Proceedings of the ASME 2023
International Design Engineering Technical Conferences and Computers and Information in Engineering Conference
IDETC-CIE2023
August 20-23, 2023, Boston, Massachusetts

DETC2023-114992

Design and System Identification of an Actuation-Coordinated Mobile Parallel Robot with Hybrid Mobile and Manipulation Motion

Han Lin
Purdue University
West Lafayette, IN

Jiayu Luo
Purdue University
West Lafayette, IN

Xiaotong Huang
Purdue University
West Lafayette, IN

Haoguang Yang
Purdue University
West Lafayette, IN

Jiaming Fu
Purdue University
West Lafayette, IN

Richard M. Voyles
Purdue University
West Lafayette, IN

Dongming Gan*
Purdue University
West Lafayette, IN
*corresponding: dgan@purdue.edu

ABSTRACT

This paper presents the development of a novel Actuation-Coordinated Mobile Parallel Robot (ACMPR), with a focus on studying the kinematics of the mobile parallel robot with three limbs (3-mPRS) comprising mobile prismatic joint-revolute joint-spherical joint. The objective of this research is to explore the feasibility and potential of utilizing omnidirectional mobile robots to construct a parallel mechanism with a mobile platform. To this end, a prototype of the 3-mPRS is built, and several experiments are conducted to identify the proposed kinematic parameters. The system identification of the 3-mPRS mobile parallel mechanism is conducted by analyzing the actuation inputs from the three mobile base robots. To track the motion of the robot, external devices such as the Vicon Camera are employed, and the data is fed through ROS. The collected data is processed based on the geometric properties, CAD design, and established kinematic equations in MATLAB, and the results are analyzed to evaluate the accuracy and effectiveness of the proposed calibration methods. The experiment results fall within the error range of the proposed calibration methods, indicating the successful identification of the system parameters. The differences between the measured values and the calculated values are further utilized to calibrate the 3-mPRS to better suit the experiment environment.

Keywords: mobile parallel robot, kinematic modeling, system identification

1. INTRODUCTION

Mobile parallel mechanisms have been applied to various fields in the past decades [1-3]. Mobile parallel robots showed inheritance of manipulation capabilities and wheeled mobile function of parallel robots [4-5]. The concept was first purposed in [6], further investigation was done on the dynamics and kinematics of a specific design [7]. Different designs of the mobile parallel system were developed for manufacturing [8-9] and manipulation [10]. Recently, a track-based parallel model was developed by [11] for potential large structures manufacturing. Multi-robot collaborative manipulation with wheeled robot mobile base and connected manipulators [12–15] was extended by the concept. Each mobile base is an omni directional mobile robot on ground with top structure physically connecting them to form a parallel robot topology. The hybrid and manipulation function were achieved by the combined motion of the omnidirectional mobile base.

There are separate mobile bases for existing mobile parallel robots, each mobile base is required to be self-support for the operation while maintaining stability. The manipulation of the platform needs accurate sensing among all mobile bases for updated configuration state for the mechanism. Due to free relative motion for the limbs, errors are easily brought to the system. This paper considers all of those factors to develop a new mobile parallel robot with mobile bases achieving coordinated actuation as in Fig. 1. Prismatic joints are used to connect the mobile bases for actuations of the manipulation of top plate

through differentiate motion. Based on this topology, the coordinated actuation makes the key feature of the new mobile parallel robot to have more accurate platform manipulation.

Figure 1: Mobile Parallel Robot

The structure can minimize the change in inertia with an advantage in heavy load carrying. The practical performance of parallel mechanisms is directly affected by positioning accuracy [16]. Many factors can restrain the direct use of pre-designed parameters for kinematic control models such as the error in joint clearance and manufacturing errors. Due to the natural characteristics of manufacturing error and clearance, a kinematic calibration process is needed to increase accuracy to be time and cost-effective [17]. Highlighting on the quality of manufacturing and assembly would not be cost effective and time efficient [18].

A kinematic calibration is needed to enhance accuracy of the system to be identified. The kinematic calibration procedure typically involves many measurements that are used as constraints to estimate the kinematic parameters. However, in practical experiments the collected data often contain noise that can propagate and affect the identified kinematic parameters. Similar studies have been conducted to identify the parameters. One study shows the method of selecting optimal pose for calibration [17]. In this specific study, the calibration of a robot was compared to both simulation and experimental results of random measurement poses to verify the idea. This paper was inspired by this method to compare the experimental model of the mobile parallel mechanism to the simulation model.

The limited test environment of the parallel mechanism makes the calibration process more difficult to perform. Since it is often not feasible or too complex/expensive to measure the actual kinematic values in practice with deploying of additional sensors, other methods have been carried out by research. External calibration methods are frequently used. Typically, the use of 3D camera systems is taken for such [19-20] because of high accuracy of dynamic measurements [21]. The markers and 3D cameras are taken to record the position values of the markers placed on the rigid parts of the 3-mPRS. Based on the constraints of the mechanism itself, the ball joints of the platform, the platform pose, and the angle between each slider can be identified.

This paper is a continuous work on the class of Actuation-Coordinated Mobile Parallel Robots with Hybrid Mobile and Manipulation proposed in [22]. A new development and system identification of the geometrically constrained kinematics of a 3-mPRS ACMPR with three limbs is conducted. The calculations are based on the established kinematic equations and a comparison between the theoretical and experimental results is conducted to validate the process.

2. DESIGN AND METHODOLOGY

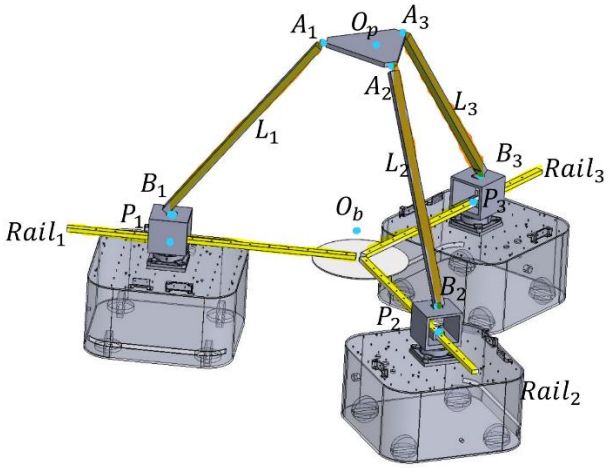


Figure 2: CAD Design of the 3-mPRS

In this section, the kinematics model, mechanical design, and experiment setups are presented. The physical structure is designed in CAD (Fig. 2) and built with 3D printed parts and other modified parts. Connecting rails and supporting limbs are modified and measured to suit the design. Markers are placed onto the robot for motion tracking. Collected data is processed and visualized with MATLAB.

