2007 IEEE International Conference on
Robotics and Automation
Roma, Italy, 10-14 April 2007

A Robust Quadruped Walking Gait for Traversing Rough Terrain

Dimitris Pongas[1], Michael Mistry[1], &,Stefan Schaal[1,2]

[1]*Computer Science & Neuroscience, University of Southern California, 3641 Watt Way, Los Angeles, CA 90089-2520*
[2]*ATR Computational Neuroscience Laboratory, 2-2-2 Hikaridai, Seika-cho, Soraku-gun, Kyoto 619-02*
pongas@usc.edu, mmistry@usc.edu, sschaal@usc.edu
http://www-clmc.usc.edu

***Abstract* – Legged locomotion excels when terrains become too rough for wheeled systems or open-loop walking pattern generators to succeed, i.e., when accurate foot placement is of primary importance in successfully reaching the task goal. In this paper we address the scenario where the rough terrain is traversed with a static walking gait, and where for every foot placement of a leg, the location of the foot placement was selected irregularly by a planning algorithm. Our goal is to adjust a smooth walking pattern generator with the selection of every foot placement such that the COG of the robot follows a stable trajectory characterized by a stability margin relative to the current support triangle. We propose a novel parameterization of the COG trajectory based on the current position, velocity, and acceleration of the four legs of the robot. This COG trajectory has guaranteed continuous velocity and acceleration profiles, which leads to continuous velocity and acceleration profiles of the leg movement, which is ideally suited for advanced model-based controllers. Pitch, yaw, and ground clearance of the robot are easily adjusted automatically under any terrain situation. We evaluate our gait generation technique on the Little-Dog quadruped robot when traversing complex rocky and sloped terrains.**

Index Terms – quadruped locomotion, static walk, crawl gait, COG trajectory, rough terrain, internal-model control

I. INTRODUCTION

Traversing rough terrain with carefully controlled foot placement and the ability to clear major obstacles is what makes legged locomotion such an appealing, and at least in biology, highly successful concept. Obviously, a robust balancing controller is the prerequisite for legged locomotion, and it is usually characterized by a special control variable, for instance, the locations of the center-of-gravity (COG) in static walking, or the zero-moment-point (ZMP) in dynamic walking. In many approaches to legged locomotion, the planning of this control variable is the crucial primary step of realizing the locomotory gait[1], and converting this more abstract plan into a particular realization in configuration space is considered the secondary step [e.g., 3, 4, 5]. In this paper we address how to generate a robust COG trajectory for a quadrupedal walking robot, and how to convert this plan into an appropriate joint-space walking pattern. What sets our approach apart from the large number of previous projects on similar topics [e.g., for a comprehensive review, see 6] is that we wish to adjust the planned COG trajectory continuously in response to the current movement of the feet of the robot, and that we wish to accomplish this adjustment in a very smooth, i.e., twice differentially, way. The advantages of this smooth adjustable COG trajectory generation are twofold. First, smooth movement is in general desirable when climbing rough terrain, as any form of jerkiness can lead to slipping and a loss of stability. Second, if the COG trajectory has continuous acceleration profiles, the joint trajectories will have continuous acceleration profiles, too. This property allows us to employ advanced model-based controllers for realizing the desired robot motion, e.g., inverse dynamics controllers or operational space controllers [e.g., 7, 8], which will generally achieve higher passive compliance of the robot motion due to reduced negative feedback gains. High passive compliance also contributes to an increased robustness when walking over rough terrain as unforeseen perturbations can be rejected in a passive way.

In the following, we will first introduce our new approach for creating a continuously adjustable, smooth COG trajectory. Afterwards, we discuss the combination of this

Figure 1: Little-Dog, a 3kg heavy, 40 cm long quadruped robot with 12 DOFs, produced by Boston Dynamics Incorporation.

