

## Article

# 6-DOFs Robot Placement Based on the Multi-Criteria Procedure for Industrial Applications

Francesco Aggogeri  and Nicola Pellegrini \* 

Department of Mechanical and Industrial Engineering, University of Brescia, Via Branze, 38, 25123 Brescia, Italy; francesco.aggogeri@unibs.it

\* Correspondence: nicola.pellegrini@unibs.it; Tel.: +39-030-3715580

**Abstract:** Robot acceptance is rapidly increasing in many different industrial applications. The advancement of production systems and machines requires addressing the productivity complexity and flexibility of current manufacturing processes in quasi-real time. Nowadays, robot placement is still achieved via industrial practices based on the expertise of the workers and technicians, with the adoption of offline expensive software that demands time-consuming simulations, detailed time-and-motion mapping activities, and high competencies. Current challenges have been addressed mainly via path planning or robot-to-workpiece location optimization. Numerous solutions, from analytical to physical-based and data-driven formulation, have been discussed in the literature to solve these challenges. In this context, the machine learning approach has proven its superior performance. Nevertheless, the industrial environment is complex to model, generating extra training effort and making the learning procedure, in some cases, inefficient. The industrial problems concern workstation productivity; path-constrained minimal-time motions, considering the actuator's torque limits; followed by robot vibration and the reduction in its accuracy and lifetime. This paper presents a procedure to find the robot base location for a prescribed task within the robot's workspace, complying with multiple criteria. The proposed hybrid procedure includes analytical, physical-based, and data-driven modeling to solve the optimization problem. The contribution of the algorithm, for a given user-defined task, is the search for the best robot base location that enables the target points, maximizing the manipulability, avoiding singularities, and minimizing energy consumption. Firstly, the established method was verified using an anthropomorphic robot that considers different levels of a priori kinematics and system dynamics knowledge. The feasibility of the proposed method was evaluated through various simulations for small- and medium-sized robots. Then, a commercial offline program was compared, considering three scenarios and fourteen robots demonstrating an energy reduction in the 7.6–13.2% range. Moreover, the unknown joint dependency in real robot applications was investigated. From 11 robot positions for each active joint, a direct kinematic was appraised with an automatic DH scheme that generates the 3D workspace with an RMSE lower than 65.0  $\mu\text{m}$ . Then, the inverse kinematic was computed using an ANN technique tuned with a genetic algorithm showing an RMSE in an S-shape task close to 702.0  $\mu\text{m}$ . Finally, three experimental campaigns were performed with a set of tasks, repetitions, end-effector velocity, and payloads. The energy consumption reduction was observed in the 12.7–22.9% range. Consequently, the proposed procedure supports the reduction in workstation setup time and energy saving during industrial operations.



**Citation:** Aggogeri, F.; Pellegrini, N. 6-DOFs Robot Placement Based on the Multi-Criteria Procedure for Industrial Applications. *Robotics* **2024**, *13*, 153. <https://doi.org/10.3390/robotics13100153>

Received: 9 September 2024

Revised: 15 October 2024

Accepted: 15 October 2024

Published: 16 October 2024



**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

**Keywords:** industrial robot; multi-criteria; modeling; energy consumption; optimization

## 1. Introduction

The positioning of a robot base influences the high repeatability, reliability, and dexterity performance of robots in manufacturing, welding, packaging, and assembly applications. Although the perception paradigm is tangible for enhancing the level of automation, in

an industrial context, the current system ability may be significantly enhanced by evaluating different factors: the safety-related boundaries, the maximum accelerations, the end-effector's arrangement, and the singularity avoidance. In addition, the existing layout adjustment cannot be easily managed due to physical constraints, such as workpiece inertia or geometry, and other input/output interdependencies that impact the placement of the robotic system. Moreover, the robot's accuracy is affected by geometric errors, deformation and distortion errors, payloads, and temperature profiles [1–4]. The workstation layout is a central factor in satisfying the user-defined tasks and the workspace complexity that is still determined through a time-consuming trial-and-error process, and it is usually managed via the high skills and expertise of operators, technicians, and engineers. In recent years, the literature has presented several studies that may be classified into two approaches: path planning and robot-to-workpiece location optimization. Robot path planning aims to search for a collision-free path between the starting and final target frame in the robot space that meets a set of constraints [5]. Planning methods are described with graph-based search procedures [6], and the algorithms were developed in particular for cooperative unmanned mobile robots. The artificial potential field-based method [7] is a technique that presents offline and online path adaptations with unknown obstacles with resultant errors of lower than 16.6% and 17.4%, respectively. Then, machine learning (ML) methodologies and simulation have been adopted for online planning in time-variant conditions [8–10]. In [11], an experimental study of the vibrations of a roller shutter gripper on a robotic palletizing station was presented using a FANUC Robotguide environment, demonstrating effective vibration reduction. In [12], Bucinskas et al. provide a methodology for the online deep Q-learning-based approach intended to increase positioning accuracy at key points by analyzing experimentally predetermined robot properties and their impact on overall accuracy. The KUKA-YouBot robot has been used, and the proposed ML-based compensation method resulted in a positioning error decrease at the trajectory of 30% of the tolerance declared. In [13], a deep neural network is presented for path estimation using reinforcement learning. Although ML has proven its superior performance in many studies, several challenges are open: The environment is often significantly complex to model, generating an extra training effort and making the learning procedure inefficient [14]. Another concern is the over-fitting problem—real robot positioning is time-consuming for the training phase and shows inadequate model extension in unstructured environments [15]. In an industrial context, the robotic applications are determined in path-constrained minimal-time motions to increase workstation productivity, considering the actuator's torque limits. The practical consequence is the increase in the robot tool center point (TCP) and vibration, and a reduction in its accuracy and lifetime. The second approach—robot-to-workpiece placement—has been studied and investigated to address the aforementioned issues. The aim is to focus on planning the manipulator operations determined by the robot base location to accomplish a given task under a set of criteria. The problem's objective functions mostly comprise the manipulability and velocity index [16], the cycle time, and the joint motion bounds. Nevertheless, the robot placement solution was discussed as a multi-dimensional function in a controlled time-variant workspace. Numerous analytical, physical-based, and data-driven formulation methods have been presented to solve this challenge. In [17], Guanhua et al. address a technique for robot location in drilling applications based on a bidimensional manifold in joint space using a particle swarm optimization (PSO) procedure applied to identify the best parameters. The industrial context is aircraft machining, such as shaped panel drilling and isometric positioning. A procedure is proposed to model the positioning error to predict and compensate for the deviations. A dual solution of bifurcation was studied and simulated. The experimental campaign conducted on a KUKA robot verified the improvement in terms of position error of 12–22%, with 255 real targets positioned and a resultant average positioning error of close to 0.50–0.69 mm. Ur-Rehman et al. in [18] present a multi-objective placement for parallel kinematics machines based on power consumption and shaking forces. The proposed approach is verified using the Orthoglide, a three-degrees-of-freedom (DoF) robot. The

four-sided trajectory was evaluated to gather pocketing operations. The shaking forces analyses revealed that the inertial contributions exerted on the PKM base influenced the energy consumption. A multi-objective genetic algorithm was used to find the boundaries for a rectangular test trajectory, with a width of 30.0 mm, a length of 60.0 mm, and a TCP velocity of 600.0 mm/s. In the work, a campaign demonstrated a reduction of 60% in the power consumption and a decrease of 17% in the shaking forces. In addition, refs. [19,20] describe optimization techniques for minimizing robot energy consumption. In [21], Doan and Lin focus on the redundancy resolution scheme to evaluate welding equipment for offshore assembly of buildings. The welding technology is applied to large workpieces and compared with the robot's workspace. The robot position is analyzed based on joint limitation, singularities, and collision avoidance. Non-differentiable terms are determined by applying a modified particle swarm optimization (MPSO) that is designed and developed. The feasibility is demonstrated via simulation using ABB RobotStudio digital twin commercial software and using an experimental dry run welding test bench. The workpiece size is 2.0 m  $\times$  3.0 m, the area of interest for welding is a diameter of 355.6 mm, and the identified placement is 200 mm  $\times$  350 mm  $\times$  250 mm. Son and Kwon, in [22], propose a method based on a convex programming approach for the location of an anthropomorphic robot with a spherical wrist. The procedure determines a convex solution without solving the inverse kinematics (IK). The KUKA KR6 R900-2 robot was selected as a representative system; formulation and simulation were completed to verify the reachability, singularity avoidance, and manipulability criteria via MATLAB and KUKA Sim Pro software. Ren et al. and works by other researchers described constrained optimization problems for painting robot manipulators [23–26]. The proposals refer to the penalty cost function and Lagrange index to search for the optimal robot base position. The works test the algorithms using the IRB5500 robot, obtaining a residual error of close to 10% for flat, cylindrical, and truncated conical surfaces. Spensieri et al., in [27], present a derivative-free model to attain multiple task positions. The results are stated via simulation for cycle time minimization; nevertheless, the work does not study rotations due to computation boundaries. Other methodologies focus on searching for a likely robot location for slow-motion application. In [28], the authors investigate the available workspace and the ability to sustain the resultant forces at the end effector during motion. Successive solutions are computed with a searching method by incrementally fixing the constraints.

Although several advances have been investigated in earlier studies, a large portion is inappropriate for real-time or shop floor practical applications due to user-defined ad hoc criteria, computational limits, and time-consuming procedures. Likely functions, boundaries, and industrial efficiency indexes differ significantly on robot workspace complexity, while the best robot position may be appraised differently depending on the selected application criteria. Robot positioning for predefined tasks, considering the reachability, is inadequate for avoiding robot singularity and guaranteeing collision avoidance. Moreover, it is necessary to react to production mix variability and layout modifications promptly, including time and cost efficiency, with a procedure that could be implemented directly by a technician or operator. Limited a priori kinematics knowledge of the robot usually leads to a bang-bang motion. In contrast, the overfitting of robot modeling drives complexities for online implementation.

Beyond the industrial practices and research lab developments, in the standardization context the focus is centered on the safety of industrial robots and service robots to enable innovative products to be brought to the market. ISO/TC 299 [29] has the goal of fostering the growth of the robotics market by introducing standards in fields like terminology, performance measurement, and modularity. In a similar vein, ISO-10218 Part 1 ("Safety of Robots") [30] and Part 2 ("Safety of Robot Integration") [31] were intended to set forth safety requirements for robots and robot systems in general. Moreover, safety requirements for designing and implementing industrial robot systems are specified in the voluntary industry consensus standard ANSI/RIA R15.06-2012 ("R15.06"). Finally, the technical report (TR) ANSI RIA TR R15.506 [32] defines the guidelines for existing industrial robot

applications. These standards ensure that technologies are used safely, efficiently, and in harmony. The authors' contribution to the progress of industrial, laboratory, and standard scenarios is to validate a procedure to find the robot base location for a prescribed task within the robot's workspace, complying with multiple criteria for industrial application. The four-step methodology is verified by simulations and experimental campaigns considering a set of tasks and scenarios that explore the a priori knowledge of kinematic/dynamic modeling and the unknown joint dependency in a real test setup. Given the user-defined task in the cell layout reference frame, the contribution of the proposed technique is to search for the optimal robot base location and configurations that consider the reachable target points and joint range of motions, maximize the manipulability, avoid singularities, and minimize the energy consumption. The procedure uses analytical formulation to examine the reaching ability of the user task. Then, for an unknown model, a scheme leads the operator to obtain the direct kinematic model via digital twin software or using a real robot to provide the input for the appraisal of the data-driven model based on the artificial neural network technique tuned via a genetic algorithm. Consequently, the criteria are evaluated to identify a feasible robot base location from collision avoidance to singularity and manipulability analyses and energy consumption computation. In case there is no suitable performance, the iterative procedure updates the robot base position—the whole procedure may be conducted in minutes rather than the hours or days that conventional methods require.

## 2. Integrated Multi-Criteria Procedure for Robot Base Location

In this section, a selection of criteria have been investigated for the 6-DOF articulated robot placement definition. The industrial input synthesis as the layout description, the existing equipment, the fixturing systems/tooling, and the surrounding environmental considerations are the starting point for the problem formulation. Optimal robot positioning searching requires iterative multi-criteria computations considering the following areas:

1. Task specification, workspace definition, and reachability measure;
2. Robot inverse kinematic data-driven modeling for collision avoidance;
3. Singularity and manipulability analyses;
4. Energy consumption appraisal.

