

The Impact of Simplifications of the Dynamic Model on the Motion of a Six-Jointed Industrial Articulated Robotic Arm Movement

Mehdi Fazilat ^{1*}, Nadjat Zioui ²

^{1,2} Université du Québec à Trois-Rivières, 3351 Bd des Forges, Trois-Rivières, QC G8Z 4M3, Canada

Email: ¹ mehdi.fazilat@uqtr.ca, ² nadjat.zioui@uqtr.ca

*Corresponding Author

Abstract—This research investigates the impact of model simplification on the dynamic performance of an ABB IRB-140 six-jointed industrial robotic arm, concentrating on torque prediction and energy consumption. The entire mathematical model of forward, reverse, differential kinematics, and dynamic model proposed based on the technical specifications of the arm, and to obtain the center of the mass and inertia matrices, which are essential components of the dynamic model, Utilizing Solidworks, we developed three CAD/CAM models representing the manipulator with varying detail levels, such as simplified, semi-detailed, and detailed. Our findings indicate minor differences in the model's torque and energy consumption graphs. The semi-detailed model consumed the most energy, except for joint 1, with the detailed model showing a 0.53% reduction and the simplified model a 6.8% reduction in energy consumption. Despite these variations, all models proved effective in predicting the robot's performance during a standard 30-second task, demonstrating their adequacy for various industrial applications. This research highlights the balance between computational efficiency and accuracy in model selection. While the detailed model offers the highest precision, it demands more computational resources, which is suitable for high-precision tasks. In discrepancy, simplified, less precise models offer computational efficiency, making them adequate for specific scenarios. Our study provides critical insights into selecting dynamic models in industrial robotics. It guides the optimization of performance and energy efficiency based on the required task precision and available computational resources. This comprehensive comparison of dynamic models underscores their applicability and effectiveness in diverse industrial settings.

Keywords—Industrial Robot Manipulators; Computer-Aided Design (CAD); D-H Representation; Dynamic Model and Simplifications; Energy Consumption; Computational Efficiency in Robotics; Model Accuracy in Robotics; Precision Engineering in Robotics.

I. INTRODUCTION

In recent years, considerable attention has been dedicated to researching robotic arm control within the broad domain of industrial robotics. This focus has notably enhanced the performance and applicability of autonomous systems in controlling joint movements [1]. Robotic arm control methods are various, each owning particular advantages and challenges. Primarily, these strategies are anchored in dynamic models, where variables are represented as torque or forces, depending on the joint's character, articulated or prismatic. Unlike robust controllers, which can adjust

unconsidered errors in the dynamic model [2][3], precise controllers are intimately dependent on the accuracy of the dynamic model [4][5] and the impact of proper and appropriate models on control strategy.

The challenge of dynamic modeling in robotics is well-documented, with studies highlighting its complexity [6]-[12]. Two fundamental methods can provide model robot dynamics, the Newton–Euler formulation and the Euler–Lagrange formulation. The Newton–Euler formulation enumerates each term separately, providing a direct way to calculate forces and torques in the robot's joints and links. On the other hand, the Euler–Lagrange formulation, which is based on the energy properties of mechanical systems, computes motion equations by balancing the kinetic and potential energies of the system. This approach requires detailed knowledge of each robot link's inertia and center of mass, a complex but crucial aspect for accurate dynamic modeling [13]. The latest necessitates detailed knowledge of each robot link's inertia and center of mass, a requirement often met through model parameter identification [14]-[22] or approximation methods [23]-[31].

However, these approaches, whether through measurement inaccuracies or oversimplifications, introduce errors. Understanding these complexities is not just an academic exercise but essential in the real world. This study's insights are particularly relevant in advanced industrial automation, where precision and efficiency in robotic arm control are paramount. Our work aims to inform complex manufacturing and automation environments more effectively and efficiently by exploring the trade-offs between model simplicity and accuracy. To clarify the impact of these modeling techniques, this study compares three dynamic models of an articulated robotic arm developed using CAD models in Solidworks. The first model, highly detailed, closely emulates the actual robot design. The second, an approximate model, utilizes parallelepipeds with specific density choices to mirror link masses. The third, a simplified model, treats links as rods, a prevalent approach in literature for its ease [32]-[39]. These models are evaluated as torque and energy consumption predictors to assess the implications of model simplifications on these crucial factors.

The accuracy of a dynamic model depends on the precise identification of the mass center and inertia matrices. Detailed dimensional specifications and scientific methods



such as CAD modeling, Experimental Modal Analysis, and Inertia Measurement Units are some methods to obtain this required data in the mechanical parts. CAD modeling presents detailed dimensional and mass distribution data, while Experimental Modal Analysis and Inertia Measurement Units directly measure inertia characteristics. This research primarily aims to investigate the effects of simplifications in integrating mass properties and then the impacts of the level of simplification on the performance of the dynamic models. Such simplifications, aimed at reducing computational demands, can impact the model's accuracy. The study seeks a balance between computational efficiency and precision, which is required in robotics, where exact movements and energy efficiency are critical.

Recent literature has focused on innovative methods and considerations for determining mass properties, such as the center of mass and inertia matrices. Woolfrey and Liu explored the use of virtual components in robotic arms, aiming to optimize control by adjusting virtual mass and inertia matrixes [40]. Zhang et al. integrated neural networks for PD control of manipulators, emphasizing gravity and inertia compensation [41]. Habibi et al. developed a dynamic model for soft robotics, considering physical characteristics like gravity and inertia [42]. Cen and Singh addressed the impact of payloads on system mass and inertia in mobile robots [43]. Fu et al. presented a Lie theory-based methodology for dynamic parameter identification in serial manipulators, enhancing the accuracy of inertia tensor and mass property estimations [44]. Le Cleac'h et al. combined differentiable physics with neural networks to simulate object motion and estimate dynamic properties [45]. Xu et al. examined the dynamic coupling in mobile manipulators, focusing on the robotic arm's center of mass [46]. Wüest, Kumar, and Loianno proposed an online estimation method for crucial dynamic properties of aerial vehicles [47]. Fang et al. developed a grasp perception method incorporating the object's center-of-mass awareness [48]. Daniel and Soloniaina's work on robot modeling for reinforcement learning control emphasized the computation of inertia tensors [49]. Hill's research on bipedal robots highlighted the use of arms in disturbance rejection, considering the inertia matrix of each link [50]. This study aims to bridge the gap in understanding the implications of model complexity on robotic arm dynamic models, a subject of growing relevance in the context of increasingly refined industrial automation demands, and indicate a trend toward merging advanced computational methods and control theories in dynamic modeling. However, there still needs to be more in applying these methodologies to more complex robotic systems and varying dynamic environments.

Our findings inform industrial robotic system designs, potentially leading to more efficient and accurate control strategies in diverse manufacturing environments. Future research could focus on overcoming these gaps, employing more efficient intelligent methods for parameter identification of the robot arm's dynamic models based on proper dynamic models, such as quantum-inspired calculations, to enhance the efficiency and accuracy of dynamic modeling in robotics. Based on this, the following sections detail the comparative analysis methodology used to

assess these dynamic models. This approach provides a comprehensive understanding of the trade-offs involved in model simplification and its effects on robotic arm control and efficiency. The primary contributions of this research are two-fold: proposing various detailed levels of the robot's dynamic model and providing a nuanced understanding of how the level of simplification in dynamic models impacts the performance and accuracy of robotic applications.

II. ROBOTIC ARM PRESENTATION

Fig. 1 shows the ABB IRB 140 (M2004) robot considered in this study. This robot's six-axis articulated structure is one of the most widely used in many industrial fields.

According to the ABB technical documentation [51], this industrial robot can be mounted on the floor or a wall at any angle or inverted for various working ranges. It is primarily used for arc welding, assembly, cleaning/spraying, machine tending, material handling, packing, and deburring. The robot weighs about 98 kg with an end effector weighing up to 5 kg, including a payload with a reach of about 810 mm that can be attached to its mounting flange. Up to 1.5 kg of equipment can be mounted on the robot's upper arm. Its joint limits allow ample functional workspace duty, as summarized in Table I [51].



Fig. 1. The ABB IRB 140 robot located in the UQTR automation laboratory

TABLE I. JOINT LIMITS OF THE ABB IRB 140 ROBOT

Joints	Type	Limits (°)
1	R	+180 to -180
2	R	+110 to -90
3	R	+50 to -230
4	R	+200 to -200
5	R	+120 to -120
6	R	+400 to -400

R stands for rotational or revolute.

The robotic manipulator includes an IRC5 controller [52][53], a multi-robot controller with PC tool support that optimises robot performance for short cycle times and precise movements, and RobotWare (Robot Studio), which allows ABB robot programming on a workstation without shutting down production [54][55]. A program can be built on the ABB Virtual Controller, which is an exact copy of the software that runs robots in production. Robot Studio allows highly realistic simulations to be performed using simple robot programs and configuration files identical to those used in real-world applications [55][56].

In this study, the ABB IRB 140 robot was employed in its standard configuration, according to the manufacturer's specifications. No modifications or customizations were made to the robot's structure or system. This standardization ensures that our research results and findings are directly applicable to the typical performance characteristics of the ABB IRB 140 as it is commonly used in industrial settings. It also provides a baseline for comparing the robot's efficiency and dynamic behaviours under standard operating conditions.

III. ROBOTIC ARM MODELS

The systematic analysis of the ABB IRB-140 robotic arm, as conducted at the UQTR Automation Laboratory, began with the input of joint space data to generate the arm's trajectory, underpinning the subsequent development of kinematic models. Using Denavit-Hartenberg parameters, forward kinematics were meticulously formulated to identify the end-effector's positions and orientations. The motions of the robot's joints were then articulated through fifth-order polynomial equations and homogeneous transformation matrices, offering a complete representation of the robot's spatial configurations.

These kinematic constructs initial input as position, velocity, and acceleration to perform the dynamic modelling phase, which employed the Euler-Lagrange formalism to delve into the robot's dynamics, mainly focusing on joint torques and energy consumptions. Such dynamics were further refined by integrating inertia data and mass center positions derived from Solidworks's detailed CAD/CAM modelling process. This foundational work transitioned seamlessly into the computer-aided optimization (CAO) phase, where three different robotic arm models were conceptualized, each varying in complexity from a simplified abstraction to a detailed simulation containing a fidelity spectrum.

The completion of kinematic and dynamic modelling facilitated an integrated analysis, enabling the precise measurement of energy demands based on the torque profiles of each joint. The dynamic models' outputs, which included torque and energy parameters, were instrumental in evaluating the robotic arm's efficiency and the impact of varying model detail levels on system performance.

