Constraint-based Ground contact handling in Humanoid Robotics Simulation

Eduardo Martin Moraud, Joshua G. Hale, Gordon Cheng

HAL Id: inria-00350114
https://inria.hal.science/inria-00350114

Submitted on 5 Jan 2009

Constraint-based Ground Contact Handling in Humanoid Robotics Simulation

Eduardo Martin Moraud, Joshua G. Hale and Gordon Cheng

Abstract— This paper presents a method for resolving contact in dynamic simulations of articulated figures. It is intended for humanoids with polygonal feet and incorporates Coulomb friction exactly. The proposed technique is based on a constraint selection paradigm. Its implementation offers an exact mode which guarantees correct behavior, as well as an efficiency optimized mode which sacrifices accuracy for a tightly bounded computational burden, thus facilitating batch simulations or the use of highly detailed foot models. The method has been applied to the simulation of two humanoid robots, CB and Hoap-2, and validated with respect to human and robot motions recorded in reality and in simulation. The friction parameters of contact have also been established empirically.

I. INTRODUCTION

Biped locomotion is a complex mechanism, highly dependant on the ground reaction forces exerted under the feet, which support the whole body and are essential in balancing and control. 3D environments often require accurate replication of such forces for the sake of visual realism. The fidelity of contact simulation is of even greater importance for *humanoid robot platforms* [5], [6] where simulation interfaces provide roboticists with preliminary results, prior to testing with a real robot, that are relied upon to guide the manual modification of control patterns, perform comparative tests and train learning systems in a safe and speedy manner. As such, these interfaces enable researchers to develop controllers off-line or gather feedback data with a high confidence that identical behaviors will result when controllers re-used in reality.

This paper presents a Coulomb friction contact algorithm intended for simulating ground contact within the humanoid robot platform developed by Hale *et al.* [4]. The method yields accurate ground reaction forces, validated against real force-plate measurements, and can be used to implement an interface to a real or simulated robot transparently. It handles the friction cone exactly and utilizes precise physical data specific to ground contact, such as the center of pressure of the feet. Our implementation also facilitates switching between two complementary decision-making components, thus offering a straightforward trade-off between accuracy and speed, as suited to usage requirements.

Eduardo Martin Moraud was with ATR, 2-2-2 Hikaridai Seika-cho Soraku-gun, Kyoto 619-0288, Japan. He and is now with the French National Institute for Research in Computer Science and Control, INRIA Nancy Grand Est research unit, 615 rue du Jardin Botanique, 54600 Villers-les-Nancy, France, `eduardo.martinmoraud@mines-paris.org`

Joshua G. Hale and Gordon Cheng are with the Department of Humanoid Robotics and Computational Neuroscience, ATR Computational Neuroscience Laboratories, 2-2-2 Hikaridai Seika-cho Soraku-gun, Kyoto 619-0288, Japan. They are also with JST-ICORP, Computational Brain Project, 4-1-8 Honcho, Kawaguchi, Saitama, Japan. `{jhale,gordon}@atr.jp`
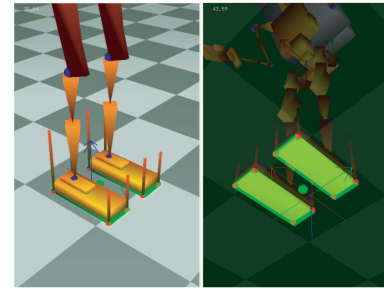
Fig. 1.   Contact simulation showing ground forces for rectangular feet

Existing contact models can be gathered into three main approaches. 'Penalty methods' handle contact forces using virtual springs and dampers at the contact points. They are easily implemented, but they focus on visual plausibility and do not enforce strict non-penetration [10]. 'Impulse-based' approaches model contact between bodies using collisions at the contact points by applying instantaneous velocity changing impulses, but in general are not very efficient for certain types of prolonged, stable or simultaneous contacts such as with the ground [8]. Finally, 'analytical methods' attempt to resolve contacts in accordance with the dynamic rules governing interactions between bodies. A typical approach is to formulate a linear complementary problem (LCP) and then calculate the contact forces which comply with the conditions defining a valid contact. Stable results are usually obtained, but solving such problems reliably is not an easy task. Moreover, when considering friction, the convexity of the LCP disappears and a correct solution cannot be guaranteed. The Coulomb friction model must be implemented using a polyhedral representation of the friction cone, which has the effect of making friction anisotropic in three-dimensional systems. An innovative alternative was proposed by Yamane and Nakamura *et al.* [14] who developed an efficient analytical method which replaces the long and complicated simplification process required for solving an LCP with a *trial-and-error* approach. The procedure is specifically intended to treat ground contact for humanoids and finds the best solution in a three-step loop: first assuming certain constraints on the contacting bodies, then computing the forces and moments necessary to enforce compliance with these constraints, and finally checking whether the resulting forces satisfy some predefined conditions defining valid contact. This process is repeated until a satisfactory solution is found. However, obtaining the most appropriate set of constraints is not always guaranteed. In addition, the

method requires the use of a specific recursive dynamics formulation [13], and being expressed for rectangular feet, it is not clear that more complex shapes can be handled.

