

# Kinematic Task Space Control Scheme for 3DOF Pneumatic Parallel Robot

Luis Hernández<sup>1</sup>, Eduardo Izaguirre<sup>2</sup>, Ernesto Rubio<sup>3</sup>, Orlando Urquijo<sup>4</sup>  
and Jorge Guerra<sup>5</sup>

<sup>1,2,3,4</sup>Automation, Robotic and Perception Group (GARP). Universidad Central de Las  
Villas (UCLV)  
<sup>5</sup>SIMPRO  
Cuba

## 1. Introduction

Parallel robots have received special attention of the systems and control community based on its high force-to-weight ratio and widespread applications (Merlet, 2006). The control schemes for a parallel robot can be divided into two strategies: joint space control (Wang et al., 2009), (Gupta et al., 2008) and task-space control (Ting & Chen, 1999). The joint space control scheme is based on the information of each actuator length and can be implemented as a decoupled independent single-input single-output (SISO) control systems for each actuator, with in general, poor compensation of the uncertainties. On the other hand, a task space controller has a potential to provide a better control for the parallel robot under system uncertainties: inertia, modeling error, friction, etc. Nevertheless this type of control scheme needs the direct measurement of system task space state (Gao et al., 2008) or task space state estimation, normally cumbersome. Task space control schemes have been presented for direct inverse dynamics control, with joint space dynamic model compensation (Feng, 1995) or task space dynamics model compensation (Li & Xu, 2008), including vision based computed torque in task space control (Paccot et al., 2008). Many of this schemes have been proved by simulation or in laboratory testbed, but not commonly on industrial motion platform.

The goal of this work is to improve the performance of the control system of an industrial 3DOF pneumatic parallel platform, used by SIMPRO company who develop driver simulator. The solution of the control problem in the joint space of the parallel structure with pneumatic actuators have been obtained by Rubio (Rubio et al., 2009), using a decoupled control based on actuator lineal model around the operational point. But the performance of this scheme is strongly dependent of the precision of the kinematic model and dynamic uncertainties. The solution proposed, the *kinematic task space control*, is based on the direct measurement of system task space state. The control system considers two loops in cascade, an internal loop solving the robot' joint control ( $q$ ), and an external loop implementing the task space control ( $x$ ).

The proposed control philosophy, based on robot kinematic model, have been used by Hernández (Hernández et al., 2008a), (Hernández et al., 2008b) in vision-based 2D and 3D control of robot manipulators. In this scheme type, for digital implementation, it is possible

to approximate the dynamic effect of the internal loop as an external loop time delay (Corke, 1996), (Hernández et al., 2010) and (Bonfe et al., 2002). In this conditions a stability analysis in discrete time is developed.

To illustrate the proposed controller, the control system stability and its performance, both co-simulation results via MATLAB/Simulink and ADAMS, as well as experimental results using the SIMPRO 3DOF pneumatic parallel robot are presented. Experimental results confirm the expected step response in the task space, with good time performance and zero steady-state error. The control scheme presented open a new research field in the task space control with algorithms for the solution of, trajectory control, feed-forward control, etc.

## 2. Parallel robot model

As shown in Fig. 1, the robotic system considered consist of a 3DOF parallel manipulator controlled by pneumatic actuators. The parallel robot is produced by SIMPRO for driving simulator purpose, the robot have sensors to measurement the joint displacements and task space states. The basic mathematical description of this system consists of the parallel robot kinematics and dynamic model and also the actuator models.



Fig. 1. SIMPRO 3DOF pneumatic driver simulator parallel platform

### 2.1 Robot kinematics model

The same manner that serial manipulators, kinematics relations of parallel robots gives the relationship between the joints variables  $\mathbf{q}$  and the corresponding position  $(x, y, z)$  and angular orientation  $(\theta, \varphi, \psi)$  of center of mass of mobile platform in cartesian space. For an  $n$ -axis parallel structure, the forward kinematic (FK) solution  $\mathbf{T}$  could be numerically computed according with the number of joints of kinematic architecture (Merlet, 2006). Generic mathematical representation of forward kinematics could be:

$$\mathbf{x} = [x \ y \ z \ \theta \ \varphi \ \psi]^T = f(q_1, q_2, \dots, q_n) = \mathbf{T} \quad (1)$$

For parallel robots the complexity of FK equations increase noteworthy with the numbers of degree of freedom, the solution is non-unique and numerical methods are currently used to obtained the solutions, unfortunately there is no known algorithm that allows the

determination of the current pose of the platform among the set of solutions. Furthermore, the computation times involved in FK algorithms are still too large for use in a real time application (Merlet, 2006).

For robot path planning, the inverse kinematic (IK) expressions  $\mathbf{T}^{-1}$  gives the joint coordinates  $\mathbf{q}$  required to reach the specified pose of mobile platform. The mathematical expression of inverse kinematics can be written as:

$$\mathbf{q} = [q_1 \ q_2 \ \dots \ q_n]^T = g(x, y, z, \theta, \varphi, \psi) = \mathbf{T}^{-1} \quad (2)$$

Schematic of the 3-DOF parallel robot under study is shown in Fig. 2. The system consists of a fixed base connected to moving platform by three actuated kinematics chains, following the RPSU-2SPS architecture. A base coordinate frame designated as  $Oxyz$  frame is fixed at the center of the base with its  $z$ -axis pointing vertically upward and the  $x$ -axis pointing backwards of the platform. Similarly a moving coordinate frame  $Px'y'z'$  is assigned to the center mass of the moving platform, with the  $z'$ -axis normal to the mobile platform. By simplicity the directions of both  $z$  and  $z'$  axes are pointing in the same unit vector.

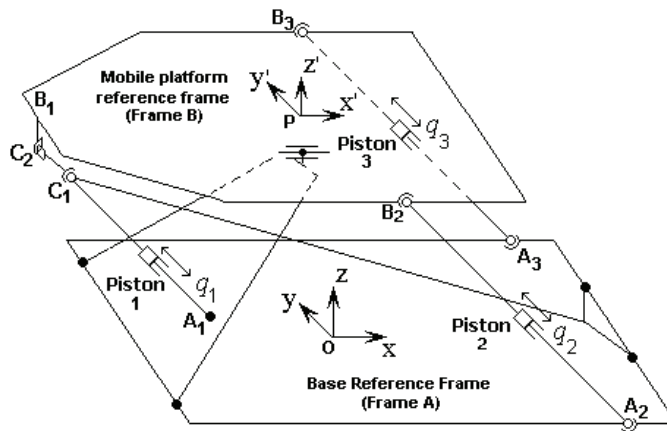


Fig. 2. The RPSU-2SPS kinematics structure of the 3-DOF motion platform.