This comprehensive approach, graphically synthesized in the flowchart Fig. 2, provided a clear flow from the initial data input to the final output analysis. This integrative methodology highlighted the robotic system's efficacy and illuminated the delicate interplay between model detail and the robot's operational efficiency.

A. Kinematics of the Robot

The forward kinematics model aims to determine the position and orientation of the robot's end-effector as a function of joint angle and displacement relative to the base frame or other reference [57]-[61]. To achieve this mathematically, a global coordinate frame must be assigned to the base frame and a local reference frame must be assigned to each joint [62]-[66]. Homogeneous transformation matrices of size 4×4 is then computed for the robot joint axes using a formalism such as D-H (Denavit-Hartenberg) to define and interpret the robot's spatial

geometry and end-effector location within a fixed reference system [67]-[70].

The kinematic function thus maintains a fixed relationship between the two successive joint axes it supports. This relationship can be defined using two parameters: the link length a and link twist α . The link offset d and joint angle θ are used to describe the nature of the connection between adjacent links [28]. Fig. 3 shows the D-H parameter and link assignments for a rotational joint.

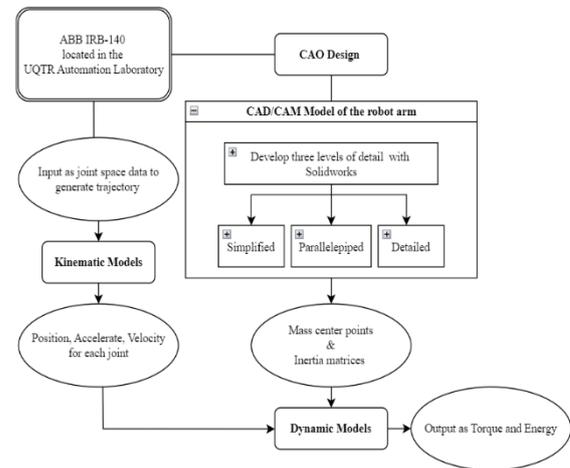


Fig. 2. The flowchart for the methodological steps

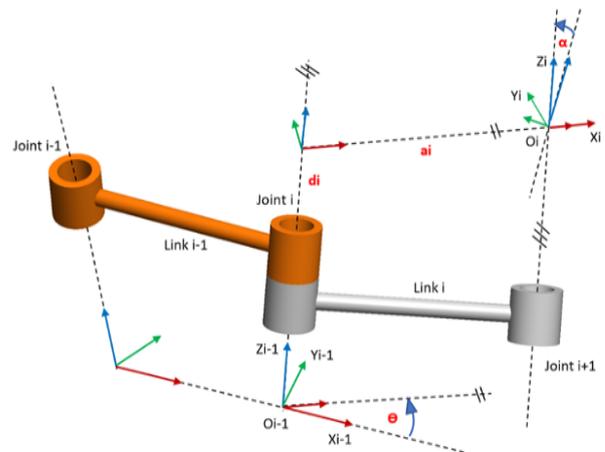


Fig. 3. D-H parameters and link assignments for a rotational joint

The parameters for link i in Figure 3 are defined as follows:

α_{i-1} Twist angle between joint axes z_i and z_{i-1} measured about x_{i-1} .

a_{i-1} Distance between joint axes z_i and z_{i-1} measured along the standard normal.

θ_i Joint angle between joint axes x_i and z_{i-1} measured about z_i .

d_i Link offset between axes x_i and x_{i-1} measured along z_i .

The four transformations between the two axes can thus be defined as follows:

$${}^{i-1}T = \text{Rot}(x_{i-1}, \alpha_{i-1}) \text{Trans}(x_{i-1}, a_{i-1}) \text{Rot}(z_i, \theta_i) \text{Trans}(0, 0, d_i) \quad (1)$$

Where ${}^{i-1}T$ is the homogeneous transformation matrix, $\text{Rot}(x_{i-1}, \alpha_{i-1})$ is the rotation around an axis x_{i-1} by an angle α_{i-1} , $\text{Trans}(x_{i-1}, a_{i-1})$ is the transfer along axis x_{i-1} to the value a_{i-1} ,

Rot (z_i, θ_i) is the rotation around axis z_i by an angle θ_i , and Trans $(0, 0, d_i)$ is the transfer along axis z to the value d .

Therefore, the following homogeneous transformation matrix can be obtained:

$${}^{i-1}T_i = \begin{pmatrix} C\theta_i & -S\theta_i & 0 & a_{i-1} \\ S\theta_i C\alpha_{i-1} & C\theta_i & -S\alpha_{i-1} & -d_i S\alpha_{i-1} \\ S\theta_i S\alpha_{i-1} & C\theta_i S\alpha_{i-1} & C\alpha_{i-1} & d_i C\alpha_{i-1} \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (2)$$

Where C and S denote the cosine and sine of an angle, respectively.

Fig. 4 shows the frame assignments, and Table II lists the D-H parameters of the ABB IRB 140 industrial robot, with the global coordinate system shown below [71]:

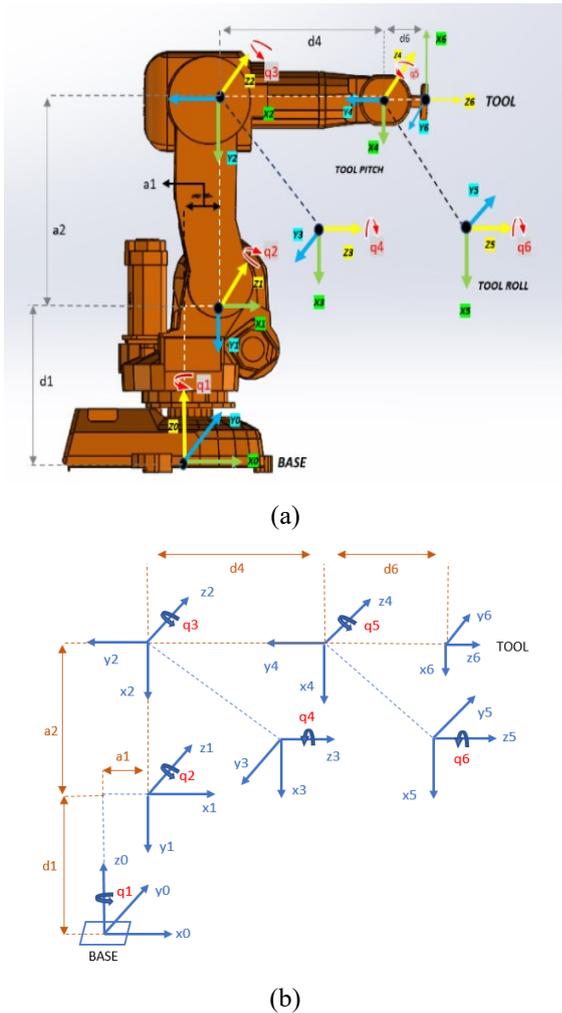


Fig. 4. ABB IRB 140 frame assignments: (a) Frames represented on the real robot, (b) Frames symbolised using DH representation

TABLE II. DENAVIT–HARTENBERG PARAMETERS FOR THE ABB IRB 140 ROBOTIC ARM

Link	a (mm)	α ($^\circ$)	d (mm)	θ ($^\circ$)
1	70	-90	$d_1 = 352$	q_1
2	-360	0	0	$q_2 + 90$
3	0	-90	0	q_3
4	0	90	$d_4 = 380$	q_4
5	0	-90	0	q_5
6	0	0	$d_6 = 65$	q_6

With the D-H parameters, the individual homogeneous transformation matrices for each link can be obtained by substituting the link parameters into Equation (2). The position and orientation are achieved by applying the forward kinematic chain in the global frame, and the pose matrix of the end-effector relative to its base frame is obtained as follows:

$${}^0T_6 = {}^0T_1 \cdot {}^1T_2 \cdot {}^2T_3 \cdot {}^3T_4 \cdot {}^4T_5 \cdot {}^5T_6 = \begin{pmatrix} r_{11} & r_{12} & r_{13} & X \\ r_{21} & r_{22} & r_{23} & Y \\ r_{31} & r_{32} & r_{33} & Z \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (3)$$

Such that:

$$\begin{aligned} r_{11} &= -S_6(S_4 C_1 C_{2'3} + C_4 S_1) - C_6(C_5(S_1 S_4 - C_4 C_1 C_{2'3}) + S_5 C_1 C_{2'3}) \\ r_{12} &= S_6(C_5(S_1 S_4 - C_4 C_1 C_{2'3}) + S_5 C_1 C_{2'3}) - C_6(S_4 C_1 C_{2'3} + C_4 S_1) \\ r_{13} &= C_5 C_1 S_{2'3} - S_5(S_1 S_4 - C_4 C_1 C_{2'3}) \\ r_{21} &= C_6(C_5(C_4 S_1 C_{2'3} + C_1 S_4) - S_5 S_1 S_{2'3}) - S_6(S_4 S_1 C_{2'3} - C_1 C_4) \\ r_{22} &= -S_6(C_5(C_4 S_1 C_{2'3} + C_1 S_4) - S_5 S_1 S_{2'3}) - C_6(S_4 S_1 C_{2'3} - C_1 C_4) \\ r_{23} &= C_5 S_1 S_{2'3} + S_5(C_4 S_1 C_{2'3} + C_1 S_4) \\ r_{31} &= -C_6(S_5 C_{2'3} + C_4 C_5 S_{2'3}) + S_4 S_6 S_{2'3} \\ r_{32} &= C_6 S_4 S_{2'3} + S_6(S_5 C_{2'3} + C_4 C_5 S_{2'3}) \\ r_{33} &= C_5 C_{2'3} - C_4 S_5 S_{2'3} \end{aligned}$$

C_i and S_i denote the cosine and sine of the joint angle q_i , C_{ij} and S_{ij} denote the cosine and sine of $q_i + q_j$. Notation $2'$ refers to $q_2' = q_2 + \pi/2$

The X, Y, and Z position coordinates of the IRB140 robot relative to the base frame are computed as follows:

$$X = C_1 a_1 + d_6(C_5 C_1 S_{2'3} - S_5(S_1 S_4 - C_4 C_1 C_{2'3})) + d_4 C_1 S_{2'3} - C_1 C_2 a_2$$

$$Y = S_1 a_1 + d_6(C_5 S_1 S_{2'3} - S_5(C_4 S_1 C_{2'3} + C_1 S_4)) + d_4 S_1 S_{2'3} - C_2 S_1 a_2$$

$$Z = d_1 + d_4 C_{2'3} + a_2 S_2 + d_6(C_5 C_{2'3} - C_4 S_5 S_{2'3})$$

B. Differential Kinematics of the Robot

Differential kinematics define the relationship between the joints' angular velocities and the corresponding end-effector linear and angular velocities. The study of velocities and static forces yields the Jacobian matrix of the manipulator, which is an essential tool for analysing and controlling robotic motion, identifying singularities and redundancy, determining inverse kinematic equations, and describing the velocity and force manipulability ellipsoids [72]-[74].

