Dynamic Modeling and Control in Operational Space of an Hexapod Robot

* C. Mahfoudi, ** K. Djouani, † M. Bouaziz and S. Rechak
**University Paris12, LIIA laboratory
† Nationale Polytechnical School, Institute of Mechanics, Algiers

Abstract: This paper concerns real-times hexapod robot force control. Based on an operational trajectory planner, a computed torque control for each leg of hexapod robot is presented. This approach takes into account the real-time force distribution on the robot legs and the dynamic model of the hexapod. First, Kinematic and dynamic modeling are presented. Then, a methodology for the optimal force distribution is given. The force distribution problem is formulated in terms of a nonlinear programming problem under equality and inequality constraints. Then, according to X. Chen et al, the friction constraints are transformed from nonlinear inequalities into a combination of linear equalities and linear inequalities. Therefore, the overall hexapod computed torque control is presented. Simulations are given in order to show the effectiveness of the proposed approach. Finally, some remarks and perspectives are given.

Key-Words: hexapod Robot, Optimal Force Distribution, Dynamic modeling and Control

1 Introduction

Hexapod robots, as part of legged vehicles, can be used in work spaces with rough terrain, e.g. map building on an uneven ground, hazardous tasks like land mine searching and removing, volcano data collection, etc.

AS shown in [1], interests to walking machine, from research and application point, of views, are twofold. First, the complexity nature of legged locomotion has been very attractive and challenging to many pioneering researches.

Due to the complexity of the legged robots, applications in real world are not significant. Major problems concerns real time dynamic control of the legged robot under several constraints. To overcome such problems, dynamic model should be integrated in every control strategy.

Before we address the hexapod robot's dynamic modeling it is helpful to have an overall view of how the robot is controlled. In the task planning stage, a trajectory planner is used to determine a path that guides the hexapod from its initial position to a given final position. Then, a gait, which gives the position and events for placing and lifting the robot legs is selected [2]. The inverse kinematic model is, than used in order to to compute the desired trajectory (positions and velocities) in joint space. A joint computed torque control strategy is used for the hexapod real time control.

The proposed approach is based on the computation of the force distribution on the legs. Due to the existence of three actuated joints in each leg, the hexapod robot has redundant actuation leading to more active joints (18) than the robot platform degree-of-freedom (6 dof), figure(1). Thus, when formulating the force distribution problem, we find fewer force moment balancing equations than unknown variables. So, the solution of these equations is not unique. Moreover, some physical constraints, that concern the contact nature, friction, ...etc, must be taken into account in the calculation of force distribution. In addition, joints torque saturation must also be considered. Thus The Force Distribution Problem (FDP) can be formulated as a nonlinear constrained programming
problem under nonlinear equality and inequality
constraints. Several approaches have been pro-
posed for solving such a problem [3], [4],[5],[6], [7],
The robot crawling is divided into 3 phases. The
first phase, only 3 legs are supporting the robot,
for instance legs 1 - 2 - 3, leading to a force dis-
tribution problem with 9 unknown variables. Fur-
thermore, in the second phase, all the six legs are
supporting the robot leading to a force distribu-
tion problem with 18 unknown variables. In or-
der to reduce the problem complexity, we consider
that the contact forces on the legs 1-2-3 can be de-
curred from the first phase by introducing a con-
tinuous, decreasing function that varies from 1 to
0. Thus, the problem dimension, in the second
phase, can be reduced from 18 to 9. The third
phase is similar to the first one, with the legs 4-5-
6 supporting the robot. In the three phases, the
force-distribution problem is the same and solved
with the same algorithm. The rest of the paper
is organized as follows. Direct and inverse ge-
ometrical models of the hexapod are presented in
section 2. In section 3 the dynamic model of an
hexapod robot is derived. Section 4 concerns the
force distribution problem. For simulation a real
time control is presented in section 5.

