Improved Tracking Performance Using Dual and Double Dual Feedback Loops

Murali K Karnam, Rohit Kalla, Anirban Nag, Saurav Agarwal and Sandipan Bandyopadhyay
Department of Engineering Design
Indian Institute of Technology Madras

***Abstract*— Control of robots and mechanisms is a challenging task, more so with inaccurate models and error-prone mechanical elements. This work tries to compensate the errors, owing to different factors, which creep into the system and hamper its performance, by appropriately employing multiple control loops, utilising redundant feedback present in the manipulator design, in real time. It looks into the different control strategies and validate them with experimental analysis by tracking a trajectory.**

Robot control, redundant sensing, control loops.

I. Introduction

In an experiment involving mechanical hardware and electronic control, where the objective is to achieve accurate tracking of a target trajectory, the errors incurred can be attributed to two major factors:

1) The mechanical errors, caused due to imperfect modeling of geometric parameters, inaccuracies in fabrication and assembly, inherent limitations such as finite stiffness, unmodelled dynamics such as friction, backlash etc.
2) The electrical/electronic errors, encompassing issues such as finite resolution of the sensors, sensor delay and noise, non-linearities and inaccuracies in the actuator response, drift in IMUs, communication delays etc.

The control performance that is actually achieved in an experiment is subservient to the above, i.e., these errors set the lower bounds on the error that the system would show with a "perfect" tracking controller. In other words, these errors remain in the system, irrespective of the sophistication or the efficacy of the control algorithm being used on top of them.

It is, therefore, imperative, that one pays attention to these systemic errors and their mitigation, in order to achieve good tracking performance. There are many established techniques for this purpose, such as the ones described below.

1) Anti-backlash drives, where the error due to the backlash present in geared motors is reduced or eliminated by incorporating design modifications [1], [2], or by compensating it with control [3].
2) Calibration systems, where the inaccuracies present in the model of a manipulator are identified quantitatively, and compensated for through proper calibration of the robot [4], [5].
3) Model-based control, where the model used for control can be improved by incorporating issues such as friction, backlash etc. in it [6].

Each of the techniques described above have their respective advantages and disadvantages. For instance, the calibration is a very common technique to mitigate the mechanical inaccuracies, but it requires very expensive and sophisticated instrumentation, and has to be repeated at regular intervals to be effective. Beyond these, one possible solution to this problem, is not to try to assess the errors at the component levels (as in parameter identification), or to do it at a system level either (like in calibration systems), but to keep track of more than one indicator of the overall tracking performance in the real-time. Each of these indicators can provide a feedback loop targeted at diminishing the error in the respective loop. In other words, while a main control loop is trying to keep the manipulator on track at the systemic level, one or more subsidiary loops are tracking errors at intermediate levels, and keeping them in check, thus contributing to the overall objective, but without even being "aware" of the same. This can only be achieved at the cost of extra sensors. However, as shown in this paper, it is often possible to come up with a redundant suite of inexpensive sensors to achieve this, albeit with the aid of some ingenuity in the mechanical design of the components of the manipulator.

This paper considers a parallel manipulator, MaPaMan-I [7], and utilises the additional sensors in it to obtain different possible feedback. These are analysed and multiple control loops are used to reduce the errors in real time for trajectory tracking. The description of the manipulator with its design features, which allow the scope for redundant feedback, is discussed in Section II. Three control strategies are then formulated based on the available feedback options which form different control loops as explained in Section III. The formulated control strategies are validated by performing trajectory tracking experiments on a physical prototype and analysing the results, which are presented in Section IV.

II. Redundant Sensing in MaPaMan-I

A. MaPaMan-I Architecture

MaPaMan-I is a three-degree-of-freedom manipulator with roll, pitch, and heave as its degrees-of-freedom [7]. It has

three legs, which are individually actuated by a motor each. Each leg consists of a parallelogram mechanism, with a strut connecting the coupler link to the end-effector plate with rotary and spherical joints at the corresponding links, which can be seen in Fig. 1.

