Journal of Intelligent and Robotic Systems **11**: 269–291, 1995.

Study of a Distributed Control Architecture for a Quadruped Robot

CLAUDE VILLARD, PHILIPPE GORCE, and JEAN-GUY FONTAINE
Laboratoire de Robotique de Paris, 10-12, Avenue de l'Europe, 78140 Velizy, France

(Received: 20 January 1993; in final form: 17 September 1993)

Abstract. Looking at legged robots, it is sometimes very important to take into account some of the practical aspects (when focusing on theoretical ones) in order to implement control-command levels.

In this way, we have treated the problem of the realization of dynamic or quasi-dynamic gaits with a quadruped robot using a new approach from which we have derived an efficient control/command scheme. This is based on a simple consideration which lies in the fact that the Dynamic Model (DM) can be decomposed into two main parts. From our point of view, we consider a part devoted to the command of the legs which could be called a Leg Inverse Dynamic Model (LIDM). We consider a second part dealing with the global characteristics of the platform. At this level, one can control the system. It will be called the LPIM (Leg to Platform Interaction Model).

This goal is reached assuming a dichotomy in a distributed architecture and by the way we present it. Further justification of our method will be given in several stages throughout the paper. We paid great attention to time-saving considerations with respect to communication protocols and data exchange at the same level and between the three levels we derived from our basic investigations.

Key words. Quadruped robot, hierarchical control/command architecture, dynamic model, realime computation.

1. Introduction

In the field of mobile robots, legged robots can be considered as recent in terms of scientific investigations. During the past decade, many studies have contributed to their development. The first significant result was the realization of the General Electric Quadruped Transporter, first tested in 1968 [42]. In the operation of this vehicle, a human provided direct master/slave control over each of its four legs. Afterwards, studies began to include a computer in the control structure to permit the lower-level tasks of coordination to be accomplished automatically. The concept of supervisory control for multilegged robots was developed. Then a human operator is called upon to solve the higher-level control tasks but it relieved of the joints-coordination function [10, 14, 51].

For the purpose of increasing their performances in the field of energy efficiency and adaptability on irregular terrain, much progress has been made on high level control/command and dynamical gaits. For that purpose, various legged machines have been built which could be regrouped into four types:

- Monopeds which allow the study of the control of a system obliged to keep equilibrium [32, 47, 52],
- Bipeds which aid in the understanding of human walking and in the conception of multilegged machines [13, 40, 55],
- Hexapods which are very helpful in high-level strategy elaboration (obstacles crossing, reflex, etc.) because they relieve stability control [7, 17, 29, 30, 45, 53].
- And finally, quadrupeds which (for dynamical gaits) centralize, within a same system, the problems of stability and control/command [1, 15, 22, 23, 34, 35, 48].

2. Previous Studies

Currently, the real-time determination of the necessary torques for realizing the movement law of a robot platform remains an open problem.

Such systems are characterized by the fact that the legs' platform, together with the ground, form open (leg in transfer) and closed (leg in contact) chain combinations. These combinations evolve during the movement and are dependent on the gait. Dynamical effects resulting from the interaction between the different robot elements are dependent on those combinations but are also dependent on the nature of the terrain. Then, their modelization is not easy.

However, many simulations about the dynamic behavior of legged robots have been developed from more or less complex modelizations.

In this approach, Shih [49] includes the effect of leg mass and compliance, joint compliance and friction, as well as the effects of leg contact with the ground. Compliance and contact are modeled by a spring/damper system. Dynamical equations are established and resolved for each body which was initially considered independently. The dynamic reaction forces and torques evaluated for each link are related to those of the neighboring links using a compliant model of the interconnecting joint. The considered joint compliant model is simulated by adjusting the considered spring/damper values. This approach can require a large number of system states depending on the gait and terrain being more realistic.

The algorithm of Lilly and Orin [37] is based on the fact that the robot is considered as a combination of multiple manipulators (legs) sensing an object (body), with the ground contact modeled as a manipulator link. This approach does include ground compliance.

Ouezdou [46] uses an algorithm with a spatial notation for unburdening the writing of dynamical equations. The effect of ground contact impact is included. This modelization is based on the robot IDM resolution using optimization techniques.

This approach does not include joint compliance. Besides, its efficiency depends on the choice of criterion for optimization. About the criterion, we could remark that Kimura [28] has made a study of the relationship between the locomotion parameters (gait, stride length, etc.) and the stability, maximum speed, and energy consumption criteria.

Finally, the approach of Freeman and Orin [11] introduces the notion of a Decoupled-Tree Structure (DTS). The closed-chain system is decoupled in an opened-chain and through the introduction of spring/damper systems at the

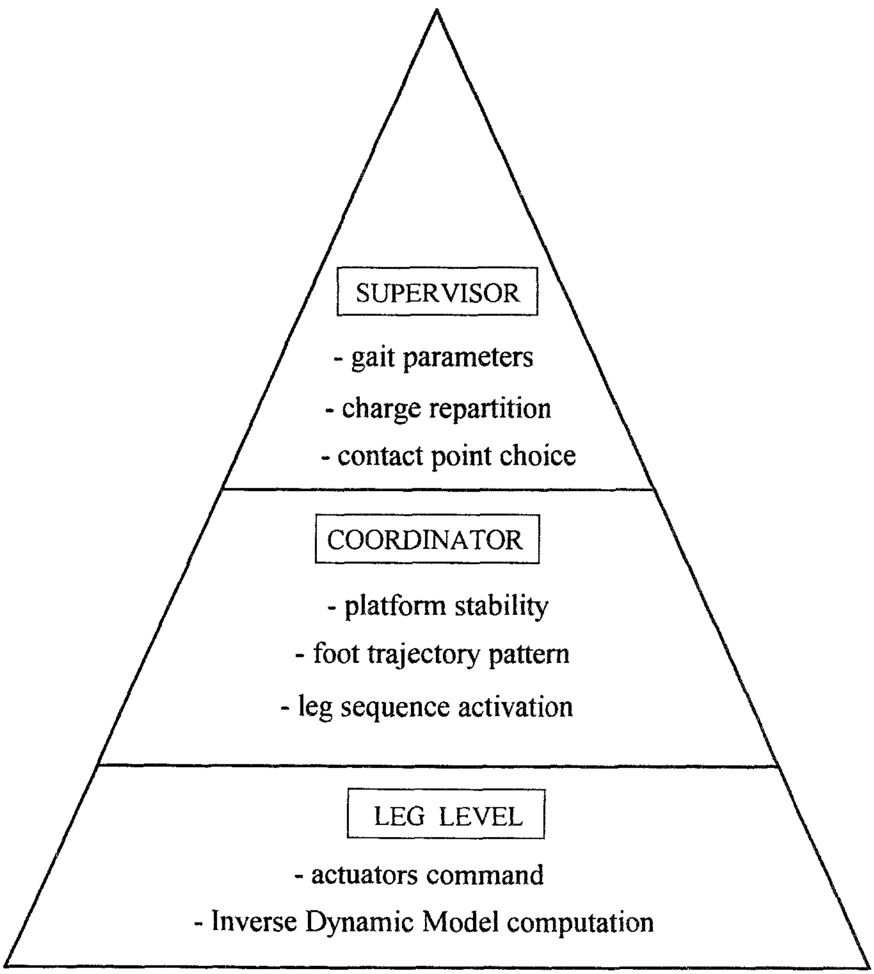


Fig. 1. Control levels of RALPHY architecture.

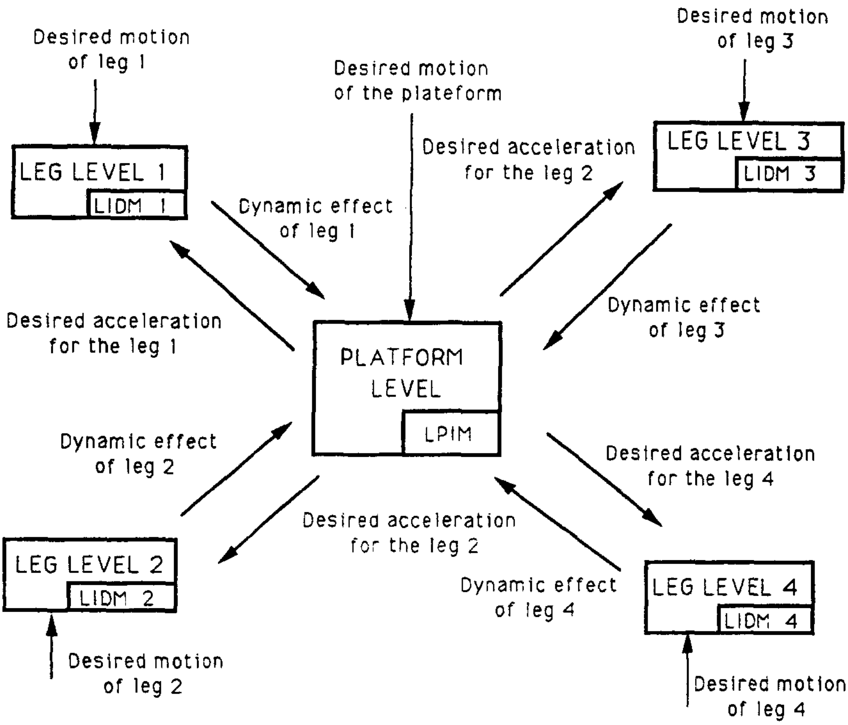


Fig. 2. Data exchange coordination between the Coordinator and the Leg Level.

ground only. It remains tree-structured when the body is the root with the legs branching from the reference member.

