In singular positions, kinematic and dynamic equations using Jacobians degenerate, giving infinite or unpredictably large results. The simplest method of dealing with this problem is to avoid singularities. There are also more advanced methods such as Damped-Least-Squares, singular posture-

passing algorithm or the inverse Jacobian cofactor matrix optimisation method [31]. Singularities do not affect the robot dynamics modelling process, but they should be kept in mind when designing a control system.

5. ROBOT DYNAMIC EQUATIONS – EULER–LAGRANGE FORMALISM

Two widely used methods for modelling manipulator dynamics are the Euler–Lagrange and Newton–Euler formalisms, as described in [30,32,33]. The former is based on variational calculus, while the latter uses recursive formulation of Newton's laws for each link in the chain.

In both methods, a simplified mass distribution model was adopted for the IRB 2400 manipulator, as illustrated in Fig. 4. The mass of the parallelogram linkage driving the third link is significantly smaller than the combined mass of links 3 through 6, making its effect on overall dynamics negligible.

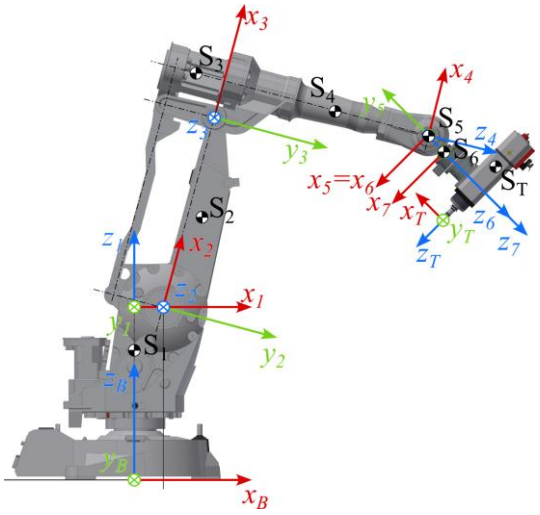


Fig. 4. Centre of mass distribution of the robot's links

The position of the centre of mass of each link relative to its local frame is defined as:

$$\mathbf{r}_{S_1}^1 = \begin{bmatrix} 0 \\ 0 \\ z_{S_1} \end{bmatrix}, \tag{37}$$

$$\mathbf{r}_{S_2}^2 = \begin{bmatrix} x_{S_2} \\ y_{S_2} \\ 0 \end{bmatrix}, \tag{38}$$

$$\mathbf{r}_{S_3}^3 = \begin{bmatrix} x_{S_3} \\ y_{S_3} \\ 0 \end{bmatrix}, \tag{39}$$

$$\mathbf{r}_{S_4}^4 = \begin{bmatrix} 0 \\ 0 \\ z_{S_4} \end{bmatrix}, \tag{40}$$

$$\mathbf{r}_{S_5}^5 = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \tag{41}$$

$$\mathbf{r}_{S_6}^6 = \begin{bmatrix} 0 \\ 0 \\ z_{S_6} \end{bmatrix}, \tag{42}$$

$$\mathbf{r}_{S_T}^7 = \begin{bmatrix} x_{S_T} \\ y_{S_T} \\ z_{S_T} \end{bmatrix}. \tag{43}$$

This section describes the dynamics using the Euler–Lagrange formulation, given by:

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} = \Omega_i, \tag{44}$$

where Ω_i is the generalised force corresponding to the i^{th} generalised coordinate, and L is the Lagrangian:

$$L = K - P, \tag{45}$$

with K representing kinetic energy and P potential energy. The kinetic energy of a multi-body manipulator is defined as:

$$K = \frac{1}{2} m_i \dot{\mathbf{o}}_{Si}^T \dot{\mathbf{o}}_{Si} + \boldsymbol{\omega}_i^T \mathbf{R}_i \mathbf{I}_i \mathbf{R}_i^T \boldsymbol{\omega}_i = \frac{1}{2} \dot{\mathbf{q}}^{\mathrm{T}} \mathbf{M} \dot{\mathbf{q}}, \tag{46}$$

where $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{n \times n}$ is the inertia matrix, m_i is the mass of the i^{th} link, $\mathbf{I}_i$ is the inertia tensor of the i^{th} link in its local frame and $\mathbf{R}_i$ is the rotation matrix from the i^{th} link frame to the base frame. The linear and angular velocities are obtained from the centre-of-mass Jacobians:

$$\dot{\mathbf{o}}_{Si} = \mathbf{J}_{v_{si}}(\mathbf{q})\dot{\mathbf{q}}, \boldsymbol{\omega}_i = \mathbf{J}_{\omega_i}(\mathbf{q})\dot{\mathbf{q}}, \tag{47}$$

where $\mathbf{J}_{v_{si}}(\boldsymbol{q})$ and $\mathbf{J}_{\omega_i}(\boldsymbol{q})$ are the Jacobians for the centre of mass velocity and angular velocity respectively.

The potential energy arises solely from gravity, assuming rigid body links without elastic deformation:

$$P = \sum_{i=1}^{n} m_i \mathbf{g}^{\mathrm{T}} \mathbf{o}_{S_i}^B, \tag{48}$$

Substituting (46) and (48) into (44) yields:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) = \boldsymbol{\Omega}, \tag{49}$$

with $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^n$ the Coriolis and centrifugal terms, $\mathbf{G}(\mathbf{q}) \in \mathbb{R}^n$ the gravity vector, and $\boldsymbol{\Omega} \in \mathbb{R}^n$ the generalised forces. The inertia matrix is calculated as:

$$\mathbf{M}(\mathbf{q}) = \sum_{i=1}^{n} \left(m_i \mathbf{J}_{v_{si}}^{\mathrm{T}}(\mathbf{q}) \mathbf{J}_{v_{si}}(\mathbf{q}) + \mathbf{J}_{\omega_i}^{\mathrm{T}}(\mathbf{q}) \mathbf{R}_i \mathbf{I}_i \mathbf{R}_i^{\mathrm{T}} \mathbf{J}_{\omega_i}(\mathbf{q}) \right) \tag{50}$$

The matrix $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ is computed using:

$$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) = \sum_{i=1}^{n} c_{ijk}(\mathbf{q}) \dot{q}_i =$$

$$\sum_{i=1}^{n} \frac{1}{2} \left(\frac{\partial \mathbf{M}_{kj}(\mathbf{q})}{\partial q_i} + \frac{\partial \mathbf{M}_{ki}(\mathbf{q})}{\partial q_j} - \frac{\partial \mathbf{M}_{ij}(\mathbf{q})}{\partial q_k} \right) \dot{q}_i. \tag{51}$$

