



# General Solution for the Dynamic Modeling of Parallel Robots

Wisama Khalil, Ouarda Ibrahim

## ► To cite this version:

Wisama Khalil, Ouarda Ibrahim. General Solution for the Dynamic Modeling of Parallel Robots. Journal of Intelligent and Robotic Systems, 2007, 49, pp. 19-37. hal-00401684

**HAL Id: hal-00401684**

**<https://hal.science/hal-00401684>**

Submitted on 3 Jul 2009

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# General Solution for the Dynamic Modeling of Parallel Robots

**Wisama KHALIL . Ouarda IBRAHIM**

*IRCCyN, U.M.R 6597 C.N.R.S, Ecole Centrale de Nantes  
1 Rue de la Noë, BP 92101, 44321 Nantes Cedex 03, France  
e-mail: wisama.khalil@irccyn.ec-nantes.fr*

**Abstract.** In this paper, we present a general method to calculate the inverse and direct dynamic models of parallel robots. The models are expressed in a closed form by a single equation in which all the elements needed are expressed. The solution is given in terms of the dynamic models of the legs, the dynamics of the platform and some Jacobian matrices. The proposed method is applied in this paper on two parallel robots with different structures.

**Key words:** Dynamic modeling, complex structures, parallel robots, simulation, Jacobian matrix.

## 1 Introduction

The parallel robots are complex multi-body systems that are difficult to model because of their several closed loops. Dynamic modeling is essential for design specifications and advanced control of parallel robots. Different techniques of the dynamic modeling of closed-loop manipulators are available in the works of [43, 33, 27, 36]. The work on the dynamics of parallel manipulator started with the dynamic analysis of Stewart platforms [17, 13]. Those studies mostly dealt with either the oscillation or the inverse dynamics problem under very simple frameworks. Later, other works presented more elaborated analysis to solve the dynamic modeling of parallel manipulators using different mechanical formalisms. For example the works of [29, 14, 28, 1, 3, 4, 31] used Lagrange–Euler formalism. The principle of virtual work has been used by [6, 41]. The principle of Hamilton was used in the dynamic modeling by [35]. On the other hand, Newton-Euler equations have been used by [40, 38, 19, 15, 16, 9, 10, 11, 23].

This paper presents a simple and general closed form solution for the inverse and direct dynamic models of parallel robots. The proposed method can be applied systematically to most parallel structures. The models are obtained by projecting the dynamics of the legs and the platform on the actuated joint axes using some simple Jacobian matrices. The dynamic models of the legs are expressed in the joint coordinates and the dynamics of the platform is expressed in the Cartesian variables.

The dynamic and kinematic models of the legs are obtained using classical methods devoted to serial robots and eventually with simple loops. Consequently, the computational complexity of the proposed models can be reduced by making use of the techniques, which were

developed many years ago for the serial robots. The dynamics of the platform is obtained by Newton-Euler equations giving the total forces and moments on a solid body.

Since the Jacobian matrix of the robot is needed we propose a new method for the calculation of the inverse Jacobian matrix of parallel robots using the Jacobian matrices of the legs.

In this paper, we apply the proposed method to the following robots:

- The Space robot, which has six degrees of freedom and three legs [5],
- The C5 robot, which has six degrees of freedom and six legs [8].

The paper is organised as follows: in section 2 we present how to obtain the inverse dynamic modeling, in section 3 we present the solution of the direct dynamic modeling, then the two examples will be treated in the subsequent sections. In Appendix we present a general method for the calculation of the inverse kinematic Jacobian matrix of parallel robots.

## 2 Inverse dynamic modeling of parallel robots

A parallel robot is a complex multi-body system having several closed loops. It is composed of a moving platform connected to a fixed base by parallel legs. We consider non redundant robots, where the number of active joints is equal to the number of degrees of freedom of the platform. The number of legs is denoted by  $m$  and the number of d.o.f of the platform is denoted by  $n$ . We assign the frame  $\Sigma_p$  fixed with the platform and frame  $\Sigma_0$  fixed with the base.

The inverse dynamic model gives the forces and torques of motorized joints as a function of the desired trajectory of the mobile platform. Before presenting our method we recall in the next section a classical method, which is generally used to model robots with closed loops.

### 2.1 CLASSICAL DYNAMIC MODELING OF CLOSED LOOP STRUCTURES

The robot inverse dynamic model of a closed loop structure, giving the forces and torques of actuated joints, denoted by  $\Gamma$ , is calculated as a function of the dynamic model of an equivalent tree structure by the following equation [27, 36, 24]:

$$\Gamma = G^T \Gamma_{tr} \quad (1)$$

$\Gamma_{tr}$  the inverse dynamic model of the equivalent tree structure, which is computed in terms of its joint variables  $(\mathbf{q}_{tr}, \dot{\mathbf{q}}_{tr}, \ddot{\mathbf{q}}_{tr})$ , such that:

$$\Gamma_{tr} = \mathbf{A}_{tr}(\mathbf{q}_{tr})\ddot{\mathbf{q}}_{tr} + \mathbf{h}_{tr}(\mathbf{q}_{tr}, \dot{\mathbf{q}}_{tr}) \quad (2)$$

where  $\mathbf{A}_{tr}$  is the inertia matrix of the equivalent tree structure, and  $\mathbf{h}_{tr}$  is the vector of the Coriolis, centrifugal and gravity forces of the equivalent tree structure. Many methods are available to calculate these elements [7, 2, 24].

$\mathbf{G}$  is the Jacobian matrix of the tree structure variables  $\mathbf{q}_{tr}$  with respect to the variables of active joints  $\mathbf{q}_a$ .

$$\mathbf{G} = \frac{\partial \mathbf{q}_{tr}}{\partial \mathbf{q}_a} \quad (3)$$

Since time does not appear explicitly in the relation  $\mathbf{q}_{tr} = \mathbf{f}(\mathbf{q}_a)$ , the matrix  $\mathbf{G}$  can also be calculated as the kinematic Jacobian matrix:

$$\mathbf{G} = \frac{\partial \dot{\mathbf{q}}_{tr}}{\partial \dot{\mathbf{q}}_a} \quad (4)$$

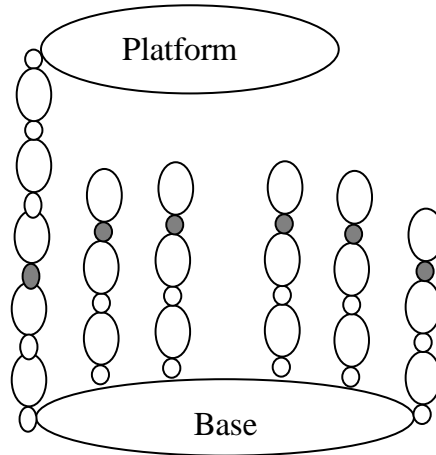
The demonstration of relation (1) is straightforward using the principle of virtual work (or virtual power) which states that the work done (or power delivered) of the actuators of the tree structure is equivalent to that of the active joints of the closed system:

$$\Gamma^T d\mathbf{q}_a = \Gamma_{tr}^T d\mathbf{q}_{tr} \quad (5)$$

$$\Gamma^T \dot{\mathbf{q}}_a = \Gamma_{tr}^T \dot{\mathbf{q}}_{tr} \quad (6)$$

The equivalent tree structure is obtained from the closed loop structure by virtually cutting each loop at one of its passive joints.

Applying this procedure for parallel robots, the platform will be attached to one leg (*Figure 1* shows the case of Gough-Stewart robot), thus the platform dynamics is obtained in terms of the joint variables of the leg on which it is attached in the equivalent tree structure. As an example, in the case of Gough-Stewart robot the platform dynamics will be a function of the six joint variables of the leg with which it is attached, which makes the model very complicated. Besides, the calculation of  $\mathbf{G}$  and the determination of the leg variables on which the platform is attached are also very complicated.



*Figure 1.* The tree equivalent structure of the parallel robot

## 2.2 INVERSE DYNAMIC MODEL OF PARALLEL ROBOTS

To obtain the dynamic models of parallel robots, we propose to exploit their structural characteristics by decomposing the system into two subsystems: the platform and the legs.

The dynamics of the platform is calculated as a function of the Cartesian variables (spatial Cartesian position, velocity and acceleration of the platform), whereas the dynamics of the legs are calculated as a function of the joint variables of the legs  $(\mathbf{q}_i, \dot{\mathbf{q}}_i, \ddot{\mathbf{q}}_i)$ . The active joint torques are obtained by the sum of these dynamics after projecting them on the active joint axes.