2.1 The 3-mPRS Geometric Model

The mobile parallel mechanism consists of three identical limbs on a robotic structure and is notated as shown in F 2. There are three ball joints arranged on the platform and are evenly distributed on the end platform. A global coordinate system is set on the ground, a base coordinate system on the surface center of the bottom plate, and a coordinate system on the platform's surface center as in Fig. 3. Each of the ball joints would have a limb connected to it, and the other end of each limb would be connected to a revolute joint on the robot. The three prismatic joints are evenly distributed on the rails that are connected to the base plate. The rails are separated with 120º between each other. A global coordinate system $O-xyz$ is set on the theoretical flat ground plane, where z-axis is perpendicular to the ground plane. A_i represents the spherical joint center in limb i; $^{\boldsymbol{p}}\boldsymbol{a_i}$ is the position vector of $\boldsymbol{O_pA_i}$ expressed in $O_p-x_py_pz_p$; B_i represents the revolute joint center in limb i; $^{\boldsymbol{b}}\boldsymbol{b_i}$ is the position vector of $\boldsymbol{O_bB_i}$ projected onto the x_by_b-plane and expressed in the base coordinate system. The revolute joint B_i

is placed on top (z_b $direction$) of the prismatic joint. A moving base coordinate system $O_b - x_b y_b z_b$ is set on the base's geometric center. The z_b-axis is perpendicular to the base's top surface plane and the x_b-axis aligns with the vector $\boldsymbol{O_b B_1}$ (through the revolute joint center in limb 1) while the $x_b y_b$-plane is parallel to the base's top surface plane. Another moving coordinate system $O_p - x_p y_p z_p$ is placed on the platform's top surface with local origin O_p at the geometric center of the triangle. The z_p-axis is perpendicular to the platform plane and the x_p-axis passing by the spherical joint center in limb 1. The three limb lengths between points A_i and B_i are fixed constant with value l.

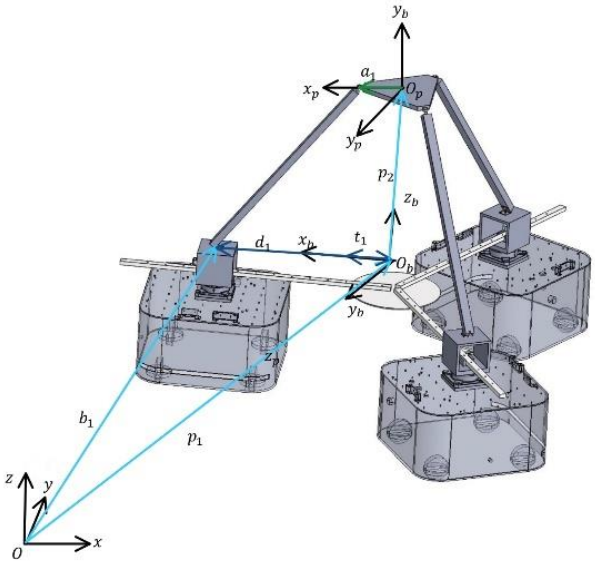


Figure 3: The Kinematics Model of the 3-mPRS

Based on the above setup as shown in Fig. 3, the basic geometric constraints of the 3-mPRS's kinematics can be given as:

$$\begin{cases} \boldsymbol{b_i} = \boldsymbol{R_z}\,{}^{\boldsymbol{b}}\boldsymbol{t_i} d_i + \boldsymbol{p_1} \\ (\boldsymbol{R_z}(\boldsymbol{R_{xy}}\,{}^{\boldsymbol{p}}\boldsymbol{a_i} + \boldsymbol{p_2}) + \boldsymbol{p_1} - \boldsymbol{b_i})^2 = l^2 \\ (\boldsymbol{R_z}(\boldsymbol{R_{xy}}\,{}^{\boldsymbol{p}}\boldsymbol{a_i} + \boldsymbol{p_2}) + \boldsymbol{p_1} - \boldsymbol{b_i})^T (\boldsymbol{R_z}\,{}^{\boldsymbol{b}}\boldsymbol{u_i}) = 0 \\ {}^{\boldsymbol{b}}\boldsymbol{t_i} = \boldsymbol{R_z}\left(\frac{(i-1)2\pi}{3}\right)[\mathbf{1} \quad \mathbf{0} \quad \mathbf{0}]^{\boldsymbol{T}} \\ {}^{\boldsymbol{b}}\boldsymbol{u_i} = \boldsymbol{R_z}\left(\frac{(i-1)2\pi}{3}\right)[\mathbf{0} \quad -\mathbf{1} \quad \mathbf{0}]^{\boldsymbol{T}} \\ i = 1,2,3 \end{cases} \quad (1)$$

Position vector $\boldsymbol{p_1} = (p_{1x}, p_{1y}, p_{1z})$ is the position vector of the base coordinate system center O_b expressed in $O - xyz$, where p_{1z} is a constant value representing the distance between the ground and O_b; $\boldsymbol{p_2} = (p_{2x}, p_{2y}, p_{2z})$ is the position vector where the platform coordinate center O_p is expressed in the base coordinate system; $\boldsymbol{b_i} = (b_{ix}, b_{iy}, b_{iz})$ is the position vector of point B_i in the global coordinate system, where $\boldsymbol{b_{iz}} = \boldsymbol{p_{1z}}$; $\boldsymbol{b_{ti}}$ is the unit vector along the rail's projection in $x_b y_b$-plane; $\boldsymbol{b_{ui}}$ is the unit vector on the axis of the revolute joint (perpendicular to the $\boldsymbol{t_i}$ vector on $x_b y_b$-plane) in limb i in the base coordinate system; d_i is the distance scaler from the revolute joint center to the local origin O_b; $\boldsymbol{R_z}$ is the z-axis rotation matrix of the base coordinate $O_b - x_b y_b z_b$ with respect to the global coordinate system; $\boldsymbol{R_{xy}}$ is the rotation matrix describing the platform coordinate system's change with respect to the base coordinate system in a two-degree-of-freedom rotation about both x_b and y_b-axis; $\boldsymbol{R_z}((i-1)\frac{2\pi}{3})$ is the z-axis rotation matrix describing a rotation angle of $(i-1)\frac{2\pi}{3}$ for limb i, constrains the 120 degrees distribution of the three sets of robots, limbs, and joints.

