

# Modular Relative Jacobian for Combined 3-Arm Parallel Manipulators

Rodrigo S. Jamisola Jr., Carlos Mastalli and Frank Ibikunle

**Abstract**—This work presents a new formulation of a modular relative Jacobian used to control combined manipulators as a single manipulator with a single effector. In particular, this modular relative Jacobian is designed for 3-arm parallel manipulators. It is called a relative Jacobian because it is expressed relative to the reference frames at the manipulator end-effectors. It is modular because it uses the existing information of each standalone manipulator component to arrive at the necessary expressions for the combined system. This work is part of a series of studies to express a single end-effector control of combined manipulators, in parallel as well as other types of base configurations. This holistic approach of controlling combined manipulators affords a drastic increase of the null-space dimension and the convenience to use all the principles of controlling a single manipulator for the resulting combined system. Derivation of the modular relative Jacobian for a 3-arm parallel manipulator is shown, together with simulation results.

**Index Terms**—Modular kinematics, 3-arm parallel manipulators, single end-effector control, relative Jacobian

## I. INTRODUCTION

Modularity of manipulator kinematics and dynamics expressions have been recently actively studied because of the increasing complexity of robot structures. In particular, robots are no longer confined to single manipulators structures, but are now consisting of two or more manipulators combined together to form one single robot, like dual-arms, humanoids, quadrupeds, hexapods, etc. Modular approach in the study of parallel robots has been used to many different types of applications, which include modular micro parallel robots [1], modular control architecture [2], modular design of parallel robots [3], kinematics and design of two variants [4], modular, wire-driven parallel robots [5], [6], design of modular parallel robots [7], multi-robot system ARGoS [8], and reconfigurable parallel robots [9], to name a few.

This study proposes to control combined 3-arm parallel manipulators (shown in Fig 1) as a one single robot with a single end-effector. The advantage of this type of controller is two-fold: (1) it drastically increase the null-space dimension of the resulting combined manipulators and (2) the principles of single manipulator control can now be applied to the combined manipulators. For example, consider a dual-arm

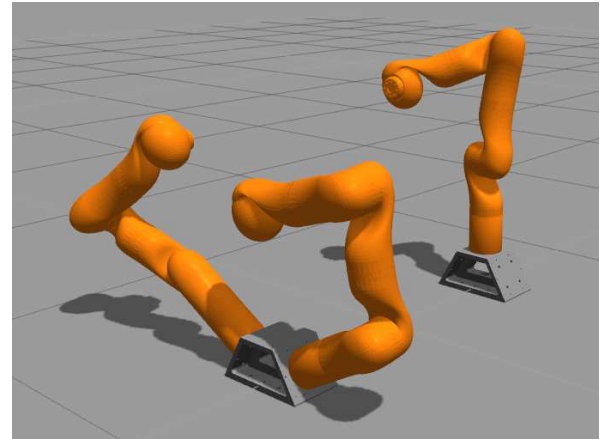


Fig. 1. A 3-arm parallel manipulator. Simulation video is shown here: <https://youtu.be/w87Ei7Z2Uis>.

robot with each arm having six degrees-of-freedom (6-DOF). When each of the two arms is independently controlled in the full space, the resulting dual-arm robot is non-redundant. However, if the two arms are controlled in the relative full space, the resulting dual-arm robot has 6-DOF in the null space. In addition, through the single end-effector control, the combined manipulators can use a single manipulator controller, such that a strict task prioritization can now be implemented throughout the entire system.

Secondly, this study proposes modularity of the derived relative Jacobian. for the 3-arms. Modularity of the derived expression adds to the ease of implementation of the proposed single end-effector control. This is because the existing information of each of the standalone manipulator components will be used to arrive at the resulting expressions for the combined systems. Rotation and wrench transformation matrices are used to transform each of the standalone Jacobians to arrive at the relative Jacobian of the 3-arms.

A more compact modular relative Jacobian was first shown in [10], which reveals a wrench transformation matrix that was not present or was not explicitly expressed in the previous relative Jacobians. It was further shown that omission of the wrench transformation matrix can affect the performance of the dual-arm, including the forces and moments exerted at the end-effectors [11]. The concept of a relative Jacobian was first introduced in [12], [13]. A recent application of the relative Jacobian to asymmetric bimanual task was shown [14].

R. S. Jamisola Jr. and F. Ibikunle are with Electrical, Electronics and Telecommunications Engineering Department, Botswana International University of Science and Technology (BIUST), Private Bag 16, Palapye, Botswana 10071 {jamisolar, ibikunlef}@biust.ac.bw.

C. Mastalli is with Department of Advanced Robotics, Istituto Italiano di Tecnologia (IIT), Via Morego 30, 16163 Genova, Italy {carlos.mastalli}@iit.it.

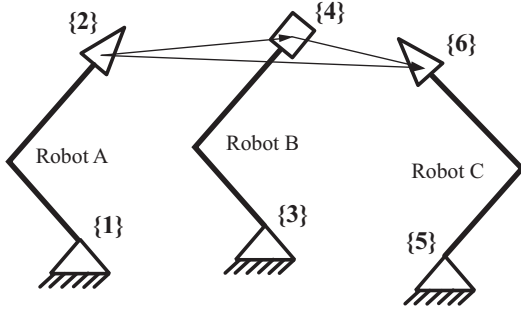


Fig. 2. An schematic diagram of a 3-arm parallel manipulator, with the corresponding reference frames and the relative position vectors.