Using the same idea of equation (1), to project the dynamics of the platform into the active joint space we multiply it by the transpose of the robot Jacobian matrix, and to project the leg dynamics into the active joint space we have to use the Jacobian between these two spaces. Thus the dynamic model of the parallel structure is given by the following equation:

$$\mathbf{\Gamma} = \mathbf{J}_p^T \mathbf{F}_p + \sum_{i=1}^m \left( \frac{\partial \dot{\mathbf{q}}_i}{\partial \dot{\mathbf{q}}_a} \right)^T \mathbf{H}_i \quad (7)$$

with:

$\mathbf{F}_p$  is the total forces and moments on the platform,

$\mathbf{J}_p$  is the  $(6 \times n)$  kinematic Jacobian matrix of the robot, which relates the platform velocity  $\mathbf{V}_p$  (translational and angular) as a function of the active joint velocities:

$$\mathbf{V}_p = \mathbf{J}_p \dot{\mathbf{q}}_a \quad (8)$$

$\mathbf{H}_i$  is the inverse dynamic model of leg  $i$ , it is a function of  $(\mathbf{q}_i, \dot{\mathbf{q}}_i, \ddot{\mathbf{q}}_i)$ , which can be obtained in terms of the platform location, velocity and acceleration, using the inverse kinematic models of the legs. We note that  $\mathbf{q}_i$  does not include the passive joint variables connecting the legs to the platform.

If the number of degrees of freedom of the robot is less than 6, the vector  $\mathbf{V}_p$  can be reduced to a vector  $\mathbf{V}_r$  which contains the  $n$  independent degrees of freedom of the robot. Thus we can express  $\mathbf{V}_p$  as a function of  $\mathbf{V}_r$ , such that:

$$\mathbf{V}_p = \mathbf{S} \mathbf{V}_r \quad (9)$$

with:

$$\mathbf{V}_r = \mathbf{J}_r \dot{\mathbf{q}}_a \quad (10)$$

and relation (8) will be rewritten as:

$$\mathbf{V}_p = \mathbf{S} \mathbf{J}_r \dot{\mathbf{q}}_a \quad (11)$$

with  $\mathbf{J}_r$  is the (n×n) reduced kinematic Jacobian matrix of the robot. We note that when n = 6, the matrix  $\mathbf{S}$  is the (6×6) identity matrix.

The calculation of  $\mathbf{J}_r$  is obtained by inverting  $\mathbf{J}_r^{-1}$ , which is easy to obtain for most parallel structures [34], in Appendix we present a general method to calculate it using the Jacobian matrices of the legs without taking into account the passive joints connecting the platform to the legs.

For the general case, where the platform has 6 degrees of freedom,  $\mathbf{F}_p$  is calculated by the following Newton-Euler equation [24]:

$$\mathbf{F}_p = \mathbf{I}_s \begin{bmatrix} \dot{\mathbf{v}}_p - \mathbf{g} \\ \dot{\boldsymbol{\omega}}_p \end{bmatrix} + \begin{bmatrix} \boldsymbol{\omega}_p \times (\boldsymbol{\omega}_p \times \mathbf{M} \mathbf{S}_p) \\ \boldsymbol{\omega}_p \times (\mathbf{I}_p \boldsymbol{\omega}_p) \end{bmatrix} \quad (12)$$

where:

$$\mathbf{V}_p = \begin{bmatrix} \mathbf{v}_p^T & \boldsymbol{\omega}_p^T \end{bmatrix}^T$$

$\mathbf{v}_p$  the linear velocity of the origin of frame  $\Sigma_p$ , which is fixed with the platform,

$\boldsymbol{\omega}_p$  the rotational velocity of frame  $\Sigma_p$ ,

$\dot{\mathbf{V}}_p$  the acceleration of the platform, differentiating (9) we obtain:

$$\dot{\mathbf{V}}_p = \dot{\mathbf{S}} \mathbf{V}_r + \mathbf{S} \dot{\mathbf{V}}_r \quad (13)$$

$\mathbf{g}$  acceleration of gravity,

$\mathbf{I}_3$  (3×3) identity matrix,

$M_p$  mass of the platform,

$\mathbf{I}_p$  (3×3) inertia matrix of the platform around the origin of the platform frame  $\Sigma_p$ ,

$\mathbf{M} \mathbf{S}_p$  (3×1) vector of first moments of the platform around the origin of the platform frame  $\Sigma_p$ :

$$\mathbf{M} \mathbf{S}_p = [\mathbf{M} X_p \quad \mathbf{M} Y_p \quad \mathbf{M} Z_p]^T,$$

$\mathbf{I}_s$  (6×6) spatial inertia matrix of the platform:

$$\mathbf{I}_s = \begin{bmatrix} M_p \mathbf{I}_3 & -\hat{\mathbf{M} \mathbf{S}_p} \\ \hat{\mathbf{M} \mathbf{S}_p} & \mathbf{I}_p \end{bmatrix} \quad (14)$$

$\hat{\mathbf{M} \mathbf{S}_p}$  designates the (3×3) skew matrix associated with the vector  $\mathbf{M} \mathbf{S}_p$ ,

$$\hat{\mathbf{M}}\mathbf{S}_p = \begin{bmatrix} 0 & -MZ_p & MY_p \\ MZ_p & 0 & -MX_p \\ -MY_p & MX_p & 0 \end{bmatrix} \quad (15)$$

The calculation of  $\frac{\partial \dot{\mathbf{q}}_i}{\partial \dot{\mathbf{q}}_a}$  is carried out by the following relation which exploits the parallel structure of the robot:

$$\frac{\partial \dot{\mathbf{q}}_i}{\partial \dot{\mathbf{q}}_a} = \frac{\partial \dot{\mathbf{q}}_i}{\partial \mathbf{v}_i} \frac{\partial \mathbf{v}_i}{\partial \mathbf{V}_p} \frac{\partial \mathbf{V}_p}{\partial \mathbf{V}_r} \frac{\partial \mathbf{V}_r}{\partial \dot{\mathbf{q}}_a} \quad (16)$$

with:  $\mathbf{v}_i$  is the vector of Cartesian velocity transferred from leg  $i$  to the platform.

We can rewrite (16) as:

$$\frac{\partial \dot{\mathbf{q}}_i}{\partial \dot{\mathbf{q}}_a} = \mathbf{J}_i^{-1} \mathbf{J}_{vi} \mathbf{S} \mathbf{J}_r \quad (17)$$

with

$\mathbf{J}_i$  is the kinematic Jacobian matrix of leg  $i$  such that:

$$\mathbf{v}_i = \mathbf{J}_i \dot{\mathbf{q}}_i \quad (18)$$

The calculation of  $\mathbf{J}_i$  can be found in many robotics books [7, 2, 39, 24].

$\mathbf{J}_{vi}$  gives the relation between  $\mathbf{v}_i$  and  $\mathbf{V}_p$ :

$$\mathbf{v}_i = \mathbf{J}_{vi} \mathbf{V}_p \quad (19)$$

$\mathbf{J}_{vi}$  permits to transform  $\mathbf{V}_p$  to the attached point between leg  $i$  and the platform and to select the appropriate components representing  $\mathbf{v}_i$ .

Finally, using (7) and (17) the inverse dynamic model of the robot is given by the following compact forms:

$$\begin{aligned} \Gamma &= \mathbf{J}_r^T \mathbf{S}^T \left[ \mathbf{F}_p + \sum_{i=1}^m \mathbf{J}_{vi}^T \mathbf{J}_i^{-T} \mathbf{H}_i \right] \\ \Gamma &= \mathbf{J}_p^T \left[ \mathbf{F}_p + \sum_{i=1}^m \mathbf{J}_{vi}^T \mathbf{J}_i^{-T} \mathbf{H}_i \right] \end{aligned} \quad (20)$$

We note that the term between the brackets in (20) represents the dynamic model of the robot expressed in the Cartesian space of the platform frame  $\Sigma_p$  [25, 30], which can be used for Cartesian space computed torque control.

Many methods can be used to calculate  $\mathbf{H}_i$  representing the inverse dynamic model of a leg  $i$  [18, 32, 7, 21, 37], one can use the method with which he is familiar. To reduce the computational cost the recursive Newton-Euler algorithm using base inertial parameters and customized symbolic methods can be used [26, 21].

**Remark:** From the definition (7) and since the active variables are independent the expression of the inverse dynamic model can be rewritten in the following form:

$$\mathbf{\Gamma} = \mathbf{H}^a + \mathbf{J}_p^T \left( \mathbf{F}_p + \sum_{i=1}^m \mathbf{J}_{v_i}^T \mathbf{J}_i^{-T}(:, \mathbf{p}_i) \mathbf{H}_i^p \right) \quad (21)$$

where  $\mathbf{H}^a$  is the vector of the active forces/torques of all the legs, this corresponds to the components of  $\mathbf{H}_i$  of the motorized joints (for  $i = 1$  with  $m$ ).

