Kinematic Analysis of 6-DOF Arms for H20 Mobile Robots and Labware Manipulation for Transportation in Life Science Labs

Submitted: 27th September 2016; accepted 7th November 2016

Mohammed M. Ali, Hui Liu, Norbert Stoll, Kerstin Thurow

DOI: 10.14313/JAMRIS_4-2016/30

Abstract:
This paper presents the kinematic analysis of the H20 humanoid mobile robot. The kinematic analysis for the robot arms is essential to achieve accurate grasping and placing tasks for object transportation. The H20 robot has dual arms with 6 revolute joints with 6-DOF. For each arm, the forward kinematics is derived and the closed-form solution for the inverse kinematic problem with different cases of singularities is found. A reverse decoupling mechanism method is used to solve the inverse kinematic problem analytically by viewing the arm kinematic chain in reverse order. The kinematics solution is validated using MATLAB with robotics toolbox. A decision method is used to determine the optimal solution within multiple solutions of inverse kinematic depending on the joints' limits and minimum joints motion. The workspace analysis of the arm is found and simulated. Finally, a verification process was performed on the real H20 arms by applying blind and vision based labware manipulation strategies to achieve the transportation tasks in real life science laboratories.

Keywords: *kinematic analysis 6-DOF robotic arm, validation of kinematic solution, labware localization and manipulation, Kinect sensor*

1. Introduction

Mobile robots are generally used to support efficient transportation for increasing productivity and saving human resources. They are widely used in different fields of automation such as product transportation [1], domestic services [2], [3], [4], teleoperation for the tasks with power tools [5] or material handling [6]. In this work, we present the use of a mobile robot (H20 robot, Dr. Robot, Canada) in a life science environment. The robot is a wireless networked autonomous humanoid mobile robot. It has a PC tablet, dual arms, and an indoor GPS navigation system (see Fig. 1). Some key technical issues such as a wireless remote control system [7], a low-cost robot localization [8], and multi-floor navigation system [9], have been solved recently to develop the transportation system of the H20 mobile robots. For object transportation, the grasping and placing tasks are very essential and have to be performed reliably, carefully, and in a safe way. The manipulation of a desired object requires the finding of the pose of these objects with respect to the arm base depending on specific sensors followed by using an accurate kinematic model to move the arm end effector from one pose to another precisely and in a safe path. The kinematic analysis is the way to describe the motion of the arm links without considering the forces that cause this motion. There are two types of kinematic problems: forward kinematics (FK) and inverse kinematics (IK). The forward kinematics describes how to find the end-effector pose relative to the arm base for the given joint angles. On the other hand, the inverse kinematics is based on finding the joint angles for the given pose of the end-effector with respect to the arm base.

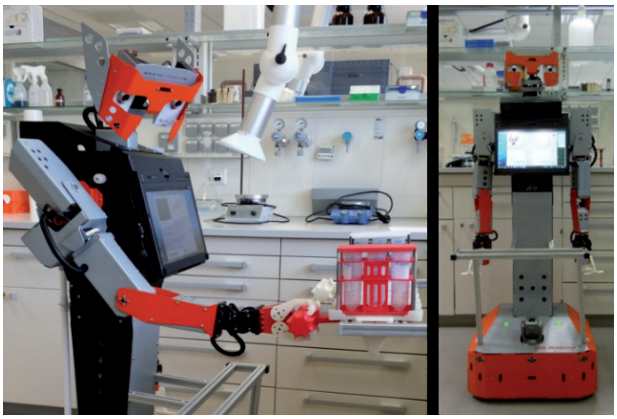


Fig. 1. H20 mobile robot

The inverse kinematics plays an active role in object manipulation because it is an important issue to enable the arm end-effector to reach the desired object accurately. Also, there are other issues, which have to be taken into the consideration when controlling the robotic arm, such as singularities, joint limits and reachable workspace. Generally, the IK problem can be solved using two approaches: analytic and numeric. But, the inverse kinematics problem does not have a unique solution and the solution which ensures collision-free configuration and minimum joint motion is considered more optimal [10]. Therefore, it is important to use a decision strategy to choose the suitable solution for the required task.

Most researchers prefer numerical methods for solving the IK problems to avoid the difficulty of finding the analytical solution [11], [12], [13]. Normally, the analytical approach is appropriate for real time applications because all the solutions can be found and it is computationally fast in comparison with the numerical approach. The analytical solution can be classified into geometric (closed-form) and algebraic. For the geometric method, the complexity of finding the IK solution increases when the manipula-

tor has more than 4 joints. Furthermore, the solution approach cannot be generalized from one manipulator to another because it depends on the number of manipulator joints, their types, structure, and coordinates frames. The closed-form solution can only be found for specific types of robotic arms, which have a particular structure with 6-DOF or less. D. L. Peiper indicates that in case there are 3 consecutive joints axes which are parallel to each other or intersecting at a single point then the closed-form solution can be existent [14]. The closed-form solution for the H20 arms can be found because the three shoulder joint axes intersect at a single point as shown in Fig. 2.

There are many researches related to the closed-form solution of the IK problem. C. G. S. Lee *et al.* proposed a closed-form solution of inverse kinematics for a 6-dof PUMA robot [15]. T. Ho *et al.* proposed a fast closed-form inverse kinematic solution for a specific 6-DOF arm [16]. G. Huang *et al.* presented a strategy for solving the inverse kinematic equations for a 6-DOF arm of humanoid meal service robot [17]. C. Man *et al.* introduced a mathematical approach for kinematic analysis of a humanoid robot [18]. R. P. Paul *et al.* proposed an inverse-transform technique to solve the IK problem for a 6-DOF robotic manipulator [19]. T. Zhao *et al.* proposed a method to divide the IK problem of a 7-DOF humanoid arm into sub-problems to find the closed-form solution taking the constraint of the elbow position into consideration [20].

M. A. Ali *et al.* proposed a reverse decoupling mechanism method to solve the IK problem of humanoid robots analytically [21]. The strategy of this method depends on viewing the kinematic chain of the manipulator in reverse order with decoupling the position and orientation. In other words, the arm can be viewed in reverse order so that the pose of the arm base can be described relative to the end effector. This method includes also decision equations to choose the suitable solution within multiple solutions. R. O'Flaherty *et al.* utilized the same method to find the closed-form solution of the IK problem for the HUBO2+ humanoid robot [22].

In this paper, the forward and inverse kinematics solutions for the 6-DOF H20 arms are derived. The closed-form solution of the IK problem has been found using the reverse decoupling mechanism method [21]. Also, the IK solution has been validated by MATLAB and verified experimentally on the H20 arms. Two approaches for labware manipulation are implemented: a blind approach using sonar sensor and a vision based approach using Kinect sensor. The paper is organized as follows: in section 2, the description of the manipulation problem is presented. In section 3, the H20 arms structure with FK and IK solutions are presented. The strategy to choose the desired IK solution and the validation of the kinematic model are given in section 4 and 5 respectively. Section 6 shows the workspace analysis of the H20 arm. Section 7 shows the verification process of the kinematic solution with the H20 arms. The client-server model is presented in section 8. The labware manipulation strategies using the ultrasonic sensor and the Kinect sensor are shown in section 9. Finally, the results are summarized with the conclusions.

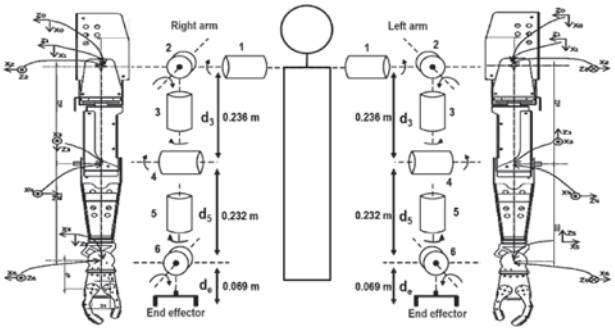


Fig. 2. H20 arms structure and coordinate frames

2. Problem Statement

For objects transportation, mobile robots usually follow a predefined path to a specified station using a guidance control system. The H20 mobile robots use the Stargazer sensor with ceiling landmarks for maneuvering between the adjacent labs (Hagisonic Company, Korea). This system inevitably causes orientation and position error of ±3cm in Z-axis and ±2cm in X-axis in front of the workstation. The error is related to two reasons. The first is the strong lighting and sunlight, which makes the star gazer unable to recognize the ceiling landmarks. The second reason is related to the odometry system, which includes encoders that are mounted on the robot wheels to provide the motion information that updates the robot pose. The odometry system accumulates errors for different reasons such as different wheels diameter, wheel-slippage, wheels misalignment and finite encoder resolution, and according to the experimental results and previous studies, the rotation of the robot is the greatest factor for odometry errors [23], [24]. Uncertainties in the pose of the robot in front of the work bench lead to failures in the grasping and placing tasks. These failures have to be avoided because the H20 robots deal with labwares, which contain chemical and biological components. The required accuracy for labware manipulation has to be less than 1 cm to guarantee safe tasks. Therefore, the more direct way of dealing with this problem is to use sensors to provide the position and orientation of the target [25]. Then, the joints of the arm have to be configured using the kinematic model to place the arm end effector precisely and safely in the desired pose if it is inside the reachable space.