[1] A noteworthy different concept of generating legged locomotion has been explored in the context of passive dynamic walking ([1]T. McGeer, "Passive dynamic walking," *International Journal of Robotics Research*, vol. 9, pp. 633-643, 1990, [2]S. Collins, A. Ruina, R. Tedrake, and M. Wisse, "Efficient bipedal robots based on passive-dynamic walkers," *Science*, vol. 307, pp. 1082-5, 2005.), which, however, will not be pursued in this article any further.

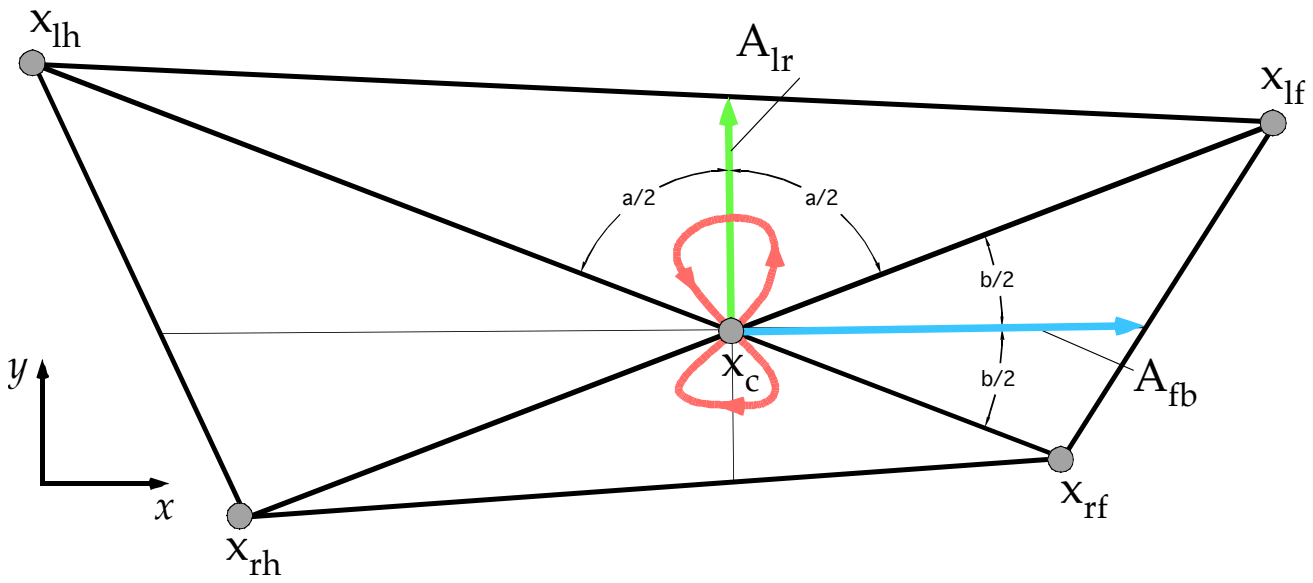

Figure 2: Top-down view of the four feet of the quadruped, where top-down is defined along the gravity vector. The desired COG trajectory is indicated by the figure-8 pattern and is computed from the location of the four feet in the x-y plane. $\mathbf{x}_{lf}$ denotes the location of the left-front foot, $\mathbf{x}_{rf}$ is the right front foot position, and $\mathbf{x}_{lh}$ and $\mathbf{x}_{rh}$ are the hind leg positions. A_{lr} and A_{fb} are the left-right swing and forward-backward swing amplitudes vectors.

COG trajectory with a static walking pattern, and the realization of the COG trajectory based on a series of 5th order splines. We present evaluations in actual robot experiments with the Little-Dog robot (Figure 1) in traversing a complex rocky terrain.

