Interactive-Motion Control of Modular Reconfigurable Manipulators

Weihai Chen*, Guilin Yang*, Edwin Hui Leong Ho* and I-Ming Chen+

*Singapore Institute of Manufacturing Technology
71 Nanyang Drive, Singapore 638075
whchen, glyang, hlho@SIMTech.a-star.edu.sg

+ School of Mechanical & Production Engineering
Nanyang Technological University, Singapore 639798
michen@ntu.edu.sg

ABSTRACT — A joystick-based interactive motion control approach is proposed for modular reconfigurable manipulators. Based on the product-of-exponentials (POE) formula, the velocity models as well as the incremental displacement models have been formulated for both serial manipulators (with arbitrary configurations and DOFs) and a class of three-legged parallel manipulators. As a result, two different control modes, i.e., the velocity control mode and the incremental displacement control mode, have been developed. A user-friendly GUI has also been developed, which can display the joystick input, the actual joint angles, and the end-effector pose simultaneously. The effectiveness of this approach has been demonstrated by a 6-DOF serial modular robot and a 6-DOF 3RPRS parallel robot.

1 Introduction

A modular reconfigurable robot system consists of a collection of robot modules such as actuators, rigid links, and end-effectors [1, 2, 3]. These modular components can be rapidly assembled into various robot configurations having different working capabilities [4, 5]. However, the formalization of a generalized control scheme for such a modular manipulator is more difficult than a conventional manipulator due to its flexibility in configuration [6, 7, 8]. Hence, “teach and play back” is an effective and convenient way for the motion control of a modular manipulator. In this circumstance, a joystick can be employed as an intuitive position or velocity input device. It makes interactive communication between operator and robot possible, which is an important feature of intelligent robots [9]. Moreover, the algorithms developed for the joystick-based motion control can be readily extended to the haptic device-based tele-presence control [10, 11, 12].

A joystick-based motion control can be realized in either joint space or Cartesian space. Obviously, the joint-space motion control is easy and straightforward. It does not need kinematics models so that it is independent of manipulator configurations. However, the major drawback of the joint-space motion control lies in that the operator has no feeling about the end-effector motions in Cartesian space. Hence, accurate position controls in Cartesian space are unable to achieve.

In this paper, we will mainly focus on the joystick-based Cartesian space motion control. Because the joystick motion is quite flimsy, absolute position control will easily result in jerky motions of the manipulators. Instead, two kinds of stable control modes, i.e., velocity control mode and incremental displacement control mode have been proposed. In general, the incremental displacement control can achieve higher positioning accuracy than the velocity control. On the other hand, the velocity control can enable the manipulator to move more smoothly than the incremental displacement control.

Since a modular reconfigurable robot can assume any possible manipulator configurations, both serial and parallel manipulators need to be considered. For the serial manipulators, there are no configuration limitations. For the parallel manipulators, however, because of the complexity of their configuration types, a class of three-legged manipulators is selected as the norm of modular parallel manipulators [13]. In order to realize the Cartesian-space motion control, it is essential to formulate the kinematic models for both serial manipulators and the three-legged parallel manipulators. Based on the POE formula, both the instantaneous kinematics models and the incremental displacement models have been formulated. With the proposed kinematic algorithms, a user-friendly GUI has been developed using VC++. By retrieving the data from both joysticks output ports and the manipulator’s joint modules, the GUI can display the control mode, the joystick input, the actual joint angles, and the actual end-effector pose simultaneously. In brief, such a joystick-based motion control method provides the end-user an easy-to-access, interactive and efficient control interface for modular reconfigurable robots.

2 Modular Reconfigurable Robot System

A modular reconfigurable robot system consists of a series of standard active-joint modules, passive-joint modules, and customized rigid links and end-effectors that can be assembled into a variety of robot configurations. The active-joint module is a self-contained compact mechatronic drive unit with the built-in motor, encoder, controller, amplifier, harmonic drive, and communication interface. The active-joints modules are 1-DOF revolute joint modules, 1-DOF prismatic joint modules, and 2-DOF wrist joint modules. The passive joint modules are basically rotary joint modules, universal joint modules and spherical...
joint modules. Some of the active- and passive-joint modules are shown in Figure 1 (a) and (b), respectively.