3. H20 Arms Kinematic

This section describes in more details the structure and the kinematic analysis of the H20 arms.

3.1. Structure of H20 Arms

The H20 mobile robot has dual arms, each consists of 6 revolute joints with 6-DOF. In addition, each arm also has a 2-DOF gripper. Fig. 2 shows the structure and the coordinate frames for the H20 arms. The length of

the upper arm d_3 and the forearm d_5 are 0.236 m and 0.232 m, respectively. Also, the distance (d_e) between the wrist joint and the end-effector is 0.069 m.

3.2. D-H Representation

The Denavit-Hartenberg representation is used to describe the translation and rotation relationship between the arm adjacent links where it provides a guide for locating coordinate systems on each link of a multi-link kinematic chain. Denavit and Hartenberg proposed to define the manipulator with four joint-link parameters for each link [26]. Fig. 3 shows a pair of adjacent links, link(i-1) and link i, their associated joints, joint (i-1), i and (i+1), and axis (i-2), (i-1) and i, respectively. A frame {i} is assigned to link i as follows:

- The Z_{i-1} lies along the axis of motion of the i^{th} joint.
- The X_i axis is normal to the Z_{i-1} axis, and pointing away from it.
- The Y_i axis completes the right–handed coordinate system as required.

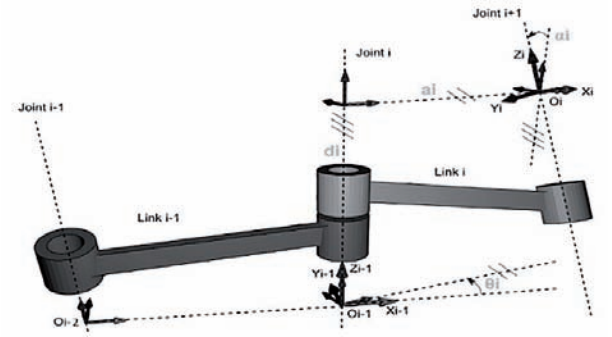


Fig. 3. D-H conventions for frame assigning [26]

There are four parameters used in the manipulator analysis: the link length (a_{i-1}), the link twist (α_{i-1}), the link offset (d_i), and the joint angle (θ_i) where (i) refers to the link number. The definitions of D-H parameters are as follows:

- Link length (a_i): The distance measured along x_i axis from the point of intersection of x_i axis with z_{i-1} axis to the origin of frame {i}.
- Link twist (α_i): The angle between z_{i-1} and z_i axes measured about x_i-axis in the right hand sense.
- Joint distance (d_i): The distance measured along z_{i-1} axis from the origin of frame {i–1} to intersection of x_i axis with z_{i-1} axis.
- Joint angle (θ_i): The angle between x_{i-1} and x_i axes measured about the z_{i-1} axis in the right hand sense.

Table 1. The D-H parameters and the joints limit

Left and Right Arms						
θ_i	$\alpha_{(i-1)}(L)$	$\alpha_{(i-1)}(R)$	$a_{(i-1)}(LR)$	$d_i(m)(L)$	$d_i(m)(R)$	Joints limit (LR)
θ_1	0°	0°	0	0	0	-20°~192°
θ_2	90°	-90°	0	0	0	-200°~-85°
θ_3	90°	-90°	0	-0.236	0.236	-195°~15°
θ_4	-90°	90°	0	0	0	-129°~0°
θ_5	90°	-90°	0	-0.232	0.232	0°~180°
θ_6	-90°	90°	0	0	0	-60°~85°

By following the D-H rules, the homogeneous transformations between adjacent links are defined. The D-H parameters and the rotational limit for each joint of the H20 arms are described in Table 1.

3.3. Forward Kinematics Computation

The forward kinematics is how to find the end-effector pose relative to the arm base for the given joint angles. This can be solved by finding the transformation matrices of the arm from one link to the next according to the D-H coordinate system. Eq. 1 represents the 4×4 general homogeneous transformation matrix of the H20 arms. By substituting the link parameter from table 1 into (1), the transformation matrices (${}^{i-1}_{i}T$(i = 1~6)) of the H20 arms are found.

$$ {}^{i-1}_{i}T = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & a_{(i-1)} \\ \sin\theta_i \cos\alpha_{(i-1)} & \cos\theta_i \cos\alpha_{(i-1)} & -\sin\alpha_{(i-1)} & -\sin\alpha_{(i-1)} d_i \\ \sin\theta_i \sin\alpha_{(i-1)} & \cos\theta_i \sin\alpha_{(i-1)} & \cos\alpha_{(i-1)} & \cos\alpha_{(i-1)} d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1) $$

According to Fig. 2, there is a distance of (0.069 m) between the wrist joint and the end-effector. The transformation matrix which describes this translation is as follows:

$$ {}^{6}_{E}T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0.069 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. $$

Finally, the forward kinematic solution which describes the pose of the end effector relative to the arm base can be obtained using Eq. (2). In this equation (n_x, n_y, n_z) is the orthogonal vector, (o_x, o_y, o_z) is the orientation vector, (a_x, a_y, a_z) is the approach vector and (p_x, p_y, p_z) is the end effector position vector.

$$ {}^{0}_{E}T = {}^{0}_{1}T \cdot {}^{1}_{2}T \cdot {}^{2}_{3}T \cdot {}^{3}_{4}T \cdot {}^{4}_{5}T \cdot {}^{5}_{6}T \cdot {}^{6}_{E}T = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (2) $$

3.4. Inverse Kinematics Computation

The inverse kinematics enables the finding of the joint angles for the given position and orientation of the end-effector with respect to the reference coordinate system. The inverse kinematic problem was solved using the reverse decoupling mechanism method [21]. With this method, the kinematic chain of the arm is viewed in reverse order. This means that the shoulder coordinate frame is described relative to the end-effector coordinate frame. In this case, the position vector will be a function of only 3 joints θ_4, θ_5, and θ_6. To start solving the inverse kinematic problem of the left and right arms, the inverse of forward kinematic matrix has to be found as follows:

$$ {}^{E}_{0}T = {}^{E}_{6}T \cdot {}^{6}_{5}T \cdot {}^{5}_{4}T \cdot {}^{4}_{3}T \cdot {}^{3}_{2}T \cdot {}^{2}_{1}T \cdot {}^{1}_{0}T = \begin{bmatrix} in_x & io_x & ia_x & ip_x \\ in_y & io_y & ia_y & ip_y \\ in_z & io_z & ia_z & ip_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3) $$

To find the solution for the joints 4, 5, and 6, the both sides of (3) have to be multiplied by $({}^{5}_{6}T \cdot {}^{6}_{E}T)$ and the result is as follows:

$$ {}^{5}_{4}T \cdot {}^{4}_{3}T \cdot {}^{3}_{2}T \cdot {}^{2}_{1}T \cdot {}^{1}_{0}T = {}^{5}_{6}T \cdot {}^{6}_{E}T \cdot {}^{E}_{0}T. \quad (4) $$

The right side of (4) is the following matrix:

$$ \begin{bmatrix} r_1n_x & r_1o_x & r_1a_x & -S_6(ip_y + d_e) + C_6ip_x \\ r_1n_y & r_1o_y & r_1a_y & ip_z \\ r_1n_z & r_1o_z & r_1a_z & -C_6(ip_y + d_e) - S_6ip_x \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (5) $$

and the left side of (4) is:

$$ \begin{bmatrix} l_1n_x & l_1o_x & l_1a_x & -d_3S_4C_5 \\ l_1n_y & l_1o_y & l_1a_y & d_3S_4S_5 \\ l_1n_z & l_1o_z & l_1a_z & d_3C_4 + d_5 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (6) $$

where S and C are the abbreviation of sine and cosine of the angle, respectively. By equating the position elements of (5) and (6), it is obtained:

$$ -S_6(ip_y + d_e) + C_6ip_x = -d_3S_4C_5, \quad (7) $$

$$ ip_z = d_3S_4S_5, \quad (8) $$

$$ -C_6(ip_y + d_e) - S_6ip_x = d_3C_4 + d_5. \quad (9) $$

To solve the joint angle θ_4, suppose the following terms: $(ip_y + d_e = rS_\gamma)$ and $(ip_x = rC_\gamma)$, where $r = (\sqrt{(ip_y + d_e)^2 + (ip_x)^2})$, $\gamma = (\text{atan2}\ (ip_y + d_e, ip_x))$, and (atan2) is the two argument arc tangent function. These terms are obtained according to the arms coordinates frame with reverse order. By substituting these terms into (7) and (9) and using the angle sum identities, the following equations can be obtained:

$$ rC_{6+\gamma} = -d_3S_4C_5, \quad (10) $$

$$ rS_{6+\gamma} = d_3C_4 + d_5. \quad (11) $$

The equation of C_4 for the left and right arms can be obtained by squaring (8), (10), and (11) and adding them. The solution is as follows:

$$ C_{4(LR)} = \frac{(ip_y + d_e)^2 + (ip_x)^2 + ip_z{}^2 - d_3{}^2 - d_5{}^2}{2d_3d_5}. \quad (12) $$