For the authors, this technique represents somewhat of a compromise between the completely decoupled method of Shih and the more coupled method of Lilly and Orin.

All these simulation tools do not allow real-time application, even if it is possible to increase the efficiency of some algorithms by introducing the notion of parallel computation.

So, to do real-time feedback control, it is necessary to devise a model which realizes a compromise between complexity and computational time but also combines a theoretical and practical approach.

In order to keep a homogeneous approach in the conception of system more and more intelligent at low-level, it is necessary to define control architectures which include real-time notions and dynamic command. With this in mind, we have treated the resolution of the RALPHY's IDM as follows:

- First, the robot is supposed to be composed of four independent subsystems each one formed by a part of the platform and one leg. This allows us to reduce our problem to the real-time IDM resolution of a particular manipulator (mobile basis and end effort evolving with the gait and nature of the terrain) and then to study the LPIM.

- Second, to insure a correct motion of the robot we have taken into account the fact that the legs are not really independent but linked to the same platform. So, control of the four legs will be assigned to an upper level: the coordinator whose role is to regulate the locomotion of our robot. For that, in real-time, it has to perform the following functions:

  - to regulate the gait robot,
  - to command the leg movements,
  - to control the stability platform,
  - to choose the repartition of the weight on each leg in contact with the ground,
  - to choose where to put the foot to obtain the appropriate sustentation.

In aiming to retain the most possible decentralized and hierarchized architecture and by using sensor information, we think that it is better to move the two latter functions to upper levels.

According to this approach, it is possible to globally define a hierarchized control/command architecture (Figure 1). Data exchanges between the coordinator and the leg level are indicated in Figure 2.

3. Up to the LIDM

We have compared our project with the case of a manipulator in order to understand the mechanism described by some authors.

By the way, in the field of manipulators, various IDM resolution algorithms have been written. Most of them are based on two very often used formalisms in robotics: Newton–Euler and Lagrange. Moreover, studies have been made to compare the efficiency of the different technics based on these two formalisms [24, 44, 50]. Notice, nevertheless, that Kane and Levinson have presented a new based on Kane's dynamical equations [25]. Our method is based on Newton–Euler's equations.

In order to reduce computational time, some authors have presented new notions. Some of them have developed parallel computational schemes. Hashimoto and Zom have recently developed an architecture founded on a network of T800s transputers [19, 20, 56]. Luh and Kasahara use more currently available microprocessors (Intel 8086 microprocessor and 8087 floating-point coprocessor) [26, 39]. Others have introduced new concepts: 'Cartesian tensor' by Balafoutis [5], 'augmented body', and 'barycenter' by He [21] or 'Lie group theory' by Mladenova [41].

However, to be efficient, these schemes and concepts require either too electronic or too perfect computers. Facing constraints like weight and dimension, we have chosen an algorithm based on a very classic description of the 'manipulator'. This algorithm executes (online) a 'serial' previously optimized computation (outline).

4. Robot Description

Basically, our robot is devoted to unstructured environments like planetary surfaces, offshore exploration, or agricultural lands. Despite a significant increase in the complexity of control systems compared to wheeled robots, legged robots are more efficient for avoiding natural obstacles or coping with them in a three-dimensional constraint space.

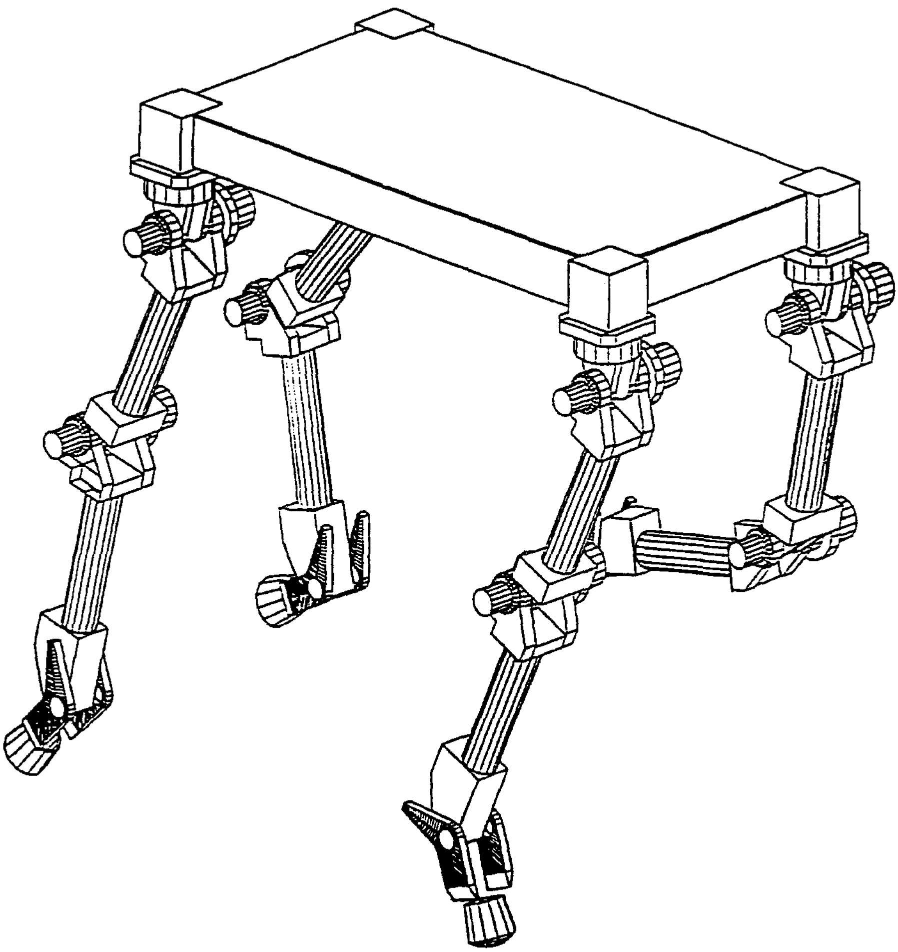
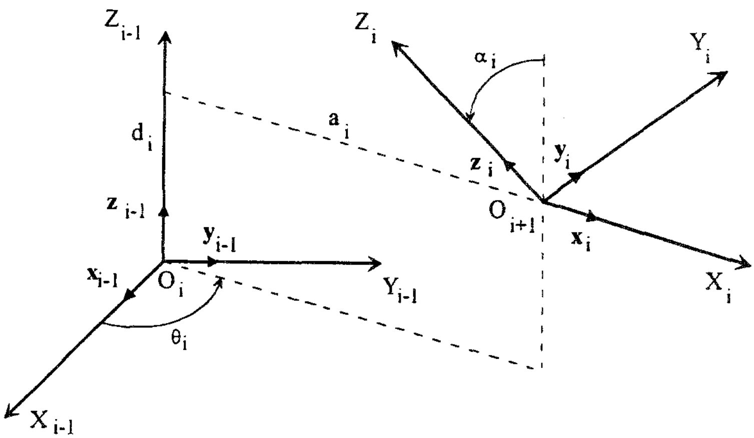


Fig. 3. RALPHY.

Our project RALPHY (Figure 3) is able to realize insect or mammal gaits. It is a four-legged structure half way between a dynamic structure (one or two legs on the ground at a time) and a static one (four or more legs on the ground at the same time). Kinematics and dynamics studies have been previously done [46]. The shoulders are driven by electric actuators for insect configurations while the others are pneumatically activated. These last ones fulfill low weight conditions which are a prerequisite for reaching high dynamics performances [9]. The study and implementation of the low-level supervisor have been presented in [12]. In this study, we will focus on two degrees of freedom on each leg.

