

# Modeling and control of biped robot dynamics

S. Caux and R. Zapata

Laboratoire d'Informatique, de Robotique et de Microélectronique, LIRMM-UM CNRS C5560-Université de Montpellier II, 161 rue Ada, 34392 Montpellier cedex 5 (France)

E-mail: caux@lirmm.fr Web: <http://www.lirmm.fr/~caux>

(Received in Final Form: November 14, 1998)

## SUMMARY

This paper addresses the problem of modeling biped dynamics and the use of such models for the control of walking, running and jumping robots. We describe two approaches to dynamic modeling: the basic Lagrange approach and the non-regular dynamic approach. The new non-regular dynamic approach takes into account discontinuities due to rigid contact between punctual feet and the ground without computing the exact impact time. The contact is close to the physical situation given by non-linear laws (impenetrability, non-smooth contact and real friction cone). Contact dynamics can be well managed with an accurate dynamic model that respects energy consistency during all the phases encountered during a step (0, 1 or 2 contacts). With this model, we can first study the equilibrium of a biped standing on one foot by a linearisation method. In the second stage, the unified modeled equation is used to establish a general control frame based on non-regular dynamical decoupling. A comparison is made and some simulation results are given with a two degree of freedom planar biped robot.

**KEYWORDS:** Biped robot; Contact dynamics; Constrained dynamics; Linearization; Non-linear decoupling.

## 1. INTRODUCTION

For a long time, many research groups have tried to adapt biological mechanisms to copy animal or human walking (e.g. mechanical horse of L. Rygg, 1893). Legged locomotion has many advantages like obstacle avoidance capabilities and the possibility of discontinuous contacts with the ground (allowing the robot to step over obstacles and climb stairs).<sup>1</sup> A small overall dimension makes biped robots very useful in constrained environments and all-terrain environments (outdoor, forest, industrial environment, etc).

Numerous prototypes have been made in the last fifteen years, but these studies have been based on dynamic modeling of behavior during the different phases encountered in walking or running. These models are generally simplified and linearized.<sup>2–4</sup> The simplifications are justified because the robots walk slowly ( $< 1$  m/s), this is quasi-dynamic walk. This means that the robot always has one foot (or more) on the ground and keeps its center of gravity in the support polygon formed by its generally very large feet.<sup>5,6</sup> Impact and slippery phenomena are often neglected

and the models used in simulations or for control purposes are limited to dynamic modeling, which is very popular in robotic fields. The switch between two different modes occurring during a step (left stance phase  $\rightarrow$  right stance phase etc.) are computed as a circular permutation of the joint vector coordinates.<sup>7</sup> A model thus corresponds to each phase and the transitions have only been estimated in few studies by adding another set of equations to take this impulsive effect into account. Lagrange's equation under its impulsive form, or zero landing velocity, is a classical assumption made to solve the transition problem.<sup>3,6</sup> But computing the exact time of impact is time consuming or impossible to solve for simulation software. Here we try to work with a fixed integration step, regardless of the contact and use this modeling to obtain a fixed sample control scheme.

In the biped research field, few papers have dealt with high speed biped robots, except for Hodgins hopper derived from the famous C.M.U. hopper.<sup>8</sup> In general, in order to increase the speed of forward displacement, a biped must run with a characteristic ballistic flight during which all the feet leave the ground. In the case of the C.M.U. hopper, the knee joints act as springs and the robot jumps from one leg to the other to reach a speed of around 4 m/s. A state machine describes the state of the robot to detect each phase and apply an *adequate* control law using a P.I.D. controller.

Here, we have tried to model a running biped throughout its displacement. Modeling and identification of the main parameters is essential.<sup>9</sup> The model we obtain manages the different phases and constraints encountered during the run. In case of running, impacts are non-linear phenomena of paramount importance to balance the robot, and we propose to design a model that includes contact dynamics.<sup>10</sup> This theory is derived from the mechanics field for rigid body contact modeling.<sup>11,12</sup>

In the first part of this paper (paragraph 2), we present Lagrange's approach for modeling the constraint dynamics of a mechanism submitted to various contacts with the ground. Then we present our equations of motion under unified formalism, including changes of modes during walking, running or jumping. Finally, we introduce several notions issued from non-regular dynamics theory to compute impact forces in every case. We discuss and compare the results obtained with regular and non-regular dynamics applied to a simulated planar biped robot.

The second part (paragraph 3), presents different control schemes concerning, the equilibrium of the robot near one

of its natural unstable equilibrium position (robot on one foot), and a general control scheme to be applied throughout the motion of the biped. The non-regular dynamic model is presented before using the classical non-linear decoupling control scheme. Simulation results are presented to illustrate our specific control scheme applied to a robot interacting with its environment.

**2. DYNAMIC MODELING OF LEGGED MACHINES**

*2.1. Lagrange's equations of motion*

The motion of a moving non-holonomic or scleronomous structure, described by  $n$  generalized coordinates  $q^i$ , can be derived by using Lagrange's equations of motion:

$$\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{q}^i} \right) - \frac{\partial L}{\partial q^i} = \Gamma_i - \sum_{j=1}^{j=m} \lambda_j c_{ji} \tag{1}$$

where:

- $L$  is the Lagrangian of the system,
- $\Gamma_i$  the  $i^{th}$  generalized force,
- $\lambda_j$  the  $j^{th}$  Lagrange multiplier.

The  $j^{th}$  constraint can be written as a linear combination of the generalized velocities  $\dot{q}^i$ :

$$\sum_{i=0}^{i=n} c_{ji} \dot{q}^i = 0 \tag{2}$$

For a legged robot,  $\lambda_j$  represents a force constraint associated to a contact constraint. For instance, in the case of a planar robot, if one leg is on the ground, two contact forces appear (horizontal and vertical forces), due to the appearance of two constraints (horizontal and vertical positions of the foot are both determined by the ground and

by the generalized coordinates.<sup>13</sup>

The classical set of dynamic equations of motion can be derived from Lagrange's equations. The direct dynamical model can be written:

$$A(q)\ddot{q} = \Gamma - B(q, \dot{q}) + C^T(q)\lambda \tag{3}$$

where:

- $A$  is the inertia matrix
- $B$  the Coriolis-Centrifugal-Gravity matrix
- $q$  the  $n$ -dimensional vector of generalized coordinates
- $C(q, t) = \{c_{ji}\}$  the  $m \times n$ -dimensional constraint matrix verifying  $C(q)\dot{q} = 0$ . The notation  $C^T$  stands for 'C transpose'.
- $\Gamma$  the  $n$ -dimensional vector of generalized forces.

In general, for legged machines, the vector  $q$  is made up of articular coordinates as well as cartesian coordinates and only the first ones are actuated. This implies that the vector  $\Gamma$  has as many zeros as the number of these cartesian coordinates. For the planar biped robot shown in Figure 1, the dynamics are 5<sup>th</sup>-dimensional, the generalized coordinates  $q = [x \ z \ \theta \ q_1 \ q_2]^T$  and  $\Gamma = [0 \ 0 \ 0 \ C_1 \ C_2]^T$ .

For control purposes, the inverse dynamical model can be written:

$$\Gamma = A(q)\ddot{q} + B(q, \dot{q}) - C^T(q)\lambda \tag{4}$$

This equation has as many expressions as the number of modes encountered during motion (zero, one or two legs on the ground changes the number of constraints 0, 2 or 3, as shown in Figure 1). These modes represent different parts of the configuration space.

Depending on the relations between the robot and its environment, we must select the corresponding set of dynamic equations:

- For a free robot we have a 5-dimensional generalized coordinate vector and no Lagrange's multiplier.
- With one contact, the normal and tangential forces ( $\mathfrak{R} = [\mathfrak{R}_N \ \mathfrak{R}_T]$ ) due to the punctual contact appears through 2 Lagrange multipliers and the generalized

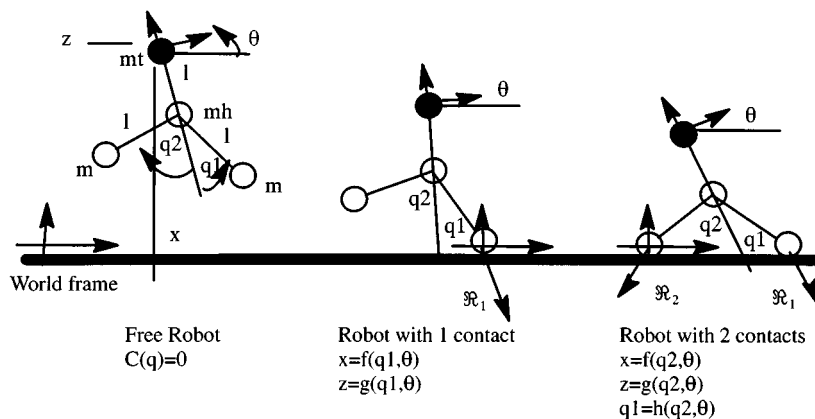


Fig. 1. The different phases encountered: To each phase corresponds an expression of the constraints, changing the dimensions of the matrix  $C(q)$  and of course the dynamics.