The method we present makes use of the 'trial-and-error' paradigm in order to exploit the efficiency and ease of implementation which it offers, but motion is now controlled by enforcing strict foot constraints. This approach has many advantages with respect to existing concepts that rely only on heuristics [3], [11]. The constraints sets correspond to intuitive contact states, as explained below, and are modified only when certain predefined rules defining valid contact are satisfied. The method therefore does not require any prior biomechanical knowledge. In addition, it can be utilized under any dynamics formulation and thus does not require modification of the dynamic description of the simulated humanoid. The result is a straightforward algorithm which handles arbitrarily shaped polygonal feet and always terminates with a satisfactory state. Furthermore, the method may be run either in a 'complete' mode which ensures rigorously computed and correct results, or a faster version may be executed which exploits predictions regarding the contact state, thus trading off accuracy for computation time. This is useful for debugging code or gathering preliminary results where speed rather than precision is desirable.

This paper is organized as follows. We first describe the fundamentals of our method *i.e.*, the constraint selection paradigm on which the model is based and the rules defining a valid contact. The *exact* decision-making algorithm is then presented, which embodies the 'trial-and-error' approach adapted to our constraint-based environment. Sections 3 and 4 present complementary conditions which may used to improve both the accuracy of the original model, through the use of the *center of pressure*, or its efficiency, through the use of a faster *one-step* approximation algorithm. Finally, the results of simulations are compared with measurements of forces yielded by a human and by a robot standing on a force plate. This *validation* is described in Section 5 and demonstrates that the proposed method is a fast and realistic model of ground contact applicable to humanoid robotics simulation.

II. EXACT CONSTRAINT MANIPULATION METHOD

A. Virtual environment

The 'robot simulation and control platform' developed by Hale *et al.* [4] was created for the humanoid robot 'CB'[2] and is intended to help gain on understanding of human behavior and transfer its underlying principles to humanoid robots [1]. The platform acts as a transparent interface to both the real and virtual world and facilitates combining the control architecture of the humanoid robot with simulation software, thus enhancing the possibilities for interaction. It incorporates support for systems such as motion capture, automated balancing, *etc.*, while accommodating discrepancies and correcting noisy measurements.

The simulation environment is a key component of the platform and its reliability is essential to obtain meaningful
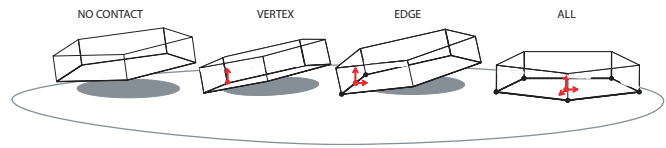

Fig. 2. Possible contact configurations, placement of the virtual joint, and motion constraints for polygonal feet with 5 vertexes. In the case of static friction, motion is unconstrained for the null-contact case; for the vertex contact the three position coordinates of the vertex are constrained; for the edge contact the positions of both vertices are constrained as well as rotations perpendicular to the edge; and for the planar contact all degrees of freedom are constrained. In the case of dynamic friction, the horizontal translational coordinates also remain unconstrained.

correlation with reality, ensuring that identical software can be used to control both simulated and actual humanoids. Our constraint-based model meets the demands of this scenario and provides reliable results which resemble the forces experienced by humans performing bipedal activity.

B. Constraint selection paradigm

Our contact model handles polygonal feet with an arbitrary number of vertexes. Most real robots have indeed planar feet which may be approximated with polygons. Hence, only certain contact configurations need to be considered, *i.e.*, a point -only one vertex touches the floor, an edge -two vertices are in contact with the ground, the complete sole -all the points that constitute the polygon are in contact, or no contact at all in a null contact situation. Each one of these states can be characterized in terms of the set of vertexes constrained to remain fixed to the ground, referred to as 'clamped' throughout the remainder of this paper, or inversely, the set of 'unclamped' vertices free to move.

Each foot has 6 degrees of freedom (DoFs), some of which may be constrained according to the contact configuration. Constraint forces are introduced into the system for this purpose. They enable the evolution the links' motion to be coordinated via a small number of variables and effectively reduce the number of DoFs in the system. These constraints are propagated through the joints dynamically and may be computed during simulation using Lagrange multipliers with Baumgarte stabilization. When applied to ground contact, since no 'actual' joint exists between the feet and the ground, a 'virtual' joint is created. Two *virtual joints* corresponding to each foot are added to the body-linked structure of the

TABLE I

CONSTRAINTS ASSOCIATED TO EACH POSSIBLE STATE

Nb points	Friction	Enforced Constraints
None	-	0
Vertex	Static	T_x, T_y, T_z
	Dynamic	T_z
Edge	Static	T_x, T_y, T_z, R_x or R_y, R_z
	Dynamic	T_z, R_x or R_y
All	Static	$T_x, T_y, T_z, R_x, R_y, R_z$
	Dynamic	T_z, R_x, R_y

system and for each foot are located at the *mean position of the points clamped, i.e.,* at a vertex, the mid-point of an edge, or the middle of the face. The coordinate frame of the foot is also colocated (the *z-axis* is perpendicular to the floor, pointing upwards, and the *y-axis* and *x-axis* correspond to the sagittal and coronal planes respectively). Ground reaction forces may then be incorporated when computing the dynamics of motion, constraining the motion to the feasible range. For example, the vertical translation T_z is constrained whenever a foot touches the ground, simulating the non-penetration requirement. Likewise, translations in the contact plane, spanned by T_x and T_y, are constrained to accord with the Coulomb model, which may demand either dynamic (sliding) or static friction. The constraints corresponding to each configuration are presented in full in Table I, where T_i indicates the translation along axis i and R_i rotation around axis i.