### 2.1. Task Specification, Workspace Definition, and Reachability Measure

Conventional manipulators comprise an open chain of serially connected joints (revolute or prismatic) to accomplish three-dimensional motions in the Cartesian workspace. The 6-DoF robot is the usual architecture of industrial manipulators, as shown in Figure 1a. A task  $T^i$  is a combination of user-defined points-to-points that the robot needs to approach via the TCP positions. To verify the robot's reachability for a defined task, it is required to confirm that all these positions are within the robot's workspace. The preliminary reachability assessment is the measure of the feasible tasks that the robot can achieve considering the manipulator geometrical features and mechanical structure abilities. The measure can be attained by approximating the wrist of the robot and the position center for joint  $J_2$ , as described in Figure 1b. Let  $\mathbf{b}$  be the robot base location for a specific task  $T^i$ , and the reachability  $\mathbf{m}$  is formulated in Equations (1)–(3):

$$m = |D^i - r_c| \quad (1)$$

$$D^i = \|w^i - b\|_2 \quad (2)$$

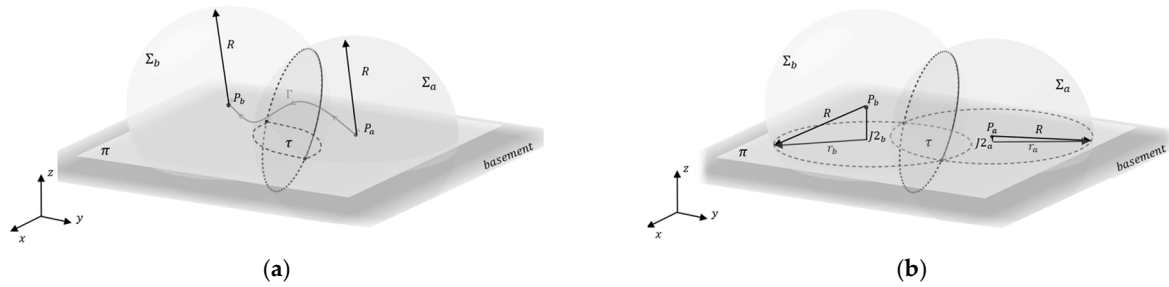
$$r_c = \frac{r + R}{2} \quad (3)$$

where  $D^i$  is the Euclidean distance between the TCP position  $w^i$  and the robot base location  $b$ , and  $r_C$  is the distance concerning the base position for the centroid of the robot for the task  $T^i$ .



**Figure 1.** The 6-DOF robot representation: serial-linked revolute joints (a); preliminary reachability of the ABB IRB1200-5/0.9 robot for reference (b).

As shown in Figure 2a, the workspace is expressed using the coordinate  $P_a$ , which identifies the reference frame of the starting point in the task  $T^i$  to the coordinate  $P_b$  (referring to the termination point), following the trajectory  $\Gamma$ . Consequently, the plane  $\pi$  parallel to the ground base is formulated, where  $z$  is the vertical position of  $\pi$ ,  $z_b$  is the vertical coordinate of the robot base, and  $\Delta z$  is the geometrical feature of the robot as the distance between the robot base and the  $J_2$  centroid. The  $\tau$  region in Figure 2a represents the area where it is possible to place the centroid in the  $J_2$  joint of the 6-DOF robot to reach points  $P_a$  and  $P_b$  for a selected task  $T^i$ .



**Figure 2.** Workspace analysis based on task  $T^i$  (a); region of the  $J_2$  robot based on task  $T^i$  (b).

The  $\tau$  region is formulated as in Equation (4). The  $\pi$  plane's vertical location bounds  $\pi_{[min-max]}$  are analytically expressed in Equations (5) and (6).

$$\tau = (x, y, z) \left| \sqrt{(x - x_{J2_a})^2 + (y - y_{J2_a})^2} \leq r_a, \sqrt{(x - x_{J2_b})^2 + (y - y_{J2_b})^2} \leq r_b, z = \pi \right. \quad (4)$$

$$\pi_{[min-max]}: [z_\pi - h; z_\pi + h] \quad (5)$$

$$h = \sqrt{R^2 + \left( \frac{\sqrt{(x_{P_a} - x_{P_b})^2 + (y_{P_a} - y_{P_b})^2 + (z_{P_a} - z_{P_b})^2}}{2} \right)^2 \cos \left( \sin^{-1} \left( \frac{\frac{|z_{P_a} - z_{P_b}|}{2}}{\frac{d}{2}} \right) \right)} \quad (6)$$

where  $d$  is the Euclidean difference relating the  $P_a$  and  $P_b$  reference coordinates.

## 2.2. Robot Data-Driven Modeling for Collision Avoidance

The user defined task  $\mathbf{T}^i$  is stated as  $S$  composed of  $n$  Cartesian frames  $[x, y, z, R_x, R_y, R_z]$  and the corresponding time  $t$ , as stated in Equation (7):

$$S = \left\{ (t, x, y, z) \mid t_0 \leq t \leq t_f, (X, y, z) \in R^3 \right\} \quad (7)$$

For a given set  $S$  of a known task  $\mathbf{T}^i$ , the inverse kinematics are computed. The model may be formulated with analytical, numerical, or data-driven representation. In the present work, the problem is solved by implementing a sequential data-driven approach [33]. The scheme applies an automated Denavit–Hartenberg (D-H) estimation tool to compute the direct kinematics (DK) model, which is needed to generate the workspace dataset [34,35]. Additionally, the artificial neural network (ANN) is chosen as the reference algorithm. The number of layers and the number of hidden neurons per layer are set via a genetic algorithm (GA) computation for each joint. The advantages of the simulated data points derived from the DK model is fewer experimental trials or virtual robot positions that impact the total setup time.

The indicator of position error at each point is selected as the distance between the TCP position and the position computed by model via the ANN algorithm, and the formulation is reported in Equation (8). The position error is negligible if the accuracy is lower than a defined threshold  $\varepsilon$  (set equal to 5.0  $\mu\text{m}$ ).

$$Err_i = \begin{cases} 0 & \text{if } |TCP_{pose} - T_{ANN}^n b| < \varepsilon \\ |TCP_{pose} - T_{ANN}^n b| & \text{otherwise} \end{cases} \quad (8)$$

where  $\mathbf{b}$ ,  $TCP_{pose}$ , and  $T_{ANN}^n$  are the robot base location, the TCP position, and the kinematic model via the ANN algorithm of the  $n$ -DOF manipulator, respectively.

Moreover, collision avoidance is a safety requirement that is integrated following Equation (9). The constraint condition is applied to all  $n$ : 6 robot joints and the  $k$  external objects and entities present in the industrial environment.

$$C_{nj} \geq O_n + O_j \quad n = \{1, \dots, 6\} \wedge j = \{1, \dots, k\} \quad (9)$$

where  $C_{nj}$  is the distance between two reference frame origins (reference  $n$ , reference  $j$ ), and  $O_n$  and  $O_j$  are the effective radius of the  $n^{\text{th}}$  robot joints and the  $j^{\text{th}}$  obstacle, respectively.

## 2.3. Singularity and Manipulability Analyses

In addition, the manipulator configuration in 3D workspace influences the task execution during industrial operation. The joint variables are represented by  $\mathbf{q} = [q_1, q_2, \dots, q_n]^T$ , and the TCP position and orientation by  $\mathbf{p} = [p_1, p_2, \dots, p_g]^T$ , assuming  $g \leq n$ . The Jacobian matrix  $\mathbf{J} \in \mathbb{R}^{6 \times 6}$  is the transformation relating to the joint and TCP movements, as stated in Equation (10):

$$\dot{\mathbf{x}}_n = \mathbf{J} \cdot \dot{\boldsymbol{\theta}} \quad (10)$$

where  $\dot{\mathbf{x}}_n \in \mathbb{R}^{6 \times 1}$  and  $\dot{\boldsymbol{\theta}} \in \mathbb{R}^{6 \times 1}$  are the TCP and joint motion vectors, respectively. The analysis is added in the proposed procedure to avoid the arrangements with  $\text{rank}(\mathbf{J}) < 6$ . Correspondingly, the manipulability evaluation  $\mu$  is adopted to assess the dexterity of the robot, as indicated with Equation (11):

$$\mu = \sqrt{\det(\mathbf{J} \cdot \mathbf{J}^T)} = \prod_i \delta_i \quad (11)$$

where  $\delta_i$  indicates the singularity of the Jacobian matrix  $\mathbf{J}$ . Therefore, the manipulability  $\mu$  tending to null is an index of singularity approaching. Similarly, the greater the manipulability index, the higher the ability to prevent singular configurations.

### 2.4. Energy Consumption Appraisal

In the present work, the dynamics are a significant factor for the robot base positioning definition. The estimation of power consumption for a prescribed task  $\mathbf{T}^i$  is derived from the determination of the generalized resultant forces of vector  $\tau$ . The relation between energy and force is coupled to the  $n$ -joint at the instant period  $[t_0; t_f]$ . Then, the mechanical power is described as in Equation (12):

$$M_P = \sum_{n=1}^n \int_{t_0}^{t_f} [\tau_n(t) \dot{\theta}_n(t)]^2 dt \quad (12)$$

The procedure calculates the robot base location to minimize Equation (12) for a selected area. In the case of an industrial 6-DOF robot, the  $\tau_2$  is significantly larger than others; therefore, the expected contribution to the energy consumption is affected by  $J_2$ .

### 3. Objective Function Formulation for Robot Positioning

The procedure to determine the optimal robot base position has been described considering the selected multi-criteria industrial drivers defined in Section 2. Figure 3 shows a flow diagram of the integrated scheme for establishing the brute-force robot base position. The starting phase is the selection of the robot with the geometrical properties, the user requirement, and the task definition under investigation. Then, a set of robot placements and the assessment of the reachability measure, including the space limits within the scope for searching optimization, are carried out. In phase 2, the robot model formulation is performed to confirm the workspace and robot configuration for the tasks based on a data-driven approach, assuring collision avoidance. The degree of accordance between model and scenario could be appraised numerically or via experimental tests. In phase 3, the robot positioning for a singularity-free task is investigated using a manipulability index that needs to be maximized. During the subsequent phase 4, the iterations are related to the energy appraisal for each base position to determine the minimum energy consumed to validate the procedure. The robot location may be obtained considering all the described criteria, and the objective function  $f$  that needs to be minimized is formulated in the following Equations (13)–(16):

$$f = \alpha \cdot \frac{f_1}{f_1^0} + (1 - \alpha) \cdot \frac{f_2}{f_2^0} + \frac{f_3}{f_3^0} \quad 0 \leq \alpha \leq 1 \quad (13)$$

$$f_1 = \sum_{task=1}^i \sum_{g=1}^3 \sum_{n=1}^4 \left( T_{ANN}^7(g, n) - TPC_{pose}^7(g, n) \right)_i^2 \quad (14)$$

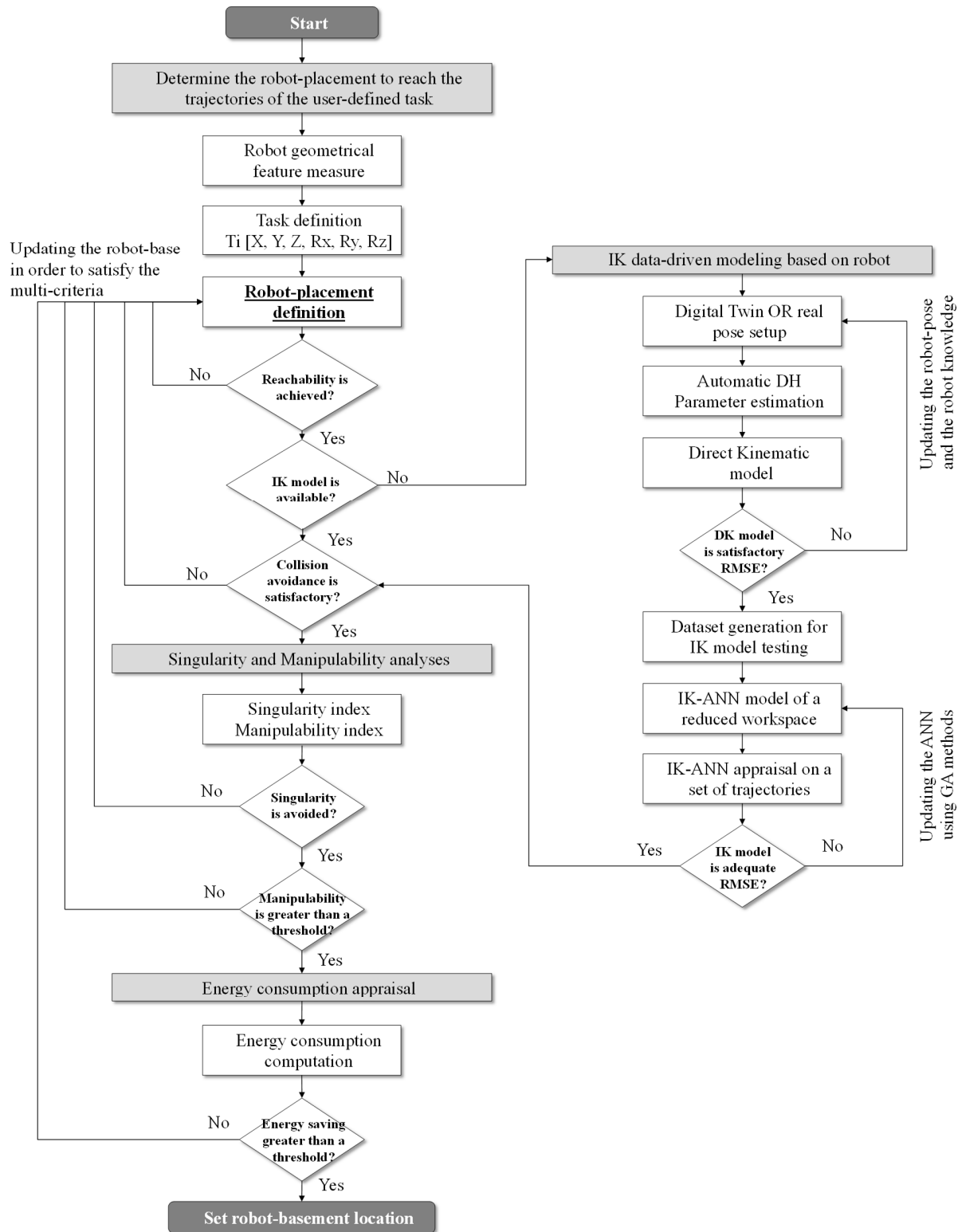
$$f_2 = \sum_{task=1}^i \left( \frac{1}{\mu_i^2} \right) \quad (15)$$

$$f_3 = \sum_{n=1}^n \beta \cdot \int_{t_0}^{t_f} [\tau_n(t) \dot{\theta}_n(t)]^2 dt \quad 0 \leq \beta \leq 1 \quad (16)$$

where  $f_1$  represents the deviation contribution between the estimated TCP positions and the related predefined positions,  $f_2$  is the sum of the inversed manipulability measures, and  $f_3$  is the mechanical power contribution at task  $\mathbf{T}^i$ . The values  $f_1^0$ ,  $f_2^0$ , and  $f_3^0$  are derived from the calculation of IK-ANN modeling, manipulability, and energy consumption, respectively. In particular,  $\mu_i$  is the manipulability index for the robot in position  $I$ ,  $\alpha$  is a weighting factor of the accuracy–manipulability compromise, and  $\beta$  is the weight factor of each joint's contribution to energy consumption. Additionally, the initial and final velocity conditions are given in Equation (17) to represent a point-to-point motion:

$$\begin{aligned}\dot{\theta}_n(t_0) &= 0 & n &= 1, \dots, n \\ \dot{\theta}_n(t_f) &= 0 & n &= 1, \dots, n\end{aligned}\quad (17)$$

where  $t_0$  and  $t_f$  are the initial and final task times for each joint  $n$ , respectively.