$\mathbf{p}_i$  represent the indices of columns corresponding to the passive joints in leg  $i$ ,

and  $\mathbf{H}_i^p$  represent the vector of the passive joints forces/torques of leg  $i$ .

Thus equation (21) can be rewritten on the following form:

$$\mathbf{\Gamma} = \begin{bmatrix} \mathbf{H}_1^a \\ \mathbf{H}_2^a \\ \vdots \\ \mathbf{H}_m^a \end{bmatrix} + \mathbf{J}_p^T \left( \mathbf{F}_p + \begin{bmatrix} \mathbf{J}_{v_1}^T \mathbf{J}_1^{-T}(:, \mathbf{p}_1) & \mathbf{J}_{v_2}^T \mathbf{J}_2^{-T}(:, \mathbf{p}_2) & \cdots & \mathbf{J}_{v_m}^T \mathbf{J}_m^{-T}(:, \mathbf{p}_m) \end{bmatrix} \begin{bmatrix} \mathbf{H}_1^p \\ \mathbf{H}_2^p \\ \vdots \\ \mathbf{H}_m^p \end{bmatrix} \right) \quad (22)$$

This form simplifies considerably the model by avoiding a useless multiplication of the active forces/couples by the projection matrix  $\frac{\partial \dot{\mathbf{q}}_i}{\partial \dot{\mathbf{q}}_a}$ . Besides, the method has the advantage of enabling calculation of the legs kinematics and dynamics by  $m$  processors in parallel.

### 3 Direct dynamic model of parallel robots

The state of parallel robot can be taken as the Cartesian position and velocity of the platform. Thus, the direct dynamic model of the robot gives the platform Cartesian acceleration as a function of the state variables and the input of the motorized joint torques/forces:

$$\dot{\mathbf{V}}_p = \mathbf{f}({}^0\mathbf{T}_p, \mathbf{V}_p, \mathbf{\Gamma}) \quad (23)$$



${}^0\mathbf{T}_p$  the homogeneous transformation matrix of frame  $\Sigma_p$  with respect to the reference frame  $\Sigma_0$ .

In a simulation algorithm  $\dot{\mathbf{V}}_p$  can be integrated to obtain  $\mathbf{V}_p$  and  ${}^0\mathbf{T}_p$  at each sampling time.

We can obtain the direct dynamic model from the inverse dynamic model (20) by substituting the leg joint accelerations  $\ddot{\mathbf{q}}_i$  as a function of  $\dot{\mathbf{V}}_p$  in  $\mathbf{H}_i$  (representing the inverse dynamic model of leg i).

Noting that the inverse dynamic of a leg i can be written as:

$$\mathbf{H}_i(\mathbf{q}_i, \dot{\mathbf{q}}_i, \ddot{\mathbf{q}}_i) = \mathbf{A}_i \ddot{\mathbf{q}}_i + \mathbf{h}_i(\mathbf{q}_i, \dot{\mathbf{q}}_i) \quad (24)$$

with  $\mathbf{A}_i$  is the inertia matrix of leg i, and  $\mathbf{h}_i$  is the vector of the Coriolis, centrifugal and gravity torques or forces of leg i.

We calculate  $\ddot{\mathbf{q}}_i$  from the second order kinematic model of leg I, which is obtained by differentiating equation (19) as:

$$\dot{\mathbf{v}}_i = \mathbf{J}_i \ddot{\mathbf{q}}_i + \dot{\mathbf{J}}_i \dot{\mathbf{q}}_i \quad (25)$$

By differentiating equation (19) with respect to time we obtain:

$$\dot{\mathbf{v}}_i = \dot{\mathbf{J}}_{vi} \mathbf{V}_p + \mathbf{J}_{vi} \dot{\mathbf{V}}_p \quad (26)$$

From (25) and (26) we obtain:

$$\ddot{\mathbf{q}}_i = \mathbf{J}_i^{-1} (\dot{\mathbf{v}}_i - \dot{\mathbf{J}}_i \dot{\mathbf{q}}_i) = \mathbf{J}_i^{-1} (\dot{\mathbf{J}}_{vi} \mathbf{V}_p + \mathbf{J}_{vi} \dot{\mathbf{V}}_p - \dot{\mathbf{J}}_i \dot{\mathbf{q}}_i) \quad (27)$$

Substituting (12), (24), (27), (9) and (13) in (20), we obtain:

$$\dot{\mathbf{V}}_r = \mathbf{A}_{robot}^{-1} (\mathbf{J}_r^T \Gamma - \mathbf{h}_{robot}) \quad (28)$$

with:

$$\mathbf{A}_{robot} = \mathbf{S}^T \left( \mathbf{I}_S + \sum_{i=1}^m \mathbf{J}_{vi}^T \mathbf{A}_{Xi} \mathbf{J}_{vi} \right) \mathbf{S} \quad (29)$$

$$\begin{aligned} \mathbf{h}_{robot} = & \mathbf{S}^T \left( \left( \mathbf{I}_S + \sum_{i=1}^m \mathbf{J}_{vi}^T \mathbf{A}_{Xi} \mathbf{J}_{vi} \right) \dot{\mathbf{S}} \mathbf{V}_r + \begin{bmatrix} \boldsymbol{\omega}_p \times (\boldsymbol{\omega}_p \times \mathbf{M} \mathbf{S}_p) \\ \boldsymbol{\omega}_p \times (\mathbf{I}_p \boldsymbol{\omega}_p) \end{bmatrix} - \begin{bmatrix} \mathbf{M}_p \mathbf{I}_{d3} \\ \mathbf{M} \mathbf{S}_p \end{bmatrix} \mathbf{g} \right) \\ & + \mathbf{S}^T \sum_{i=1}^m \left\{ \mathbf{J}_{vi}^T \left( \mathbf{A}_{Xi} (\dot{\mathbf{J}}_{vi} \mathbf{S} \mathbf{V}_r - \dot{\mathbf{J}}_i \dot{\mathbf{q}}_i) + \mathbf{h}_{Xi} \right) \right\} \end{aligned} \quad (30)$$

where:

$\mathbf{A}_{\text{robot}}$  is the total inertia matrix of the robot with respect to the platform Cartesian space.

$\mathbf{A}_{\text{xi}}$  is the inertia matrix of leg  $i$  referred to the Cartesian space of the terminal point (or frame) of leg  $i$ , it is equal to  $\mathbf{A}_{\text{xi}} = \mathbf{J}_i^T \mathbf{A}_i \mathbf{J}_i^{-1}$

and  $\mathbf{h}_{\text{xi}}$  is the Coriolis, centrifuge and gravity forces referred to the terminal Cartesian space of leg  $i$ , it is given as  $\mathbf{h}_{\text{xi}} = \mathbf{J}_i^T \mathbf{h}_i$

Many methods can be used to calculate these terms:

$\mathbf{h}_i$  can be computed using the Newton-Euler inverse dynamic model of leg  $i$  by setting  $\ddot{\mathbf{q}} = 0$ ,  $\mathbf{h}_i = \mathbf{H}_i(\mathbf{q}_i, \dot{\mathbf{q}}_i, \ddot{\mathbf{q}}_i = 0)$  [42, 22], for  $\mathbf{A}_i$  one can use the method with which he is familiar. To reduce the computational cost the recursive Newton-Euler algorithm using base inertial parameters and customized symbolic methods can be used to obtain  $\mathbf{h}_i$  and  $\mathbf{A}_i$  [21].

## 4 Dynamic models of the Space robot

### 4.1 DESCRIPTION OF THE ROBOT

The 6 d.o.f. (degrees of freedom) Space robot [5] is composed of a moving platform connected to a fixed base by three (U-P-S) extendable legs. The extremities of each leg are fitted with a 2 d.o.f. universal joint (U) at the base (the first one is actuated) and a 3 d.o.f. spherical joint (S) at the platform. The lengths of the legs are actuated using prismatic joints (P). We note that the legs of this robot have the same structure as the classical Gough-Stewart parallel robot [34].