The gravity vector is derived from the gradient of the potential energy:

$$\mathbf{G}(\mathbf{q}) = \frac{\partial P(\mathbf{q})}{\partial \boldsymbol{q}}. \tag{52}$$

The generalised force vector includes actuation torques $\mathbf{u}$ and friction forces $\mathbf{F}(\dot{\mathbf{q}})$

$$\boldsymbol{\Omega} = \mathbf{u} - \mathbf{F}(\dot{\mathbf{q}}). \tag{53}$$

Friction forces are modelled using the expression from [34]:

$$\tau_f = \left(F_C + (F_S - F_C) e^{-\left|\frac{\dot{q}}{\omega_{St}}\right|^2} \right) \mathrm{sgn}(\dot{q}) + F_v \dot{q}. \tag{54}$$

This equation combines static (F_S), Coulomb (F_C), and viscous (F_v) components with a Gaussian-type Stribeck term (ω_{St}). The friction function versus speed is shown in Fig. 5.

DOI 10.2478/ama-2025-0064

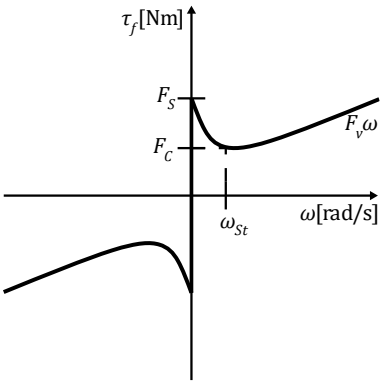


Fig. 5. Graph illustrating the friction model

The friction vector $\mathbf{F}(\dot{\mathbf{q}}) = [\tau_{f1} \quad \cdots \quad \tau_{fn}]^{\mathrm{T}}$ takes the form:

$$\mathbf{F}(\dot{\mathbf{q}}) = \begin{bmatrix} \left(F_{C_1} + \left(F_{S_1} - F_{C_1}\right)e^{-\left|\frac{\dot{q}_1}{\omega_{St_1}}\right|^2}\right)\mathrm{sgn}(\dot{q}_1) + F_{v_1}\dot{q}_1 \\ \vdots \\ \left(F_{C_n} + \left(F_{S_n} - F_{C_n}\right)e^{-\left|\frac{\dot{q}_n}{\omega_{St_n}}\right|^2}\right)\mathrm{sgn}(\dot{q}_n) + F_{v_n}\dot{q}_n \end{bmatrix}. \tag{55}$$

This leads to the complete dynamic equation of the ABB IRB 2400 manipulator:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q},\dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) + \mathbf{F}(\dot{\mathbf{q}}) = \mathbf{u}, \tag{56}$$

Due to their complexity, the explicit elements of matrices $\mathbf{M}$, $\mathbf{C}$ and $\mathbf{G}$ are provided in the appendix [35]. The torque vector is denoted: $\mathbf{u} = [u_1 \quad \cdots \quad u_6]^{\mathrm{T}}$. In summary, the Euler–Lagrange method provides a matrix form suitable for dynamic simulation and controller design.

6. DYNAMIC EQUATIONS OF ROBOT MOTION – NEWTON–EULER FORMALISM

The second widely used approach for modelling the dynamics of robotic manipulators is the Newton–Euler formalism. In contrast to the Lagrangian approach, this method analyses the motion of each individual link separately. The computations are performed using a recursive algorithm known as the forward-backward procedure.

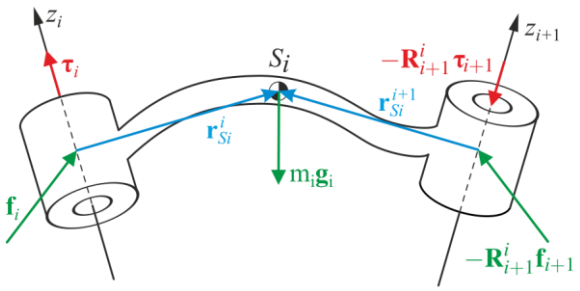


Fig. 6. Diagram of forces and torques acting on the i^{th} link of the robot

In the forward recursion, spatial velocities and accelerations of the centre of mass of each link are computed starting from the base, with the motion of each link depending on the preceding one. In the backward recursion, forces and torques acting on the joints are determined by iterating from the end-effector back to the base. Fig. 6 illustrates the force and torque interactions acting on a single manipulator link, represented in the coordinate frame fixed to the corresponding link.

In the diagram, vector $\mathbf{f}_i$ represents the force exerted by link $i-1$ on link i, while $\mathbf{f}_{i+1}$ denotes the force exerted by link $i+1$ on link i. In accordance with Newton's third law, this force is equal in magnitude and opposite in direction to the reaction force acting on link $i+1$, and must be rotated into the local coordinate frame of link i using the rotation matrix $\mathbf{R}^i_{i+1}$. A similar notation applies to the torques $\boldsymbol{\tau}_i$ and $-\mathbf{R}^i_{i+1}\boldsymbol{\tau}_{i+1}$. The torque vector can be expressed as:

$$\boldsymbol{\tau}_i = [\tau_{xi} \quad \tau_{yi} \quad u_i]^T. \tag{57}$$

The actuator torque u_i is the projection of the torque vector onto the z-axis of the i^{th} link. The gravitational force $m_i\mathbf{g}_i$ is defined in the local coordinate frame using the gravity vector function $\mathbf{g}_i$.

The dynamics analysis is based on the force and torque equilibrium equations for the i^{th} link, written as:

$$m_i\boldsymbol{a}_{S_i} = \mathbf{f}_i - \mathbf{R}^i_{i+1}\mathbf{f}_{i+1} + m_i\mathbf{g}_i \tag{58}$$

$$\mathbf{I}_i\boldsymbol{\varepsilon}_i + \boldsymbol{\omega}_i \times (\mathbf{I}_i\boldsymbol{\omega}_i) = \boldsymbol{\tau}_i - \mathbf{R}^i_{i+1}\boldsymbol{\tau}_{i+1} + \mathbf{f}_i \times \mathbf{r}^i_{c_i} - \left(\mathbf{R}^i_{i+1}\mathbf{f}_{i+1}\right) \times \mathbf{r}^{i+1}_{S_i} - \boldsymbol{\tau}_{f_i} \tag{59}$$

where: $\mathbf{a}_{S_i}$ – acceleration of the centre of mass of the i^{th}link in its local frame, $\boldsymbol{\varepsilon}_i$ – angular acceleration in the local frame, $\mathbf{I}_i$ – inertia tensor of the i^{th} link with respect to its centre of mass, $\mathbf{r}^i_{S_i}$ – position vector from the joint to the centre of mass of link i, $\mathbf{r}^{i+1}_{S_i}$ – position vector from joint $i+1$ to the centre of mass of link i, $\boldsymbol{\tau}_{f_i}$ – torque due to joint friction.