The actuators are double effect electro-pneumatic cylinders whose lineal displacements produce the 3-DOF of robot, consisting in two rotations around the  $x'$  and  $y'$  axes, represented by roll ( $\theta$ ) and pitch ( $\varphi$ ) angles respectively, and linear displacement along the  $z'$  axis (elevation), defined by the variable  $h$ . So, the moving platform can simulate different sceneries in correspondence with virtual reality world that is shown in LCD display located inside the cabin with is supported by mobile platform. This type of applications are developed by SIMPRO company.

The vectorial formulation allows us to simply construct a set of equations which contains the same number of equations as the unknown variables. According with this procedure a closed vector cycle is constituted between the points  $A_i$  and  $B_i$  in correspondence with the illustration represented in Fig. 3.

Establish the inverse kinematics model is essential for the position control of robot. Then, for each kinematics chain, a vectorial function can be formulating by expressing the actuated joint coordinates as a function of cartesian coordinates ( $\mathbf{x}$ ) whose define the pose of the mobile

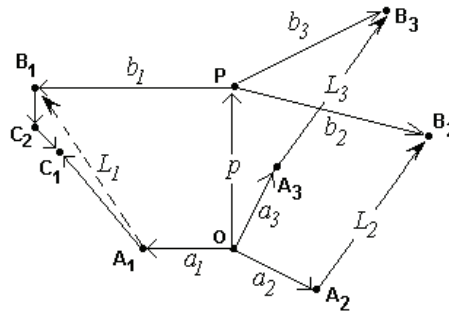


Fig. 3. Representation of closed loop vectors for each active legs.

platform. According to equation (2) is necessary to found the relation  $\mathbf{A}_i \mathbf{B}_i = g(\mathbf{x})$  in order to calculate the inverse kinematics model of robot.

Considering the schematic presented in Fig. 3, follows that:

$$\mathbf{p} + \mathbf{b}_i = \mathbf{a}_i + \mathbf{L}_i \quad (3)$$

where:

$$\mathbf{p} = \mathbf{OP} = [P_x, P_y, P_z]^T - [O_x, O_y, O_z]^T \quad (4)$$

$$\mathbf{a}_i = \|\mathbf{OA}_i\|_2 \quad (5)$$

$$\mathbf{b}_i = \|\mathbf{PB}_i\|_2 \quad (6)$$

$$\mathbf{L}_i = \|\mathbf{A}_i \mathbf{B}_i\|_2 \quad (7)$$

The position vector of the mobile platform with reference to the fixed frame is defined by the vector  $\mathbf{p} = \mathbf{OP}$ , that is, platform position in cartesian space is defined by  $z'$  coordinates of point  $P$ . Consequently the orientation of mobile platform is determined by  $\theta$  and  $\varphi$  angles.

Due to mechanical restrictions of robotic structure, the yaw angle is zero ( $\psi = 0$ ), then, the rotation matrix is defined by only the roll and pitch angles. Following the ZYX convention for Euler angles, the rotation matrix can be computed by the expression:

$$\mathbf{R}_{x'y'z'}^{xyz} = \mathbf{R}_B^A = \begin{bmatrix} \cos(\varphi) & \sin(\varphi)\sin(\theta) & \sin(\varphi)\cos(\theta) \\ 0 & \cos(\theta) & -\sin(\theta) \\ -\sin(\varphi) & \cos(\varphi)\sin(\theta) & \cos(\varphi)\cos(\theta) \end{bmatrix} \quad (8)$$

It can be easily seen that the described architecture is characterized by 3-DOF's in cartesian space; in general case the appropriated linear displacements of each actuator are related with changes of orientation and elevation of the mobile platform. Then, the location of the moving platform can be described by a position vector  $\mathbf{p}$  and a rotation matrix  $\mathbf{R}_B^A$ .

By construction the fixed reference frame coordinates of  $A_i$ 's points are known, while the coordinates of  $B_i$ 's may be determined from the position and orientation of mobile platform. Hence the vector  $\mathbf{A}_i \mathbf{B}_i$  plays an important role in the solution of inverse and direct kinematics problems.

Considering equations (3) to (7) and (8), the general closed-loop equation for the SIMPRO motion simulator platform as the form:

$$\mathbf{A}_i \mathbf{B}_i = \mathbf{OP} + \mathbf{R}_B^A \mathbf{PB}_i - \mathbf{OA}_i \quad (9)$$

The central kinematics chain (corresponding with the actuator No. 1) have different architecture for the others, and then, new additional equation is necessary to be considered:

$$\mathbf{A}_1 \mathbf{B}_1 = \mathbf{A}_1 \mathbf{C}_1 + \mathbf{C}_2 \mathbf{C}_1 - \mathbf{B}_1 \mathbf{C}_2 \quad (10)$$

Substituting (10) in (9) and considering the all closed loop equations for each active leg, we have:

$$\mathbf{A}_1 \mathbf{C}_1 = \mathbf{OP} + \mathbf{R}_B^A \mathbf{PB}_1 - \mathbf{OA}_1 + \mathbf{B}_1 \mathbf{C}_2 + \mathbf{C}_2 \mathbf{C}_1 \quad (11)$$

$$\mathbf{A}_2 \mathbf{B}_2 = \mathbf{OP} + \mathbf{R}_B^A \mathbf{PB}_2 - \mathbf{OA}_2 \quad (12)$$

$$\mathbf{A}_3 \mathbf{B}_3 = \mathbf{OP} + \mathbf{R}_B^A \mathbf{PB}_3 - \mathbf{OA}_3 \quad (13)$$

The design of robot included two additional rigid bodies interconnected between the base and the moving platform to provide mechanical stability and stiffness of robotic structure. This particular characteristic produce a curvilinear movement of mobile platform during elevation as is shown in Fig.4.

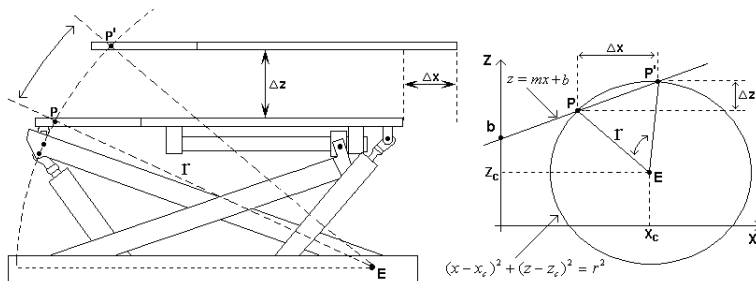


Fig. 4. Additional displacement  $\Delta x$  of mobile platform during elevation.