## II. NAMING CONVENTION FOR SYMBOLS

In Fig. 2, the schematic diagram of a 3-arms parallel manipulator is shown, together with the corresponding reference frames. The base reference frames are odd-numbered, while the end-effector reference frames are even-numbered. The relative position vectors are also shown.

Consider reference frames  $\{i\}$  and  $\{j\}$ , such that  ${}^i p_j$  is the position of frame  $\{j\}$  with respect to frame  $\{i\}$ , and  ${}^i R_j$  is the rotation of frame  $\{j\}$  with respect to frame  $\{i\}$ . In addition, a Jacobian  ${}^i J_j$  can be expressed with respect to those frames. From the figure, we state the following conventions for the Jacobians of the standalone manipulators. The Jacobian for robot A is  ${}^1 J_2$ , for robot B is  ${}^3 J_4$  and for robot C is  ${}^5 J_6$ , each is expressed with respect to the indicated reference frame indices.

We assign the position Jacobian  ${}^i J_{pj}$  and orientation Jacobian  ${}^i J_{oj}$  as components of the Jacobian  ${}^i J_j$ , that is,  ${}^i J_j = [{}^i J_{pj}, {}^i J_{oj}]^T$ . The joint velocities  $\dot{q}_{ij} = [\dot{q}_i, \dot{q}_j]^T$ , such  $\dot{q}_i$  and  $\dot{q}_j$  are the joint velocities of the robot with end-effector frames  $\{i\}$  and  $\{j\}$ , respectively. For example  ${}^1 J_2 = [{}^1 J_{p2}, {}^1 J_{o2}]^T$  is the Jacobian for robot A, and  ${}^2 J_4 = [{}^2 J_{p4}, {}^2 J_{o4}]^T$  is the relative Jacobian of the dual-arm consisting of robots A and B. The dual-arm joint velocities  $\dot{q}_{24} = [\dot{q}_2, \dot{q}_4]^T$ , where  $\dot{q}_2$  are the joint velocities of robot A  $\dot{q}_4$  are the joint velocities of robot B. Naming convention for symbols and most symbols used in this work are shown in Table I.

## III. DERIVATION OF THE MODULAR 3-ARM RELATIVE JACOBIAN

In this section, we present the derivation of the modular relative Jacobian for three parallel manipulators. We derive the modular relative Jacobian of the 3-arm parallel manipulator by expressing the end-effector of the robot C with respect to the end-effector of robot A. We will do this by taking two robots at a time, the same method that was invoked for the modular relative Jacobian of a dual-arm as derived in [10]. We show here the modular relative Jacobian for dual-arms as shown in [10], such that the relative Jacobian for a dual-arm consisting of robots A and B in Fig. 2 is

$${}^2 J_4 = [-{}^2 \Psi_4 {}^2 \Omega_1 {}^1 J_2 \quad {}^2 \Omega_3 {}^3 J_4], \quad (1)$$

TABLE I  
SYMBOLS-NAMING CONVENTION

Sym.	Description
${}^i p_j$	position of frame $\{j\}$ w.r.t. frame $\{i\}$ ; its first derivative is ${}^i \dot{p}_j$
${}^i R_j$	orientation of $\{j\}$ w.r.t. $\{i\}$ ; its first derivative is ${}^i \dot{R}_j$
${}^i \omega_j$	rotational velocity of $\{j\}$ w.r.t. $\{i\}$
${}^i J_j$	$[{}^i J_{pj}, {}^i J_{oj}]^T$ Jacobian from $\{i\}$ to $\{j\}$
${}^i J_{pj}$	position component of Jacobian ${}^i J_j$
${}^i J_{oj}$	orientation component of Jacobian ${}^i J_j$
${}^1 J_2$	Jacobian of robot A
${}^3 J_4$	Jacobian of robot B
${}^5 J_6$	Jacobian of robot C
${}^2 J_4$	relative Jacobian of dual-arm robots A and B
${}^4 J_6$	relative Jacobian of dual-arm robots B and C
${}^2 J_6$	relative Jacobian of dual-arm robots A and C
${}^3 J_6$	relative Jacobian of 3-arm robots A, B and C
$\dot{q}_2$	joint velocities of robot A
$\dot{q}_4$	joint velocities of robot B
$\dot{q}_6$	joint velocities of robot C
$\dot{q}_{24}$	$[\dot{q}_2, \dot{q}_4]^T$ joint velocities of dual-arm robots A and B
$\dot{q}_{46}$	$[\dot{q}_4, \dot{q}_6]^T$ joint velocities of dual-arm robots B and C
$\dot{q}_{246}$	$[\dot{q}_2, \dot{q}_4, \dot{q}_6]^T$ joint velocities of 3-arm robots A, B, and C
${}^2_3 p_6$	3-arm relative position of $\{6\}$ w.r.t. $\{2\}$
${}^2_3 \dot{p}_6$	3-arm relative translational velocity of $\{6\}$ w.r.t. $\{2\}$
${}^2_3 \omega_6$	3-arm relative rotational velocity of $\{6\}$ w.r.t. $\{2\}$

and the relative Jacobian of a dual-arm consisting of robots B and C in Fig. 2 is

$${}^4 J_6 = [-{}^4 \Psi_6 {}^4 \Omega_3 {}^3 J_4 \quad {}^4 \Omega_5 {}^5 J_6]. \quad (2)$$