The Jacobian matrix, a basis in the kinematic analysis of robotics, is critical for relating joint velocities with the end-effector, a key factor in determining the robot's performance and manoeuvrability. Specifically tailored to the unique joint configurations of each robot, the Jacobian matrix significantly enhances motion planning and control capabilities, which are crucial for executing precision tasks with the robotic arm. It addresses singularities and redundancies, enabling the robot's operational efficiency and safety. Our research employs the Jacobian matrix for smooth and singularity-free motion planning, a necessary factor for ensuring reliable and practical outcomes in our studies. Utilizing MATLAB simulations, we have demonstrated the

Jacobian's considerable impact in practical settings, highlighting notable improvements in accuracy and speed.

A Jacobian matrix is a multidimensional form of the derivative. It can be of any dimension (including non-square), depending on the number of joints. The number of rows in a Jacobian matrix is equal to the number of degrees of freedom in Cartesian space; for example, three rows would be present if the robot's position only was considered, whereas there would be six if both its position and orientation were considered. The number of columns in the matrix corresponds to the number of joints comprising the manipulator.

Considering the end-effector linear velocity vector \dot{p}_e , the angular velocity vector ω_e , and the joint velocity vector \dot{q} , J_p is the $(3 \times n)$ matrix that links the linear velocity vector to the joint speed vector. J_o is the $(3 \times n)$ matrix that links the angular velocity vector to the joint speed vector as expressed in Equations (4) and (5) or in the compact form expressed in Equation (6) [24][75].

$$\dot{P}_e = J_p(q)\dot{q} \quad (4)$$

$$\omega_e = J_o(q)\dot{q} \quad (5)$$

$$V_e = \begin{pmatrix} \dot{P}_e \\ \omega_e \end{pmatrix} = J(q)\dot{q} \quad (6)$$

where V_e is the end-effector velocity and $J = \begin{pmatrix} J_p \\ J_o \end{pmatrix}$ is the Jacobian matrix.

If the relationship between the joint space variable and the orientation space variable is highly nonlinear, the inverse kinematics solution will be redundant and closed form or even non-existent. The inverse kinematics problem initially involves a linear mapping of the joint velocity space and the operational velocity space using differential equations. Depending on the desired end effector position and orientation, the corresponding joint velocity can be obtained via simple inversion of the Jacobian matrix, which therefore must be invertible, i.e., J is square and its determinant is non-zero. The inverse differential kinematics model can then be computed using Equation (7).

$$\dot{q} = J^{-1}V_e \quad (7)$$

Where J^{-1} is the inverse of the Jacobian matrix.

As noted above, the Jacobian matrix can be of any dimension, and is not always square or invertible; in this instance, the pseudo-inverse form of the generalized inversion can be used [24].

C. Dynamic Model

A dynamic model allows the expression of the robot's function in terms of joint acceleration forces and torque. The most widely used method to determine this model is the Euler-Lagrange approach. The model of an n -jointed robotic arm can be expressed by Equation (8) [24].

$$M \cdot \ddot{q} + V + G = \tau \quad (8)$$

Where \ddot{q} is the joint acceleration vector, M is the inertia matrix, V is the Coriolis vector, G is the gravitational vector,

and τ is the force and torque vector. The inertial data and mass centre position were obtained using SolidWorks 3D CAD modelling software for the detailed, semi-detailed (rectangular), and simplified models in this study. Parameters such as the geometry and density of each link (assumed to be uniform in all models) are considered, in addition to significant values related to mass and other physical properties such as the mass centre position and inertia matrices, which are represented in their respective reference frames for the three proximal links. A dynamic model's precision relies on accurately estimating the mass center and inertia matrices. Straightforward dimensional specifications and scientific techniques such as CAD modeling, Experimental Modal Analysis, and Inertia Measurement Units are employed to obtain this required data in engineering. CAD modelling presents detailed dimensional and mass distribution data, while Experimental Modal Analysis and Inertia Measurement Units directly measure inertia characteristics. This research primarily aims to examine the influences of simplifications in obtaining mass properties and then the effects of the level of simplification on the performance of the dynamic models. Such simplifications, desired to reduce computational demands, can affect the model's precision. The study seeks a balance between computational efficiency and precision, which is required in robotics, where exact movements and energy efficiency are critical.

Assuming a constant and homogeneous density, the comparative mass of each link can be estimated. In this study, the volume of each link was determined using SolidWorks software. The relationship of an element's volume to the robot's total volume is multiplied by the total mass to give an estimated link mass value.

The following matrices I_i correspond to the inertia tensors of the proximal links over their mass centres and are expressed relative to the base reference frame [24].

$$I_1 = \begin{bmatrix} 0 & R \\ 1 & R \end{bmatrix} {}^1I_1 + \begin{bmatrix} 0 & R \\ 1 & R \end{bmatrix}^T \quad (9)$$

$$I_2 = \begin{bmatrix} 0 & R_2 & R \\ 1 & R_2 & R \end{bmatrix} {}^2I_2 + \begin{bmatrix} 0 & R_2 & R \\ 1 & R_2 & R \end{bmatrix}^T \quad (10)$$

$$I_3 = \begin{bmatrix} 0 & R_2 & R_3 & R \\ 1 & R_2 & R_3 & R \end{bmatrix} {}^3I_3 + \begin{bmatrix} 0 & R_2 & R_3 & R \\ 1 & R_2 & R_3 & R \end{bmatrix}^T \quad (11)$$

1I_1 , 2I_2 and 3I_3 are the inertia matrices.

In the dynamic analysis of this manipulator, joint friction was not considered. For the tracking of a path, the vector of generalized forces was considered as shown below:

$$\sum_{i=1}^n M_{ij}\ddot{q}_j + V_i + G_i = \tau_i, \quad i=1,2,\dots, \quad (12)$$

M_{ij} represents the inertial forces, V_i represents the centrifugal and Coriolis forces, and G_i indicates the gravitational forces. This equation can also be used to determine the joints' acceleration values, as follows:

$$\ddot{q} = M^{-1}(Q - V - G) \quad (13)$$

$$M = \sum_{i=1}^n (J_{vi}^T m_i J_{vi} + J_{oi}^T I_i J_{oi}) \quad (14)$$

$$V_i = \sum_{j=1}^n \sum_{k=1}^n \left(\frac{\partial M_{ij}}{\partial q_k} - \frac{1}{2} \frac{\partial M_{jk}}{\partial q_i} \right) \dot{q}_j \dot{q}_k \quad (15)$$

$$G_i = -\sum_{j=1}^n m_j g^T J_{vj}^T \quad (16)$$

The inertia matrix of the manipulator is symmetric and positive definite and therefore always invertible [24].

D. CAD Design of the Robot

The decision to create three different ABB IRB 140 arm industrial robot models in SolidWorks is based on the need for a comprehensive understanding of kinematic and dynamic behaviour. The accurate, detailed model in Fig. 5 is essential for simulations that require high fidelity and detailed analysis. This model duplicates the robot's structure with maximum precision, making it ideal for investigating complex dynamics and interactions within the actual robot's mechanism. In contrast, the less detailed rectangular model, such as the parallelepiped model in Fig. 6, balances detail and computational efficiency. Its simplified geometric representation suits quicker simulations where extreme detail is not required. This model serves as a midpoint, validating results from the detailed model and offering a more efficient alternative for less demanding simulations. Lastly, the simplified model in Fig. 7 reduces the robot to its fundamental components. This model benefits high-level analyses and educational purposes, focusing on understanding the robot's basic mechanics rather than its detailed construction. Simplification might not accurately represent the varying density and weight distribution in different parts of the robot, potentially affecting the precision of dynamic simulations, such as calculating torque and evaluating the energy.

During the CAD design process, a fundamental assumption is the uniform and constant density across all parts and links of the robot. While this simplifies the computational characteristic, it is a potential source of inaccuracy, as it needs to account for the material variations in the actual robot. Another assumption involves the mass and inertia characteristics provided by SolidWorks, based on the model's geometry and assumed material properties. These estimations are required for dynamic analysis but may not perfectly match the real-world robot due to factors like manufacturing tolerances and material inconsistencies.

The assumptions made during the CAD design process are practical but can significantly influence the accuracy of the models. While the uniform density assumption simplifies the modelling, it might lead to simulation discrepancies, particularly in dynamic behaviour. Likewise, although functional for initial analysis, the derived inertia characteristics require improvement for more precise simulations. The details of each model and the results for the proximal links obtained from the Solidworks mass properties tool after modelling are summarised in Table III.

E. Robot Performance Assessment

To evaluate the accuracy of the three models' predictions, we analysed the energy consumption by the three proximal joints and the robot's total energy consumption over the same duration and path of movement. In the proposed modelling approach, the energy consumption of each joint at a specific time can be calculated from the joint torque and angular velocity using Equation (17). The robot's integrated energy

consumption is assumed to be the sum of the energy consumption values of all the joints.

$$E_i = \int_{t_0}^{t_f} \tau_i(t) \cdot \dot{q}_i(t) dt \quad (17)$$

We employ a theoretical approach to evaluate the energy consumption of robotic arms. This approach centers on dynamic models that simulate each joint's torque τ_i and angular velocity \dot{q}_i . Detailed explanations of how these parameters are simulated are provided in the methodology section. The simulated values are then used in Equation (17) to calculate the energy consumption of each joint. Theoretical method offers a comprehensive analysis of energy consumption, avoiding the practical challenges and complexities associated with the installation and calibration of physical sensors on the robotic arm.

Focusing on a theoretical and simulation-based approach contributes significantly to understanding energy dynamics in robotic arms, especially in scenarios where direct measurement is impractical or impossible. This approach also aligns with current trends in employing computational models to analyze complex system contributions to robotics fields.