The first equation in Eq. 1 represents the transformation of the position vector of point B_i (revolute joint center) from the base coordinate system to the global coordinate system is achieved through a z-axis rotation with a xy-plane translation in the global coordinate system. The second equation describes the length of the limb is a constant value constrained by the spherical joint center and the revolute joint center. The third equation represents the geometric constrain of the spherical joint center's movement, which should only be in the plane perpendicular to the revolute joint's axis for each limb i. The fourth equation represents that each unit vector ${}^{\boldsymbol{b}}\boldsymbol{t_i}$ is achieved by rotating the unit vector $[\mathbf{1}\ \mathbf{0}\ \mathbf{0}]$ about z-axis for an angle of $(i-1)\frac{2\pi}{3}$. The last equation represents that each unit vector ${}^{b}u_i$ is achieved by rotating the unit vector $[\mathbf{0}\ -\mathbf{1}\ \mathbf{0}]$ about z-axis for an angle of $(i-1)\frac{2\pi}{3}$.

2.2 Inverse Kinematics

The goal of inverse kinematics is to solve for the theoretical actuation inputs, b_i, based on given platform orientation $\boldsymbol{R} = \boldsymbol{R_z R_{xy}}$ and position $\boldsymbol{p} = \boldsymbol{p_1} + \boldsymbol{R_z p_2}$ (both with respect to the global coordinate system). According to the geometric constraints given in Eq. 1, there is

$$\begin{cases} (\boldsymbol{R}^{\boldsymbol{p}}\boldsymbol{a_i} + \boldsymbol{p} - \boldsymbol{b_i})^2 = l^2 \\ (\boldsymbol{R}^{\boldsymbol{p}}\boldsymbol{a_i} + \boldsymbol{p})^T (\boldsymbol{R_z}\,{}^{\boldsymbol{b}}\boldsymbol{u_i}) = -\boldsymbol{b_i^T}(\boldsymbol{R_z}\,{}^{\boldsymbol{b}}\boldsymbol{u_i}) \\ i = 1,2,3 \end{cases} \quad (2)$$

Eq. 2 provides two constraints in each limb i for the two unknows of $\boldsymbol{b_i} = (b_{ix}, b_{iy}, b_{iz})$. The first equation is a quadratic polynomial of (b_{ix}, b_{iy}); the second equation is linear between them. The two equations should provide two solutions, and they are two intersecting points between a circle (centered at the spherical joint center A_i and has a radius of l) and the prismatic joint line $\boldsymbol{u_i}$. To satisfy continuous motion, only one of them is feasible.

2.3 Prototyping and Experimental Setup

To validate and further study this new kind of 3-mPRS system, a prototype of this structure has been built on three omnidirectional mobile robots called the Omniveyors as shown

in Fig. 4. The Omniveyors robot has four omnidirectional powered caster wheels to roll and steer to the control directions. Each of the robots would have a platform that has multiple 80mm*80mm evenly distributed M5 ports installed on the center of its top surface, allowing mounting of the slider supports and slider bars. The limbs are made of carbon fiber rods; the ball joints are made of machined metal, pinned to the 3D-printed platform, and connected to the limb using a 3D-printed cylinder and a screw. The base prismatic joint is realized using a set of aluminum rails and a slider. The slider is mounted onto slider support on the Omniveyor robot and right below the revolute joint. The revolute joint is made of a fixed caster wheel to ensure the stability of the structure. The three rails are connected to a 3D-printed plate in the center, serving as the bottom plate of the entire robot.

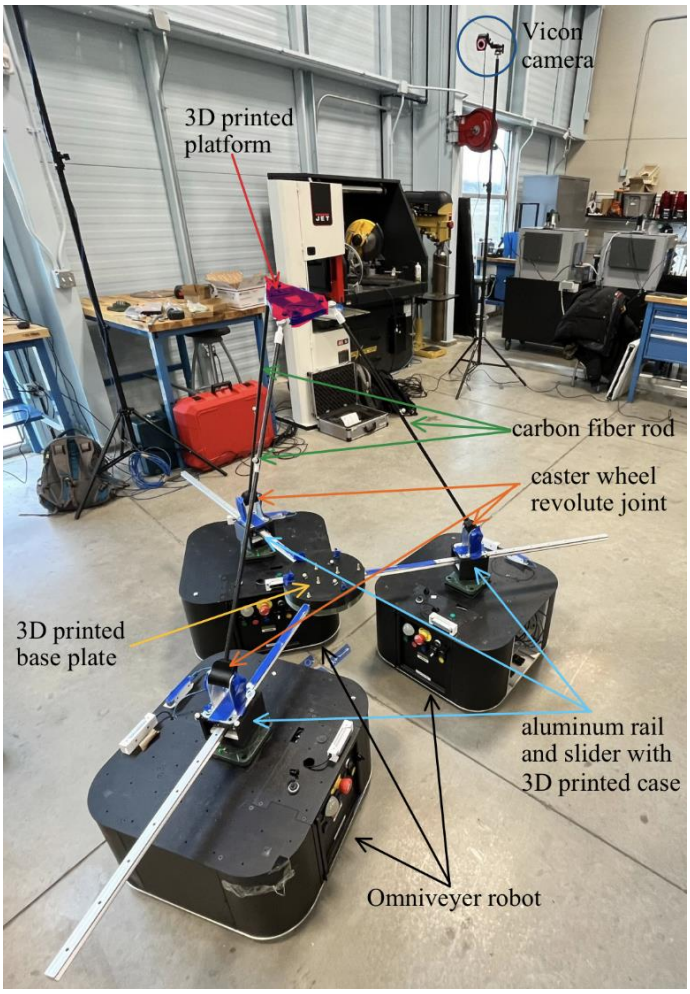


Figure 4: Experiment Setup

The robot was assumed to be tested on flat ground, but they are tested on uneven terrain. To get accurate results for the global coordinates, testing data were recorded using a Vicon camera system. Several test motions were conducted using the set up shown in Fig. 5.