Lastly, the relative Jacobian for dual-arm robots A and C is

$${}^2 J_6 = [-{}^2 \Psi_6 {}^2 \Omega_1 {}^1 J_2 \quad {}^2 \Omega_5 {}^5 J_6]. \quad (3)$$

Such that the wrench transformation matrix  ${}^i \Psi_j$  is defined as

$${}^i \Psi_j = \begin{bmatrix} I & -S({}^i p_j) \\ 0 & I \end{bmatrix} \quad (4)$$

and the rotation matrix  ${}^i \Omega_j$  is expressed as

$${}^i \Omega_j = \begin{bmatrix} {}^i R_j & 0 \\ 0 & {}^i R_j \end{bmatrix}. \quad (5)$$

Given  $\omega = [\omega_x, \omega_y, \omega_z]^T$ , the operator  $S(\omega)$  is the skew symmetric operator used to replace the cross-product operator and is expressed as

$$S(\omega) = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix}. \quad (6)$$

To complete the definition of the modular dual-arm manipulators the shown robots in Fig. 2, we define the relative position vectors between the end-effectors, called  ${}^i p_j$  for the paired robots. We express them here as

$$\begin{aligned} {}^2 p_4 &= {}^2 R_1 ({}^1 p_3 + {}^1 R_3 {}^3 p_4 - {}^1 p_2) \\ {}^4 p_6 &= {}^4 R_3 ({}^3 p_5 + {}^3 R_5 {}^5 p_6 - {}^3 p_4) \\ {}^2 p_6 &= {}^2 R_1 ({}^1 p_5 + {}^1 R_5 {}^5 p_6 - {}^1 p_2). \end{aligned} \quad (7)$$

Now we are ready to derive the modular relative Jacobian for the 3-arm parallel manipulator,  ${}^2_3J_6$ , by invoking a similar approach used in [10]. That is, we express translational and rotational velocities of the end-effectors with respect to each other. Thus the relative position of frame  $\{6\}$  with respect to frame  $\{2\}$  is expressed as

$${}^2_3p_6 = {}^2p_4 + {}^2R_4 {}^4p_6. \quad (8)$$

And taking the derivative of the above equation results in

$$\begin{aligned} {}^2_3\dot{p}_6 &= {}^2\dot{p}_4 + {}^2\dot{R}_4 {}^4p_6 + {}^2R_4 {}^4\dot{p}_6 \\ &= {}^2\dot{p}_4 + S({}^2\omega_4) {}^2R_4 {}^4p_6 + {}^2R_4 {}^4\dot{p}_6 \\ &= {}^2\dot{p}_4 - S({}^2R_4 {}^4p_6) {}^2\omega_4 + {}^2R_4 {}^4\dot{p}_6. \end{aligned} \quad (9)$$

Because angular velocities are linear, we can express the relative angular velocity of frame  $\{6\}$  with respect to frame  $\{2\}$  as

$${}^2_3\omega_6 = {}^2\omega_4 + {}^2R_4 {}^4\omega_6. \quad (10)$$

Combining (9) and (10), we get

$$\begin{bmatrix} {}^2_3\dot{p}_6 \\ {}^2_3\omega_6 \end{bmatrix} = \begin{bmatrix} {}^2p_4 - S({}^2R_4 {}^4p_6) {}^2\omega_6 + {}^2R_4 {}^4\dot{p}_6 \\ {}^2\omega_4 + {}^2R_4 {}^4\omega_6 \end{bmatrix} \quad (11)$$

We express the above expressions in terms of the relative Jacobians

$$\begin{aligned} &\begin{bmatrix} {}^2_3\dot{p}_6 \\ {}^2_3\omega_6 \end{bmatrix} \\ &= \begin{bmatrix} {}^2J_{p4}\dot{q}_{24} - S({}^2R_4 {}^4p_6) {}^2J_{o4}\dot{q}_{24} + {}^2R_4 {}^4J_{p6}\dot{q}_{46} \\ {}^2J_{o4}\dot{q}_{24} + {}^2R_4 {}^4J_{o6}\dot{q}_{46} \end{bmatrix} \\ &= \begin{bmatrix} I & -S({}^2R_4 {}^4p_6) \\ 0 & I \end{bmatrix} \begin{bmatrix} {}^2J_{p4} \\ {}^2J_{o4} \end{bmatrix} \dot{q}_{24} \dots \\ &\quad \dots + \begin{bmatrix} {}^2R_4 & 0 \\ 0 & {}^2R_4 \end{bmatrix} \begin{bmatrix} {}^4J_{p6} \\ {}^4J_{o6} \end{bmatrix} \dot{q}_{46} \\ &= \begin{bmatrix} I & -S({}^2R_4 {}^4p_6) \\ 0 & I \end{bmatrix} {}^2J_4 \begin{bmatrix} {}^2R_4 & 0 \\ 0 & {}^2R_4 \end{bmatrix} {}^4J_6 \begin{bmatrix} \dot{q}_{24} \\ \dot{q}_{46} \end{bmatrix} \\ &= [{}^{2,4}\Psi_6 {}^2J_4 \quad {}^2\Omega_4 {}^4J_6] \begin{bmatrix} \dot{q}_{24} \\ \dot{q}_{46} \end{bmatrix} \\ &= [{}^{2,4}\Psi_6 [-{}^2\Psi_4 {}^2\Omega_1 {}^1J_2 \quad {}^2\Omega_3 {}^3J_4] \dots \\ &\quad \dots {}^2\Omega_4 [-{}^4\Psi_6 {}^4\Omega_3 {}^3J_4 \quad {}^4\Omega_5 {}^5J_6]] \begin{bmatrix} \dot{q}_{24} \\ \dot{q}_{46} \end{bmatrix} \\ &= [-{}^{2,4}\Psi_6 {}^2\Psi_4 {}^2\Omega_1 {}^1J_2 \quad ({}^{2,4}\Psi_6 {}^2\Omega_3 - {}^2\Omega_4 {}^4\Psi_6 {}^4\Omega_3) {}^3J_4 \dots \\ &\quad \dots {}^2\Omega_4 {}^4\Omega_5 {}^5J_6] \begin{bmatrix} \dot{q}_{24} \\ \dot{q}_{46} \end{bmatrix}. \end{aligned} \quad (12)$$