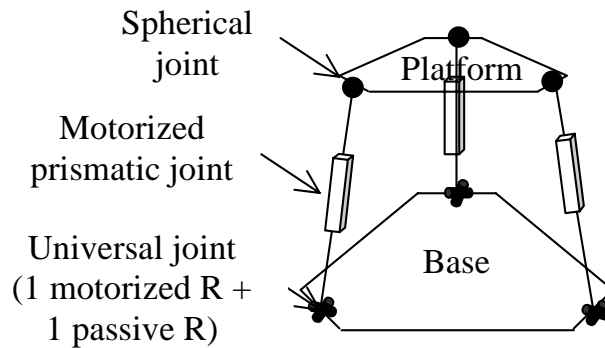


Figure 2. Space robot

Assuming that  $B_i$  is the point connecting leg  $i$  to the base and  $P_i$  is the point connecting leg  $i$  to the platform. The frame  $\Sigma_0$  is defined fixed with the base, its origin is  $B_1$ , and frame  $\Sigma_p$  is fixed with the mobile platform with  $P_1$  as origin. We place these frames as shown in Figure 3:

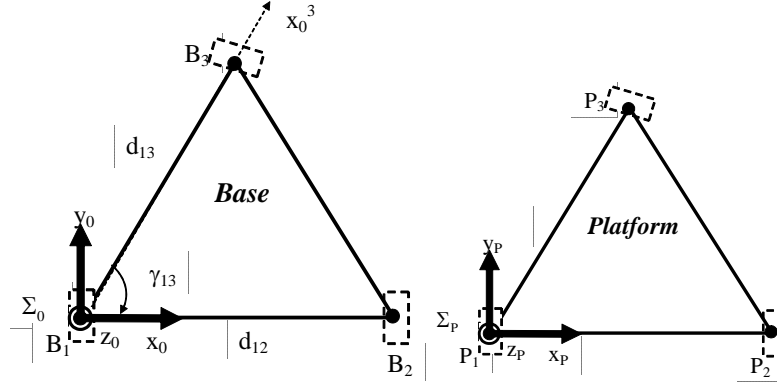


Figure 3. Base frame  $\Sigma_0$  and Platform frame  $\Sigma_p$

The notations of Khalil and Kleinfinger [20], are used to describe the geometry of the tree structure composed of the base and the legs. The definition of the local link frames of leg  $i$  are given in Figure 4, while the geometric parameters are given in Table 1. The parameters of Table 1 are:

$a(j)$  denotes the frame antecedent to frame  $j$ ,

$\mu(j)$  and  $\sigma(j)$  describe the type of joint:

- $\mu(j) = 1$  if joint  $j$  is active and  $\mu(j) = 0$  if it is passive,
- $\sigma(j) = 1$  if joint  $j$  is prismatic and  $\sigma(j) = 0$  if it is revolute,

The parameters  $(\gamma_j, b_j, \alpha_j, d_j, \theta_j, r_j)$  are used to determine the location of frame  $\Sigma_j$  with respect to its antecedent  $F_i$ .

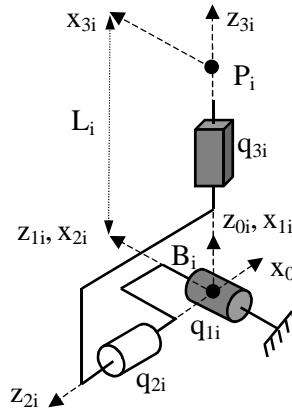


Figure 4. Link frames of one leg

| $j_i$ | $a(j_i)$ | $\mu_{ji}$ | $\sigma_{ji}$ | $\gamma_{ji}$ | $b_{ji}$ | $\alpha_{ji}$ | $d_{ji}$ | $\theta_{ji}$ | $r_{ji}$ |
|-------|----------|------------|---------------|---------------|----------|---------------|----------|---------------|----------|
| 1i    | 0        | 1          | 0             | $\gamma_{1i}$ | 0        | $-\pi/2$      | $d_{1i}$ | $\theta_{1i}$ | 0        |
| 2i    | 1i       | 0          | 0             | 0             | 0        | $\pi/2$       | 0        | $\theta_{2i}$ | 0        |
| 3i    | 2i       | 1          | 1             | 0             | 0        | $\pi/2$       | 0        | 0             | $L_i$    |

for  $i = 1, \dots, 3$  and  $\gamma_{11} = \gamma_{12} = d_{11} = 0$

Table 1: Geometric parameters of the leg  $i$  frames

#### 4.2 CALCULATION OF THE JACOBIAN MATRICES

The following notations are used:

$\mathbf{P}_1\mathbf{P}_i$  position vector between points  $P_1$  and  $P_i$ ,

$\mathbf{v}_i$  linear velocity of point  $\mathbf{P}_i$ , it is calculated in terms of the platform velocity as follows:

$$\mathbf{v}_i = \mathbf{v}_p + \boldsymbol{\omega}_p \times \mathbf{P}_1\mathbf{P}_i \quad (31)$$

$\mathbf{q}_a$  vector of the active joint variables, (first and third variables of each leg)

$$\mathbf{q}_a = [q_{11} \quad q_{31} \quad q_{12} \quad q_{32} \quad q_{13} \quad q_{33}]^T \quad (32)$$

with:

$q_{ji}$  denotes the position of joint  $j$  of leg  $i$ ,

$\mathbf{q}_i$  vector of the joint position of leg  $i$ , it does not take into account the variables of the spherical joint between the leg and the platform:

$$\mathbf{q}_i = [q_{1i} \quad q_{2i} \quad q_{3i}]^T \quad (33)$$

The Jacobian matrices required to calculate the dynamic model are calculated as follows:

i) The inverse kinematic model of a leg, which gives the joint velocities ( $\dot{q}_{1i}, \dot{q}_{2i}, \dot{q}_{3i}$ ,  $i = 1$  to  $3$ ) as a function of the linear velocity of point  $P_i$  connecting leg  $i$  with the platform. It corresponds to the inverse kinematic model of a serial structure with three joints (RRP):

$$\dot{\mathbf{q}}_i = \mathbf{J}_i^{-1} \mathbf{v}_i \quad (34)$$

$\mathbf{J}_i^{-1}$  is the  $(3 \times 3)$  inverse Jacobian matrix of leg  $i$ . It is calculated by inverting  $\mathbf{J}_i$ .

The Jacobian matrix of leg  $i$ , is calculated as follows [7, 2, 24]:

$$\mathbf{J}_i = [\mathbf{a}_{1i} \times \mathbf{B}_i\mathbf{P}_i \quad \mathbf{a}_{2i} \times \mathbf{B}_i\mathbf{P}_i \quad \mathbf{a}_{3i}] \quad (35)$$

with:

$\mathbf{a}_{ji}$  the unit vector along the joint axis  $j$  of leg  $i$ ,

$\mathbf{B}_i\mathbf{P}_i$  position vector from  $B_i$  to  $P_i$ ,  $\mathbf{B}_i\mathbf{P}_i = L_i \cdot \mathbf{a}_{3i}$ , where  $L_i$  the length of leg  $i$ , see *Figure 4*.

Substituting  $\mathbf{a}_{1i}, \mathbf{a}_{2i}, \mathbf{a}_{3i}$  and  $\mathbf{B}_i \mathbf{P}_i$  in terms of the geometric parameters of the legs from Table 1 we obtain  $\mathbf{J}_i$ . Inverting analytically  $\mathbf{J}_i$  we compute the matrix  $\mathbf{J}_i^{-1}$ . As an example, the matrix  $\mathbf{J}_1^{-1}$  with respect to frame  $\Sigma_0$  is given by:

$${}^0\mathbf{J}_1^{-1} = \begin{bmatrix} -S q_{1i}/L_i & S q_{2i} & 0 & -C q_{1i}/L_i & S q_{2i} \\ C q_{1i} C q_{2i}/L_i & S q_{2i}/L_i & -C q_{2i} S q_{1i}/L_i \\ C q_{1i} S q_{2i} & -C q_{2i} & -S q_{1i} S q_{2i} \end{bmatrix} \quad (36)$$

$C^*$  and  $S^*$  represent respectively  $\cos(*)$  and  $\sin(*)$ .

We note that the 3<sup>d</sup> row elements of  ${}^0\mathbf{J}_i^{-1}$  represent the components of the unit vector along the prismatic joint axis  ${}^0\mathbf{a}_{3i}$ .

The singular configurations of equation (36) occur when  $L_i = 0$  and  $\sin(q_{2i}) = 0$ , which are outside the operating space of the robot.