Thus, the solution of θ_4 for the left and right arms can be found.

$$ \theta_{4(LR)} = \text{atan2}\left(\pm real\left(\sqrt{1 - C_4^2}\right), C_4\right). \quad (13) $$

The arc tangent function (atan2(sinθ,cosθ)) is more consistent to be used for finding the angle value (θ) than arc cosine and arc sine. This is because of the inaccurate behavior to determine the required angle in case of using the arc sine and arc cosine functions. The complex numbers are generated in case the target position is not within the reachable workspace of the arm. Therefore, (*real*) function is used to ignore the imaginary parts of complex numbers and take only the real part in the joint solutions. Thus, the solution that is closest to the target position can be obtained [22]. After finding the value of θ_4, the equation of S_5 can be solved using (8) to get the solution of θ_5 as follows:

$$ S_{5(L)} = \frac{ip_z}{S_4d_3}, S_{5(R)} = \frac{-ip_z}{S_4d_3}, \quad (14) $$

$$ \theta_{5(LR)} = \text{atan2}\left(S_5, \pm real\left(\sqrt{1 - S_5^2}\right)\right). \quad (15) $$

The solution of θ_6 can be obtained by dividing (11) by (10) to get the following equation:

$$ \frac{S_{6+\gamma}}{C_{6+\gamma}} = \frac{d_3C_4 + d_5}{-d_3S_4C_5}. \quad (16) $$

Thus, the value of θ_6 can be found as follows:

$$ \theta_{6(LR)} = \text{WToPi}(\text{atan2}(d_3C_4 + d_5, -d_3S_4C_5) - \gamma), \quad (17) $$

where (WToPi) is a function to wrap the angle to the interval between –π and π [22]. To find the solution for the joints 1, 2, and 3, the both sides of (3) have been multiplied by $({}^{3}_{4}T \cdot {}^{4}_{5}T \cdot {}^{5}_{6}T \cdot {}^{6}_{E}T)$ and the result is as follows:

$$ {}^{3}_{2}T \cdot {}^{2}_{1}T \cdot {}^{1}_{0}T = {}^{3}_{4}T \cdot {}^{4}_{5}T \cdot {}^{5}_{6}T \cdot {}^{6}_{E}T \cdot {}^{E}_{0}T. \quad (18) $$

The left side of (18) is the following matrix:

$$ \begin{bmatrix} S_1S_3 + C_1C_2C_3 & C_2C_3S_1 - C_1S_3 & -S_2C_3 & 0 \\ C_3S_1 - C_1C_2S_3 & -C_1C_3 - C_2S_1S_3 & S_2S_3 & 0 \\ -C_1S_2 & -S_1S_2 & -C_2 & -d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (19) $$

By taking the element (3,3) in the left and right sides of (18) and equating them, the equation of C_2 which is used to find θ_2 can be obtained as follows:

$$ C_{2(\text{L})} = ia_y(C_4C_6 - C_5S_4S_6) + ia_x(C_4S_6 + C_5C_6S_4) - ia_zS_4S_5, $$

$$ C_{2(\text{R})} = -ia_y(C_4C_6 - C_5S_4S_6) - ia_x(C_4S_6 + C_5C_6S_4) - ia_zS_4S_5, $$

$$ \theta_{2(LR)} = \text{atan2}\left(\pm\text{real}\left(\sqrt{1 - C_2^2}\right), C_2\right). \quad (20) $$

For the solution of θ_3, the elements (1,3) and (2,3) in the left and right sides of (18) are compared and divided by each other to get S_3 and C_3 which are used to find the solution of θ_3:

$$C_{3(L)} = ia_y(C_6S_4 + C_4C_5S_6) + ia_x(S_4S_6 - C_4C_5C_6) + ia_zC_4S_5,$$

$$C_{3(R)} = -ia_y(C_6S_4 + C_4C_5S_6) - ia_x(S_4S_6 - C_4C_5C_6) + ia_zC_4S_5,$$

$$S_{3(L)} = -ia_yS_5S_6 + ia_xC_6S_5 + ia_zC_5,$$

$$S_{3(R)} = ia_yS_5S_6 - ia_xC_6S_5 + ia_zC_5,$$

$$\theta_{3(LR)} = \text{atan2}(S_3, C_3). \quad (21)$$

The last step is to find the solution of θ_1, which is obtained by comparing the elements (3,1) and (3,2) in the left and right sides of (18) and dividing them by each other to get S_1 and C_1 which are used to find the solution of θ_1 as the following:

$$C_{1(L)} = in_y(C_4C_6 - C_5S_4S_6) + in_x(C_4S_6 + C_5C_6S_4) - in_zS_4S_5,$$

$$C_{1(R)} = in_y(C_4C_6 - C_5S_4S_6) + in_x(C_4S_6 + C_5C_6S_4) + in_zS_4S_5,$$

$$S_{1(L)} = io_y(C_4C_6 - C_5S_4S_6) + io_x(C_4S_6 + C_5C_6S_4) - io_zS_4S_5,$$

$$S_{1(R)} = io_y(C_4C_6 - C_5S_4S_6) + io_x(C_4S_6 + C_5C_6S_4) + io_zS_4S_5,$$

$$\theta_{1(LR)} = \text{atan2}(S_1, C_1). \quad (22)$$

According to the previous joints' solutions, it can be noticed that θ_2, θ_4, and θ_5 have two solutions. This causes 8 total solutions for the inverse kinematics problem of the H20 arms. The strategy about how to choose the optimal solution will be discussed in section 4.

3.5. Dealing with the Singularity Cases

Singularities are arm configurations in which one or more degrees of freedom are eliminated when some joints' axes align with each other. Thus, the number of solutions for the IK problem will be infinite. Three cases of singularity have been determined within the joints limits. The inverse kinematic solution for every case of singularity is as follows:

A) When θ4 = 0 and θ2 ≠ -π: The axis of the third joint is aligned with the fifth joint axis as shown in Fig. 4.

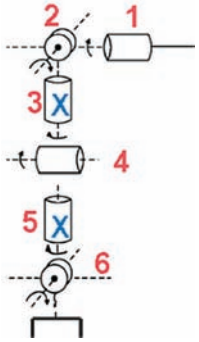


Fig. 4. Case A where the 3rd and 5th joints are aligned

To find the IK solution for this case, keep the previous value of θ_3 and define $\theta_T = \theta_3 + \theta_5$. Firstly, make $\theta_4 = 0$, then start with the solution of θ_6 by using the elements (1,4) and (2,4) of (3) and dividing them by each other to get S_6 and C_6. The solution of θ_6 for the left and right arms are found as follows:

$$\frac{-ip_x}{-ip_y - d_e} = \frac{S_{6(LR)}}{C_{6(LR)}},\ \theta_{6(LR)} = \text{atan2}(S_6, C_6). \quad (23)$$

For the solution of θ_1, compare the elements (3,1) and (3,2) in the left and right sides of (4).

$$\theta_{1(LR)} = \text{atan2}\,(S_6io_x + C_6io_y, S_6in_x + C_6in_y). \quad (24)$$

The elements (3,1) and (3,3) in the left and right sides of (4) are compared to get the solution of θ_2 which is as follows:

$$\theta_{2(L)} = \text{atan2}\,((-S_6in_x - C_6in_y)/C_1, C_6ia_y + S_6ia_x),$$

$$\theta_{2(R)} = \text{atan2}\,((-S_6in_x - C_6in_y)/C_1, -C_6ia_y - S_6ia_x). \quad (25)$$

And if $C_1 < 0$, then $\theta_2 = \text{wrapToPi}\,(\theta_2 + \pi)$.

The solution of θ_5 can be found by solving θ_T first. This is done by using the elements (1,3) and (2,3) in the left and right sides of (4) with the angle sum identities to get S_{3+5} and C_{3+5}. Thus, the solution of θ_T is as follows:

$$S_{(3+5)(L)} = -ia_z,\ S_{(3+5)(R)} = ia_z$$

$$C_{(3+5)(LR)} = ia_xC_6 - ia_yS_6$$

$$\theta_{T(LR)} = \text{atan2}(S_{(3+5)}, C_{(3+5)}). \quad (26)$$

If $S_{2(L)} < 0$, then $\theta_{T(L)} = \text{wrapToPi}\,(\theta_{T(L)} + \pi)$.
And if $S_{2(R)} > 0$, then $\theta_{T(R)} = \text{wrapToPi}\,(\theta_{T(R)} + \pi)$.