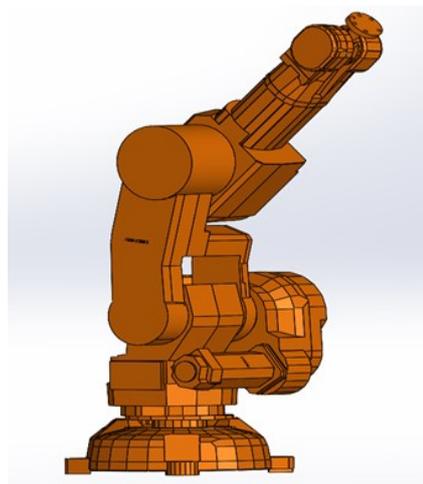


Fig. 5. Detailed SolidWorks model of the ABB IRB 140 robotic arm

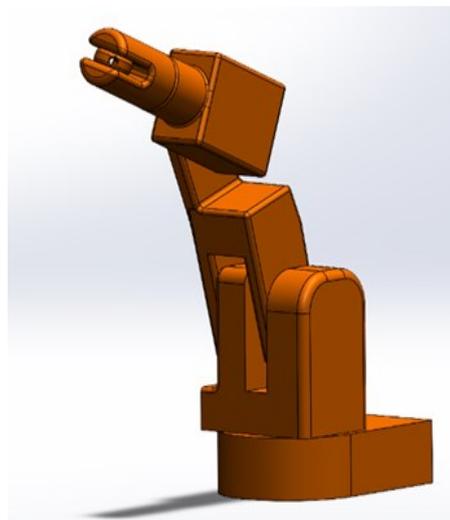


Fig. 6. Semi-detailed rectangular SolidWorks model of the ABB IRB 140 robotic arm



Fig. 7. Simplified SolidWorks model of the ABB IRB 140 robotic arm

TABLE III. MASS PROPERTY RESULTS OF EACH MODEL CALCULATED USING SOLIDWORKS SOFTWARE

Link	parameter(unit)	Detailed model	Semi-detailed model	Simplified model
1	Weight (kg)	35	35	35
	Xc (mm)	277.87	17.87	0
	Yc	373.12	103.12	181.37
	Zc	-199.03	-79.03	0
	Ixx (kg.m2)	6.5	1.5	1.1
	Ixy	1.1	0.1	0
	Ixz	3.05	0.05	0
	Iyy	2.02	0.002	0
	Iyz	5.07	0.07	0.1e-5
	Izz	1.4	0.04	1.1
2	Weight (kg)	25	25	25
	Xc (mm)	218.29	178.29	0
	Yc	229.73	9.73	255.24
	Zc	112.43	72.43	0
	Ixx (kg.m2)	0.9	0.1	1.6
	Ixy	-0.03	-0.03	0.1e-5
	Ixz	0.1	0.01	0
	Iyy	1.3	0.03	0
	Iyz	-0.01	-0.01	0
	Izz	0.95	0.05	1.6
3	Weight (kg)	25	25	25
	Weight (kg)	18	18	18
	Xc (mm)	-24.56	14.56	0
	Yc	-219.9	-199.96	195.69
	Zc	-25.86	-15.86	0
	Ixx(kg.m2)	2.5	1.5	0.6
	Ixy	-0.001	-0.001	0.3e-5
	Ixz	0.09	0.09	0
	Iyy	2.7	0.7	0
	Iyz	-0.8	-0.02	0
Izz	0.5	0.2	0.6	
Weight (kg)	18	18	18	

IV. RESULTS AND DISCUSSION

The dynamic performance of an ABB IRB-140 six-jointed industrial robotic arm was considered in the present research, with a principle on the effect of model simplification on torque prediction and energy consumption. Our study extends and analyses three varying models of the robotic arm, simplified, semi-detailed, and detailed, using precise kinematics models, differential kinematics, and an advanced dynamic model based on the arm's technical specifications. The design of three levels of CAO design, facilitated by Solidworks, aimed to obtain the center of mass

and inertia matrices, which are crucial for understanding the dynamic behaviour of the manipulator. Through this analysis, we aim to unravel the interplay between model complexity and its importance on computational efficiency and accuracy, eventually providing a comprehensive understanding of model selection in industrial robotics. This research, therefore, not only investigates the performance of these models in a standard functional context but also seeks to inform the optimization of robotic systems, balancing precision, energy consumption, and computational demands.

Based on the mass and inertia characteristics of the robot and the calculated end-effector position, orientation, velocity, and acceleration and the torque of each joint in the three proximal links (i.e., the principal determinants of the end-effector position), the dynamic kinematic model of the robot developed in SolidWorks was examined in MATLAB in full detail over a 30-second interval without a payload at the effector end of the robotic arm. The forward and robot kinematic models were derived from the D-H parameters, and a procedure was developed to solve the inverse kinematics. Trajectory planning was based on a fifth-order polynomial. The values in Table I were used to cover all possible angles of each of the robot's rotating joints. Fig. 8 displays the changes in angle over time for the three proximal links of the ABB IRB-140 robotic arm, highlighting the dynamic positioning of these joints during the task operation. The corresponding velocity and acceleration diagrams are shown in Fig. 9 and Fig. 10. Fig. 9 shows the velocity variations of the robotic arm's proximal joints, highlighting the speed dynamics during the arm's movements. Fig. 10 depicts the acceleration patterns of the proximal joints, capturing the robotic arm's dynamic response during its operating path.

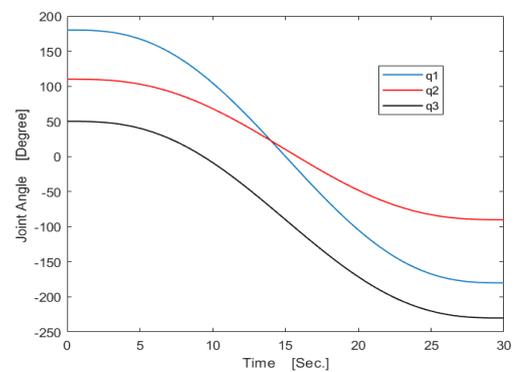


Fig. 8. IRB 140 robot proximal joint angle variations

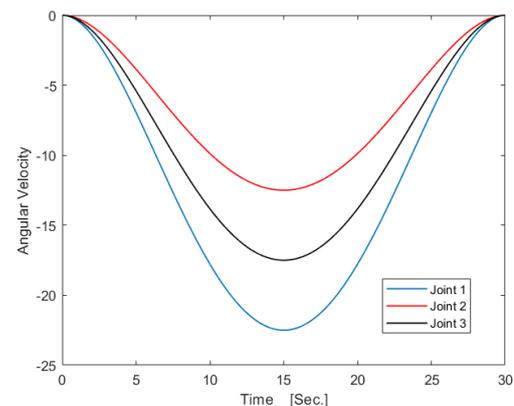


Fig. 9. IRB 140 robot proximal joint angle variations

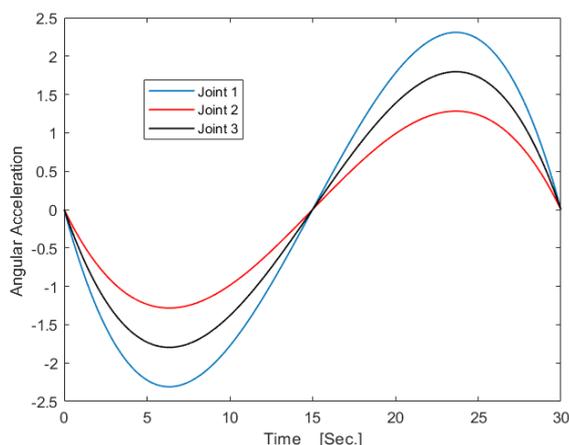


Fig. 10. IRB 140 robot proximal joint angle variations

The torque variations for the position and path corresponding to a single, uninterrupted time interval are shown in Fig. 11 to Fig. 13. These figures show the torque variations for the robotic arm's first three proximal joints, presenting insights into the mechanical loads experienced during its operation.

As shown, the results obtained for all three joints are consistent and appear relatively close for the semi-detailed and simplified models. The detailed model tends to exhibit higher torque at most time points in joint 1, for a short period in joint 2, and throughout almost all of the modelled period in joint 3. Notably, the semi-detailed and simplified models both track the detailed model quite closely for at least some of the time; however, this behaviour varies between different joints. The results also confirm that joint 2 experienced much more torque than the other joints, which likely occurs because this joint provides most of the reaching capability of the robotic arm.

In addition to its weight and gravitational force, the link also bears a share of the weight and gravitational force of the third link. As the robot reaches further from its centre of gravity, the mass of the remaining links creates additional

torque. The observed bends in the torque curves result from changes in the direction of the arm movement based on the defined task path. In addition, these changes in direction in the defined path cause the torque values of all three joints be equal at various points during the simulation.

Due to the choice of a fifth-order polynomial trajectory, the joint torque curves for all three models exhibit little or no noise. The torque starts at zero for joint 1, rises quickly, and then drops smoothly to zero. For joints 2 and 3, the torque starts at non-zero values, drops, rises to new values, and then ends at different values at the end of the path. These profiles arise due to the positions and movements against gravity encountered by the arm over its defined trajectory.

In the case of joints 1 and 2, the semi-detailed and simplified models show higher torque than the detailed model during some periods, whereas the opposite is generally observed for joint 3. In addition, the simplified model exhibits slightly less torque than the semi-detailed model at the beginning and end of the simulation and lags the semi-detailed dynamics throughout the path. Since the link weight and density are identical in all the models, these differences in torque must therefore reflect differences in the centre of mass and inertia tensors. Differences in the torque at joints 1 and 2 likely arise due to the weight and torque imposed by other links. The gravitational force of the other links affects the first and second joints, and only the third joint carries its gravitational force without much additional torque.

Based on the frame assignments defined for the robotic arm, link 1 does not exert a gravitational force and undergoes joint acceleration only; thus, the torque is at its lowest value in this link. The second and third links exert gravitational force and are subject to angular acceleration in the joints. The combination of these two acceleration vectors defines the overall acceleration of these two links.

When the joint acceleration falls to zero, most of the torque is expected to be gravitational, i.e., an opposing force resisting the movement of the robotic arm.