Having established this mechanism of constraints, the identification of the appropriate contact state becomes the critical component of this contact resolution method. If the constraints are not updated when necessary, the resulting motion will diverge from reality. The well known 'magnetic shoes' syndrome [7] could result for example in the floor attracting the foot, preventing it from being lifted or rotated. In addition, a lack of accuracy in the handling of constraints might lead to fluctuations in the contact state, causing undesirable 'chattering'. Precise decision-making is therefore fundamental in order to handle with the dynamic patterns of humanoid motion. A very specific set of rules, both hierarchical and complementary, was thus established so as to clearly define valid contact states and guarantee that stable results are obtained over contiguous time steps.

C. Rules describing valid contact

The rules defining the appropriate contact state first consider the **vertices** of the contact polygon. These points define the bounds of the surface in contact with the ground and their behavior provides accurate information which can be used to ensure valid motion. The violations that may occur can be categorized into three types, *geometric*, *kinematic* and *negative forces*, each of which is tracked and prioritized according to a convenient set of thresholds, *e.g.*, in terms of position, a distance of $+d$ above and $-d$ below ground (1mm was used). This thresholded model of the floor absorbs the lack of precision that results from numerical inaccuracies inherent in floating point computation. While this represents a compromise in the strict rigid body dynamics ostensibly modelled during simulation, it actually increases the reality
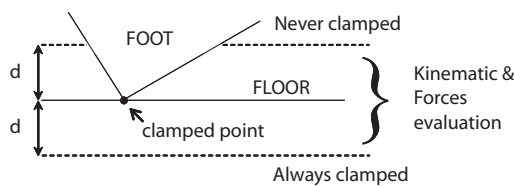

Fig. 3.   Rules for clamping vertices and thresholded model of the ground
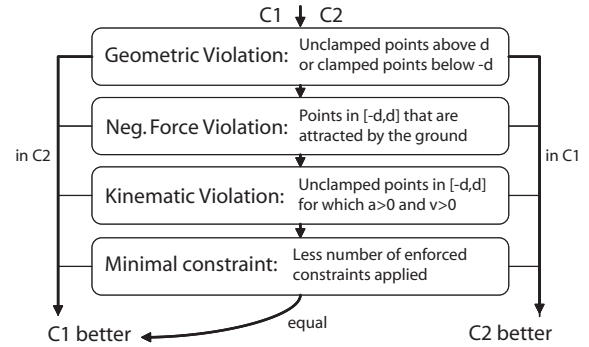

Fig. 4.   Prioritized binary comparison of two candidates C1 and C2

of the simulation by absorbing the infinite oscillations that result from infinitely hard bodies interacting inelastically. The model thus permits a small degree of compliance by allowing objects to slightly penetrate the floor, smoothes collisions and hence prevents the feet from chattering against the ground forever.

Contact configurations are evaluated according to the following criteria: geometric validity is considered first, ensuring that the feet do not penetrate the ground plane and that vertices are not clamped unless they touch the floor. Therefore, any point below -*d* must be clamped and any point above +*d* must be unclamped. This geometric condition is an absolute characteristic of the foot, not dependant on accelerations which may change in response to changes in ground reaction forces, and must therefore be satisfied regardless of other conditions. When no such violations occur, the thresholded region $[-d, d]$ is analyzed and more refined conditions are considered. Firstly, negative forces are taken into account, ensuring that neither foot is being attracted by the ground at any vertex. Any vertex which experiences a negative force from the floor should therefore be unclamped. Next, kinematic conditions are imposed, as all 'separating' vertices (for which $\vec{a_z} > a_t$ and $\vec{v_z} > -v_t$ where a_t, v_t are positive thresholds for which we chose $0.1\mathrm{ms}^{-2}$ and $0.1\mathrm{ms}^{-1}$) ought to remain unclamped; all others below the $+d$ threshold should be clamped. Finally, a principle of minimal constraint is utilized to ensure that any vertices that do not violate any higher priority conditions are unclamped, thus allowing the foot to move as free as possible while satisfying all other rules.

Additionally, Coulomb **friction** controls the horizontal sliding of the feet along the ground. There are two cases. Static friction requires that forces remain within the friction cone $\sqrt{f_x^2 + f_y^2} \leq \mu_s f_z$, where μ_s is the static friction coefficient, and prevents all translational motion in the contact plane. When the force necessary to keep the foot static is outside the friction cone, the behavior is modelled using dynamic friction. The constraints T_x, T_y restricting translation along the contact plane are released and replaced by a force opposing the direction of motion and linearly proportional to the vertical component of the ground reaction force.

The foot can then slide according to the dynamic friction equation $\sqrt{f_x^2 + f_y^2} = \mu_d f_z$ in the slipping direction given by

$$\left(\frac{f_x}{\sqrt{f_x^2 + f_y^2}}, \frac{f_y}{\sqrt{f_x^2 + f_y^2}}, 0 \right)^T$$

In all, the global organization of our model, which is mainly hierarchical, quantifies the requirements of valid motion in a consistent manner. It enables a *comparison* and *ranking* of plausible configurations as shown in Figure 4 as well as establishing an order for generating new potential configurations.

D. Candidate generation & model selection

Having established the conditions defining valid contact, the remaining task is to identify the appropriate contact state at a given instant. Our search model is based on 'trial-and-error' methods and thus involves generating and testing a sequence of states whenever the rules are violated. The procedure consecutively attempts different plausible contact states, compares them as described above and thus converges towards the valid solution.