Where:

- $q_1, q_2$  are the angles of leg 1 and 2 referred to the body.
- $l$  is the length of the different segments, and  $m, mt, mh$  the different masses.
- $(x, z, \theta)$  are the Cartesian coordinates of the head of the robot expressed in the world frame.
- $\mathfrak{R} = [\mathfrak{R}_N \ \mathfrak{R}_T]$  the local normal and tangential reaction forces vector due to contact.

coordinates become 3-dimensional.

- With both feet in contact, 4 forces appear ( $\mathfrak{R}_{N1}$   $\mathfrak{R}_{T1}$   $\mathfrak{R}_{N2}$   $\mathfrak{R}_{T2}$ ), but only 3 independent Lagrange multipliers describe the constraints, and only 2 generalized coordinates are needed to describe the robot.

**First approach of a unified solution.** To come up with a homogeneous formalization of a motion (for walking, running or jumping tasks), it is necessary to take all these expressions into account in the same equation by adding exteroceptive information to equation (4) to obtain:

$$\Gamma = A(q)\ddot{q} + B(q, \dot{q}) - C^T(q, \mu)\lambda(\mu) \quad (5)$$

where  $\mu$  is a ‘‘bifurcation’’ parameter characterizing the Robot/Environment interaction and therefore, the mode the configuration vector belongs to.  $\mu$  has four possible values corresponding to the four different Matrices  $C$  describing the four different phases (0 contact, 1 contact on left leg, one contact on right leg, double contact). These values are given by sensors on each foot to detect the different contacts.

Accurate modeling is of paramount importance because:

- for simulation, the generalized forces  $\Gamma$  are given and we need to compute the accelerations  $\ddot{q}$ .
- to control a real robot, it is necessary to compute the state-feedback function  $\Gamma(q)$ .

Here,  $\ddot{q}$  is derived from the state vector  $[q \ \dot{q}]$  (or measured) and contact forces are either measured or computed by using equation (5) written for a model of the robot (the computed solution can sometimes be easier) but we note that control must be computed with a fixed sample time.

The transition from one mode to another (driven by  $\mu$ ) can be obtained by re-initializing the state vector  $[q \ \dot{q}]$  at the exact time of this switch. In the case of legged machines, these changes of modes represent discontinuities of equation (5) appearing between sample times. Therefore, these switches should generate acceleration changes during the sample interval just after contact (which is impossible, because the exact time of contact is unknown). In order to solve this problem, we can either change the acceleration at the sample time after the impact or reduce the sample interval before the impact. In the first case, a modelling error occurs in the position or acceleration computation. In the second case, the process has to be reiterated several times and becomes time consuming and unusable for control.

Hence, the basic Lagrange’s equations cannot take the impulsion phases into account due to the switches between modes (Figure 2).

### 2.2. Non-regular dynamics

Another approach to impulsion phases management involves rewriting the dynamical equations of motion in an implicit form [11], with only velocities considered:

$$A(q[i+1]) \frac{\dot{q}[i+1] - \dot{q}[i]}{h} = F + R[i+1] \quad (6)$$

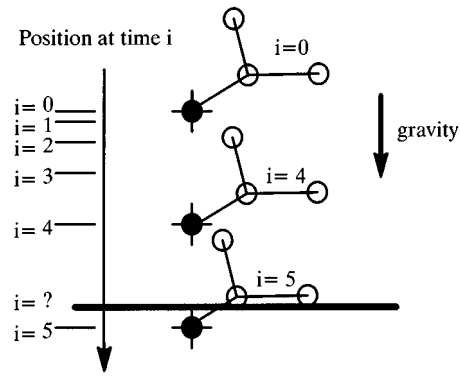


Fig. 2. A falling planar biped robot.

where:

- $q[i+1]$  is the generalized coordinate vector at time  $i+1$ .
- $h = t[i+1] - t[i]$  is the sample interval.
- $R$  the sum of all generalized forces due to contact.
- the force term is:

$$F = \Gamma[i+1] - B(q[i+1], \dot{q}[i+1]) \quad (7)$$

Equation (6) is similar to equation (3) but computed at time  $i+1$  and uses a percussional computation of  $\ddot{q}$  between the 2 critical samples  $i$  and  $i+1$  (NB: it is not an assumption of an integration method but an assumption on the left and right limits of  $\ddot{q}$  at impact). Equation (6) has 3 unknowns  $q[i+1]$ ,  $\dot{q}[i+1]$  and  $R[i+1]$  depends on the ‘initial’ condition  $\dot{q}[i]$  and can be expressed:

$$P(u, v) = 0 = A(u)(v - \dot{q}[i]) - h(F + R[i+1]) \quad (8)$$

where  $u, v$  respectively stand for  $q[i+1]$  and  $\dot{q}[i+1]$  for the sake of simplicity.

This approach uses the iterative Newton-Raphson method of resolution between  $i$  and  $i+1$ , which provides the solution of (8) as the limit of the sequence:

$$v_{k+1} = v_k - \frac{\partial P^{-1}}{\partial v}(u_k, v_k) \times P(u_k, v_k) \quad (9)$$

where  $v_k$  denotes the  $k^{th}$  iteration of  $v$  (so  $\dot{q}[i+1]$ ) and  $u_k$  the  $k^{th}$  iteration of  $u$  (so  $q[i+1]$ ). At each iteration, the generalized coordinate is derived from the velocity by an implicit integration.

$$u_{k+1} = u_k + h v_{k+1} \text{ so } u_{k+1} = f(v_{k+1}) \quad (10)$$

**Detail of the non-regular algorithm.** The 6-step algorithm works as follows:

- (i) For the unforced system ( $R=0$ ), we seek the limit of the sequence:

$$v_{k+1}^f = v_k^f - W_k \times (A_k(v_k^f - \dot{q}[i]) - hF_k) \quad (11)$$

where  $f$  stands for ‘free’,  $A_k$  denotes the  $k^{th}$  iteration of  $A(u)$  and where

$$W_k = \left[ A_k - h \frac{\partial F_k}{\partial \dot{q}} - h^2 \frac{\partial F_k}{\partial q} \right]^{-1} = \frac{\partial P(u, v)^{-1}}{\partial v} \quad (12)$$

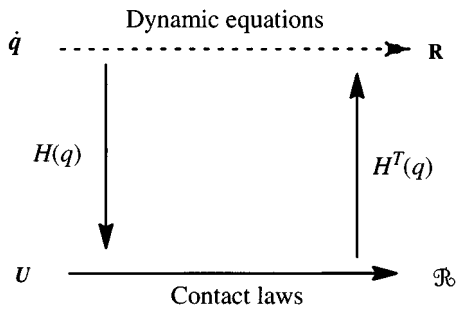


Fig. 3. Local ( $\mathfrak{R} U$ ) and generalized ( $R \dot{q}$ ) variables relations.

□ (ii) The generalized free velocity at time  $i + 1$  is the limit of this sequence:  $\nu^f = \lim_{k \rightarrow \infty} \nu_k^f$ . The free generalized coordinate is also derived from this velocity  $u^f = q[i] + h\nu^f$ .

□ (iii) This coordinate vector determines whether or not there is an impact between  $i$  and  $i + 1$ . This determination involves computing the direct kinematic model  $H(q)$  of the robot. The matrix  $H$  relates the generalized velocities  $\dot{q}$  to the velocities  $U$  of all candidates for contact (feet for instance) through a relation  $U = H(q)\dot{q}$ .

Using this matrix we can dynamically manage the generalized coordinates and the contact laws at each iteration step, which allows us to define another way to compute the dynamics and to take impact phenomena into account, as shown in Figure 3.

We must now define a frame for each contact point (Figure 4). Velocity determination in this frame consists of computing the direct kinematic model  $H(q)$  of the robot.