**Figure 3.** Flow chart of the multi-criteria procedure for brute-force robot base location.

#### 4. Simulation of Robot Base Placement and Case Studies

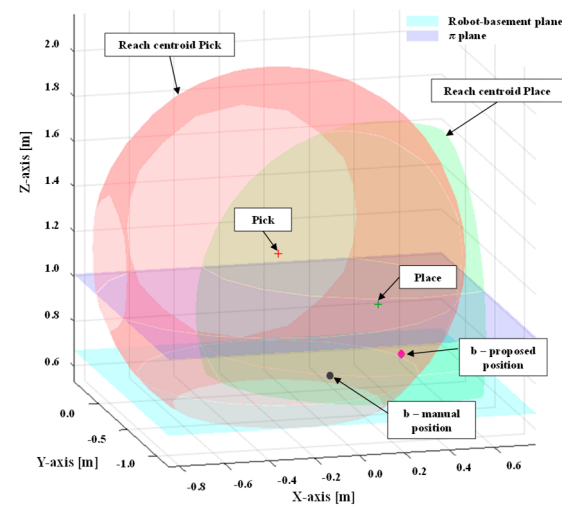
The robot placement procedure was validated analytically and using the digital twin technique for a pick-and-place task application that is significant in the manufacturing industry. Some Ti tasks with two to eight target frames are presented, considering the outcomes of the multi-criteria objective function.

##### 4.1. Robot-to-Workpiece Placement: Analytical Method

The first step of the proposed procedure is the collection of the following inputs: the robot link dimensions, the layout constraints, and the TCP positions (coordinates and orientation), as shown in Table 1—Case 1. The handled working volume based on the pick-and-place reference frames, the robot base horizontal plane, and the user-defined positions (pick, place, b—manual position) are illustrated in Figure 4 in the MATLAB environment.

**Table 1.** Input variables, bounds, and parameters.

| Case 1               |                       |                                                                  |                     |                      |                           |                             |
|----------------------|-----------------------|------------------------------------------------------------------|---------------------|----------------------|---------------------------|-----------------------------|
| Target TCP Position  | Position [X, Y, Z] mm | Rotation [R <sub>X</sub> , R <sub>Y</sub> , R <sub>Z</sub> ] deg | Robot J1–J2 Link mm | Robot Base Height mm | z-Axis Robot Placement mm | b-Manual Position [X, Y] mm |
| TCP <sub>Pick</sub>  | [−100, −200, 1200]    | [0, 0, 0]                                                        | 448.0               | 339.1                | 670.0                     | [200, 100]                  |
| TCP <sub>Place</sub> | [500, 400, 800]       | [0, 0, 0]                                                        |                     |                      |                           |                             |



**Figure 4.** Case 1—graphic representation: pick reference frame, place reference frame;  $\pi$  plane, b-manual position.

In Case 1, the weighting factor  $\beta$  related to energy consumption is assumed to be identical for each joint. To maximize the manipulability and the robot dexterity, the  $\alpha$  value is a significative assumption that also impact the acceptable TCP positional deviation; therefore, the initial set is equal to 0.15.

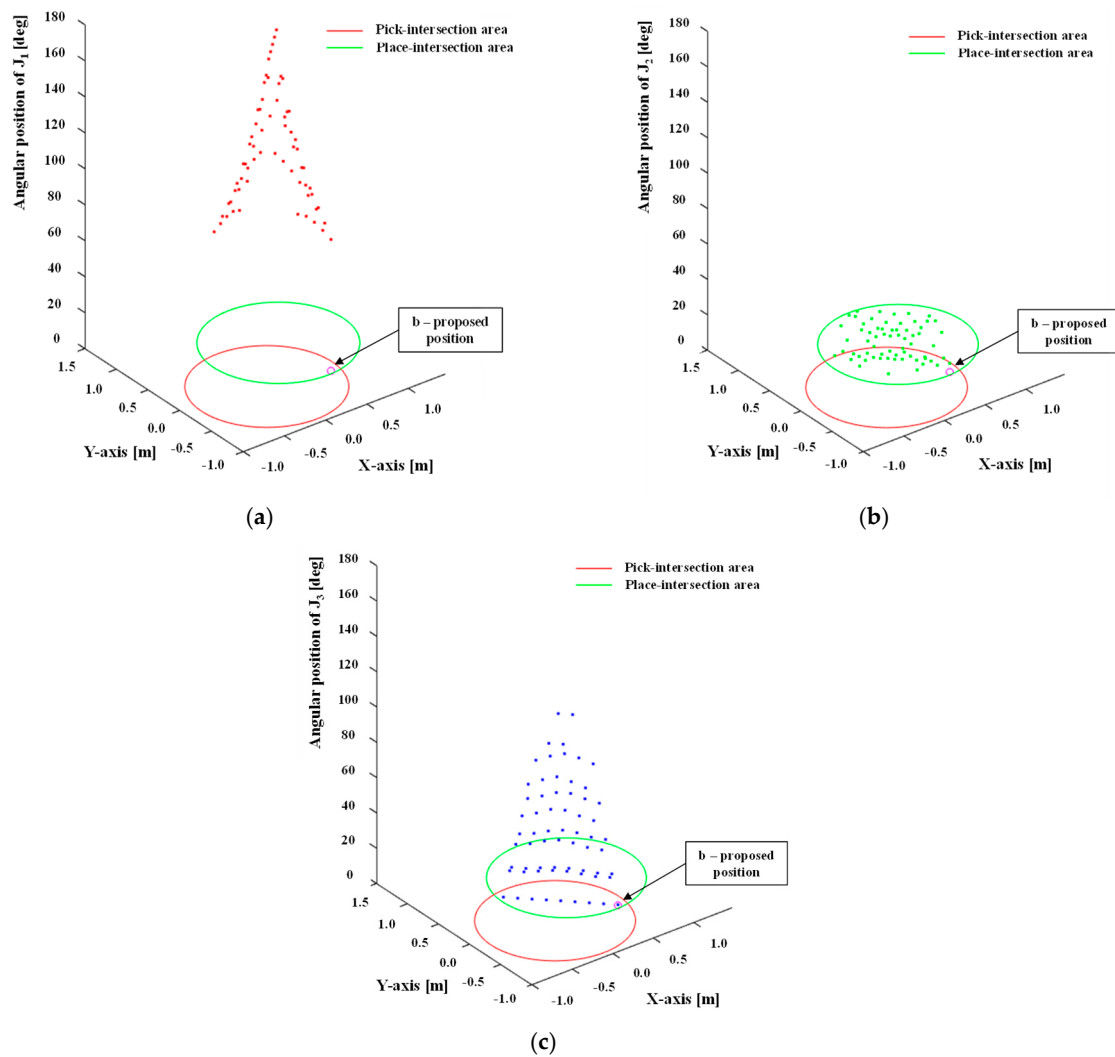
Table 2 summarizes the analytical results of Case 1, comparing the proposed b-position on the XY plane for a chosen z-plane placement of 670.0 mm and the b-position manually identified qualitatively by the user. The b-position proposed is placed at [X, Y] = [600.0 mm, 300.0 mm], and the manual b-position is assessed at [X, Y] = [200.0 mm, 100.0 mm].

Figure 5a–c show the proposed b-position of the robot placement on the XY plane, corresponding to the solution using an identical weighting factor for each DOF—here, the J<sub>1</sub>, J<sub>2</sub>, and J<sub>3</sub> joints of the robot configuration. To demonstrate the achievement of the proposed procedure concerning manual position appraisal, Figure 5a shows the impact of the b-position on the J<sub>1</sub> range of motion; similarly, Figure 5b reports the J<sub>2</sub> influence, and

Figure 5c shows the  $J_3$  effect. Case 1 is considerably affected by the range of motion of  $J_1$  and  $J_3$ , impacting TCP accuracy, manipulability, and energy consumption criteria.

**Table 2.** Output variables: b-manual comparison with b-proposed positions on the XY plane.

| Case 1               |                       |                                        |                           |                             |                               |
|----------------------|-----------------------|----------------------------------------|---------------------------|-----------------------------|-------------------------------|
| Target               | Position [X, Y, Z] mm | Rotation [ $R_X$ , $R_Y$ , $R_Z$ ] deg | Robot Placement z-Axis mm | b-Manual Position [X, Y] mm | b-Proposed Position [X, Y] mm |
| TCP <sub>Pick</sub>  | [−100, −200, 1200]    | [0, 0, 0]                              | 670.0                     | [200, 100]                  | [600, −300]                   |
| TCP <sub>Place</sub> | [500, 400, 800]       | [0, 0, 0]                              |                           |                             |                               |



**Figure 5.** Case 1—angular joint contribution based on the b-position of the robot placement on the  $\pi$ -plane:  $J_1$  contribution (a);  $J_2$  contribution (b);  $J_3$  contribution (c).

Case 2 is evaluated in order to investigate the contribution of the vertical position of the  $J_2$  centroid on the robot base location. The task  $T^1$  is equivalent to that in Case 1, and the robot configuration is selected with vertical z-axis moved from 670.0 mm to 400.0 mm. The output variables, b-manual, and b-proposed positions on the XY plane are reported in Table 3.

**Table 3.** Output variables: b-manual comparison with b-proposed positions on the XY plane.

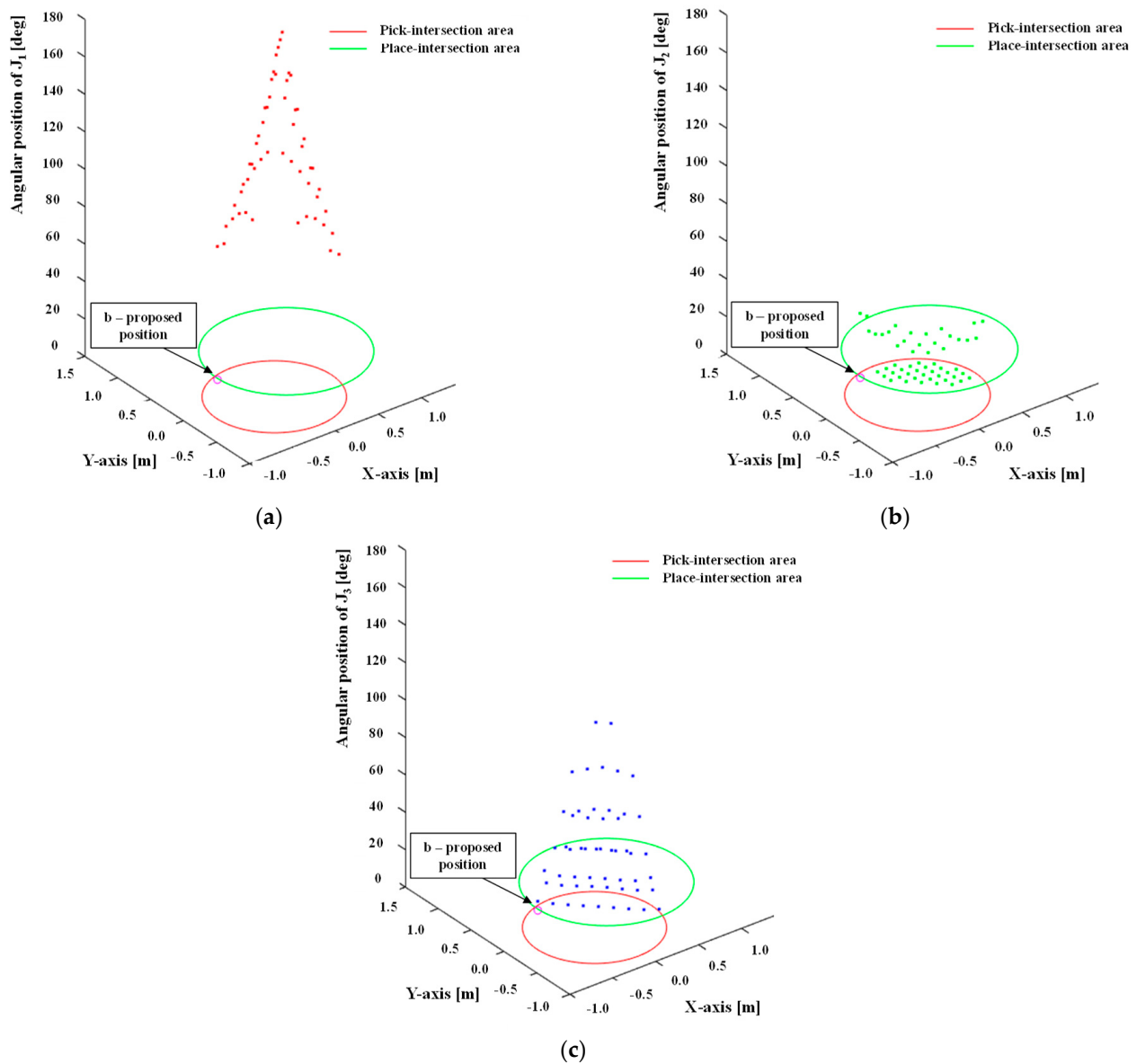
| Case 2             |                       |                                  |                           |                             |                               |
|--------------------|-----------------------|----------------------------------|---------------------------|-----------------------------|-------------------------------|
| Target             | Position [X, Y, Z] mm | Rotation [ $R_X, R_Y, R_Z$ ] deg | Robot Placement z-Axis mm | b-Manual Position [X, Y] mm | b-Proposed Position [X, Y] mm |
| T <sub>Pick</sub>  | [−100, −200, 1200]    | [0, 0, 0]                        | 400.0                     | [200, 100]                  | [−300, 400]                   |
| T <sub>Place</sub> | [500, 400, 800]       | [0, 0, 0]                        |                           |                             |                               |

The manual b-position is considered constant in Case 1 at  $[X, Y] = [200.0 \text{ mm}, 100.0 \text{ mm}]$ . The b-position analytical results are computed for Case 2 as  $[X, Y] = [−300.0 \text{ mm}, 400.0 \text{ mm}]$ ; meanwhile, the results of Case 1 are  $[X, Y] = [600.0 \text{ mm}, −300.0 \text{ mm}]$ , highlighting a difference in the location in the two scenarios and confirming the goodness of the preliminary assessment.