5. Leg Description

The Newton–Euler recursive formalism allows us to obtain a dynamic model of the leg when writing, for each link, dynamic equations of applied forces and moments. The 'recursive' expression is explained by the fact that on the one hand the ith body velocity and acceleration are calculated as a function of those of the $(i-1)$th body and on the other hand, the effort applied on this ith body is determined as a function of the effort applied by the $(i+1)$th body. Practical development of this formalism uses the convention introduce by Luh, Walker, and Paul [38] which consists in projecting the relative values of a body in a coordinate frame linked to this body. We have preferred to use this initial version of the recursive Newton–Euler formalism because it allows us to obtain a noncompacted set of vectorial expressions. So, it is more easy to develop



$$
{}^{i-1}A_i = \begin{bmatrix} \cos\theta_i & -\sin\theta_i \cdot \cos\alpha_i & \sin\theta_i \cdot \sin\alpha_i & a_i \cdot \cos\theta_i \\ \sin\theta_i & \cos\theta_i \cdot \cos\alpha_i & -\cos\theta_i \cdot \sin\alpha_i & a_i \cdot \sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}
$$

Fig. 4. Denavit–Hartenberg's convention.

a customized algorithm adapted to the capacities of the embedded processors, that is to say, containing only a list of basic scalar operations. With regard to the homogeneous transformation matrix, using the transformation between the coordinate frame i and $i+1$, we have retained the Denavit–Hartenberg convention (Figure 4) [8].

5.1. NOTATION

Unless mentioned otherwise, we shall use the following notation in the rest of this paper:

${}^{i}\mathbf{r}_{i,i+1}$: position vector of the $(i + 1)$th coordinate frame relative to the ith coordinate frame expressed in the ith frame;
${}^{i}\mathbf{r}_{j,ck}$: position vector to the center of mass kth relative to the jth coordinate frame expressed in the ith frame;
${}^{j}\mathbf{w}_{k}({}^{j}\dot{\mathbf{w}}_{k})$: angular velocity (acceleration) of the kth coordinate expressed in the jth frame;
$\theta_i(\dot{\theta}_i, \ddot{\theta}_i)$: angle (velocity, acceleration) between the x_i axis of the $\mathbf{R}_i$ coordinate frame and the $\mathbf{x}_{i+1}$ axis of the $\mathbf{R}_{i+1}$ coordinate frame following the Denavit–Hartenberg convention;
$\mathbf{k}_{i-1}$: unit vector along the joint axis $i - 1$;
${}^{j}\mathbf{a}_{k}$: absolute acceleration of the **k**th coordinate frame expressed in the **j**th coordinate frame;
${}^{i}\mathbf{a}_{ci}$: absolute acceleration of the **i**th center of mass expressed in the **i**th coordinate frame;
${}^{i}\mathbf{f}_{j,k}$: force applied to the **k**th body by the **j**th body expressed in the **i**th coordinate frame;
${}^{i}\mathbf{M}_{j,k}$: moment applied to the **k**th body by the **j**th body expressed in the **i**th coordinate frame;
τ_i: torque applied on the **i**th joint;
$\mathbf{R}_{c0}$: platform coordinate frame expressed in the center of mass.

5.2. KINEMATICS OF THE LEG

Like a classical manipulator, the leg can be basically represented by a serial linkage of rigid bodies connected sequentially by either a prismatic or revolute joint. In our case, the joint will always be a revolute joint.

The absolute acceleration of the origin O_{i+1} of the $(i + 1)$th frame can be expressed in terms of the ith frame as follow:

$$ {}^{i}\mathbf{a}_{i+1} = {}^{i}\mathbf{a}_{i} + {}^{i}\dot{\mathbf{w}}_{i+1} \times {}^{i}\mathbf{r}_{i,i+1} + {}^{i}\mathbf{w}_{i+1} \times \left({}^{i}\mathbf{w}_{i+1} \times {}^{i}\mathbf{r}_{i,i+1}\right) \quad (1) $$

($\times$ is the classical vectorial product), where

$$ {}^{i}\mathbf{w}_{i+1} = {}^{i}\mathbf{w}_{i} + {}^{i}\dot{\theta}_{i+1}\mathbf{z}_{i}, \quad (2) $$

$$ {}^{i}\dot{\mathbf{w}}_{i+1} = {}^{i}\dot{\mathbf{w}}_{i} + {}^{i}\ddot{q}_{i+1}\mathbf{z}_i + {}^{i}\mathbf{w}_i \times {}^{i}\dot{q}_{i+1}\mathbf{z}_i. \tag{3} $$

With respect to the $(i+1)$th frame, we obtain

$$ \begin{aligned} {}^{i+1}\mathbf{a}_{i+1} &= {}^{i+1}\mathbf{R}_i {}^{i}\mathbf{a}_{i+1}, \qquad {}^{i+1}\mathbf{w}_{i+1} = {}^{i+1}\mathbf{R}_i \cdot {}^{i}\mathbf{w}_{i+1}, \\ {}^{i+1}\dot{\mathbf{w}}_{i+1} &= {}^{i+1}\mathbf{R}_i \cdot \dot{\mathbf{w}}_{i+1}. \end{aligned} \tag{4} $$

Then, we can express the center of mass acceleration of the $(i+1)$th link by

$$ \begin{aligned} {}^{i+1}\mathbf{a}_{ci+1} = {}^{i+1}\mathbf{a}_{i+1} &+ {}^{i+1}\dot{\mathbf{w}}_{i+1} \times {}^{i+1}\mathbf{r}_{i+1,ci+1} + {}^{i+1}\mathbf{w}_{i+1} \times \\ &\times \left({}^{i+1}\mathbf{w}_{i+1} \times {}^{i+1}\mathbf{r}_{i+1,ci+1}\right). \end{aligned} \tag{5} $$

Equations (1) through (5) are used in the first phase of our computational algorithm.

5.3. DYNAMICS OF THE LEG

Using Newton–Euler equations relative to the ith frame, we could write the two following equations:

$$ {}^{i}\mathbf{f}_{i-1,i} - {}^{i}\mathbf{f}_{i,i+1} = m_i \cdot {}^{i}\mathbf{a}_{ci}, \tag{6} $$

$$ {}^{i}\mathbf{M}_{i-1,i} - {}^{i}\mathbf{M}_{i,i+1} + {}^{i}\mathbf{r}_{i,ci} \times {}^{i}\mathbf{f}_{i,i+1} - {}^{i}\mathbf{r}_{i-1,ci} \times {}^{i}\mathbf{f}_{i-1,i} = {}^{i}\mathbf{M}_i, \tag{7} $$

where ${}^{i}\mathbf{M}_i = {}^{i}\mathbf{I}_i \cdot {}^{i}\dot{\mathbf{w}}_i + {}^{i}\mathbf{w}_i \times ({}^{i}\mathbf{I}_i \cdot {}^{i}\mathbf{w}_i)$ and ${}^{i}\mathbf{I}_i$ is the inertia matrix about the centroid attached frame which is parallel to the ith frame.

With respect to the ith frame, we obtain

$$ {}^{i-1}\mathbf{f}_{i-1,i} = {}^{i-1}\mathbf{R}_i \cdot {}^{i}\mathbf{f}_{i-1,i}, \tag{8} $$

$$ {}^{i-1}\mathbf{w}_{i-1,i} = {}^{i-1}\mathbf{R}_i \cdot {}^{i}\mathbf{M}_{i-1,i}. \tag{9} $$

Following (9), we can write

$$ \tau_i = {}^{i-1}\mathbf{M}_{i-1,i} \cdot \mathbf{k}_{i-1}. \tag{10} $$

In (6), ${}^{i}\mathbf{a}_{ci}$ has been obtained from ${}^{0}\mathbf{a}_{c0}$ where the gravitational effect has been included:

$$ {}^{0}\mathbf{a}_{c0} = {}^{0}\mathbf{a}'_{c0} - {}^{0}\mathbf{g}, \tag{11} $$

where ${}^{0}\mathbf{a}'_{c0}$ is the absolute centroidal acceleration of the 0 body expressed in the Oth frame.

5.4. IDM RESOLUTION ALGORITHM

With the previous kinematic and dynamic equations, we have written a computational algorithm of the necessary torques to be applied to the joints for a desired platform movement. This algorithm has two parts which allow us to compute the centroidal acceleration of each body from the kinematic equations of the leg for the first one (Figure 5) and the joint torques from dynamic equations for the second (Figure 6).

In this algorithm, each equation has been previously optimized with a program we have developed. It gives an explicit form of these equations where all useless terms have been eliminated (multiplications by 0 or 1 as well as additions by 0).

In our case, this program allows us to execute only half of the equations compared to the basis model (Table I). But, in order to minimize the computational time, it is always possible to see the problem form a high parallel computation point of view and to realize a parallel implementation like the computational scheme in Table II.