In our constraint-based approach, testing a candidate requires setting the appropriate constraints and simulating their influence within the dynamics of the whole system. The relationship between all interconnected bodies (and the effect of both feet) can then be considered when calculating forces and accelerations. The computation for identifying the appropriate contact state is therefore interleaved with forward dynamics integration steps and in doing so, the search loop, core of this iterative approach, ensures that *exact* results are methodically obtained. Hence, every time a configuration is analyzed the rules are evaluated for each contact point and if violations occur, new candidates are generated. The corresponding new constraints are imposed and the dynamics calculations are performed, making it possible to evaluate the validity of the new state, compare it to previous configurations and store the result in a priority queue. This process is repeated until a violation-free configuration is found.
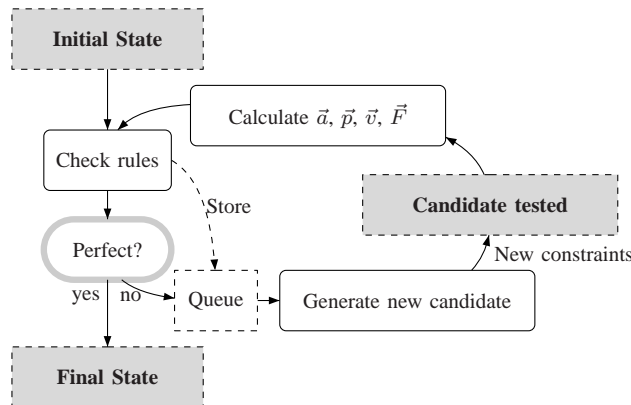

Fig. 5. Control flow of the exact method, including the iterative search loop and perfect state

The priority queue ranks all tested candidates and is used to *generate new states* in an efficient way. It is retained throughout the search process so as to keep an ordered record of the best configurations and ensures that new candidates are always created from the best solution found so far. Instead of performing a brute force search over all possible configurations, new candidates are obtained methodically from the previously tested configuration at the top of the queue. Configurations ranked at the bottom of the queue are thus less likely to be used to generate candidate configurations, because they are less likely to satisfy all the necessary conditions. The coherence between subsequent time steps further prevents unnecessary iterations, reducing the search burden and enhancing efficiency.

Candidate generation is thus as follows: new candidates are created for each foot by correcting the violations existing in the state at the top of the queue. Corrections are performed for each individual vertex according to the prioritized conditions explained above, *i.e*, first correcting geometric violations, then correcting negative forces, then kinematic problems, and finally satisfying the principle of minimal constraint. When no untested candidate may be created from the configuration at the head of the queue, that state is removed from the list and the process is repeated with the state next on the queue configuration. The method for generating candidates is defined so as to ensure the full connectivity of the space of all possible configurations. Hence, any current state will, if necessary, yield any other possible configuration.

The search-loop continues until the *perfect state* is reached, in which all the contact points from both feet comply with all predefined conditions, ensuring that at all times a stable solution is found which is better than any other possible configuration. Finding this candidate quickly with a high probability is the key to the approximate method described later in this paper.

III. CENTER OF PRESSURE

The use of thresholding is necessary to prevent chattering and to modify the strict rigid body model to produce more realistic behavior. As a consequence, the model selection may in some cases be satisfied by multiple 'perfect' configurations. This is particularly common when complex polygonal shapes with many closely located vertices are used to model the feet. The use of thresholds during model selection encourages the configuration to remain the same whenever possible, thus reducing unnecessary transitions between configurations and the corresponding computational overhead. The configuration is adjusted when violations exceed the threshold values, and it was observed that the accuracy could be improved without provoking state chattering by considering the center of pressure (CoP). An extra condition which considers the CoP of each foot was implemented, improving the efficiency of the existing rules and refining the original formulation in terms of accuracy and stability by specifying its behavior in the threshold region, previously considered to be ambiguous.
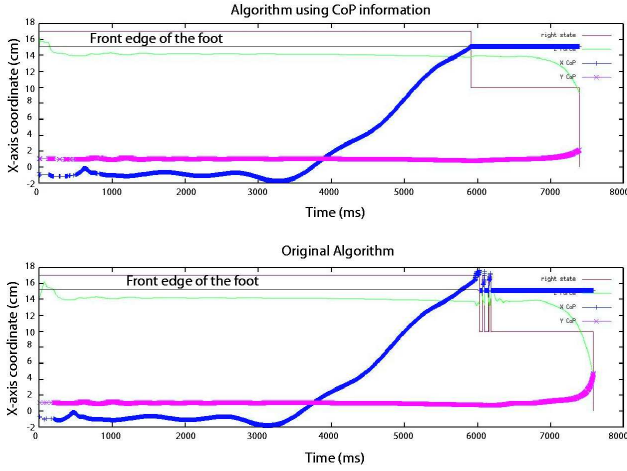
Fig. 6. Comparison of the position of the CoP (blue, forwards; purple, sideways), GRF (green), and configuration (red) during a forwards fall over the toes. *Top:* model incorporating CoP information. *Bottom:* original thresholded model. The CoP moves to the front of the foot as the robot bends, crosses the edge of the sole and leaves the support polygon resulting in discontinuities in the state of the foot until the transition is established.