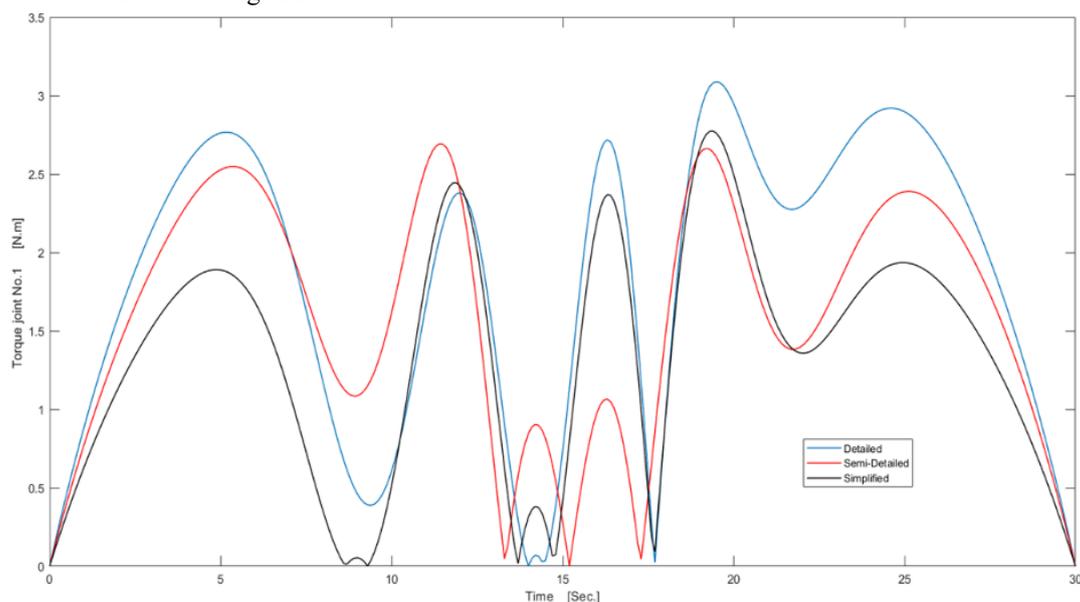


Fig. 11. Torque variations for the first proximal joint of the IRB 140 robotic arm

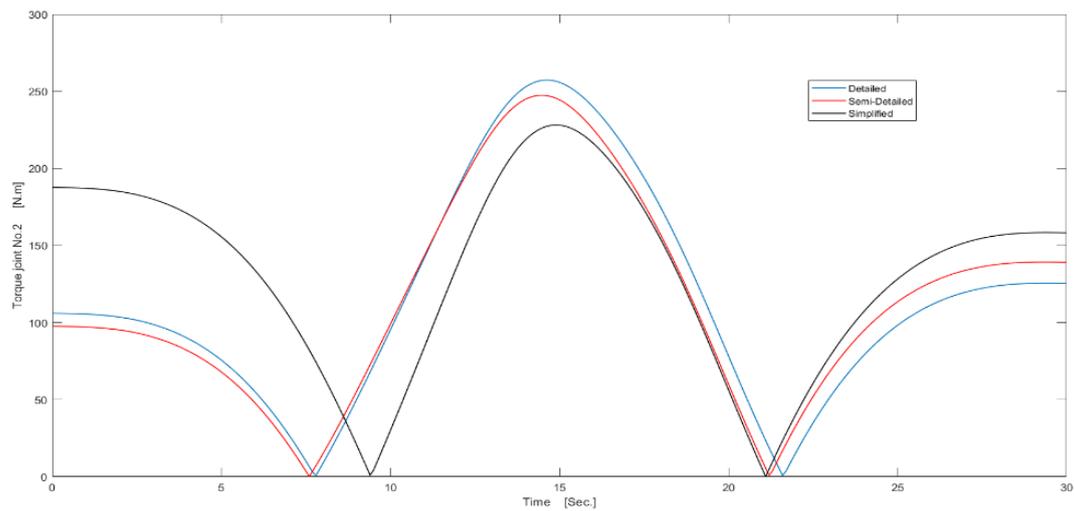


Fig. 12. Torque variations for the second proximal joint of the IRB 140 robotic arm

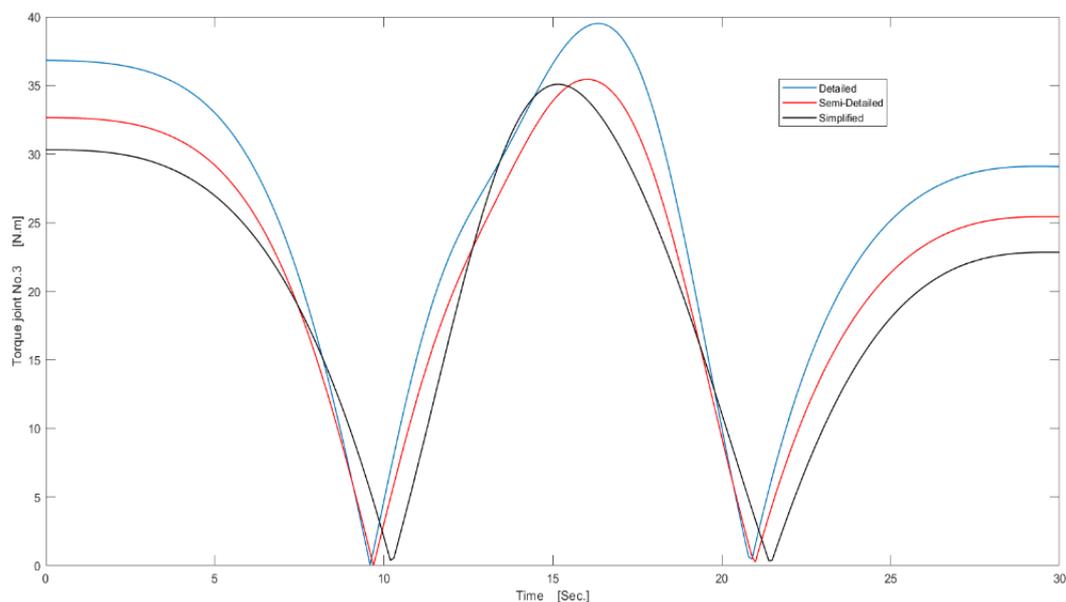


Fig. 13. Torque variations for the third proximal joint of the IRB 140 robotic arm

One of the main aims of this study is to examine how well the different dynamic models predict the energy consumption of each rotational joint and the robot overall, as determined by Equation (17). These results are shown in Fig. 14 (joints) and Fig. 15 (robot). Fig. 14 compares the energy consumption of each joint in the robotic arm across the three model variations, highlighting the differences in energy efficiency. Fig. 15 illustrates the total energy consumption of the robotic arm over a standard 30-second task, emphasizing each model's energy efficiency.

The energy consumption patterns are broadly similar in form in all three joints and models. Joint 2, which has the highest torque, consumes the most energy. During the 30-second simulation in this study, the energy consumption of the joints and the whole robotic arm increased, with the most rapid energy consumption increase recorded in the middle of the simulation. The energy consumption increases reflect changes in gravitational forces on the links as they move away from their initial centres of mass. Joint 2 and the arm show strikingly similar energy consumption patterns in all

three models, and the semi-detailed and simplified models for joint 2 track each other closely. For the whole arm, the detailed and semi-detailed models are very similar and thus may reflect the influence of each model's mass properties and shape characteristics. The semi-detailed model yields the highest energy consumption in both cases, whereas the simplified model gives the lowest values, except for joint 1. Relative to the semi-detailed model, the overall energy consumption difference is 0.53% less in the detailed model and 6.8% less in the simplified model. Based on these results, the three dynamic models tested in this study are all worthy of confidence for predicting energy consumption by articulated robotic arms.

It is necessary to verify the underlying physical and theoretical reasons for response to the observed torque and energy consumption variations across the detailed, semi-detailed, and simplified. The intricacy of each model plays a key role in its predictive accuracy. The detailed model, capturing more complex dynamics, including acceptable joint interactions and precise physical properties, delivers more

accurate torque predictions and energy consumption profiles. In contrast, the semi-detailed and simplified models, with inevitable oversimplifications in dynamics and geometry, may need to capture these intricate details more effectively, leading to noticeable discrepancies. Specifically, the variation in the center of mass and inertia tensors across models significantly impacts torque requirements. The detailed model's precise calculations offer a closer representation of real-world dynamics, whereas the simplified models, with their approximations, might either underestimate or overestimate these forces.

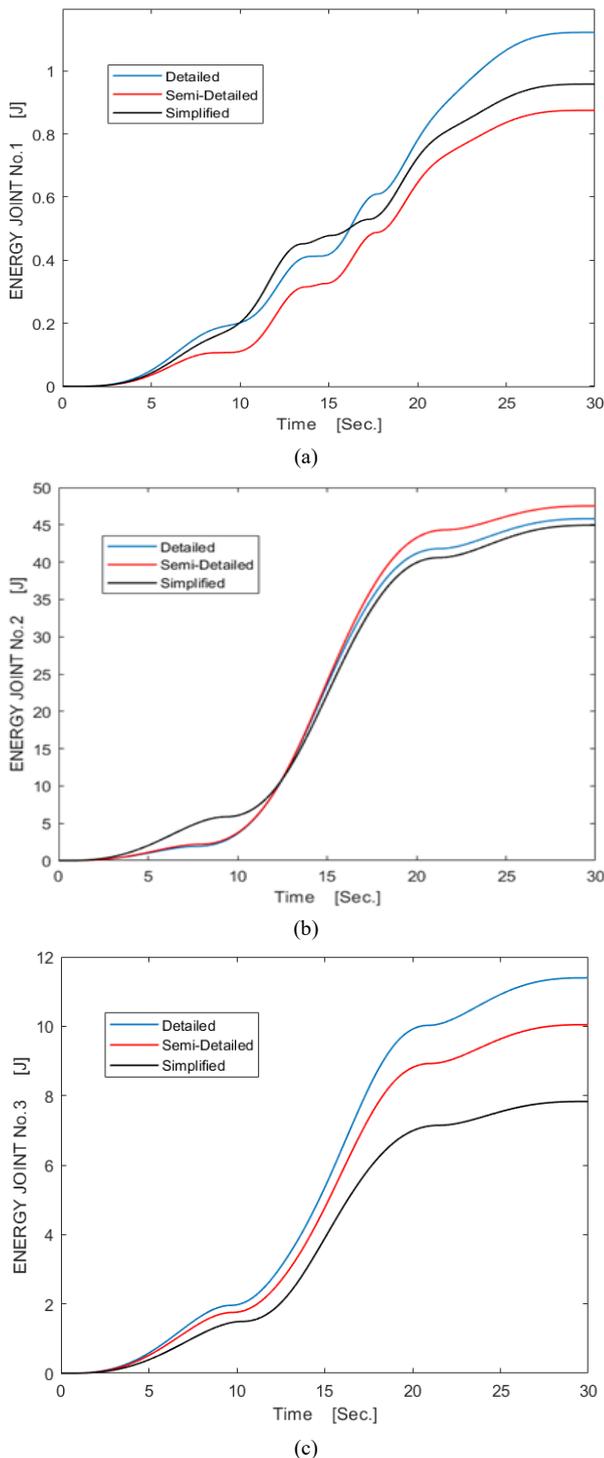


Fig. 14. Energy variations for the third proximal joint of the IRB 140 robotic arm