Finally, the solution of θ_5 can be obtained as follows:

$$\theta_{5(LR)} = \text{wrapToPi}(\theta_T - \theta_3). \quad (27)$$

B) When θ2 = -π and θ4 ≠ 0: The axis of the first joint is aligned with the third joint axis as shown in Fig. 5.

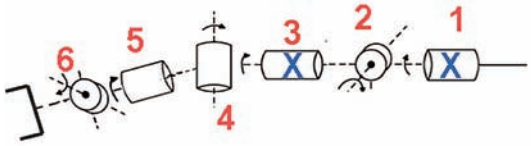


Fig. 5. Case B where the 1st and 3rd joints are aligned

The solutions of θ_4, θ_5, and θ_6 are found using the same equations as in section 3.4. Also, the previous value of θ_1 will be kept and define $\theta_T = \theta_1 + \theta_3$. To find the solution of θ_5, first make $\theta_2 = -\pi$. Then, start with the solution of θ_T by using the elements (3,1) and (3,2) in the left and right sides of (4) with the angle sum identities to obtain S_{1+3} and C_{1+3}. The solution is as follows:

$$S_{(1+3)(L)} = -io_x S_6 - io_y C_6,\ S_{(1+3)(R)} = io_x S_6 + io_y C_6,$$

$$C_{(1+3)(L)} = -in_x S_6 - in_y C_6,\ C_{(1+3)(R)} = in_x S_6 + in_y C_6,$$

$$\theta_{T(LR)} = \text{atan2}\left(S_{(1+3)}, C_{(1+3)}\right). \quad (28)$$

If $S_4 > 0$, then $\theta_{T(L)} = \text{wrapToPi}\ (\theta_{T(L)} + \pi)$.

If $S_4 < 0$, then $\theta_{T(R)} = \text{wrapToPi}\ (\theta_{T(R)} + \pi)$.

Finally, the solution of θ_3 can be obtained as follows:

$$\theta_{3(LR)} = \text{wrapToPi}\ (\theta_T - \theta_1). \quad (29)$$

C) When θ4 = 0 and θ2 = -π: The joints 1, 3, and 5 are collinear as shown in Fig. 6.

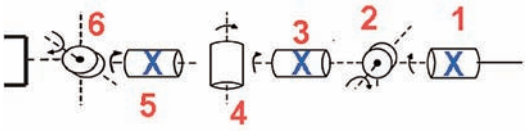


Fig. 6. Case C, the 1st, 3rd, and 5th joints are aligned

For this case, the solutions of θ_6 is found using the same equations as in case A (when $\theta_4 = 0$ and $\theta_2 \neq -\pi$). Also, keep the previous values of θ_1 and θ_3, then define $\theta_T = \theta_1 + \theta_3 + \theta_5$. To find the solution of θ_5, first make $\theta_2 = -\pi$ and $\theta_4 = 0$. Thus, start with the solution of θ_T by using the elements (2,1) and (2,2) in the left and right sides of (4) with the angle sum identities to obtain S_{1+3+5} and C_{1+3+5} as follows:

$$S_{(1+3+5)(L)} = in_z,\ S_{(1+3+5)(R)} = -in_z$$

$$C_{(1+3+5)(L)} = -io_z,\ C_{(1+3+5)(R)} = io_z$$

$$\theta_{T(LR)} = \text{atan2}\left(S_{(1+3+5)}, C_{(1+3+5)}\right). \quad (30)$$

Finally, the solution of θ_5 can be obtained as follows:

$$\theta_{5(LR)} = \text{wrapToPi}\left(\theta_T - \theta_1 - \theta_3\right). \quad (31)$$

4. The Selection of a Desired Solution

For object manipulating tasks, the pose information of the object relative to the arm base determines whether this object is inside the workspace or not. In case that the object is outside the workspace, then there is no solution for the inverse kinematic problem. But, there will be 8 solutions for the required pose if it is inside the reachable workspace (as detailed in section 3.4). To choose the suitable solution within the 8 solutions, the joints limits have to be taken into consideration. If one joint value related to a specific solution is not within the joint limits, the solution will be ignored. In case that there are multiple solutions, where all the joints' values of every solution are within the joints' limits, a selecting algorithm has to be used. The solution with minimum joint motion will be selected using this algorithm. This is done by finding the sum of the squared joint values for the arm previous configuration and for every possible solution as follows: $(\theta_1^2 + \theta_2^2 + \theta_3^2 + \theta_4^2 + \theta_5^2 + \theta_6^2)$ [22].

The next step is to compare the sum of the previous configuration with the sum of every solution. The solution with the closest sum value to the sum value of the previous configuration is the desired solution.

5. Validation of the Kinematic Model

MATLAB software with Robotic toolbox has been used to validate the inverse kinematics solution. The joints results with the simulation plot give a clear proof for the inverse kinematic behavior of the robotic arms. The validation process has been done after giving random joints values as an input to the forward kinematic model. The pose information of the end effector, which is received from the FK model, is inserted to the inverse kinematic model that includes the joints limits with the selecting algorithm. Then, the joints values that are inserted to the FK model are compared with the result of the IK model as shown in Fig. 7.

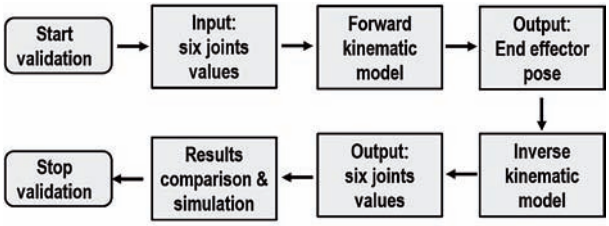


Fig. 7. The validation process for the IK solution

Different arm configurations have been used for validation processes. The results have been compared and plotted and the joints values which were inserted to the FK model are identical with the joints values which were obtained from the IK model as shown in the example of Fig. 8. In this example, Fig. 8.A shows the simulation plot of the arm with the position and orientation of the end effector according to the joint values inserted to the FK model. Fig. 8.B represents the simulation plot of the arm according to the joints obtained from the IK solution.

a ***b***

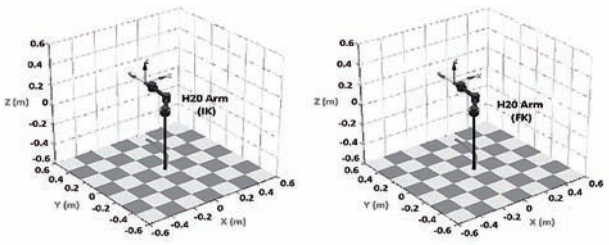


Fig. 8. The simulation plot of the H20 arm according to the configuration [50°, -90°, -90°, -30°, 180°, 10°]

6. Workspace Analysis

The workspace is the space which is swept out by the arm end effector after executing all possible motions. The workspace is one of the essential parameters for robotic arm performance in addition to its speed and accuracy. The calculation of the arm work-

space is very important to decide whether the desired object, which has to be manipulated, is inside the reachable space or not. The workspace of the robotic arm can be determined according to the links length, joints type, and joints limit. The length of the H20 arm is 0.537 m (d_3 = 0.236 m, d_5 = 0.232 m, d_e = 0.069 m) as shown in Fig. 2. Related to the joints, the H20 arm consists of 6 rotary joints (limits values' mentioned in Table 1). MATLAB software with Robotic toolbox has been used to calculate the workspace of the H20 arm by inserting the links length and all the possible joint values within the joints limit to the FK model to find the position of the end-effector for every sample. Finally, all possible positions which the robotic arm can reach will be found. The simulation of workspace envelope for the H20 arm is shown in Fig. 9.

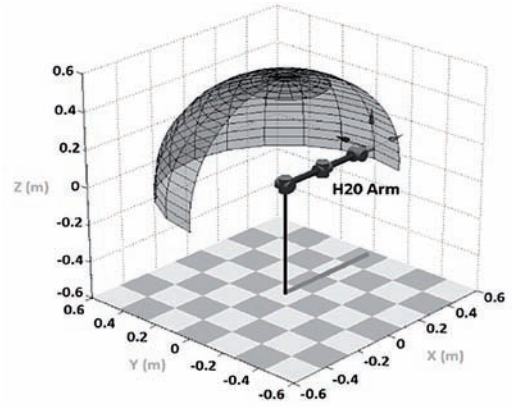


Fig. 9. The workspace envelope of the H20 arm

7. The Kinematic Model Verification

This section describes the verification procedures of the developed kinematic model with the real robotic arm. A labware manipulation strategy has been implemented using the IK solution and ultrasonic sensor to calculate the distance between the H20 robot base and the required work bench. Some essential steps such as calibration and accuracy testing have been implemented as preliminary procedures before performing the labware manipulation strategy.