In the second to the last equality of (12), we substitute the dual-arm relative Jacobians of (1) and (2). To further simplify, we group terms together, such that the modular relative Jacobian for a 3-arm parallel manipulator can be expressed as

$$\begin{aligned} &{}^2_3J_6 \\ &= [-{}^{2,4}\Psi_6 {}^2\Psi_4 {}^2\Omega_1 {}^1J_2 \quad ({}^{2,4}\Psi_6 {}^2\Omega_3 - {}^2\Omega_4 {}^4\Psi_6 {}^4\Omega_3) {}^3J_4 \dots \\ &\quad \dots {}^2\Omega_4 {}^4\Omega_5 {}^5J_6] \end{aligned} \quad (13)$$

where  ${}^{i,j}\Psi_k$  means that the wrench transformation matrix has the cross-product operator defined as  $S({}^iR_j {}^j p_k)$ .

We then need to simplify (13) column by column. We invoke Matlab matrix notation to do this. Thus the first column of  ${}^2_3J_6$  is

$$\begin{aligned} {}^2_3J_6(:,1) &= -{}^{2,4}\Psi_6 {}^2\Psi_4 {}^2\Omega_1 {}^1J_2 \\ &= -\begin{bmatrix} I & -S({}^2R_4 {}^4p_6) \\ 0 & I \end{bmatrix} \begin{bmatrix} I & -S({}^2p_4) \\ 0 & I \end{bmatrix} {}^2\Omega_1 {}^1J_2 \\ &= -\begin{bmatrix} I & -S({}^2p_4 + {}^2R_4 {}^4p_6) \\ 0 & I \end{bmatrix} {}^2\Omega_1 {}^1J_2 \\ &= -\begin{bmatrix} I & -S({}^2p_6) \\ 0 & I \end{bmatrix} {}^2\Omega_1 {}^1J_2 \\ &= -{}^2\Psi_6 {}^2\Omega_1 {}^1J_2. \end{aligned} \quad (14)$$

The second column of  ${}^2_3J_6$  is

$$\begin{aligned} &{}^2_3J_6(:,2) \\ &= ({}^{2,4}\Psi_6 {}^2\Omega_3 - {}^2\Omega_4 {}^4\Psi_6 {}^4\Omega_3) {}^3J_4 \\ &= \left( {}^{2,4}\Psi_6 {}^2\Omega_3 - \begin{bmatrix} {}^2R_4 & 0 \\ 0 & {}^2R_4 \end{bmatrix} \begin{bmatrix} I & -S({}^4p_6) \\ 0 & I \end{bmatrix} {}^4\Omega_3 \right) {}^3J_4 \\ &= \left( {}^{2,4}\Psi_6 {}^2\Omega_3 - \begin{bmatrix} {}^2R_4 & -{}^2R_4 S({}^4p_6) {}^2R_4^T {}^2R_4 \\ 0 & {}^2R_4 \end{bmatrix} {}^4\Omega_3 \right) {}^3J_4 \\ &= \left( {}^{2,4}\Psi_6 {}^2\Omega_3 - \begin{bmatrix} {}^2R_4 & -S({}^2R_4 {}^4p_6) {}^2R_4 \\ 0 & {}^2R_4 \end{bmatrix} {}^4\Omega_3 \right) {}^3J_4 \\ &= \left( {}^{2,4}\Psi_6 {}^2\Omega_3 - \begin{bmatrix} I & -S({}^2R_4 {}^4p_6) \\ 0 & I \end{bmatrix} \begin{bmatrix} {}^2R_4 & 0 \\ 0 & {}^2R_4 \end{bmatrix} {}^4\Omega_3 \right) {}^3J_4 \\ &= ({}^{2,4}\Psi_6 {}^2\Omega_3 - {}^{2,4}\Psi_6 {}^2\Omega_3) {}^3J_4 \\ &= 0. \end{aligned} \quad (15)$$

This make the relative Jacobian of the 3-arm parallel manipulator to be

$${}^2_3J_6 = [-{}^2\Psi_6 {}^2\Omega_1 {}^1J_2 \quad 0 \quad {}^2\Omega_5 {}^5J_6] \quad (16)$$

which is identical to (3), except for the middle zero column.