## ii) Calculation of $\mathbf{J}_{vi}$

$\mathbf{v}_i$  is the linear Cartesian velocity of point  $\mathbf{P}_i$ , it is given by (31), which can be rewritten as:

$$\mathbf{v}_i = \begin{bmatrix} \mathbf{I}_3 & -\hat{\mathbf{P}_i \mathbf{P}_i} \end{bmatrix} \mathbf{V}_p \quad (37)$$

So:

$$\mathbf{J}_{vi} = \frac{\partial \mathbf{v}_i}{\partial \mathbf{V}_p} = \begin{bmatrix} \mathbf{I}_3 & -\hat{\mathbf{P}_i \mathbf{P}_i} \end{bmatrix} \quad (38)$$

where  $\hat{\mathbf{P}_i \mathbf{P}_i}$  designates the (3×3) skew matrix associated with the vector  $\mathbf{P}_i \mathbf{P}_i$ , see (15).

iii) The inverse kinematic model of the robot, which gives the active joint velocities ( $\dot{q}_{1i}, \dot{q}_{3i}$ ,  $i = 1$  and 3) as a function of the spatial velocity of the platform.

$$\dot{\mathbf{q}}_a = \mathbf{J}_r^{-1} \mathbf{V}_p \quad (39)$$

with:  $\mathbf{J}_r^{-1}$  is the inverse Jacobian matrix of the Space robot, which is computed as given in Appendix, where:

$$\dot{q}_{ji} = \begin{bmatrix} \mathbf{J}_i^{-1}(j,:) & -\mathbf{J}_i^{-1}(j,:) \hat{\mathbf{P}_i \mathbf{P}_i} \end{bmatrix} \mathbf{V}_p \quad (40)$$

where:  $\mathbf{J}_i^{-1}(j,:)$  is the  $j^{\text{th}}$  row of the matrix  $\mathbf{J}_i^{-1}$ .

Finally, since  $\hat{\mathbf{P}}_1 \mathbf{P}_1 = \mathbf{0}$ :

$$\mathbf{J}_r^{-1} = \begin{bmatrix} \mathbf{J}_1^{-1}(1,:) & \mathbf{0} \\ \mathbf{J}_1^{-1}(3,:) & \mathbf{0} \\ \mathbf{J}_2^{-1}(1,:) & -\mathbf{J}_2^{-1}(1,:) \hat{\mathbf{P}}_1 \mathbf{P}_2 \\ \mathbf{J}_2^{-1}(3,:) & -\mathbf{J}_2^{-1}(3,:) \hat{\mathbf{P}}_1 \mathbf{P}_2 \\ \mathbf{J}_3^{-1}(1,:) & -\mathbf{J}_3^{-1}(1,:) \hat{\mathbf{P}}_1 \mathbf{P}_3 \\ \mathbf{J}_3^{-1}(3,:) & -\mathbf{J}_3^{-1}(3,:) \hat{\mathbf{P}}_1 \mathbf{P}_3 \end{bmatrix} \quad (41)$$

For this robot  $\mathbf{S} = \mathbf{I}_6$ , where  $\mathbf{I}_6$  is the (6×6) identity matrix.

#### 4.3 INVERSE DYNAMIC MODEL OF THE SPACE ROBOT

The Newton-Euler equation of the mobile platform is given by relation (12). Thus, the dynamic model is obtained by applying (20) as:

$$\mathbf{\Gamma} = \mathbf{J}_r^T \left( \mathbf{F}_p + \sum_{i=1}^3 \left[ \begin{bmatrix} \mathbf{I}_3 \\ \mathbf{P}_1 \hat{\mathbf{P}}_i \end{bmatrix} \mathbf{J}_i^{-1} \mathbf{H}_i(\mathbf{q}_i, \dot{\mathbf{q}}_i, \ddot{\mathbf{q}}_i) \right] \right) \quad (42)$$

Using the form of relation (22) and noting that:

$$\mathbf{H}_i^p = \mathbf{H}_{2i} \quad (43)$$

$$\mathbf{H}_i^a = \begin{bmatrix} \mathbf{H}_{1i} & \mathbf{H}_{3i} \end{bmatrix}^T \quad (44)$$

we obtain:

$$\mathbf{\Gamma} = \begin{bmatrix} \mathbf{H}_{11} \\ \mathbf{H}_{31} \\ \mathbf{H}_{12} \\ \mathbf{H}_{32} \\ \mathbf{H}_{13} \\ \mathbf{H}_{33} \end{bmatrix} + \mathbf{J}_p^T \left( \mathbf{F}_p + \begin{bmatrix} \mathbf{J}_{v1}^T \mathbf{J}_1^{-T}(:,2) & \mathbf{J}_{v2}^T \mathbf{J}_2^{-T}(:,2) & \mathbf{J}_{v3}^T \mathbf{J}_3^{-T}(:,2) \end{bmatrix} \begin{bmatrix} \mathbf{H}_{21} \\ \mathbf{H}_{22} \\ \mathbf{H}_{23} \end{bmatrix} \right) \quad (45)$$

#### 4.4 DIRECT DYNAMIC MODEL OF THE SPACE ROBOT

The direct dynamic model is obtained by applying the procedure given in section 3, we obtain:

$$\dot{\mathbf{V}}_p = \mathbf{A}_{robot}^{-1} (\mathbf{J}_r^{-T} \mathbf{\Gamma} - \mathbf{h}_{robot}) \quad (46)$$

where  $\mathbf{A}_{\text{robot}}$  is obtained using (29) as:

$$\mathbf{A}_{\text{robot}} = \begin{bmatrix} \mathbf{M}_p \mathbf{I}_3 & -\mathbf{M} \hat{\mathbf{S}}_p \\ \hat{\mathbf{M}} \mathbf{S}_p & \mathbf{I}_p \end{bmatrix} + \sum_{i=1}^3 \begin{bmatrix} \mathbf{I}_3 & \\ & \hat{\mathbf{P}}_i \mathbf{P}_i \end{bmatrix} \mathbf{A}_{\text{Xi}} \begin{bmatrix} \mathbf{I}_3 & -\hat{\mathbf{P}}_i \mathbf{P}_i \end{bmatrix} \quad (47)$$

and  $\mathbf{h}_{\text{robot}}$  is given by (30):

We start by calculating the term  $\dot{\mathbf{J}}_{\text{vi}} \mathbf{V}_p$ , by differentiating equation (31) we obtain:

$$\dot{\mathbf{v}}_i = \dot{\mathbf{v}}_p + \dot{\boldsymbol{\omega}}_p \times \mathbf{P}_i \mathbf{P}_i + \boldsymbol{\omega}_p \times (\boldsymbol{\omega}_p \times \mathbf{P}_i \mathbf{P}_i) \quad (48)$$

Which can be rewritten as:

$$\dot{\mathbf{v}}_i = \begin{bmatrix} \mathbf{I}_3 & \hat{\mathbf{P}}_i \mathbf{P}_i \end{bmatrix} \dot{\mathbf{V}}_p + \boldsymbol{\omega}_p \times (\boldsymbol{\omega}_p \times \mathbf{P}_i \mathbf{P}_i) \quad (49)$$

Using (26), we deduce:

$$\mathbf{J}_{\text{vi}} \mathbf{V}_p = \boldsymbol{\omega}_p \times (\boldsymbol{\omega}_p \times \mathbf{P}_i \mathbf{P}_i) \quad (50)$$

Finally:

$$\mathbf{h}_{\text{robot}} = \begin{bmatrix} \boldsymbol{\omega}_p \times (\boldsymbol{\omega}_p \times \mathbf{M} \mathbf{S}_p) \\ \boldsymbol{\omega}_p \times (\mathbf{I}_p \boldsymbol{\omega}_p) \end{bmatrix} - \begin{bmatrix} \mathbf{M}_p \mathbf{I}_3 \\ \hat{\mathbf{M}} \mathbf{S}_p \end{bmatrix} \mathbf{g} + \sum_{i=1}^3 \left\{ \begin{bmatrix} \mathbf{I}_3 \\ \hat{\mathbf{P}}_i \mathbf{P}_i \end{bmatrix} \left( \mathbf{A}_{\text{Xi}} (\boldsymbol{\omega}_p \times (\boldsymbol{\omega}_p \times \mathbf{P}_i \mathbf{P}_i) - \dot{\mathbf{J}}_i \dot{\mathbf{q}}_i) + \mathbf{h}_{\text{Xi}} \right) \right\} \quad (51)$$

## 5 Dynamic modeling of the C5 robot

### 5.1 DESCRIPTION OF THE C5 ROBOT [8]

The C5 parallel robot consists of a base and a platform linked together by six linear actuators (*Figure 5*) [8]. The platform is designed by a cube. Each leg is embedded in the base at point  $\mathbf{B}_i$  and connected to one face of the mobile platform through a C5 joint at point  $\mathbf{P}_i$ . The base frame,  $\Sigma_0$ , is located such that leg 1 and 2 are parallel to the x axis, legs 3 and 4 are parallel to the y axis and legs 4 and 5 are parallel to the z axis. The platform frame,  $\Sigma_p$ , is located at the front right top corner (*Figure 5*), the frame axes are defined as in *Figure 5*. The C5 joint is a complex joint with 3 rotational and 2 translational degrees of freedom, it is constructed by a spherical joint tied to two cross sliding plates (point  $\mathbf{P}_i$  is located at the centre of the spherical-joint of leg i).