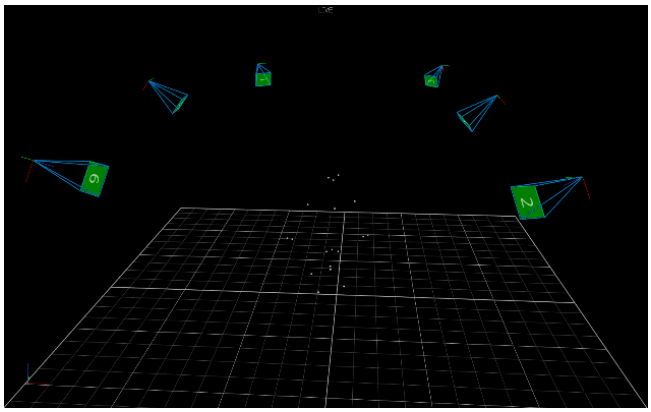


Figure 5: Vicon Camera-based Motion Record Setup

There were fifteen markers used for this experiment to record the motion of the $O_p - x_p y_p z_p$ and compare them with the theory. The markers and their intended functions are distributed as the followings: three markers would be on the top platform to construct the $O_b - x_b y_b z_b$ coordinate system; three markers placed on the limbs with one on each to observe the motion of the limb while the robot is moving as a system; six markers are located at the prismatic slider with two on each of the robot (the revolute joints is fixed on to a slider, therefore only two markers needed to construct a coordinate system for each end); 3 markers would be placed on the bottom platform to create the coordinate system. The relationship between two local coordinate systems, $O_b - x_b y_b z_b$ and $O_p - x_p y_p z_p$, and the global coordinate system $O - xyz$ are described with a rotational matrix $\boldsymbol{R_{xyz}}$ and translational vector of $\boldsymbol{p_1}$. The distance between the revolute joint and the z-axis of the bottom plate is measured with d_i $(i = 1,2,3)$.

3. KINEMATICS PARAMETER IDENTIFICATION

3.1 System Setup

For calibration, the first step is to identify the coordinates of points, at each moment there would be 15 markers with coordinates recorded in the global coordinate system. The Vicon cameras operate at 100Hz frequency, recording one dataset (15 sets of 3D coordinates, 45 numerical values) every 10ms. The datasets are organized according to their location continuity and arranged into arrays corresponding to each marker at each time stamp in MATLAB. The arranged datasets are plotted, and each marker is identified with its physical location on the robot. The dataset plot recreates the movement of each robot alongside the rotation and translation of the entire system in the global coordinate system. The angle between each slider/rail is calculated. After the angles are retrieved, the spherical centers of each ball joint mounted to the platform are identified; the revolute joint axle center is also identified. This can be realized by utilizing the markers placed on the three limbs. The Vicon cameras retrieved the arc movement of the markers in the global coordinate system.

3.2 Spherical Joint Center Identification

Due to the natural restraints of the experimental environment and manufacturing errors of the robot, the collected data showed disturbance from the theoretical assumption. To identify the spherical center, least squares fittings were taken to identify the spherical joint center and the revolute joint center. With the local frame, throughout time the marker on a limb would form a sphere around the joints. With the sphere centered at the joint and having points on the limbs moving in an arc-like manner with respect to the platform. This is achieved by the linear motion from the Omniveyor robots away and towards the center of base coordinate. local frame and revolute joint local frames, the spherical center would be the joint center as in Fig. 6. The radius is retrieved from the rewritten equation, where the ideal circle equation is used to identify the length of the limbs.

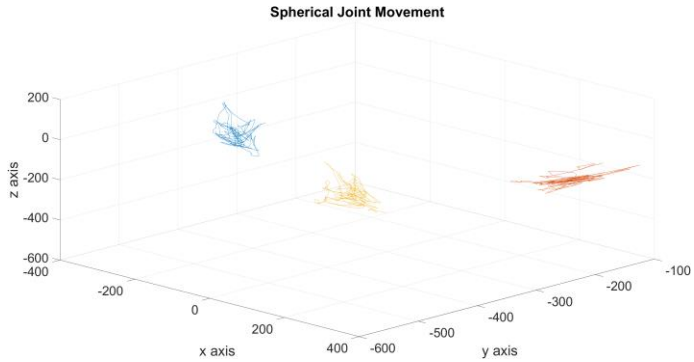


Figure 6: Sphere Surface in Op-xyz

To identify the ball joint centers A_1, A_2, and A_3 with respect to the platform, each limb is taken as a rotation around the point O_{p-xyz}, creating a sphere with A_i as its center, and the marker on the limb l_i on its surface as in Fig. 6. All the coordinates are transformed into O_{p-xyz} frame and optimized for better fitting throughout time.

$$\begin{cases} (x_i - x_c)^2 + (y_i - y_c)^2 + (z_i - z_c)^2 = r^2 \\ ax_i + by_i + cz_i + d = x_i^2 + y_i^2 + z_i^2 \\ a = 2x_c, b = 2y_c, c = 2z_c \\ d = r^2 - x_c^2 - y_c^2 - z_c^2 \end{cases} \quad (3)$$

In equation 3, (x_c, y_c, z_c) is the center of the sphere, which is the physical ball joint. (x_i, y_i, z_i) is the coordinate of the limb marker in O_b. By applying equation 3 to all the markers at each timestamp, a data cloud with respect to O_p is established. Equation 4 and 5 show the matrix operation solving for (x_c, y_c, z_c). The data cloud is fitted to a sphere with the least square fitting, which the sphere center being the center of ball joint and pole revolution with respect to O_p.

$$\boldsymbol{X} = \boldsymbol{A^{-1}B} \quad (4)$$

where $\mathbf{A} = \begin{bmatrix} x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ \dots & \dots & \dots & \dots \\ x_n & y_n & z_n & 1 \end{bmatrix}, \mathbf{X} = \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}, \mathbf{B} = \begin{bmatrix} x_1^2 + y_1^2 + z_1^2 \\ x_2^2 + y_2^2 + z_2^2 \\ \dots \\ x_n^2 + y_n^2 + z_n^2 \end{bmatrix}$

$$\begin{cases} x_c = \frac{a}{2} \\ y_c = \frac{b}{2} \\ z_c = \frac{c}{2} \\ r = \frac{\sqrt{4d + a^2 + b^2 + c^2}}{2} \end{cases} \quad (5)$$

To identify the revolute joints, the movement of each limb i with respect to each robot is as a rotation around the revolute joint B_i. The motion of the marker on limb i forms a circle throughout time with the center of the circle being the revolute joint's center in each robot's local frame O_{ri}. A similar model as shown in Eq. 3-5 is used to determine the revolute joint center with the point cluster formed as part of a circle on the plane perpendicular to $\boldsymbol{u_i}$. All the coordinates are transformed into local coordinate systems for better fitting, and transformed back to global once the sphere center is identified.

3.3 Prismatic Joint Direction Identification

The angles between each slider/rail are identified using each robot's movement along the time stamps. A base coordinate system was created using the three markers placed on the bottom plate of the robot. One of the markers is used as the origin of the local frame, and one vector connecting this marker and the other marker is set to be the x-axis of the local frame. Z-axis is the cross product of two vectors on the plane, and the y-axis is the cross-product of the x- and z-axis. With the base coordinate system embellished, the coordinates of each robot can now be converted into the base coordinate system ${}_bB_i$. Due to the rail constraint, the slider's movement can only move towards or away from the base plate. Each prismatic joint has been given two markers to identify the movement by recording the position throughout the time. The movement was identified as three clusters moving towards and away from the base coordinate as in Fig. 7.