In each contact frame, we can thus express the two exact physical laws describing relations between the relative forces and velocities. By this computation, we obtain the generalized forces  $R$  respecting non-linear contact laws.

□ (iv) If there is no contact, the algorithm increments  $i$  and goes to step (i). In this case, equation (11) converges on one step like usual integration methods. If contacts occur, the method implies that:

$$\nu_{k+1} = \nu_{k+1}^f + h W_k R_{k+1} \tag{13}$$

The multiplication of both sides of equation (13) by the jacobian matrix  $H(q_k)$  gives:

$$U_{k+1} = U_{k+1}^f + h H(u_k) W_k H^T(u_k) \mathfrak{R}_{k+1} \tag{14}$$

where  $\mathfrak{R}_{k+1} = H^{-T}(u_k) \times R_{k+1}$  are the contact forces.

As before, we seek the limit of the sequence  $U_{k+1}$ , but this

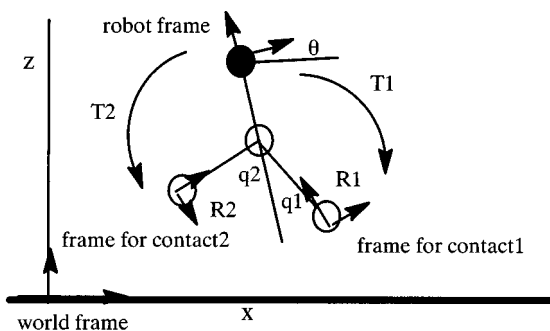


Fig. 4. Location of the different frames.

time we have to first compute an approximation of the term  $\mathfrak{R}_{k+1}$ . This latter vector verifies unilateral and/or bilateral constraints in order to respect Coulomb's laws of contacts and Signorini's laws (impenetrability of the ground and non-attraction by the ground).

□ (v) If the relative velocity  $U_{k+1}$  is zero after contact, equation (14) could be solved by using the iterative Gauss-Seidel's method. In this case, the convergence of the algorithm would be tested by  $\|\mathfrak{R}_{k+1}^{m+1} - \mathfrak{R}_{k+1}^m\| \leq \epsilon$ , where  $\mathfrak{R}_{k+1}^m$  is the  $m^{th}$  estimation of  $\mathfrak{R}_{k+1}$ . Unfortunately, the velocities  $U_{k+1}$  are not zero. Fortunately, equation (14) represents a hyperplane in space ( $\mathfrak{R} U$ ) with the given orientation  $hH(u_k)W_kH^T(u_k)$ . As we are only interested in estimating the reactions  $\mathfrak{R}_{k+1}$ , we have to know the intersection of this hyperplane with the  $\mathfrak{R}$ -subspace characterized by the equation  $U_{k+1} = 0$ . Therefore, at step  $m$ ,  $\mathfrak{R}_{k+1}^m$  is computed by Gauss-Seidel's algorithm and modified in order to respect Coulomb and Signorini's laws. This time the convergence of the algorithm has to be tested by  $\|U_{k+1}^{m+1} - U_{k+1}^m\| \leq \epsilon$  recomputed from equation (14).

In order to illustrate how the reactions are computed, let us assume the very simple example of a planar contact of one point ( $\mathfrak{R} = [\mathfrak{R}_N \mathfrak{R}_T]$  and  $U = [U_N U_T]$ ). The space ( $\mathfrak{R} U$ ) is 4-dimensional.

Once  $\mathfrak{R}_N$  is chosen ( $\mathfrak{R}_{k+1,N}^m$  of the  $m^{th}$  iteration), Coulomb's laws of contacts are applied:

$$|\mathfrak{R}_T| < \xi \mathfrak{R}_N \tag{15}$$

Equation (14) represents a plane in 3-dimensional space ( $\mathfrak{R}_N \mathfrak{R}_T U_T$ ). Its intersection with ( $U_T \mathfrak{R}_T$ ) is a line with a given slope.

□ (vi) If  $\forall m \geq M$ ,  $U_{k+1}^m$  remains stable  $\mathfrak{R}_{k+1}$  is correctly set (equation (14) is verified) and  $R_{k+1} = H^T(u_k) \mathfrak{R}_{k+1}$  can be computed. Hence, the limit of equation (13) can be found and  $q[i + 1]$ ,  $\dot{q}[i + 1]$  and  $R[i + 1]$  obtained.

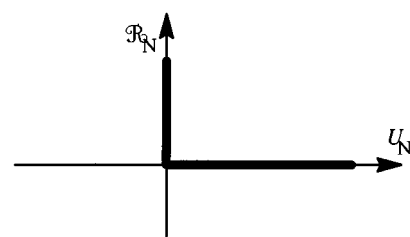


Fig. 5. Signorini's graph relating the normal reaction to the normal velocity.

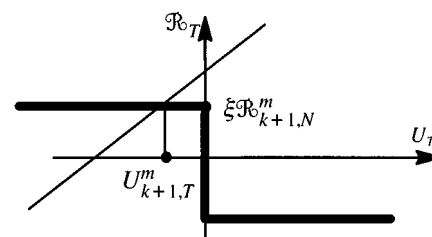


Fig. 6. Coulomb's law of friction.

2.3. Comparisons of explicit and implicit algorithms

To compare both algorithms, we propose two brief schemes (Figure 7, Figure 8). We can note the similarities and differences between these to simulation methods.

In the robotic field, mechanical systems are generally well modelled with masses and damped-springs. The equations are expressed under an explicit form. Lagrange dynamics can be solved classically (Figure 7).

With knowledge of the state vector of the previous integration step  $i$ , we compute the coordinates at time  $i+dt$ . We compute accelerations with the inverse dynamics and then integrate two times to obtain the corresponding velocities and positions at time  $dt$ . We must verify that the solutions obtained by testing the foreseen positions of points of contact with the position of the environment are in the physical space. This position must respect the constraints of impenetrability. If the positions are wrong (contact has been encountered) the estimation is wrong, and we must decrease the iteration step  $dt$  to resolve the dynamics and approach the impact time in order to be able to accurately switch the model and obtain a smaller error. We must therefore make an over-sampling to accurately manage discontinuities due to contact.

The implicit method (Figure 8) has a constant integration step (the estimation given is the state at time  $i+h$ ) but to

manage the contact laws we must make several estimations of this state (stage 2) in order to be energetically correct.<sup>14</sup> We need constant sampling time for control of the actuators. It is of paramount importance to have a model which is able to deliver a result at constant sampling time, to use its information about the behavior in a control scheme.<sup>15</sup>

2.4. Simulation results

We compared these two approaches on a simulated biped planar robot with Matlab®. This robot is described in Figure 9. The generalized coordinates are the position and orientation of the head ( $x z \theta$ ) and the angles between the body and the two legs ( $q_1 q_2$ ).

The robot is launched with a horizontal velocity  $V_x = m/s$ . The initial configuration vector is: ( $x_0 = 0m z_0 = 1.5m \theta_0 = -0.5rd q_{10} = 0.9rd q_{20} = -1rd$ ). The generalized forces are the two torques  $\Gamma_1$  and  $\Gamma_2$ , which servocontrol the legs to their initial configuration (no leg movement). We then show the results obtained with non-regular modeling, which is the only physically realistic approach with a fixed integration step.

**Detail of the Cartesian coordinates.** The Figure 11 represents the behavior of the cartesian coordinates. We can

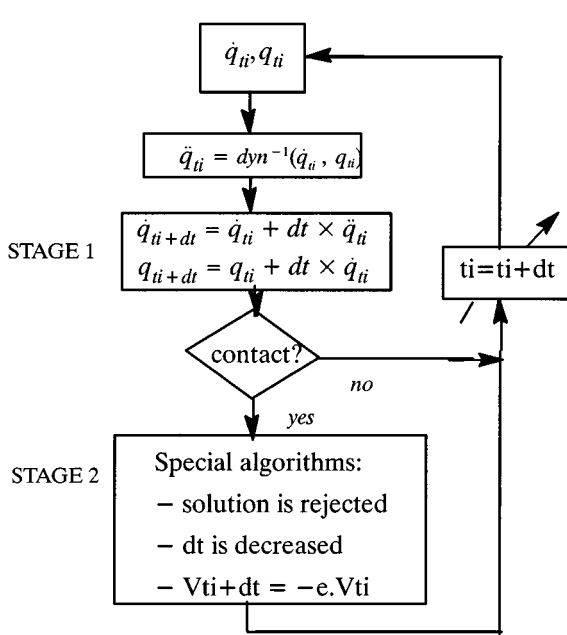


Fig. 7. Explicit algorithm for Lagrange.