Consider the straight line between points  $PP'$  is defined by equation  $z = mx + b$ , where  $m$  and  $b$  are the slope and intercept respectively, additionally the circle with center in  $(x_c; z_c)$  and ratio  $r$  is described by the circumference equation  $(x - x_c)^2 + (z - z_c)^2 = r^2$ . Then, the movement along  $z$ -axis as a function of displacement of moving platform in the  $x$ -axis is calculated from:

$$\Delta z = m \left( \frac{2m(b - z_c) - 2x_c^2}{1 + m^2} - \frac{4k_1}{1 + m^2} \right) \quad (14)$$

where:

$$k_1 = x_c^2 + z_c^2 + b(b + 2z_c) - r^2$$

Because equation (14) compute the new coordinates of  $B_i$ 's points of mobile platform during elevation, it's incorporate to the solution of IK problem.

Given the initial position  $L_{oi}$  of limbs it is possible to calculate the displacements of lineal actuators  $q_i$  according to (15), considering the scheme of Fig. 5.

$$q_i = L_i - L_{oi} \quad (15)$$

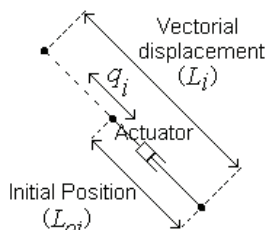


Fig. 5. Illustration of initial position of actuators in each robot's limbs.

## 2.2 Robot dynamics model

In the absence of friction or other disturbances, the dynamics of parallel robot can be written as (Merlet, 2006)

$$\mathbf{M}_x(\mathbf{x})\ddot{\mathbf{x}} + \mathbf{C}_x(\mathbf{x}, \dot{\mathbf{x}})\dot{\mathbf{x}} + \mathbf{G}_x(\mathbf{x}) = \mathbf{J}^T \mathbf{f} = \mathbf{f}_x \quad (16)$$

where:

- $\mathbf{M}_x(\mathbf{x})$   $n \times n$  symmetric positive definite manipulator inertia matrix
- $\mathbf{x}$   $n \times 1$  vector of space displacements
- $\mathbf{C}_x(\mathbf{x}, \dot{\mathbf{x}})$   $n \times n$  centripetal and Coriolis matrix
- $\mathbf{G}_x(\mathbf{x})$   $n \times 1$  vector of gravitational term
- $\mathbf{J}$   $n \times n$  robot Jacobian matrix
- $\mathbf{f}$   $n \times 1$  vector of applied joint forces

In other hand the dynamic model in joint space can have, in some configuration, the form, (Li & Xu, 2008)

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \mathbf{f} \quad (17)$$

where:

- $\mathbf{M}(\mathbf{q})$   $n \times n$  symmetric positive definite manipulator inertia matrix
- $\mathbf{q}$   $n \times 1$  vector of joint displacements
- $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$   $n \times n$  centripetal and Coriolis matrix
- $\mathbf{G}(\mathbf{q})$   $n \times 1$  vector of gravitational term
- $\mathbf{f}$   $n \times 1$  vector of applied joint forces

## 2.3 Pneumatic Actuators Model

Our platform actuators are pneumatic cylinders. To obtain the dynamic model of a pneumatic cylinder, normally (Brun et al., 2000), the following consideration are taken: only viscous friction is present, the temperature is constant and equal in both cylinder chambers and the gas is ideal. Additionally, according Rubio (Rubio, 2009), the influence of the underlap

characteristic of the valve is considered. Under this conditions, the transfer function of the system, position  $Y(s)$  versus control action  $U(s)$ , is obtained with the form,

$$\frac{Y(s)}{U(s)} = \frac{b_1 s + b_0}{s(c_3 s^3 + c_2 s^2 + c_1 s + c_0)} \quad (18)$$

Once defined the operating point of the valve, the coefficients  $c_i$  and  $b_i$ , are variables in dependent of the position  $y_0$ .

According the experimental work developed in (Rubio, 2009), a good approximation of equation (18) can be,

$$Gyu(s) = \frac{b}{s(s^2 + a_1 s + a_0)} \quad (19)$$

where  $b$  is the system gain,  $a_1 = 2\tilde{\zeta}\tilde{\omega}_n$  and  $a_2 = \tilde{\omega}_n^2$ ; where  $\tilde{\omega}_n$  and  $\tilde{\zeta}$  are respectively the open loop undamped natural frequency and damping ratio of the system. These parameter are not constants, they vary in dependency of the position  $y_0$ .

### 3. Control problem

The control problem is formulated as the design of a controller which computes a control signal  $\Delta$  corresponding to the movement of the robot's in such a way that the desired task space position be reaches following wanted performances index.

#### 3.1 Control problem formulation

For control the desired state,  $[\theta_d \quad \varphi_d \quad h_d]^T$ , is the position of center of mass platform moving part. The *task state error* being defined as:

$$\tilde{y} = y_d - y = \begin{bmatrix} \tilde{\theta} \\ \tilde{\varphi} \\ \tilde{h} \end{bmatrix} = \begin{bmatrix} \theta_d \\ \varphi_d \\ h_d \end{bmatrix} - \begin{bmatrix} \theta \\ \varphi \\ h \end{bmatrix}$$

which could be calculated at every measurement time and used to move the robot in a direction allowing its decrease. Therefore, the control aims at ensuring that

$$\lim_{t \rightarrow \infty} \tilde{y} = \lim_{t \rightarrow \infty} [\tilde{\theta} \quad \tilde{\varphi} \quad \tilde{h}]^T = 0$$

We make the as assumptions for the control problem that, only the control problem is evaluated with initial error  $\tilde{\zeta}(0)$  is sufficiently small and there exists a robot joint configuration  $\mathbf{q}_d$  for which  $\tilde{\zeta}_d = \tilde{\zeta}(\mathbf{q}_d)$ . This condition ensures that the control problem is solvable.

#### 3.2 Kinematic task space control

For our control problem formulation, the task space vector of the robot can be measured by external sensors. As such, a direct knowledge of the desired joint position  $\mathbf{q}_d$  is not available. Nevertheless, the desired joints position can be obtained as a result of the estimated control signal  $\Delta$  and the solution of the kinematics problems.

The implemented closed-loop block diagram can be described as shown in Fig. 6. The control system has two loops in cascade, the internal loop solving the robots' joint control, and the external loop implementing a *kinematic task space control*.