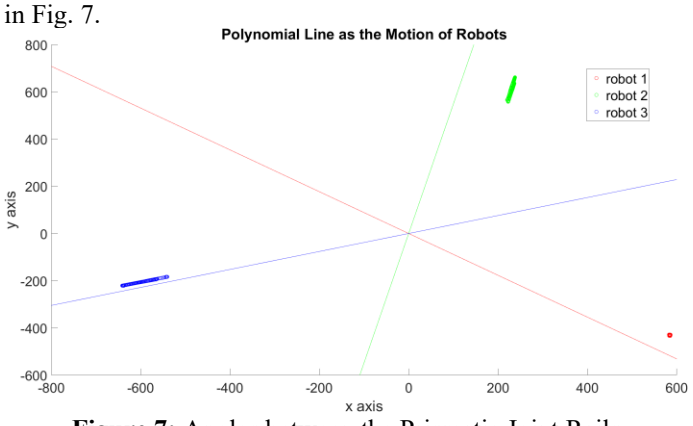


Figure 7: Angles between the Prismatic Joint Rails

Three best fitted lines were applied to the clusters, with the three lines identified, the angle between each line would be angle

between the robots. MATLAB function polyfit is called to fit the cluster of points to first degree polynomial, and the slopes are used to calculate the angles as shown in Eq. 6.

$$\begin{cases} m_i = polyfit(x_{B_i}, y_{B_i}) = (m_i, c_i) \\ line_i = m_i x + c_i \\ \theta_{i,i+1} = \cos^{-1}(\frac{line_i \cdot line_{i+1}}{|line_i \cdot line_{i+1}|}) \end{cases} \quad (6)$$

3.4 Limb Length Identification

To identify the lengths of the robotic limbs, a_i is transformed from local frame into the global frame $O - xyz$ by multiplying with the transformation matrix ${}^{O}_{o_p}\boldsymbol{T}$. The transformed vector a_i can be used to calculate the length of the limb as $l_i = (\boldsymbol{a_i} + \boldsymbol{p_1} - \boldsymbol{b_i})^2$. The position vector $\boldsymbol{p_1}$ is $O_o O_b$. Length of limb l_i can be calculated with two methods: direct vector magnitude and sum of two spheres' radius. The second method is applied here with each of the sphere's radius, which are obtained during the process of locating the center of the ball-joints and the revolute joints.

$$\begin{cases} l_i^2 = ({}^{O}_{o_p}\boldsymbol{T}^{p}\boldsymbol{a_i} + \boldsymbol{p} - \boldsymbol{b_i})^2 \\ l_i = r_{ip} + r_{ib} \end{cases} \quad (7)$$

The first equation in Eq. 7 is a variation of Eq. 1, which uses position vector to calculate the length of the limb i according to the geometric constraints, while the second equation uses the theoretical radius of the motion sphere of one certain point on limb i with respect to the platform coordinate system.

4. RESULTS AND MOTION VALIDATION

4.1 Identification Results

Based on the above identification methods, these key kinematics parameters are identified and summarized in Table 1.

	Identified value	Measured value	Error (%)
l_1 (mm)	945.56	965.23	2.034
l_2 (mm)	969.80		0.477
l_3 (mm)	964.89		0.032
$\|a_1\|$ (mm)	151.26	158.75	4.720
$\|a_2\|$ (mm)	150.25		5.356
$\|a_3\|$ (mm)	149.46		5.852
$\theta_{b1,b2}$ (degree)	121.19	120	0.99
$\theta_{b1,b3}$ (degree)	121.19		0.98
$\theta_{b2,b3}$ (degree)	117.61		1.98

Table 1: Experiment Parameters

Eq. 7 is used to calculate the length of three limbs, where the actual length of the limbs (the carbon fiber rods) is manually measured. The size of the platform and spherical joint parameter $|a_i|$ is calculated with position of three spherical joint A_i and the center of the triangle they form (theoretical O_p). The results are acceptable but with around 5% error, which is mainly due to the weak spherical joint connection to the 3D printed platform and the spherical joint clearance. The angle between each rail is calculated with Eq. 6 using data set where the prismatic joints move along the rails. The results of calculated limb length l_i are and the actual measured length of limbs have a maximum error of 2.034%. The error is likely to be introduced by two factors: (1) the tracking system error and (2) the manufacturing and assembly error of the 3D printed components. The spherical joint, being a metal reflective object, also would be recorded in the Vicon camera system occasionally as a data point to be mixed with the markers placed on top of the platform. The 3D printed part that links the limb to the spherical joint and the prismatic joint is loosely assembled on purpose to prevent any damage done to the system in case of the Omniveyor robots moving over angle allowance. These factors caused most error calculating the length of the limbs, up to 20mm. The results of the spherical joint parameters $|a_i|$ has relatively large error (around 5%) and variation. This is likely be caused by the distortion of position A_i and the triangle reconstruction process. The initial triangle formed by the calculated position of A_i is not exactly equilateral triangle, so the position vectors $\boldsymbol{a_i}$ have different magnitude/length. This could be a result of the error in limb marker position and the motion of the platform coordinate system transformation. The result error of change in angle between robots are within 2%, which is likely due to the manufacturing error of the 3D printed base and lagging in the asynchronized robot movement caused by network delays in the control cycles. The failure of robot motion synchronization caused unnecessary movement along $\boldsymbol{u_i}$ direction.

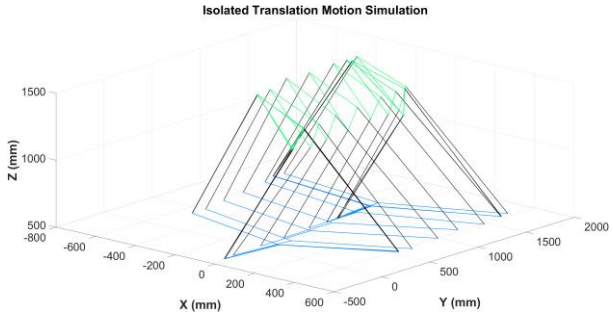
4.2 Experimental Validation

To verify the performance of the system, forward kinematics simulation of the structure is employed to analyze the theoretical motion of the mobile parallel robot during translational motion by incorporating the velocity and acceleration of each robot. The resultant behavior of the platform is then compared with the experimental data obtained through the Vicon captured markers to validate the model's accuracy. The position of the 15 markers placed on the system are recorded with timestamps by the Vicon camera set during the motion and are used to reconstruct the system. The Vicon recording operates at 100Hz. Noise and bad spots in the recorded data are identified and eliminated during reconstruction and calculation. A band-pass filter is used to eliminate the bad spots: the filter eliminated any set of data have points more 15 or less than 15. Then each set of data goes through a least-distance algorithm, where each point in one set is linked to a point in the next data set with least distance. Same