Thus, in this type of formulation, the third arm will always move in the null-space of the dual arm. A holistic modular kinematic expression for the 3-arm parallel manipulator can be expressed as

$$\begin{aligned} \dot{q}_{246} &= {}^2_3J_6^+ {}^2x_6 + (I - {}^2_3J_6^+ {}^2_3J_6) {}^2_3J_4^+ {}^2x_4 \dots \\ &\quad \dots + (I - {}^2_3J_6^+ {}^2_3J_6) (I - {}^2_3J_4^+ {}^2_3J_4) \nabla z \end{aligned} \quad (17)$$

where  $\dot{q}_{246} = [\dot{q}_2, \dot{q}_4, \dot{q}_6]^T$ ,  ${}^2_3J_4 = [{}^2J_4 \ 0]$ , and  $\nabla z$  is the null space posture. The null space projection of  $\nabla z$  can be computed as shown in [15], where maximum number of tasks was utilized and prioritized despite singularities. The expression in (17) shows that modularity of the kinematics expressions for null space is achieved in both end-effector and null-space motions.

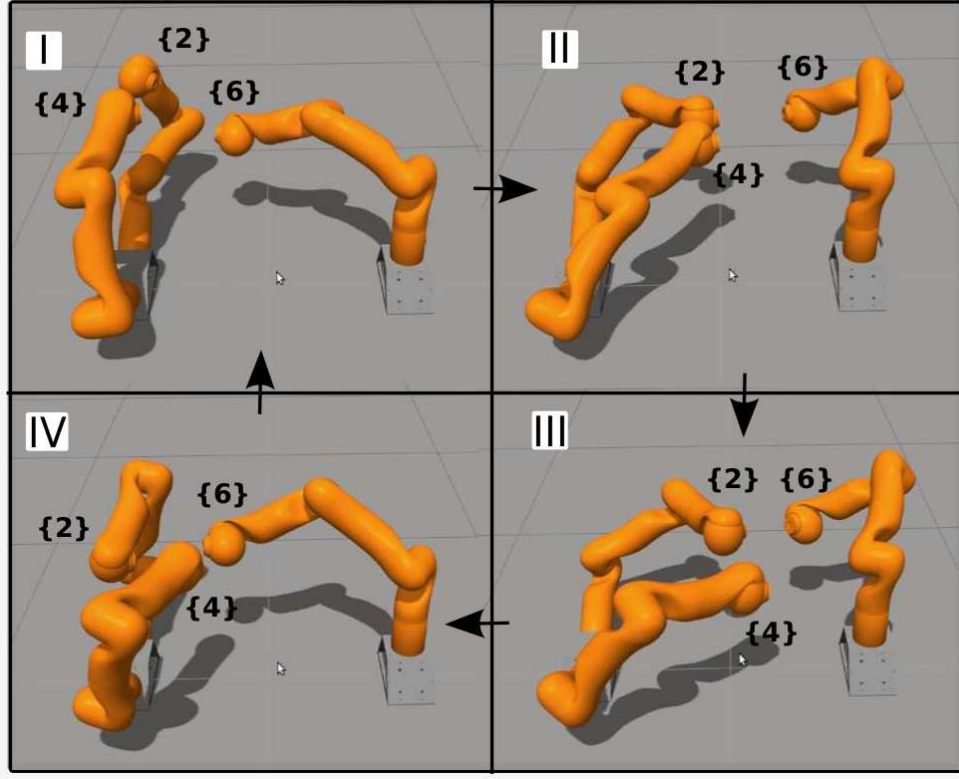


Fig. 3. Snapshots of Gazebo simulation where the 3-arm parallel manipulators move in a coordinated manner, in single manipulator control.

#### IV. SIMULATION USING GAZEBO

This section presents simulation results of a 3-arm parallel manipulator using the modular relative Jacobian derived in the previous section. Robot simulator Gazebo 2.2.5 is used as the simulation platform with Robot Operating Systems (ROS) Indigo. A Universal Robotic Description Format (URDF) of the 7-DOF KUKA LWR was created. The simulation is running under Ubuntu Trusty 14.04 LTS 64-bit with Intel Core i5-4210U quad-core processor.

##### A. Simulation Controller

The controller in the simulation is a controller with purely kinematic information, without any dynamics information included. This can be a limitation in the simulation. The simulator takes in torque inputs from the controller. And because the implemented control is purely a kinematics controller, the output of the velocity controller stated below was directly converted to torques and passed to the simulator. The velocity controller is expressed as

$$\begin{aligned} \dot{q}_{246} = & J_R^+ \Delta(x_R) + (I - J_R^+ J_R) \frac{1}{3} J_2^+ \Delta({}^1x_2) \\ & \dots + (I - J_R^+ J_R) (I - \frac{1}{3} J_2^+ \frac{1}{3} J_2) \frac{2}{3} J_4^+ \Delta({}^2x_4) \dots \\ & \dots + (I - J_R^+ J_R) (I - \frac{1}{3} J_2^+ \frac{1}{3} J_2) (I - \frac{2}{3} J_4^+ \frac{2}{3} J_4) \nabla z \end{aligned} \quad (18)$$

where  $J_R = \frac{2}{3} J_6$  and  $x_R = \frac{2}{3} x_6$  is the relative position and orientation vector. For the delta function, given  $x$  as the input,