II. COG-Trajectory Formation

For statically stable walking, the COG of the quadruped should always stay in the support polygon of the current stance legs. We follow the ideas of [9, 10] to use a sinusoidal sway in the longitudinal and sideway direction to accomplish the COG trajectory. However, we wish to parameterize this COG trajectory purely based on the current positions of the feet of the robot and not as a simple fixed sinusoidal trajectory with pre-defined amplitude, as such a non-adaptive trajectory can violate static stability when climbing rough terrains, i.e., when foot placements are chosen irregularly.

The basic idea of our approach is depicted in Figure 2. When viewed in a top-down fashion along the gravity vector, at any moment of time, the four feet of the robot form a polygon, and the intersection $\mathbf{x}_c$ of the two diagonals of this polygon is the anchor for the COG trajectory. Note that all vector variables in Figure 2 are just two-dimensional in the *x-y* plane, and that for our initial considerations, it is irrelevant that one of the legs is the swing leg and its foot may actually be moving in the air. We choose the bisector vectors of the angle *a* and the angle *b* (see Figure 2) as the direction/amplitude of motion for the side-sway and the forward-backward-sway, respectively. More formally, we can write this as

$$\mathbf{x}_{cog} = \mathbf{x}_c + \alpha_{lr}\mathbf{A}_{lr}\sin\left(\omega t + \frac{\pi}{2}\right) + \alpha_{fb}\mathbf{A}_{fb}\sin(2\omega t) \qquad (1)$$

where $\omega = 2\pi f$, and f is the frequency of the walking pattern, and α_{lr} and α_{fb} denote the fraction of the amplitude vectors $\mathbf{A}_{lr}$ and $\mathbf{A}_{fb}$ that the COG should travel. The forward-backward motion needs to move at double frequency (see below). When the four feet of the robot are not moving, the COG trajectory follows the figure-8 pattern depicted in Figure 2, for a choice of $\alpha_{lr} = 0.5$ and $\alpha_{fb} = 0.1$. The vectors $\mathbf{A}_{lr}$ and $\mathbf{A}_{fb}$ are easily computed from the triangle geometry in Figure 2, i.e., purely the knowledge of $\mathbf{x}_{lf}, \mathbf{x}_{rf}, \mathbf{x}_{lh}, \mathbf{x}_{rh}$.

Next, it can easily be verified that Eqn.(1) is twice differentiable if $\mathbf{x}_{lf}, \mathbf{x}_{rf}, \mathbf{x}_{lh}, \mathbf{x}_{rh}$ are twice differentiable, too. Normally, three of the feet are in stance phase and have zero velocity and acceleration, and only one leg, the swing leg, moves smoothly towards another foothold. If this swing trajectory is twice differentiable, i.e., has continuous accelerations, the COG trajectory will remain smooth, too. Thus, the figure-8 pattern of the COG trajectory can be adjusted continuously even if one of the legs is swinging. The stability margins of the COG trajectory are continuously updated with the geometry of the four feet, such that a stable walking gait will be realized at all times, unless a support triangle becomes degenerate — this special case needs to be avoided by the foot placement planner, and is thus not addressed in this paper.

Finally, we need to address how the walking cycle is coordinated with the COG trajectory. We consider the classical left-front, right-hind, right-front, left-hind sequence of walking, depicted in Figure 3. We also included the possibility of a duty factor of more than 0.75 but allowing a four-leg stance phase. Thus, in the quarter cycle that one leg is the swing leg, it goes through an initial Four-Leg Stance phase, then a Lift-Off phase, a Touch-Down phase, and fi-