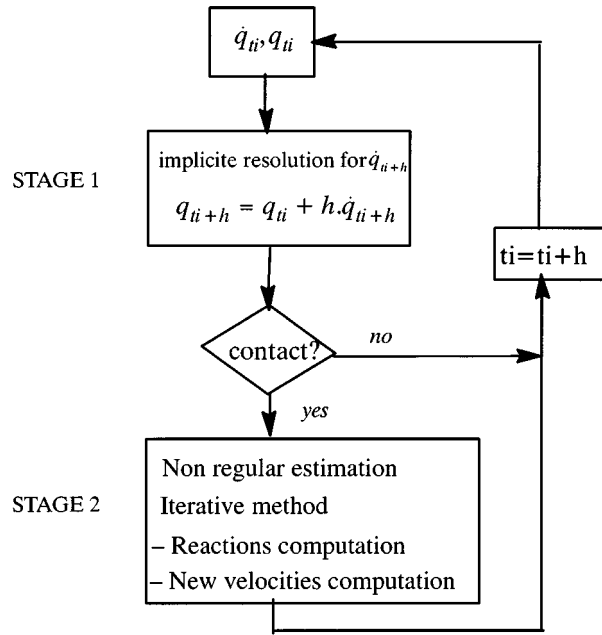


Fig. 8. Non regular implicit algorithm.

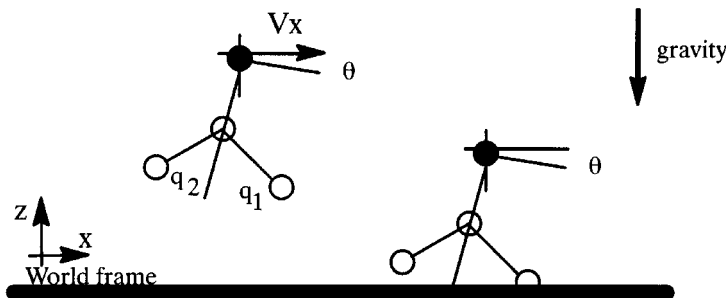


Fig. 9. Experimental setup.



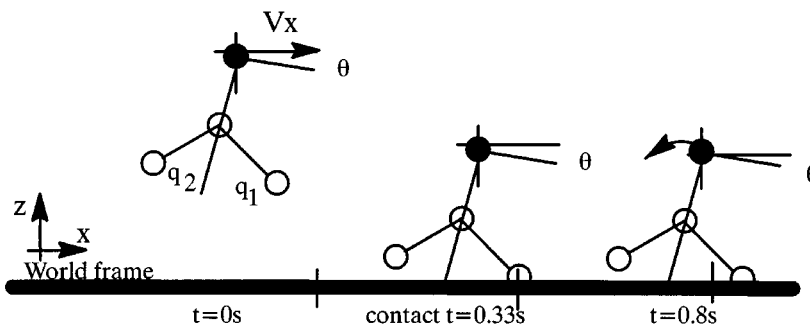


Fig. 10. Results of the simulation.

note a ballistic flight (Figure 11-b) of the robot with a constant angle  $\theta$  for  $t < 0.33s$  (Figure 11-c). During this period, the robot is free and both Lagrange and non regular methods provide the same solution because there is no contact.

After  $t=0.33s$ , the non-regular method is active. The robot has persistent friction and starts to slip in the direction of the x-axis (x continues to grow (Figure 11-a)). The robot loses energy until it stops and its center of gravity is located at the left of the point of contact, hence the robot falls backwards (x decrease).

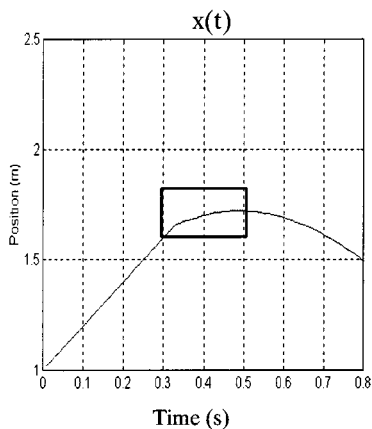
**Behavior of the external forces.** The Figure 12-a represents the normal force using the non-regular approach. The impulsion is well detected (Dirac's function). There are

several impacts due to bouncing and stabilization appears after  $t=0.55s$ . Note also that the normal force decreases at the end of the simulation because the robot turns backward until the second foot reaches the ground (not shown here).

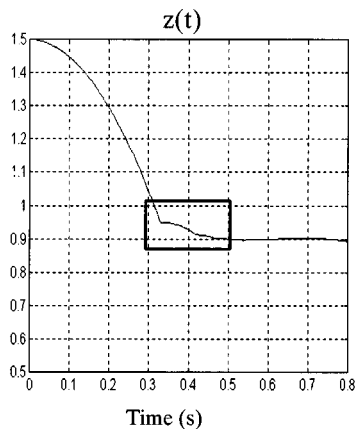
The Figure 12-b represents the tangential force, the constant value shown after  $t=0.55s$  means that the robot pushes not vertically on the ground, but this does not mean slipping because the reaction force is in the friction cone. This conclusion is verified by a null tangential velocity for the contact point (not shown here).

### 3. CONTROL OF DISPLACEMENT OF THE NON-REGULAR MODEL

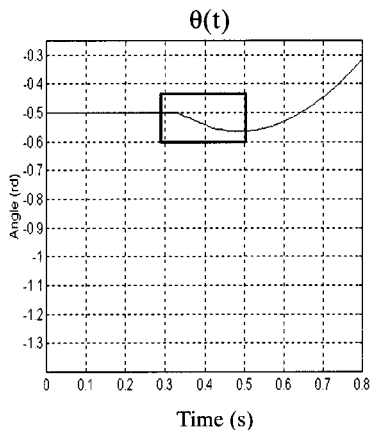
This modeling can be applied to every mechanism with interactions and discontinuous contact with its environment



11-a- Position of the head on axis x



11-b- Height of the head on axis z



11-c- Pitch angle of the head


 This square focuses attention on the time when the impact occurs.

Fig. 11. Cartesian behavior of the head of the robot.

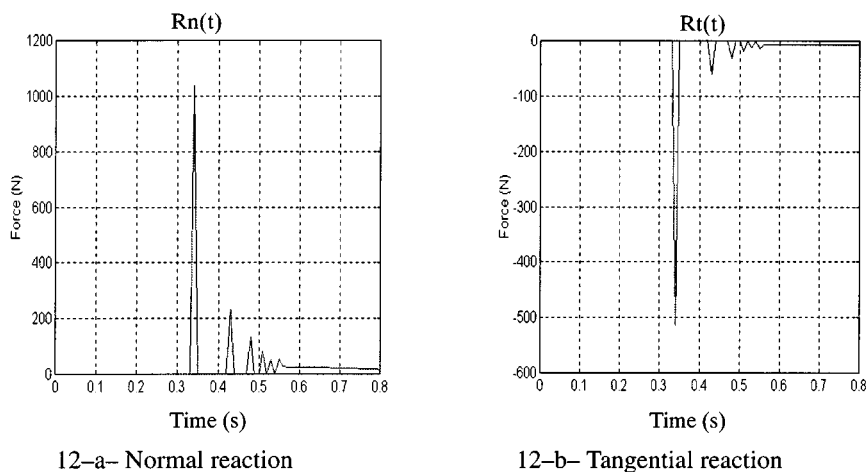


Fig. 12. Local reaction forces.

(manipulators, gears etc.). In this paragraph, we study the unified equation issued from the above theory in order to stabilize the biped with a linearization step and control its displacement.

3.1. Linearization and singular points

To specify a control scheme, we first study the system around some point of equilibrium. The unified equation describing the sytem can be written:

$$A(q)\ddot{q} = (\Gamma - B(q, \dot{q}, R)) \tag{16}$$

Here we can easily solve the dynamics for each configuration phase as we want to isolate specific singular points. Considering the planar biped structure, we have defined three possible configurations, as shown in Figure 13 (0, 1 or 2 contacts).