$$\Delta(x) = k_p(x_d - x) + k_v(\dot{x}_d - \dot{x}) + k_I \sum_{t=0} (x_d - x) \quad (19)$$

where  $x_d$  is the desired  $x$ ,  $\dot{x}_d$  desired velocity of  $x_d$ ,  $\dot{x}$  is the velocity of  $x$ ,  $t$  is the time, and  $k_p$ ,  $k_v$ , and  $k_I$  are the proportional, derivative, and integral gains. The 3-arm null-space Jacobians are  $\frac{2}{3} J_4 = [\frac{2}{3} J_4 \ 0]$  and  $\frac{1}{3} J_2 = [\frac{1}{3} J_2 \ 0 \ 0]$ . The  $\nabla z$  is the null-space gradient that controls the posture of the arms, such that  $\nabla(z) = [\Delta(q_2), \Delta(q_4), \Delta(q_6)]^T$ .

For (18), the first term on the left hand side of the equation controls the relative motion of robot  $C$  end-effector with respect to the robot  $A$  end-effector, in a dual-arm kind of control strategy. In this approach, the end-effector of robot  $B$  lies in the null-space. The second term controls the relative motion of robot  $B$  end-effector with respect to robot  $A$  end-effector. The third term moves the robot  $A$  end-effector (which is the overall reference frame) with respect to the world frame. Because the end-effectors of robot  $B$  and  $C$  moves with respect to the robot  $A$  end-effector, all three end-effectors will move, as robot  $A$  end-effector is moving. The desired robot posture as defined in  $\nabla z$  is accommodated as long as it does not have any conflict with the three other higher priority tasks.

##### B. The Desired Values

The desired values are the following (with lengths in meters and angles in degrees):  $\frac{2}{3} x_{6d} = [0, 0, 0.3, 0, 180, 0]^T$  ( $x$ ,  $y$ , and  $z$  position and roll, pitch and yaw orientation),  $\frac{2}{3} x_{4d} = [-0.3, 0, 0, 0, 0, 0]^T$ ,  $q_{2d} = [0, +60, 0, -45, 0, -45, 0]^T$ ,  $q_{4d} = [0, -60, 0, +45, 0, +45, 0]^T$ , and  $q_{6d} = [0, +60, 0, -45, 0, -45, 0]^T$ . All desired velocities are zero. The desired values  ${}^1x_{2d}$  changes according in

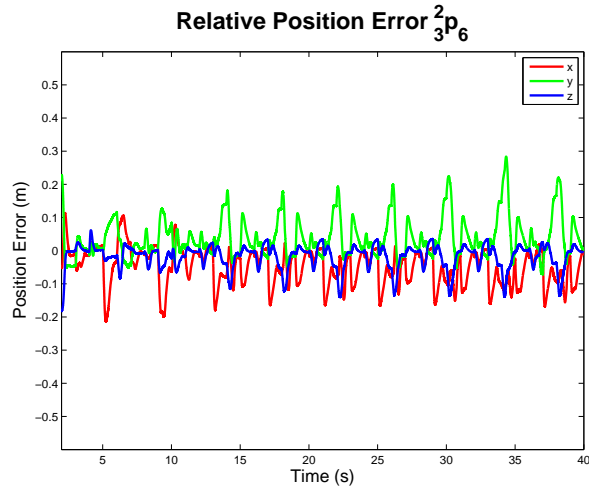


Fig. 4. The relative position error  ${}^2_3p_6$ .

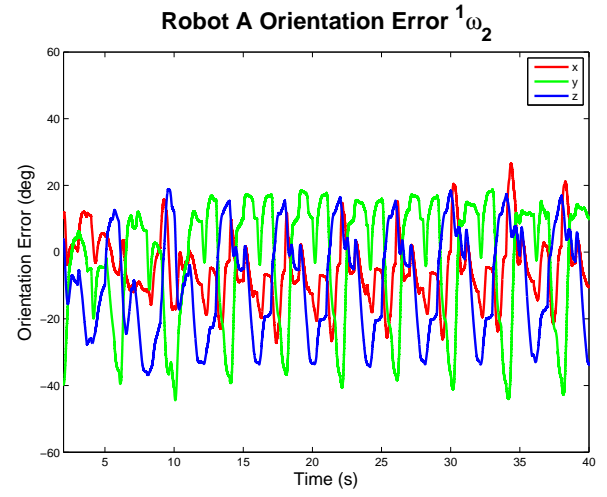


Fig. 7. The robot A orientation error  ${}^1\omega_2$ .

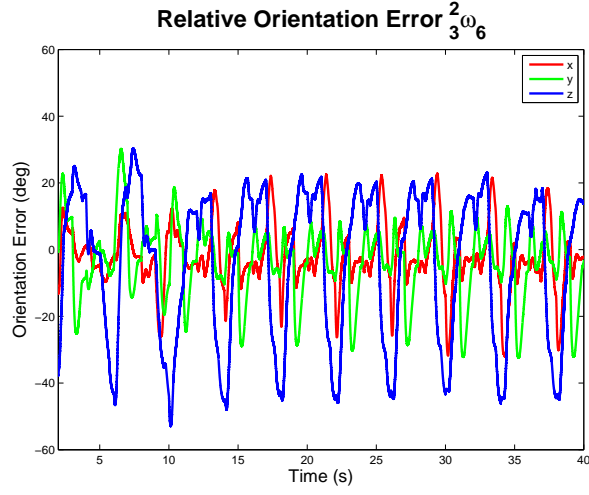


Fig. 5. The relative orientation error  ${}^2_3\omega_6$ .