Figure 6a–c shows the b-position determined by joints' contribution to altering the vertical placement. The weighing factor  $\beta$  of joints  $J_1$ – $J_3$  are the same, although the impact is significant for  $J_1$  and  $J_3$  in Figure 6a,c, respectively. The observed difference in  $[X, Y]$  relating to Case 1 and Case 2 in terms of robot location is  $[X, Y]$  as  $[900.0 \text{ mm}, 700 \text{ mm}]$ . The energy consumption comparison of Case 2 for the b-manual and b-proposed robot placement is shown in Figure 7a,b—which is based on the ABB IRB1200-5/0.9 features for reference. In Figure 7a, the energy consumption in the Joule bar chart shows the benefit of lower energy consumption of close to 7.2% of proposed b-position under the same trajectory motion conditions (e.g., velocity, fly-by parameter, motion type). In Figure 7b, the power–time chart highlights the lower instantaneous request, within the 11.6–24.3% range, for the b-proposed position, guaranteeing task execution. Further scenarios have been carried out to study the analytical outcome for pick-and-place task execution using a robot with lower payload of maximum 3.0 kg. Table 4 reports the input/output data used for the procedure to fulfil the TCP positions. Case 3 and Case 4 differ for the specified robot placement on the z-axis, which is moved from 670.0 mm to 400.0 mm, respectively. The proposed b-position is placed for Case 3 at  $[X, Y] = [0.0 \text{ mm}, −100.0 \text{ mm}]$  and in Case 4 at  $[X, Y] = [0.0 \text{ mm}, −500.0 \text{ mm}]$ .

**Table 4.** Input variables, bounds, and parameters.

| Case 3—Input       |                          |                                                                     |                           |                                |                                  |                                   |
|--------------------|--------------------------|---------------------------------------------------------------------|---------------------------|--------------------------------|----------------------------------|-----------------------------------|
| Target             | Position<br>[X, Y, Z] mm | Rotation<br>[R <sub>X</sub> , R <sub>Y</sub> , R <sub>Z</sub> ] deg | Robot J1–J2<br>link mm    | Robot Base Height<br>mm        | z-Axis Robot Placement mm        | b-Manual<br>Position<br>[X, Y] mm |
| T <sub>Pick</sub>  | [0, 0.1, 1200]           | [0, 0, 0]                                                           | 448.0                     | 339.1                          | 670.0                            | [500, 500]                        |
| T <sub>Place</sub> | [0, 0.1, 1000]           | [0, 0, 0]                                                           |                           |                                |                                  |                                   |
| Case 3—Output      |                          |                                                                     |                           |                                |                                  |                                   |
| Target             | Position<br>[X, Y, Z] mm | Rotation<br>[R <sub>X</sub> , R <sub>Y</sub> , R <sub>Z</sub> ] deg | Robot Placement z-Axis mm | b-Manual<br>Position [X, Y] mm | b-Proposed<br>Position [X, Y] mm |                                   |
| T <sub>Pick</sub>  | [0, 0.1, 1200]           | [0, 0, 0]                                                           | 670.0                     | [500, 500]                     | [0, −100]                        |                                   |
| T <sub>Place</sub> | [0, 0.1, 1000]           | [0, 0, 0]                                                           |                           |                                |                                  |                                   |
| Case 4—Output      |                          |                                                                     |                           |                                |                                  |                                   |
| Target             | Position<br>[X, Y, Z] mm | Rotation<br>[R <sub>X</sub> , R <sub>Y</sub> , R <sub>Z</sub> ] deg | Robot Placement z-Axis mm | b-Manual<br>Position [X, Y] mm | b-Proposed<br>Position [X, Y] mm |                                   |
| T <sub>Pick</sub>  | [0, 0.1, 1200]           | [0, 0, 0]                                                           | 400.0                     | [500, 500]                     | [0, −500]                        |                                   |
| T <sub>Place</sub> | [0, 0.1, 1000]           | [0, 0, 0]                                                           |                           |                                |                                  |                                   |



**Figure 6.** Case 2—angular joint contribution based on the b-position of the robot placement on the  $\pi$ -plane:  $J_1$  contribution (a);  $J_2$  contribution (b);  $J_3$  contribution (c).

Figure 8a–f illustrates the proposed b-position of the robot placement on the XY plane, corresponding to the solution using an identical weighting factor for each DOF—here, the  $J_1$ ,  $J_2$ , and  $J_3$  joints of the robot configuration in Case 3 and Case 4. Compared to the previous Case 1 and Case 2, the impact of  $J_1$  and  $J_3$  are not predominant in either Case 3 or Case 4. The observed difference in  $[X, Y]$  between Case 3 and Case 4 in terms of robot location is  $[0.0 \text{ mm}, 400 \text{ mm}]$ . The joint perturbation on the b-placement estimation is an open challenge that needs to be addressed.

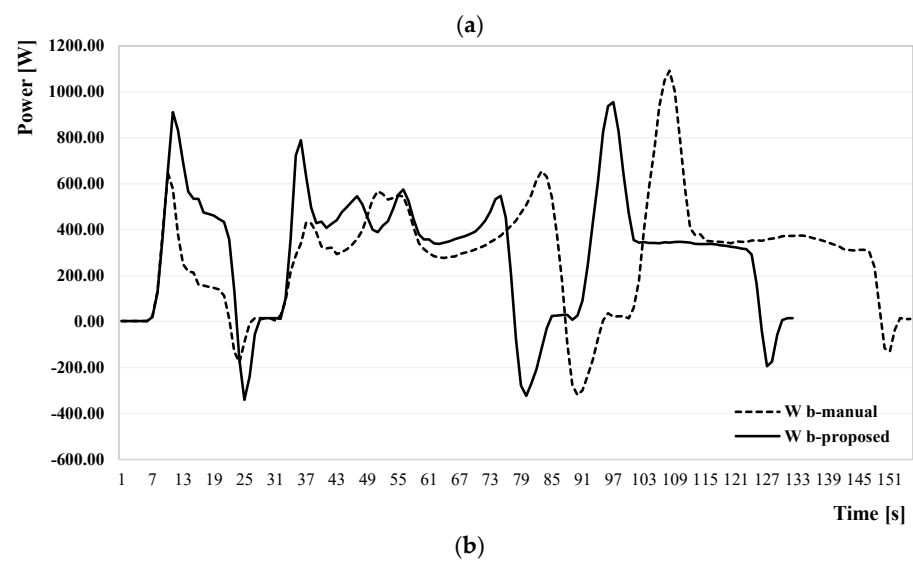
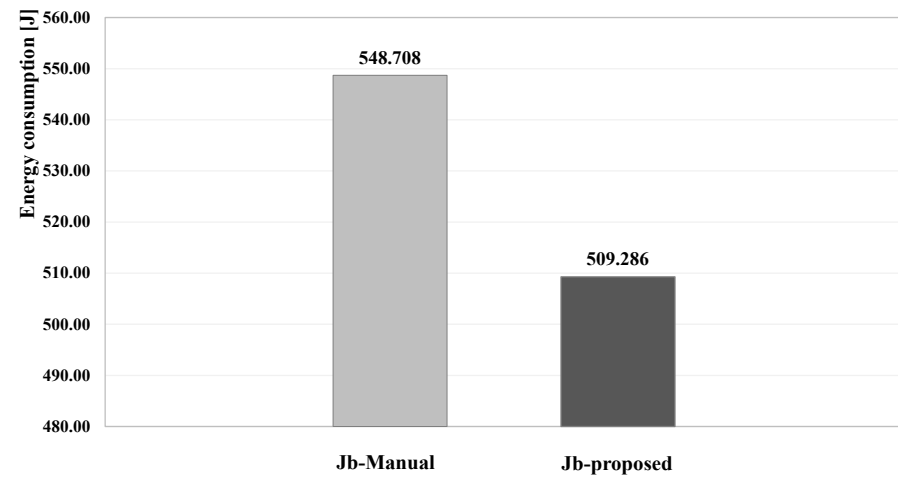


Figure 7. ABB IRB1200-5/0.9 robot: energy consumption bar chart (a); power-time chart (b).

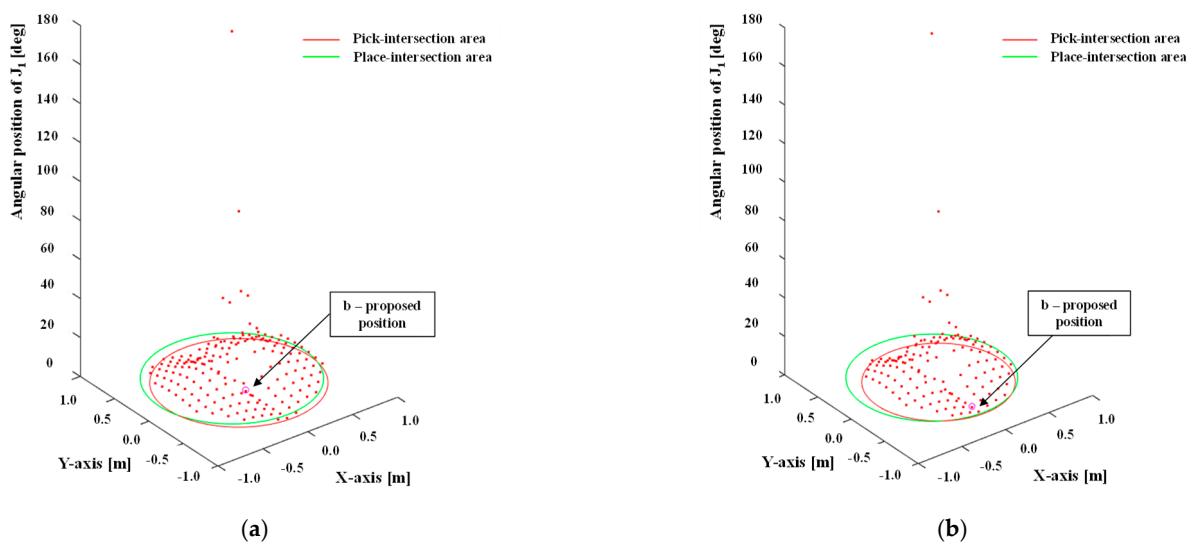
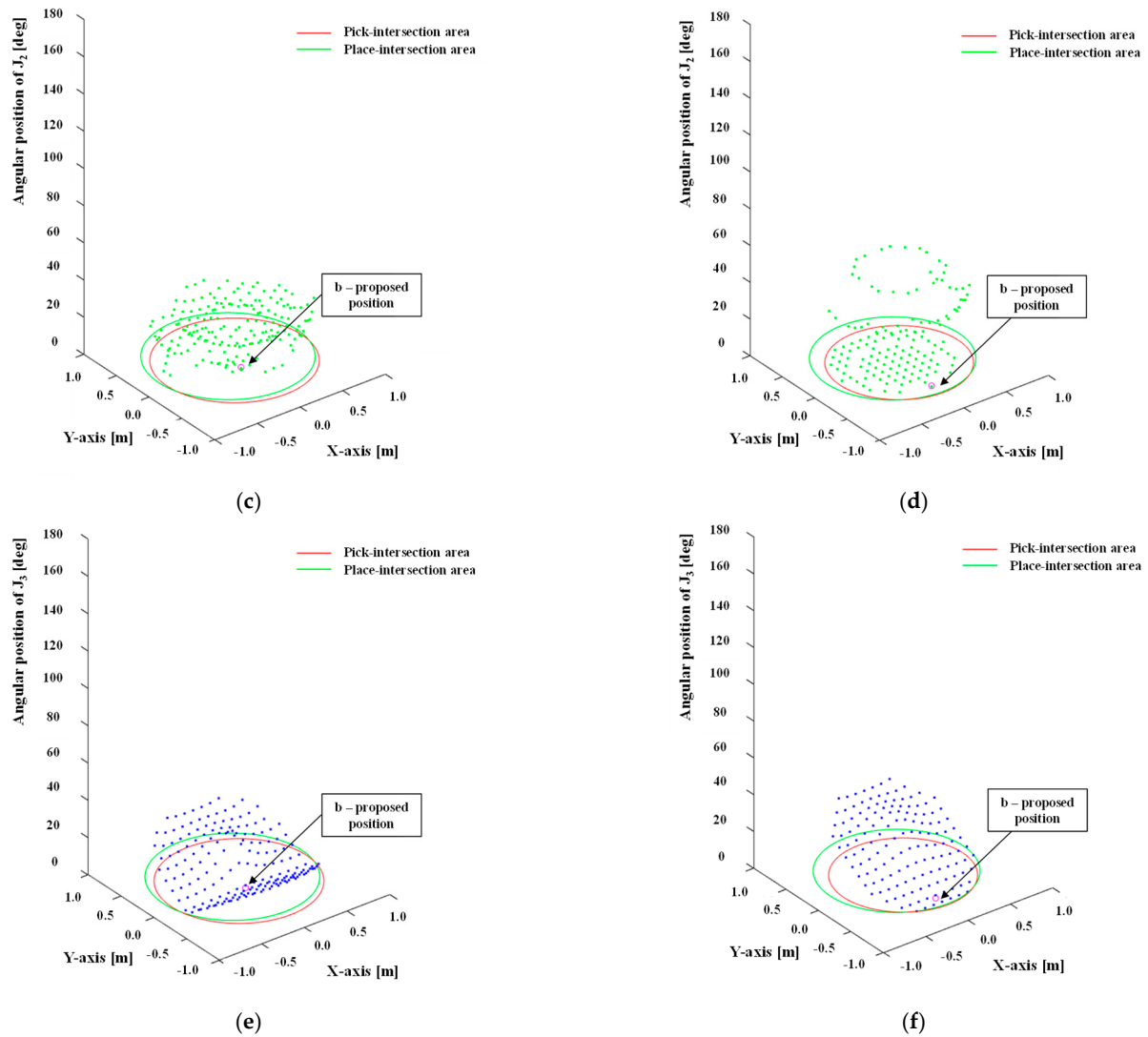