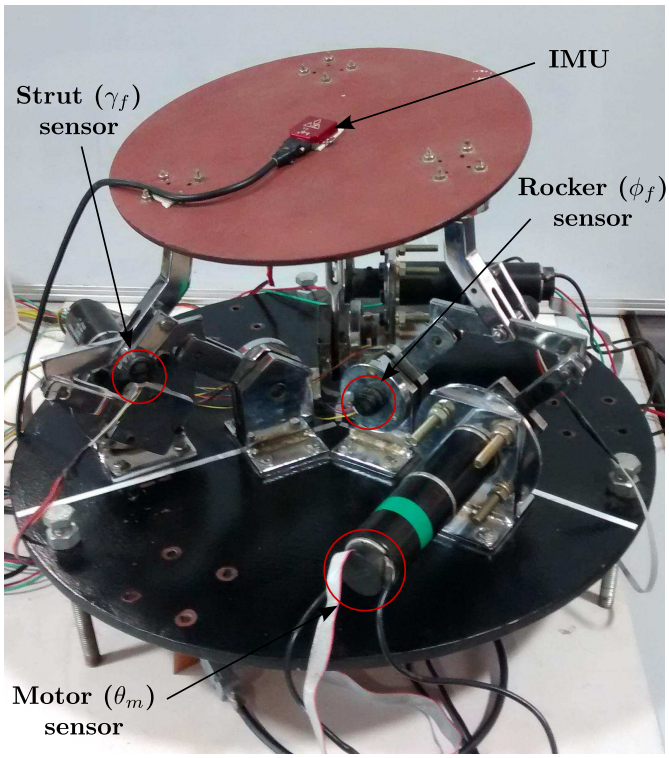

Fig. 1. Architecture of MaPaMan-I and its design features

B. Key Design Features of MaPaMan-I

The physical prototype of MaPaMan-I used in this work is described in [8], and has a suite of senors incorporated, which can be utilised in obtaining redundant feedback for use in the control system design. The performance of the complete system can be improved by using the feedback loops at multiple stages in error control. The possible feedback options using the sensor system present can be described as given below.

1) Motor encoders ($\boldsymbol{\theta}_m$): These give the positions of the motors, which can be compared with the reference inputs for them. These can also be used to calculate the position of the end-effector by solving forward kinematics of the mechanism and to compare with the desired position in the task space. They do not see the effect of the inaccuracies present in the gear box or the mechanism.
2) Rocker encoders ($\boldsymbol{\phi}_f$): Each leg of MaPaMan-I being a parallelogram mechanism, the angle measured at the rocker is equal to the *actual* angle made by the actuated link. Ideally these sensors should be able to capture the backlash present in the motor. Hence, it can also be compared with the commanded joint positions or the pose and position of the top platform of the manipulator in task space using the forward kinematics. In the absence of backlash and other mechanical inaccuracies, $\boldsymbol{\theta}_m = \boldsymbol{\phi}_f$.
3) Strut encoders ($\boldsymbol{\gamma}_f$): The strut angle measured with this sensor, along with either the motor or rocker measurement can be used to find the end-effector position. The computed task space position incorporate the mechanical inaccuracies in the parallelogram mechanism completely, along with the backlash present in the motor. Although, this would still not be able to assess the inaccuracies in the strut itself and beyond it.
4) End-effector IMU: This gives a feedback of the pose of the manipulator at the task space directly and encompasses all the possible inaccuracies present in the system.

Using the above sensor suite, different control strategies can be designed. Three such designs are implemented on MaPaMan-I, as described in Section III. These three are chosen based on the feedback available and the associated advantages. In the hardware setup used for this paper, the encoders present in rockers and struts have an angular resolution of $0.25°$, and the encoders present in the motors have an angular resolution of $0.35°$. The IMU used has a resolution of $0.05°$ and an accuracy of $1.0°$. Since the performance of the control system would be significantly dependent on the accuracy of the feedback, the IMU sensor is not used in the control scheme. Instead, it is used to compare the performance of the different control strategies employed.

III. Control Strategies

Multiple control loops to improve the performance of a system have been previously developed [9] and implemented for parallel mechanisms [3], [8]. These have looked into the implementation of the error control at the joint space level alone. Here, the control scheme is implemented at the task space with an additional control loop appended to the dual-loop control scheme to improve the performance. Totally three strategies are described as below with each one adding a control loop to the existing hierarchy.