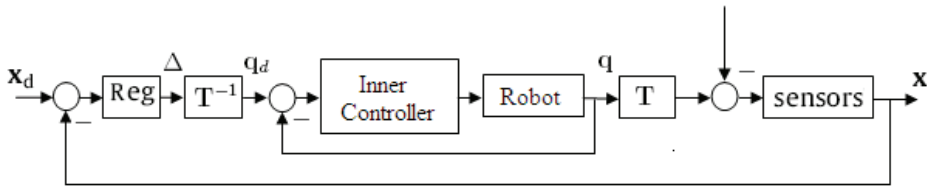


Fig. 6. Control scheme

The inner control loop has an open control architecture; in this architecture it is possible to implement any type of controller. One possibility is to use a non-linear controller in the state variables, called Model Based Computed Torque Control (Yang et al., 2008), having the following control equation, for a robot with  $n$ -DOF,

$$\tau = \mathbf{M}(\mathbf{q})[\ddot{\mathbf{q}}_d + \mathbf{K}_{vi}\dot{\mathbf{q}} + \mathbf{K}_{pi}\tilde{\mathbf{q}}] + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q})$$

Where  $\mathbf{K}_{pi} \in \mathbb{R}^{n \times n}$  and  $\mathbf{K}_{vi} \in \mathbb{R}^{n \times n}$  are the symmetric positive-definite matrices and  $\tilde{\mathbf{q}} = \mathbf{q} - \mathbf{q}_d$ . It is possible to demonstrated that with this configuration the system behaves in a closed loop as a linear multivariable system, decoupled for each robot's joint, suggesting that the matrices could be specified as:

$$\mathbf{K}_{pi} = \text{diag}\{\omega_1^2, \dots, \omega_n^2\}$$

$$\mathbf{K}_{vi} = \text{diag}\{\omega_1, \dots, \omega_n\}$$

In this way each joint behaves as a critically damping second order linear system with bandwidth  $\omega_i$ . The bandwidth  $\omega_i$  determines the speed of response of each joint. In such way the dynamic effect of the internal loop could be independent with regard to the external loop, being under the conditions that:

$$\mathbf{q}(t) = \mathbf{q}_d(t) \quad \forall t > 0 \quad (20)$$

This control philosophy have been used by Hernández (Hernández et al., 2008b) in vision-based 3D control of robot manipulators.

Nevertheless, control systems is implemented fully as a sampled data systems. Åström (Åström K. J. and Wittenmark, 1990) established that the sampling rate of digital control systems should be between 10 and 30 time the desired closed loop bandwidth. For the case low cost driver simulators system the close loop bandwidth should be around 0.1 Hz. A sampling rate for the external loop between 100 to 30ms is excellent for the wanted close loop bandwidth of the system.

In the analysis of the control problem in the field of digital control systems, other dynamic representation can be made for the internal loop, of the robot control system could be as one or two delay units of the external loop (Corke, 1996) or for the robot (Bonfe et al., 2002). Using this consideration we modify Equation (20) as,

$$\mathbf{q}(k) = \mathbf{q}_d(k-1) \quad \forall k > 0 \quad (21)$$



#### 4. Stability analysis

A simple  $I$  controller can be used in this control scheme (Hernández et al., 2008b), for that case the control law can be given by:

$$\Delta = \mathbf{K}_I \int \tilde{\xi} \quad (22)$$

Where  $\mathbf{K}_I \in \mathbb{R}^{2 \times 2}$  is the symmetric integral matrix:

$$\mathbf{K}_I = \begin{bmatrix} K_{I_1} & 0 & 0 \\ 0 & K_{I_2} & 0 \\ 0 & 0 & K_{I_3} \end{bmatrix} \quad (23)$$

Similarity as the visual control work (Hernández et al., 2010),  $\Delta$  can be interpreted as the coordinates increment in the task space as a result of the direct mensuration of center of mass of moving platform position. Solving the inverse kinematics problem  $T^{-1}$  it is possible to obtain  $\mathbf{q}_d$ .

Considering that the task space is measured by linear sensors its can be represented by the gain matrix  $\mathbf{K}$  as,

$$\mathbf{K} = \begin{bmatrix} K_\theta & 0 & 0 \\ 0 & K_\varphi & 0 \\ 0 & 0 & K_h \end{bmatrix}$$

Taking into account Fig. 6, obtaining the discrete equivalence of the controller of Equation (22), according to Equations (23) and (21) and taking a sampling period of  $60ms$ ; a simplified diagram can be obtained as shown in Fig. 7.

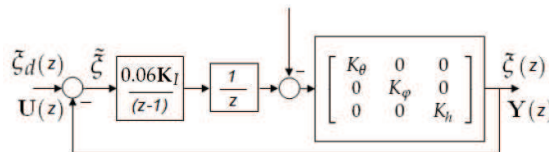


Fig. 7. Simplified control scheme

According to Fig. 7 the closed loop transfer function can be written as:

$$\frac{0.06\mathbf{K}_I\mathbf{K}}{(z^2 - z)} [\mathbf{Y}_d(z) - \mathbf{Y}(z)] = \mathbf{Y}(z) \quad (24)$$

Solving and taking the inverse Z transform we obtain:

$$\mathbf{y}(k+2) - \mathbf{y}(k+1) = -0.06\mathbf{K}_I\mathbf{K}\mathbf{y}(k) + 0.06\mathbf{K}_I\mathbf{K}\mathbf{y}_d(k) \quad (25)$$

The Equations (25) can be represented in the space state as,

$$\begin{bmatrix} \mathbf{y}(k+1) \\ \mathbf{y}(k+2) \end{bmatrix} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -0.06\mathbf{K}_I\mathbf{K} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \mathbf{y}(k) \\ \mathbf{y}(k+1) \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ 0.06\mathbf{K}_I\mathbf{K} \end{bmatrix} \mathbf{y}_d(k) \quad (26)$$

where

$$\mathbf{G} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -0.06\mathbf{K}_I\mathbf{K} & \mathbf{I} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ -0.06K_\theta K_{I_1} & 0 & 0 & 1 & 0 & 0 \\ 0 & -0.06K_\varphi K_{I_2} & 0 & 0 & 1 & 0 \\ 0 & 0 & -0.06K_h K_{I_3} & 0 & 0 & 1 \end{bmatrix}$$

and

$$\mathbf{H} = \begin{bmatrix} \mathbf{0} \\ 0.06\mathbf{K}_I\mathbf{K} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0.06K_\theta K_{I_1} & 0 & 0 \\ 0 & 0.06K_\varphi K_{I_2} & 0 \\ 0 & 0 & 0.06K_h K_{I_3} \end{bmatrix}$$