2 Geometrical Modelling

Before presenting the direct and inverse geome-
trical model, let us consider the hexapod architec-
ture. As the hexapod legs are identical, only one
leg modelling is considered, the leg j architecture
is given in figure (10). Every leg "j" (j=1,...,6) is
fixed at the plate-forme by a revolute joint situ-
ted at \( l_j \) distance from the center of gravity of
the plate-forme (the body). The angle \( \phi_j \) rep-
resents the orientation of the coordinate frame
\((x_{1,j},y_{1,j},z_{1,j})\) fixed at the first articulation
of the leg and the coordinate fixed coordinate frame
of the body \((x_0,y_0,z_0)\). A walking robot is consid-
erate as an arborescent robot with some closed
loops. So to study this kind of robots we use the
method defined by Wissama and Klifinger [12].
The transformation matrix from ith joint’s at-
tached coordinate frame to the (i-1)th joint’s at-
tached coordinate frame is given by figure (11):

\[
\begin{bmatrix}
C \gamma_i C \theta_i - S \gamma_i S \theta_i - S \gamma_i C \alpha_i \theta_i \\
C \gamma_i C \theta_i + C \gamma_i C \alpha_i \theta_i - S \gamma_i S \theta_i + C \gamma_i C \alpha_i \theta_i \\
S \gamma_i S \theta_i \\
0 \\
0
\end{bmatrix}
\]

Thus:

\[
\begin{bmatrix}
C \gamma_i C \theta_i - S \gamma_i S \theta_i - S \gamma_i C \alpha_i \theta_i \\
C \gamma_i C \theta_i + C \gamma_i C \alpha_i \theta_i - S \gamma_i S \theta_i + C \gamma_i C \alpha_i \theta_i \\
S \gamma_i S \theta_i \\
0 \\
0
\end{bmatrix}
\]

The table (1) describes the transformation from
the world ground coordinate frame \((X,Y,Z)\) to
the coordinate frame at the contact point "4" of
each leg. The transformation providing the exact
position of the contact point"4" of any leg in the
absolute coordinate frame fixed at the ground is
given by:

\[
R_T = R T_0 T_1 T_2 T_3 T_4
\]

When the position and the orientation of the last
coordinate frame fixed to the end of each leg "j"
are known ,we apply the method proposed by
Paul [13]. It provides the values of the joints co-
ordinates \( \theta_{i,j} (i = 1, 2, 3) (j = 1, ..., 6) \).
3 Hexapod dynamics model

3.1 introduction

The robot dynamics is given by [14] [12] [15] :

\[ \Gamma = f(\theta, \dot{\theta}, \ddot{\theta}) \]  

(3)

Where, \( \theta \), \( \dot{\theta} \), and \( \ddot{\theta} \) are respectively the generalizes coordinates, speeds and acceleration.

The explicit form of Eq(3) can be expressed as follows for any leg "j":

\[ \Gamma = M(\theta)\ddot{\theta} + C(\theta, \dot{\theta}) \dot{\theta} + Q(\theta) + J^T f \]  

(4)

where \( M(\theta) \in \mathbb{R}^{3\times3} \), \( C(\theta, \dot{\theta}) \in \mathbb{R}^{3\times1} \), \( Q(\theta) \in \mathbb{R}^{3\times1} \) and \( J^T f \in \mathbb{R}^{3\times1} \).

- \( M(\theta) \), matrix \((n \times n)\) representing the inertia of the robot, which is deduced from the kinetic energy.
- \( C(\theta, \dot{\theta}) \), a vector \((n \times 1)\) representing coriolis torques and centrifuges forces.
- \( Q(\theta) = [Q_1, \ldots, Q_n] \), a vector of gravity torque and forces.
- \( f \) the reaction of the ground
- \( E \) and \( U \) are respectively the kinetic and the potential energy of the system

The Eq(3) can be rewritten as:

\[ \Gamma = M(\theta)\ddot{\theta} + H(\theta, \dot{\theta}) + J^T f \]  

(5)

with :

\[ H(\theta, \dot{\theta}) = C(\theta, \dot{\theta}) \dot{\theta} + Q(\theta) \]

Then, we can write:

\[ H(\theta, \dot{\theta}) = \Gamma ; \text{ i.e. } \ddot{\theta} = 0 \text{ and } f = 0 \]

So if we use Newton-Euler formalism with \( \ddot{\theta} = 0 \) and \( f = 0 \) we obtained the value of \( H(\theta, \dot{\theta}) \). This transformation is very important and permits to:

- extract the acceleration vector \( \ddot{\theta} \):
  \[ \ddot{\theta} = M^{-1}(\theta)(\Gamma - J^T f - H(\theta, \dot{\theta})) \]

- avoid the computation of the vector \( C(\theta, \dot{\theta}) \) which has redundant algorithm.