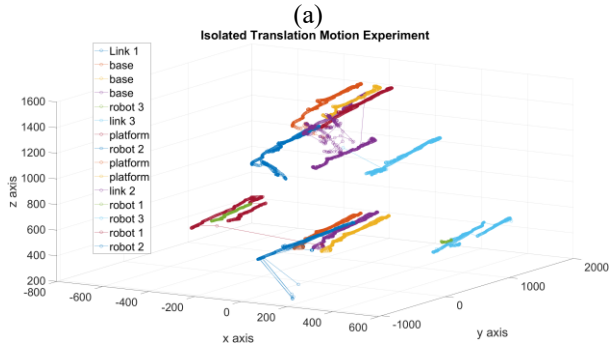
marker's moving distance between each time stamp is compared, and sets with overly large movements are eliminated. By reconstructing the data set in this order, major noise and bad spots are eliminated. However, bad points created by reflection close to designated markers are hard to be distinguished and eliminated, and the physical shaking and bumping of the system also caused much variation in the recorded data.

Two motion types were applied to validate the identified kinematics model including a pure translation motion and a hybrid motion with both translation and orientation.

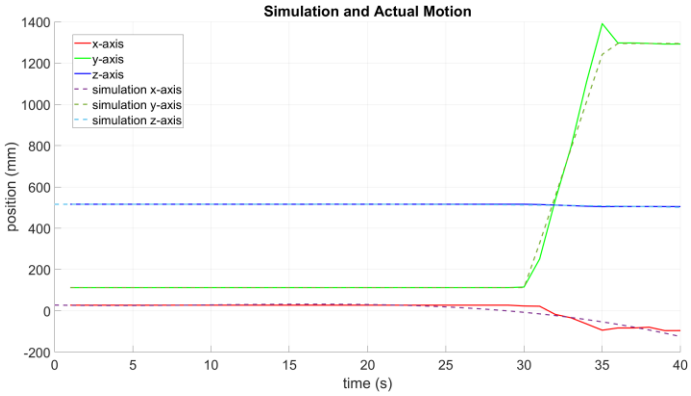
In Fig. 8, a pure translation motion is presented, where the robot is set to move along a straight line for 40 seconds, starting from O_b overlapping with O. The intended motion is set to have O_b moving along a straight line that is $60°$ around the z-axis. During this motion, the system ($O_b - x_b y_b z_b$ and $O_p - x_p y_p z_p$) has no rotation with respect to the global coordinate system. The simulated motion is given in Fig. 8(a), and the experimental Vicon marker cluster trail is plotted in Fig. 8(b). As shown in Fig. 8(c), the overall motion is broken down into 3 components as in x, y, z directions. The predicted motion and actual recorded motion in each direction are compared: the system has a period of idle time for 27 seconds with very small movement due to the Omniveyor robots adjusting their positions to have O_b overlaying with O, and then starts to move in the path specified above. There is nearly no y-axis movement, which corresponds to the Fig. 9(a). From Fig. 8(d), the variation of position in all directions are within 2%, making the simulation reliable in motion prediction.



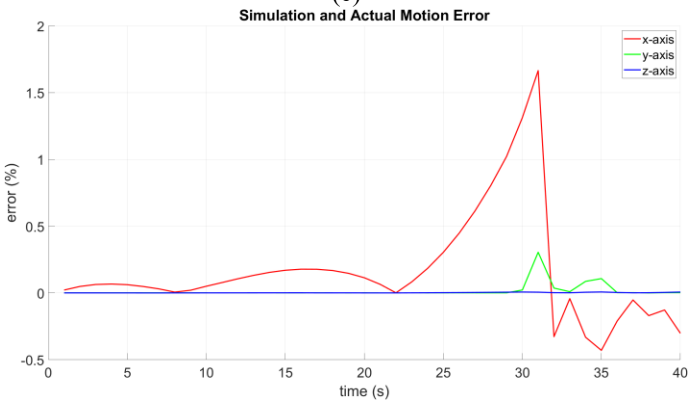
(a)



(b)



(c)



(d)

Figure 8: Validation Results of a Pure Translation motion in (a) Simulation and (b) Experiment (c) Comparison of Motion in 3-axis (d) Error

The orientation accuracy of the 3-mRPS is represented by the average variation of the platform's roll, pitch, and yaw during an isolated translation operation (there should be no variation in straight line motion, which represents the platform would have no rotation motion), as well as the angle between the rails/moving robots. These two parameters would greatly affect the performance of the 3-mRPS and cause error between the theoretical kinematic model and the actual performance. The accuracy of the system is represented by the kinematic model's ability to predict the motion of the 3-mRPS with the given initial condition and kinematic inputs. For pure translation motion shown in Fig.8, the platform orientation was recorded by the angles in Fig. 9(a) below. During the idle time, the platform should have $0°$ roll and pitch, indicating the platform experienced no rotation on the $O - xy$ plane. The $57.29°$ yaw during the idle time represents the initial position error of the system's orientation. As shown in Fig. 9(b), during the idle time the translation movement along the straight line represented by the yaw component has a stable error (2.74%) due to an initial orientation shift; while the system turbulence represented by roll and pitch component (rotation about the xy-plane) has in general 0 error with occasional sudden peaks, which could be

caused by the physical shaking and bumping of the system while it adjusting its position. However, starting from 27th second, the system begins to move, and the angle variation also increases. The system experiences a very large error in roll, up to 33.42º, which shows extreme unstableness. However, yaw error is much more stable as the average variation is within 3.43º. During experiment, the linkage between platform and the limbs would disconnect and reconnect as the limbs fell out of the 3D printed spherical joint connector. This is the major cause of the large error in the platform motion.