(a) Active-joint modules (b) Passive-joint modules

Fig. 1 Robot modules

With an inventory of such robot modules, various robot modular manipulators and a 6-DOF 3RPRS parallel modular manipulator are illustrated in Fig. 2 (a) and (b), respectively. Some of the active- and passive-joint modules can be rapidly constructed. A 6-DOF serial modular manipulator with respect to frame 0 (base), respectively. Hence, we have

\[ T_{0,0}^{-1} = \begin{bmatrix} R_{0,0} & P_{0,0} \\ 0 & 1 \end{bmatrix}, \]

where \( R_{0,0} \in SO(3) \) and \( P_{0,0} \in R^{3 \times 1} \) are the orientation matrix and the position vector of frame 0 (end-effector) with respect to frame 0 (base), respectively. Hence, we have

\[ \hat{T}_{0,0} T_{0,0}^{-1} = \begin{bmatrix} R_{0,0} R_{0,0}^T - R_{0,0} R_{0,0} P_{0,0} + P_{0,0} \\ 0 & 0 \end{bmatrix}, \]

where \( \hat{T}_{0,0} T_{0,0}^{-1} \in SE(3) \) is termed the spatial velocity of the end-effector frame \( n \) with respect to the base frame 0 [14], denoted by \( \dot{V}_{0,n} \), which can be given by

\[ \dot{V}_{0,n} = \begin{bmatrix} \dot{\omega}_{0,n} & \dot{\upsilon}_{0,n} \\ 0 & 0 \end{bmatrix}, \]

the twist coordinates of the spatial velocity \( \dot{V}_{0,n} \), i.e., \( V_{0,n} \in R^{6 \times 1} \) can be written as:

\[ V_{0,n} = J_{0,n}^T \dot{q}, \]

where the angular component, \( \omega_{0,n} \), is the instantaneous angular velocity of the body as viewed in the spatial (base) frame; the linear component, \( \upsilon_{0,n} \), is the velocity of a (possibly imaginary) point on the same rigid body which is traveling through the origin of the spatial frame [14]. Based on Eq.(3), Eq.(7) can also be written as

\[ V_{0,n} = J_{0,n}^T \dot{q} \]

where

\[ J_{0,n}^T(q) = \begin{bmatrix} \omega_{0,n} \times p_{0,n} + \dot{p}_{0,n} \\ (R_{0,n} \dot{R}_{0,n}) \upsilon \end{bmatrix} \]

The matrix \( J_{0,n}^T(q) \in R^{6 \times n} \) is termed as the spatial manipulator Jacobian Matrix [14]. It’s \( i^{th} \) column can be written as

\[ \dot{q}_{0,n}(q) = \begin{bmatrix} \dot{\omega}_{0,n} \\ \dot{\upsilon}_{0,n} \end{bmatrix}, \]

Converting it into twist coordinates, we have

\[ \dot{\omega}_{0,n} = \dot{\upsilon}_{0,n} = T_{0,n}^{-1} \dot{q}_{0,n} T_{0,n}^{-1} = T_{0,n}^{-1} \dot{q}_{0,n} T_{0,n}^{-1}, \]
\[
\left(\frac{\partial^{T}R_{i}}{\partial q_0}T_{j}^{-1}\right)^{\top} = Ad_{T_{0,i}}, s_{i},
\]

where \(Ad_{T_{0,i}}, s_{i}\) is termed as the Adjoint Representation of \(T_{0,i}\) [14].