The singular points can be computed by solving equation (16) with zero accelerations and velocities. The singular points here are the stable or unstable equilibrium points of the structure in each configuration phase. If torques are applied to let the legs on position, stable equilibrium of the robot can only be reached with 2 contacts and of course there is no equilibrium in flight and unstable equilibrium on one foot. By rewriting (16) we obtain thus the following equation (17):

$$\Gamma = B(q, R) \tag{17}$$

In this article, we do not consider the case of free robots and double contact phase. We only explain and treat the robot in its unstable equilibrium on one foot.

Solving (17), we obtain five relations on generalized forces ( $R$ ), generalized torques ( $\Gamma$ ) and generalized positions ( $q$ ) of the robot.

$$F_x = 0 \tag{18}$$

$$F_z = \sum m_i * g \tag{19}$$

$$\sin(\theta) = \frac{\Gamma_1 + \Gamma_2}{l_i m_r * g} \tag{20}$$

$$\sin(\theta + q_1) = \frac{-\Gamma_1}{l(m_i + m + m_h) * g} \tag{21}$$

$$\sin(\theta + q_2) = \frac{\Gamma_2}{l m * g} \tag{22}$$

Relations (18) and (19) show that at equilibrium, the center of gravity of the robot is located exactly above the point of contact ( $\mathfrak{H}_T = F_x = 0$ ) and the normal force  $\mathfrak{H}_N$  is equal to the weight of the robot ( $F_z$ ) (19). Equation (22) shows that the torque of the free leg ( $\Gamma_2$ ) in equilibrium only compensates for its mass.

This set of equations has an infinite number of solutions ( $\Gamma, q$ ).

We can give for example the upright position ( $q = q_1 = q_2 = 0$ ) (Figure 14-1), equilibrium at right angles ( $\theta = \pi/2, q_1 = -\pi/2, q_2 = 0$ ) (Figure 14-2), equilibrium

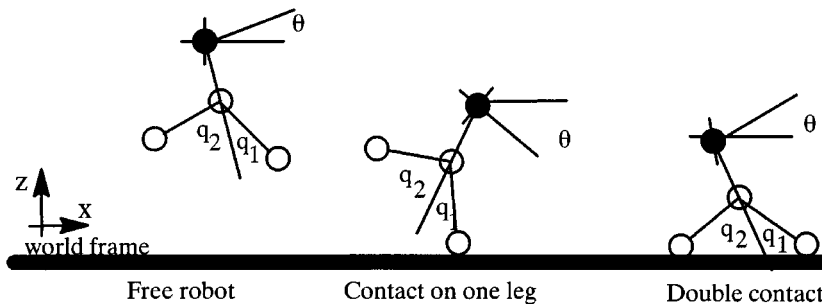


Fig. 13. Configurations delivering singular points.

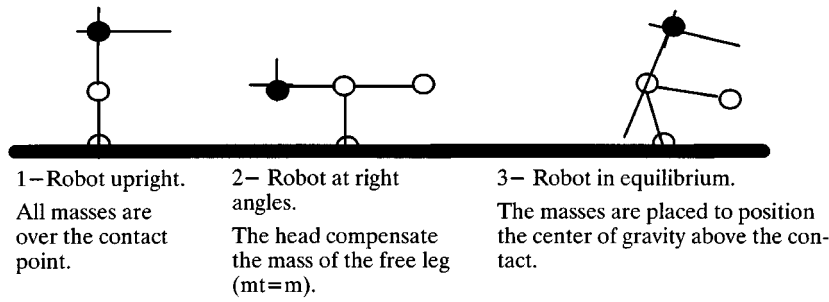


Fig. 14. Special point of equilibrium.

given by other solutions of the three equations (20), (21) and (22) (Figure 14-3).

**Linearization with one contact.** We linearize the set of equations (16) which gives  $\ddot{q}=f(q, \dot{q}, u)$  around a given singular point  $(q0, \dot{q}0, u0)$ . For little variation  $(\delta q, \delta \dot{q}, \delta u)$  around the considered point, we obtain:

$$\dot{X} = \begin{bmatrix} \delta \dot{q} \\ \delta \ddot{q} \end{bmatrix} = \begin{bmatrix} 0_{3 \times 3} & I_3 \\ \frac{\partial f}{\partial q} & \frac{\partial f}{\partial \dot{q}} \end{bmatrix}_{(q0, \dot{q}0, u0)} X + \begin{bmatrix} 0_{3 \times 2} \\ \frac{\partial f}{\partial u} \end{bmatrix}_{(q0, \dot{q}0, u0)} \delta u \quad (23)$$

Where:

- $0_{m \times n}$  represents the zero matrix (dim  $m \times n$ )
- $I_n$  represents the identity matrix (dim  $n \times n$ )

This can be written again in the well known first order linear differential form:

$$\dot{X} = A_{cde} X + B_{cde} V \quad (24)$$

With the dynamics written in (16) and the partial derivatives through the different variables computed at the point  $(X0, u0)$ , we obtain the coefficients of the above equation:

–  $A_{cde} = \begin{bmatrix} 0_{3 \times 3} & I_3 \\ \frac{\partial \ddot{q}}{\partial x} \end{bmatrix}_{(X0, u0)}$  Evolution matrix of the system  $6 \times 6$ .

–  $B_{cde} = \begin{bmatrix} 0_{3 \times 2} & I_3 \\ \frac{\partial \ddot{q}}{\partial u} \end{bmatrix}_{(X0, u0)}$  Control matrix  $6 \times 2$ .

–  $V = \delta u = (\delta \Gamma_1 \ \delta \Gamma_2)^T$  Vector of the two torques around  $u0$ .

We now study the controllability around the used point  $(X0, u0)$ . We must study all the singular points of the system. We use the Kalman criterion: using the matrix  $A_{cde}$  and  $B_{cde}$  we build the controllability matrix and, if the rank of this matrix is equal to the order of the system, the whole system and all the variables are controllable. We have not computed all the singular point, only the three presented in (Figure 14). In each point, Kalman’s criterion is verified, that means all of these configuration are controllable and we

can linearly stabilize the system around the designed point (no impact and no transition problem).

**Control with state feedback.** The more generic method to servocontrol the system around one singular point is to make a state feedback on the linearized point to place the poles at desired values and make it stable. On the first order linear differential equation (24), we thus compute the control  $V=U - L*X$  to obtain the equivalent system:

$$\dot{X} = (A_{cde} - B_{cde} * L)X + B_{cde} U \quad (25)$$

$L$  is computed in order to place the poles of the equivalent evolution matrix  $(A_{cde} - B_{cde} * L)$ . The poles must have negative real parts and we choose the values  $-3, -3.3, -3.9, -4.2$  and  $-4.6$  for the next experiments. We can build the corresponding control scheme to study equilibrium and the robustness of the system (Figure 15).

The system has null eigenvalues but state feedback allows us to place the poles anywhere. The system is servocontrolled only to the resting point considered because all the feedback has been made with linearization around it. The disturbances are introduced through a noise input and an error on the initial conditions vector. The error made on the initial condition corresponds to an error made on the equilibrium position at the beginning of the test. The behaviour of the non-linear system (16) is equal to that of the linearized system (25) because it is controllable around this point.

3.2. Simulations on balance

We study the position as shown in Figure 14-3 (free foot mass and head mass compensate for the hip mass). We simulate the behavior of the biped for two different initial positions near the solution of equilibrium ( $\theta_0 = -0.41rd, q1_0 = 0.57rd, q2_0 = 0.82rd$ ) with noise (white between  $\pm 0.1$  Nm) on torques limited to 10 Nm.

- test 1:  $\theta_{ini} = -0.45$  The robot recovers its stability (Figure 16 a-b-c)
- test 2:  $\theta_{ini} = -0.381rd$  The robot falls backwards and never straightens up (Figure 16 a-b-c).