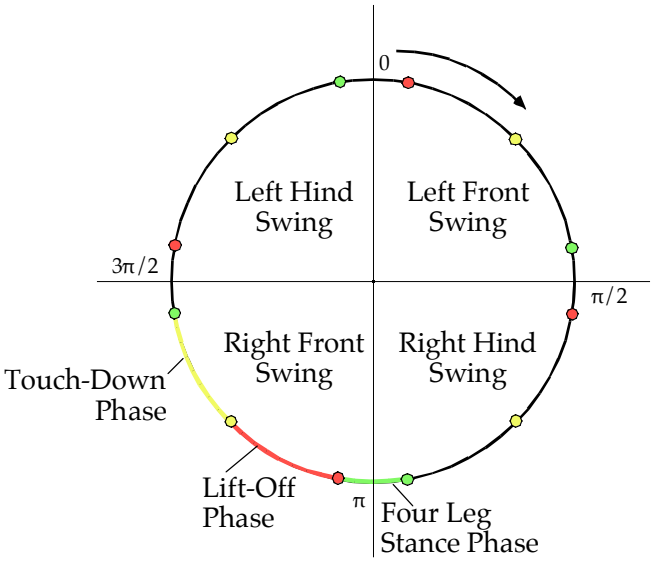

Figure 3: Swing phases of the walking pattern depicted on one run through the entire walking cycle.

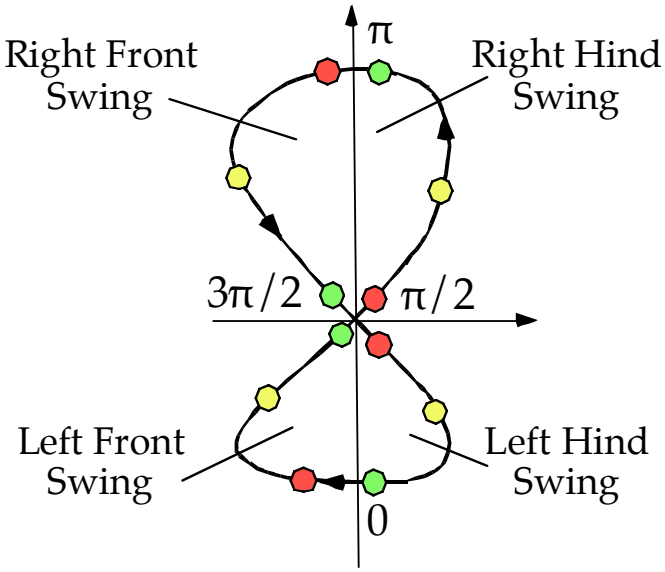

Figure 4: Swing phases of the walking pattern depicted on the COG trajectory of Figure 2.

nally a Four-Leg stance phase again. For the remaining ¾ cycle of the walking pattern, it will just remain a stance leg (which is not depicted in Figure 3. The behavior phase timing in Figure 3 is in accordance to the COG timing in Eqn.(1), i.e., we can draw the phases of Figure 3 equally well on the figure-8 pattern of the COG, as shown in Figure 4. When comparing this figure with Figure 2, it should be apparent that the COG is always in the appropriate support triangle during each leg's swing phase.

III. Yaw, Pitch, and Ground Clearance

The previous section only addressed the COG trajectory in the *x*-*y* plane relative to the current foot positions. We also need to fill in yaw, pitch, roll, and the z-coordinate (ground clearance) of the COG. These variables are normally under control of a planner, but in our implementation,