Figure 4: Forces acting on the hexapod system

3.2 Newton-Euler Formalism

Remark: The contact forces on the ground, \( f_x, f_y, f_z \) are 0 if the leg is lifted and \( \neq 0 \) otherwise.

The Newton-Euler algorithm can be established as follow [12]:

3.2.1 Velocities and accelerations computation

let \( C_i \) be any link of a leg "j", figures (4)(5), \( v_i \) the velocity of the gravity center \( G_i \) and \( \omega_i \) the angular velocity of the link \( C_i \). Let \( \gamma_i = v_i \) the acceleration of \( G_i \) and \( \alpha_i = \dot{\omega}_i \) the angular acceleration of the link \( C_i \). Then:

\[ P_{oi} = P_{o,i-1} + P_{i-1,i} \]  

(6)

and

\[ d_{oi} = P_{oi} + d_{i,i} \]  

(7)

after derivation :

\[ P_{oi} = P_{o,i-1} + \omega_{i-1} \wedge P_{i-1,i} + \sigma_i \dot{\theta}_i Z_i \]  

(8)

In our case, we have only rotational articulations so \( \sigma_i = 0 \), then:

\[ v_i = d_{oi} = P_{oi} + \omega_{i-1} \wedge d_{i,i} \]

The second derivation gives:

\[ \dot{P}_{oi} = \dot{P}_{o,i-1} + a_{i-1} \wedge P_{i-1,i} + \omega_{i-1} \wedge (\omega_{i-1} \wedge P_{i-1,i}) + \sigma_i (2\dot{\theta}_i \dot{\omega}_{i-1} \wedge Z_i + \dot{\theta}_i Z_i) \]
Figure 5: Kinematics parameters representation of a link ”i” in a leg ”j”

And
\[
\gamma_i = \dot{\gamma}_i = \ddot{d}_{oi} = \ddot{P}_{oi} + a_i - d_{i,i} + \omega_i \wedge \left( \omega_{i-1} \wedge d_{i,i} \right) \tag{10}
\]

\( \gamma_i \) represents the gravity center acceleration.

There fore we can define the global acceleration with the following relation:
\[
a_i = \dot{P}_{oi} + g Z_0 \tag{11}
\]

where, \( g \) is the gravity acceleration with:
\[
a_i - \omega_i \wedge \left( \omega_i \wedge \omega_i \right) = \mathbf{b}_i \tag{12}
\]

In condensed forme:
\[
\mathbf{b}_i = \hat{\mathbf{b}} + \mathbf{b}_i \hat{\mathbf{b}} \tag{13}
\]

\( \hat{\mathbf{b}} \) and \( \mathbf{b}_i \hat{\mathbf{b}} \) : represent the skew symmetric matrixes.

Eq(11) can be represented in condensed forme:
\[
a_i = \sigma_i t_i + r_{i-1} \tag{14}
\]

with:
\[
\begin{cases}
  t_i = 2 \dot{\omega}_{i-1} \wedge \mathbf{Z}_i + \dot{\mathbf{Z}}_i \\
r_{i-1} = \alpha_{i-1} + \dot{\mathbf{b}}_{i-1} \mathbf{P}_{i-1,i}
\end{cases}
\]

In our case, \( \sigma_i t_i = 0 \) because \( \sigma_i = 0 \), then we will have:
\[
\gamma_i + g Z_0 = \alpha_i + \mathbf{b}_i d_{i,i} \tag{15}
\]

Figure 6: Equilibrium of a link ”i” in a leg ”j”