The robot cannot be servocontrolled to the desired steady point in the presence of high disturbance (test 2). The free leg rotates several times to provide internal forces without being able to recover the balance of the robot (Figure 16-c). However, there is a domain that enables correct control, this



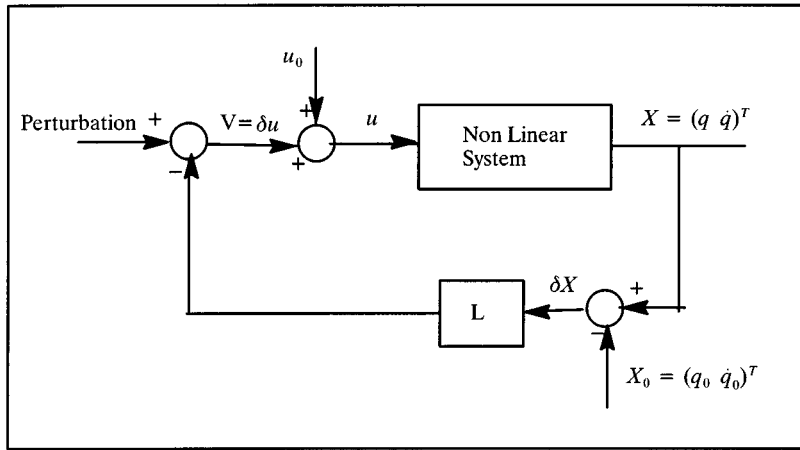


Fig. 15. Linear control scheme.

domain corresponds to the domain of validity of the linearization. This is why in the second test (2) the robot moves away from the desired point. The feedback is not well computed and the robot never stays stable (Figure 16 a-b-c).

It is important to note that we can balance a *quasi* double inverted pendulum, without actuators at ground contact point, but this double pendulum is counterbalanced with the mass of the free leg.

Here, we only study the controllability of the biped but we did not define any trajectories to follow. The system is

controllable but this does not mean that we can follow a trajectory, moreover we are limited to the domain of validity of this control scheme. We can therefore only compute torques to go from one point to another in finite time regardless of the trajectory, but here the trajectory obtained must always stay in the domain. We shall now try to control the displacement of the robot using another approach.

3.3. Under-actuated systems

We noted that  $\Gamma$  has zero components, which means that we have an under-actuated system. We can separate equation

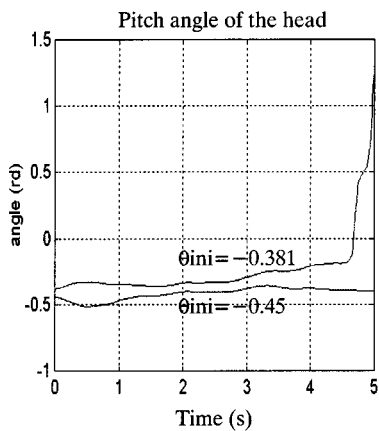


fig 16 -a- Pitch angles servo controlled

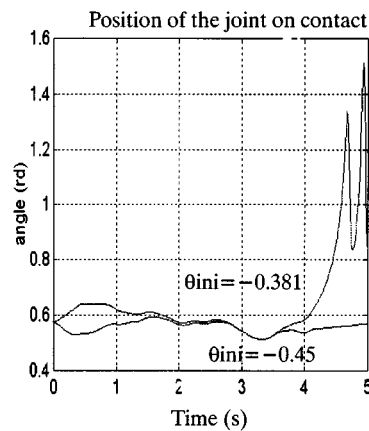


fig 16 -b- q1 servo controlled

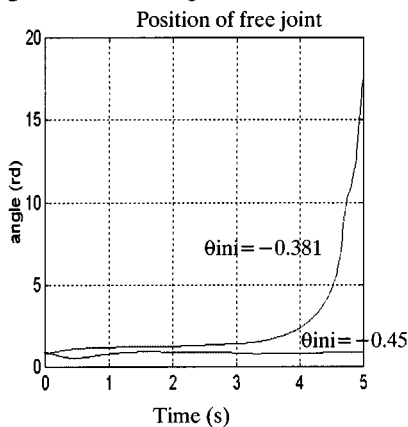


fig 16 -c- q2 servo controlled

Fig. 16. Evolution of the three coordinates.

(5) in two subsystems. The first one (indexed “c”) is the controlled part. The complementary system (indexed “f”) lets the other variables evolve freely according to the coupling between each variables. We obtain:

$$\begin{bmatrix} 0 \\ C \end{bmatrix} = \begin{bmatrix} Af(q) \\ Ac(q) \end{bmatrix} \ddot{q} + \begin{bmatrix} Bf(q, \dot{q}, R) \\ Bc(q, \dot{q}, R) \end{bmatrix} \quad (26)$$

To control parts of the acceleration vector  $\ddot{q}$ , we must define which components have to be driven and reduce the system to a controllable subsystem. We choose 2 components (indexed “c”) from among the 5 generalized accelerations. Note that there are 10 different ways to choose the variables, and thus 10 different ways to write  $Af$ ,  $Ac$ ,  $Bf$  and  $Bc$  components. Two dependant sub-systems can be written in each case:

$$[0] = [Aff(q)](\ddot{q}_f) + [Bf(q, \dot{q}, R)] + [Afc(q)](\ddot{q}_c) \quad (27)$$

$$[C] = [Acc(q) \ Acf(q)] \begin{pmatrix} \ddot{q}_c \\ \ddot{q}_f \end{pmatrix} + [Bc(q, \dot{q}, R)] \quad (28)$$

Separating and combining equations (27) and (28), we finally obtain the dynamics of the reduced controllable system (dimension 2):

$$A_{rs}(q)\ddot{q}_c = C - B_{rs}(q, \dot{q}, R) \quad (29)$$

Where:

- $A_{rs}(q) = Acc - Acf * (Aff^{-1}) * Afc(q)$  Equivalent evolution matrix ( $2 \times 2$ )
- $B_{rs}(q, \dot{q}) = Bc - Acf * (Aff^{-1}) * Bf$  Equivalent vector of all forces ( $2 \times 1$ )

**Global control scheme.** Now we can use the classical non-linear dynamical decoupling on this equivalent sub-system. Here the control involves computing the torque vector  $C$  using the known matrix  $A_{rs}$  and  $B_{rs}$  as follows:

$$C = A_{rs}(q)e + B_{rs}(q, \dot{q}, R) \quad (30)$$

Regarding (29) and (30), we obtain (if  $A_{rs}$  is invertible (idem for  $Aff$ )):  $e = \ddot{q}$ . In other words, we obtain a behavior of 2 double integrators on the 2 chosen coordinates.

Finally, to control these double integrators, classical PID controllers are added to follow the desired trajectories. We build the unified control scheme Figure 17. Here, we choose

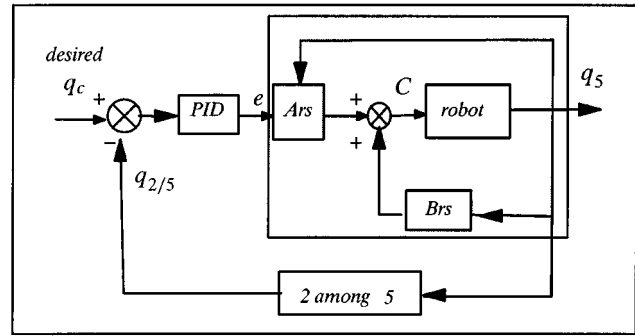


Fig. 17. Unified control scheme.

to control the displacement  $x$  and the pitch angle  $\theta$  that makes one kind of decomposition. In this case, we must still avoid different configurations where the computed torques are infinite.

Using the computation of the reaction forces computed by the non-regular model we obtain a non-regular vector  $Brs$  which represents the coriolis, centrifugal, gravity and reaction forces. We have built a non-regular control scheme able to compute torques regardless of the phases encountered during walking or running of the biped (fly phase, one contact or two contacts).

We made the first decomposition to reduce the system to one with as many coordinates as number of actuators, and we used the dynamics to make a decoupling feedback before considering the control of the equivalent system with PID controls.

### 3.4. Simulation results

The same planar biped robot (Figure 9) is used with two hip joints limited to 10 Nm. We let the robot fall (from less than 1 cm) to the ground, and we try to track two desired trajectories:

- a global linear displacement  $x$  ( $V_{xdes} = 0.3 \text{ m/s}$ )
- a constant posture of the body ( $\theta_{des} = 0 \text{ rd}$ )

The robot starts with leg 1 near the ground ( $q1 = -0.1 \text{ rd}$ ) and the free leg higher ( $q2 = 1.4 \text{ rd}$ ). In this case, we are thus able to control the displacement and attitude of the head of the robot while letting its legs move freely. The next figure (Figure 18) shows different samples obtained during the simulation.