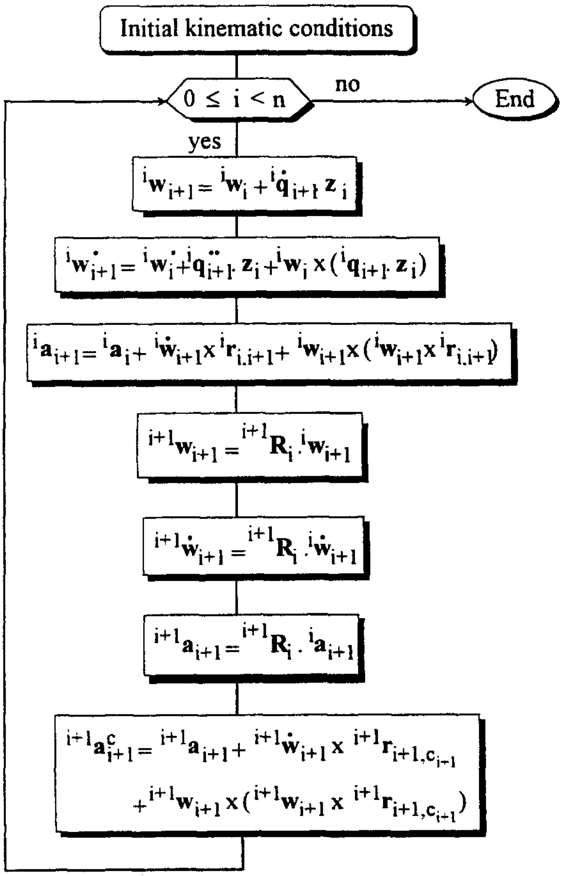


Fig. 5. Kinematic algorithm.

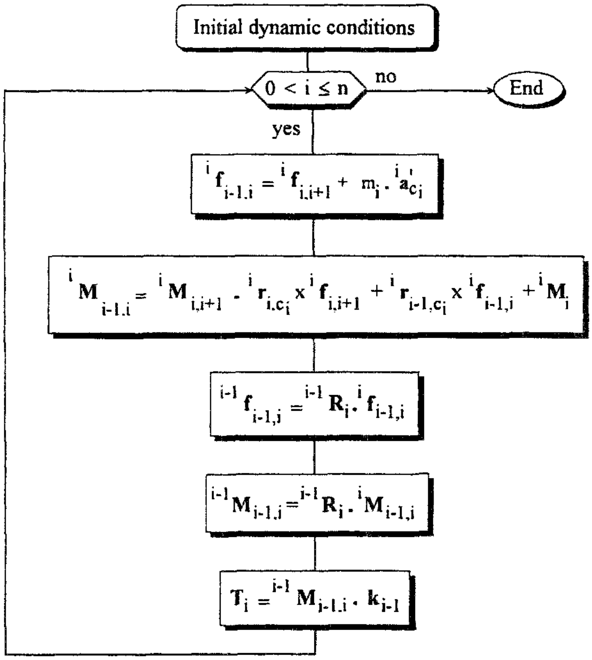


Fig. 6. Dynamic algorithm.

Table I. Number of operations

	Not optimised implementation		Optimised implementation	
	Number of additions	Number of multiplications	Number of additions	Number of multiplications
${}^{i}\mathbf{w}_{i+1}$	1	0	1	0
${}^{i}\dot{\mathbf{w}}_{i+1}$	3	2	3	2
${}^{i}\mathbf{a}_{i+1}$	18	27	12	20
${}^{i+1}\mathbf{w}_{i+1}$	5	12	2	4
${}^{i+1}\dot{\mathbf{w}}_{i+1}$	5	12	2	4
${}^{i+1}\mathbf{a}_{i+1}$	5	12	2	4
${}^{i+1}\mathbf{a}^{c}_{i+1}$	18	21	6	7
Σ	55	86	28	41
${}^{i}\mathbf{f}_{i-1,i}$	3	3	3	3
${}^{i}\mathbf{M}_{i-1,i}$	39	45	13	19
${}^{i-1}\mathbf{f}_{i-1,i}$	5	12	2	4
${}^{i-1}\mathbf{M}_{i-1,i}$	5	12	2	4
Σ	52	72	20	30
$\sum$	107	158	48	71

Table II. Parallel computation

Computation steps	Processor 1	Processor 2
1	${}^{0}\mathbf{w}_1$	...
2	${}^{1}\mathbf{w}_1$	...
3	${}^{0}\dot{\mathbf{w}}_1$	${}^{1}\mathbf{w}_2$
4	${}^{1}\dot{\mathbf{w}}_1$	${}^{2}\mathbf{w}_2$
5	${}^{0}\mathbf{a}_1$	${}^{1}\dot{\mathbf{w}}_2$
6	${}^{1}\mathbf{a}_1$	${}^{2}\dot{\mathbf{w}}_2$
7	${}^{1}\mathbf{a}_{c1}$	${}^{1}\mathbf{a}_2$
8	${}^{1}\mathbf{f}_1$	${}^{2}\mathbf{a}_2$
9	${}^{1}\mathbf{M}_1$	${}^{2}\mathbf{a}_{c2}$
10	...	${}^{2}\mathbf{f}_2$
11	${}^{2}\mathbf{f}_{1,2}$	${}^{2}\mathbf{M}_2$
12	${}^{1}\mathbf{f}_{1,2}$	...
13	${}^{2}\mathbf{M}_{1,2}$	${}^{1}\mathbf{f}_{0,1}$
14	${}^{1}\mathbf{M}_{1,2}$	${}^{0}\mathbf{f}_{0,1}$
15	...	${}^{1}\mathbf{M}_{0,1}$
16	...	${}^{0}\mathbf{M}_{0,1}$

6. Up to the LIPM

Symmetrical gait could be characterized by the following four parameters [18]:

- λ = stride length,
- f = stride frequency or inversely stride duration [T],
- β = duty factor of a foot. It is a fraction of the duration of the stride for which the leg is on the ground,
- ϕ = relative phase of a foot. It is the stage of the stride at which it is set down, expressed as a fraction of the duration of the stride following the setting down on an arbitrarily chosen reference foot.

Then, the mean speed of the platform is: $u = \lambda \cdot f$.

As it is possible to execute a gait with various values of λ and f, only β and ϕ are really intrinsic to the gait notion. Then, to facilitate the coordinator task, it is better to use normalized gaits with respect to λ and f. Moreover, to simplify the reasoning during the gait changes, these normalized gaits could be 'linearized', in other words, represented by segments of a straight line.

Thus, for the coordinator, horizontal (X) (Figure 7) and vertical (Y) (Figure 8) leg movement components become:

However, we reiterate that the leg orders are obtained from real gaits. Only the coordinator will work with reducing gaits.

With regard to the stability platform, we propose to use a principle scheme coming from studies realized in the coordination of multiple robotic mechanisms [33, 36, 43].

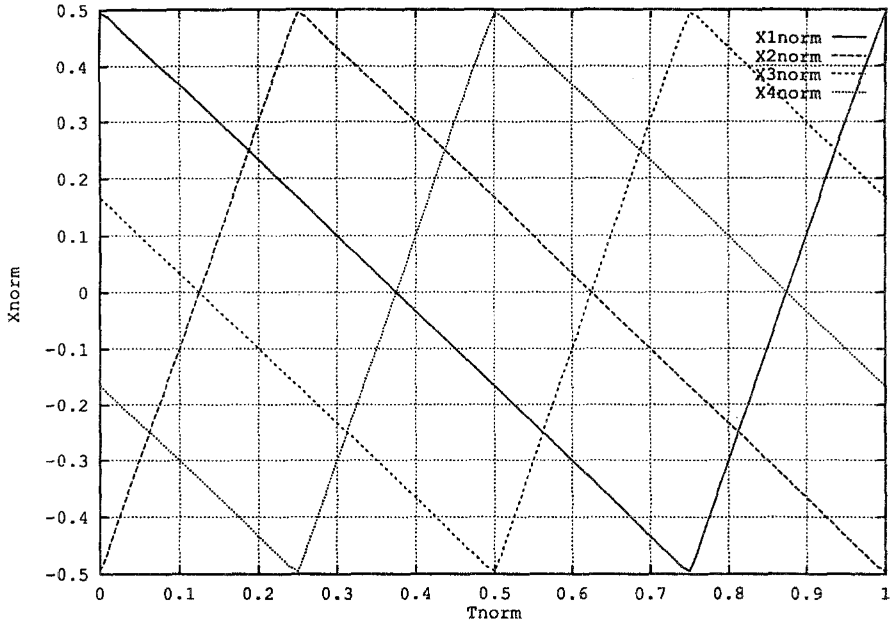


Fig. 7. Normalized horizontal gait (respect to X axis).