3.2.2 Forces and torques computation

- Let \( \mathbf{c}_i \), the interaction torque exerted by the link \( C_{i-1} \) on the link \( C_i \) figure(6).
- \( \mathbf{f}_i \), the interaction force exerted on \( o_i \) by the link \( C_{i-1} \) on the link \( C_i \).
- \( \mathbf{F}^i = \mathbf{f} + \sigma_i \Gamma^i \mathbf{Z}_i \), the force exerted, on \( o_i \) by the \( i^{th} \) actomyon on the link \( C_i \), where \( \Gamma^i \) is a scalar, representing the moment of the motor.
- \( \mathbf{C}^i = \mathbf{c}^i + \sigma_i \Gamma^i \mathbf{Z}_i \), the moment exerted, by the link \( C_{i-1} \) and the \( i^{th} \) actomyon on the link \( C_i \).

The equilibrium of the link \( C_i \) can be represented by:
\[
\mathbf{F}_{res} = \mathbf{F}^i - \mathbf{F}^{i+1} - m_i g Z_0 = m_i \gamma_i \tag{16}
\]
\[
\mathbf{C}_{res} = \mathbf{C}^i - \mathbf{C}^{i+1} - d_{i,i} \wedge \mathbf{F}^i + d_{i+1,i} \wedge \mathbf{F}^{i+1} \tag{17}
\]

and
\[
\mathbf{C}_{res} = \psi^i \mathbf{a}_i - \omega_i \wedge (\psi^i \omega_i) \tag{18}
\]

Where, \( \psi^i \) is the inertia matrixes of the link \( C_i \), from Eq(16):
\[
\mathbf{F}^i = \mathbf{F}^{i+1} + m_i (\alpha_i + \mathbf{b}_i d_{i,i}) \tag{19}
\]

Then, from Eq(17) and (18) we obtained:
\[
\mathbf{C}^i = \mathbf{C}^{i+1} + \psi^i \mathbf{a}_i + \omega_i \wedge (\psi^i \omega_i)
+ d_{i,i} \wedge (\mathbf{F}^i - \mathbf{F}^{i+1}) + \mathbf{P}_{i,i+1} \wedge \mathbf{F}^{i+1}
3.2.3 The kinetic energy computation

The total kinetic energy of the system is given as follows:

\[ E = \sum_{j=1}^{n} E_j \]

(20)

where \( E_j \), kinetic energy of the link \( C_j \), expressed by the following equation:

\[ E_j = \frac{1}{2} (\omega_j^T \varphi^j \omega_j + m_j V_{Gj}^T V_{Gj}) \]

(21)

\( \omega_j \) : the instantaneous rotation velocity of the link "\( C_j \)" expressed in the coordinate frame "\( j \)".

\( V_{Gj} \) : linear velocity of the gravity center of the link "\( C_j \)" expressed in the coordinate frame "\( j \)".

as the figure (7) show:

![Diagram of Link C_j](image)

**Figure 7: Modelling a link of a leg**

\[ V_{Gj} = V_j + \omega_j d_{jj} \]

(22)

as we know:

\[ k_j = \varphi^j - m_j \dot{d}_{jj} \dot{d}_{jj} \]

(23)

Eq(21) can be transformed:

\[ E_j = \frac{1}{2} (\omega_j^T k_j \dot{\omega}_j + m_j V_j^T \dot{V}_j + 2m_j \dot{d}_{jj}^T (\dot{V}_j \land \dot{\omega}_j)) \]

(24)

with:

\[ \dot{\omega}_j = \dot{A}_{j-1} \dot{\omega}_{j-1} + \sigma_j \dot{\theta}_j \dot{Z}_j = \dot{\omega}_{j-1} + \sigma_j \dot{\theta}_j \dot{Z}_j \]

(25)

and,

\[ \dot{V}_j = \dot{A}_{j-1} (j^{-1} V_{j-1} + \dot{j}^{-1} \omega_{j-1} \land P_{j-1,j}) + \sigma_j \dot{\theta}_j \dot{Z}_j \]

(26)

\( \dot{A}_{j-1} \in \mathbb{R}^{3 \times 3} \), represented the orientation matrix.