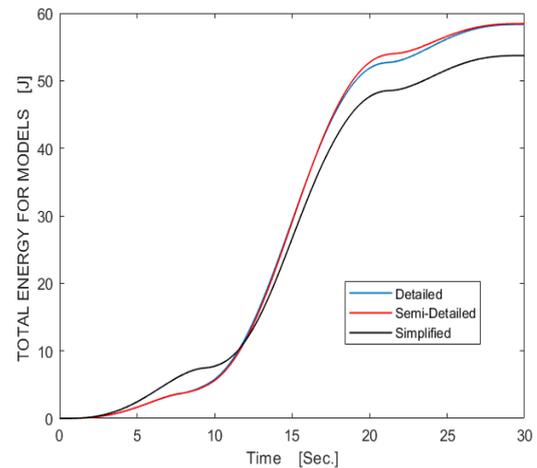


Fig. 15. Energy variations for the third proximal joint of the IRB 140 robotic arm

Moreover, the detailed model more accurately shows the influence of link weight and gravitational forces. This factor becomes especially essential in joints primarily responsible for the arm's movement, as their torque requirements vary substantially with the arm's position relative to its center of gravity. Finally, a comparative analysis of energy consumption highlights how each model's complexity influences its energy efficiency. The detailed model provides a fine sense of energy dynamics, particularly in joints with higher energy demands. These insights are invaluable for optimizing robotic systems, where accuracy, computational efficiency, and energy conservation are paramount. Therefore, understanding these differences in model complexity and their impact on torque and energy consumption is crucial for informed decision-making in the design and optimization of industrial robotic systems.

In comparing our findings with related literature, we note several parallels and distinctions in the dynamics of robotic arm manipulators. Earlier studies, such as those by Boiadjiev et al. [76] and Ding et al. [77], have highlighted the importance of model complexity in predicting torque and energy consumption in robotic arms. Our research develops these findings by providing an analysis of how different model simplification levels (simplified, semi-detailed, and detailed) impact these predictions. Our research aligns with Mahdavian et al. [78], meaning that more complex models predict higher torque requirements, particularly in joints with more intricate movement patterns. However, our study uniquely contributes by quantifying the energy consumption across these models, a factor that needs to be explored in previous research. The work by Fang et al. [79] on energy-efficient hydraulic joints for mobile robotic manipulators and the operational dynamics of the OceanWATERS lander arm by Catanoso et al. [80] provide complementary perspectives to our findings on energy efficiency. This comparative analysis reaffirms the general directions documented in previous studies. It adds new dimensions to performance model complexity in industrial robotic arms, particularly in energy efficiency and dynamic performance. As our study highlights, balancing computational efficiency and accuracy in model selection is critical for optimizing performance based on the required task precision and available computational resources. Our findings provide a practical

attachment to previous work, confirming comprehended trends and offering new insights into the industrial robotic arm dynamics in the present prospect, where energy efficiency is increasingly prioritized.

An existing knowledge gap is a balance between model accuracy and computational efficiency in industrial robotics by comparing various model complexities, enhancing the performance of robotic simulations. This research has significant practical applications, especially in optimizing energy use and robust design in industrial settings, potentially reducing costs and enhancing sustainability. We assumed uniform density across each link and ignored joint friction. While these assumptions simplify modeling, they overlook material distribution influences and friction's impact on energy and torque. Therefore, our findings might only partially reflect real-world complexities. Addressing these aspects in future research could enhance the model's accuracy and applicability in practical scenarios. Future research directions include exploring more precise or proving outcomes of the accurately designed models with advanced optimization algorithms like quantum-inspired particle swarms. Our findings highlight the significance of proper modelling in robotics, providing valuable guidance for professionals in optimizing performance and efficiency in industrial robotic systems.

V. CONCLUSION

In most industrial robotics studies, only one dynamic model is typically used, with comparisons between models being rare. In this study, we developed three models for the ABB IRB 140 industrial robotic manipulator arm, which has six degrees of freedom. These models, developed in SolidWorks, are detailed, semi-detailed, and simplified. We characterized the end-effector position and orientation by presenting kinematic models linked to a proposed elaborate dynamic model using mass and inertial data obtained from Solidworks's mass properties tool to calculate joint torque for the three proximal links. These links are crucial for the accuracy of the robotic arm's movements. We used the Spong formulation for forward and inverse differential kinematics [24]. The mathematical models of the robot were analyzed in detail in MATLAB. We derived the robot's forward kinematics from D-H parameters and executed a trajectory planning based on a fifth-order polynomial.

Our key finding is that the links' dimensions and geometry significantly influence the mass center position and inertial matrix characteristics of the robot links, which impacts straightforward torque equations. Selecting the correct position for the center of mass and inertia tensor elements in dynamic models is essential. Interestingly, the simplified model, often used in classical computational methods, yields acceptable results. The accuracy of these results in reflecting the actual robot's behavior needs further validation. It is important to note that any model-derived data for mass properties will have an error percentage that must be considered in dynamic equations.

MATLAB helps implement algorithms, estimate and validate models, and simulate system responses. MATLAB is required for its robust algorithmic capabilities, efficient data processing, and simulation tools in computational

robotics. It excels in implementing complicated mathematical models, analyzing kinematic and dynamic data, and visualizing robotic behaviors, making it a valuable tool for research and development in this field. Computational time and efficiency for solving algebraic equations matter in robotics. A comparative study for three different models of the case study in the same operation system (OS) and hardware configuration indicates that the time for analyzing the dynamic model to calculate torque equation in the same trajectory for a detailed model was 83.2 percent, and semi-detailed was 47.63 percent more than the simplified model. Another important finding is that all three robot models show similar total energy consumption over a movement path in a given time, which indicates that each model could be effective in various scenarios. In robotics, some calculations are complex, involving multiple points. Therefore, a simplified model version can enhance and optimize the calculation process and make significant contributions to execution time and efficiency, addressing key industrial robotics challenges like improving energy efficiency and sustainability in continuous operations, optimizing production processes for higher throughput in manufacturing while offering flexibility in robot deployment across various applications. These insights promise considerable improvements in diverse manufacturing environments. Industries can choose models based on specific needs: precision for aerospace, speed for manufacturing, and customization for unique applications, balancing robustness and computational resources.

The developed model facilitates by assuming homogeneous density across each link and disregarding joint friction. This approach might only partially capture the real world, such as the effects of varied material distribution and friction on energy and torque. Future research addressing these factors could improve model precision and real-world relevance. Admitting these limitations, such as assuming uniform density across each robotic link and neglecting joint friction, can also lead to advanced control strategies, enhancing the accuracy and effectiveness of robotic arms in practical, complex environments. Introducing a control module and a robustness controller could further enhance the precision of the models, minimizing the impact of input errors.

Future research could focus on refining these aspects, utilizing advanced optimization algorithms like quantum-inspired particle swarms to identify more accurate and practical parameters for each model. Additionally, there is scope for assessing the torque requirements for various robot positions and operations and comparing energy consumption during different tasks. Quantum computing carries notable potential for enhancing the modeling of complicated systems like the dynamic models of robotic arms [71][81]-[83]. Its ability to process extensive data at exceptional speeds makes it ideal for tackling the intricate calculations involved in robotics [84]. By utilizing quantum-inspired algorithms, such as particle swarm optimization, researchers can have more accurate and efficient modeling, overcoming the limitations of classical computing methods. This approach could lead to breakthroughs in comprehending the behaviors of robotic systems, enabling more precise control and optimization. The

application of quantum computing in this domain promises to revolutionize how dynamic models are developed and applied, potentially leading to more sophisticated and capable robotic systems in various industrial applications [85].

A key takeaway from our study is the importance of detail in modeling so that detailed models offer higher accuracy, and simplified models significantly reduce computational time and enhance efficiency, which is critical for professionals in the field as it guides the selection of appropriate models based on the specific requirements of precision and efficiency in industrial robotic systems.

ACKNOWLEDGMENT

This research was funded by the PAIR fund from UQTR.