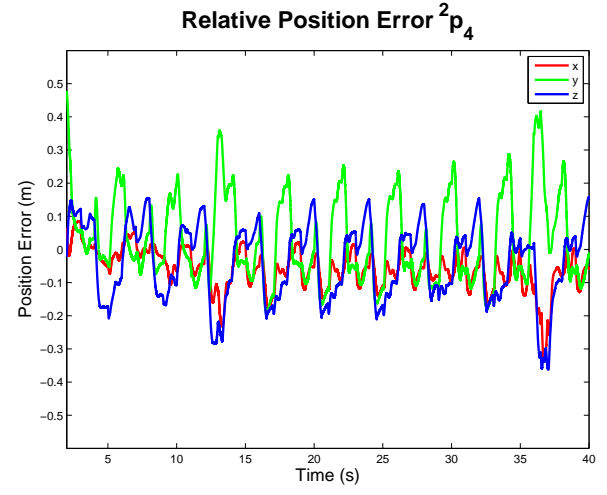


Fig. 8. The relative position error  ${}^2p_4$ .

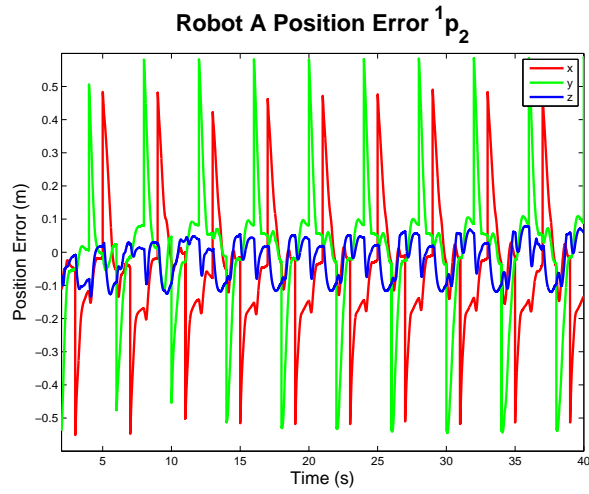


Fig. 6. The robot A position error  ${}^1p_2$ .

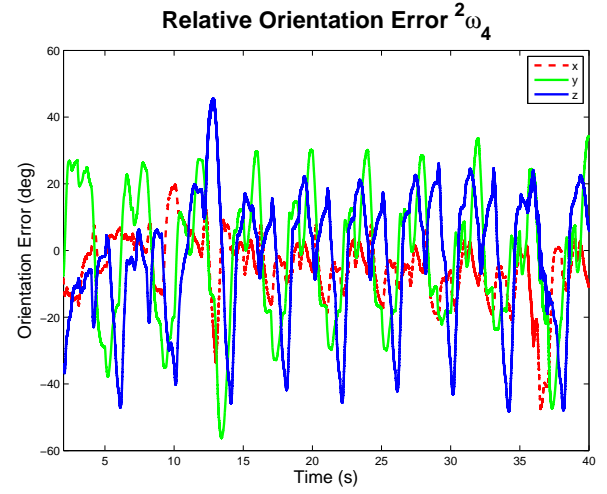


Fig. 9. The relative position error  ${}^2\omega_4$ .

a point-to-point motion to the time increment of 1s as follows:  ${}^1x_{2d} = [0.5, 0, 0.5, 90, 0, 0]^T_{t=0}$ ,  ${}^1x_{2d} =$



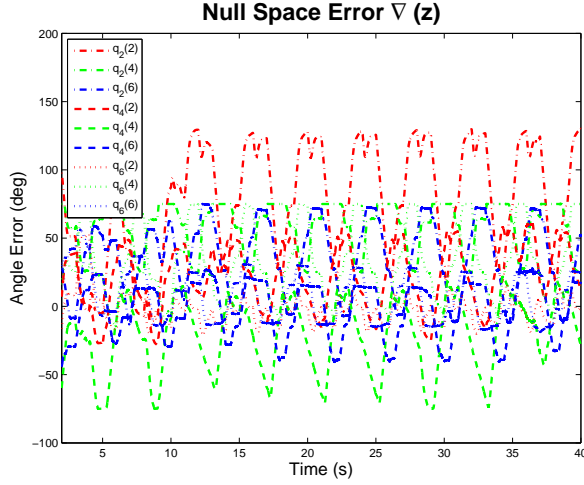


Fig. 10. The null space posture error  $\nabla(z)$ .

$[0.5, -0.5, 0.5, 90, 0, 0]_{t=1}^T$ ,  ${}^1x_{2d} = [0, -0.5, 0.5, 90, 0, 0]_{t=2}^T$ , and  ${}^1x_{2d} = [0, 0, 0.5, 90, 0, 0]_{t=3}^T$ . Then  ${}^1x_{2d}$  loops back in a 4s cycle of desired values. A simulation of the described desired motion is shown in Fig. 3.

Note that Gazebo simulator does not run in real-time. The gains are set at  $k_P(1:3) = 3000$  for position and  $k_P(4:6) = 1500$  for orientation,  $k_V = 200$ , and  $k_I = 0.1$ . Note that the  $\Delta(q)$  function in the null-space used  $k_P = 200$ , and  $k_V = k_I = 0$ . Now we are ready to show the simulation results.