The robot starts to fall without initial velocities, the robot reaches the desired trajectories and is servo-controlled near

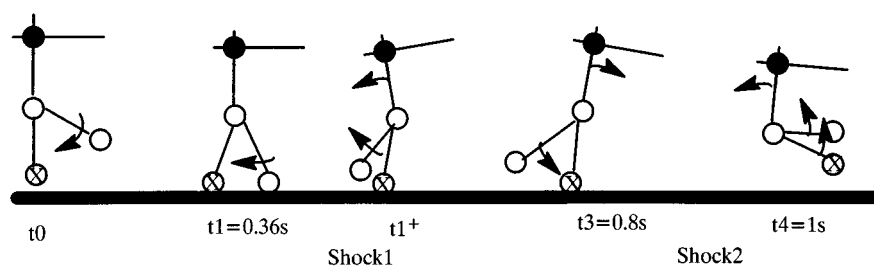


Fig. 18. Robot in contact with limited torques.

these specified values. (Figure 19-a and Figure 19-e). The robot lands on the rigid ground on one foot, moves with contact before the free foot (Figure 19-d) strikes the ground ( $t=0.36s$ ). After this disturbance, the robot is still controlled before the second shock ( $t=0.9s$ ), the robot takes off and enters in an unexpected flying phase, but during the ballistic flight the robot is still controlled (Figure 19-b).

- From  $t=0$  to  $t=0.36s$ , the robot follows the desired trajectories ( $x$  and  $\theta$ ). These trajectories can be tracked due to the movement of the free leg (Figure 19-d).
- We can notice the effect of the first impact on the pitch

angle at  $t=0.36$ . The impact is not significant on the  $x$  coordinate (Figure 19-a), but the disturbance is evident on the pitch angle (Figure 19 e).

- From  $t=0.36s$  to  $t=0.5s$ , the robot is on its two feet and this is a slipping phase. This could be avoided by the action of a knee which may allow the robot to cross its legs without hitting the ground.
- From  $t=0.5s$  to  $t=0.95s$ , the robot is on one foot and the free leg is on the other side of the body. Its action is well defined to balance the robot because the  $x$  displacement is still correct and the pitch angle returns to the desired position (0 for instance).

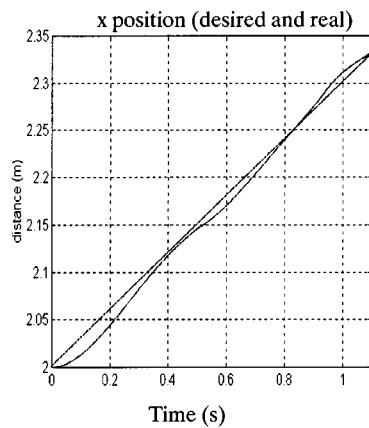


fig 19 -a- Horizontal position of the head

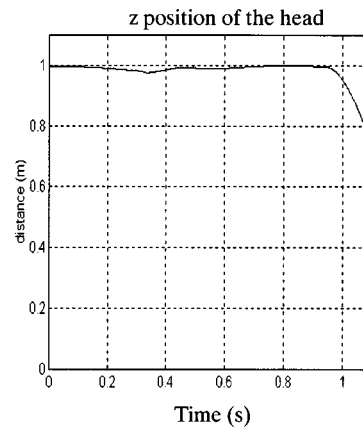


fig 19 -b- Height of the head

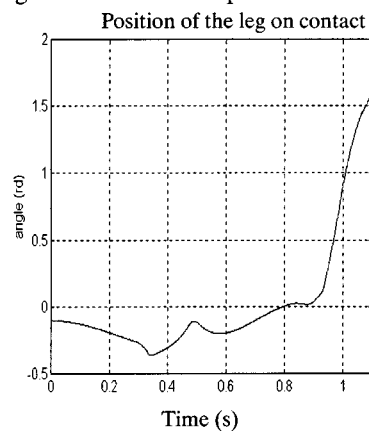


fig 19 -c- Joint 1 (on contact)

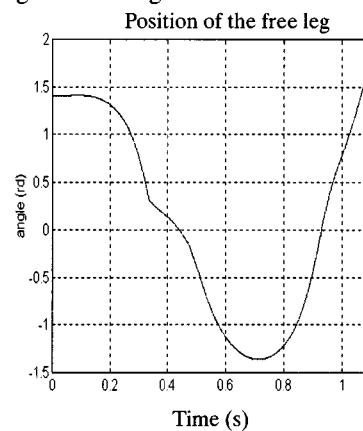


fig 19 -d- Joint 2 (free)

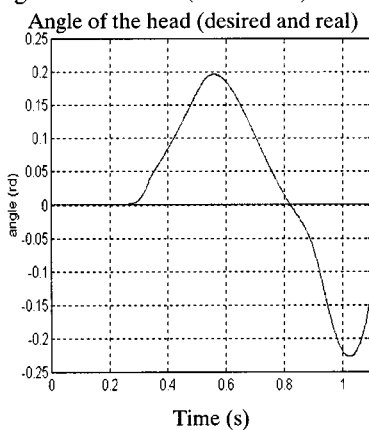


fig 19 -e- Pitch angles.

Fig. 19. Time course of the five generalized coordinates.

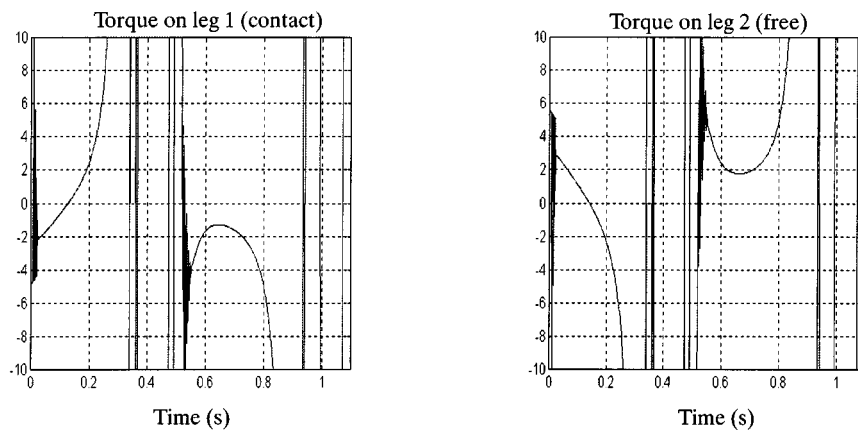


Fig. 20. Computed limited torques.

- The second impact occurs at  $t=0.95s$ . The free lag again strikes the ground strongly (Figure 19-d). An error occurs on the x position and on the pitch angle (Figure 19-e).
- For  $t > 0.95s$ , the robot is servocontrolled during the flight phase, but it still continues to fall thus prohibiting the walk that we want to obtain.

**Description of torques.** The control scheme uses the computed generalized reaction forces to compute the desired torques (Figure 20) allowing robot control during all the phases. There is no switch between different models and

no switch between different control schemes. We only use the external forces estimated to use the same dynamics all the time.

**Description of reaction forces.** The external forces are shown in the next figures (Figure 21-a-b-c-d). These reaction forces allow us to identify the different phases encountered during the simulation.

Most of the time the robot is in contact on one foot, but 2 contact phases on the second foot appear (at  $t=0.4s$ , and