7.1. Angle to Servo Position Conversion

As an initial step for applying the kinematic model with the robotic arm, a conversion process has been performed to convert the required angles values' of the joints to the related servo motors positions. The positioning resolution of the H20 arm servos (joints) is 0.09°/unit. Thus, the servo has to move 1,000 units to rotate 90° degrees. This resolution value can't be used with the H20 arms because they are unstable and have weak joints where the joints compliance causes positional errors. The effects of gravity, weight of arm parts, payload, and inertia cause the elasticity of each joint [27]. Also, the differences between the actual physical joint zero position and the physical joint zero position reported by the robot controller normally causes accuracy errors for the robotic arm. To cope with this issue, a digital tilt meter has been used (see Fig. 10). Different angle values have been configured for every joint using the tilt meter and the value of the related servo motor has been registered at each configuration. This process helps to build the equations of angle to servo conversion which is important for decreasing the accuracy errors.

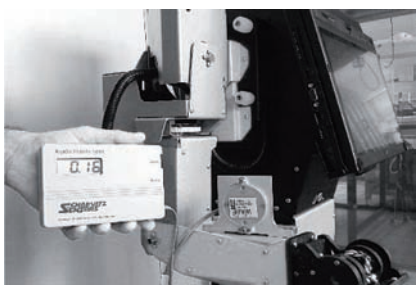


Fig. 10. Conversion process using tilt meter

7.2. Accuracy and Repeatability of H20 Arms

The checking of the accuracy and repeatability of the robotic arms is an essential step for the tasks of object manipulation. The repeatability of the robotic arm describes how precisely this arm can return to a taught position. In general, larger robots have larger errors in repeatability. On the other hand, the accuracy of the robotic arm describes how precisely this arm can reach the required position. One of the main technological limitations in the robotics industry is the improvement of the accuracy by reducing the error between the tool frame and the goal frame. The precision depends on some elements such as the resolution of the control system, joint compliance, and the imprecision of the mechanical linkages and DC servo motors. Also, the accuracy and the repeatability depend upon many other different factors such as friction, temperature, loading, and manufacturing tolerances. In case that the robotic arm does not provide the required accuracy, the arm has to be calibrated. Robot calibration can be performed using both contact and noncontact probing methods. Non-contact methods include the use of beam breakers, laser sensors, visual servoing, etc. [27]. The accuracy and repeatability of the H20 arm have been checked using a grid paper and a marker attached at the end-effector as shown in Fig. 11. A grasping configuration has been prepared to enable the end-effector to reach the space position of X=30 mm, Y=180 mm, Z=380 mm related to the arm shoulder.

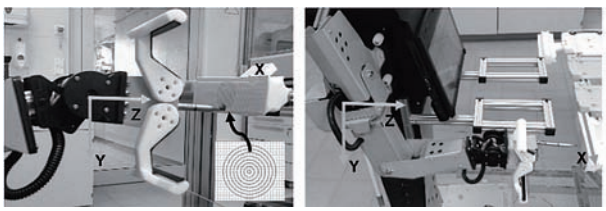


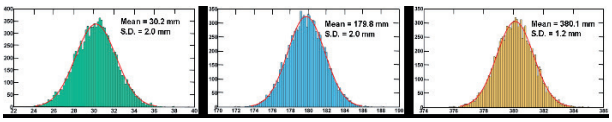
Fig. 11. The grid paper and the end-effector marker

The arm grasping movement has been repeated 40 times where the position of the end-effector has been registered at the end of every movement with the indication of the marker on the grid paper. The registered positions range of the end-effector are described in Table 2. It can be noticed from the case "before calibration" in Table 2 that there is an accuracy error in the Y-axis where the expected position

is not within the range of the registered positions. On the other hand, the expected position in X and Z axes are within the range of the registered positions. The reason of this error is the weakness of the joints with the joint compliance which is effected by the weight of arm parts. To improve the accuracy of reaching the required position, the robotic arm has to be calibrated. The accuracy and repeatability of the H20 arm have been checked again after performing a calibration process and the positions of the end-effector have been registered as shown in the row "after calibration" in Table 2. Also, the Gaussian distributions of the end-effector positions after calibration with the related mean and standard deviation are shown in Fig. 12. According to the results obtained from this experiment, the accuracy of the used arm in reaching a specific position according to the related configuration is as the following: (X:±4 mm, Y:±4 mm, Z:±2 mm).

Table 2. The expected and registered positions

Case	X(mm)	Y(mm)	Z(mm)
The expected position (hand related to shoulder)	30	180	380
The registered positions (before calibration)	26~34	195~203	378~382
The registered positions (after calibration)	26~34	176~184	378~382



(A) (B) (C)

Fig. 12. Gaussian distribution, A: for X-values, B: for Y-values, C: for Z-values

A calibration process for the robotic arm has been performed to keep the end-effector at a fixed height of 180 mm for different distances between the shoulder and the end-effector where the value 180 mm represents the height between the robot shoulder and the labware handle on the workstation. This has been done by inserting a specific Y-value to the IK model at each specific distance as shown in Fig. 13. For example, in case that the required distance is 400 mm, the Y-value which has to be inserted to the IK is 155 mm, to keep the end-effector at the height of 180 mm.

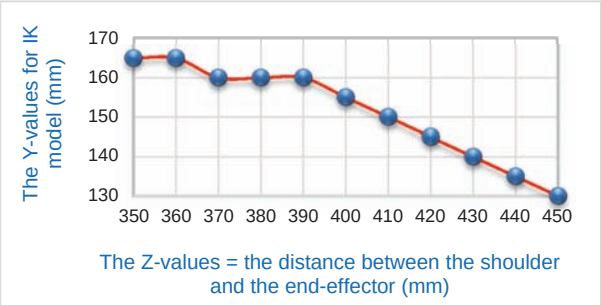


Fig. 13. The calibration process to keep the end-effector at the height of 180 mm for different distances

8. Client-Server Communication Model

The programming code to control each device or component related to the life science automation system has been developed using a specific language. It is complex task to integrate different control platforms in a single one due to the size of the platforms and the usually different programming languages. Therefore, it is required to develop a communication system that enables the simultaneous interaction of all the devices for a flexible process execution. The control systems of the robot components are connected in a common LAN. The client-server communication model can enable the control system of each component to interact with the others over Ethernet using a specific IP address and port number. The client initiates the process with the server by requesting a connection to a specific socket address using TCP/IP where the socket address is a combination of IP address and a port number. If the requested port is free, then the server will establish the connection to communicate with the client. A client-server model has been developed to connect the arm manipulation system (AMS) with the H20 navigation control system (NCS) [8], [9]. Both control systems exchange the orders and information to perform the transportation task for the labwares.

9. Labware Manipulation

For the verification of the developed IK solution with real applications, object manipulation strategies have been performed to achieve the labware transportation in different life science laboratories using H20 robots. According to the workspace of the H20 arms and to the robot position in front of the labware station, each arm can manipulate 2 labwares alongside each other as shown in Fig. 14. R1 and R2 labwares can be manipulated using the right arm. Whereas, the left arm can manipulate the L1 and L2 labwares. The labware containers have fixed positions on the workstation.

The information about the tasks and target is sent from the process management system (PMS) to the NCS which in turn transfers it to the AMS. Two approaches for labware manipulation have been implemented: a blind sonar sensor based method and a vision based method using Kinect sensor. Both methods have been developed using Microsoft Visual Studio 2015 with C# programming language. The projects are running on a Windows 10 platform in the H20 tablet.

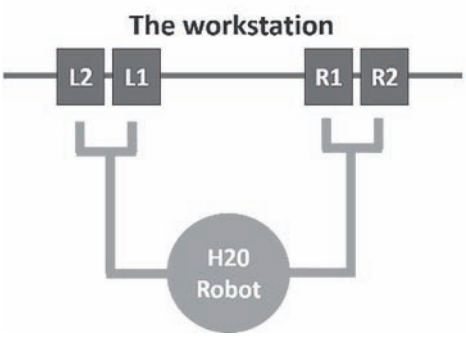


Fig. 14. Manipulation ability of H20 arms

9.1. Arm Manipulation Using Sonar Sensor

This strategy has been performed using the developed kinematic model and the built-in DUR5200 ultrasonic sensors. The ultrasonic sensor can be used for different applications such as map building for mobile robot environment, collision avoidance, robot range finder and distance detection. The DUR5200 ultrasonic range sensor module can detect the range information from 4 cm to 340 cm. The distance data is precisely calculated by the time interval between the instant when the measurement is enabled and the instant when the echo signal is received. For the process of grasping and placing, the API and the communication process between the NCS and the AMS have been implemented. As the H20 robot arrives at the desired location in front of the workstation, the first step, which will be performed, is the orientation correction of the robot to be straight. As the labware container has a specific posture on the workstation according to its design (see Fig. 15), the pitch and roll orientation related to the robot are fixed. But the yaw orientation has to be corrected. The yaw orientation has been corrected using two sonar sensors mounted on the base of the H20 robot (see Fig. 17). The distance (Z) from each channel to the workstation is checked and the robot rotates till the values of both sensors are equalized. Also, the height of every workstation in the labs is known. The error value of the desired robot position in front of the workstation is ±2cm in the X-axis. This error has been compensated by the design of grippers and labware container handles as shown in Fig. 16. There is a space range of ±3cm between the lengths of grippers and handle where this space range compensates the X-error of robot position to guarantee a secure grasping.