Figure 8. Cont.



**Figure 8.** Case 3 and Case 4 - angular joint contribution based on the b-position of the robot placement on the  $\pi$ -plane:  $J_1$  contribution Case 3 (a);  $J_1$  contribution Case 4 (b);  $J_2$  contribution Case 3 (c);  $J_2$  contribution Case 4 (d);  $J_3$  contribution Case 3 (e);  $J_3$  contribution Case 4 (f).

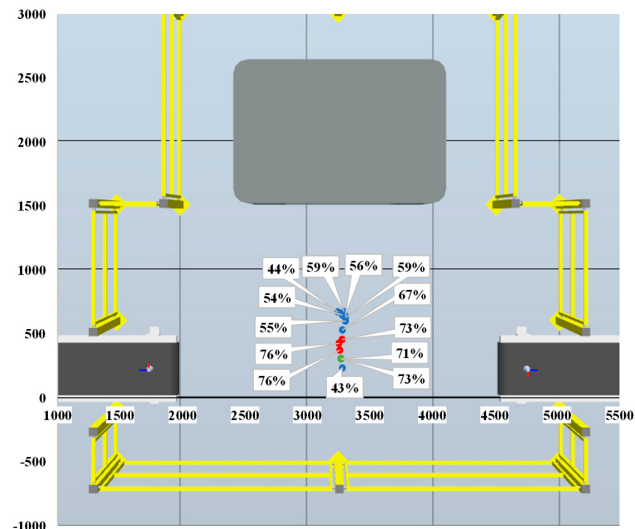
#### 4.2. Robot-to-Workpiece Placement: Simulation Method

To evaluate the procedure, a comparison with a digital twin tool was conducted, assuming prior knowledge of the model for a selected set of robots. The models were obtained with Windows PC with Intel-Core4.8GHz-i7. The industrial layout was represented with commercially I/O systems for machine-tending applications. The layout architecture and a set of workstation elements with their target positions (5 to 8 references) are described in Table 5. Fourteen ABB robots were analyzed, with reach characteristics in the 1440.0 mm–2500.0 mm range, using ABB RobotStudio software. The TCP velocity was set at 1.50 m/s. The digital twin SW may reproduce the motion of the robotic system, providing kinematic constraints and detecting the potential collisions.

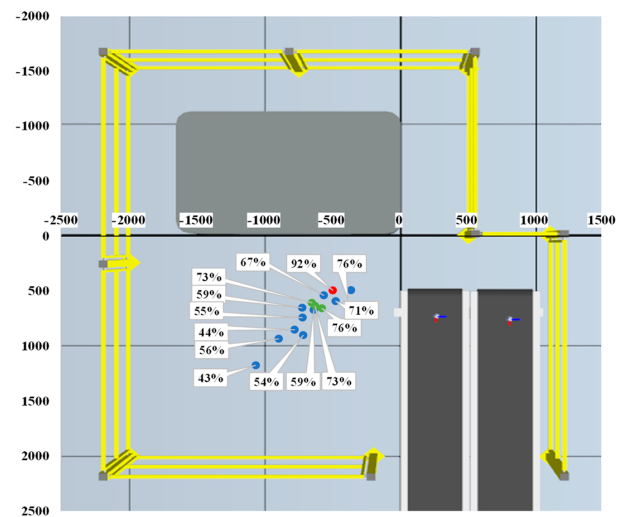
**Table 5.** Simulation scenario representation: layout, input/output equipment, and related task positions.

| Layout Type | I/O System             | Position [X mm, Y mm, Z mm] and Rotation [R <sub>x</sub> deg, R <sub>y</sub> deg, R <sub>z</sub> deg]                                                                |
|-------------|------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Linear      | Input system conveyor  | [1830, 250, 630] [−90, 0, 90]                                                                                                                                        |
|             | Output system conveyor | [4670, 250, 630] [−90, 0, −90]                                                                                                                                       |
|             | Workstation            | Target 1 [3250, 1750, 620] [−90, 0, 0]<br>Target 2 [2980, 1750, 750] [0, 90, 90]<br>Target 3 [3250, 1750, 750] [0, −90, −90]                                         |
| U-type      | Input system conveyor  | [670, 250, 430] [−90, 0, 0]                                                                                                                                          |
|             | Output system conveyor | [670, 750, 730] [−90, 0, 0]                                                                                                                                          |
|             | Workstation            | Target 1 [250, 750, 620] [−90, 0, 90]<br>Target 2 [−250, −1020, 750] [0, 90, −180]<br>Target 3 [−250, −480, 750] [0, −90, 0]                                         |
| U-type      | Input system conveyor  | [670, 1100, 730] [−90, 0, 0]                                                                                                                                         |
|             | Output system pallet   | Position 1 [370, 70, 270] [−90, 0, 0]<br>Position 2 [370, 730, 270] [−90, 0, 0]<br>Position 3 [1430, 70, 270] [−90, 0, 0]<br>Position 4 [1430, 730, 270] [−90, 0, 0] |
|             | Workstation            | Target 1 [−250, −750, 620] [−90, 0, 90]<br>Target 2 [−250, −1020, 750] [0, 90, −180]<br>Target 3 [−250, −480, 750] [0, −90, 0]                                       |

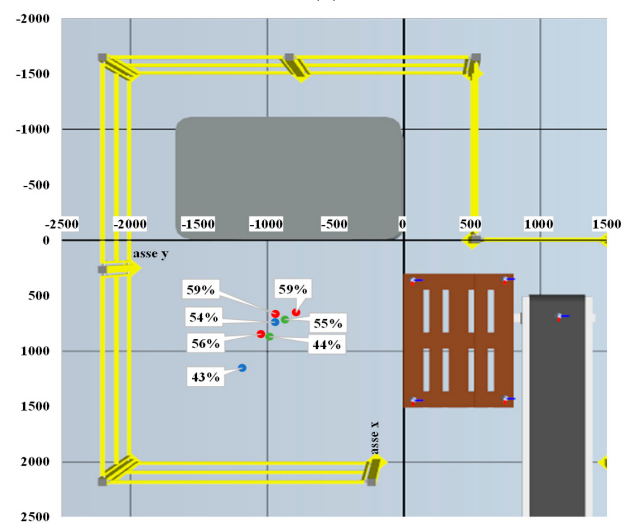
The simulation problem was presented with three variables, X, Y, and Z, of the b-placement position. The searching area was identified to reduce the computation time and to satisfy the reachability and manipulability criteria. In the considered scenarios, the maximum pick-and-place trajectory distance was selected as 1100.0 mm. The robot placement verifications in 2D-space were completed. The results are displayed in Figure 9a–c and the selected index was the ratio between the trajectory distance and the robot reach feature. The results for each layout architecture are shown in Figure 9a for linear—conveyor to conveyor, in Figure 9b for side—conveyor to conveyor, and in Figure 9c for side—conveyor to pallet. Six to fourteen verified robots are shown for the three layouts. The b-placement coordinate in blue color shows compliance with four criteria, the green color indicates medium risk for the reachability criteria, and the red color indicates high risk of not satisfying the four criteria. The difference between the digital twin and the proposed procedure was less than 5% in terms of energy consumption estimation for the same robot placement. The results of the scenarios in Table 5 show an energy consumption reduction from red positions (high risk) to blue positions (satisfy criteria) in the range of 24.5% to 26.4%. The benefit of appropriate robot selection and b-placement was reduced in 7.6–13.2% in the case of medium risk to satisfy the reachability and manipulability criteria. The goodness and effectiveness of the proposed method were verified in comparison with scenarios of high expertise in prior robots and time consumption for the maturity of a representative digital twin.



(a)



(b)



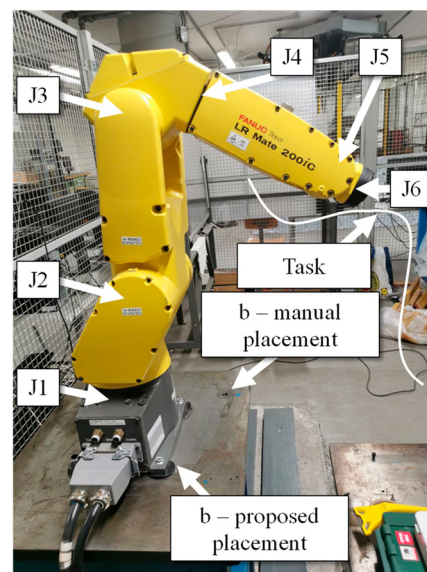
(c)

**Figure 9.** Normalized index of the 6-DOF robot base position on the XY plane in different scenarios using ABB RobotStudio: linear-conveyor to conveyor (a); side-conveyor to conveyor (b); side-conveyor to pallet (c).

## 5. Experimental Validation: Results and Discussion

A series of experimental verifications of the robot placement scheme were carried out on the industrial FANUC robot equipped with an external sensor for TCP position recognition and redundant system of power consumption measurement. The robot locations were obtained considering the unknown IK model and without a prior knowledge of joint dependency. Evaluating the optimized multi-criteria positioning results and the manual placement, the feasibility and validity of procedure were confirmed.

The configuration of robotic system is shown in Figure 10, and an LR Mate 200iC was employed for the test campaign. The specification of the robot arm can be stated: number of axes, 6; maximum robot payload, 5.0 kg; maximum reach, 704.0 mm; repeatability,  $\pm 0.02$  mm; and an R-30iA controller. In addition, the joint maximum speed and range of motion are stated in Table 6.



**Figure 10.** Testing area and layout setup using the FANUC LR Mate 200iC robot.

**Table 6.** FANUC LR Mate 200iC speed, mechanical range of motion, selected range of motion.

| Joint | Maximum Velocity<br>[deg/s] | Range of Motion<br>[deg] | Selected Range of<br>Motion [deg] |
|-------|-----------------------------|--------------------------|-----------------------------------|
| 1     | 350                         | $\pm 340$                | $\pm 80$                          |
| 2     | 350                         | $\pm 200$                | $\pm 40$                          |
| 3     | 400                         | $\pm 388$                | $\pm 65$                          |
| 4     | 450                         | $\pm 380$                | $\pm 65$                          |
| 5     | 450                         | $\pm 240$                | $\pm 45$                          |
| 6     | 720                         | $\pm 720$                | $\pm 65$                          |

The procedure for the robot base location may consider the optimization of the 6-DoF system for every configuration in the 3D workspace. The initial decision was to configure the robot on the ground. In the selected testing scenario, the second and third joints were not independent, due to the robot architecture needing to be managed via prior knowledge. For that reason, the proposed procedure, which considers the robot kinematic model as a black box, starts with a data collection of 11 target points for each joint equally distributed in the range of motion. In Table 6, the searching space is bound within the limited number of target points to lower the computation time.