A. Single-loop Control

In this method, the desired trajectory, $\boldsymbol{X}_d(t)$, of the manipulator is converted to the desired position of the motor, $\boldsymbol{\theta}_d(t)$, using the inverse kinematics equations, which is directly commanded to the motor controller. The control scheme is shown in the Fig. 2. The inner loop employs a proportional-derivative (PD) control, which works with the feedback from the encoder present in the motor. The desired position is evaluated as:

$$\boldsymbol{V}(t) = \boldsymbol{K}_p \boldsymbol{e}_m(t) + \boldsymbol{K}_d \dot{\boldsymbol{e}}_m(t), \quad \text{where} \qquad (1)$$
$$\boldsymbol{e}_m(t) = \boldsymbol{\theta}_d(t) - \boldsymbol{\theta}_m(t).$$

Here, $\boldsymbol{K}_p$ and $\boldsymbol{K}_d$ are proportional and derivative controller gain constants, respectively. $\boldsymbol{V}(t)$ is the output from the controller, $\boldsymbol{\theta}_m(t)$ is the position feedback of the motor

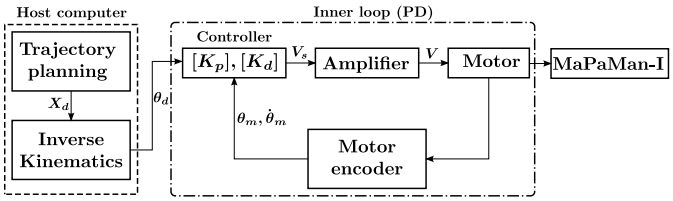

Fig. 2. Schematic of single-loop control

encoders, $\boldsymbol{e}_m(t)$ is the error in the position, and $\dot{\boldsymbol{e}}_m(t)$ is the corresponding velocity error. This does not see the errors accumulated due to any imperfections present in the system, including backlash in the gear box attached to the motor.

B. Dual-loop Control

In this method, an additional feedback loop is added to the single-loop control, using the measured rocker position, $\boldsymbol{\phi}_f$, as shown in Fig. 3. This is compared with the desired input to form the *outer loop* with a PD control as:

$$\begin{aligned} &\boldsymbol{\theta}_c = \boldsymbol{\theta}_d + \boldsymbol{K}_{po}\boldsymbol{e}_r(t) + \boldsymbol{K}_{do}\dot{\boldsymbol{e}}_r(t), \quad \text{where} \\ &\boldsymbol{e}_r(t) = \boldsymbol{\theta}_d(t) - \boldsymbol{\phi}_f(t). \end{aligned} \tag{2}$$

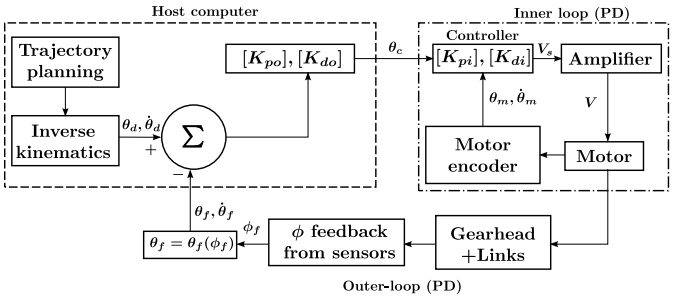

Fig. 3. Schematic of dual-loop control

Here, $\boldsymbol{\theta}_c$ is the commanded position trajectory sent to the inner-loop, and $\boldsymbol{e}_r(t)$ is the error in the position of the motor as seen from the rocker. This control strategy should be able to mitigate the errors present in the system due to the backlash, and thus the performance is expected to be better than the single-loop control.

C. Double Dual-loop Control

In this method another control loop is added to the dual-loop control, which corrects the errors present in the task space using an additional feedback, as shown in Fig. 4. Task space feedback, $\boldsymbol{X}_f(t)$, is estimated using both the rocker and strut encoders, which give $\boldsymbol{\phi}_f$ and $\boldsymbol{\gamma}_f$ respectively, as:

$$\boldsymbol{X}_f(t) = f(\boldsymbol{\phi}_f, \boldsymbol{\gamma}_f) \tag{3}$$

This is used to control the error in the task space using a proportional-integral (PI) control which forms the outer-loop 1 as:

$$\begin{aligned} &\boldsymbol{X}_{dl} = \boldsymbol{X}_d(t) + \boldsymbol{K}_{p1}\boldsymbol{e}_x(t) + \boldsymbol{K}_i \int_0^t \boldsymbol{e}_x(\tau)d\tau, \quad \text{where} \\ &\boldsymbol{e}_x(t) = \boldsymbol{X}_d(t) - \boldsymbol{X}_f(t). \end{aligned} \tag{4}$$

This is further brought down to the desired motor position, $\boldsymbol{\theta}_{dl}$, using inverse kinematic equations. The outer-loop 2 is now applied over the task space corrected position as:

$$\boldsymbol{\theta}_c = \boldsymbol{\theta}_{dl} + \boldsymbol{K}_{p2}\boldsymbol{e}_r(t) + \boldsymbol{K}_{do}\dot{\boldsymbol{e}}_r(t). \tag{5}$$

This control strategy should account for the error in the four bar mechanism in each leg, up to the strut in addition to the backlash in the gearheads. Thus theoretically, it is expected to perform better then the other two strategies described. This has been validated in Section IV.