Using the client-server communication model, the orders are sent from the NCS to the AMS. The order includes the required task (grasping or placing), the height (Y-value) of the workstation, the desired target (R1, R2, L1, L2) which determines the X- value position, and the distance (Z-value) between the robot

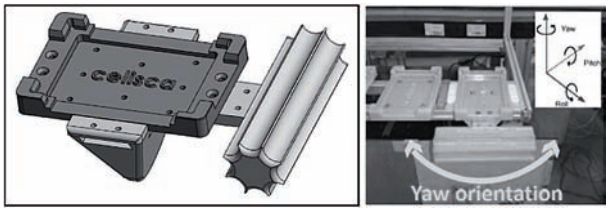


Fig. 15. 3D design and posture of labware container

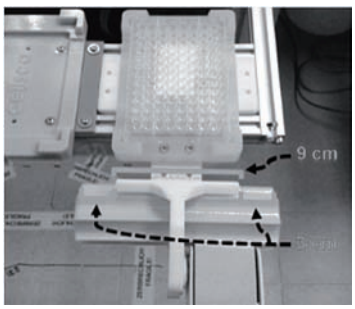


Fig. 16. The manner of grasping the handle

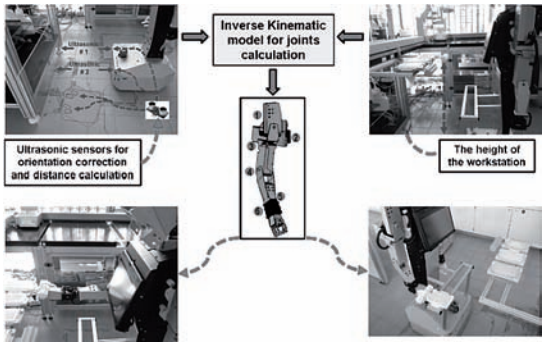


Fig. 17. The framework of labware manipulation

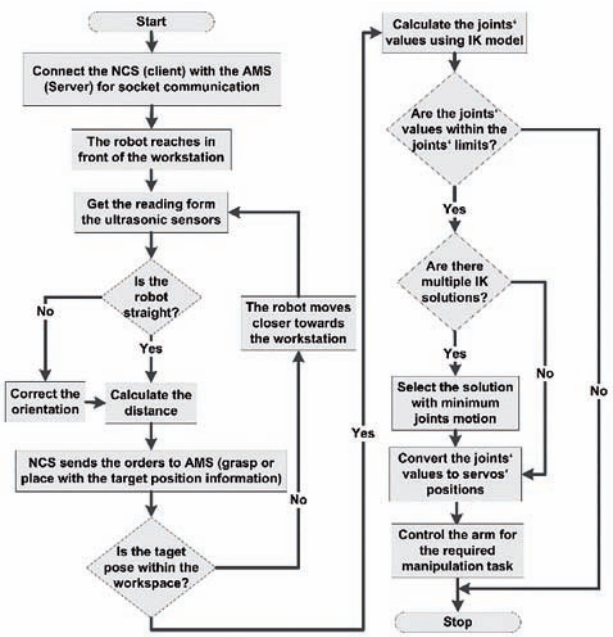


Fig. 18. The flowchart of labware manipulation

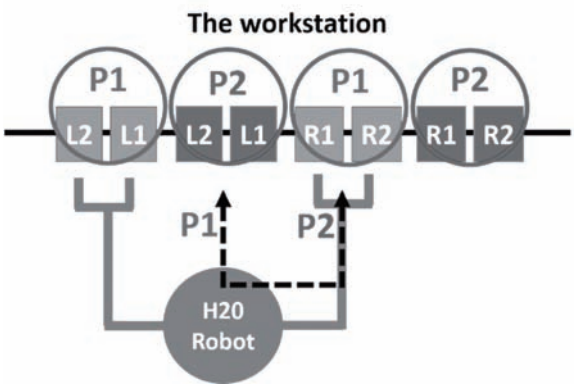


Fig. 19. Multiple positions of robot for manipulation

base and the workstation obtained from the sonar sensor. Depending on this information, the labware pose related to the arm shoulder is found and checked whether it is inside the arm reachable space or not. If it is inside the reachable space, the IK solution will be calculated and the joints limits will be checked. If there are multiple solutions, then a decision procedure will be used to select the solution with minimum joints motion. Afterwards, the converting equations will be used to convert the angle value of each joint to a servo position value to move the arm to the required pose as shown in Fig. 17. The average of accuracy with

this strategy of labware manipulation is less than 1cm and the task of grasping or placing takes about 40 seconds to finish the process. The flowchart of this manipulation strategy is shown in Fig. 18.

For the purpose of safe transportation, a holder has been mounted on the robot body (see Fig. 17) to guarantee a straight and secure posture for the labware which contains chemical and biological components.

This strategy can be applied also for multiple robot positions in front of the workbench. Different positions can be defined for the robot to reach. Then, 4 labwares locations on the workstation can be determined for each robot position. Using this way, multiple labware manipulation for multiple robot positions can be realized as shown in Fig. 19. The shift distance between the two positions is 29 cm. This strategy depends on the required robot position and the required labware position as follows: P1R1, P1R2, P1L1, P1L2, P2R1, P2R2, P2L1, and P2L2.

The possible weight, which the arm can manipulate using this design of gripper with handle, is about 350g. This limited payload is related to the weak wrist joint of the H20 arms. To manipulate heavier labwares blindly, a vertical handle has been designed as shown in Fig. 20. Using this design, the arm configuration will be in the form which locks the weak wrist joint

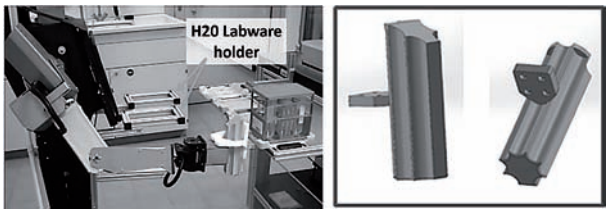
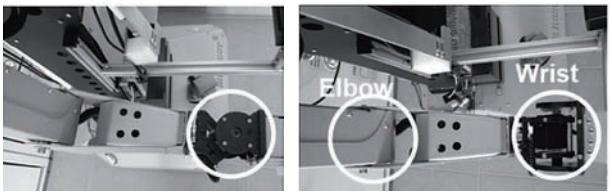


Fig. 20. The design of vertical handle



A *B*

Fig. 21. The arm structure for manipulation. A: horizontal handle. B: vertical handle

when the grippers grasp the handle. In this case, the lifting process depends on the elbow joint which has more powerful torque (see Fig. 21). The possible payload that can be lifted safely with this design is 500 g.

Fig. 21.A shows the wrist and elbow joint of the H20 arm for the case of grasping horizontal handle. The wrist joint is very weak and it is unable to lift heavy labwares which leads to unsecure manipulations. With the vertical handle, the lifting movement of the wrist joint will be locked as shown in Fig. 21.B. The elbow joint, that is more powerful than wrist joint, will be responsible for lifting the heavy labwares from the workstation.

9.2. Arm Manipulation Using Kinect Sensor

The required labware has to be distinguished and manipulated wherever it is located on the workstation. To perform that visually, the required target has to be identified and its pose related to the robot has to be calculated. Different visual sensors can be used for this purpose such as stereo vision and 3D camera. The Kinect sensor, which is a kind of 3D camera, is considered as a preferred solution for such tasks since it provides the depth information without the need of deep image processing steps as in stereo vision. There are 2 kinds of Kinect, V1 and V2. Both of them have been used to perform the labware manipulation with H20 robots. The Kinect sensor has been fixed on the H20 body using a holder with a suitable height and tilt angle to guarantee a clear and wide view for the whole workstation as shown in Fig. 22.

Fig. 22. The holder of Kinect sensor

For transporting multiple labware, it is necessary to have an intelligent behavior to grasp the desired handle where the required labware is positioned. The Kinect sensor V1 has been used to detect and localize single color objects [28]. The labware container handle and its placing holder have been detected using RGB color filtering as shown in Fig. 23. The Kinect V1 has 640×480 and 320×240 color frame and depth frame resolution, respectively.

To improve the identification of different handles, a new design with flat panels on the upper side has been developed. Proper grippers have been designed also to fit the new handle. The upper flat panel is used for fixing different colored or pictorial marks to distinguish multiple handles as shown in Fig. 24 [29].