The CoP is the point on the sole where the moment of the contact forces is perpendicular to the contact plane [9], [12]. It thus corresponds to the point on the sole of the foot where the net force and torque can be equivalently applied as a single force with no horizontal torque component ($\tau_x = 0$ and $\tau_y = 0$). Its location can be easily obtained from the constraint forces and torques applied through the virtual joint:

$$\begin{cases} x_{CoP} = \tau_y^{vj}/f_z^{vj} + x_{vj} \\ y_{CoP} = \tau_x^{vj}/f_z^{vj} + y_{vj} \end{cases}$$

The CoP is useful because it provides the information required to determine exactly when constraints ought to be *released*. When removing a rotational constraint, the torques around the corresponding axis disappear and the foot becomes able to rotate. It is thus a necessary physical condition at the instant of release that the moments along such an axis are zero. Hence, a transition from one state to another which releases a (rotational) constraint can only occur when the CoP touches an edge of the foot.

An example showing the influence that utilizing the CoP can have on configuration selection may be seen in Figure 6. The ground reaction forces in this case are simulated during a simple motion that causes a single illustrative transition. A humanoid, which is initially standing on both feet, bends forwards until the CoG leaves its polygon of stability. The feet then rotate around the front edges as the humanoid falls forwards.

IV. ONE-STEP MODEL

In addition, we have also developed a modification which facilitates a trade off between accuracy and computation time. The modification is of value when high precision results are not required, for instance, when performing preliminary analyses of new control code, gathering large quantities of data for learning processes, and when debugging control code. It enhances the rate of simulation may also be used to run the program in real time when handling highly complex haptic surfaces. The volume of computation is predetermined, so the computation time can be regulated precisely.

This concept of this 'one-step model' is to identify the best candidate from the current state in a single step with a high probability. The search-loop is thus replaced, a unique candidate is generated and simulation is no longer stopped for testing or comparing. Two requirements are generated. Firstly, problematic vertexes cannot be corrected individually at each iteration, so all violations must instead be examined *globally*. Secondly, the most suitable contact configuration for the next time-step must be *predicted* using only the information available in the current state. This probabilistic approach means that in some cases the best candidate is not selected within a single iteration so the simulation may proceed for a number of time steps with a configuration which permits violation of the appropriate contact constraints. The correct configuration is nevertheless selected after a number of iterations, and the effectiveness of the probabilistic selection ensures that divergences do not lead to drastically unrealistic behavior.

In order to handle the selection, the space of possible contact configurations is represented vectorially. Configurations are handled as vectors within a vectorial space σ_B spanning all possible contact configurations. Classic mathematical tools can be used to identify the configuration which is most likely to occur in the next iteration without testing and comparing multiple configurations. The procedure considers all the violations *en masse* and mathematically identifies the candidate which globally minimizes the problematic vertices. The core of the exact method is however maintained.

A. Vectorial space

Let

$$\left\{ \vec{b_i} \right\}_{i=0..N-1}, N \in \mathbb{N}, \vec{b_i} \in \{0,1\}^N$$

be an orthonormal basis where each vector $\vec{b_i}$ represents the ith vertex of the contact polygon. The space σ_B, spanned by the vectors of this basis, thus describes all the configurations of vertices and is defined as

$$\sigma_B = \left\{ \sum_{i=0}^{N-1} k_i \vec{b_i} \right\} \text{ where } \begin{cases} k_i = 1, & \text{point } i \text{ clamped} \\ k_i = 0, & \text{point } i \text{ free} \end{cases}$$

The *space of solutions* $\mathscr{A}$, defined by the set of vectors corresponding to physically possible contact configurations, is thus a sub-set of σ_B and is included in the space described above:

$$\mathscr{A} \subset \sigma_B$$

Other vectors which represent configurations that do correspond a possible configuration may also be contained in σ_B, *e.g.*, configurations where $2 < N_{clamped} < N$ vertices are clamped (representing neither a vertex, edge nor face), or pairs of non-adjacent vertices that therefore do not constitute an edge.

B. Perfect vector

The 'perfect state' remains the most desirable configuration, but it is instead represented by a *perfect vector*. The procedure for obtaining it also differs from the previous approach. Instead of iteratively correcting consecutive candidates, thus aiming to reach the perfect vector at a certain point, the new model begins by directly obtaining a state $\vec{p} \in \sigma_B$ where each vertex is in its perfect position *when considered independently*. Hence each contact point is evaluated separately according to the rules defining valid contact and a result of either 'clamped' or 'unclamped' is obtained for each vertex. The *theoretically* perfect vector $\vec{p}$ is then determined as the linear combination of these perfect points. However, this result does not necessarily correspond to a plausible configuration, since no attention has been paid to the question of whether the candidate is physically possible. The remaining task is to determine which is the best candidate among those physically possible, *i.e.*, those belonging to the space of solutions $\mathscr{A}$.

The most significant aspect of this approach is the separation of the problem into two complementary components. First, identifying the *theoretically perfect state* that perfectly suits the instantaneous characteristics of the motion, and second, identifying the best corresponding *real candidate* (which must correspond to the clamping of a vertex, an edge, the full face, or no points at all).