After rearranging equations (58) and (59), the following recursive relations are obtained:

$$\mathbf{f}_i = m_i\mathbf{a}_{S_i} + \mathbf{R}^i_{i+1}\mathbf{f}_{i+1} - m_i\mathbf{g}_i, \tag{60}$$

$$\boldsymbol{\tau}_i = \mathbf{I}_i\boldsymbol{\varepsilon}_i + \boldsymbol{\omega}_i \times (\mathbf{I}_i\boldsymbol{\omega}_i) + \mathbf{R}^i_{i+1}\boldsymbol{\tau}_{i+1} - \mathbf{f}_i \times \mathbf{r}^i_{S_i} + \left(\mathbf{R}^i_{i+1}\mathbf{f}_{i+1}\right) \times \mathbf{r}^{i+1}_{S_i} + \boldsymbol{\tau}_{f_i}, \tag{61}$$

By applying these equations iteratively from $i = n$ to $i = 1$, it is possible to compute the forces and torques acting on each joint of the robot. The recursion begins by considering the force and torque exerted by the tool mounted on the robot flange:

$$\mathbf{f}_{n+1} = \mathbf{f}_T = m_T\mathbf{a}_{c_T} - m_T\mathbf{g}_6 \tag{62}$$

$$\boldsymbol{\tau}_{n+1} = \boldsymbol{\tau}_T = \mathbf{I}_T\boldsymbol{\varepsilon}_6 + \boldsymbol{\omega}_6 \times (\mathbf{I}_T\boldsymbol{\omega}_6) - \mathbf{f}_T \times \mathbf{r}^7_{S_T} \tag{63}$$

The gravitational acceleration vector for each link, with the base of the manipulator attached to a horizontal surface, is defined as:

$$\mathbf{g}_i = -\mathbf{R}^B_i g[0 \quad 0 \quad 1]^T \tag{64}$$

To obtain a complete dynamic solution, it is necessary to express the Cartesian-space variables $\mathbf{a}_{S_i}$, $\boldsymbol{\omega}_i$ and $\boldsymbol{\varepsilon}_i$ in terms of the configuration variables $\mathbf{q}$, $\dot{\mathbf{q}}$, $\ddot{\mathbf{q}}$. The angular velocity of each link is computed from the angular velocity in the base frame:

$$\boldsymbol{\omega}_i = (\mathbf{R}^B_i)^{\mathrm{T}}\boldsymbol{\omega}^B_i. \tag{65}$$

Angular acceleration is computed as:

$$\boldsymbol{\varepsilon}_i = (\mathbf{R}_i^B)^{\mathrm{T}} \dot{\boldsymbol{\omega}}_i^B. \tag{66}$$

Due to the complexity of the results, detailed expressions are provided in the part A of the appendix [35].

The centre of mass positions of the links relative to their local frames were defined previously using equations (37)-(43). The vector from the end of link iii to its centre of mass is given by:

$$\mathbf{r}_{S_i}^{i+1} = \mathbf{r}_{S_i}^i - \mathbf{o}_{i+1}^i. \tag{67}$$

The acceleration of the centre of mass of the i^{th} link is calculated using forward recursion as:

$$\mathbf{a}_{S_i} = \mathbf{a}_i + \boldsymbol{\varepsilon}_i \times \mathbf{r}_{S_i}^i + \boldsymbol{\omega}_i \times \left(\boldsymbol{\omega}_i \times \mathbf{r}_{S_i}^i\right). \tag{68}$$

To verify the correctness of the dynamic equations derived using the Euler–Lagrange formalism, the resulting equations were reformulated to match the structure of equation (56). The derived matrices from both methods were found to be equivalent, confirming that the dynamic model has been correctly formulated.

7. PHYSICAL PARAMETERS OF THE MODEL

One of the most challenging aspects of modelling a robotic system is determining its physical parameters, such as masses, mass moments of inertia, and friction coefficients (Fig. 7). The masses of individual links of the IRB 2400 manipulator were estimated based on technical documentation of the experimental setup components.

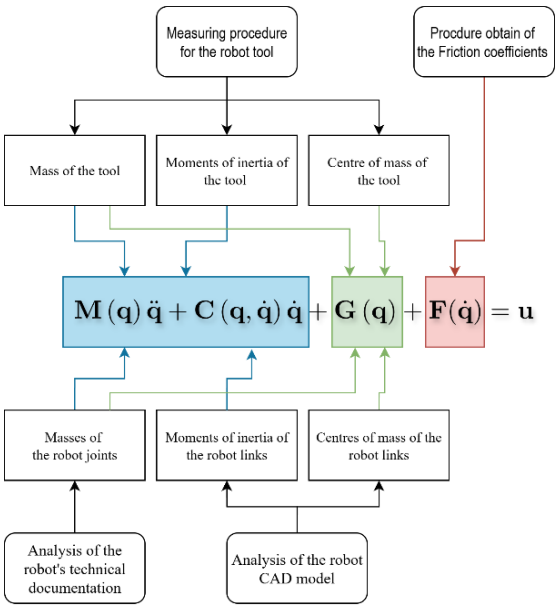


Fig. 7. Diagram of the procedure for determining the physical parameters of the model

The positions of centres of mass and the values of mass moments of inertia were determined through analysis of the robot's CAD model [14,36,37]. The estimated values are presented in Tab. 3. Any tool can be mounted on the robot, as long as it does not exceed the payload limits defined by the manufacturer. In this work, an electric spindle was used as the end-effector, and its physical parameters are shown in Tab. 4. The values were obtained using the robot's internal measurement procedure, which analyses force sensor data for various tool positions to estimate the tool's mass, centre of mass, and inertia.