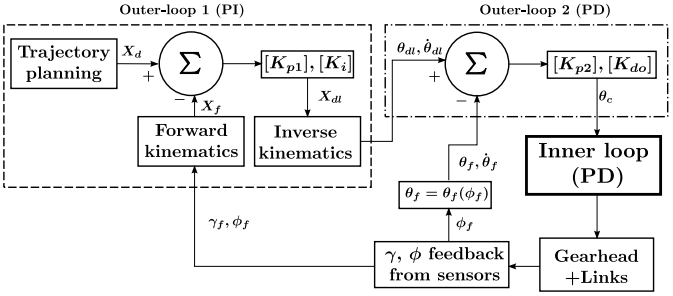

Fig. 4. Schematic of double dual-loop control

IV. Experimental Results

The three control strategies presented in Section III are validated for their performance by implementing them to track a defined trajectory on MaPaMan-I and the results are elaborated in this section[1]. The commanded trajectory data used in these experiments simulate the motion of a bicycle in tracking an '8' pattern on the road. The trajectory data is obtained experimentally by mounting an IMU on the seat of the bicycle. The desired trajectory has predominantly varying roll motion, with secondary pitch motion, and a constant heave position, with disturbances in the roll and pitch motions.

For multi-degree of freedom manipulator control, $\boldsymbol{K}_p$ and $\boldsymbol{K}_d$ are diagonal matrices and can be represented as scalar multiplied with the identity matrix of the required dimension. The gain matrices have been chosen as:

$$\begin{aligned} \boldsymbol{K}_p &= K_p\boldsymbol{I}, \\ \boldsymbol{K}_d &= K_d\boldsymbol{I}, \end{aligned}$$

such that $K_p, K_d > 0$ and $\boldsymbol{I}$ is an identity matrix. For the dual-loop and double dual-loop control, the values of the gains used are described in Table I and Table II, which are obtained by manual tuning for their best performance.

[1]All the angles given are in degrees and lengths are in millimetres, unless specified otherwise.

TABLE I

GAINS FOR DUAL-LOOP CONTROL

	Roll	Pitch	Heave
K_p	0.0005	0.0005	0.0005
K_d	$2\sqrt{K_p}$	$2\sqrt{K_p}$	$2\sqrt{K_p}$

TABLE II

GAINS FOR DOUBLE DUAL-LOOP CONTROL

	Roll	Pitch	Heave	Phi
K_p	0.005	0.005	0.0005	0.0005
K_i	10	2	2	-
K_d	-	-	-	$2\sqrt{K_p}$

The performance of each system is evaluated by comparing the errors for tracking the desired trajectory, which has been estimated using the IMU placed on the end-effector. Since the accuracy of the IMU system used in this setup is low, the errors are also compared by computing the end-effector position using the forward kinematics formulation and information from the encoders present in the rocker and the strut. For brevity, the results are shown only for a representative time window of 10s of the total simulation time of 126s, since the trend seen here is followed throughout. The experiments are repeated five times for each control strategy and the results presented are averaged over these runs for the maximum and RMS errors.

The desired and estimated trajectories obtained from the IMU are shown in Fig. 5 and 6. It can be observed that the performance of the system improves with each control loop being added, markedly in the roll motion of the system. The errors in tracking the trajectory estimated using the sensors are shown in Fig. 7, 8, and 9 for roll, pitch and heave motions respectively. It can be observed that the trajectory is followed with better accuracy as each control loop is added to the system. This can be quantitatively observed by comparing the maximum and RMS errors in each control strategy employed. The errors from using single-loop are given in Table III. The errors from using dual-loop control are given in Table IV, which show a better performance compared to the single-loop control as expected. Finally the results for double-dual loop control are given in Table V, whose RMS errors are the lowest among all the three. It has to be noted that the maximum errors do not necessarily follow the same trend and this is attributed to the low accuracy of the IMU and the noise present in the trajectory, where the stated control strategies do not perform adequately. The percentage improvement in the estimated RMS errors as compared to the single-loop control are calculated to be 23% and 63% for roll, 27% and 10% for pitch, and 20% and 67% for heave in the dual-loop and double dual-loop controls, respectively.