HSV (Hue, Saturation, Value) color segmentation method with shape (rectangle) and area detection has been used with Kinect V2 to find the required handle. Also, a mark with specific features can be fixed on

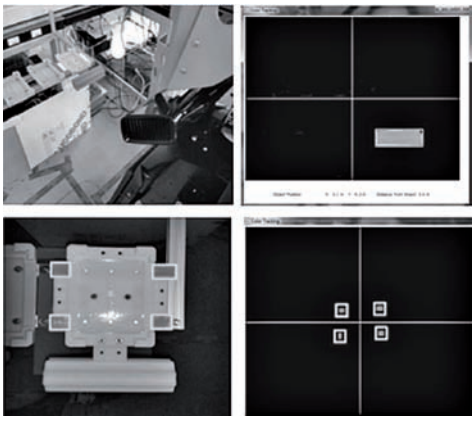


Fig. 23. The detection of handle and placing holder [28]

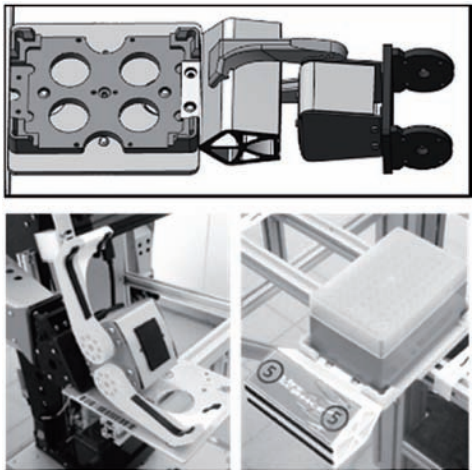


Fig. 24. The design of grippers and handle [29]

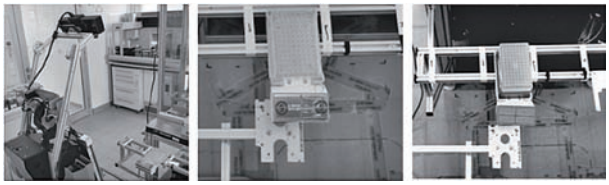


Fig. 25. Handle detection using SURF and HSV [29]

the handle to be recognized using SURF (Speeded-Up Robust Features) algorithm. A polygon with cross is drawn around the target to define it and to identify its center point as shown in Fig. 25 [29].

The detection and localization strategies have been applied also to find the holder related to the required position for labware placing tasks. HSV can be considered the most powerful system to be used for color segmentation because it is more robust to the changes of lighting conditions in comparison with RGB color system. The high resolution of the RGB and depth cameras of Kinect V2 make it very desired to be used for object detection and localization. The RGB camera of Kinect V2 captures color frames with a resolution of 1920×1080 pixels, whereas the IR camera, which is used for depth frame acquisition, has 512×424 pixels resolution.

Using the design shown in Fig. 24, about 350 g can be manipulated. To manipulate heavier labware visually, it is complex to use the vertical handle shown in Fig. 20. It is difficult to identify multiple vertical handles in the view due to their design. To cope with this issue, the required torque, which the wrist joint has to provide for lifting the labware, has to be decreased. This can be achieved by removing the handle attached to labware container to decrease the lever arm of the wrist joint. In this case, the labware weight center will be closer to the wrist. New fingers with labware containers have been designed for this goal as shown in Fig. 26 [30]. The maximum payload which can be handled with this design is 700 g.

To perform the visual manipulation using this design, the labware itself has to be recognized and localized. Since the labwares have transparent or white lids to protect the components from cross contamination, it is not applicable to recognize and differentiate them on the workbench. To cope with this issue, a specific mark has been fixed on each labware lid as shown in Fig. 27. The mark image gives adequate features to differentiate multiple labware. Different labware marks have been recognized and localized using Kinect sensor V2 with SURF algorithm [31]. The recognition of the labware is assigned by drawing a polygon around its mark with cross to specify the center point.

After the step of target recognition, the center point of this target is obtained. Then, the position of this center point related to Kinect is found using a mapping process. The required point is mapped from the color frame space to the Kinect space coordinates. The position of the center point is used as reference for estimating the grasping or placing point positions where the arm end effector has to reach [30]. To move the robotic arm to the goal, an extrinsic calibration has to be applied. The purpose of this step is to transform the position information from the Kinect space to be related to the arm shoulder space. Then, the inverse kinematic model is used to control the arm joints and guide the end effector to the target. The Kinect-to-shoulder transformation includes the difference in the position and the tilt angle (t) between them according to the Kinect holder as show in Fig. 28. This transformation is vital to use the visual input as a reference for manipulation or interaction.

The Kinect-to-shoulder transformation consists of two steps. The first is the transformation from the Kinect sensor to the hinge, then, the transformation

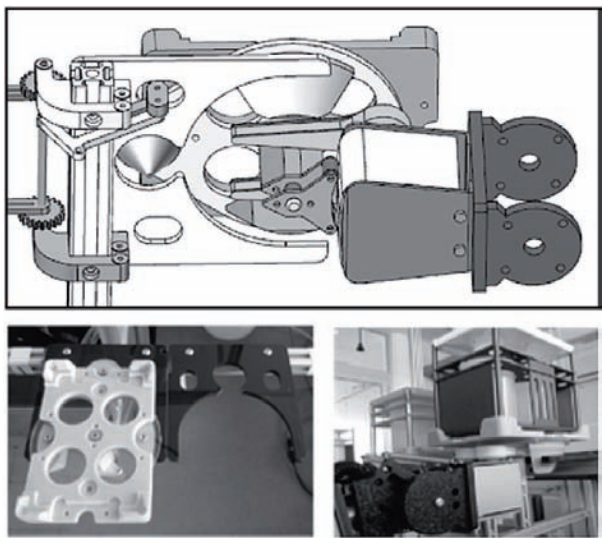


Fig. 26. Finger and labware container design [30]

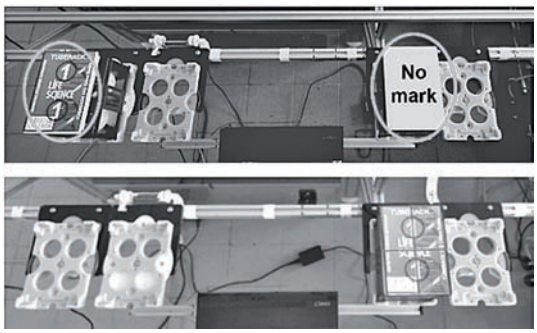


Fig. 27. Labware lid with and without mark [30]

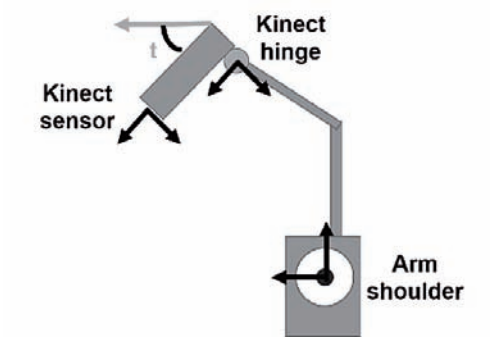


Fig. 28. Kinect-to-shoulder transformation

from hinge to shoulder. The transformation matrices are as follows:

$$ {}^{k}_{h}T = \begin{bmatrix} 1 & 0 & 0 & a \\ 0 & 1 & 0 & b \\ 0 & 0 & 1 & c \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad {}^{h}_{sh}T = \begin{bmatrix} cost & -sint & 0 & d \\ sint & cost & 0 & e \\ 0 & 0 & 1 & f \\ 0 & 0 & 0 & 1 \end{bmatrix}, $$

where, a,b,c represent the position differences in x,y,z axes between the Kinect and hinge. On the other hand, d,e,f represent the position difference in x,y,z axes between the hinge and arm shoulder. The tilt angle is represented by `t`. The transformation from Kinect sensor to hinge is easier in the process because the changes are just in the translation but not in the rotation. To find the final matrix to be inserted to the IK model, the process shown in Fig. 29 is performed.

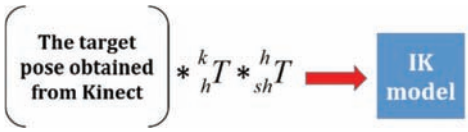


Fig. 29. The final matrix calculation process

The required time for performing the visual grasping is about 69 seconds while about 59 seconds are required for the visual placing.

10. Conclusion

In this paper, the forward and inverse kinematics solution for the H20 robot arms have been derived. Also, the IK solutions for different singularity cases have been found. The reverse decoupling mechanism method has been used to solve the IK problem analytically. The derived solution of the IK problem can be used for any other robotic arm which has the same joints structure and coordinate frames. A decision model has been used to select the desired joint values within multiple choices. Computer simulations have been used to validate the IK solution and to calculate the reachable workspace of the H20 arms. Two labware manipulation strategies have been performed using the sonar sensor and the Kinect sensor.