### C. Numerical Results

The error results of the numerical simulation from the Gazebo simulation in Fig. 3 are shown from Fig. 4 to 10. The end-effector of robot A (frame {2}) is the reference frame for the motion of robot C end-effector (frame {6}) and motion of robot B end-effector (frame {4}). However, motion of {6} with respect to {2} is the highest priority, as shown in (18). Second priority is the motion of {2} with respect to its base, third priority is the motion of {4} with respect to {2}, and last priority is the null-space posture of the joints.

For the entire motion, only {2} is controlled to move at an unending square path while the relative position and orientation of {6} and {4} with respect to {2} is fixed at the desired values. A video of the experiment is shown here: <https://youtu.be/w87Ei7Z2Uis>. The resulting motion is that all the end-effectors are moving as a results of specified relative motion, according to the hierarchy of task prioritization of a single manipulator control.

Thus, the least position error is reflected by the error in  ${}^2_3p_6$  shown in Fig. 4, the task with the highest priority. In most cases, the task with the higher priority has the least errors compared the less priority tasks, except when at certain manipulator configurations that are difficult to achieve for the given desired values. The results of this simulation can be greatly improved when the dynamics of the system is compensated or canceled in the controller.

## V. CONCLUSION

This work derived a modular relative Jacobian of a 3-arm parallel manipulator, based on the dual-arm relative Jacobian approach of computation. In this new expression, the Jacobian of the third manipulator always lie in the null of the the overall Jacobian. In addition, it was shown that this approach affords a task prioritization control that is effectively a single manipulator, and thus task prioritization can be strictly implemented.

## REFERENCES

- [1] C. Ng, S. Ong, and A. Y. Nee, "Design and development of 3-dof modular micro parallel kinematic manipulator," *The International Journal of Advanced Manufacturing Technology*, vol. 31, no. 1-2, pp. 188–200, 2006.
- [2] J. Maaß, N. Kohn, and J. Hesselbach, "Open modular robot control architecture for assembly using the task frame formalism," *International Journal of advanced robotic systems*, vol. 3, no. 1, pp. 1–10, 2006.
- [3] A. E. Graham, S. Q. Xie, K. C. Aw, W. Xu, and S. Mukherjee, "Design of a parallel long bone fracture reduction robot with planning treatment tool," in *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*. IEEE, 2006, pp. 1255–1260.
- [4] D. Pîslă, N. Plitea, A. Vidrean, B. Prodan, B. Gherman, and D. Lese, "Kinematics and design of two variants of a reconfigurable parallel robot," in *Reconfigurable Mechanisms and Robots, 2009. ReMAR 2009. ASME/IFToMM International Conference on*. IEEE, 2009, pp. 624–631.
- [5] J.-P. Merlet, "Marionet, a family of modular wire-driven parallel robots," in *Advances in Robot Kinematics: Motion in Man and Machine*. Springer, 2010, pp. 53–61.
- [6] J.-P. Merlet and D. Daney, "A portable, modular parallel wire crane for rescue operations," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010, pp. 2834–2839.
- [7] C. E. Syrseloudis, I. Z. Emiris, T. Lilas, and A. Maglara, "Design of a simple and modular 2-dof ankle physiotherapy device relying on a hybrid serial-parallel robotic architecture," *Applied Bionics and Biomechanics*, vol. 8, no. 1, pp. 101–114, 2011.
- [8] C. Pincirolì, V. Trianni, R. OGrady, G. Pini, A. Brutschy, M. Brambilla, N. Mathews, E. Ferrante, G. Di Caro, F. Ducatelle, *et al.*, "Argos: a modular, parallel, multi-engine simulator for multi-robot systems," *Swarm intelligence*, vol. 6, no. 4, pp. 271–295, 2012.
- [9] G. J. Hamlin and A. C. Sanderson, *Tetrobot: A modular approach to reconfigurable parallel robotics*. Springer Science & Business Media, 2013, vol. 423.
- [10] R. S. Jamisola and R. G. Roberts, "A more compact expression of relative jacobian based on individual manipulator jacobians," *Robotics and Autonomous Systems*, vol. 63, pp. 158–164, 2015.
- [11] R. S. Jamisola Jr, P. Kormushev, D. G. Caldwell, and F. Ibikunle, "Modular relative jacobian for dual-arms and the wrench transformation matrix."
- [12] C. Lewis and A. Maciejewski, "Trajectory generation for cooperating robots," in *Systems Engineering, 1990., IEEE International Conference on*, 1990, pp. 300–303.
- [13] C. Lewis, "Trajectory generation for two robots cooperating to perform a task," in *Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on*, vol. 2, Apr 1996, pp. 1626–1631.
- [14] J. Lee, P. Chang, and R. S. Jamisola, "Relative impedance control for dual-arm robots performing asymmetric bimanual tasks," *Industrial Electronics, IEEE Transactions on*, vol. 61, no. 7, pp. 3786–3796, 2014.
- [15] R. S. Jamisola, P. H. Chang, and J. Lee, "Guaranteeing task prioritization for redundant robots given maximum number of tasks and singularities," in *TENCON 2012-2012 IEEE Region 10 Conference*. IEEE, 2012, pp. 1–6.