As described earlier, the spatial velocity of the end-effector is different from the conventional expression of the end-effector velocity, which makes it difficult to be implemented to intuitive motion control. The natural and conventional way to represent the end-effector velocity has the form of

\[
V_{0,n} = [\dot{p}_{0,n}, \omega_{0,n}],
\]

where \(\dot{p}_{0,n}\) represents the linear velocity and \(\omega_{0,n}\) represents the angular velocity. Both are viewed in the base frame. Such a velocity is also termed as the hybrid velocity of a rigid body [14]. Thus, according to Eqs. (7) and (12), Eq.(8) can be rewritten as

\[
V_{0,n}^h = \dot{\alpha}_{0,n}^{h} \hat{q},
\]

where \(\dot{\alpha}_{0,n}^{h} = \begin{bmatrix} I_{3} & 0_{3} \end{bmatrix}^{\top} J_{0,n}^{h} \in R^{6\times 1}\) represents the hybrid Jacobian matrix of the manipulator. Hence, the joint velocity can be given by

\[
\dot{q} = (J_{0,n}^{h})^{\top} V_{0,n}^h,
\]

where \((J_{0,n}^{h})^{\top}\) is the pseudo-inverse of \(J_{0,n}^{h}\).

### 3.2 Incremental displacement analysis

Although the velocity control mode can generate smooth motions, its control accuracy is not high enough. When a robot moves close to its target position, the incremental displacement control mode need to be employed so as to satisfy the accuracy requirement. Define the (hybrid) incremental displacement of a rigid body as

\[
\delta D_{0,n}^a = \begin{bmatrix} \delta \dot{p}_{0,n}^a \ \delta \omega_{0,n}^a \end{bmatrix},
\]

where the vector \(\delta \dot{p}_{0,n}^a = [\delta \dot{p}_{0,a}^a, \delta \dot{p}_{0,a}^b, \delta \dot{p}_{0,a}^c]^{\top} \in R^{3\times 1}\) and \(\delta \omega_{0,n}^a = [\delta \omega_{0,a}^a, \delta \omega_{0,a}^b, \delta \omega_{0,a}^c]^{\top} \in R^{3\times 1}\) represent the incremental position and orientation vectors of the end-effector with respect to the base, respectively.

Based on the definition of hybrid velocity as well as Eq.(15), the relationship between the incremental joint angles and the hybrid incremental displacement can be given by

\[
dq = (J_{0,n}^{h})^{\top} \delta D_{0,n}^a,
\]

where \(dq = [dq_1, dq_2, \ldots, dq_6]^{\top} \in R^{6\times 1}\) represents the joint angle incremental vector.

### 4 Kinematics for Parallel Manipulators

A class of three-legged 6-DOF modular parallel manipulator is the subject of this Section. Considering the symmetrical design, each leg is identical to the others. In addition, each leg is a 6-DOF serial kinematic chain with two actuators, as shown in Fig. 4. To further simplify the kinematic structure, a 3-DOF passive spherical joint is placed at each of the leg-ends.

#### 4.1 Forward displacement analysis

For each of the legs, assume that the first two joints are active joints and the third one is a passive revolute joint module in which an incremental encoder is installed to measure its joint displacement. The purpose of the forward displacement analysis is to determine the end-effector poses with given active-joint angles.

Define frame \(A\) as the local frame attached to the mobile platform, frame \(B\) as the base frame, and point \(A_i (i = 1, 2, 3)\) as the center of the spherical joint coupling the leg with the mobile platform. As shown in Fig. 4, if the coordinates of the point \(A_i\) with respect to frame \(A\) and frame \(B_{i,3}\) are given by \(p_{i}^a = (x_{ai}^a, y_{ai}^a, z_{ai}^a)^{\top}\) and \(p_{i}^b = (x_{ai}^b, y_{ai}^b, z_{ai}^b)^{\top}\), respectively, the forward kinematics of the three-legged parallel robots can be given by [15]

\[
T_{B,A} = \begin{bmatrix} p_1 & p_2 & p_3 & p_1 \times p_3 & p_3 \times p_2 & p_2 \times p_3 \end{bmatrix}^{-1},
\]

where the positional vector of the point \(A_i\) is written as

\[
\delta \overline{p}_i = T_{B_{i,3}, B_{i,1}}(0)e^{\hat{s}_{ij} \theta_{ij}} T_{B_{i,1}, B_{i,2}}(0)e^{\hat{s}_{ij} \theta_{ij}} T_{B_{2,1}, B_{1,3}}(0) e^{\hat{s}_{ij} \theta_{ij}} p_i^{B_{i,3}}.
\]

In Eq.(18), \(\hat{s}_{ij} \in \mathfrak{se}(3) (i = 1, 2, 3; j = 1, 2)\) is the twist of joint axis \(ij\) expressed in frame \(ij\); \(p_{i2} = p_{i1} - p_{i1}^a\); \(p_{i3} = p_{i2} - p_{i1}^a\) and \(p_{i1} = p_{i2} - p_{i3}\), \(p_{i2} = p_{i3}\), \(p_{i3} = p_{i2}\).