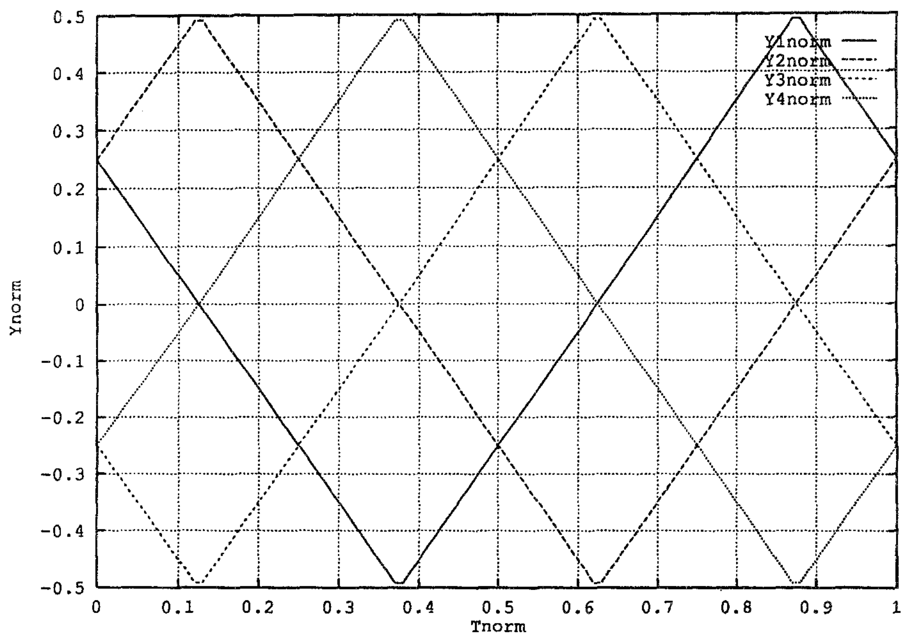


Fig. 8. Normalized vertical gait (respect to Y axis).

The goal of this control scheme is to define a set of joint torques to maintain platform equilibrium. This equilibrium is insured by controlling the absolute acceleration of the leg/platform contacts to follow the desired movement of the platform. In fact, our method is often used in the trajectory control system of robot manipulators [27, 38, 54]. Nevertheless, our control variables are not joint torques but the leg/platform contact accelerations and, consequently, the resultant forces applied on the platform

$$\begin{bmatrix} m_0 \cdot \mathbf{J}_3 & 0 \\ 0 & {}^0\mathbf{I}_0 \end{bmatrix} \cdot \begin{bmatrix} {}^0\mathbf{a}_{c0} \\ {}^0\dot{\mathbf{w}}_{c0} \end{bmatrix} + \begin{bmatrix} -m_0 \cdot {}^0\mathbf{g} \\ {}^0\mathbf{w}_{c0} \times \left({}^0\mathbf{I}_0 \cdot {}^0\mathbf{w}_{c0}\right) \end{bmatrix} = \begin{bmatrix} {}^0\mathbf{F}_0 \\ {}^0\mathbf{M}_0 \end{bmatrix}, \tag{12}$$

where

m_0 is the platform mass;
$\mathbf{J}_3$ is the 3×3 identity matrix;
${}^0\mathbf{I}_0$ is the platform inertia matrix;
${}^0\mathbf{a}_{c0}({}^0\dot{\omega}_{c0})$ is the platform absolute (angular) acceleration;
${}^0\mathbf{g}$ is the gravitational vector;
${}^0\omega_{c0}$ is the platform angular velocity;
${}^0\mathbf{F}_0$ and ${}^0\mathbf{M}_0$ are the resultant force and moment applied to the platform by the legs:

$$ {}^0\mathbf{F}_0 = \sum_{k=1}^{4} \mathbf{F}_k \quad \text{and} \quad {}^0\mathbf{M}_0 = \sum_{k=1}^{4} \mathbf{M}_k. \tag{13}$$

$\mathbf{F}_k(\mathbf{M}_k)$ is the resultant force (moment) applied to the platform by the kth leg.

$$\mathbf{F}_k = -{}^0\mathbf{f}_{0,1}^k \quad \text{and} \quad \mathbf{M}_k = -{}^0\mathbf{M}_{0,1}^k - {}^0\mathbf{r}_{c0,0} \times {}^0\mathbf{f}_{0,1}^k. \tag{14}$$

So, in a reduced form:

$$\mathbf{A} \cdot \ddot{\mathbf{X}} + \mathbf{B} = \mathbf{F}. \tag{15}$$

If, at each time, X_d represents the desired trajectory of the plarform's center of mass and X the real one, it is possible to insure the following correction:

$$\ddot{\mathbf{X}}_c(t+1) = \ddot{\mathbf{X}}_d(t+1) + \mathbf{K}_v \cdot \left(\dot{\mathbf{X}}_d(t+1) - \dot{\mathbf{X}}(t)\right) + \mathbf{K}_p \cdot \left(\mathbf{X}_d(t+1) - \mathbf{X}(t)\right), \tag{16}$$

where $\ddot{\mathbf{X}}_c(t+1)$ is the platform center of mass acceleration control input for time $t+1$, and K_p and $K_v \in R^{6\times6}$ are constant matrices which guarantee asymptotic stability.

From Equation (16), we can calculate leg/platform contact point accelerations, such as:

$$\begin{aligned} {}^0\mathbf{a}_0^k(t+1) &= {}^0\mathbf{a}_{c0}^c(t+1) + {}^0\dot{\mathbf{w}}_{c0}^c(t+1) \times {}^0\mathbf{r}_{c0,0}^k \\ &\quad + {}^0\mathbf{w}_{c0}^d(t+1) \times \left({}^0\mathbf{w}_{c0}^d(t+1) \times {}^0\mathbf{r}_{c0,0}^k\right) \end{aligned} \tag{17}$$

with

$$\ddot{\mathbf{X}}_c(t+1) = \begin{bmatrix} {}^0\mathbf{a}^c_{c0}(t+1) \\ {}^0\dot{\mathbf{w}}^c_{c0}(t+1) \end{bmatrix} \quad \text{and} \quad \dot{\mathbf{X}}_d(t+1) = \begin{bmatrix} {}^0\mathbf{v}^d_{c0}(t+1) \\ {}^0\mathbf{w}^d_{c0}(t+1) \end{bmatrix}. \tag{18}$$

If at time $t+1$ the platform center of mass acceleration control input given by (16) is applied, then this implies that the leg/platform interaction efforts become:

$$\mathbf{F}_c(t+1) = \mathbf{A} \cdot \ddot{\mathbf{X}}_c(t+1) + \mathbf{B}. \tag{19}$$

Then, the problem is to distribute properly this correction on each leg in such a way that the effect of the leg/ground interaction efforts could be traduced by a leg/platform interaction efforts resultant equal to $\mathbf{F}_c(t+1)$. As the load is shared by several kinematic chains, there is more than one solution to this problem. Using optimization techniques, it is possible to deduce from a set of possible solutions one which leads to the minimization of an objective function. Many algorithms have been developed to solve this force distribution problem (for example, [6, 16, 31, 43, ...]). In our case, we have retained the Simplex method. The methods habitually used in nonlinear programming have been discarded because, contrary to the Simplex method, the efficiency of these methods is mainly based on the adjustment of some of the determining, e.g. starting points, projection step, or penalization function. As is necessary to effect several tests to define these adjusting parameters, it would be impossible to generalize the approach. Moreover, the Simplex method uses only elementary operations, so it is quite conceivable to implement it on a basic embedded processors. So the force distribution problem could be formulated as follows:

$$\begin{aligned} &\text{Minimize: } \mathbf{F} = \mathbf{C} \cdot \mathbf{F} \\ &\text{under: } \mathbf{G} \cdot \mathbf{F} = \mathbf{E} \\ &\text{and: } \mathbf{H} \cdot \mathbf{F} \leqslant \mathbf{I} \end{aligned} \tag{20}$$

$\boldsymbol{\Phi}$ = objective function $[1 \times 1]$,
$\mathbf{C}$ = cost vector $[1 \times 6 \cdot n_c]$,
$\mathbf{F}$ = contact force/moment vector $[6 \cdot n_c \times 1]$,
$\mathbf{G}$ = matrix of the equality constraint coefficients $[n_e \times 6 \cdot n_c]$,
$\mathbf{E}$ = vector of the equality constraints values $[n_e \times 1]$,
$\mathbf{H}$ = matrix of the inequality constraint coefficients $[n_i \times 6 \cdot n_c]$,
$\mathbf{I}$ = vector of the inequality constraints values $[n_i \times 1]$,
n_c = number of contacts,
n_e = number of equality constraints,
n_i = number of inequality constraints.

Although it is not possible to define a universal objective function, the control/command scheme that we have retained to insure the stability of RALPHY's platform could be, nevertheless, represented as shown in Figure 9.