REFERENCES

- [1] M. Suomalainen, Y. Karayiannidis, and V. Kyrki, "A survey of robot manipulation in contact," *Robotics and Autonomous Systems*, vol. 156, p. 104224, Oct. 2022, doi: 10.1016/j.robot.2022.104224.
- [2] M. Schwenzer, M. Ay, T. Bergs, and D. Abel, "Review on model predictive control: an engineering perspective," *The International Journal of Advanced Manufacturing Technology*, vol. 117, no. 5–6, pp. 1327–1349, Aug. 2021, doi: 10.1007/s00170-021-07682-3.
- [3] G.-H. Yang and J. Yao, "Nonlinear adaptive output feedback robust control of hydraulic actuators with largely unknown modeling uncertainties," *Applied Mathematical Modelling*, vol. 79, pp. 824–842, Mar. 2020, doi: 10.1016/j.apm.2019.10.062.
- [4] D. Xuan and J. Bae, "A Precise Neural-Disturbance Learning Controller of Constrained Robotic Manipulators," *IEEE Access*, vol. 9, pp. 50381–50390, 2021, doi: 10.1109/ACCESS.2021.3085907.
- [5] M. Goubej, J. Königsmarková, R. Kampinga, J. Nieuwenkamp, and S. Paquay, "Employing Finite Element Analysis and Robust Control Concepts in Mechatronic System Design-Flexible Manipulator Case Study," *Applied Sciences*, vol. 11, no. 8, p. 3869, 2021, doi: 10.3390/app11083869.
- [6] C. Xiao and Y. Jianyong, "Robust Adaptive Control for a class of uncertain nonlinear systems with an optimized smooth input," in *2021 33rd Chinese Control and Decision Conference (CCDC), IEEE*, 2021, doi: 10.1109/CCDC52312.2021.9607054.
- [7] P. Kremer, J. L. Sanchez-Lopez, and H. Voos, "A Hybrid Modelling Approach for Aerial Manipulators," *Journal of Intelligent & Robotic Systems*, vol. 105, no. 4, p. 74, 2022, doi: 10.1007/s10846-022-01675-y.
- [8] T. Yaren and S. Kücüük, "Dynamic modeling of 3-DOF RRP type serial robotic manipulator using Lagrange-Euler method," in *International Marmara Sciences Congress (Autumn)*, pp. 372–378, 2019.
- [9] M. J. Thomas, M. L. Joy, and A. P. Sudheer, "Kinematic and dynamic analysis of a 3-PR US spatial parallel manipulator," *Chinese Journal of Mechanical Engineering*, vol. 33, pp. 1–17, 2020, doi: 10.1186/s10033-020-00468-9.
- [10] Y. Chen *et al.*, "Dynamic modeling and experimental validation of a water hydraulic soft manipulator based on an improved newton—Euler iterative method," *Micromachines*, vol. 13, no. 1, p. 130, 2022, doi: 10.3390/mi13010130.
- [11] L. Yu, "An Effective Compensation Strategy for Dynamic Model based on Improved Kane Principal Formulation," *Journal of The Institution of Engineers (India): Series C*, vol. 103, no. 4, pp. 589–596, 2022, doi: 10.1007/s40032-021-00670-4.
- [12] Y. Park, "An Automatic Program of Generation of Equation of Motion and Dynamic Analysis for Multi-body Mechanical System using GNU Octave," *Journal of Applied and Computational Mechanics*, vol. 7, no. 3, pp. 1687–1697, 2021, doi: 10.22055/JACM.2021.37445.3027.
- [13] M. Trojanová and T. Cakurda, "Validation of the dynamic model of the planar robotic arm with using gravity test," *MM Science Journal*, vol. 5, pp. 4210–4215, 2020.
- [14] C. Zhuang, Y. Yao, Y. Shen, and Z. Xiong, "A convolution neural network based semi-parametric dynamic model for industrial robot," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 236, no. 7, pp. 3683–3700, 2022, doi: 10.1177/09544062211060249.
- [15] J. Deng, W. Shang, B. Zhang, S. Zhen, and S. Cong, "Dynamic Model Identification of Collaborative Robots Using a Three-Loop Iterative Method," in *2021 6th IEEE International Conference on Advanced Robotics and Mechatronics (ICARM)*, pp. 937–942, 2021, doi: 10.1109/ICARM51412.2021.9512786.
- [16] Q. Leboutet, J. Roux, A. Janot, J. R. Guadarrama-Olvera, and G. Cheng, "Inertial Parameter Identification in Robotics: A Survey," *Applied Sciences*, vol. 11, no. 9, pp. 4303, 2021, doi: 10.3390/app11094303.
- [17] Y. Han, J. Wu, C. Liu, and Z. Xiong, "An iterative approach for accurate dynamic model identification of industrial robots," *IEEE Transactions on Robotics*, vol. 36, no. 5, pp. 1577–1594, 2020, doi: 10.1109/TRO.2020.2992386.
- [18] M. Bottin, S. Cocuzza, N. Comand, and A. Doria, "Modeling and identification of an industrial robot with a selective modal approach," *Applied Sciences*, vol. 10, no. 13, p. 4619, 2020, doi: 10.3390/app10134619.
- [19] S. A. Zimmermann, T. F. Berninger, J. Derkx, and D. J. Rixen, "Dynamic modeling of robotic manipulators for accuracy evaluation," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 814–850, 2020.
- [20] J. G. Romero, R. Ortega, and A. Bobtsov, "Parameter estimation and adaptive control of Euler–Lagrange systems using the power balance equation parameterisation," *International Journal of Control*, vol. 96, no. 2, pp. 475–487, 2021, doi: 10.1080/00207179.2019.1661354.
- [21] A. J. Muñoz-Vázquez, V. Parra-Vega, A. Sánchez-Orta, and J. D. Sánchez-Torres, "High-gain PI-like control of Euler–Lagrange mechanical systems: Simulations and experiments in 2DoF robotic manipulators," *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 236, no. 4, pp. 857–869, 2022, doi: 10.1177/09596518211032626.
- [22] X. He, M. Lu, and F. Deng, "Trajectory Tracking Control of Uncertain Euler-Lagrange Systems: A Robust Control Approach," in *2021 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 1855–1860, Dec. 2021, doi: 10.1109/ROBIO54115.2021.9713830.
- [23] L. Kong, S. Zhang, and X. Yu, "Approximate optimal control for an uncertain robot based on adaptive dynamic programming," *Neurocomputing*, vol. 423, pp. 308–317, 2021, doi: 10.1016/j.neucom.2020.10.112.
- [24] M. Fazilat. *Advanced modelling and control of industrial robotic manipulator arm*. Diss. Université du Québec à Trois-Rivières, 2022.
- [25] Z. Yang. *Modeling, simulation and control of microrobots for the microfactory*. Electronic Theses and Dissertations, 2023.
- [26] Q. Zou, D. Zhang, S. Zhang, and X. Luo, "Kinematic and dynamic analysis of a 3-DOF parallel mechanism," *International Journal of Mechanics and Materials in Design*, vol. 17, no. 3, pp. 587–599, 2021, doi: 10.1007/s10999-020-09511-x.
- [27] E. Gaponenko, L. Rybak, and D. Malyshev, "Numerical Method for Determining the Operating Area of a Robot with Relative Manipulation Mechanisms," *Journal of Machinery Manufacture and Reliability*, vol. 49, no. 6, pp. 474–489, 2020, doi: 10.3103/S1052618820060063.
- [28] R. Grassmann and J. Burgner-Kahrs, "On the Merits of Joint Space and Orientation Representations in Learning the Forward Kinematics in SE (3)," in *Proceedings of Robotics: Science and Systems*, Jun. 2019.
- [29] J. Li, Y. Wang, Z. Liu, X. Jing, and C. Hu, "A new recursive composite adaptive controller for robot manipulators," *Space: Science & Technology*, vol. 2021, pp. 1–7, Oct. 2021.
- [30] F. Trotti, E. Ghignoni, and R. Muradore, "A Modified Recursive Newton-Euler Algorithm Embedding a Collision Avoidance Module," in *2021 European Control Conference (ECC)*, pp. 1150–1155, 2021, doi: 10.23919/ECC54610.2021.9654868.
- [31] G. R. Petrović and J. Mattila, "Mathematical modelling and virtual decomposition control of heavy-duty parallel-serial hydraulic manipulators," *Mechanism and Machine Theory*, vol. 170, p. 104680, 2022, doi: 10.1016/j.mechmachtheory.2021.104680.
- [32] C. Tan, H. Zhao, and H. Ding, "Identification of dynamic parameters of closed-chain industrial robots considering motor couplings," *Computers and Electrical Engineering*, vol. 99, p. 107740, 2022, doi: 10.1016/j.compeleceng.2021.107740.