#### 4.2 Velocity analysis

The objective of velocity analysis is to determine the relationship between the six active-joint rates and the velocity of mobile platform. The twist annihilator method
proposed in [16] is employed here for the formulation of the velocity model.

Differentiating both side of Eq.(18) with respect to time, we can obtain

\[
\begin{bmatrix}
\dot{p}_1 \\
0
\end{bmatrix} = T_{B_BA} \dot{r}_{iA} + T_{B_BA} \times \dot{q}_{iA} = T_{B_BA} \times \dot{q}_{iA} + T_{B_BA} \dot{r}_{iA}
\]

\[
+ T_{B_BA} \dot{r}_{iA} = i, j = 1, 2, 3,
\]

where \( \dot{q}_i = (i, j; i = 1, 2, 3) \) is the rate of joint \( i \), \( T_{B_BA} \) is the forward kinematic transformation of frame \( i \) with respect to frame \( B \), which is given by

\[
T_{B_BA} = T_{B_Bi} T_{B_A} T_{B_A}
\]

(20)

\[
T_{B_BA} = T_{B_Bi} T_{B_A} T_{B_A}
\]

(21)

Let \( R_{B_Bi} \in SO(3) \) be the orientation matrix of \( B_Bi \), \( s_{ij} \) is the twist coordinates of joint \( i \) expressed in frame \( ij \), and \( T_{B_Bi} = \hat{b}_{ij} \) is the skew symmetric (cross-product) matrix related to vector \( p_{ij} \). Hence, Eq.(19) can be simplified as [15]

\[
\dot{p}_{ij} = \dot{u}_{ij} \dot{q}_{ij} + \dot{u}_{ij} \dot{q}_{ij} + \dot{u}_{ij} \dot{q}_{ij} + \dot{u}_{ij} \dot{q}_{ij}
\]

(22)

where \( u_{ij} = R_{B_Bi} T_{B_B} s_{ij} \). Since we are interested in the relationship between the active-joint rates and the mobile-platform velocity, the passive-joint rates should be eliminated from Eq.(22). To this end, we dot-multiply both sides of Eq.(22) by twist annihilator \( u_{ij} \times u_{ij} \) to get joint rate \( \dot{q}_{ij} \). We have

\[
\dot{q}_{ij} = (u_{ij} \times u_{ij})^T \dot{u}_{ij} = (u_{ij} \times u_{ij})^T \dot{p}_{ij}
\]

(23)

Let \( \dot{V}_{B_A} \) represents the spatial velocity of the mobile platform, which can be computed as

\[
\begin{bmatrix}
\dot{p}_1 \\
0
\end{bmatrix} = \dot{V}_{B_A} \begin{bmatrix}
\dot{p}_1 \\
0
\end{bmatrix} = \begin{bmatrix}
\dot{p}_1 \\
0
\end{bmatrix} = \begin{bmatrix}
\dot{p}_1 \\
0
\end{bmatrix} = \begin{bmatrix}
\dot{p}_1 \\
0
\end{bmatrix} = \begin{bmatrix}
\dot{p}_1 \\
0
\end{bmatrix} = \begin{bmatrix}
\dot{p}_1 \\
0
\end{bmatrix} = \begin{bmatrix}
\dot{p}_1 \\
0
\end{bmatrix} = \begin{bmatrix}
\dot{p}_1 \\
0
\end{bmatrix}
\]

(24)

where \( T_{P_A} = [I_3 - \dot{p}_{ij}] \). Let \( V_{B_A} = [\dot{V}_{B_A}, \omega_{B_A}] \). Then we have

\[
\dot{p}_{ij} = T_{P_A} V_{B_A}
\]

(25)