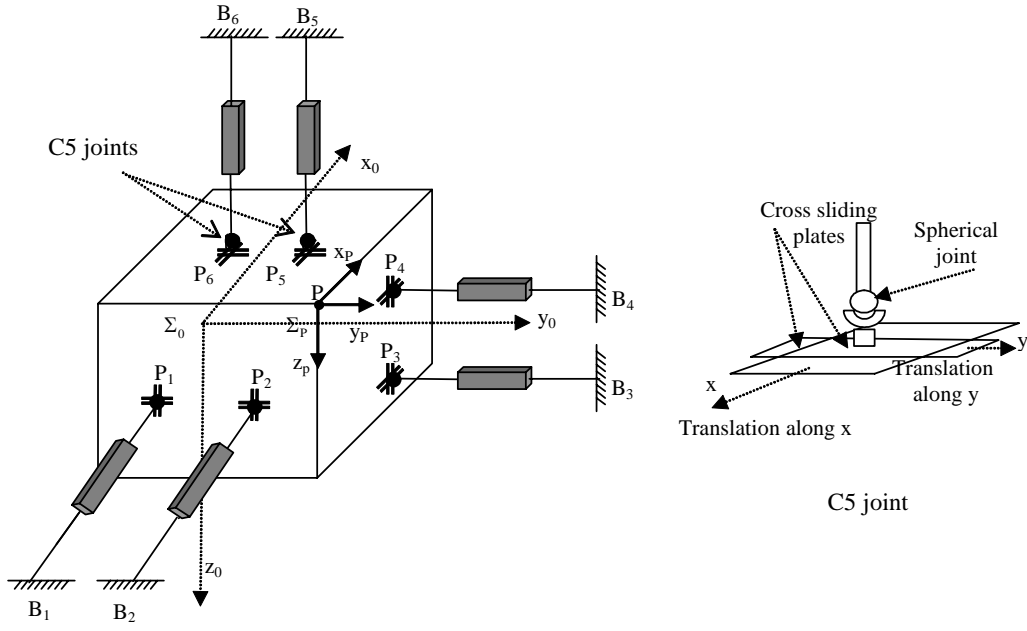


Figure 5. The C5 Robot (position of the frames and description of a C5 joint)

## 5.2 CALCULATION OF THE JACOBIAN MATRICES

The Cartesian variable of leg  $i$  transmitted to the platform is the displacement along the normal to the planar face of the platform on which the leg is connected, thus the kinematic model of leg  $i$  is given by the following scalar equation:

$$v_i = \mathbf{r}_i^T \mathbf{a}_i \dot{q}_i \quad (52)$$

$v_i$  is the terminal velocity of leg  $i$ ,

$\mathbf{a}_i$  denotes the  $(3 \times 1)$  unit vector along the leg  $i$  axis, such that:

$$\mathbf{a}_1 = \mathbf{a}_2 = [1 \ 0 \ 0]^T, \mathbf{a}_3 = \mathbf{a}_4 = [0 \ 1 \ 0]^T, \mathbf{a}_5 = \mathbf{a}_6 = [0 \ 0 \ 1]^T$$

$\mathbf{r}_i$  is the  $(3 \times 1)$  unit vector normal to the planar face of the platform on which the leg is connected, such that:

$$\mathbf{r}_1 = \mathbf{r}_2 = \mathbf{s}, \mathbf{r}_3 = \mathbf{r}_4 = \mathbf{n}, \mathbf{r}_5 = \mathbf{r}_6 = \mathbf{a}$$

where  $(\mathbf{s}, \mathbf{n}, \mathbf{a})$  represent the columns of the orientation matrix of the platform frame with respect to the base frame:

$${}^0\mathbf{R}_p = [\mathbf{s} \ \mathbf{n} \ \mathbf{a}] = \begin{bmatrix} s_x & n_x & a_x \\ s_y & n_y & a_y \\ s_z & n_z & a_z \end{bmatrix} \quad (53)$$



Thus, the Jacobian of leg  $i$  is given by the scalar:

$$J_i = \mathbf{r}_i^T \mathbf{a}_i \quad (54)$$

The velocity of point  $P_i$  denoted  $\mathbf{v}_i$  as a function of  $\mathbf{V}_p$  is given as follows:

$$\mathbf{v}_i = \mathbf{r}_i^T (\mathbf{v}_p + \boldsymbol{\omega}_p \times \mathbf{PP}_i) \quad (55)$$

Thus  $\mathbf{J}_{vi}$  the Jacobian matrix which transforms the platform velocity to point  $P_i$ , and project it on the  $\mathbf{r}_i$  axis, is obtained as:

$$\mathbf{J}_{vi} = \begin{bmatrix} \mathbf{r}_i^T & -\mathbf{r}_i^T \hat{\mathbf{PP}}_i \end{bmatrix} \quad (56)$$

The inverse kinematic model of the robot gives the active joint velocities as a function of the velocity of the platform, where:

$$\dot{\mathbf{q}}_a = [\dot{q}_1 \quad \dot{q}_2 \quad \dot{q}_3 \quad \dot{q}_4 \quad \dot{q}_5 \quad \dot{q}_6]^T \quad (57)$$

using (52) and (55) we can write (see Appendix):

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \\ \dot{q}_5 \\ \dot{q}_6 \end{bmatrix} = \begin{bmatrix} \frac{\mathbf{s}^T}{s_x} & -\frac{\mathbf{s}^T \hat{\mathbf{PP}}_1}{s_x} \\ \frac{\mathbf{s}^T}{s_x} & -\frac{\mathbf{s}^T \hat{\mathbf{PP}}_2}{s_x} \\ \frac{\mathbf{n}^T}{n_y} & -\frac{\mathbf{n}^T \hat{\mathbf{PP}}_3}{n_y} \\ \frac{\mathbf{n}^T}{n_y} & -\frac{\mathbf{n}^T \hat{\mathbf{PP}}_4}{n_y} \\ \frac{\mathbf{a}^T}{a_z} & -\frac{\mathbf{a}^T \hat{\mathbf{PP}}_5}{a_z} \\ \frac{\mathbf{a}^T}{a_z} & -\frac{\mathbf{a}^T \hat{\mathbf{PP}}_6}{a_z} \end{bmatrix} \begin{bmatrix} \mathbf{v}_p \\ \boldsymbol{\omega}_p \end{bmatrix} \quad (58)$$

The singular configurations of equation (58), correspond to  $s_x = 0$ ,  $n_y = 0$  and  $a_z = 0$ , but the robot cannot reach these configurations owing to mechanical limits.

For this robot  $\mathbf{S} = \mathbf{I}_6$ , where  $\mathbf{I}_6$  is the (6×6) identity matrix.

### 5.3 INVERSE DYNAMIC MODEL

The dynamic model of each leg  $i$  is given by the scalar equation:

$$H_i = (M_i + I_{a_i}) \ddot{q}_i + h_i \quad (59)$$

with  $M_i$  is the mass of the moving part of leg  $i$ , and  $I_{a_i}$  is the inertia of actuator  $i$ .

Assuming the gravity is equal to  $g$  and directed along  $-z_0$ :

$$h_i = 0 \text{ for } i = 1, \dots, 4 \quad (60)$$

$$h_i = M_i g \text{ for } i = 5, 6 \quad (61)$$

The Newton-Euler equation of the platform is the same as (12).

Since the dynamic model of each leg is only a function of the motorized joints, thus using (22), and noting that  $\mathbf{H}_i^a = H_i$  the inverse dynamic model is obtained as:

$$\mathbf{\Gamma} = \begin{bmatrix} H_1 \\ H_2 \\ H_3 \\ H_4 \\ H_5 \\ H_6 \end{bmatrix} + \mathbf{J}_p^T \mathbf{F}_p \quad (62)$$

### 5.4 DIRECT DYNAMIC MODEL

To obtain the direct dynamic model we use the form (20) for the inverse dynamic model, the direct dynamic model is given as:

$$\dot{\mathbf{V}}_p = \mathbf{A}_{robot}^{-1} \left( \mathbf{J}_r^T \mathbf{\Gamma} - \mathbf{h}_{robot} \right) \quad (63)$$

Using (29) and (30) to determine  $\mathbf{A}_{robot}$  and  $\mathbf{h}_{robot}$ , for that we should calculate the term  $\mathbf{J}_{vi} \mathbf{V}_p$ :

By differentiating equation (55), we obtain:

$$\dot{\mathbf{v}}_i = \mathbf{r}_i^T \left( \dot{\mathbf{v}}_p + \dot{\boldsymbol{\omega}}_p \times \mathbf{P} \mathbf{P}_i - \boldsymbol{\omega}_p \times \mathbf{v}_p \right) \quad (64)$$

Using (26), we obtain:

$$\mathbf{J}_{vi} \mathbf{V}_p = -\mathbf{r}_i^T \left( \boldsymbol{\omega}_p \times \mathbf{v}_p \right) \quad (65)$$