Tab. 3. Physical parameters of the robot

Parameter	Sym.	Link 1	Link 2	Link 3
Mass (kg)	m_i	192	26.5	25.7
Centre of mass (mm)	S_{ix}	21	253.4	139.5
	S_{iy}	22	47.7	-99.5
	S_{iz}	-188.5	7	-9.7
Moments of inertia (kgm^2)	I_{ixx}	9.82183	0.23517	0.0143
	I_{iyy}	6.16127	1.22338	0.01412
	I_{izz}	8.30224	1.16539	0.00892
	I_{ixy}	0.50571	0.58498	-0.0259
	I_{ixz}	-1.0596	0.22733	0.01089
	I_{iyz}	0.10946	0.56479	-0.0003
		Link 4	**Link 5**	**Link 6**
Mass (kg)	m_i	29.7	2.8	0.8
Centre of mass (mm)	S_{ix}	0.131	-0.587	0.208
	S_{iy}	-1.81	-0.245	0.035
	S_{iz}	-296.7	0.540	72.270
Moments of inertia (kgm^2)	I_{ixx}	0.74128	0.0034	0.00002
	I_{iyy}	0.71936	0.00332	0
	I_{izz}	0.09626	0.00419	0
	I_{ixy}	0.208	0.00023	0
	I_{ixz}	0.035	0.00023	0
	I_{iyz}	0.000711	0.0003	0

Tab. 4. Physical parameters of the spindle

Parameter	Sym.	Spindle
Mass (kg)	m_T	12.7
Centre of mass (mm)	S_{Tx}	-38.3
	S_{Ty}	0
	S_{Tz}	129.3
Moments of inertia (kgm^2)	I_{Txx}	0.269
	I_{Tyy}	0.274
	I_{Tzz}	0.193
	I_{Txy}	0
	I_{Txz}	0
	I_{Tyz}	0

The TCP was defined as the furthest point on the tool mounted in the spindle chuck. The coordinates of this point, as recorded in transformation matrix (12), are:

$$\begin{cases} x_T = 240\ mm \\ y_T = 0 \\ z_T = 162\ mm \end{cases}. \tag{69}$$

Friction coefficients for the joint friction model described in equation (54) were determined experimentally, following the methodology described in [38]. The IRC5 controller used to operate the IRB 2400 manipulator enables real-time access to motion and torque data through the External Guided Motion interface [39] or via test signals managed through the Tune Master application [40] on a PC. To estimate friction parameters, a series of experiments was conducted where each joint was moved individually at a specified angular velocity near a reference position. During these tests, the remaining joints were positioned to minimize gravitational effects. While this was not feasible for joints 2 and 3 due to the robot's

DOI 10.2478/ama-2025-0064

mounting constraints, suitable configurations were found for the remaining joints. The gravitational torque was compensated by comparing actuator torques measured during motion in both positive and negative directions in the same joint position [41].

When a manipulator joint moves at constant velocity, inertial effects its effect is absorbed into the link inertias, and the motion equation simplifies to:

$$\tau_{fi}(\dot{q}_i, q_i) + \tau_{gi}(q_i) = u_i. \tag{70}$$

where: u_i – actuator torque for joint i, τ_{f_i} – friction torque, τ_{g_i} – gravitational torque. Assuming friction torque is symmetric around zero velocity, the torques recorded for constant positive and negative velocities ${\dot{q}_i}^{\dagger}$ give the result:

$$\begin{cases} \tau_{f_i}({\dot{q}_i}^{\dagger}) + \tau_{g_i}({q_i}^{\dagger}) = u_i^{+}, \\ \tau_{f_i}(-{\dot{q}_i}^{\dagger}) + \tau_{g_i}({q_i}^{\dagger}) = u_i^{-}. \end{cases} \tag{71}$$

Subtracting both equations gives:

$$\tau_{f_i}({\dot{q}_i}^{\dagger}) - \tau_{f_i}(-{\dot{q}_i}^{\dagger}) = u_i^{+} - u_i^{-}, \tag{72}$$

Assuming $\tau_{f_i}(-{\dot{q}_i}^{\dagger}) \cong -\tau_{f_i}({\dot{q}_i}^{\dagger})$, the friction torque can be estimated as:

$$\tau_{f_i}({\dot{q}_i}^{\dagger}) = \frac{u_i^{+} - u_i^{-}}{2}. \tag{73}$$

Fig. 8 presents the measured angular position, velocity, and actuator torque for joint 2, moving in both positive and negative directions at a velocity of ${\dot{q}_2}^{\dagger} = 9{,}3\ \mathrm{rad}/s$ near the position ${q_2}^{\dagger} = 0$.

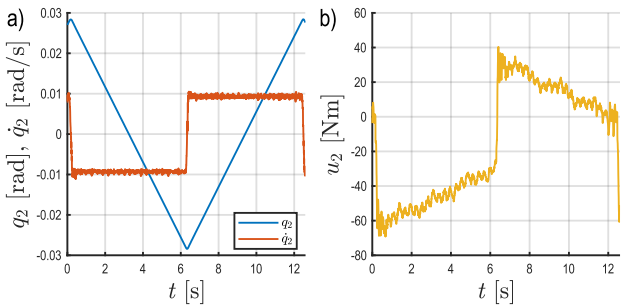


Fig. 8. a) Angular position and velocity of joint 2 at a constant velocity, b) recorded actuator torque

Since most of the friction phenomena described by equation (54) occur at low velocities, step sizes for increasing velocity during the tests were defined as:

$$\Delta{\dot{q}_i}^{\dagger} = \begin{cases} 0{,}00001\,\frac{rad}{s}, \text{ for } {\dot{q}_i}^{\dagger} = \langle 0{,}00001; 0{,}015) \frac{rad}{s} \\ 0{,}0002\,\frac{rad}{s}, \text{ for } {\dot{q}_i}^{\dagger} = \langle 0{,}015; 0{,}2) \frac{rad}{s} \\ 0{,}001\,\frac{rad}{s}, \text{ for } {\dot{q}_i}^{\dagger} = \langle 0{,}2; 0{,}5\rangle \frac{rad}{s} \end{cases} \tag{74}$$

Using the recorded data, values of ${\dot{q}_i}^{\dagger}$ and u_i at reference positions were extracted. For joint 2, the point ${q_2}^{\dagger} = 0$ was chosen. These values were substituted into equation (73) to estimate the friction torque as a function of speed. To determine the coefficients, *MATLAB's Curve Fitter* toolbox was used to fit the analytical model to the experimental data. Fig. 9 shows the comparison between measured and estimated friction torques for joint 2 at ${q_2}^{\dagger} = 0$, limited to the $0 - 0.15\ rad/s$ range to emphasize the Stribeck effect.