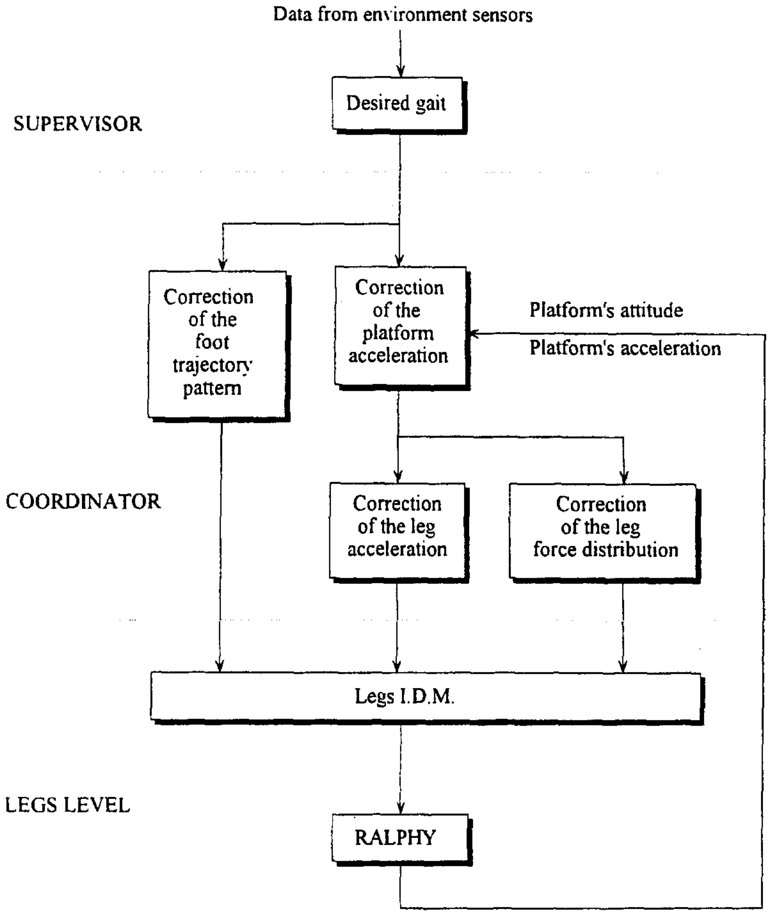


Fig. 9. Control/command scheme of RALPHY.

To illustrate the effect of this scheme, we have developed a simulation of the behavior of RALPHY's platform. Within this simulation, we have modelized the ground contact force from records made from different bipeds and quadrupeds [2, 3]. Then, applying the dynamic similarity hypothesis [4] to our quadruped robot, we have obtained our own model. Quasi-dynamic gaits have been chosen because they allow us to retain a great speed range without inducing some incompatibilities between displacement velocity and the gait used. Moreover, it is possible to start a movement with a quasi-dynamic gait when it is not the case with a dynamic one.

So, to validate the force pattern, we have made several tests where no correction has been applied to the platform. For example, using $\beta = 0.75, \Phi = 0.25, \lambda = 0.2$ m, and $T = 1$ s we have obtained the results given in Figures 10 and 11.

In this case, with only correction to the acceleration, the behavior takes on the form given in Figures 12 and 13.

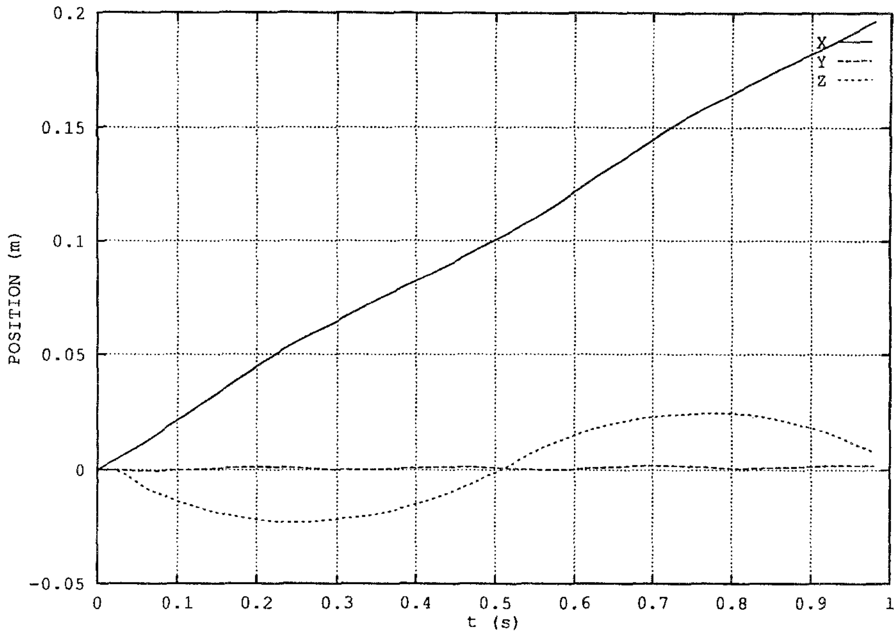


Fig. 10. Position of the platform without correction.

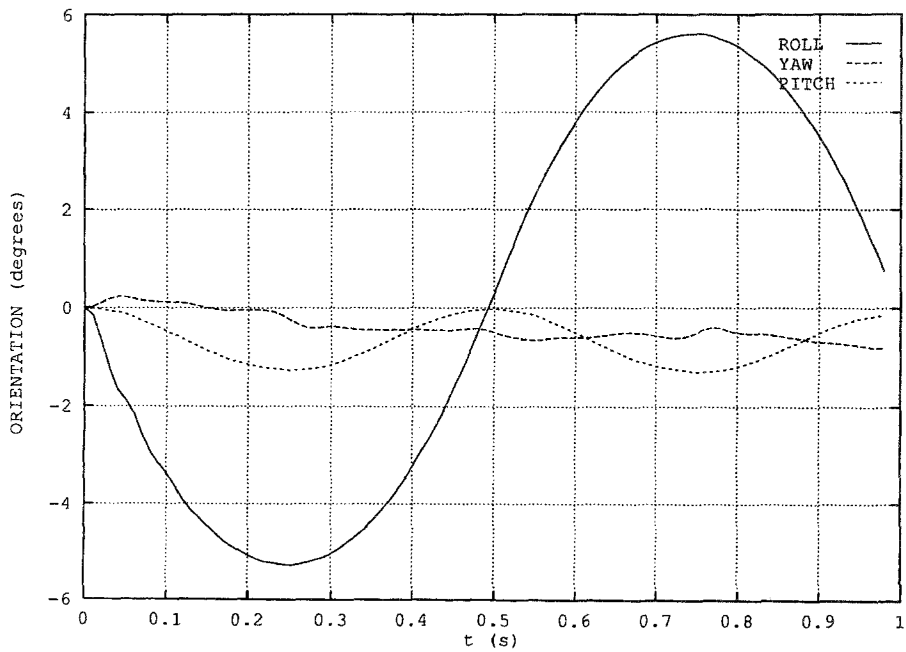


Fig. 11. Orientation of the platform without correction.

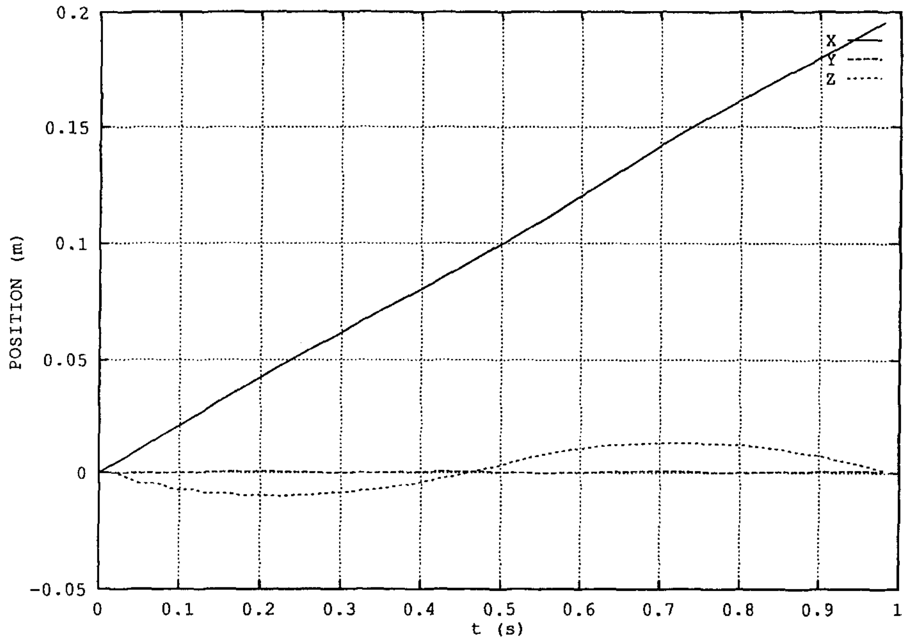


Fig. 12. Position of the platform with acceleration correction.