Following the the Equations (26) is easy to conclude that the control system is decoupled in each task space coordinate, with the form for the height,

$$\begin{bmatrix} h(k+1) \\ h(k+2) \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ -0.06K_{I_3}K_h & 1 \end{bmatrix} \begin{bmatrix} h(k) \\ h(k+1) \end{bmatrix} = \begin{bmatrix} 0 \\ 0.06K_{I_3}K_h \end{bmatrix} h_d(k) \quad (27)$$

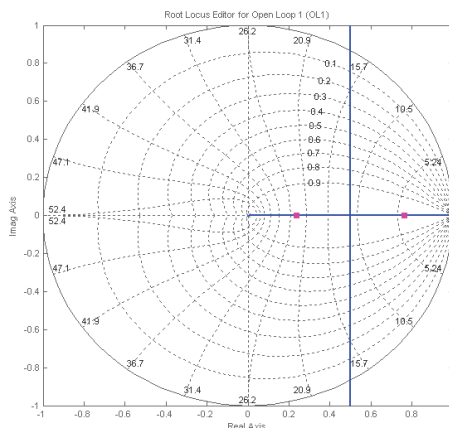


Fig. 8. Root-locus plot of height model for  $I$  controller with closed poles for gain  $K_h K_{I_i} = 3$ .

According equation (27) the root loci of height control system is presented in Fig 8, in this case the closed loop poles are selected for the gain  $K_h K_{I_i} = 3$ . An overdamped transient response to the step input is expected for this design. The system is stable with  $K_h K_{I_i} < 20$ . A similar analysis can be done for Roll and Pitch task space coordinates.

A better system performance is obtained with a  $PI$  controller, for example with the controller,

$$C = \frac{0.06(z + 0.2)}{(z - 1)}$$

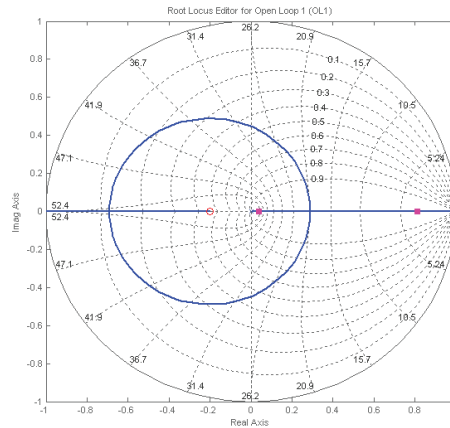


Fig. 9. Root-locus plot of height model for  $PI$  controller with closed poles for gain  $K_h K_{I_i} = 20$ .

The root locus plot is modified as is shown in the Fig. 9, in this case the system is stable with  $K_h K_{I_i} < 50$ .

## 5. Experimental study

A 3-DOF platform with open computer control architecture used by SIMPRO as movement simulator, is used as a study case, Fig. 1. The inner and the external loops are implemented in a Pentium-D 3.00-GHz connected to the robot through a Humusoft MF624 board which reads the potentiometer's joint position sensors, executes the control algorithm and gives the control signal to the electro-pneumatic valve with a sampling period of  $1ms$ . The task space position is acquired by the same card, reading an encoder for the height and Pitch and Roll via IMU. In this loop is solves the inverse kinematic problem and the values  $\mathbf{q}_d$  are given to the internal loops with sampling period of  $60ms$ . The control algorithm has been implemented using MATLAB/Simulink with the *Real Time Workshop Toolbox* and *Real Time Windows Target*.

### 5.1 Practical design consideration

As shown in Fig. 6 the control system has two loops. The external loop execute the Kinematic task space control. For the case low cost driver simulators system the close loop bandwidth should be around  $0.6rad/s$ . According (Åström K. J. and Wittenmark, 1990) the external loop sampling period should be between 100 to  $30ms$ . The equation (21) is satisfied, if we choose the dynamic of the internal loop dominated by conjugated complex poles with  $\zeta = 0.7$  and  $\omega_n = 10rad/s$ , with  $60ms$  of sapling period for the external loop. A similar performance is obtained if the sampling period is  $30ms$  and the condition (21) is modified as  $\mathbf{q}(k) = \mathbf{q}_d(k-2) \quad \forall k > 0$ .

#### 5.1.1 Internal loop control

The parallel robot joint actuators are pneumatic cylinders. For this control a linear controller by pole placement is proposed.

The actuator dynamic have the form of the equation (19). According Rubio's work (Rubio, 2009), the controller have the form,

Table 1. Main specifications of the SIMPRO 3 DOF parallel robot

Description	Parameters
Actuator displacements	500 mm
Max. acceleration of actuators	980 mm/s <sup>2</sup>
Initial elevation	1070 mm
Pitch and Roll angles	± 18 °
Elevation movement	± 215 mm
Mass	1034 kg
Load/mass ratio	2.18

$$\frac{S(s)}{R(s)} = \frac{k_p (s + k_i) (s^2 + a_1 s + a_0)}{s (s + \omega_a)^2} \quad (28)$$

The transfer function of the internal closed loop is proposed as,

$$\frac{B_m}{A_m} = \frac{k_p b (s + k_i) (s^2 + a_1 s + a_0)}{(s + p_1)(s + p_2)(s^2 + 2\zeta\omega_n s + \omega_n^2)(s^2 + a_1 s + a_0)} \quad (29)$$

where  $p_1$  and  $p_2$  are two non dominant poles, selected by the designer joint to  $\omega_n$  and  $\zeta$ . With this parameters, based in the equation (29), the coefficients of controller (28) are found by the pole placement method.

## 5.2 Simulation

Following the scheme of Fig.6 a simulation process has been developed using MATLAB/Simulink-ADAMS. The principal parameters of the robot's geometric configuration Fig. 2 are specified in Table 1.

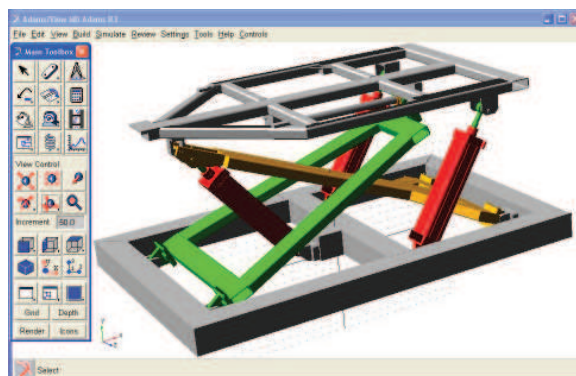


Fig. 10. Virtual model of 3-DOF parallel robot development in ADAMS.