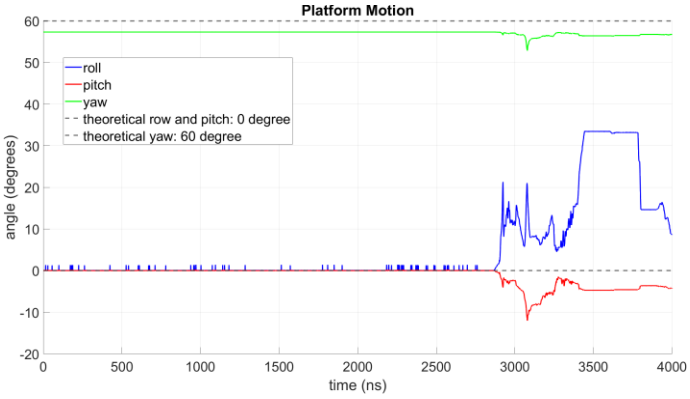
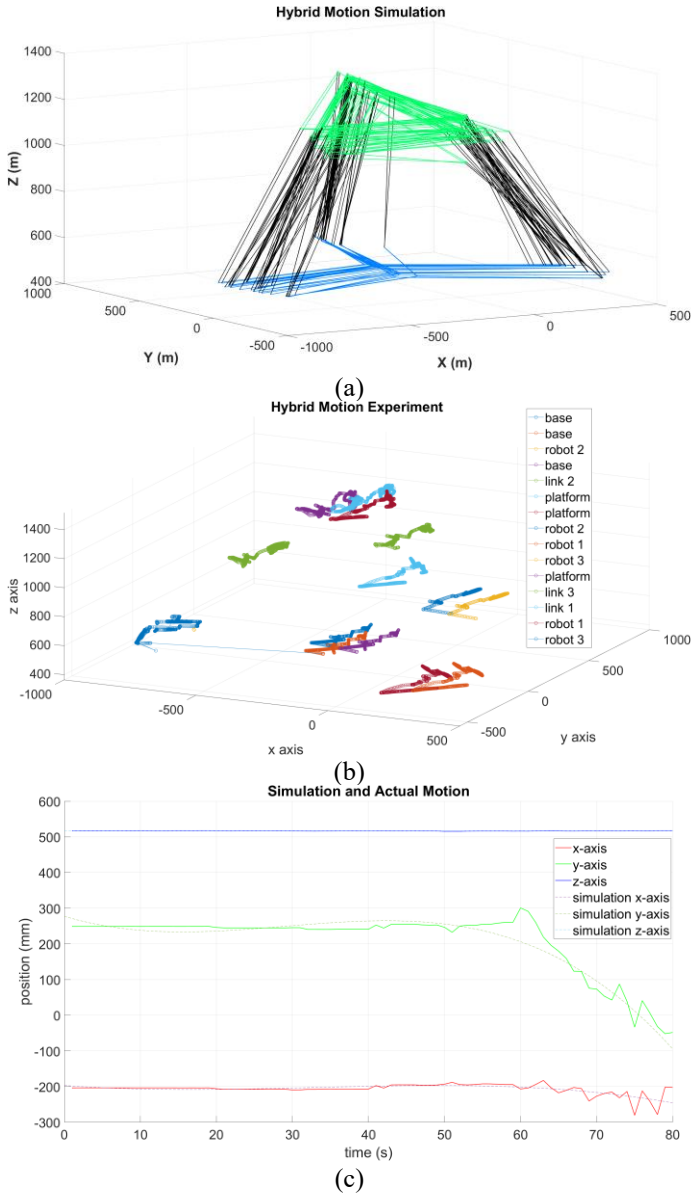
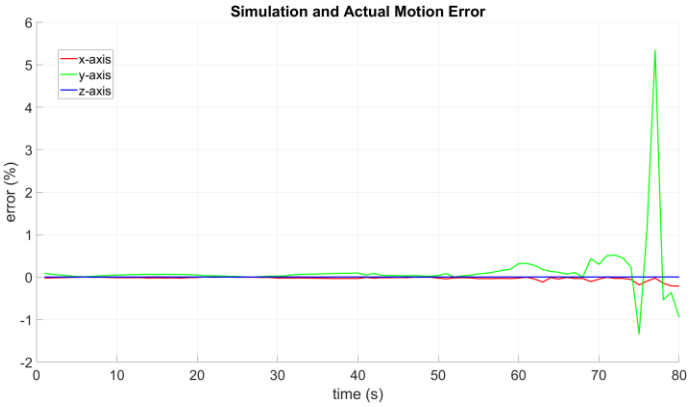


Figure 9: Roll, Pitch, Yaw (a) Variation (b) Error in Time

In Fig. 10, a hybrid motion with translation and rotation movement is presented. The whole motion is conducted in two sections: first the system remains steady (O_b overlapping with O), and each Omniveyor robot moves toward the base, then back to its original position; second, the whole system then moves along a straight line and rotate around z_b-axis at the same time. The simulated path is for the system O_b to start from O and travel along a straight line towards $[200, 300, z_{O_b}]$. After it reaches the destination, the system would go back to O in the same manner. During the translation, the system would rotate about z_b-axis clockwise and counterclockwise periodically. The movement is made in 80 seconds, with stops in between. The second part of the motion (translation with rotation) representation is given in Fig. 10(a) to study the rotation motion of the platform coordinate system, and the Vicon marker cluster trail for the whole motion is plotted in Fig. 10(b). Both x- and y-axis motion shown in Fig. 10(c) have pike-like shaking, while z-axis motion is smooth and stable. These spike-like outliers correlate with the rotation motion as shown in Fig. 10(a), which also leads to the zig-zag path shown in Fig. 10(b). In Fig. 10(d), the motion error between simulation and experiment shows that during the first section of the experiment, the system O_b has little movement (error < 0.1%), and during the second section of the experiment the error increases dramatically up to 5.33% in y-axis direction motion. This factor validates the fact that incorporating rotation movement during translation would greatly increase motion error in comparison with pure translation motion.





(d)

Figure 8: Hybrid Motion Validation with Results in (a) Simulation and (b) Experiment (c) Comparison of Motion in 3-axis (d) Error

With the average position variation of the platform being small, the control algorithm was well tailored for the given task. However, the structure design demonstrated much instability and caused much error. For the variation of errors of the studied 3-mPRS mobile parallel robot, the variation of simulation and movement error range is within 2%. The platform movement error regarding roll, pitch, yaw is however great, close to and average of 17% for roll, and less than 4% for pitch and yaw. The error for the identified length of the limb is within 2.04%, the error for the identified spherical joint center parameter error is within 5.9%, the error of identified angle between the robots are within 2%. The system's verification was realized by comparing the theoretical end effector with the prototype's real-life mode. The results fall in similar range. Both the theoretical and experimental results showed that the 3-mPRS robotic structure is controllable, the real-life model falls within the prediction of the theoretical model.