Finally using the general relations (29) and (30):

$$\mathbf{A}_{\text{robot}} = \begin{bmatrix} \mathbf{M}_p \mathbf{I}_3 & -\hat{\mathbf{M}} \mathbf{S}_p \\ \hat{\mathbf{M}} \mathbf{S}_p & \mathbf{I}_p \end{bmatrix} + \sum_{i=1}^6 \begin{bmatrix} \mathbf{r}_i \\ \hat{\mathbf{P}} \mathbf{P}_i \mathbf{r}_i \end{bmatrix} \mathbf{A}_{\text{Xi}} \begin{bmatrix} \mathbf{r}_i^T & -\mathbf{r}_i^T \hat{\mathbf{P}} \mathbf{P}_i \end{bmatrix} \quad (66)$$

$$\mathbf{h}_{\text{robot}} = \begin{bmatrix} \boldsymbol{\omega}_p \times (\boldsymbol{\omega}_p \times \mathbf{M} \mathbf{S}_p) \\ \boldsymbol{\omega}_p \times (\mathbf{I}_p \boldsymbol{\omega}_p) \end{bmatrix} - \begin{bmatrix} \mathbf{M}_p \mathbf{I}_3 \\ \hat{\mathbf{M}} \mathbf{S}_p \end{bmatrix} \mathbf{g} + \sum_{i=1}^6 \begin{bmatrix} \mathbf{r}_i \\ \hat{\mathbf{P}} \mathbf{P}_i \mathbf{r}_i \end{bmatrix} \left( \mathbf{A}_{\text{Xi}} \left( -\mathbf{r}_i^T \boldsymbol{\omega}_p \times (\mathbf{v}_p + \mathbf{a}_i \dot{q}_i) \right) + h_{\text{Xi}} \right) \quad (67)$$

where  $\mathbf{A}_{\text{Xi}}$  and  $h_{\text{Xi}}$  are given by the following scalar equations:

$$\mathbf{A}_{\text{Xi}} = \mathbf{J}_i^{-T} (\mathbf{M}_i + \mathbf{I} \mathbf{a}_i) \mathbf{J}_i^{-1} \quad (68)$$

$$h_{\text{Xi}} = \mathbf{J}_i^{-T} h_i \quad (69)$$

Since  $\mathbf{J}_i$  is a scalar, thus  $\mathbf{J}_i^{-T} = \mathbf{J}_i^{-1}$  and relations (68) and (69) are written as:

$$\mathbf{A}_{\text{Xi}} = \frac{(\mathbf{M}_i + \mathbf{I} \mathbf{a}_i)}{\mathbf{J}_i^2} \quad (70)$$

$$h_{\text{Xi}} = \frac{h_i}{\mathbf{J}_i} \quad (71)$$

## 6 Conclusion

In this paper, a general method to compute the inverse and direct dynamic models of parallel robot is presented. The models are expressed in a closed form that exploits the structural characteristics of parallel robots. The models are calculated in terms of the dynamic model of the legs and the dynamics of the platform and some Jacobian matrices. Besides, a simple and general method for the calculation of the inverse kinematic Jacobian matrix of parallel robots is proposed. The computation of all the elements needed to calculate these models can be obtained using classical and known techniques, which are developed for serial robots. The method is applied in this paper on two robots with different structures and could be applied for most parallel robots.

## Acknowledgment

This work has been carried out through the French national project MP2 of ROBEA and the European IP project n° 011815 (NEXT- Next generation production systems).

## Appendix

The calculation of the inverse and direct dynamic models requires the computation of the inverse Jacobian matrices of the legs and the kinematic Jacobian matrix of the robot.

Here we propose a simple method for the calculation of the inverse kinematic Jacobian matrix of the robot using the inverse Jacobian matrices of the legs.

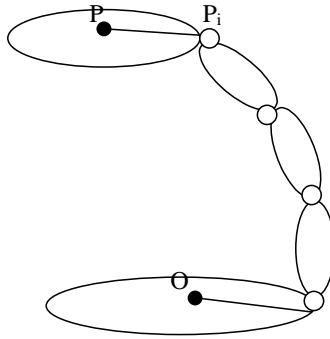
The kinematic Jacobian matrix of the robot represents the relation between the active joint velocities  $\dot{\mathbf{q}}_a$  and the spatial velocity of the platform  $\mathbf{V}_r$  :

$$\mathbf{V}_r = \mathbf{J}_r \dot{\mathbf{q}}_a \quad (72)$$

In general,  $\mathbf{J}_r$  is calculated by inverting the inverse Jacobian matrix  $\mathbf{J}_r^{-1}$  which is obtained more easily.

To calculate  $\mathbf{J}_r^{-1}$ , the first solution is to make use of the relation between joint velocities and the platform velocity through each leg  $i$  from the fixed base frame to the platform frame (*Figure 6*) [2]:

$$\mathbf{V}_r = \mathbf{J}_{ci} \dot{\mathbf{q}}_{ci} \quad (73)$$



*Figure 6.* Path from the base to the platform through leg  $i$

with:

$\dot{\mathbf{q}}_{ci}$  is the vector of joint velocities of leg  $i$  including the passive joints between the platform and the terminal link of leg  $i$ .

$\mathbf{J}_{ci}$  is the kinematic Jacobian matrix of the serial path from the fixed base to the platform frame through leg  $i$ .

Solving (73) for the active velocity joints of leg  $i$ , such that:

$$\dot{\mathbf{q}}_{ai} = \mathbf{L}_{ci}(\mathbf{q}_{ci}) \mathbf{V}_r \quad (74)$$

The inverse kinematic Jacobian matrix of the robot is obtained by grouping  $\mathbf{L}_{ci}$  of all the legs:

$$\mathbf{J}_r^{-1} = \begin{bmatrix} \mathbf{L}_{c1} \\ \vdots \\ \mathbf{L}_{cm} \end{bmatrix} \quad (75)$$

Relation (73) makes use of the passive joints between leg  $i$  and the platform. These variables can be avoided to have a simpler solution by using the relation between the velocity of the attached points between the platform and the legs  $\mathbf{V}_i$  in terms of  $\mathbf{V}_r$  from one side and in terms of  $\dot{\mathbf{q}}_i$  from the other side:

$$\mathbf{v}_i = \mathbf{J}_{vi} \mathbf{S} \mathbf{V}_r \quad (76)$$

and:

$$\mathbf{v}_i = \mathbf{J}_i \dot{\mathbf{q}}_i \quad (77)$$

$\dot{\mathbf{q}}_i$  is the vector of joint velocities of the leg  $i$  without considering the passive joints connecting the platform to the leg.

$\mathbf{J}_i$  is the kinematic Jacobian matrix of leg  $i$  from the fixed base to the attached point of leg  $i$ .

$\mathbf{J}_{vi}$  represents the Jacobian matrix that permits to transform the platform velocity to the attached point of leg  $i$  and to select the proper components.

Solving (77) for the active joints of leg  $i$ , gives:

$$\dot{\mathbf{q}}_{ai} = \mathbf{L}_i(\mathbf{q}_i) \mathbf{v}_i \quad (78)$$

Thus using (76), and regrouping for all the legs, we deduce:

$$\dot{\mathbf{q}}_a = \begin{bmatrix} \mathbf{L}_1 \mathbf{J}_{v1} \\ \vdots \\ \mathbf{L}_m \mathbf{J}_{vm} \end{bmatrix} \mathbf{S} \mathbf{V}_r \quad (79)$$

thus:

$$\mathbf{J}_r^{-1} = \begin{bmatrix} \mathbf{L}_1 \mathbf{J}_{v1} \\ \vdots \\ \mathbf{L}_m \mathbf{J}_{vm} \end{bmatrix} \mathbf{S} \quad (80)$$

As examples:

- For Space robot:

$$\mathbf{J}_{vi} = \begin{bmatrix} \mathbf{I}_3 & -\hat{\mathbf{P}}\mathbf{P}_i \end{bmatrix}, \mathbf{L}_i = \begin{bmatrix} \mathbf{J}_i^{-1}(1,:) \\ \mathbf{J}_i^{-1}(3,:) \end{bmatrix}, \text{ with } \mathbf{J}_i \text{ is a } (3 \times 3) \text{ matrix, and } \mathbf{S} = \mathbf{I}_6,$$

- For C5 robot  $\mathbf{J}_{vi} = \begin{bmatrix} \mathbf{r}_i^T & -\mathbf{r}_i^T \hat{\mathbf{P}}\mathbf{P}_i \end{bmatrix}$ , and  $\mathbf{L}_i = \mathbf{J}_i^{-1}$ , with  $\mathbf{J}_i$  is a scalar (54).