### 5.2.1 Kinematic model

Because kinematics studies the relations of the joint motions and the center of mass of moving platform movement, the interest of equations involving the kinematics relations are very important for control purposes of robots with parallel configurations.

The main specifications of motion platform under study is summarize in the Table 1, note the excellent load capacity and the relatively small workspace both typical characteristics of parallel robots.

Initial value of elevation of moving platform is  $h=1265$  mm; and the corresponding coordinates locations (in mm) of points  $A_i$ ,  $B_i$  and  $C_1$ ,  $C_2$  are given below.

$$\begin{aligned}A_1 &= [-180; 0; 0]^T \\A_2 &= [180; -500; 0]^T \\A_3 &= [180; 500; 0]^T \\B_1 &= [-840; 0; h]^T \\B_2 &= [896; -500; h]^T \\B_3 &= [896; 500; h]^T \\C_1 &= [-760; 0; h - 210]^T \\C_2 &= [-840; 0; h - 150]^T\end{aligned}$$

Using the notation illustrated in Fig.3, the inverse kinematic problem can be solved by writing the loop closure equation for each actuated kinematic chain.

According with Table 1 and the equations (11) to (13) the of joints coordinates  $\mathbf{q}$  are found, knowing the values of the elevation of moving platform ( $h$ ), the roll and pitch angles ( $\theta, \varphi$ ), and the rotation matrix (8) as,

$$(L_1)^2 = [2076 - \lambda_0 - 940\cos(\varphi)]^2 + [740 + h + 940\sin(\varphi)]^2 \quad (30)$$

$$(L_2)^2 = [1397 - \lambda_0 + 720\cos(\varphi) + \lambda_1]^2 + \lambda_2^2 + (\lambda_3 - \lambda_4)^2 \quad (31)$$

$$(L_3)^2 = [1397 - \lambda_0 + 720\cos(\varphi) - \lambda_1]^2 + \lambda_2^2 + (-\lambda_3 - \lambda_4)^2 \quad (32)$$

where:

$$\begin{aligned}\lambda_0 &= \sqrt{1672^2 - 1720h - h^2} \\ \lambda_1 &= 500\sin(\theta)\sin(\varphi) \\ \lambda_2 &= 500\cos(\varphi) - 500 \\ \lambda_3 &= 500\sin(\theta)\cos(\varphi) \\ \lambda_4 &= 720\sin(\varphi) + 945 + h\end{aligned}$$

Substituting (30), (31) and (32) into (15) is possible to compute the vector of joints values  $[q_1, q_2, q_3]^T$  of parallel robot from cartesian pose of center of mass of moving platform  $[\theta, \varphi, h]^T$ , establishing the inverse kinematic problem.

$$\begin{aligned}q_1 &= \sqrt{L_1} - L_{01} \\ q_2 &= \sqrt{L_2} - L_{02} \\ q_3 &= \sqrt{L_3} - L_{03}\end{aligned} \quad (33)$$

The parallel robot have been modeled in the software package MSC.Adams (ADAMS, 2005), excellent tool for the kinematic and dynamic simulation of complex mobile structures, (Li & Xu, 2008). So, the inverse kinematic equations have been validated using the robot's

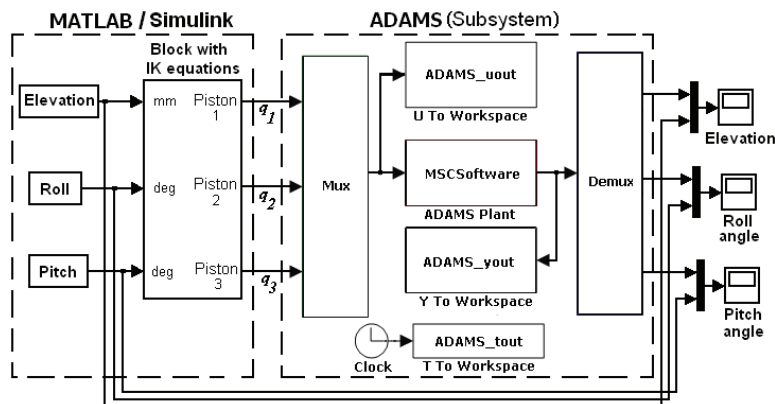


Fig. 11. Kinematic simulation Matlab/Simulink-ADAMS.

geometric virtual model in a co-simulation Matlab/Simulink-ADAMS, according with the block diagram of Fig. 11.

To obtain the desired values of actuated joints positions from the desired trajectory of mobile platform, the IK equations expressing in (33) are programming in Matlab/Simulink. The functions of  $h(t)$ ,  $\theta(t)$ , and  $\varphi(t)$  are the inputs of the IK block and the variables  $q_1$ ,  $q_2$ ,  $q_3$  are the outputs; the block with the geometric model of robot (developed in ADAMS) received these variables as inputs which are then exported to ADAMS environment. The resulted outputs (position and orientation of the moving platform) of the virtual model in ADAMS are compared with the desired trajectory of center of mass of moving platform.

### 5.2.2 Joint control

The electro-pneumatic system consist of proportional flow valve type MPYE-5-3/8 connected to double-acting pneumatic cylinder FESTO DNC-125-500. To perform the dynamic identification of electro-pneumatic system the scheme of Fig. 12 was implemented. The bandwidth and magnitude of pseudo random binary signal (PRBS) in combination with the proportional gain  $K_p$  have been selected to obtain the adequate excitation of system.

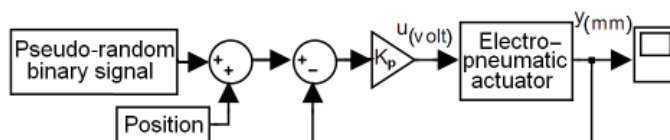


Fig. 12. Block diagram used in the dynamic identification of electro-pneumatic system.