C. Decision-making

When no perfect candidate is reachable ($\vec{p} \notin \mathscr{A}$), the best state among the 'non-perfect' alternatives must be chosen, implying that points which theoretically should be clamped must be unclamped, or vice versa. The rules defining valid contact are used for comparing configurations, but under the one-step method they are implemented in terms of coefficients a_i that quantify the cost of each violation. These weights are established so as to maintain the hierarchical model over the N vertices, and ensure that violations ranked higher in the hierarchy are always prioritized over those lower down. CoP violations are treated with the highest priority and are given the biggest coefficient a_{cop}, geometric validity is quantified by $a_g = a_{cop}/N$, the cost of not satisfying negative forces is weighted by $a_{nf} = a_g/N$ and kinematic violations are weighted by $a_k = a_{nf}/N$.
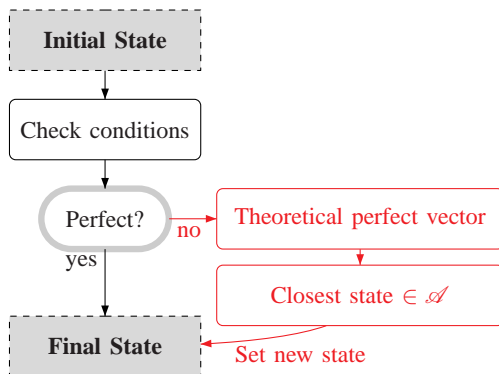

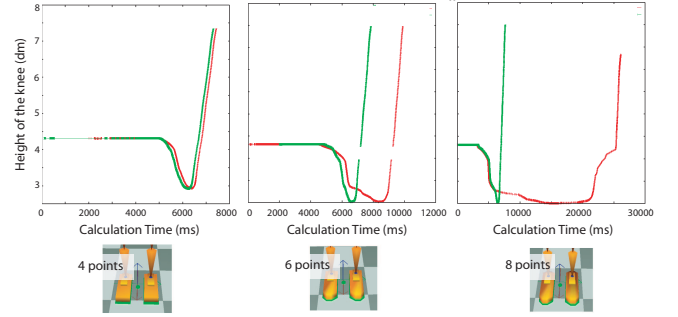Fig. 7.   Control flow of the one-step method

Fig. 8.   Comparison of calculation-times using the one-step model (green) and the exact method (red) during a jumping motion performed with foot shapes of varying complexity

The distance to the perfect vector can then be defined in terms of the violations weighted by the corresponding coefficients. This is calculated for each solution $\vec{s} \in \mathscr{A}$. The best vector $\vec{b}$ minimizes this distance, thus reducing the overall impact of problematic vertices, and satisfies:

$$\vec{b} = \min_{s \in \mathscr{A}} d(\vec{p}, \vec{s}) = \min_{s \in \mathscr{A}} \sum_{i=0}^{N-1} a_i \left| p_i - s_i \right|.$$

The configuration search therefore only involves calculating a limited number of multiplications and additions. The computational load of these operations is similar at each time step, and the resulting simulated motion is obtained more quickly. Moreover, adding one extra point to the polygon has little influence on the required calculation-time. An example is shown in Figure 8, where the benefit of this method is clear for simulations involving complex polygonal feet, during a motion in which the robot first bends and then jumps.

V. VALIDATION OF GROUND FORCES

We performed a number of experiments in order to validate the presented method and confirm its reliability. Simulated motions were compared with actual measurements from an AMTI OR6-7 force plate (which has a resolution of approximately 1N and a sensitivity of 1.7mV). Data was recorded for two types of motion. One involved a human standing on two platforms (one foot on each) and performing a hip-swaying movement, which was compared with similar motion performed in simulation by the full-sized humanoid robot, CB. The other involved a real small-sized robot (Hoap-2) performing a sequence of squatting motions, which were compared with their simulated counterparts.

A. Comparing Human Motion with Simulated Motion

The first experiment was intended to verify the human-like behavior of the simulated humanoid, and hence validate our model's functionality as an accurate tool for studying human/humanoid locomotion. The ground reaction forces experienced by a simulated human-sized robot with 50 degrees of freedom were analyzed and compared with recordings obtained from a real person, both performing a hip swaying motion.

The human subject wore solid rectangular 'robot shoes' identical to the robot's own foot surfaces, in order to properly replicate any effects related to the shape of the feet. The motion began from a centrally-balanced standing position, and the weight was gradually shifted to one side until the CoG was vertically aligned with the left leg and the right foot rested on its edge. The pressure applied by each foot was monitored and its profile was compared to that during simulation in which the same external conditions were imposed (friction, weight, angles of separation between the legs, speed and final state). The results in Figure 9 show the forces aligned with the three axes of the foot frame, and demonstrate a close correlation in each direction.