ACKNOWLEDGEMENTS

This work was funded by the Federal Ministry of Education and Research (FKZ: 03Z1KN11, 03Z1KI1) and the German Academic Exchange Service (Ph.D. stipend M. M. Ali). The authors would also like to thank the Canadian DrRobot Company for the technical support of the H20 mobile robots in this study.

AUTHORS

Mohammed M. Ali* – Center for Life Science Automation (CELISCA), University of Rostock, Rostock 18119, Germany,
mohammed.myasar.ali@celisca.de.

Hui Liu – Center for Life Science Automation (CELISCA), University of Rostock, Rostock 18119, Germany, Hui.Liu@celisca.de.

Norbert Stoll – Institute of Automation, University of Rostock, Rostock 18119, Germany,
Norbert.Stoll@uni-rostock.de.

Kerstin Thurow – Center for Life Science Automation (CELISCA), University of Rostock, Rostock 18119, Germany,
Kerstin.Thurow@celisca.de.

*Corresponding author

REFERENCES

[1] Chung H., Hou C., Chen Y., Chao C., "An intelligent service robot for transporting object". In: *IEEE International Symposium on Industrial Electronics (ISIE)*, Taipei, Taiwan, 2013, 1–6.

[2] Ciocarlie M., Hsiao K., Jones E. G., Chitta S., Rusu R. B., Şucan I. A., "Towards reliable grasping and manipulation in household environments". In: *12th International Symposium on Experimental Robotics (ISER)*, Springer Berlin Heidelberg, 2014, 241–252. DOI: 10.1007/978-3-642-28572-1_17.

[3] Graf B., Reiser U., Hägele M., Mauz K., Klein P., "Robotic home assistant Care-O-bot® 3-product vision and innovation platform". In: IE*EE Workshop on Advanced Robotics and its Social Impacts (ARSO)*, Tokyo, Japan, 2009, 139–144.

[4] Vahrenkamp N., Berenson D., Asfour T., Kuffner J., Dillmann R., "Humanoid motion planning for dual-arm manipulation and re-grasping tasks". In: IEEE/RSJ *International Conference on Intelligent Robots and Systems (IROS)*, St. Louis, USA, 2009, 2464–2470. DOI: 10.1109/IROS.2009.5354625.

[5] O'Flaherty R., Vieira P., Grey M. X., Oh P., Bobick A., Egerstedt M., Stilman M., "Humanoid robot teleoperation for tasks with power tools". In: *IEEE International Conference on Technologies for Practical Robot Applications (TePRA)*, Woburn, MA, 2013, 1–6. DOI: 10.1109/TePRA.2013.6556362.

[6] Tsay T. J., Hsu M. S., Lin R. X., "Development of a mobile robot for visually guided handling of materia". In: *IEEE International Conference on Robotics and Automation (ICRA)*, Taipei, Taiwan, 2003, 3397–3402.

[7] Liu H., Toll N. S, Junginger S., Thurow K., "A common wireless remote control system for mobile robots in laboratory". In: I*EEE Conference on Instrumentation and Measurement Technology (I2MTC*), Graz, Austria, 2012, 688–693.

[8] Liu H., Stoll N., Junginger S., Thurow K., "Mobile Robot for Life Science Automation," *International Journal of Advanced Robotic Systems*, vol. 10, 2013, 1–14. DOI: 10.5772/56670.

[9] Abdulla A. A., Liu H., Stoll N., Thurow K., "A New Robust Method for Mobile Robot Multifloor Navigation in Distributed Life Science Laboratories," *J. Control Sci. Eng.*, vol. 2016, Jul. 2016, 1–17. DOI: 10.1155/2016/3589395.

[10] Iqbal J., ul Islam R., Khan H., "Modeling and Analysis of a 6 DOF Robotic Arm Manipulator", *Canadian Journal on Electrical and Electronics Engineering*, vol. 3, 2012, 300–306.

[11] Tevatia G., Schaal S., "Inverse kinematics for humanoid robots". In: *IEEE International Conference on Robotics and Automation (ICRA)*, San Francisco, CA, USA, 2000, 294–299. DOI: 10.1109/ROBOT.2000.844073.

[12] Mistry M., Nakanishi J., Cheng G., Schaal S., "Inverse kinematics with floating base and constraints for full body humanoid robot control". In: *IEEE-RAS International Conference on Humano Robots*, Daejeon, Korea, 2008, 22–27. DOI: 10.1109/ICHR.2008.4755926.

[13] Nie L., Huang Q., "Inverse kinematics for 6-DOF manipulator by the method of sequential retrieval". In: *Proceedings of the International Conference on Mechanical Engineering and Material Science,* China, 2012, 255–258.

[14] Pieper D. L., *The kinematics of manipulators under computer control*, Ph.D. Dissertation, Stanford University, 1968.

[15] Lee C. G. S., Ziegler M., "Geometric approach in solving inverse kinematics of PUMA robots," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 20, 1984, 695–706. DOI: 10.1109/TAES.1984.310452.

[16] Ho T., Kang, C.-G., Lee S., "Efficient closed-form solution of inverse kinematics for a specific six-DOF arm", *International Journal of Control Systems and Automation*, vol. 10, no. 3, 2012, 567–573. DOI: 10.1007/s12555-012-0313-9.

[17] Huang G.-S., Tung C.-K., Lin H.-C., Hsiao S.-H., "Inverse kinematics analysis trajectory planning for a robot arm". In: *8th Asian Control Conference (ASCC)*, Kaohsiung, Taiwan, 2011, 965–970.

[18] Man C.-H., Fan X., Li C.-R., Zhao Z.-H., "Kinematics analysis based on screw theory of a humanoid robot," *Journal of China University of Mining and Technology*, vol. 17, no. 1, 2007, 49–52. DOI: 10.1016/S1006-1266(07)60011-X.

[19] Paul R. P., Shimano B. E., Mayer G., "Kinematic control equations for simple manipulators", *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 11, 1981, 449–455. DOI: 10.1109/CDC.1978.268148.

[20] Zhao T., Yuan J., Zhao M., Tan D., "Research on the Kinematics and Dynamics of a 7-DOF Arm of Humanoid Robot". In: *IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Kunming, China, 2006, 1553–1558. DOI: 10.1109/ROBIO.2006.340175.

[21] Ali M. A., Park H. A., Lee C. G., "Closed-form inverse kinematic joint solution for humanoid robots". In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Taipei, Taiwan, 2010, 704–709.

[22] O'Flaherty R., Vieira P., Grey M., Oh P., A. Bobick, Egerstedt M., Stilman M., "Kinematics and Inverse Kinematics for the Humanoid Robot HUBO2," Georgia Institute of Technology, Atlanta, GA, USA, Technical Report, 2013.

[23 Borenstein J., Everett H. R., Feng L., "Where am I? Sensors and methods for mobile robot positioning," University of Michigan, USA, 1996.

[24] Borenstein J., "The CLAPPER: A dual-drive mobile robot with internal correction of dead-reckoning errors". In: *IEEE International Conference on Robotics and Automation*, San Diego, CA, 1994, 3085–3090. DOI: 10.1109/ROBOT.1994.351095.

[25] Lysenkov I., Rabaud V., "Pose estimation of rigid transparent objects in transparent clutter". In: I*EEE Conference on Robotics and Automation (ICRA)*, Karlsruhe, Germany, 2013, 162–169. DOI: 10.1109/ICRA.2013.6630571.

[26] Denavit J., Hartenberg R. S., "A kinematic notation for lower-pair mechanisms based on matrices," *ASME Journal of Applied Mechanics*, vol. 22, 1955, 215–221.

[27] Conrad K. L., Shiakolas P. S., Yih T. C., "Robotic calibration issues: Accuracy, repeatability and calibration". In: *Proceedings of the 8th Mediterranean Conference on Control and Automation (MED2000)*, Rio, Patras, Greece, 2000.

[28] Ali M. M., Liu H., Stoll R., Thurow K., "Arm grasping for mobile robot transportation using Kinect sensor and kinematic analysis". In: *IEEE International Conference on Instrumentation and Measurement Technology (I2MTC)*, Pisa, Italy, 2015, 516–521. DOI: 10.1109/I2MTC.2015.7151321.

[29] Ali M. M., Liu H., Stoll R., Thurow K., "Intelligent Arm Manipulation System in Life Science Labs Using H20 Mobile Robot and Kinect Sensor". In: *IEEE International Conference on Intelligent Systems (IS'16)*, Sofia, Bulgaria, 2016, 382–387.

[30] Ali M. M., Liu H., Stoll R., "Multiple Lab Ware Manipulation in Life Science Laboratories using Mobile Robots". In: *IEEE International Conference on Mechatronics*, Prague, Czech Republic, 2016, 415–421.

[31] Ali M. M., H. Liu, N. Stoll, and K. Thurow, "An Identification and localization Approach of Different Labware for Mobile Robot Transportation in Life Science laboratories. In *IEEE International Symposium on Computational Intelligence and Informatics*, Budapest, Hungary, 2016, 353–358.