These values are used to determine the direct kinematic identified by the product of the  $R_{x,y,z}$  rotation matrixes with the end effector translation matrix  $M_t$ , as in Equations (18)–(22):

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = R_z \cdot R_y \cdot R_x \cdot M_t \quad (18)$$

$$R_x = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(w) & -\sin(w) & 0 \\ 0 & \sin(w) & \cos(w) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (19)$$

$$R_y = \begin{bmatrix} \cos(p) & 0 & \sin(p) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(p) & 0 & \cos(p) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (20)$$

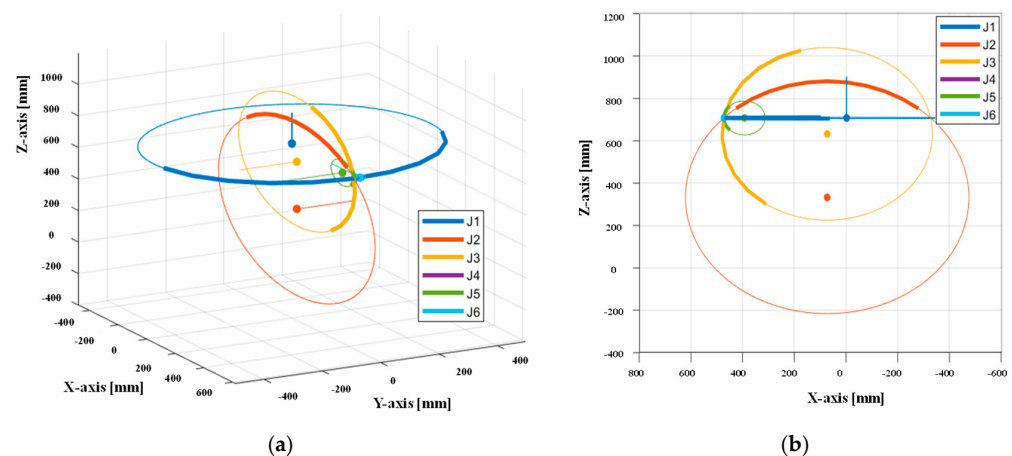
$$R_z = \begin{bmatrix} \cos(r) & \sin(r) & 0 & 0 \\ -\sin(r) & \cos(r) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (21)$$

$$M_t = \begin{bmatrix} 1 & 0 & 0 & TCP_x \\ 0 & 1 & 0 & TCP_y \\ 0 & 0 & 1 & TCP_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (22)$$

The first result of the proposed modeling was the DH parameter of an unknown system, as shown in Table 7; the 3D representation of joint space, as shown in Figure 11a; and the XZ plane joint space, as represented in Figure 11b.

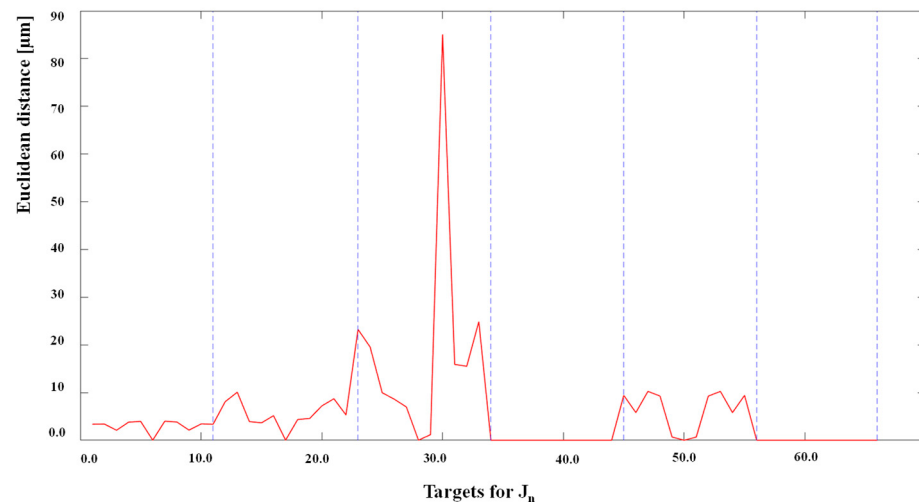
**Table 7.** D-H parameters for the FANUC LR Mate 200iC.

| Joint | $\theta_i$ [rad] | $d_i$ [mm]             | $a_i$ [mm]             | $\alpha_i$ [rad] |
|-------|------------------|------------------------|------------------------|------------------|
| 1     | 0.0              | $3.30 \times 10^{+02}$ | $7.50 \times 10^{+01}$ | $\pi/2$          |
| 2     | $\pi/2$          | 0.00                   | $3.00 \times 10^{+02}$ | 0.0              |
| 3     | 0.0              | 0.00                   | $7.50 \times 10^{+01}$ | $\pi/2$          |
| 4     | 0.0              | $3.20 \times 10^{+02}$ | 0.00                   | $-\pi/2$         |
| 5     | 0.0              | 0.00                   | 0.00                   | $\pi/2$          |
| 6     | 0.0              | $1.40 \times 10^{+02}$ | 0.00                   | 0.0              |



**Figure 11.** Data-driven representation of joint space for the robot under investigation: 3D scheme (a); XZ plane (b).

Then, the direct kinematic was tested on the 11 measured target points using the parameters from the automatic generation of DH with low data input. Figure 12 shows the Euclidean error distance of the J1, J2, J3, and J5 joints. The maximum error was at J3 (dependent joint to J2), with a magnitude of 0.875 mm, while the average contribution of J1, J2, and J5 was lower than 18.2  $\mu\text{m}$ . The direct kinematic was obtained properly with the low-pose time allocation considering the robotic system as a black box.



**Figure 12.** Error obtained with Euclidean distance from the measured to DH estimated targets.

Therefore, a set of three trajectories was placed in the reachable workspace, as shown in Figure 13, for the point-to-point task, with each trajectory including eight discrete target points (red color). The robot base placement was marked at the Cartesian origin (green color). The task starting point was set as [0.0 mm; −517.0 mm; 517.0 mm; 0.0 mm; 0.0 mm; 0.0 mm].

Table 8 shows the direct kinematic results of automatic DH generation for the prescribed task. The RMSE indicator was selected to evaluate the DK error. The RMSE showed a value of lower than 65.0  $\mu\text{m}$  in point-to-point positions.

**Table 8.** Data-driven direct kinematic RMSE error for a set of trajectories.

| Trajectory | RMSE [mm] | Min–Max RMSE [mm] |
|------------|-----------|-------------------|
| Circle     | 0.059     | 0.037–0.076       |
| S-shaped   | 0.061     | 0.049–0.077       |
| Rectangle  | 0.057     | 0.046–0.072       |

Consequently, Algorithm 1 was used to determine the inverse kinematic model to establish the robot base position with the ANN technique configured by the GA method.

---

**Algorithm 1: Determining the base position to satisfy the multi-criteria boundaries**

---

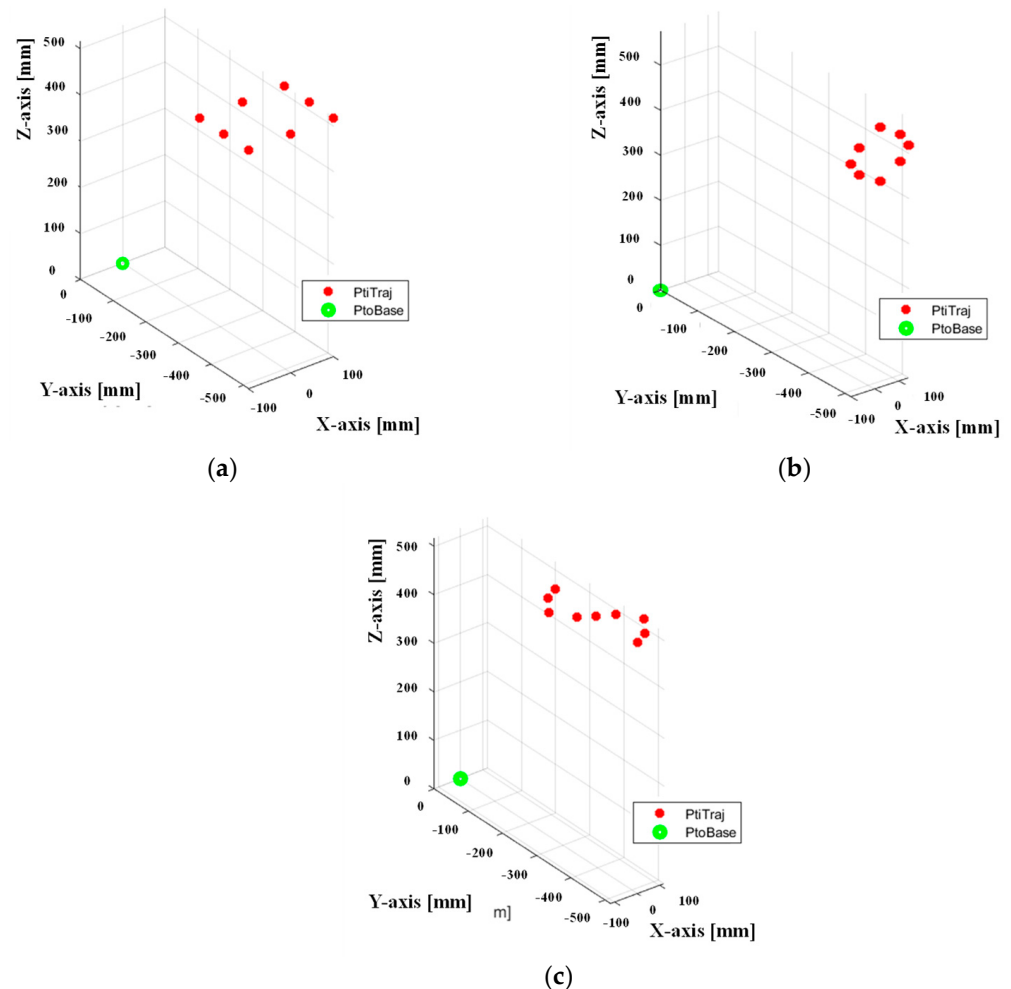
1. Trajectories measured with the real LR-Mate 200ic robot are used.
  2. Let  $\mathbf{J}$  be the joint's vector and  $\mathbf{P}$  be the Cartesian TCP vector of the trajectories.
  3. The direct kinematic model automatically acquired is applied to  $\mathbf{J}$ , obtaining  $\mathbf{P}_{DH}$ .
  4. The resultant inverse kinematic model is established for  $\mathbf{P}_{DH}$ , obtaining  $\mathbf{J}_{ANN}$ .
  5. The direct kinematic model is applied to  $\mathbf{J}_{ANN}$ , obtaining  $\mathbf{P}_{ANN}$ .
  6. The error is stated, comparing  $\mathbf{P}_{DH}$  and  $\mathbf{P}_{ANN}$ .
  7. The  $\mathbf{J}_{ANN}$  is adopted to verify the multicriteria boundaries.
-

Algorithm 1 minimizes the dependency of the inverse kinematic model from the direct kinematic model. To evaluate Algorithm 1 in the proposed procedure, it was estimated in a reduced workspace section, as in Table 9, to improve the ANN training computation efficacy and to limit redundant configurations. Similarly, the assumptions to neglect J4 and J6 were based on avoiding specific singularities in the present study.

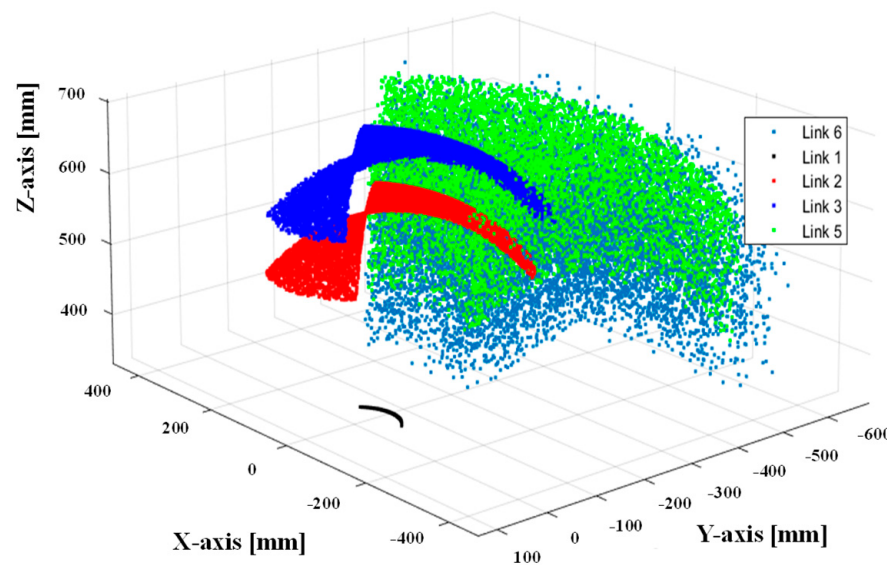
**Table 9.** Joint vector utilized to train the ANN network.

| Joint | Acquisition Range of Motion [deg] |
|-------|-----------------------------------|
| 1     | −135; −45                         |
| 2     | −45; +45                          |
| 3     | −30; 0                            |
| 4     | 0; 0                              |
| 5     | −115; 0                           |
| 6     | 0; 0                              |

Figure 14 shows the workspace resultant from the automatic DH analysis used as the training dataset for joints J1, J2, J3, J5, and J6. The dataset was composed of 10,000 points, with 75% used for training, 15% for testing, and 15% for ANN model validation.



**Figure 13.** Point-to-point for task Ti: rectangle trajectory (a); circle trajectory (b); S-shape trajectory (c).



**Figure 14.** Reduced workspace from the DK model to train the ANN network.

The model was completed using the ANN algorithm [36–41]. In the present work, four neural networks were produced sequentially from the robot base, as shown in Table 10. The neural networks were configured employing the genetic algorithm (GA) [42,43]. The GA method for the selected number of ANN hidden layers determines how many neurons and which activation function provide the best results. The number of hidden layers was in the range of three to eight for each network. The number of neurons for each layer varied from 15 to 210, and the activation functions set was represented by purelin or tansig.