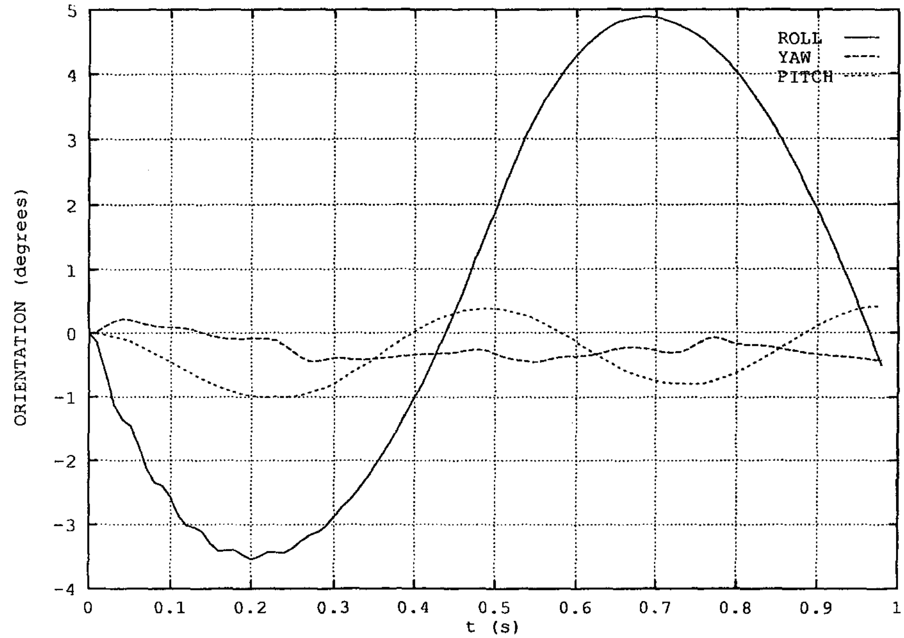


Fig. 13. Orientation of the platform with acceleration correction.

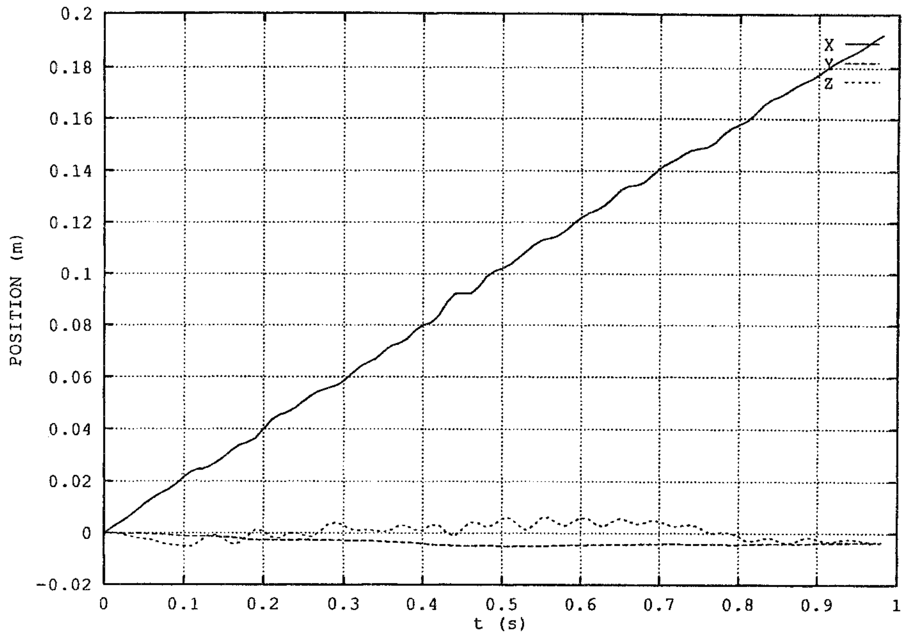


Fig. 14. Position of the platform with acceleration correction and force distribution.

The result, given in Figures 12 and 13, shows that it is necessary to reduce the angular variations of the platform's movement by a vertical force redistribution with an objective function which minimizes the gap between the vertical forces applied on each side. So, the problem becomes

$$\begin{aligned} &\text{Minimize } \left|\left(F_{1,y}+F_{2,y}\right)-\left(F_{3,y}+F_{4,y}\right)\right| \\ &\text{under } \sum_{i=1}^{4} F_{i,y}=F_{c,y} \\ &\text{and } \mathbf{F}_y \geqslant \mathbf{min} \\ &\mathbf{F}_y \leqslant \mathbf{max1} \\ &\Delta\mathbf{F}_y \leqslant \mathbf{max2} \end{aligned} \tag{21}$$

where the different values of i correspond to:

1 = left rear leg,	2 = left fore leg,
4 = right rear leg,	3 = right fore leg.

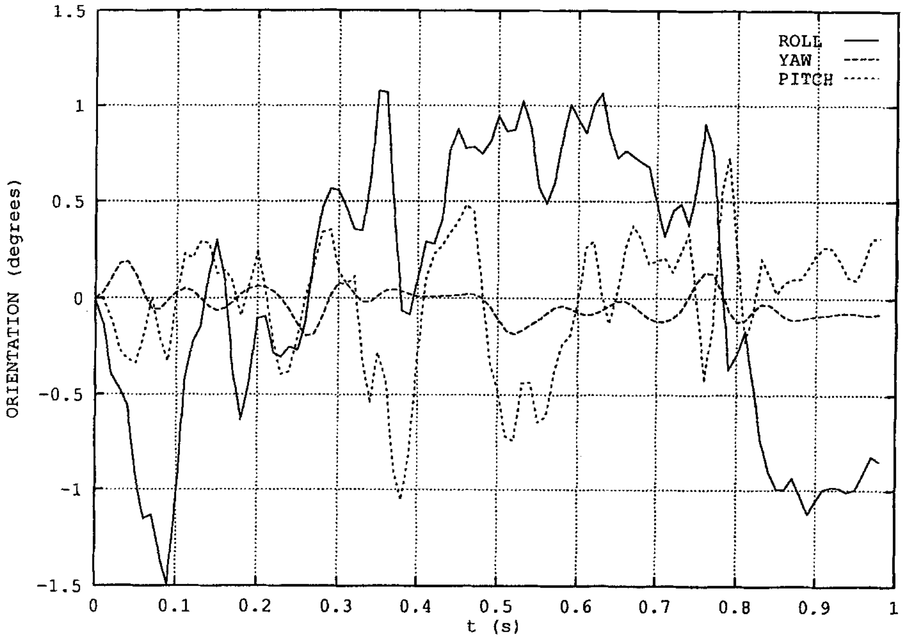


Fig. 15. Orientation of the platform with acceleration correction and force distribution.

Remarks.

- the goal of the first inequality constraint is to avoid all the solutions for which the contact forces are zero,
- the second inequality constraint allows us to exclude all the solutions for which the algorithms will converge to some configurations which involve the instability of the platform,
- the last constraint permits us to limit the gap between the retained solution over the stride cycle.

The effect of this correction is given in Figures 14 and 15.

Note that, in reality, the behavior will be more smooth as the leg are equipped with pneumatic actuators which play the role of mechanical filters.

8. Conclusion

In this paper, we have described a control/command technique for a quadruped robot destined to be used on RALPHY. The goal of this technique is to take into account both dynamical aspects and real-time notions. For that, we have divided the dynamic model of the whole robot into two parts: the Leg Inverse Dynamic Model (LIDM) and the Leg to Platform Interaction Model (LPIM). This allows us to command each leg at a low level (Leg Level) and to move the platform control to an upper level (coordinator). This control is insured by a correction of the platform acceleration and by a redistribution of the vertical force applied by the legs. To validate the control/command scheme finally presented, we have developed a simulation of the behavior of RALPHY's platform. Implementation this will the subject of future papers.

References

1. Adachi, H., Koyachi, N. and Nakano, E.: Mechanism and control of a quadruped walking robot, *IEEE Control Systems Magazine* (Oct. 1988), 14–19.
2. Alexander, R. McN. and Jayes, A.S.: Vertical movements in walking and running, *J. Zoology (London)* **185** (1978), 27–40.
3. Alexander, R. McN.: Optimum walking techniques for quadrupeds and bipeds, *J. Zoology (London)* **192** (1980), 97–117.
4. Alexander, R. McN. and Jayes, A.S.: A dynamic similarity hypothesis for the gaits of quadrupedal mammals, *J. Zoology (London)* **201** (1983), 135–152.
5. Balafoutis, C.A., Misra, P., and Patel, R.V.: A cartesian tensor approach for fast computation of manipulator dynamics, *Proc. IEEE Int. Conf. on Robotics and Automation* 1988, pp. 1348–1353.
6. Cheng, F.T. and Orin, D.E.: Efficient formulation of the force distribution equation for simple closed-chain robotic mechanisms, *IEEE Trans. Systems Man Cybernet.* **21**(1) (1991), 25–32.
7. Choi, B.S. and Song, S.M.: Fully automated obstacle crossing gaits for walking machines, *Proc. of the 1988 IEEE Int. Conf. on Robotics and Automation*, Vol. 2, April 24–29, 1988, pp. 802–807.
8. Denavit, J. and Hartenberg, R.S.: A kinematic notation for lower pair mechanism based on matrices, *Trans. ASME, J. Appl. Mech.*, Vol. 22, 1955, pp. 215–221.
9. Fontaine, J.G.: *Commande directe assistée: Principe et application au contrôle d'actionneurs électriques et pneumatiques pour la robotique*, Thèse de Doctorat de l'Université d'Orsay (Paris 11), Paris, 1987.
10. Franck, A.A. and Ghee, R.B.: Some consideration relating to the design of autopilots for legged vehicules, *J. Terramechanics* **6**(1) (1969), 23–35.
11. Freeman, P.S. and Orin, D.E.: Efficient dynamic sumulation of a quadruped using a decoupled tree-structure approach, *The Int. J. of Robotics Research* **10**(6) (1991), 619–627.
12. El Gamah, H., Amaral, P.F.S., Fontaine, J.G., Villard, C. and Gorce, P.: A low supervisor for a legged robot, *Proc. of the 23rd Int. Symp. on Industrial Robots, ISIR'92*, Barcelona, Spain, 1992.
13. McGeer, T.: Passive dynamic walking, *Internat. J. Robotics Res.* **9**(2) (1990), 62–82.
14. McGhee, R.B. and Pai, A.L.: An approach to computer control for legged vehicules, *J. Terramechanics* **11**.
15. Glower, J.S. and Ozguner, U.: Control of a quadruped trot, *Proc. of the 1986 IEEE Int. Conf. on Robotics and Automation*, Vol. 3, April 7–10, pp. 1496–1501.
16. Gorce, P., Villard, C. and Fontaine, J.G.: Grasping, coordination and optimal force distribution in multifingered mechanism, *Robotica*, 1993 (in press).
17. Gorinevsky, D.M. and Shneider, Yu.A.: Force control in locomotion of legged vehicles over rigid and soft surfaces, *Internat. J. Robotics Res.* **9**(2) (1990), 4–23.
18. Hamdam, S., Fontaine, J.G. and Picard, M.: Autonomous legged Robots: Technics for gait transitions, *Symp. on Flexible Automation*, Kyoto, Japan, 1990, pp. 125–128.