Substituting Eq.(25) into Eq.(23), we have

\[
\dot{q}_{ij} = (u_{ij} \times u_{ij})^T \dot{u}_{ij} = (u_{ij} \times u_{ij})^T \dot{p}_{ij} V_{B_A}
\]

(26)

Following a similar procedure, we have

\[
\dot{q}_{ij} = (u_{ij} \times u_{ij})^T \dot{u}_{ij} = (u_{ij} \times u_{ij})^T \dot{p}_{ij} V_{B_A}
\]

Hence, the instantaneous kinematic relationship between the mobile platform twist coordinate, \( V_{B_A} \), and the active-joint rate, \( \dot{q}_a \), can be written as

\[
J_a \dot{q}_a = J_f \dot{V}_{B_A},
\]

(28)

where

\[
J_a = \text{diag} \{ \dot{q}_{a1}, \dot{q}_{a2}, \dot{q}_{a3}, \dot{q}_{a4}, \dot{q}_{a5}, \dot{q}_{a6} \}
\]

(29)

\[
\dot{q}_a = [\dot{q}_{a1}, \dot{q}_{a2}, \dot{q}_{a3}, \dot{q}_{a4}, \dot{q}_{a5}, \dot{q}_{a6}]
\]

(30)

\[
J_f = \begin{bmatrix}
(u_{12} \times u_{13})^T T_{P_1} \\
(u_{13} \times u_{11})^T T_{P_1} \\
(u_{22} \times u_{23})^T T_{P_2} \\
(u_{23} \times u_{21})^T T_{P_2} \\
(u_{32} \times u_{33})^T T_{P_2} \\
(u_{33} \times u_{31})^T T_{P_2}
\end{bmatrix},
\]

(31)

with the notations below:

\[
\dot{q}_{a1} = (u_{12} \times u_{13})^T \dot{u}_{11}, \quad \dot{q}_{a2} = (u_{13} \times u_{11})^T \dot{u}_{12}
\]

\[
\dot{q}_{a3} = (u_{22} \times u_{23})^T \dot{u}_{21}, \quad \dot{q}_{a4} = (u_{23} \times u_{21})^T \dot{u}_{22}
\]

\[
\dot{q}_{a5} = (u_{32} \times u_{33})^T \dot{u}_{31}, \quad \dot{q}_{a6} = (u_{33} \times u_{31})^T \dot{u}_{32}
\]

Introducing hybrid velocity concept to Eq.(28), we have

\[
J_a \dot{q}_a = J_f \dot{V}_{B_A},
\]

(32)

where \( \dot{p}_{B_A} \in R_{31} \) is the position vector of the forward kinematics \( T_{B_A} \), and \( \dot{V}_{B_A} = [\dot{p}_{B_A}, \dot{r}_{B_A}] \) is the hybrid velocity of the mobile-platform frame \( A \). Define manipulator forward hybrid Jacobian matrix as \( J_h \), we have

\[
J_h \dot{q}_a = J_h \dot{V}_{B_A},
\]

(33)

where

\[
J_h = \begin{bmatrix}
I_{33} \\
0
\end{bmatrix}
\]

(34)

If the inverse kinematic matrix, \( J_a \), is nonsingular, then the active-joint rate vector, \( \dot{q}_a \), is given by

\[
\dot{q}_a = J_a^{-1} J_h \dot{V}_{B_A}
\]

(35)

### 4.3 Incremental displacement analysis

Similar to the serial manipulators, the relationship between the incremental joint angles and the hybrid incremental displacement can be given by

\[
dq_a = J_a^{-1} J_h d\theta_{B_A}
\]

(36)

where \( dq_a = [dq_{a1}, dq_{a2}, \ldots, dq_{a6}] \in R_{61} \) represents the active-joint angle incremental vector, and \( d\theta_{B_A} \in R_{61} \) represents the incremental position and orientation vector of the mobile-platform frame \( A \) with respect to the frame \( B \).
throttle. The stick handle has three DOFs such as the left-right motion (LRmove), forward-backward motion (FBmove), and twist motion (Twist). For Cartesian space control, LRmove, FBmove, and Twist are used to control the manipulate motion (translation and rotation) about X, Y, and Z-axis, respectively. For joint space control, only the forward-backward motion of the stick handle will be used to control the selected joint. The throttle is used to perform fine step control because it can capture small input.