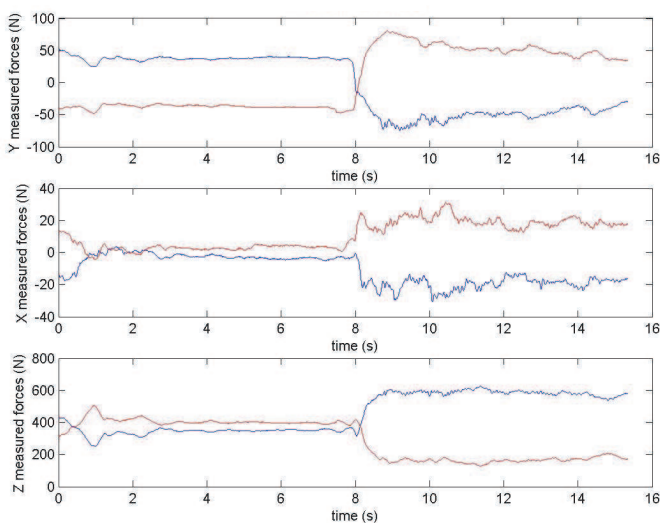**Simulated Ground Forces**
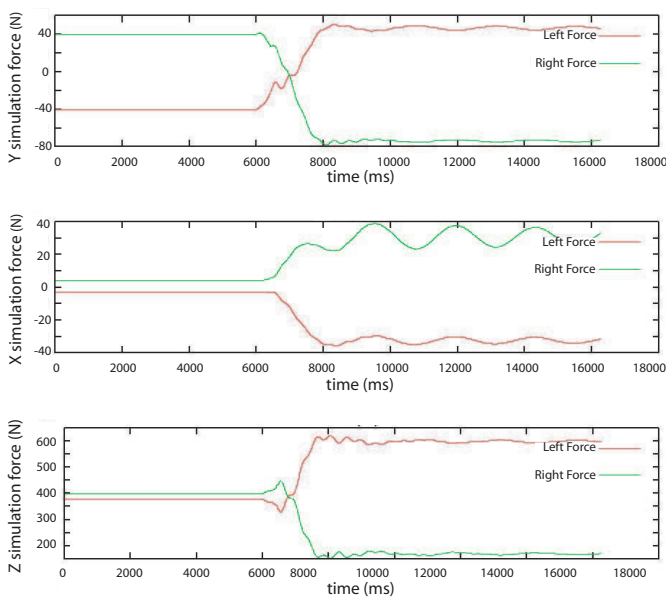
Real Human Ground Forces

Fig. 9. Ground Reaction Forces during a hip-swaying movement. The top graphs show forces along the *Y-axis* (sideways) and correspond to sideways friction forces that prevent the legs from separating. The *X-axis* forces are shown in the center graphs. The bottom graphs show the *Z-axis* reaction forces, and it may be seen that the weight shifts from its position in the initial state, which is equally distributed over both legs, to a final distribution in which it is mainly applied on the left side -the supporting foot.
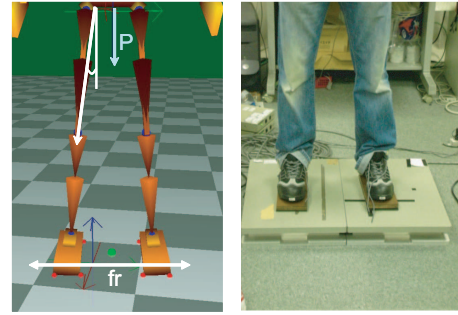

Fig. 10. Initial pose in the first experiment

Some behavioral differences may be observed regarding the stabilization periods required after the lateral sway is completed. In the case of the robot, these oscillations are governed by the proportional derivative (PD) controller which yields trigonometric transitions that differ from the gentle sway observed with the human. The results emphasize the fact that such characteristics differ, and prompts the refinement of biologically inspired controllers.

B. Comparing Robot Motion with Simulated Motion

The second validation involved using an actual humanoid robot and was aimed at more specifically validating the characteristics of ground contact.

The robot performed several squatting and standing motions on a single force plate at increasingly faster speeds, thus scaling the effect of inertia on the ground reaction force. The recordings were filtered using a Butterworth 20Hz low-pass filter of order 5 in order to remove noise in the force sensor readings and the resulting profiles were compared with the corresponding simulated counterparts. Exactly the same open-loop controller was employed for both the real and simulated robots. The resulting *Z-axis* forces are shown in Figure 11 for two consecutive squatting movements taking 1.25s and 1s respectively.

These results confirm the accuracy of the inertial and kinematic parameters used for the Hoap-2 simulation model, and demonstrate the precision of our ground contact simulation.

C. Empirical measurements

In addition to validating the simulator, empirical measurements were also made in order to tune the contact model. In particular, the static friction between various surfaces and the robots' feet were performed so that the Coulomb friction model could be properly tuned in the simulation environment, ensuring identical conditions in both real and simulated environments.

Various pairs of surfaces were tested, such as the force plate against the plastic surface of the Hoap-2 robot's feet, and soles of the CB robot's feet which were also used in the 'robot shoes' worn by the human subject. A mechanism was constructed which allowed these pairs of materials to be easily matched, loaded with several weights, and pulled tangentially until they began to slip.
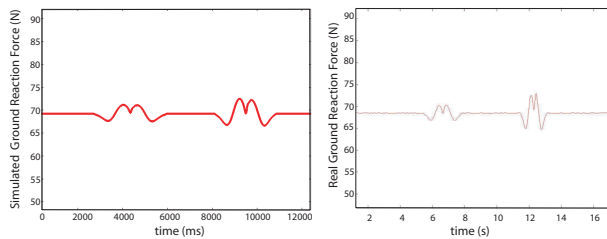
Fig. 11. Comparison of ground reaction forces during two squatting motions performed at different speeds by the small-sized humanoid 'Hoap2'.

The forces required to overcome static friction was thus retrieved for different weights and the corresponding friction coefficient was estimated by means of a linear regression, revealing coefficients of 0.13 for the plastic robot feet ($r^2 = 95, 83\%$) and 0.45 for the 'robot shoes' ($r^2 = 99, 25\%$).

VI. CONCLUSIONS AND FUTURE WORKS

This paper presented a technique for accurately and efficiently modelling ground contact during the dynamic simulation of humanoids. The method is based on an enumeration and selection of possible contact configurations, which are enforced using motion constraints. The constraints were enforced using Lagrange multipliers, although other mechanisms may be possible. Candidates configurations are evaluated according to a prioritized set of rules defining valid contact in terms of geometric and kinematic characteristics, ground reaction forces, and specific complementary information related to biped ground contact such as the center of pressure of each foot.