3.2.4 Elements \( M_{ij} \) of the matrix \( M \)

In the end we computed the matrix \( M \) from Eqs(20)(24):

The elements \( M_{ii} \) of the matrix \( M \) is equal to the coefficient of \( \dot{\theta}_i^2 / 2 \) in the expression of kinetic energy, and the elements \( M_{ij} \), if \( i \neq j \) is equal to the coefficient of \( \dot{\theta}_i \theta_j \).

\[ M_{11} = I_{z1} + S_2 I_{x2} + C_2^2 I_{y2} + S23 I_{x3} + C23 I_{y3} + m_3 C23 C_{23} + Ia_1 \]

\[ M_{12} = M_{13} = 0 \]

\[ M_{22} = I_{z2} + I_{z3} + S_3 I_{x3} + C_3^2 I_{y2} + m_3 C3 I_{23} + Ia_2 \]

\[ M_{23} = I_{33} + 1/2 m_3 C3 I_{23} \]

With, \( (Ia_1, Ia_2 \ and \ Ia_3) \) represent the inertsias of the motors. These results are obtained with diagonal inertia matrix of the legs links.

At the end of computation for all legs, we can describe the equilibrium of the plat-form as following, figure (8).

![Diagram of Equilibrium of the plat-form](image)

**Figure 8: Equilibrium of the plat-form**

- let \( \gamma_{0,0} \) and \( \omega_{0,0} \) respectively the plat-form desired linear acceleration and angular acceleration in the coordinates frame \((x_0, y_0, z_0)\).
- \( P_{0,j} \), the force applied by the leg "\( j \)" at the articulation "\( 1 \)" on the plat-form "\( 0 \)".
- \( C_{0,j} \), the moment applied by the \( j \)th leg in the articulation "\( 1 \)" on the plat-form.
- \( P_{01,j} \), the distance between the articulation "\( 1 \)" of the \( j \)th leg and the origin of the coordi-
nate frame \((x_0, y_0, z_0)\) expressed in the same coordinate frame.

The application of the dynamic fundamental principle, at the mass center of the plate-form provides the following matrix equation:

\[
\begin{pmatrix}
m_0 \mathbf{I}_3 & 0 \\
0 & \varphi^0
\end{pmatrix}
\begin{pmatrix}
\gamma_{0,0} \\
\omega_{0,0}
\end{pmatrix} + \begin{pmatrix}
-m_0 \mathbf{g}_0 \\
\omega_{0,0} \wedge (\varphi^0 \omega_{0,0})
\end{pmatrix} = \begin{pmatrix}
\mathbf{F} \\
\mathbf{M}
\end{pmatrix}
\tag{27}
\]

Where,

- \(m_0\) and \(\varphi^0\) are respectively the masse and the inertia matrix of the plat-form.
- \(\mathbf{I}_3\), the identity matrix \((3 \times 3)\).
- \(\mathbf{g}_0\) : gravity vector

\(\mathbf{F}\) and \(\mathbf{M}\) are definite as follow:

\[
\begin{align*}
\mathbf{F} &= \sum_{j=1}^{6} F_{1,j}^0 \\
\mathbf{M} &= \sum_{j=1}^{6} (C_{1,j}^0 + P_{0,1,j}^0 \wedge \mathbf{F}_{1,j}^0)
\end{align*}
\tag{28}
\]

4 Force Distribution Problem

4.1 problem Formulation

The force system acting on a hexapod robot is shown in figure (9). For simplicity, only the force components on the foot are presented here. In the general case, rotational torques at the feet are neglected. Let \((x_0, y_0, z_0)\) be the robot fixed body coordinate frame in which the body is located in the \((x_0, y_0)\) plane and \((x_{1,j}, y_{1,j}, z_{1,j})\) denote the coordinate frame fixed at the foot "j", in which the leg \(j\) lies in the \((x_{1,j}, z_{1,j})\) plane and its \(z\) axis is normal to the support surface of the foot which is assumed to be parallel to the \((x_0, y_0, z_0)\) plane. \(\mathbf{F} = [F_x \ F_y \ F_z]^T\) and \(\mathbf{M} = [M_x \ M_y \ M_z]^T\) denote respectively the robot body force vector and moment vector, which result from the gravity and the external force acting on the robot body. Define \(f_{x,j}, f_{y,j}\), and \(f_{z,j}\) as the components of the force acting on the supporting foot "j" in the directions of \(x_0, y_0\) and \(z_0\), respectively. The number of supporting feet, \(n\), can vary between 3 and 6 for a hexapod robot. The robot’s quasi-static force/moment equation can be written as:

\[
\begin{align*}
\sum_{j=1}^{n} f_j &= \mathbf{F} \\
\sum_{j=1}^{n} \mathbf{OP}_j \wedge f_j &= \mathbf{M}
\end{align*}
\tag{29}
\]

Figure 9: Orientation of coordinate frame

where \(\mathbf{OP}_j\) is the position vector joining contact point of the leg "j" and the gravity center of the body. The general matrix form of this equation can be written as:

\[
\mathbf{A} \mathbf{G} = \mathbf{W}
\tag{30}
\]

with:

\[
\begin{align*}
\mathbf{G} &= [f_1^T \ f_2^T \ \cdots \ f_n^T]^T \in \mathbb{R}^{3n} \\
\mathbf{f}_j^T &= [f_{x,j} \ f_{y,j} \ f_{z,j}]^T \in \mathbb{R}^3 \\
\mathbf{W} &= [\mathbf{F}^T \ \mathbf{M}^T]^T \in \mathbb{R}^{6n}
\end{align*}
\]

\[
\mathbf{A} = \begin{pmatrix}
\mathbf{I}_3 & \cdots & \mathbf{I}_3 \\
\mathbf{B}_1 & \cdots & \mathbf{B}_n
\end{pmatrix} \in \mathbb{R}^{6n \times 3n}
\]

\[
\mathbf{B}_j \equiv \mathbf{OP}_j \equiv \begin{pmatrix}
0 & -P_{z,j} & P_{y,j} \\
P_{z,j} & 0 & -P_{x,j} \\
-P_{y,j} & P_{x,j} & 0
\end{pmatrix} \in \mathbb{R}^{3 \times 3}
\]

where \(\mathbf{I}_3\) is the identity matrix and \(\mathbf{G}\) is the foot force vector, corresponding to three \((\mathbf{G} \in \mathbb{R}^9)\) or six \((\mathbf{G} \in \mathbb{R}^{18})\) supporting legs. \(\mathbf{A}\) is a coefficient matrix which is a function of the positions of the supporting feet, and \(\mathbf{B}_j\) is a skew symmetric matrix consisting of \((P_{x,j}, P_{y,j}, P_{z,j})\), which is the position coordinate of the supporting foot "j" in \((x_0, y_0, z_0)\). \(\mathbf{W}\) is a total body force/moment vector. It is clear that Eq(30) is an underdetermined system and its solution is not unique. In other words, the foot forces have many solutions according to the equilibrium equation. However, the foot forces must meet the needs for the following physical constraints, otherwise they become invalid:
1. Supporting feet should not slip when the robot walks on the ground. It results in the following constraint:

$$\sqrt{f_{x,j}^2 + f_{y,j}^2} \leq \mu f_{z,j} \quad (31)$$

where $\mu$ is the static coefficient of friction of the ground.

2. Since the feet forces are generated from the corresponding actuators of joints, the physical limits of the joint torques must be taken into account. It follows that:

$$-\tau_{jmax} \leq J_j^T J_{ij} A_{ij} \left( \begin{array}{c} f_{x,j} \\ f_{y,j} \\ f_{z,j} \end{array} \right) \leq \tau_{jmax} \quad (32)$$

for $(j = 1, \ldots, n)$, where $J_j \in \mathbb{R}^{3 \times 3}$ is the Jacobian of the leg $"j"$, $\tau_{jmax} \in \mathbb{R}^{3 \times 1}$ is the maximum joint torque vector of the leg "j", and $A_{ij} \in \mathbb{R}^{3 \times 3}$ is the orientation matrix of $(x_{0,j}, y_{1,j}, z_{1,j})$ with respect to $(x_0, y_0, z_0)$.