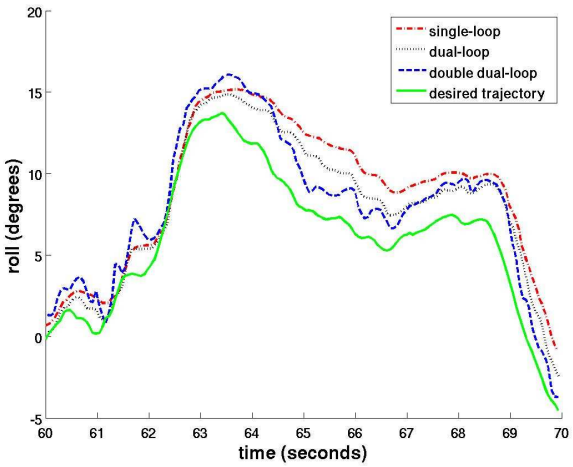

Fig. 5. Comparison of the roll motion of the trajectory as obtained from IMU for different control strategies. It can be observed that double dual-loop follows the desired trajectory most closely.

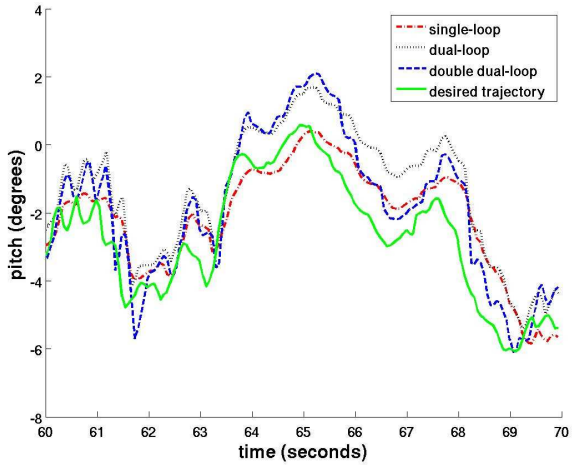

Fig. 6. Comparison of the pitch motion of the trajectory as obtained from IMU for different control strategies. It can be observed that double dual-loop follows the desired trajectory most closely.

TABLE III

ERRORS IN SINGLE-LOOP CONTROL

	Max Error			RMS Error		
	Roll	Pitch	Heave	Roll	Pitch	Heave
Computed[1]	3.01	2.27	4.00	1.32	0.83	1.92
IMU	5.43	3.98	-	2.51	1.02	-

[1] The computed end-effector position using the forward kinematics formulation and information from the encoders present in the rocker and the strut.

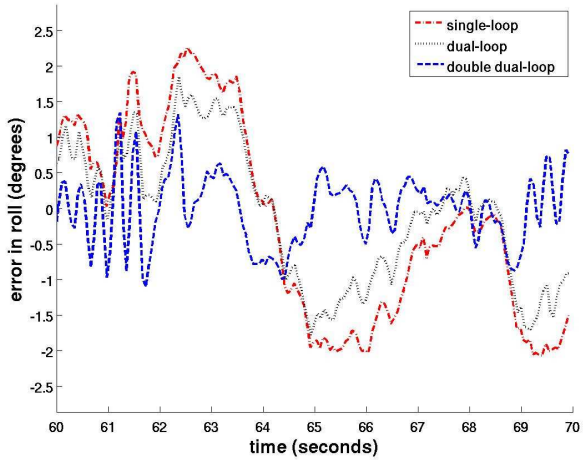

Fig. 7. Error in the roll motion of the trajectory tracking for different control strategies as compared to the computed values obtained from forward kinematics and information from sensors located at rockers and struts. The max and rms error values are given in Table III, Table IV, and Table V.

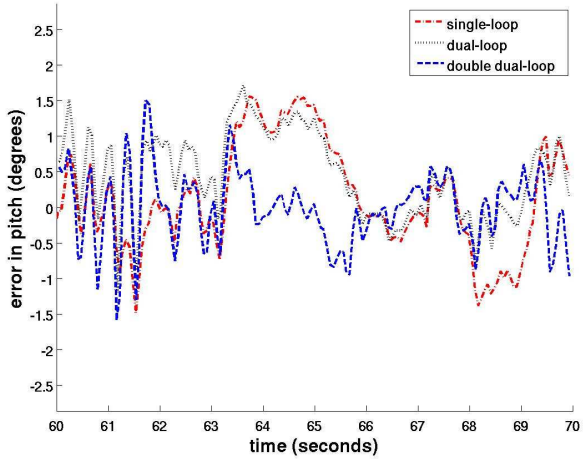

Fig. 8. Error in the pitch motion of the trajectory tracking for different control strategies as compared to the computed values obtained from forward kinematics and information from sensors located at rockers and struts.