- For Gough-Stewart robot, which has six legs as those of the Space robot, but only the third joint is active:

$\mathbf{J}_{vi} = \begin{bmatrix} \mathbf{I}_3 & -\hat{\mathbf{P}}\mathbf{P}_i \end{bmatrix}$  and  $\mathbf{L}_i = \mathbf{J}_i^{-1}(3,:)$ , with  $\mathbf{J}_i$  is a  $(3 \times 3)$  matrix, which is similar to that of Space robot.

## References

1. Ait-Ahmed, M.: Contribution à la modélisation géométrique et dynamique des robots parallèles. Ph.D. thesis, LAAS, Toulouse (1993).
2. Angeles J.: Fundamentals of Robotic Mechanical Systems, Springer-Verlag, New York (2002).
3. Bhattacharya S., Hatwal H., and Ghosh A.: An on-line estimation scheme for generalized Stewart platform type parallel manipulators, J. Mechanism and Machine Theory **32**(1), 79-89 (January 1997).
4. Bhattacharya S., Nenchev D.N., and Uchiyama M.: A recursive formula for the inverse of the inertia matrix of a parallel manipulator, J. Mechanism and Machine Theory **33**(7), 957-964 (October 1998).
5. Beji L.: Modélisation, identification et commande d'un robot parallèle, Ph.D. thesis, université d'Evry Val d'Essonne (1997).
6. Codourey A., and Burdet E.: A body oriented method for finding a linear form of the dynamic equations of fully parallel robot, IEEE Conf. on Robotics and Automation, pp.1612-1619, Albuquerque, New Mexico, U.S., 1997.
7. Craig J.J.: Introduction to robotics: mechanics and control, Addison Wesley Publishing Company, Reading, USA (1986).
8. Dafaoui El-M., Amirat Y., Pontnau J., and François C.: Analysis and Design of a Six-DOF Parallel Manipulator, Modeling, Singular Configurations, and Workspace, IEEE Trans. on Robotics and Automation **14**(1), 78-92 (February 1998).
9. Dasgupta B., and Mruthyunjaya T.S.: Closed-form dynamic equations of the general Stewart platform through the Newton-Euler approach, J. Mechanism and Machine Theory **33**(7), 993-1012 (October 1998).
10. Dasgupta B., and Mruthyunjaya T.S.: A Newton-Euler formulation for the inverse dynamics of the Stewart platform manipulator, J. Mechanism and Machine Theory **33**(8), 1135-1152 (November 1998).
11. Dasgupta B., and Choudhury P.: A general strategy based on the Newton-Euler approach for the dynamic formulation of parallel manipulators, J. Mechanism and Machine Theory **34**(6), 801-824 (August 1999).
12. Featherstone R.: The calculation of robot dynamics using articulated-body inertias, The Int. J. of Robotics Research **2**(3), 87-101 (1983).

13. Fichter E. F.: A Stewart platform based manipulator: general theory and practical construction. *Int. J. of Robotics Research* **5**(2), 157-181 (October 1986).
14. Geng, Z., Haynes, S., Lee, J. D. and Carrol, R. L.: On the dynamic model and kinematic analysis of a class of Stewart platforms, *Robotics and Autonomous Systems* **9**, 237-254 (1992).
15. Gosselin C. M.: Parallel computational algorithms for the kinematics and dynamics of parallel manipulators, *IEEE Int. Conf. on Robotics and Automation*, Vol. 1, pp.883-889, New York 1993.
16. Gugliemetti P., and Longchamp R.: A Closed form Inverse Dynamic Model of the Delta Parallel Robot, In *4th IFAC Symp. on Robot Control*, Syroco, pp. 51-56, Capri, 19-21 September 1994.
17. Hoffman, R. and Hoffman, M.: Vibrational modes of an aircraft simulator motion system. In *Proc. 5th World Congress on Theory of Machines and Mechanisms*, pp. 603-606, Montreal, July 1979.
18. Hollerbach J.M.: An iterative lagrangian formulation of manipulators dynamics and a comparative study of dynamics formulation complexity, *IEEE Trans. on Systems, Man, and Cybernetics* **SMC-10** (11), 730-736 (1980).
19. Ji Z.: Study of the effect of leg inertia in Stewart platform, In *IEEE Int. Conf. on Robotics and Automation*, pp. 121-126, Atlanta, May 1993.
20. Khalil W., and Kleinfinger J.-F.: A new geometric notation for open and closed-loop robots, *Proc. IEEE Conf. on Robotics and Automation*, pp. 1174-1180, San Francisco, April 1986.
21. Khalil W., and Kleinfinger J.-F.: Minimum operations and minimum parameters of the dynamic model of tree structure robots, *IEEE J. of Robotics and Automation* **RA-3**(6), 517-526 (December 1987).
22. Khalil W., and Creusot D.: SYMORO+: a system for the symbolic modelling of robots, *Robotica* **15**, 153-161 (1997).
23. Khalil W. and Guegan S.: Inverse and Direct Dynamic Modeling of Gough-Stewart Robots, *IEEE Transactions on Robotics and Automation*, 20(4), 754-762 (August 2004).
24. Khalil W., and Dombre E.: Modeling, identification and control of robots, Hermes Penton, London-Paris (2002).
25. Khatib O.: A unified approach for motion and force control of robot manipulators: the operational space formulation, *IEEE J. of Robotics and Automation* **RA-3**(1), 43-53 (February 1987).
26. Khosla P.K.: Real-time control and identification of direct drive manipulators, Ph. D. Thesis, Carnegie Mellon University, Pittsburgh, USA (1986).
27. Kleinfinger J.-F., and Khalil W.: Dynamic modelling of closed-chain robots, *16th Int Symp on Industrial Robots*, pp. 401-412, Bruxelles, 1986.
28. Lebret, G., Liu, G. K. and Lewis, F. L.: Dynamic analysis and control of a Stewart platform manipulator. *J. of Robotic Systems* **10**(5), 629-655 (July 1993).
29. Lee, K. M. and Shah, D. K.: Dynamic analysis of a three-degrees-of-freedom in-parallel actuated manipulator, *IEEE J. of Robotics and Automation* 4(3), 361-368 (June 1988).
30. Lilly K.W., and Orin D.E.: Efficient  $O(N)$  computation of the operational space inertia matrix, *Proc. IEEE Int. Conf. on Robotics and Automation*, pp. 1014-1019, Cincinnati, USA, May 1990.
31. Liu M.-J., Li C.-X., and Li C.-N.: Dynamics analysis of the Gough-Stewart platform manipulator, *IEEE Transaction on Robotics and Automation* **16**(1), 94-98 (February 2000).
32. Luh J.Y.S., Walker M.W., and Paul R.C.P.: On-line computational scheme for mechanical manipulators, *Trans. of ASME, J. of Dynamic Systems, Measurement, and Control* **102**(2), 69-76 (1980).

33. Luh J.Y.S., and Zheng, J. F.: Computation of Input generalized Forces for Robots with Closed Kinematic Chain mechanisms, IEEE Transactions on Robotics and Automation **RA-12**, 95-103 (1985).
34. Merlet J.-P.: Parallel robots, Kluwer Academic Publ., Dordrecht, The Netherland (2000).
35. Miller K.: Optimal Design and Modeling of Spatial Parallel Manipulators, The Int. J. of Robotics Research **23**(2), 127-140 (February 2004).
36. Nakamura Y., and Ghodoussi M.: A computational scheme of closed link robot dynamics derived by d'Alembert Principle, Int. Conf. on robotics and automation, IEEE, pp.1354-1360, Philadelphia 1988.
37. Ploen S.R., and Park F.C.: Coordinate-Invariant Algorithms for Robot Dynamics, IEEE Trans. On Robotics and automation **15**(6), 1130-1135 (December 1999).
38. Reboulet C., and Berthomieu T.: Dynamic models of a six degree of freedom parallel manipulators, ICAR, pp.1153-1157, Pise, June 1991.
39. Sciavicco L., and Siciliano B.: Modeling and control of robot manipulators McGraw Hill, New York (1994).
40. Sugimoto K.: Computational scheme for dynamic analysis of parallel manipulators, Transaction of the ASME, J. of mechanics, Transmission and Automation in Design **111**, 29-33 (1989).
41. Tsai L-W.: Solving the inverse dynamics of a Stewart-Gough manipulator by the principle of virtual work, J. of Mechanical design **122**, 3-9 (March 2000).
42. Walker M.W., and Orin D.E.: Efficient dynamic computer simulation of robotics mechanism, Trans. of ASME, J. of Dynamic Systems, Measurement, and Control **104**, 205-211 (1982).
43. Wittenburg. J.: Dynamics of Systems of Rigid Bodies. BG, Teubner (1977).