19. Hashimoto, K. and Kimura, H.: A new parallel algorithm for inverse dynamics, *Internat. J. Robotics Res.* **8**(1) (1989), 63–76.
20. Hashimoto, K., Ohashi, K. and Kimura, H.: An implementation of a parallel algorithm for real-time model-based control on a network of microprocessors, *Internat. J. Robotics Res.* **9**(6) (1990), 7–47.
21. He, X. and Goldenberg, A.A.: An algorithm for effcient computation of dynamics of robotic manipulators, *J. Robotic Systems* **7**(5) (1990), 689–702.
22. Hirose, S.: A study of design and control of a quadruped walking vehicle, *Internat. J. Robotics Res.* **3**(2) (1984), 113–133.
23. Hirose, S. and Kunieda, O.: Generalized standard foot trajectory for a quadruped walking vehicle, *Internat. J. Robotics Res.* **10**(1) (1991), 3–11.
24. Hollerbach, J.M.: An itérative lagrangian formulation of manipulators dynamics and a comparative study of dynamics formulation complexity, *IEEE Trans. Systems Man Cybernet.* **SMC-10**(11) (1980), 730–736.
25. Kane, J.R. and Levinson, D.A.: The use Kane's dynamical equations in Robotic, *Internat. J. Robotics Res.* **2**(3) (1983), 3–21.
26. Kasahara, H. and Narita, S.: Parallel processing of robot-arm control computation on a multimicroprocessor, *IEEE J. Robotics and Automation* **Ra-1**(2) (1985), 104–113.
27. Khatib, O.: The operational space formulation in robot manipulator control, *Proc. of the 15th Int. Symp. on Industrial Robots*, 1985, pp. 165–172.
28. Kimura, H., Shimoyama, I. and Miura, H.: Dynamics in the dynamic walk of a quadruped robot, *Advanced Robotics* **4**(3) (1990), 283–301.
29. Klein, C.A. and Briggs, R.L.: Use of active compliance in the control of legged vehicles, *IEEE Trans. Systems Man Cybernet.* **SMC-10**(7) (1980), 393–400.
30. Klein, C.A., Olson, K.W. and Pugh, D.R.: Use of force and attitude sensors for locomotion of a legged vehicle over irregular terrain, *Internat. J. Robotics Res.* **2**(2) (1983), 3–17.
31. Klein, C.A. and Kittivatcharapong, S.: Optimal force distribution for the legs of a walking machine with friction cone constraints, *IEEE Trans. on Robotics and Automation* **6**(1) (1990), 73–85.
32. Koditschek, D.E. and Bühler, M.: Analysis of a simplified hopping robot, *Internat. J. Robotics Res.* **10**(6) (1991), 587–605.
33. Kosuge, K., Koga, N., Furuta, K. and Nosaki, K.: Coordinated motion control of robot arm based on virtual internal model, *Proc. 1989 IEEE Internat. Conf. on Robotics and Automation*, 1989, pp. 1097–1102.
34. Lee, T.T. and Shih, C.L.: Real time computer control of a quadruped walking robot, *Trans. ASME, J. Dynamic Systems, Measurement, and Control* **108** (1986), 346–353.
35. Lee, T.T. and Shih, C.L.: A study of a gait control of a quadruped walking vehicle, *IEEE J. of Robotics and Automation* **RA-2**(2) (1986), 61–69.
36. Li, Z., Hsu, P. and Sastry, S.: Grasping and coordinated manipulation by a multifingered Robot Hand, *Internat. J. Robotics Res.* **8**(8) (August. 1989), 33–50.
37. Lilly, K.W. and Orin, D.E.: Efficient dynamic simulation for multiple chain robotic systems, *3rd Annual Conf. on Aerospace Computational Control* 1989, 73–87.
38. Luh, J.Y.S., Walker, M.W. and Paul, R.C.P.: On-line computational scheme for mechanical manipulators, *Trans. ASME, J. Dynamic Systems, Measurement, and Control* **102**(2) (1980), 69–76.
39. Luh, J.Y.S. and Lin, C.S.: Scheduling of parallel computation for computer controlled mechanical manipulators, *IEEE Trans. Systems Man Cybernet.* **SMC-12**(2) (1982), 214–234.
40. Miura, H. and Shimoyama, I.: Dynamic Walk of a Biped, *Internat. J. Robotics Res.* **3** (1984), 60–74.

41. Mladenova, C.: Mathematical modelling and control of manipulator systems, *Robotics and Computer Integrated Manufacturing* **18**(4) (1991), 233–242.
42. Mosher, R.S.: Exploring the potential of quadruped, *Int. Automotive Engineering Conf.*, SAE paper No 690191, Detroit, Michigan, 1969.
43. Nakamura, Y., Nagai, K. and Yoshikawa, T.: Dynamics and stability in coordination of multiple robotic mechanisms, *Internat. J. Robotics Res.* **8**(2) (1989), 44–61.
44. Neuman, C.P. and Murray, J.: Computational robot dynamics: fondations and applications, *J. Robotics Systems* **2**(4) (1985), 425–452.
45. Orin, D.E.: Supervisor control of multi-legged robot, *Internat. J. Robotics Res.* **1**(1) (1982), 71–91.
46. Ouezdou, F.B., Pasqui, V., Bidaud, P. and Guinot, J.C.: Kinematic and dynamic analysis of legged robots, *10th CISM-IFTOMN Symp. on Theory and Practice of Robots and Manipulators, RoManSy'90*, Crasow, Poland, 1990.
47. Raibert, M.H.: Hopping in legged systems modeling and simulation for the 2D ones-legged case, *IEEE Trans. Systems Man Cybernet.* **14**(3) (1984), 451–463.
48. Raibert, M.H., Chepponis, M. and Brown Jr., H.B.: Running on four legged as though they were one, *IEEE J. of Robotics and Automation* **RA-2**(2) (1986), 70–82.
49. Shih, L., Franck, A.A. and Ravani, B.: Dynamic simulation of legged machines using a compliant joint model, *Internat. J. Robotics Res.* **6**(4) (1987), 33–46.
50. Silver, W.M.: On the equivalence of lagrangian and Newton–Euler dynamics for manipulators, *Internat. J. Robotics Res.* **1**(2) (1982), 60–70.
51. Sutherland, I.E.: A walking robot, *Pittsburg PA: the Marcian Chronicles*, 1983.
52. Vakakis, A.F., Burdick, J.W. and Caughey, T.K.: An 'interesting' strange attractor in the dynamics of a hopping robot, *Internat. J. Robotics Res.* **10**(6) (1991), 606–618.
53. Wong, H.C. and Orin, D.E.: Reflex control of the prototype leg during contact and slippage, *IEEE Internat. Conf. on Robotics and Automation* 1988, pp. 808–813.
54. Yoshikawa, T.: Dynamic hybrid position/force control of robot manipulators description of hard constraints and calculation of joint driving force, *IEEE Internat. Conf. on Robotics and Automation*, 1986, pp. 1393–1399.
55. Zheng, Y.F. and Shen, J.: Gait synthesis for the SD-2 Biped Robot to climb sloping surface, *IEEE Trans on Robotics and Automation* **6**(1) (February 1990), 86–96.
56. Zomaya, A.Y. and Morris, A.S.: Dynamic simulation and modeling of robot manipulators using parallel architectures, *Internat. J. Robotics Res.* **6**(3) (1991), 129–139.