Unlike other methods, our technique avoids polyhedral approximations of the friction cone and handles static and dynamic Coulomb friction exactly. It can be applied under any dynamics formulation and is therefore compatible with precomputed dynamics formulations which may be subjected to symbolic optimization to improve computational efficiency, hence providing a straightforward solution procedure which is easier to implement than an LCP approach.

Our model yields realistic ground forces and has been validated against real measurements obtained from two force-plates on which motions were performed by both humans and humanoid robots, demonstrating high fidelity replication of motions involving significant inertial interactions. Friction responses were also empirically tuned.

The algorithm was constructed with humanoid simulation in mind, and exploits the relatively simple nature of a biped with polygonal feet to improve computational efficiency. The speed of the method degrades for more complex feet, and the number of possible contact configurations increases geometrically with the number of legs so the method may not be suitable for models with a large number of links contacting the ground. An accuracy sacrificing efficiency modification was however designed, which is implemented as an alternative configuration selection process. While the exact method guarantees that the correct configuration is selected at every time-step, this modified 'one-step' model provides a probabilistic alternative yielding faster results.

In future work we intend to investigate the limits of the 'one-step' trade-off in terms of model complexity versus accuracy, by varying the complexity of the feet and the number of feet, and to investigate the benefits of precompiling a dynamics engine for each possible contact state. The latter investigation is motivated by the fact that explicitly incorporating the constraints into the simulation improves the efficiency of the dynamics calculations by avoiding constraint resolution completely. The alternative simulation models may even be compiled as separate processes, loaded from permanent storage as necessary, because it is likely that many configurations will not occur during typical behavior.

REFERENCES

[1] Chris Atkeson, J.G. Hale, Frank Pollick, Marcia Riley, Shinya Kotosaka, Stefan Schaal, Tomohiro Shibata, Gaurav Tevatia, Ales Ude, Sethu Vijayakumar, and Mitsuo Kawato. Using humanoid robots to study human behaviour. *IEEE Intelligent systems Magazine, special issue on humanoids*, 15(4), July-August 2004.

[2] Gordon Cheng, Sang-Ho Hyon, Jun Morimoto, Ales Ude, Joshua G. Hale, Glenn Colvin, Wayco Scroggin, and Stephen C. Jacobsen. CB: a humanoid research platform for exploring neuroscience. *Advanced Robotics*, 21(10):1097–1114, 2007.

[3] Carlos Canudas de Wit. On the concept of virtual constraints as a tool for walking robot control and balancing. *Annual Reviews in Control*, 28(2):157–166, 2004.

[4] Joshua G. Hale, Benjamin Hohl, Sang ho Hyon, Takamitsu Matsubara, Eduardo Martin Moraud, and Gordon Cheng. Highly precise dynamic simulation environment for humanoid robots with a transparent interface. *Advanced Robotics, Special Issue on Humanoid Technologies and Systems*, 22(9), July 2008. (Submitted).

[5] Fumio Kanehiro, Hirohisa Hirukawa, and Shuuji Kajita. OpenHRP: Open architecture humanoid robotics platform. *I. J. Robotics Res.*, 23(2):155–165, 2004.

[6] Fumio Kanehiro, Natsui Miyata, Shuuji Kajita, Kiyoshi Fujiwara, Hirohisa Hirukawa, Yoshihiko Nakamura, Katsu Yamane, Ichitaro Kohara, Yuihiro Kawamura, and Yoshiyuki Sankai. Virtual humanoid robotic platform to develop controllers of real humanoid robots without porting. In *Proceedings of the 2001 IEEE/SRJ International Conference on Intelligent Robots and Systems*, pages 1093–1099, October 2001.

[7] Dinh Duy Le, Ronan Boulic, and Daniel Thalmann. Integrating age attributes to virtual human locomotion. *International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 34(5).

[8] Brian Mirtich. *Impulse-based Dynamic Simulation of Rigid Body Systems*. PhD thesis, University of California, Berkeley, 1996.

[9] Philippe Sardain and Guy Bessonnet. Forces acting on a biped robot. Center of pressure - Zero moment point. *IEEE Transactions on Systems, Man and Cybernetics, Part A*, 34(5):630–637, 2004.

[10] Harald Schmidl and Victor J. Milenkovic. A fast impulsive contact suite for rigid body simulation. In *IEEE trans. on Visualization and Computer Graphics*, volume 10, pages 189–197, 2004.

[11] Harold C. Sun and Dimitris N. Metaxas. Automating gait generation. In *SIGGRAPH*, pages 261–270, 2001.

[12] Miomir Vukobratovic and Branislav Borovac. Zero-moment point - Thirty five years of its life. *International Journal of Humanoid Robotics*, 1(1):157–173, 2004.

[13] Katsu Yamane and Yoshihiko Nakamura. Dynamics computation of structure-varying kinematic chains for motion synthesis of humanoid. In *Proc. IEEE int. Conf. on Robotics and Automation*, pages 714–721, 1999.

[14] Katsu Yamane and Yoshihiko Nakamura. Dynamics filter - Concept and implementation of on-line motion generator for human figures. *IEEE Transactions on Robotics and Automation*, 19(3):421–432, 2003.