The results of experimental dynamic identification, for the electro-pneumatic systems models (19) are,  
Piston 1

$$\frac{Y_1(s)}{U_1(s)} = \frac{245.94}{s(s^2 + 27.72s + 253)} \quad (34)$$

Pistons 2 and 3

$$\frac{Y_2(s)}{U_2(s)} = \frac{Y_3(s)}{U_3(s)} = \frac{2008.3}{s(s^2 + 7.276 + 1349)} \quad (35)$$

The closed loop system is designed for two conjugated complex poles dominated with  $\zeta = 0,7$  and  $\omega_n = 10 \text{ rad/s}$ . According the equations (28) and (29), using pole placement method (Rubio et al., 2009), are obtained the transfers functions of controllers, Controller for actuator 1

$$\frac{U_1(s)}{E_1(s)} = \frac{265(s^2 + 7.726s + 253)(s + 3.03)}{s(s^2 + 146.7s + 6267)} \quad (36)$$

Controllers for actuators 2 and 3

$$\frac{U_2(s)}{E_2(s)} = \frac{U_3(s)}{E_3(s)} = \frac{32(s^2 + 7.726s + 1349)(s + 3.03)}{s(s^2 + 146.7s + 6267)} \quad (37)$$

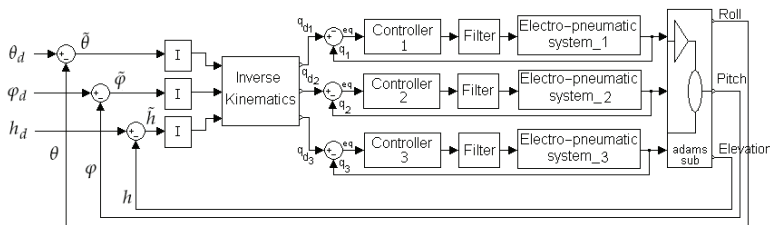


Fig. 13. Kinematic task space control simulation Matlab/Simulink-ADAMS.

### 5.2.3 Kinematic task space control simulation

The general control strategy is simulated according the block diagram of Fig. 13. In Matlab/Simulink is implemented: the control of the external loop (22), with  $K_\theta K_{I_1} = K_\phi K_{I_2} = K_h K_{I_3} = 3$ ; the Inverse kinematic equations, (30) to (33); and the decoupled joint control, equations (36) and (37). The joint control output is introduced to the no-linear cylinder model of each joint (Rubio et al., 2009), which give the position input to the ADAMS geometric model. The ADAMS outputs are the task space position, used as feedback for the external loops.

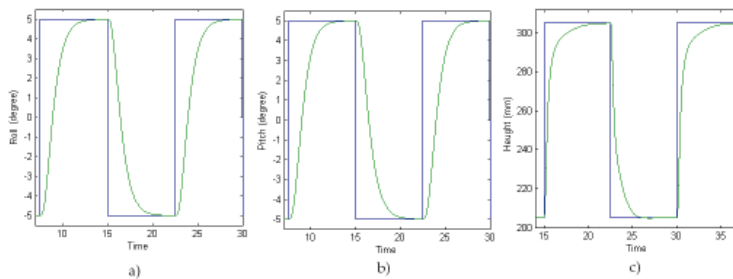


Fig. 14. ADAMS-Matlab/Simulink Simulation output, in blue the wanted value, in green the output. a) Roll, b) Pitch and c) Height.

The results of the simulation process is shown in the Fig. 14 where in blue are wanted value and in green the ADAMS output. In Fig. 14 a) Roll, in b) Pitch and in c) Height.

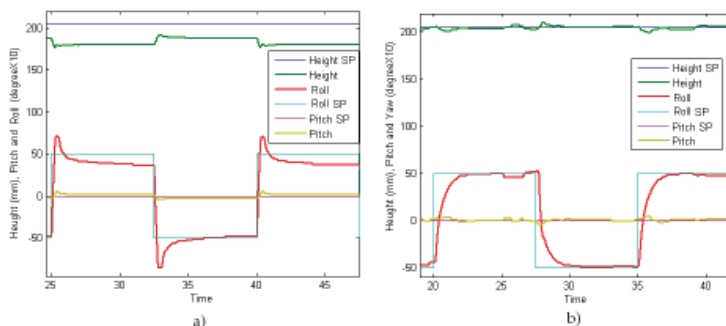


Fig. 15. Platform task space output to step variation in  $\phi_d$ . a) With only joint control b) With joint control and kinematic task space control.

### 5.3 Experimental results

The control scheme proposed was implemented on the 3-DOF parallel robot used by SIMPRO as moving simulator. The control algorithm of the inner and the external loops have been implemented using MATLAB/Simulink with the *Real Time Workshop Toolbox* and *Real Time Windows Target*. The task space variables are measures with an encoder for the height and Pitch and Roll via IMU. Some experiments have been developed in real platform, initially the system receive step variation in  $\phi_d$ , and the task space output is evaluated with only joint control, Fig. 15 a), and after the complete scheme proposed with joint control in the internal loops and kinematic task space control in the external loop, Fig. 15 b).

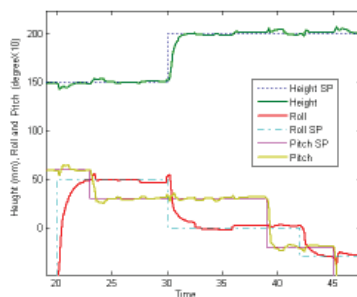


Fig. 16. Platform task space output with kinematic task space control with step variation in:  $\theta_d$ ,  $\phi_d$  and  $h_d$ .

The better performance of the kinematic task space control is evident, the steady state error disappear, not only in  $\phi$ , but also in  $h$ , which receive variation in the scheme with only joint control. The actual transient response is similar as the theoretically predict and similar as the simulation process with Matlab/Simulink-ADAMS, Fig.14. In the Fig. 16 is presented the platform task space output to simultaneous step variation in,  $\theta_d$ ,  $\phi_d$  and  $h_d$ , with good performance in steady state and transient response.



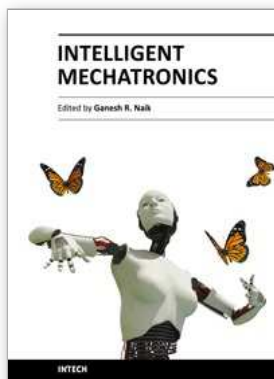
## 6. Conclusion

In this work a *kinematic task space control* scheme is implemented in an 3DOF pneumatic parallel industrial robot, produced by SIMPRO company for motion simulator applications. The controller is structured in two loops in cascade, an internal loop solving the robot's joint control, and an external loop implementing the task space control. The dynamic effect of the internal loop is approximated as an external loop time delay, because the internal loop is designed faster than external loop. In these conditions the stability of the whole system in discrete time is balanced for I controllers in the external loop. The robot's inverse kinematics, the robot's model and simulation in ADAMS and the control of the pneumatic actuator of the internal loop are important solution in the control scheme implementation. In the chapter is demonstrated the improvement of the performance of the control system with the scheme *kinematic task space control*, in relation with the joint control scheme. In this direction co-simulation via MATLAB/Simulink and ADAMS, as well as experimental results using the SIMPRO 3DOF pneumatic parallel robot are presented. The experimental results confirm the expected performance of the system in the task space. As future works the control scheme presented can be improved with better controller in the external loop in order to be faster the control system, following trajectory, etc.