**Table 10.** ANN sequential network identification number, input, and output.

| ID | Input                           | Output |
|----|---------------------------------|--------|
| 1  | x, y, z, Rx, Ry, Rz             | J1     |
| 2  | x, y, z, Rx, Ry, Rz, J1         | J2     |
| 3  | x, y, z, Rx, Ry, Rz, J1, J2     | J3     |
| 4  | x, y, z, Rx, Ry, Rz, J1, J2, J3 | J5     |

The loss function was formulated as follows in Equation (23):

$$f = (0.8 \cdot RMSE_v + 0.2 \cdot RMSE_t) + (MaxE_v(1 - ABS(Rsq_v))) \quad (23)$$

In Equation (23), the RMSE is the root mean square error, the MaxE is the maximum error, and Rsq is the coefficient of determination. The v subscript indicates the validation appraisal, and the t subscript specifies the testing part of the dataset. The resultant ANN sequential architecture configuration is reported in Table 11.

The PC used for the model computation has an Intel Core i5-8500 processor with 16 GB RAM. The GA technique was applied with the following parameters for the ANN generation: the “scaled conjugate gradient” was used as a training function; the loss function corresponded to the “mean squared error”, and the calculations were executed with the MatLab parallel computation to reduce the required optimization’s computational time. In Table 12, the main inputs for the generated models are presented. In particular, the adopted technique allowed the optimized mix network to obtain the best configuration of the number of layers and number of neurons for each joint.

**Table 11.** The resultant ANN sequential architecture for LR MATE 200-ic.

| ID | Output | Configuration |
|----|--------|---------------|
| 1  | J1     |               |
| 2  | J2     |               |
| 3  | J3     |               |
| 4  | J5     |               |

**Table 12.** GA/ANN optimization inputs for the FANUC LR Mate 200iC.

| Requirements          | Sequential |
|-----------------------|------------|
| Number of joints      | J1–J3      |
| Algorithm limitations | Mix        |
| Activation functions  | Tansig     |
| Epochs                | 1000       |

The settings showed an offline computational time of close to 70 h. This influence was neglected in the present work due to its offline nature. The final network architecture for the studied robot is expressed from base to TCP reference as follows:

$$\begin{aligned}
 J_1 &\{50 \ 45 \ 50 \ 30 \ 45 \ 40 \ 15\} \\
 J_2 &\{55 \ 60 \ 35 \ 35\} \\
 J_3 &\{50 \ 40 \ 10\}
 \end{aligned}$$

On the other hand, considering the network architecture from TCP reference to base frame, the ANN structure is:

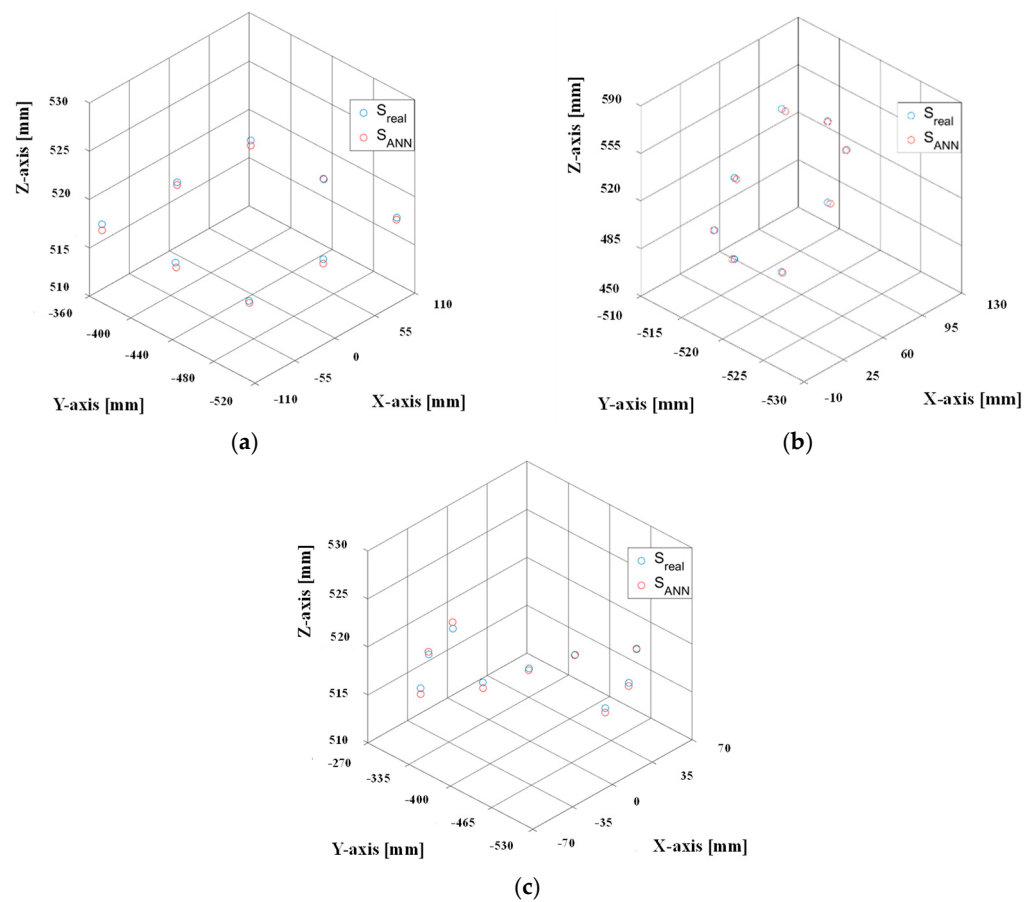
$$\begin{aligned}
 J_1 &\{50 \ 45 \ 55 \ 15\} \\
 J_2 &\{35 \ 45 \ 70 \ 35 \ 65 \ 25 \ 20\} \\
 J_3 &\{60 \ 60 \ 30 \ 15\}
 \end{aligned}$$

Figure 15 shows the positions calculated using a DK model applied to real joints. At the same time, the positions generated by the joints obtained through the ANN are shown.

Table 13 presents the RMSE error and the minimum and maximum error range of the experimental and the IK-ANN-derived trajectory comparison for a set of simple paths, rectangle, circle, and S-shaped. The deviation magnitude was close to the standard robot repeatability.

**Table 13.** Data-driven inverse-kinematic RMSE error for a set of trajectories.

| Trajectory | RMSE [mm] | Min–Max RMSE [mm] |
|------------|-----------|-------------------|
| Circle     | 0.567     | 0.338–0.701       |
| S-shaped   | 0.702     | 0.358–1.060       |
| Rectangle  | 0.546     | 0.288–0.803       |



**Figure 15.** Comparison of  $P_{DH}$  and  $P_{ANN}$  for the point-to-point of task  $T^1$ : rectangle trajectory (a); circle trajectory (b); S-shaped trajectory (c).

Finally, the minimum energetic robot contribution, based on a generated model for a prescribed task, considering the multicriteria boundaries, was computed for a set of base placements. Three campaigns were performed, as summarized in Table 14: two pick-and-place tasks and one path-following task. The task comprised 20 repetitions for a specific value of TCP velocity and payload, as stated: [50%; 50%], [70%; 20%], and [35%; 70%].

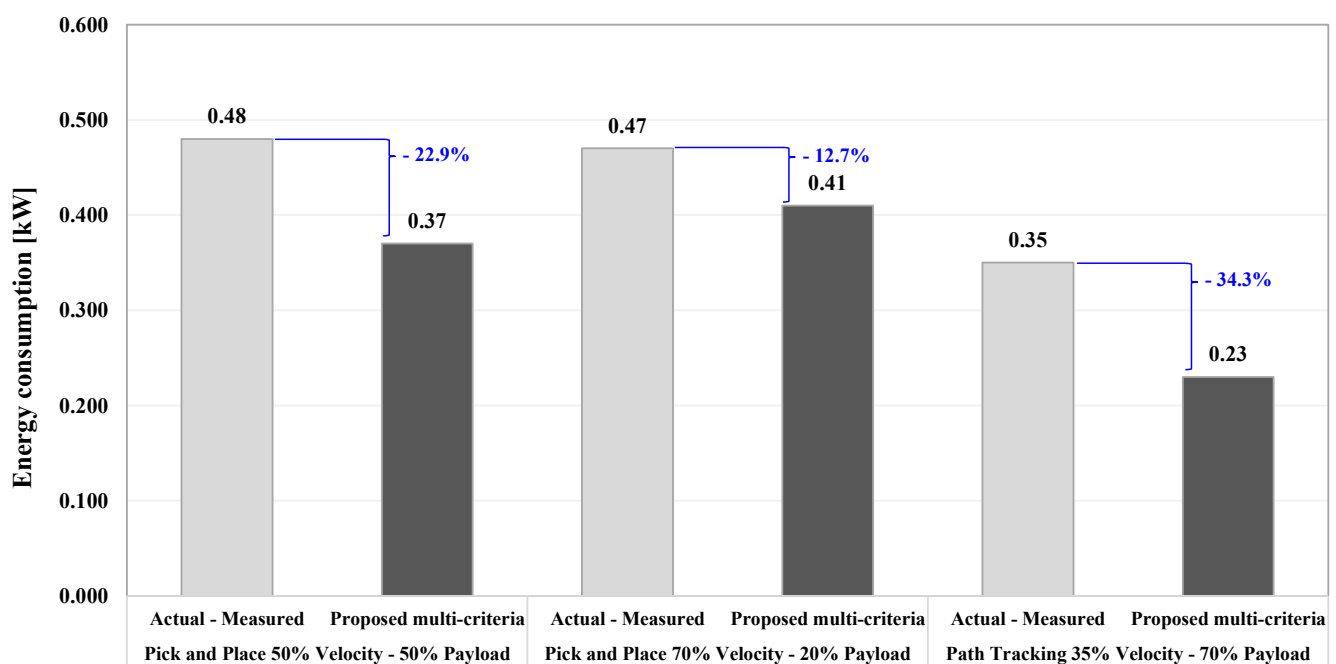
To demonstrate the compliance of the proposed procedure, the comparison between the actual measured position and the actual computed position showed a mean absolute error in the 0.370–0.699 mm range in the trajectory following <300.0 mm paths, with a consequent energy consumption estimation error in the 6.25–10.64% range, confirming the suitable representation of the scenario under investigation. The worst performance in terms of trajectory tracking was assessed for high-velocity pick-and-place tasks. Meanwhile, the path following task at low velocity and medium payload showed a lower error (representing welding application), and the low-velocity setting with light payload presented an error of close to 8.57%. Finally, Figure 16 shows the energy consumption measured during the experimental trials compared with the energy consumption of the proposed optimal robot base position.

The optimal robot base location via a multi-criteria procedure confirmed a significant energy consumption reduction for the three applications. The high-speed pick-and-place tasks showed a reduction in energy consumption in the 12.7–22.9% range, and the slow-motion path tracking presented an energy savings of close to 34.3%. These results highlight the significant impact of the proposed procedure in welding, deburring, gluing, and part assembly applications, where the cycle time is not of primary importance and the path-following accuracy is significant in combination with medium–high robot payload. Nevertheless, the high-speed result confirms the lower energy savings in single task com-

pletion, but, considering the hourly usage in daily/monthly working in the application, the cumulative reduction is significant. Consequently, it can be established that the proposed procedure helps to reduce workstation setup time and save energy during the industrial usage of robots. The demonstrated results confirm the proposed procedure's scalability and replicability in industrial scenarios, thus facilitating and supporting users and technicians in speed and acceleration optimization, improving trajectory programming and scheduling, and achieving benefits similar to commercial proprietary software in avoiding unnecessary movements and reducing cycle time. In addition, providing prior knowledge on the energy consumption of the equipment could foster an appropriate maintenance program culture that comprises regular checks and replacements of worn parts, cleaning moving components, and testing degraded sub-assemblies. Future investigations will consider the effect of the temperature gradient of robot self-heating or joint degradation on the procedure accuracy or concurrent contributions. In particular, operations criteria (e.g., maintainability), sensors (e.g., thermocouples, accelerometers), and signal features able to perceive the surrounding environment will be included to reduce the error in the procedure and the workstation offline setup time.

**Table 14.** Energy consumption: LR MATE 200-ic and multi-criteria procedure performance.

| Task                                       | Robot Basement Placement | Energy Consumption kW |
|--------------------------------------------|--------------------------|-----------------------|
| Pick-and-place<br>50% Velocity—50% payload | Actual—measured          | 0.480                 |
|                                            | Actual—computed          | 0.450                 |
|                                            | Proposed multi-criteria  | 0.370                 |
| Pick-and-place<br>70% Velocity—20% payload | Actual—measured          | 0.470                 |
|                                            | Actual—computed          | 0.420                 |
|                                            | Proposed multi-criteria  | 0.410                 |
| Path tracking<br>35% Velocity—70% payload  | Actual—measured          | 0.350                 |
|                                            | Actual—computed          | 0.320                 |
|                                            | Proposed multi-criteria  | 0.230                 |



**Figure 16.** Energy observed in the experimental trials: actual position and multi-criteria position location.

In the current application, the computation time was obtained in less than 10 min of modeling via Algorithm 1 searching and 30 min of DK position setup. This method provides a robot base location that satisfies user task reachability, avoids singular configuration, maximizes manipulability, defines data-driven IK modeling, and minimizes energy consumption.