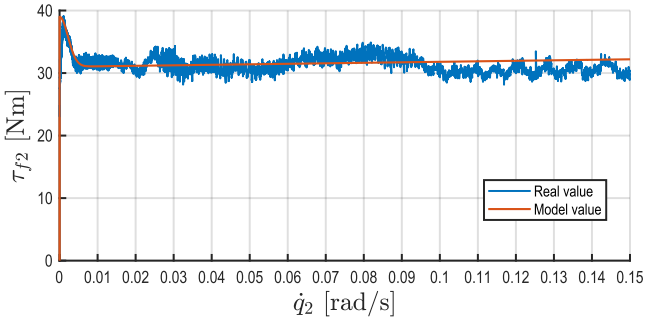


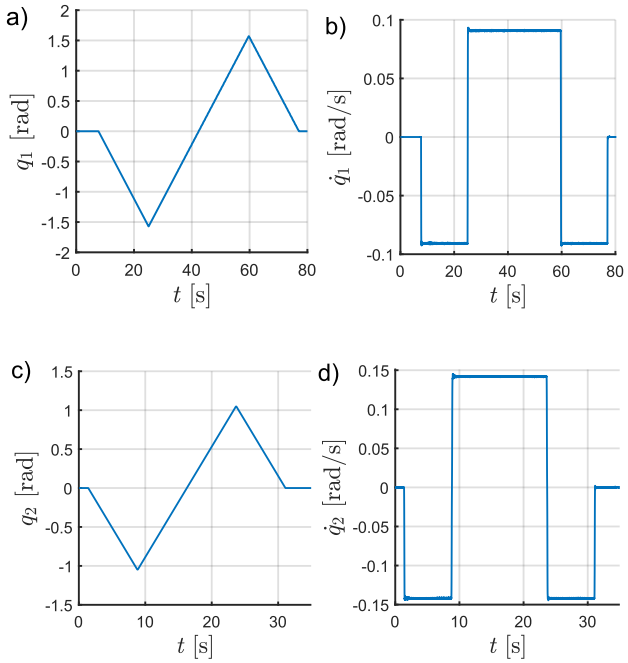
Fig. 9. Experimental and estimated friction torque values for joint 2

The experiment was repeated for all six joints of the IRB 2400 robot. The resulting friction parameters are summarized in Tab. 5.

Tab. 5. Friction model parameters for each joint

Joint	F_S [Nm]	F_C [Nm]	F_v [Nms/rad]	ω_{St} [rad/s]
1	36.3	24.2	11.85	0.0085
2	39.2	31.2	8.323	0.0031
3	25.3	16.9	9.3613	0.0094
4	19.2	7.9	7.1228	0.0175
5	39.7	13.7	7.2654	0.0176
6	17.5	8.8	2.323	0.0162

To validate the agreement between the robot model and the real system, the inverse dynamics problem was solved using motion data recorded from the physical robot. The computed actuator torques were then compared with measured values. Fig. 10 and Fig. 11 present the angular positions and velocities for all six joints used as simulation inputs. Fig. 12 shows the comparison of modelled and measured torques.



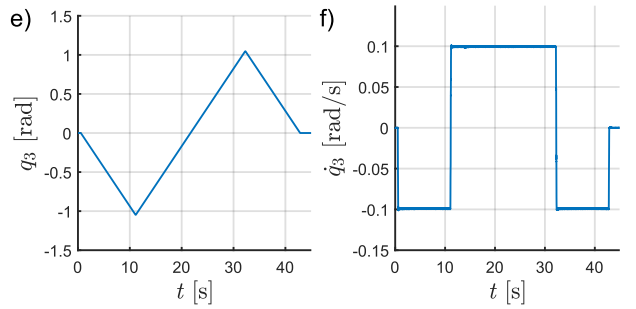


Fig. 10. Inputs for simulation – joint angles and velocities (joints 1-3)

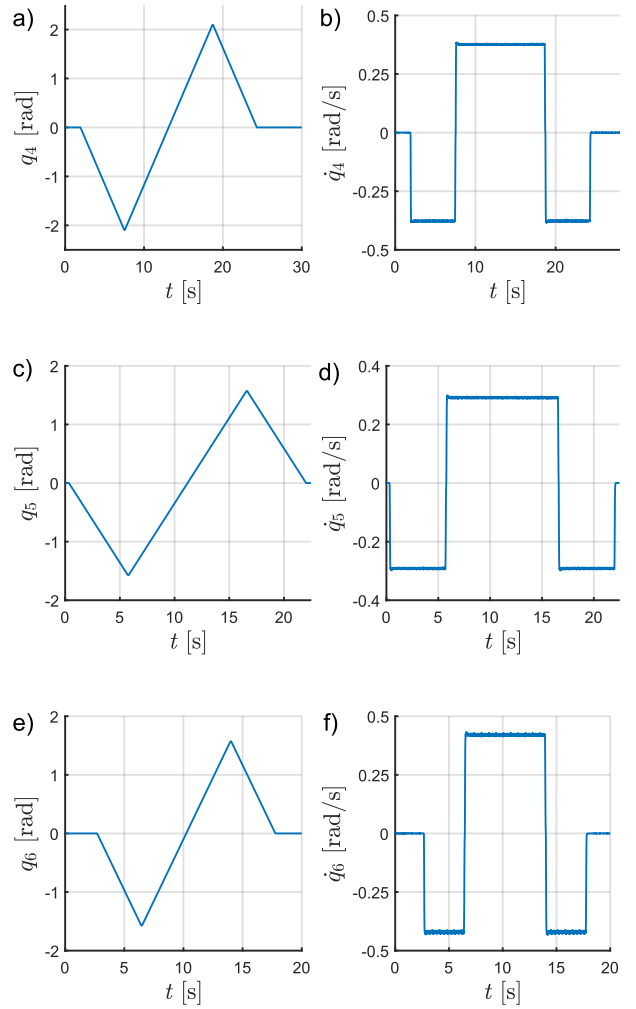
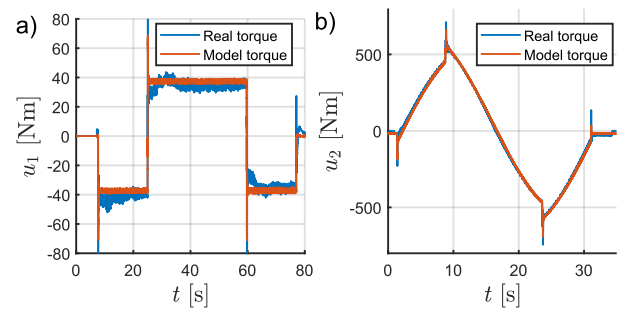