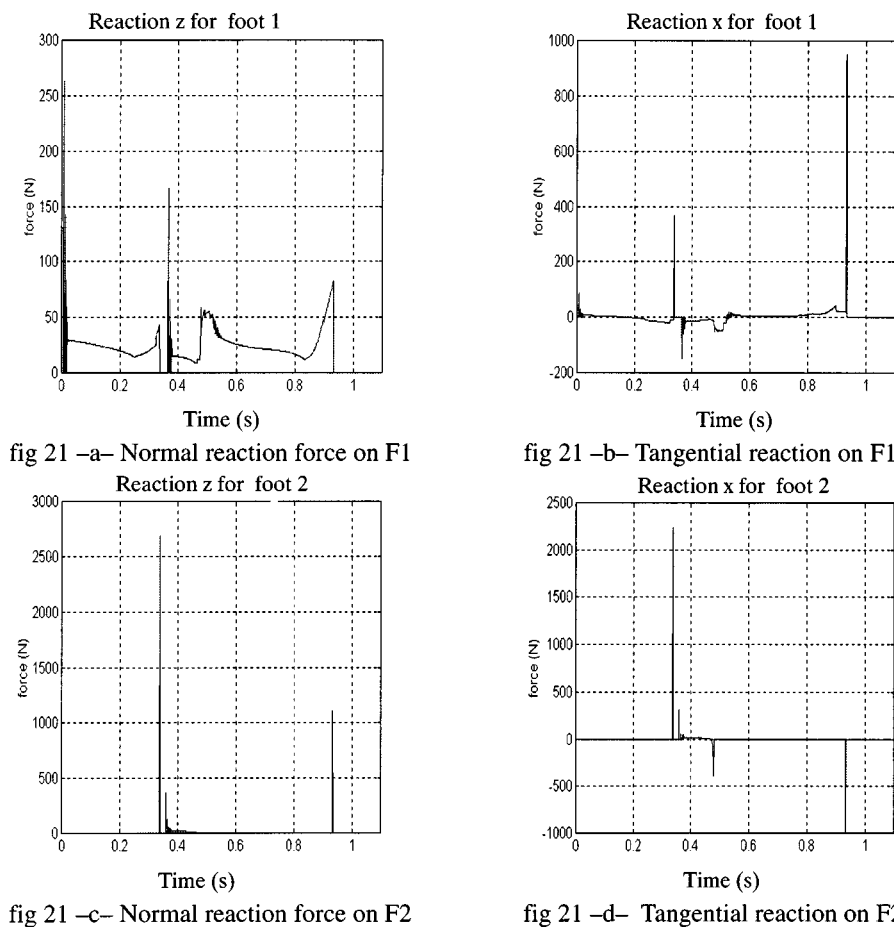


fig 21 -a- Normal reaction force on F1

fig 21 -b- Tangential reaction on F1

fig 21 -c- Normal reaction force on F2

fig 21 -d- Tangential reaction on F2

Fig. 21. Normal and tangential reactions.

$t=0.95$ ) which leads to a bouncing and impact phenomenon. At the end of the simulation, the robot takes off and enters a flying phase with no feet on the ground.

At the beginning, the robot is free ( $t=0$  to  $0.01s$ ) after there is a landing phase with substantial impulse forces and a constant contact phase on one foot. The movement depends on the evolution of the free leg as described above (Figure 19-d).

In this example, the robot follows the desired trajectories regardless of the constraints under the robot. Takeoff is due to the stroke of the free leg on the ground and the control scheme computes two torques, which are important and the two legs lose contact. The ballistic flight is well managed but the movement cannot continue because the step configuration is not respected.

#### 4. ANALYSIS AND CONCLUSIONS

Much current research deals with legged locomotion but the problem remains partially unsolved. Interesting structures have been studied for a long time and some prototypes bounce or walk slowly. In this paper, we addressed the problem of dynamic modeling of legged structures and examples are given with a planar biped robot.

We focused on establishing a unified equation to globally describe the behavior of a walking robot in every configuration to be able to define a global control scheme. We have derived non-regular dynamic modeling from the basic Lagrange's equations using contact dynamics. By using non-regular dynamics and implicit equation solving, the robot may be kept from intruding the ground, contrary to regular dynamics. There is no switch between modes, but unified modeling regardless of the phases encountered (0, 1 or 2 contacts). This new method has different kinds of advantages. Slipping and impact are well managed, respecting unilateral and non-linear constraints. There is no fixed geometrical condition and no computation *a posteriori*. A constant integration step can be used throughout the displacement of the robot. We estimate all the generalized coordinates and all forces and velocities acting during all phases on each contact point.

A control scheme was studied too. The control scheme must be built in two stages. In one stage, we dealt with the under-actuation of the system ( $I$  has as many zero components as cartesian coordinates) to obtain a controllable sub-system. In the second stage, we used the knowledge obtained by our non-regular dynamic model to make a non-regular dynamical decoupling. We thus obtained an equivalent linear system which can be controlled. Simulations were done to implement the theory, and the global control scheme seems to be satisfactory. Despite the unified control scheme, no stable walking is achieved due to the trajectory tracking approach. The problem now is to define reachable trajectories. In fact, the desired trajectories must comply with limitations due to the dynamics and take limitations of the actuators into account. One approach to solve this problem would be to define a switch from different sets of desired trajectories, depending on the possibilities of the biped during the different walking

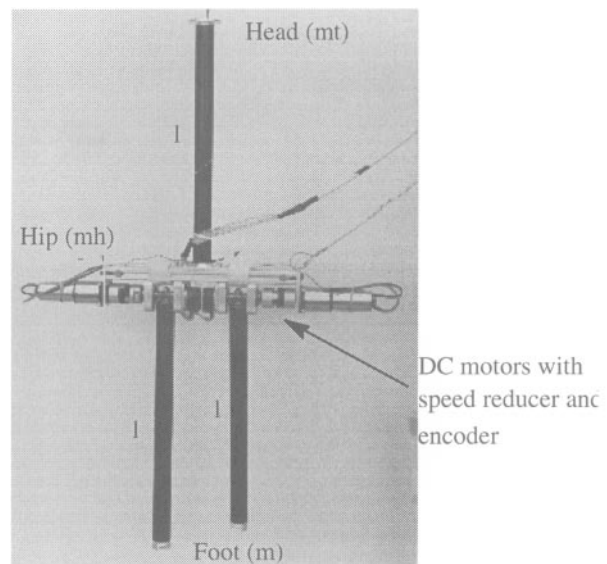


Fig. 22. Front view of our prototype.

phases, or to consider a learning or fuzzy control scheme rather than trajectory tracking.

As shown in the picture (Figure 22), we are now in the process of building a prototype to validate our model. The first experiment must test equilibrium on one foot before testing a walk.

#### References

1. H-J Weidemann, F. Pfeiffer and J. Eltze, "A Design Concept For Legged Robots Derived From The Walking Stick Insect" *Proceedings of the 1993, International Conference on Intelligent Robots and Systems* (1993) pp. 545–550.
2. H. Miura and I. Shimoyama, "Dynamic Walk of a Biped" *Int. J. Robotics Research* **3**(2), 60–74 (1984).
3. S. Kajita and K. Tani, "Study of Dynamic Biped Locomotion on Rugged Terrain" *Proceedings of the 1991 International Conference on Robotics and Automation* (1991) pp. 1405–1411.
4. P. Nagy, W. Whittaker and S. Desa, "A Walking Description for Statically-stable Walkers based on Walker/Terrain Interaction" *Proceedings of the 1992 International Conference on Robotics and Automation* (1992) **Vol. 1**, pp. 149–156.
5. J. Furusho and A. Sano, "Sensor-based Control Of A Nine-link Biped", *Int. J. Robotics Research* **9**(2), 83–93 (1990).
6. P.H. Channon, S.H. Hopkins and D.T. Pham, "Modelling and Control of A Bipedal Robot" *J. Systems Engineering* No. 2, 46–59 (1992).
7. T. McGeer, "Passive dynamic walking", *Int. J. Robotic Research* **9**(2), 62–82 (1990).
8. M.H. Raibert, 'Legged Robot that Balance'. (M.I.T.-Press, Cambridge, MA, 1986).
9. N. M'Sirdi, N. Manamani and N. Gauthier, "Modelization and Identification of a Pneumatic Legged Robot" *Proceedings of the 1996 conference on Computational Engineering in Systems Applications* (1996) **Vol. 4**, 803–808.
10. S. Caux and R. Zapata, "Towards An Unified Form of Biped Robot Dynamic Modeling", *Proceedings of the International Conference on Systems, Man and Cybernetics, Orlando-U.S.A.* (12–15 October, 1997) pp. 3249–3254.
11. J.J. Moreau, "Unilateral Contact and Dry Friction in Finite Freedom Dynamics", **In: Nonsmooth Mechanics and Applications**, C.I.S.M. courses and lectures, No. 302 (Springer-Verlag, Wien, NY, 1988), pp. 1–82.
12. Y. Hurmuzlu and D. Marghitu, "Rigid Body Collisions of



- Planar Kinematic Chains With Multiple Contact Points”, *Int. J. Robotics Research* **13**(1), (1994) 82–92.
13. C.L. Golliday and H. Hemami, “An Approach To Analyzing Biped Locomotion Dynamics and Designing Robot Locomotion Controls”, *Transactions on Automatic Control* **AC-22**, No. 6, 963–972 (Dec., 1977).
  14. W. Stronge, “Rigid Bodies Collisions With Friction”, *Proc. Roy. Soc. London* pp. 169–181.
  15. K. Inagahi and H. Kobayashi, “Dynamical motion control for quadruped walking with autonomous distributed system”, *Proc. International Conference on Intelligent Robots and Systems* (1996) **Vol. 2**, pp. 1006–1009.