3. In order to have definite contact with the ground, there must exist a $f_{z,j}$ such that:

$$f_{z,j} \geq 0 \quad (33)$$

In the following, we propose an approach for problem size reduction, linearisation and solving for the hexapod case. Clearly, it is difficult to solve such a nonlinear programming problem for real-time feet force distribution with complex constraints. The solution of this system is developed in [2].

5 Computed-torque control

Suppose that desired trajectory $X_d(t)$ has been selected for the arm motion. To ensure trajectory tracking by the joint variable errors [14] [12][16][15].

$$e(t) = X_d - X(t) \quad (34)$$

Then the overall robot arm input becomes:

$$\Gamma = AJ^{-1} (\dot{X} - J\dot{\theta}) + H \quad (35)$$

$$X(t) = \bar{X} + k_v (X_d - \bar{X}) + k_p (X_d - X) \quad (36)$$

This controller is shown in figures (12),(13)

5.1 Choice of PD Gains

It is usual to take the $n \times n$ matrices diagonal so that:

$$k_v = diag[k_{vi}], \quad k_p = diag[k_{pi}]$$

and $k_{vi} = \omega_n^2$, $k_{pi} = 2\xi \omega_n$ with $\xi$ the damping ratio and $\omega_n$ the natural frequency. The PD gains are usually selected for critical damping $\xi=1$. Then, to avoid exciting the resonant mode, we should select natural frequency to half the resonant frequency $\omega_n < \omega_r / 2$.

5.2 Simulation results

In order to show the effectiveness of proposed approach, some simulations were conducted under Matlab. We consider that the hexapod robot is crawling in a linear trajectory ($Y=3X$), on an uneven ground, in the X-Y plane. Furthermore,
Coomatetices
X = [x, y, z] Of the leg Cartesian Kinematics, for 0, z

\[ F = [f_x, f_y, f_z] \]

\[ X = f(\theta_1, \theta_2, \theta_3, \theta_4, \theta_5, \theta_6) \]

\[ \theta_1 \rightarrow \theta_6 \]

\[ \text{Mass} \text{ of \ leg} \]

\[ 80 \]

\[ 60 \]

\[ 40 \]

\[ 20 \]

\[ 0 \]

\[ 300 \]

\[ 200 \]

\[ 100 \]

\[ \text{Trajectory of the hexapod} \]

\[ \text{Body} \]

\[ \text{Legs} \]

\[ \text{Trajectory of the hexapod} \]

force tensor acting at the body center are (Fx = -3, Fy = 5, Fz = -50 [N], Mx = 0, My = 2, Mz = 1 [Nm]). The basic mechanism, size and parameters of Hexapod robot are shown in Figure (10) and (11), where a = 0.25 [m], b = 0.6 [m], \( l_1 = 0.05 \) [m], \( l_2 = 0.20 \) [m], \( l_3 = 0.30 \) [m] and \( l_4 \approx 0 \) [m]. There are three actuated joints \( \theta_1, \theta_2, \text{and} \theta_3, \text{in the leg } "j", \text{for } (j=1,\ldots,6). \) The masses and the inertia of links are respectively (m1 = 0.1, m2 = 0.07, m3 = 0.03 [kg]) and (Ix1 = 1.36, Iy1 = 0.297, Iz1 = 1.6, Ix2 = 2.1, Iy2 = 2.29, Iz2 = 0.33, Ix3 = 0.001, Iy3 = 0.05, Iz3 = 0.05 [kg cm²]). The simulation is presented for two cycles of walking corresponding to 15 seconds. The lifted legs do a cycloid trajectory.

6 Conclusion

In this paper, we have presented a dynamic decoupled control for a hexapod walking machine. The proposed approach is based on a trajectory planner in operational space, a real time computing of the force distribution on the hexapod legs and a joint computed torque control strategy at the level. The force distribution problem has been formulated in terms of non-linear programming problem has been solved as a quadrature optimization problem. Simulations where given in order to show the effectiveness of the proposed approach.

References


Figure 15: Errors in coordinates joints