Fig. 11. Inputs for simulation – joint angles and velocities (joints 4-6)



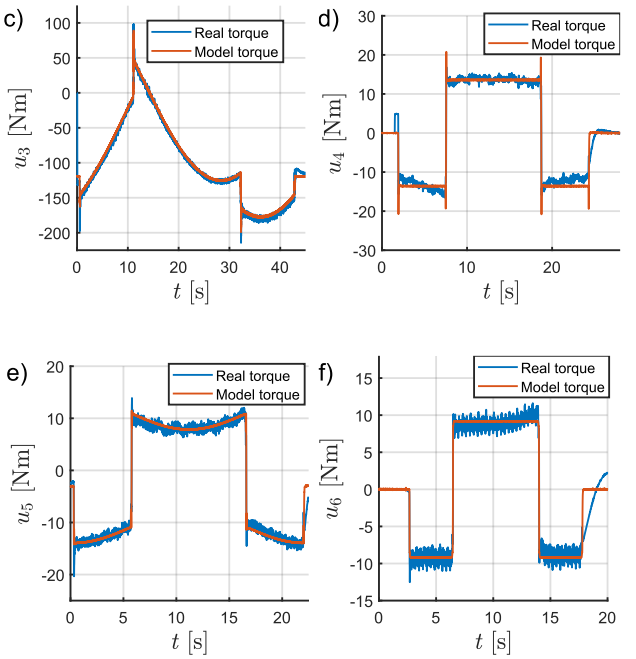


Fig. 12. Modelled vs. real actuator torques for joints 1–6

To evaluate model accuracy, three metrics were used:

– Mean Absolute Error (MAE):

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}\left|u_{model,i} - u_{real,i}\right| \tag{75}$$

– Root Mean Square Error (RMSE):

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(u_{model,i} - u_{real,i}\right)^2} \tag{76}$$

– Percentage RMSE relative to average torque:

$$\mathrm{RMSE/AVG}\,[\%] = \frac{RMSE}{\frac{1}{N}\sum_{i=1}^{N}\left|u_{real,i}\right|} \cdot 100\% \tag{77}$$

where $u_{real,i}$ is the real actuator torque, and $u_{model,i}$ is the simulated torque.

Tab. 6. Accuracy indicators for each joint

Joint	$\boldsymbol{MAE}$ $(\boldsymbol{Nm})$	$\boldsymbol{RMSE}$ $(\boldsymbol{Nm})$	$\boldsymbol{AVG(u_{real})}$ $(\boldsymbol{Nm})$	$\boldsymbol{RMSE/AVG}$ [%]
1	3.16	4.08	35.13	11.61
2	8.52	10.89	319.60	3.41
3	2.9	3.79	102.91	3.69
4	0.73	1.17	13.69	8.56
5	0.58	0.78	10.03	7.80
6	0.64	0.81	8.96	9.06

The analysis of the obtained results confirms a good agreement between the model predictions and the measurements recorded on the real system. Notably, the Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) indicators for joints 2 and 3 represent approximately 3.5% of the average driving torque (Tab. 6). This suggests that the model provides a very accurate representation of the driving torque dynamics for these joints. In the case of joints 4, 5, and 6, although the absolute values of MAE and RMSE are relatively low, the RMSE/AVG index ranges between 7.8% and 9.06%, indicating a slightly less accurate, yet still satisfactory,

DOI 10.2478/ama-2025-0064

model performance. This is confirmed by the comparative plots presented in Fig. 12b and Fig. 12c. The highest modelling error is observed for joint 1, where the RMSE/AVG index reaches 11.61%. As shown in Fig. 12a, the driving torque response for this joint exhibits oscillations and a slower stabilization compared to the model prediction. For the remaining joints, the RMSE/AVG index does not exceed 10%, and the plotted responses indicate that the model captures the torque dynamics with satisfactory accuracy.

8. CONCLUSIONS

This paper presents a comprehensive approach to modelling an industrial robot, using the ABB IRB 2400 as an example. The developed kinematic model, based on the Denavit-Hartenberg method, enabled precise determination of transformations between consecutive links of the robot and facilitated the computation of the Jacobian matrix, which is essential for analysing the end-effector's velocity and task-space forces. While the classical and modified Denavit-Hartenberg formulations are kinematically equivalent, the MDH parametrisation adopted here aligns with the controller's native frame conventions, avoids artificial zero link lengths and ad hoc offsets, and yields better-conditioned Jacobians around the spherical wrist, which in our toolchain translates into faster evaluation of forward kinematics and Jacobians. More importantly, the presented framework goes beyond motion-only simulation by explicitly modelling gravity, Coriolis/centrifugal effects and an identified Stribeck-enhanced friction law. As a result, it predicts joint torques and end-effector wrenches with accuracy sufficient for real-time use, exposes the impact of singular configurations through the Jacobian and its conditioning, and enables assessments that kinematics-only tools cannot provide.

In the dynamic modelling process, two methodologies were employed: the Euler-Lagrange method and the Newton-Euler method. This dual approach allowed for the derivation of two equivalent equations of motion, effectively reducing the risk of modelling errors. Subsequently, the physical parameters of the model (masses, moments of inertia, and friction coefficients) were identified, and the model was validated in a simulation environment. The verification relied on controller-reported actuator torques streamed via EGM at 250 Hz; these values are computed by the internal model of the IRC5 controller and their absolute accuracy is not disclosed by the manufacturer. This limitation was mitigated by cross-checking trends against CAD-consistent gravity loads and by using two independent dynamic derivations (Euler–Lagrange and Newton–Euler), which reduced the risk of algebraic inconsistencies and improved numerical robustness. The resulting model already enables feedforward torque compensation and singularity-aware path planning for robotic machining, feasibility checks and peak-load prediction in digital-twin workflows, including large-scale additive processes, and rapid what-if studies on process parameters without hardware modification.