- [33] F. Janabi-Sharifi, A. Jalali, and I. D. Walker, "Cosserrat rod-based dynamic modeling of tendon-driven continuum robots: A tutorial," *IEEE Access*, vol. 9, pp. 68703-68719, 2021, doi: 10.1109/ACCESS.2021.3114305.
- [34] N. Cretescu and M. Neagoe, "Dynamic Modelling of an Isoglide T3 Type Parallel Robot," *New Advances in Mechanisms, Mechanical Transmissions and Robotics: MTM & Robotics 2020 2*, pp. 235-248, 2021.
- [35] C. Alessi, E. Falotico and A. Lucantonio, "Ablation Study of a Dynamic Model for a 3D-Printed Pneumatic Soft Robotic Arm," in *IEEE Access*, vol. 11, pp. 37840-37853, 2023, doi: 10.1109/ACCESS.2023.3266282.
- [36] I. Giorgio and D. Del Vecovo, "Energy-based trajectory tracking and vibration control for multilink highly flexible manipulators," *Mathematics and Mechanics of Complex Systems*, vol. 7, no. 2, pp. 159-174, 2019, doi: 10.2140/memocs.2019.7.159.
- [37] J. Cai *et al.*, "Modeling method of autonomous robot manipulator based on DH algorithm," *Mobile Information Systems*, vol. 2021, pp. 1-10, 2021, doi: 10.1155/2021/5523096.
- [38] A. Tekinalp *et al.*, "Topology, dynamics, and control of an octopus-analog muscular hydrostat," *arXiv preprint, arXiv:2304.08413*, 2023.
- [39] M. Dupac, "Mathematical modeling and simulation of the inverse kinematic of a redundant robotic manipulator using azimuthal angles and spherical polar piecewise interpolation," *Mathematics and Computers in Simulation*, vol. 209, pp. 282-298, 2023, doi: 10.1016/j.matcom.2022.09.019.
- [40] J. Woolfrey and D. Liu, "An Optimal Dynamic Control Method for Robots with Virtual Links," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 12843-12848, Oct. 2022.
- [41] Y. Zhang, D. Kim, Y. Zhao, and J. Lee, "PD control of a manipulator with gravity and inertia compensation using an RBF neural network," *International Journal of Control, Automation and Systems*, vol. 18, no. 12, pp. 3083-3092, 2020, doi: 10.1007/s12555-020-0055-3.
- [42] H. Habibi, C. Yang, R. Kang, I. D. Walker, I. S. Godage, X. Dong, and D. T. Branson, "Modelling an actuated large deformation soft continuum robot surface undergoing external forces using a lumped-mass approach," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5958-5963, Oct. 2018.
- [43] H. Cen and B. K. Singh, "Nonholonomic wheeled mobile robot trajectory tracking control based on improved sliding mode variable structure," *Wireless Communications and Mobile Computing*, vol. 2021, pp. 1-9, 2021, doi: 10.1155/2021/6673200.
- [44] Z. Fu, J. Pan, E. Spyrakos-Papastavridis, Y. H. Lin, X. Zhou, X. Chen, and J. S. Dai, "A lie-theory-based dynamic parameter identification methodology for serial manipulators," *IEEE/ASME Transactions on Mechatronics*, vol. 26, no. 5, pp. 2688-2699, 2020, doi: 10.1109/TMECH.2021.3082115.
- [45] S. Le Cleac'h *et al.*, "Differentiable physics simulation of dynamics-augmented neural objects," *IEEE Robotics and Automation Letters*, vol. 8, no. 5, pp. 2780-2787, 2023, doi: 10.1109/LRA.2023.3230452.
- [46] P. Xu, J. Zhang, Y. Cui, K. Zhang, X. Chen, and Q. Tang, "Dynamic Modelling and Position-force Control for a Mobile Manipulator Based on Particles System Theory," in *2021 IEEE International Conference on Unmanned Systems (ICUS)*, pp. 884-889, Oct. 2021.
- [47] V. Wüest, V. Kumar, and G. Loianno, "Online estimation of geometric and inertia parameters for multirotor aerial vehicles," in *2019 International Conference on Robotics and Automation (ICRA)*, pp. 1884-1890, May 2019, doi: 10.1109/ICRA.2019.8793934.
- [48] H. -S. Fang *et al.*, "AnyGrasp: Robust and Efficient Grasp Perception in Spatial and Temporal Domains," in *IEEE Transactions on Robotics*, vol. 39, no. 5, pp. 3929-3945, Oct. 2023, doi: 10.1109/TRO.2023.3281153.
- [49] M. Goharimanesh, A. Mehrkish, and F. Janabi-Sharifi, "A fuzzy reinforcement learning approach for continuum robot control," *Journal of Intelligent & Robotic Systems*, vol. 100, pp. 809-826, 2020, doi: 10.1007/s10846-020-01199-5.
- [50] J. Hill. *Active disturbance rejection for the bipedal walk of a humanoid robot using the motion of the arms*. The University of Alabama in Huntsville, 2011.
- [51] ABB Robotics, "IRB 140 - Robots (Robotics)," 2023. Available: <http://www.abb.com/2023>.
- [52] J. S. Toquica and J. M. S. Motta, "A methodology for industrial robot calibration based on measurement sub-regions," *The International Journal of Advanced Manufacturing Technology*, pp. 1-18, 2021, doi: 10.1007/s00170-021-07694-z.
- [53] M. Z. Silva *et al.*, "Industrial robotic arm in machining process aimed to 3D objects reconstruction," in *2021 22nd IEEE International Conference on Industrial Technology (ICIT)*, vol. 1, 2021.
- [54] P. Obal *et al.*, *Monitoring the parameters of industrial robots. Methods and Techniques of Signal Processing in Physical Measurements*, 2019.
- [55] M. Sebastian *et al.*, "Study on Robot Simulation and Code Generation Softwares," *International Journal on Emerging Research Areas (IJERA)*, vol. 3, no. 1, pp. 364-369, 2023.
- [56] H. Lin, R. Mandal, and F. S. Wibowo, "BN-LSTM-based energy consumption modeling approach for an industrial robot manipulator," *Robotics and Computer-Integrated Manufacturing*, vol. 85, p. 102629, 2024, doi: 10.1016/j.rcim.2021.102629.
- [57] M. V. V. Satyanarayana, P. P. Kumar, S. D. Praveen, and S. P. Raju. *Computational Design of Kinematics of a Robotic Arm*. Vishnu Institute of Technology, 2022.
- [58] F. Gonçalves *et al.*, "A recursive algorithm for the forward kinematic analysis of robotic systems using Euler angles," *Robotics*, vol. 11, no. 1, p. 15, 2022, doi: 10.3390/robotics11010015.
- [59] A. Talli and D. Marebal, "End-effector position analysis of SCARA robot by using MATLAB," in *Smart Sensors Measurements and Instrumentation: Select Proceedings of CISCON 2020*, pp. 25-33, 2021.
- [60] N. Panigrahi and K. K. Dash, "Kinematic Model Design of a 6 DOF Industrial Robot," in *Applications of Robotics in Industry Using Advanced Mechanisms: Proceedings of International Conference on Robotics and Its Industrial Applications 2019, 2020*.
- [61] A. Wolniakowski *et al.*, "Kinematic Modeling of a Trepanation Surgical Robot System," *Applied Sciences*, vol. 13, no. 16, p. 9110, 2023, doi: 10.3390/app13169110.
- [62] C. Cristoiu *et al.*, "The Importance of Embedding a General forward Kinematic Model for Industrial Robots with Serial Architecture in Order to Compensate for Positioning Errors," *Mathematics*, vol. 11, no. 10, p. 2306, 2023, doi: 10.3390/math11102306.
- [63] C. J. Chen *et al.*, "Reflect on fundamental, modeling and control of industrial robotic manipulators for simulation purposes," in *AIP Conference Proceedings*, vol. 2530, no. 1, 2023, doi: 10.1063/5.0121942.
- [64] L. Lattanzi *et al.*, "Geometrical calibration of a 6-axis robotic arm for high accuracy manufacturing task," *The International Journal of Advanced Manufacturing Technology*, vol. 111, pp. 1813-1829, 2020, doi: 10.1007/s00170-020-06466-9.
- [65] R. Raja, A. Dutta, and B. Dasgupta, "Learning framework for inverse kinematics of a highly redundant mobile manipulator," *Robotics and Autonomous Systems*, vol. 120, p. 103245, 2019, doi: 10.1016/j.robot.2019.103245.
- [66] C. Cristoiu *et al.*, "The Importance of Embedding a General forward Kinematic Model for Industrial Robots with Serial Architecture in Order to Compensate for Positioning Errors," *Mathematics*, vol. 11, no. 10, p. 2306, 2023, doi: 10.3390/math11102306.
- [67] R. M. Ramli, N. A. Herman, and M. A. H. M. Adib, "Simulation on Robot Arm Manipulator Modeling using Dynamic Kinematic," *2020 Emerging Technology in Computing, Communication and Electronics (ETCCE)*, pp. 1-6, 2020, doi: 10.1109/ETCCE51779.2020.9350907.
- [68] A. Talli and D. Marebal, "End-effector position analysis of SCARA robot by using MATLAB," in *Smart Sensors Measurements and Instrumentation: Select Proceedings of CISCON 2020*, pp. 25-33, 2021.
- [69] S. Shrey *et al.*, "Forward kinematic analysis of 5-DOF LYNX6 robotic arm used in robot-assisted surgery," *Materials Today: Proceedings*, vol. 72, pp. 858-863, 2023, doi: 10.1016/j.matpr.2021.12.088.
- [70] H. He, "Modeling and Simulation for a Five DOF Robotic Arm Manipulator," in *Highlights in Science, Engineering and Technology*, vol. 43, pp. 300-307, 2023.
- [71] M. Fazilat, N. Zioui, and J. St-Arnaud, "A novel quantum model of forward kinematics based on quaternion/Pauli gate equivalence: Application to a six-jointed industrial robotic arm," *Results in*

- Engineering*, vol. 14, p. 100402, 2022, doi: 10.1016/j.rineng.2021.100402.
- [72] H. Zhang *et al.*, "Analysis of obstacle avoidance strategy for dual-arm robot based on speed field with improved artificial potential field algorithm," *Electronics*, vol. 10, no. 15, p. 1850, 2021, doi: 10.3390/electronics10151850.
- [73] V. M. Brathikan, P. Prashanth, and S. Vikash, "Design and development of 5-axis robot," *Materials Today: Proceedings*, vol. 62, pp. 2179-2184, 2022, doi: 10.1016/j.matpr.2021.08.375.
- [74] I. S. Howard, "Design and Kinematic Analysis of a 3D-Printed 3DOF Robotic Manipulandum," in *Annual Conference Towards Autonomous Robotic Systems*, pp. 227-239, 2023.
- [75] A. Crisan and I. Negrean, "The Jacobian matrix based on the transfer matrices," *ACTA Technica Napocensis-Series: Applied Mathematics, Mechanics, and Engineering*, vol. 62, no. 1, 2019.
- [76] G. Boiadjiev, E. Krastev, I. Chavdarov, and L. Miteva, "A novel, oriented to graphs model of robot arm dynamics," *Robotics*, vol. 10, no. 4, p. 128, 2021, doi: 10.3390/robotics10040128.
- [77] Y. Ding, X. Zhu, X. Sun, J. Zhang, and X. Chen, "Soft sensor simulation of minimum energy consumption of joint manipulator drive system based on improved BP neural network," in *Journal of Physics: Conference Series*, vol. 1626, no. 1, p. 012020, Oct. 2020, doi: 10.1088/1742-6596/1626/1/012020.
- [78] M. Mahdavian *et al.*, "Optimal trajectory generation for energy consumption minimization and moving obstacle avoidance of a 4DOF robot arm," in *2015 3rd RSI International Conference on Robotics and Mechatronics (ICROM)*, 2015.
- [79] D. Fang, J. Yang, J. Shang, Z. Wang, and Y. Feng, "A Novel Energy-Efficient Wobble Plate Hydraulic Joint for Mobile Robotic Manipulators," *Energies*, vol. 11, no. 11, p. 2915, 2018, doi: 10.3390/en11112915.
- [80] D. Catanoso, A. Chakrabarty, J. Fugate, U. Naal, T. M. Welsh, and L. J. Edwards, "OceanWATERS lander robotic arm operation," in *2021 IEEE Aerospace Conference (50100)*, pp. 1-11, Mar. 2021, doi: 10.1109/AERO50100.2021.9438224.
- [81] N. Zioui, A. Mahmoudi, and M. Tadjine, "Representing quantum spins in different coordinate systems for modelling rigid body orientation," *Karbala International Journal of Modern Science*, vol. 9, no. 3, p. 11, 2023, doi: 10.33640/2405-609X.3605.
- [82] N. Zioui, Y. Mahmoudi, A. Mahmoudi, M. Tadjine, and S. Bentouba, "A novel quantum-computing-based quaternions model for a robotic arm position," *International Journal of Computational Intelligence in Control*, vol. 13, no. 2, pp. 71-77, 2021.
- [83] N. Zioui, Y. Mahmoudi, A. Mahmoudi, M. Tadjine, and S. Bentouba, "A New Quantum-computing-based Algorithm for Robotic Arms and Rigid Bodies' Orientation," *Journal of Applied and Computational Mechanics*, vol. 7, no. 3, pp. 1836-1846, 2021, doi: 10.22055/JACM.2021.36439.2894.
- [84] N. Zioui, A. Mahmoudi, and M. Tadjine, "Design of a new hybrid linearising-backstepping controller using quantum state equations and quantum spins," *International Journal of Automation and Control*, vol. 17, no. 4, pp. 397-417, 2023, doi: 10.1504/IJAAC.2023.124412.
- [85] N. Zioui, A. Mahmoudi, Y. Mahmoudi, and M. Tadjine, "Quantum computing-based state domain equations and feedback control," *Results in Applied Mathematics*, vol. 19, p. 100385, 2023, doi: 10.1016/j.rinam.2022.100385.