A joystick control interface is illustrated in Fig. 6. This interface contains four parts. Part one displays the actual end-effector pose and joint angles. Part two has two columns. The left column is designed for the joint space control, which displays joint rates (or joint angle increments). The right column is used for Cartesian space control, which displays the end-effector’s velocity (or displacement increments). Part three displays joystick value. Part four is used to choose the control types and modes.

There are two kinds of control types, namely joint space or Cartesian space. The control type can be switched by pressing joystick button 1 and 2 (in either ascending or descending orders). The control mode (velocity control or incremental displacement control) can be switched by pressing joystick button 7 (Fig. 5), and the message shown the control mode is displayed in Status box that is located in the left area of part four. The Status box also displays other messages such as Initializing, Reset, Halt etc.

Joystick messages from USB interface of PC is periodically inquired through setting a timer. The enquiry can be responded by VC++ system structure function DIJOYSTATE defined in the head file dinput.h.

6 Experimental Studies

6.1 A 6-DOF serial manipulator

A 6-DOF serial manipulator as shown in Fig. 2 (a) is used to verify the velocity control mode. Choose appropriate mapping factors to let the end-effector of the manipulator have maximum velocity 160deg/sec. Using the forward kinematics algorithm, the initial pose of the end-effector is

\[
T[0] = \begin{bmatrix}
0 & 1 & 0 & 0 \\
1 & 0 & 0 & -325 \\
0 & 0 & -1 & 225 \\
0 & 0 & 0 & 1
\end{bmatrix}.
\]

Now let us control the end-effector to move in -X direction. First, select the control type in the Axis Chosen box to be X. Then, move the stick handle of the joystick towards right direction with velocity about 50mm/sec till the position output value of X axis shown in the interface to be -160mm. As expected, the end-effector of the manipulator moved along the negative X-axis direction to the desired point (Figure 7)

6.2 Parallel robot experiment

A 6-DOF 3RPRS parallel manipulator as shown in Fig. 2 (b) is used to verify both the velocity and incremental displacement control modes. The manipulator has initial pose as \( R = I \) and \( P = [0, 0, 609]^T \)

- Translation along Z-axis
  - Select the control type in the Axis Chosen box to be Z. Then, rotate the stick handle of the joystick in clockwise direction with a velocity about 50mm/sec so as to move the
end-effector along the negative direction of Z-axis. When the end-effector is closed to the target location \((Z=370\text{mm})\), the control mode is switched to incremental displacement control mode with 0.1\(\text{mm}\) resolution. This straight-line control is illustrated in Fig. 8.

![Fig. 8 Robot at working location](image)

- Rotation about X-axis

Select the control type in the Axis Chosen box to be Rx. Then, move the stick handle of the joystick towards left side with a velocity about 0.1 rad/sec to rotate the end-effector around the positive direction of X-axis. When the end-effector is closed to the target orientation \((\theta_i = 0.436\text{rad}\) or 25\text{degree})\), the control mode is switched to incremental displacement control with angular increment 0.01\text{rad}. This rotation control is illustrated in Fig. 9.

![Fig. 9 Rotate about X-axis positive direction](image)

### 7 Summary

In this paper, a joystick-based interactive motion control interface has been developed for modular reconfigurable manipulators. The kinematics models and algorithms are formulated based on POE formula, which are suitable for both serial and parallel manipulators. Two different control modes, namely velocity control mode and incremental displacement control mode, have been proposed. The velocity control mode can achieve faster and smoother manipulator motion so that it can be employed for coarse motion control. The incremental displacement control, on the other hand, is mainly used for high-resolution step motion control. By combining these two different control modes sequentially, a fast, smooth, and precision motion control can be achieved. The effectiveness of this intuitive and interactive control approach has been experimentally demonstrated.

### Acknowledgment

This work is supported by Singapore Institute of Manufacturing Technology under the applied research project grant C01-A-137-AR and the university collaboration project grant U97-A-006.

### References