The comparison of simulation results with reference data showed sufficient agreement, confirming the correctness of the developed dynamic model. The obtained model can be confidently used for further studies. It serves as a foundation for developing a digital twin of the robot, which can be employed in simulation-based testing of advanced control algorithms. In the future, the model can also be extended to incorporate phenomena related to joint compliance, enabling more accurate analysis of the robot's interaction with its environment.

REFERENCES

1. Robotics IIF of. International Federation of Robotics [Internet]. IFR International Federation of Robotics. Available from: https://ifr.org/wr-industrial-robots
2. Verl A, Valente A, Melkote S, Brecher C, Ozturk E, Tunc LT. Robots in machining. CIRP Annals [Internet]. 2019 Jan 1;68(2):799-822. Available from: https://doi.org/10.1016/j.cirp.2019.05.009
3. Burghardt A, Szybicki D, Kurc K, Muszyńska M. Optimization of process parameters of edge robotic deburring with force control. International Journal of Applied Mechanics and Engineering. 2016 Dec 1;21(4):987-95. Available from: https://doi.org/10.1515/IJAME-2016-0060
4. Burghardt A, Szybicki D, Kurc K, Muszyńska M, Mucha J. Experimental Study of Inconel 718 Surface Treatment by Edge Robotic Deburring with Force Control. Strength of Materials [Internet]. 2017 Jul 1;49(4):594-604. Available from: https://doi.org/ 10.1007/S11223-017-9903-3
5. Kurc K, Burghardt A, Szybicki D, Gierlak P, Łabuński W, Muszyńska M, et al. Robotic machining in correlation with a 3D scanner. Mechanics and Mechanical Engineering [Internet]. 2020 Sep 1;24(1):36-41. Available from: https://doi.org/10.2478/mme-2020-0003
6. Szybicki D. Zastosowanie idei cyfrowych bliźniaków w projektowaniu oraz programowaniu stacji zrobotyzowanych. Rzeszów: Oficyna Wydawnicza Politechniki Rzeszowskiej. 2023;143–152.
7. Choi SH, Park KB, Roh DH, Lee JY, Mohammed M, Ghasemi Y, et al. An integrated mixed reality system for safety-aware human-robot collaboration using deep learning and digital twin generation. Robotics and Computer-Integrated Manufacturing. 2022 Feb 1;73:102258. Available from:https://doi.org/10.1016/J.RCIM.2021.102258
8. Gregor R, Babinec A, Duchoň F, Dobiš M. Hand Guiding a Virtual Robot Using a Force Sensor. Acta Mechanica et Automatica [Internet]. 2021 Sep 27;15(3):177-86. Available from: https://doi.org/10.2478/ama-2021-0023
9. Ugarte M, Etxeberria L, Unamuno G, Bellanco JL, Ugalde E. Implementation of Digital Twin-based Virtual Commissioning in Machine Tool Manufacturing. Procedia Computer Science [Internet]. 2022 Jan 1;200:527-36. Available from: https://doi.org/10.1016/j.procs.2022.01.250
10. Striffler N, Voigt T. Concepts and trends of virtual commissioning – A comprehensive review. Journal of Manufacturing Systems [Internet]. 2023 Dec 1;71:664-80. Available from: https://doi.org/10.1016/j.jmsy.2023.10.013
11. Mohammad M, Babriya V, Sobh T. Modeling A Deburring Process, Using DELMIA V5®. In: Iskander M, Kapila V, Karim MA, editors. Technological Developments in Education and Automation. Dordrecht: Springer Netherlands. 2010; 549–58. https://doi.org/10.1007/978-90-481-3656-8_100
12. Garbev A, Atanassov A. Comparative Analysis of RoboDK and Robot Operating System for Solving Diagnostics Tasks in Off-Line Programming. In: 2020 International Conference Automatics and Informatics (ICAI) [Internet]. 2020; 1-5. Available from: https://doi.org/10.1109/ICAI50593.2020.9311332
13. Leali F, Pellicciari M, Pini F, Berselli G, Vergnano A. An Offline Programming Method for the Robotic Deburring of Aerospace Components. In: Neto P, Moreira AP, editors. Robotics in Smart Manufacturing. Berlin, Heidelberg: Springer. 2013; 1-13. https://doi.org/10.1007/978-3-642-39223-8_1
14. Reinl C, Friedmann M, Bauer J, Pischan M, Abele E, von Stryk O. Model-based off-line compensation of path deviation for industrial robots in milling applications. In: 2011 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM) [Internet]. 2011; 367-72. Available from: https://doi.org/10.1109/AIM.2011.6027113
15. Gierlak P. Position/Force Control of Manipulator in Contact with Flexible Environment. Acta Mechanica et Automatica [Internet]. 2019 Mar 1;13(1):16-22. Available from:https://doi.org/10.2478/ama-2019-0003