TABLE IV

ERRORS IN DUAL-LOOP CONTROL

	Max Error			RMS Error		
	Roll	Pitch	Heave	Roll	Pitch	Heave
Computed	2.66	1.91	3.40	1.01	0.60	1.51
IMU	4.63	3.89	-	2.13	1.14	-

V. CONCLUSION

This paper presents a method to integrate redundant sensing with control strategies as a means to bypass calibration, and mitigate the errors arising from mechanical inaccuracies of a system. The method is used to correct the errors dynamically in a judicious manner leading to improvement in its trajectory tracking performance. It has been achieved

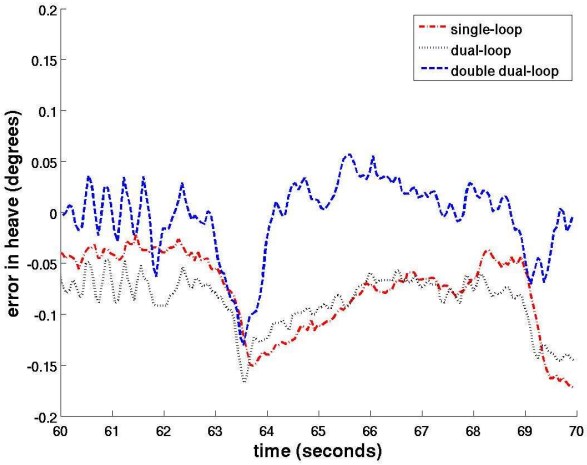

Fig. 9. Error in the pitch motion of the trajectory tracking for different control strategies as compared to the computed values obtained from forward kinematics and information from sensors located at rockers and struts.

TABLE V

ERRORS IN DOUBLE DUAL-LOOP CONTROL

	Max Error			RMS Error		
	Roll	Pitch	Heave	Roll	Pitch	Heave
Computed	2.19	2.43	3.14	0.49	0.49	0.62
IMU	3.97	4.22	-	1.66	1.63	-

using the feedback from redundant sensing appropriately and forming control loops to eliminate the errors due to the inaccuracies of backlash in gearheads, parametric uncertainities, unmodelled dynamics etc. The experimental analyses performed further validate that a marked improvement in the performance of the system can be achieved.

REFERENCES

[1] B. Rothstein and B. Paden, "Method and apparatus for controlling backlash in motor drive systems," Mar. 17 1998, US Patent 5,729,100.

[2] L. C. Hale and A. H. Slocum, "Design of anti-backlash transmissions for precision position control systems," *Precision Engineering*, vol. 16, no. 4, pp. 244–258, 1994.

[3] A. Agarwal, C. Nasa, and S. Bandyopadhyay, "Dual-loop control for backlash correction in trajectory tracking of a planar 3-RRR manipulator," in *Machines and Mechanisms, 2011. Proceedings., 15th National Conference on*, 2011, pp. 438–445.

[4] S. Hayati, K. Tso, and G. Roston, "Robot geometry calibration," in *Robotics and Automation, 1988. Proceedings., 1988 IEEE International Conference on*, Apr 1988, pp. 947–951 vol.2.

[5] R. He, Y. Zhao, S. Yang, and S. Yang, "Kinematic-parameter identification for serial-robot calibration based on POE formula," *Robotics, IEEE Transactions on*, vol. 26, no. 3, pp. 411–423, June 2010.

[6] A. Ghosal, *Robotics: Fundamental Concepts and Analysis*, 4th ed. New Delhi: Oxford University Press, 2009.

[7] R. A. Srivatsan and S. Bandyopadhyay, "On the position kinematic analysis of MaPaMan: A reconfigurable three-degrees-of-freedom spatial parallel manipulator," *Mechanism and Machine Theory*, vol. 62, pp. 150–165, 2013.

[8] T. Mehta, "Dynamics, trajectory-tracking and testing of MaPaMan-I," M. Tech. Report, Indian Institute of Technology Madras, Chennai, India, May 2012.

[9] J. Tal, "Two feedback loops are better than one," *Machine Design*, vol. 71, 1999.