## 6. Conclusions

This paper proposes a procedure to locate the robot base to realize a prescribed task within the robot's workspace, complying with multiple criteria for industrial applications. The robot base location includes verification of the reachable target points, evaluation of the joint motions, maximization of manipulability, avoidance of singularities, and minimization of the energy consumption. The study focuses on the development of a general hybrid procedure that comprises analytical, physical-based, and data-driven modeling to solve the optimization problem. The approach was compared with state-of-the-art research practices showing advanced abilities.

Firstly, the analytical formulation was verified using anthropomorphic robots considering different levels of a priori kinematics and dynamics knowledge of the system. The results for small and medium-sized robot cases show a compliant base position that provides 7.27–24.3% saving of energy consumption in the manipulation of payloads of less than 8.0 kg compared with manual position selection. Then, six to fourteen robot locations were investigated in three layouts: (i) linear—conveyor to conveyor, (ii) side—conveyor to conveyor, and (iii) side—conveyor to pallet, comparing the digital twin commercial software with the proposed procedure. The observed difference was lower than 5% in terms of energy consumption estimation for the same robot placement. The outcomes in terms of energy consumption reduction from high risk and not-compliant criteria was in the range of 24.5% to 26.4%. The benefit was appraised in the 7.6–13.2% range in case of medium risk to satisfy the reachability and manipulability criteria. Nevertheless, the high expertise in terms of the prior robot model and time-consuming activity for the development of a representative digital twin model is still needed.

In addition, this work also explored the unknown a priori kinematics and dynamics situation. In particular, the unknown joint dependency in a real test setup was investigated. Experiments were conducted on a FANUC robot in pick-and-place and contouring tasks. The proposed scheme led the operator to obtain the direct kinematic model via digital twin software or using a real robot to provide the input for the appraisal of the data-driven IK model based on the artificial neural network technique tuned via a genetic algorithm. From 11 target points, for each active joint, a direct kinematic model was appraised with an automatic DH scheme that generated the 3D workspace with an RMSE of less than 65.0  $\mu\text{m}$  in point-to-point positions. Then, the inverse kinematic model was computed using an ANN technique tuned with a genetic algorithm, and the number of hidden layers was in the three-to-eight range, the number of neurons for each layer varied from 15 to 210, and the activation functions set were represented by purelin or tansig, showing an RMSE in the S-shaped task of close to 702.0  $\mu\text{m}$ . Finally, three experimental campaigns were performed with a set of tasks, repetitions, end-effector velocity, and payloads. The energy consumption reduction was observed for the pick-and-place application in the 12.7–22.9% range and for the slow-motion task close to 34.3%. Consequently, it can be established that the proposed procedure helps to reduce the workstation setup time and save energy during the industrial usage of robots.

**Author Contributions:** All the authors contributed equally. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no external funding.

**Data Availability Statement:** The dataset is available on request from the authors.

**Conflicts of Interest:** The authors declare that they have no known competing financial interest or personal relationship that could have appeared to influence the work reported in this paper.

## References

- Li, G.; Zhu, W.; Dong, H.; Ke, Y. A method for robot placement optimization based on two-dimensional manifold in joint space. *Robot. Comput. Integr. Manuf.* **2021**, *67*, 102002. [\[CrossRef\]](#)
- Yang, J.; Wang, D.; Fan, B.; Dong, D.; Zhou, W. Online absolute pose compensation and steering control of industrial robot based on six degrees of freedom laser measurement. *Opt. Eng.* **2017**, *56*, 9. [\[CrossRef\]](#)
- Santolaria, J.; Yagüe, J.-A.; Jiménez, R.; Aguilar, J.-J. Calibration-based thermal error model for articulated arm coordinate measuring machines. *Precis. Eng.* **2009**, *33*, 476–485. [\[CrossRef\]](#)
- Wu, J.H.; Tao, Y.R. Review on Research Status of Positioning Accuracy Reliability of Industrial Robots. *China Mech. Eng.* **2020**, *31*, 2180–2188.
- Wang, X.W.; Zhou, X.; Xia, Z.L.; Gu, X.S. A survey of welding robot intelligent path optimization. *J. Manuf. Process.* **2021**, *63*, 14–23. [\[CrossRef\]](#)
- Votion, J.; Cao, Y.C. Diversity-Based Cooperative Multivehicle Path Planning for Risk Management in Costmap Environments. *IEEE Trans. Ind. Electron.* **2019**, *66*, 6117–6127. [\[CrossRef\]](#)
- Orozco-Rosas, U.; Montiel, O.; Sepulveda, R. Pseudo-bacterial Potential Field Based Path Planner for Autonomous Mobile Robot Navigation. *Int. J. Adv. Robot. Syst.* **2015**, *12*, 81. [\[CrossRef\]](#)
- Wang, B.Y.; Liu, Z.; Li, Q.B.; Prorok, A. Mobile Robot Path Planning in Dynamic Environments Through Globally Guided Reinforcement Learning. *IEEE Robot. Autom. Lett.* **2020**, *5*, 6932–6939. [\[CrossRef\]](#)
- Sartoretti, G.; Kerr, J.; Shi, Y.; Wagner, G.; Kumar, T.S.; Koenig, S.; Choset, H. Primal: Pathfinding via reinforcement and imitation multiagent learning. *IEEE Robot. Autom. Lett.* **2019**, *4*, 2378–2385. [\[CrossRef\]](#)
- Li, Q.; Gama, F.; Ribeiro, A.; Prorok, A. Graph neural networks for decentralized multi-robot path planning. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems, Las Vegas, NV, USA, 25–29 October 2020.
- Kaczmarek, W.; Borys, S.; Panasiuk, J.; Siwek, M.; Prusaczyk, P. Experimental Study of the Vibrations of a Roller Shutter Gripper. *Appl. Sci.* **2022**, *12*, 9996. [\[CrossRef\]](#)
- Bucinskas, V.; Dziedzickis, A.; Sumanas, M.; Sutins, E.; Petkevicius, S.; Butkiene, J.; Virzonis, D.; Morkvenaite-Vilkonciene, I. Improving Industrial Robot Positioning Accuracy to the Microscale Using Machine Learning Method. *Machines* **2022**, *10*, 940. [\[CrossRef\]](#)
- Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A.A.; Veness, J.; Bellemare, M.G.; Graves, A.; Riedmiller, M.; Fidjeland, A.K.; Ostrovski, G.; et al. Human-level control through deep reinforcement learning. *Nature* **2015**, *518*, 529–533. [\[CrossRef\]](#) [\[PubMed\]](#)
- Sutton, R.S.; Barto, A.G. *Introduction to Reinforcement Learning*; MIT Press: Cambridge, MA, USA, 2018.
- Pan, X.; Wang, W.; Zhang, X.; Li, B.; Yi, J.; Song, D. How you act tells a lot: Privacy-leaking attack on deep reinforcement learning. In Proceedings of the International Conference on Autonomous Agents and Multi-Agent Systems, Montreal, QC, Canada, 13–17 May 2019; pp. 368–376.
- Aspragathos, N.A.; Foussias, S. Optimal location of a robot path when considering velocity performance. *Robotica* **2002**, *20*, 139–147. [\[CrossRef\]](#)
- Li, G.; Zhu, W.D.; Dong, H.; Ke, Y. Error compensation based on surface reconstruction for industrial robot on two-dimensional manifold. *Ind. Robot* **2022**, *49*, 735–744. [\[CrossRef\]](#)
- Ur-Rehman, R.; Caro, S.; Chablat, D.; Wenger, P. Multi-objective path placement optimization of parallel kinematics machines based on energy consumption, shaking forces and maximum actuator torques: Application to the Orthoglide. *Mech. Mach. Theory* **2010**, *45*, 1125–1141. [\[CrossRef\]](#)
- Gürel, S.; Gultekin, H.; Emiroglu, N. Scheduling a dual gripper material handling robot with energy considerations. *J. Manuf. Syst.* **2023**, *67*, 265–280. [\[CrossRef\]](#)
- Ghungrad, S.; Mohammed, A.; Haghighi, A. Energy-efficient and quality-aware part placement in robotic additive manufacturing. *J. Manuf. Syst.* **2023**, *68*, 644–650. [\[CrossRef\]](#)
- Doan, N.C.N.; Lin, W. Optimal robot placement with consideration of redundancy problem for wrist-partitioned 6R articulated robots. *Robot. Comput. Integr. Manuf.* **2017**, *48*, 233–242. [\[CrossRef\]](#)
- Son, S.W.; Kwon, D.S. A convex programming approach to the base placement of a 6-DOF articulated robot with a spherical wrist. *Int. J. Adv. Manuf. Technol.* **2019**, *102*, 3135–3150. [\[CrossRef\]](#)
- Ren, S.; Xie, Y.; Yang, X.; Xu, J.; Wang, G.; Chen, K. A method for optimizing the base position of mobile painting manipulators. *IEEE Trans. Autom. Sci. Eng.* **2017**, *14*, 370–375. [\[CrossRef\]](#)
- Yu, Q.; Wang, G.; Hua, X.; Zhang, S.; Song, L.; Zhang, J.; Chen, K. Base position optimization for mobile painting robot manipulators with multiple constraints. *Robot Comput. Integr. Manuf.* **2018**, *54*, 56–64. [\[CrossRef\]](#)
- Andulkar, M.V.; Chiddarwar, S.S.; Marathe, A.S. Novel integrated offline trajectory generation approach for robot assisted spray painting operation. *J. Manuf. Syst.* **2015**, *37 Pt 1*, 201–216. [\[CrossRef\]](#)
- Wang, S.; Xu, Z.; Wu, C.; Hua, L.; Zhu, D. Towards region-based robotic machining system from perspective of intelligent manufacturing: A technology framework with case study. *J. Manuf. Syst.* **2023**, *70*, 451–463. [\[CrossRef\]](#)

27. Spensieri, D.; Carlson, J.S.; Bohlin, R.; Kressin, J.; Shi, J. Optimal robot placement for tasks execution. *Procedia CIRP* **2016**, *44*, 395–400. [[CrossRef](#)]
28. Malhan, R.K.; Kabir, A.M.; Shah, B.; Gupta, S.K. Identifying feasible workpiece placement with respect to redundant manipulator for complex manufacturing tasks. In Proceedings of the 2019 International Conference on Robotics and Automation (ICRA), Montreal, QC, Canada, 20–24 May 2019; pp. 5585–5591.
29. Available online: <https://www.iso.org/committee/5915511.html> (accessed on 10 October 2024).
30. Available online: <https://www.iso.org/standard/51330.html> (accessed on 10 October 2024).
31. Available online: <https://www.iso.org/standard/41571.html> (accessed on 10 October 2024).
32. Available online: <https://webstore.ansi.org/standards/ria/riatrr155062014> (accessed on 10 October 2024).
33. Tagliani, F.L.; Pellegrini, N.; Aggogeri, F. Machine Learning Sequential Methodology for Robot Inverse Kinematic Modeling. *Appl. Sci.* **2022**, *12*, 9417. [[CrossRef](#)]
34. Faria, C.; Vilaça, J.L.; Monteiro, S.; Erlhagen, W.; Bicho, E. Automatic Denavit-Hartenberg Parameter Identification for Serial Manipulators. In Proceedings of the IECON 2019—45th Annual Conference of the IEEE Industrial Electronics Society, Lisbon, Portugal, 14–17 October 2019; Volume 1, pp. 610–617.
35. Aggogeri, F.; Pellegrini, N.; Taesi, C.; Tagliani, F.L. Inverse kinematic solver based on machine learning sequential procedure for robotic applications. In Proceedings of the 2021 International Symposium on Intelligent Robotics and Systems (ISoIRS 2021), Online, 7–9 November 2022; Volume 2234.
36. Zhu, Z.; Liu, Y.; He, Y.; Wu, W.; Wang, H.; Huang, C.; Ye, B. Fuzzy PID Control of the Three-Degree-of-Freedom Parallel Mechanism Based on Genetic Algorithm. *Appl. Sci.* **2022**, *12*, 11128. [[CrossRef](#)]
37. Teodoro, I.-P. *A Novel Optimization Robust Design of Artificial Neural Networks to Solve the Inverse Kinematics of a Manipulator of 6 DOF*; IEEE: New York, NY, USA, 2021; pp. 838–843.
38. Toquica, J.S.; Oliveira, P.S.; Souza, W.S.; Motta, J.M.S.; Borges, D.L. An analytical and a deep learning model for solving the inverse kinematic problem of an industrial parallel robot. *Comput. Ind. Eng.* **2020**, *151*, 106682. [[CrossRef](#)]
39. Derya, S. A comparison of optimization algorithms for deep learning. *Int. J. Pattern Recognit. Artif. Intell.* **2020**, *34*, 51–65.
40. Nektarios, A.; Aspragathos, N.A. Optimal location of a general position and orientation end-effector's path relative to manipulator's base, considering velocity performance, Robot. *Comput.-Integr. Manuf.* **2010**, *26*, 162–173. [[CrossRef](#)]
41. Soylak, M.; Oktay, T.; Turkmen, İ. A simulation-based method using artificial neural networks for solving the inverse kinematic problem of articulated robots. *Proc. Inst. Mech. Eng. Part E J. Process Mech. Eng.* **2017**, *231*, 470–479. [[CrossRef](#)]
42. Azimirad, V.; Shorakaei, H. Dual hierarchical genetic-optimal control: A new global optimal path planning method for robots. *J. Manuf. Syst.* **2014**, *33*, 139–148. [[CrossRef](#)]
43. Givehchi, M.; Ng, A.; Wang, L. Evolutionary optimization of robotic assembly operation sequencing with collision-free paths. *J. Manuf. Syst.* **2011**, *30*, 196–203. [[CrossRef](#)]

**Disclaimer/Publisher's Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.