16. Gierlak P. Adaptive Position/Force Control of a Robotic Manipulator in Contact with a Flexible and Uncertain Environment. Robotics [Internet]. 2021 Feb 12;10(1):32.
Available from: https://doi.org/10.3390/robotics10010032
17. Prajadhiana KP, Manurung YHP, Fateri M, Choo HL, Rahaman WEWA, Adenan MS, et al. Distortion analysis of WAAM component using thermo-mechanical, inherent strain and experimental methods. Prog Addit Manuf [Internet]; 2025 Apr 25.
Available from: https://doi.org/10.1007/s40964-025-01102-8
18. Manurung YHP, Prajadhiana KP, Adenan MS, Awiszus B, Graf M, Haelsig A. Analysis of material property models on WAAM distortion using nonlinear numerical computation and experimental verification with P-GMAW. ArchivCivMechEng [Internet]. 2021 Feb 16;21(1):32.
Available from: https://doi.org/10.1007/s43452-021-00189-4
19. Prajadhiana KP, Manurung YHP, Bauer A, Adenan MS, Syahriah NI, Mohamed MA et al. Experimental verification of computational and sensitivity analysis on substrate deformation and plastic strain induced by hollow thin-walled WAAM structure. Rapid Prototyping Journal [Internet]. 2021 Oct 11;28(3):559-72.
Available from: https://doi.org/10.1108/RPJ-06-2020-0135
20. Prajadhiana KP, Taufek T, Wan Abdul Rahaman WE, Manurung YHP, Adenan MS, Mat MF, et al. Distortion Analysis Method for Wire Arc Additive Manufacturing Component Using Thermomechanical Computation with Enhanced Separation and Deposition Algorithm. 3D Printing and Additive Manufacturing [Internet]. 2024 Aug;11(4):e1616–28.
Available from: https://doi.org/10.1089/3dp.2023.0033
21. Ahmad SN, Manurung YH, Adenan MS, Yusof F, Mat MF, Prajadhiana KP, et al. Experimental validation of numerical simulation on deformation behaviour induced by wire arc additive manufacturing with feedstock SS316L on substrate S235. Int J Adv Manuf Technol [Internet]. 2022 Mar 1;119(3):1951-64.
Available from: https://doi.org/10.1007/s00170-021-08340-4
22. Ahmad SN, Manurung YHP, Prajadhiana KP, Busari YO, Mat MF, Muhammad N, et al. Numerical modelling and experimental analysis on angular strain induced by bead-on-plate SS316L GMAW using inherent strain and thermomechanical methods. Int J Adv Manuf Technol [Internet]. 2022 May 1;120(1):627-44.
Available from: https://doi.org/10.1007/s00170-022-08684-5
23. Zhang D, Wang L, Bi Z. Kinematic, Dynamic Modeling and Remote Control of a Robotic Machine. In: 2007 IEEE International Conference on Control and Automation [Internet]. 2007; 2441-6. Available from: https://doi.org/ 10.1109/ICCA.2007.4376801
24. Ye SH, Wang Y, Ren YJ, Li DK. Robot Calibration Using Iteration and Differential Kinematics. J Phys: Conf Ser [Internet]. 2006 Oct;48(1):1.
Available from: https://doi.org/10.1088/1742-6596/48/1/001
25. Motta JMST, de Carvalho GC, McMaster RS. Robot calibration using a 3D vision-based measurement system with a single camera. Robotics and Computer-Integrated Manufacturing [Internet]. 2001 Dec 1;17(6):487-97.
Available from: https://doi.org/10.1016/S0736-5845(01)00024-2
26. Zheng Q, Jia J, Zhu P, Xiao Y, Ma W, Zhao S. Kinematics analysis and trajectory planning of ABB-IRB2400 robot. UPB Sci Bull. 84.
27. Garcia JG, Robertsson A, Ortega JG, Johansson R. Automatic Calibration Procedure for a Robotic Manipulator Force Observer. In: Proceedings of the 2005 IEEE International Conference on Robotics and Automation [Internet]. 2005; 2703-8. Available from: https://doi.org/ 10.1109/ROBOT.2005.1570522
28. Product specification - IRB 2400 [Internet]. Available from: https://library.e.abb.com/public/c648b6ab496748c1b22499a0ac451da6/3HAC042195%20PS%20IRB%202400%20on%20IRC5-en.pdf?x-sign=9s04sLzDVfl1FMiR7YKzqISKK8NL1YaWeKP8nDcheXVfeqWdc57xpg0NeUSNlqAE
29. Application manual Force Control RobotWare 6.13. ABB ROBOTICS; 2021.
30. Spong MW, Hutchinson S, Vidyasagar M. Robot Modeling and Control. 2nd ed. John Wiley & Sons Ltd; 2020.
31. Takeuchi A, Kato D, Sekioka M, Hirogaki T, Aoyama E. Study on Singularity Passing Method for Large Industrial Robot Based on Proper Element Improvement of Inverse MatrixStudy on Singularity Passing Method for Large Industrial Robot Based on Proper Element Improvement of Inverse Matrix. IJMERR [Internet]. 2024;13(2):278-83.
Available from: https://doi.org/10.18178/ijmerr.13.2.278-283
32. Kozłowski K, Dutkiewicz P, Wróblewski W. Modelowanie i sterowanie robotów. Wydawnictwo Naukowe PWN; 2003.
33. Craig JJ. Introduction to robotics : mechanics and control. 4th ed. Pearson; 2021.
34. Olsson H, Åström KJ, Canudas De Wit C, Gäfvert M, Lischinsky P. Friction Models and Friction Compensation. European Journal of Control. 1998 Jan 1;4(3):176-95.
Available from:https://doi.org/10.1016/S0947-3580(98)70113-X
35. Appendix to Comprehensive Model of the Abb Irb 2400 for Simulation and Control Applications [Internet]. GitHub.
Available from: https://github.com/pobal/A-COMPREHENSIVE-MODEL-OF-THE-ABB-IRB-2400-FOR-SIMULATION-AND-CONTROL-APPLICATIONS.git
36. Wu J, Wang J, You Z. An overview of dynamic parameter identification of robots. Robotics and Computer-Integrated Manufacturing [Internet]. 2010 Oct 1;26(5):414-9.
Available from: https://doi.org/10.1016/j.rcim.2010.03.013
37. del Sol E, Pagala P, King R, Ferre M. External force estimation for telerobotics without force sensor. Advances in Intelligent Systems and Computing [Internet]. 2014;253:631-44.
Available from: https://doi.org/10.1007/978-3-319-03653-3_45/COVER
38. Indri M, Lazzero I, Antoniazza A, Bottero AM. Friction modeling and identification for industrial manipulators. In: 2013 IEEE 18th Conference on Emerging Technologies & Factory Automation (ETFA) [Internet]. Cagliari Italy. IEEE. 2013;1-8.
Available from: https://doi.org/10.1109/ETFA.2013.6647958
39. Obal P, Gierlak P. EGM Toolbox-Interface for Controlling ABB Robots in Simulink. Sensors 2021; 21(22):7463.
Available from: https://doi.org/10.3390/S21227463
40. Obal P, Burghardt A, Kurc K, Szybicki D, Gierlak P. Monitoring the Parameters of Industrial Robots. In: Hanus R, Mazur D, Kreischer C, editors. Methods and Techniques of Signal Processing in Physical Measurements [Internet]. Cham: Springer International Publishing; 2019;230–8. Lecture Notes in Electrical Engineering; 548.
Available from: https://doi.org/ 10.1007/978-3-030-11187-8_19
41. Bittencourt AC, Gunnarsson S. Static friction in a robot joint-modeling and identification of load and temperature effects. Journal of Dynamic Systems. Measurement and Control. Transactions of the ASME [Internet]. 2012 Sep 1;134(5).
Available from: https://doi.org/10.1115/1.4006589/395008

Paweł Obal: https://orcid.org/0000-0003-4269-447X

Piotr Gierlak: https://orcid.org/0000-0003-4545-8253