## 7. References

- ADAMS (2005). *MSC.ADAMS Basic Full Simulation Package*, [www.mscsoftware.com](http://www.mscsoftware.com).
- Åström K. J. and Wittenmark, B. (1990). *Computer controlled systems: theory and design*, Englewood Cliffs (NJ).
- Bonfe, M., Minardi, E. & Fantuzzi, C. (2002). Variable structure pid based visual servoing for robotic tracking and manipulation, in IEEE (ed.), *International Conference on Intelligent Robots and Systems*, Lausanne, Switzerland.
- Brun, X., Belgharbi, M., Sesmat, S., Thomasset, D. & Scavarda, S. (2000). Control of an electropneumatic actuator, comparison between some linear and nonlinear control laws, *Journal of Systems and Control Engineering* (Control in Fluid Power Systems).
- Corke, P. I. (1996). *Visual control of robots : high-performance visual servoing*, Robotics and mechatronics series ; 2, Research Studies Press ; Wiley, Taunton, Somerset, England New York.
- Feng, G. (1995). A new adaptive control algorithm for robot manipulators in task space, *IEEE TRANSACTIONS ON ROBOTICS AND AUTOMATION*. Vol. 11.(NO. 3): 457–462.
- Gao, J., Webb, P. & Gindy, N. (2008). Vevaluation of a low-cost inertial dynam. measurement system, in IEEE (ed.), *Proc. of the IEEE Conf. on Robotics, Automation and Mechatronics*, Chengdu, China, pp. 21–24.
- Gupta, A., O'Malley, M. K., Patoglu, V. & Bugar, C. (2008). Design, control and performance of ricewrist: A force feedback wrist exoskeleton for rehabilitation and training, *The Internat. Journal of Robotics Research* Vol. 27(No. 2): 233–251.
- Hernández, L., Sahli, H. & González, R. (2010). Vision-based 2d and 3d control of robot manipulators, in A. Jimenez (ed.), *Robot Manipulators Trends and Development*, IN-TECH, Vienna, Austria, pp. 441–462.
- Hernández, L., Sahli, H., González, R. & González, J. (2008b). Simple solution for visual servoing of camera-in-hand robots in the 3d cartesian space., *Proceedings in the Tenth International Conference on Control, Automation, Robotics and Vision (ICARCV 2008)*., IEEE, Hanoi, Viet Nam, pp. 2020–2025.

- Hernández, L., Sahli, H., González, R., Rubio, E. & Guerra, Y. (2008a). A decoupled control for visual servoing of camera-in-hand robot with 2d movement., *Proceedings in Electronics, Robotics and Automotive Mechanics Conference, 2009. CERMA '08.*, IEEE, Morelos, México, pp. 304–309.
- Li, Y. & Xu, Q. (2008). Dynamic modeling and robust control of a 3-prc translational parallel kinematic machine, *Robotics and Computer-Integrated Manufacturing.* . [www.elsevier.com/locate/rcim](http://www.elsevier.com/locate/rcim).
- Merlet, J. P. (2006). *Parallel Robots, 2nd Edition*, Springer, France, ISBN: 1402041322.
- Paccot, F., Lemoine, P., Andreff, N., Chablat, D. & Martinet, P. (2008). A vision-based computed torque control for parallel kinematic machines, *Proceedings of IEEE International Conference on Robotics and Automation.*, IEEE, Pasadena, USA.
- Rubio, E. (2009). *Modelación, identificación y control de actuadores electro-neumáticos para aplicaciones industriales*, PhD thesis, Universidad Central "Marta Abreu" de Las Villas.
- Rubio, E., Hernández, L., Aracil, R., Saltaren, R. & Guerra, J. (2009). Implementation of decoupled model-based controller in a 2-dof pneumatic platform used in low-cost driving simulators., *Proceedings in Electronics, Robotics and Automotive Mechanics Conference, 2009. CERMA '09.*, IEEE, Cuernavaca, Morelos, México, pp. 338 – 343.
- Ting, Y. & Chen, Y. S. and Wang, S. M. (1999). Task-space control algorithm for stewart platform., *Proceedings of the 38th conference on decision and control*, IEEE, Phoenix, Arizona, USA, p. 3857–3862.
- Wang, J., Wu, J., Wang, L. & You, Z. (2009). Dynamic feed-forward control of a parallel kinematic machine, *Mechatronics* Vol. 19(No. 3): 313–324.
- Yang, Z., Wu, J., Mei, J., Gao, J. & Huang, T. (2008). Mechatronic model based computed torque control of a parallel manipulator, *International Journal of Advanced Robotic Systems* 5(1): 123–128. [www.elsevier.com/locate/rcim](http://www.elsevier.com/locate/rcim).



## **Intelligent Mechatronics**

Edited by Prof. Ganesh Naik

ISBN 978-953-307-300-2

Hard cover, 248 pages

**Publisher** InTech

**Published online** 28, February, 2011

**Published in print edition** February, 2011

This book is intended for both mechanical and electronics engineers (researchers and graduate students) who wish to get some training in smart electronics devices embedded in mechanical systems. The book is partly a textbook and partly a monograph. It is a textbook as it provides a focused interdisciplinary experience for undergraduates that encompass important elements from traditional courses as well as contemporary developments in Mechatronics. It is simultaneously a monograph because it presents several new results and ideas and further developments and explanation of existing algorithms which are brought together and published in the book for the first time.

### **How to reference**

In order to correctly reference this scholarly work, feel free to copy and paste the following:

Luis Hernández, Eduardo Izaguirre, Ernesto Rubio, Orlando Urquijo and Jorge Guerra (2011). Kinematic Task Space Control Scheme for 3DOF Pneumatic Parallel Robot, Intelligent Mechatronics, Prof. Ganesh Naik (Ed.), ISBN: 978-953-307-300-2, InTech, Available from: <http://www.intechopen.com/books/intelligent-mechatronics/kinematic-task-space-control-scheme-for-3dof-pneumatic-parallel-robot>

# **INTeCH**

open science | open minds

### **InTech Europe**

University Campus STeP Ri  
Slavka Krautzeka 83/A  
51000 Rijeka, Croatia  
Phone: +385 (51) 770 447  
Fax: +385 (51) 686 166  
[www.intechopen.com](http://www.intechopen.com)

### **InTech China**

Unit 405, Office Block, Hotel Equatorial Shanghai  
No.65, Yan An Road (West), Shanghai, 200040, China  
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元  
Phone: +86-21-62489820  
Fax: +86-21-62489821