5. CONCLUSIONS

This paper developed and introduced the identification and verification of a 3-mPRS mobile parallel robot with coordinated actuation bases. The accuracy and durability of the 3-mPRS is tested and the mechanical constraints are identified. The length of the limb, angle between each slider/rail, and the trajectory of the system are identified and validated. The motion of the robots and the behavior of the platform are identified and simulated. The translation motion and rotation motion of the system are decoupled and studied separately. The kinematic model is examined and tested with simulation. A prototype of the 3-mPRS was constructed and used for a series of experiments. Both the experimental data and theoretical predictions show that the novel design of the 3-mPRS is accurate and has potential in mobile manipulation applications

ACKNOWLEDGEMENTS

This work is supported by the National Science Foundation (NSF) grant under CMMI-2131711.

REFERENCES

[1] G. V. E., “Contribution to discussion of papers on research in automobile stability, control and tyre performance,” *Proc. of Auto Div. Inst. Mech. Eng.*, vol. 171, pp. 392–395, 1957, Accessed: Oct. 12, 2022. [Online]. Available: https://cir.nii.ac.jp/crid/1570854175769550208.bib?lang=en

[2] A. Pott, “Cable-driven parallel robots: Theory and application,” in *Springer Tracts in Advanced Robotics*, vol. 120, 2018. doi: 10.1007/978-3-319-76138-1.

[3] J. S. Dai and D. R. Kerr, “A six-component contact force measurement device based on the Stewart platform,” *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering*, vol. 214, no. 5, 2000, doi: 10.1243/0954406001523696.

[4] R. M. Voyles and A. C. Larson, “TerminatorBot: A novel robot with dual-use mechanism for locomotion and manipulation,” *IEEE/ASME Transactions on Mechatronics*, vol. 10, no. 1, 2005, doi: 10.1109/TMECH.2004.842245.

[5] Y. Chen *et al.*, “A Robotic Lift Assister: A Smart Companion for Heavy Payload Transport and Manipulation in Automotive Assembly,” *IEEE Robot Autom Mag*, vol. 25, no. 2, 2018, doi: 10.1109/MRA.2018.2815704.

[6] R. M. Voyles and R. Godzdanker, “Side-slipping locomotion of a miniature, reconfigurable limb/tread hybrid robot,” in *Proceedings of the 2008 IEEE International Workshop on Safety, Security and Rescue Robotics, SSRR 2008*, 2008. doi: 10.1109/SSRR.2008.4745878.

[7] N. Pedemonte *et al.*, “FASTKIT: A Mobile Cable-Driven Parallel Robot for Logistics,” in *Springer Tracts in Advanced Robotics*, vol. 132, 2020. doi: 10.1007/978-3-030-22327-4_8.

[8] A. Olarra, J. M. Allen, and D. A. Axinte, “Experimental evaluation of a special purpose miniature machine tool with parallel kinematics architecture: Free leg hexapod,” *Precis Eng*, vol. 38, no. 3, 2014, doi: 10.1016/j.precisioneng.2014.02.009.

[9] P. Ben Horin, S. Djerassi, M. Shoham, and R. Ben Horin, “Dynamics of a six degrees-of-freedom parallel robot actuated by three two-wheel carts,” *Multibody Syst Dyn*, vol. 16, no. 2, 2006, doi: 10.1007/s11044-006-9016-4.

[10] R. Ben-Horin, M. Shoham, and S. Djerassi, “Kinematics, dynamics and construction of a planarly actuated parallel robot,” *Robot Comput Integr Manuf*, vol. 14, no. 2, 1998, doi: 10.1016/S0736-5845(97)00035-5.

[11] S. Shentu, F. Xie, X. J. Liu, and Z. Gong, “Motion Control and Trajectory Planning for Obstacle Avoidance

of the Mobile Parallel Robot Driven by Three Tracked Vehicles," *Robotica*, vol. 39, no. 6, 2021, doi: 10.1017/S0263574720000880.

[12] N. Kumar and S. Coros, "Trajectory optimization for a class of robots belonging to Constrained Collaborative Mobile Agents (CCMA) family," in *Proceedings - IEEE International Conference on Robotics and Automation*, 2020. doi: 10.1109/ICRA40945.2020.9197048.

[13] H. Yang, S. Krut, F. Pierrot, and C. Baradat, "On the design of mobile parallel robots for large workspace applications," in *Proceedings of the ASME Design Engineering Technical Conference*, 2011, vol. 6, no. PARTS A AND B. doi: 10.1115/DETC2011-48101.

[14] P. F. Muir and C. P. Neuman, "Kinematic modeling of wheeled mobile robots," *J Robot Syst*, vol. 4, no. 2, 1987, doi: 10.1002/rob.4620040209.

[15] C. P. Tang and V. N. Krovi, "Manipulability-based configuration evaluation of cooperative payload transport by mobile manipulator collectives," *Robotica*, vol. 25, no. 1, 2007, doi: 10.1017/S0263574706002979.

[16] J. -P. Merlet, *Parallel Robots*, vol. 128. Berlin/Heidelberg: Springer-Verlag, 2006. doi: 10.1007/1-4020-4133-0.

[17] J. Zhou, H. N. Nguyen, and H. J. Kang, "Selecting optimal measurement poses for kinematic calibration of industrial robots," *Advances in Mechanical Engineering*, vol. 2014, 2014, doi: 10.1155/2014/291389.

[18] X. Chai, L. Nurahmi, J. S. Dai, and D. Gan, "Kinematic calibration of a 3RRPS metamorphic parallel mechanism," in *Proceedings of the ASME Design Engineering Technical Conference*, 2020, vol. 10. doi: 10.1115/DETC2020-22169.

[19] S. Bai and M. Y. Teo, "Kinematic calibration and pose measurement of a medical parallel manipulator by optical position sensors," *J Robot Syst*, vol. 20, no. 4, 2003, doi: 10.1002/rob.10081.

[20] N. Machicoane, A. Aliseda, R. Volk, and M. Bourgoin, "A simplified and versatile calibration method for multi-camera optical systems in 3D particle imaging," *Review of Scientific Instruments*, vol. 90, no. 3, 2019, doi: 10.1063/1.5080743.

[21] D. Zhan, L. Yu, J. Xiao, and T. Chen, "Multi-camera and structured-light vision system (MSVS) for dynamic high-accuracy 3D measurements of railway tunnels," *Sensors (Switzerland)*, vol. 15, no. 4, 2015, doi: 10.3390/s150408664.

[22] D. Gan, J. Fu, M. Rastgaar, B. C. Min, and R. Voyles, "Actuation-coordinated mobile parallel robots with hybrid mobile and manipulation function," Journal of Mechanisms and Robotics, vol. 14, no. 4, 2022, . doi: 10.1115/1.4053821.