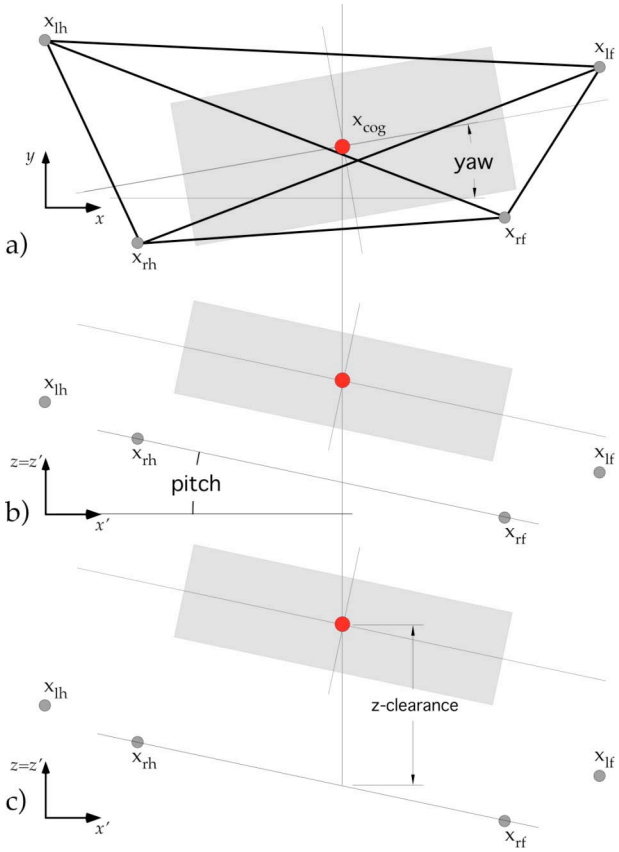

Figure 5: a) Yaw, b) pitch, and c) round clearance adjustment of the robot after determination of the COG.

we reduced the number of explicit planning variables as much as possible by creating an auto-adjustment for the pitch angle and the ground clearance. The procedure we follow is given in Figure 5 and consists of three steps, each of which guarantees that the *x*-*y* location of the COG remains unchanged, i.e., only rotations about the COG and translations along the gravity vector (z-axis) are permitted:

1. The body of the robot is rotated about the z-axis with a given desired yaw angle in world coordinates from the planner. This rotation results in a new body-centered coordinate system *x'*-*y'*-*z'* (Figure 5a).
2. The body pitch angle is computed as the angle between the line through the lowest front and the lowest hind foot and the *x'* axis (Figure 5b).
3. Finally, the COG is shifted along the *z'* axis to achieve a given desired z-clearance above the ground (Figure 5b). If after this z-translation, any of the robot legs is stretched beyond the feasible workspace of the robot, the z-translation is reduced such that all legs can reach their footholds.

IV. Conversion of COG Trajectory to Configuration Space

At this point, we arrived at a specific body pose of the robot relative to the current foot placements, and, over time, we create a smooth COG trajectory. There are many ways how the COG trajectory can be converted into appropriate

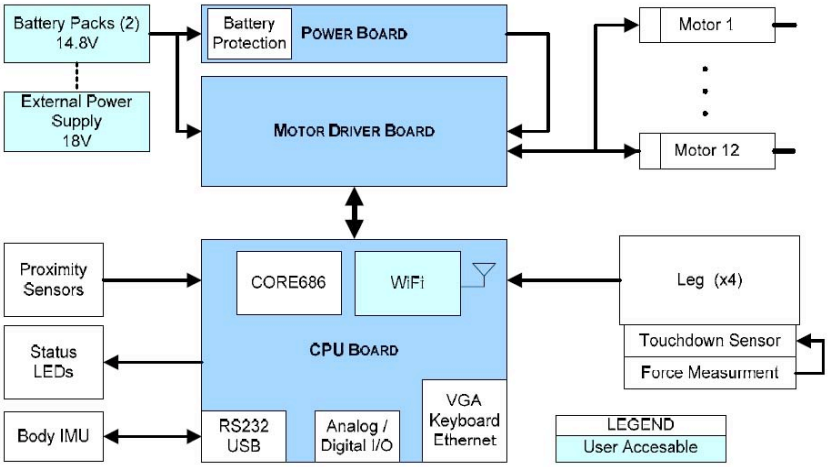

Figure 6: Little Dog Hardware and Control Setup

joint-space movement. In our current implementation we chose a 5th order spline-based approach. Each of the nod-points along the circle of Figure 3 represents a 3D spline node in body-centered coordinates for each leg, such that there are 12 nodes total per leg. If we assume a duty factor df=0.75+ε, the spline nodes of the left-front swing are timed at:

$$\begin{aligned} \text{Lift-off:} \quad & \varepsilon/2 \\ \text{Touch-Down:} \quad & \pi/4 \\ \text{Four-Leg Stance:} \quad & \pi/2-\varepsilon/2 \end{aligned} \tag{2}$$

for a lift-off and touch-down phase of equal duration. The next swing leg, the right hind leg, has π/2 offset w.r.t. (2), the right front leg has π offset, and the left-hind leg has 3π/2 offset. While actually only three spline nodes would be required to code for a smooth swing phase of a leg and the corresponding stance phase, we added 9 spline nodes during the stance phase of each leg at the same times as needed to code for the other legs' swing phases. This strategy provides an opportunity to change the walking pattern at every quarter of the walking cycle. When one leg is about to become the swing leg, the planner selects its next foothold, and given the timing of the spline nodes, the next spline nodes for the touch-down phase, four-leg stance phase, and lift-off phase are determined. The location of the spline node at the beginning of the touch-down phase is interpolated half-way between the current foot position and the desired foot hold, and its z-position is increased by a step-height parameter, which codes for how high the swing leg is supposed to lift-off during the swing phase. All other spline nodes are located at the desired foothold of the leg.

Thus, at every spline node, we have all the information required to compute the COG position as explained above, and additionally we obtain the COG velocity and acceleration. The footholds of all feet at a particular spline node are then converted into a body centered coordinate system, which results in a relative position, velocity, and acceleration of each foot w.r.t.. the body — exactly what is needed as boundary conditions for a 5th order spline. It should be noted that all stance feet share the same velocity and acceleration, as the stance legs cannot move relative to each other.

The spline nodes prescribe thus the 3D trajectory of each foot in body coordinates. As our robot has only 3DOFs per leg, an analytical inverse kinematics computation allows us to convert the foot movement into joint movement, which can be tracked by a PD controller with floating base inverse dynamics control [11].

V. EVALUATIONS

Our testbed, the Little-Dog robot (Figure 1) is a small quadruped robot, designed by Boston Dynamics Incorporation (BDI, Cambridge, MA). It is about 0.3m long, 0.18m wide, and 0.26m tall, with a total weight of approximately 2.5kg. Little-Dog's 3DOF legs are powered by electric motors. Each actuator can be controlled at 100Hz by either PD control or a torque control loop from a Linux host computer. There is an onboard computer that performs sensing and actuator control at 500Hz, and that communicates with the host computer through a wireless connection at 100Hz.

Little-Dog has four contact sensors (one on each foot), a proximity sensor in the head, position sensors to measure joint angles, and an IMU for body orientation and acceleration. Little-Dog is powered by onboard batteries or can be tethered to an external power source. The absolute world position and orientation are provided by an external motion capture system (VICON). Figure 6 and Figure 7 illustrate the control setup and the schematic of the kinematics of the robot.

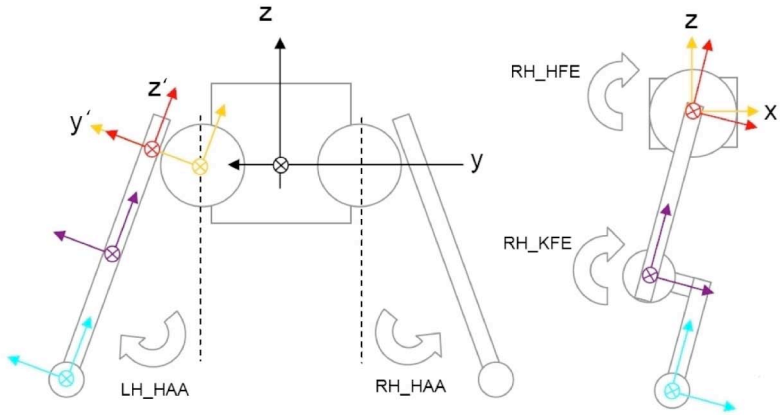

Figure 7: Leg kinematics of Little-Dog

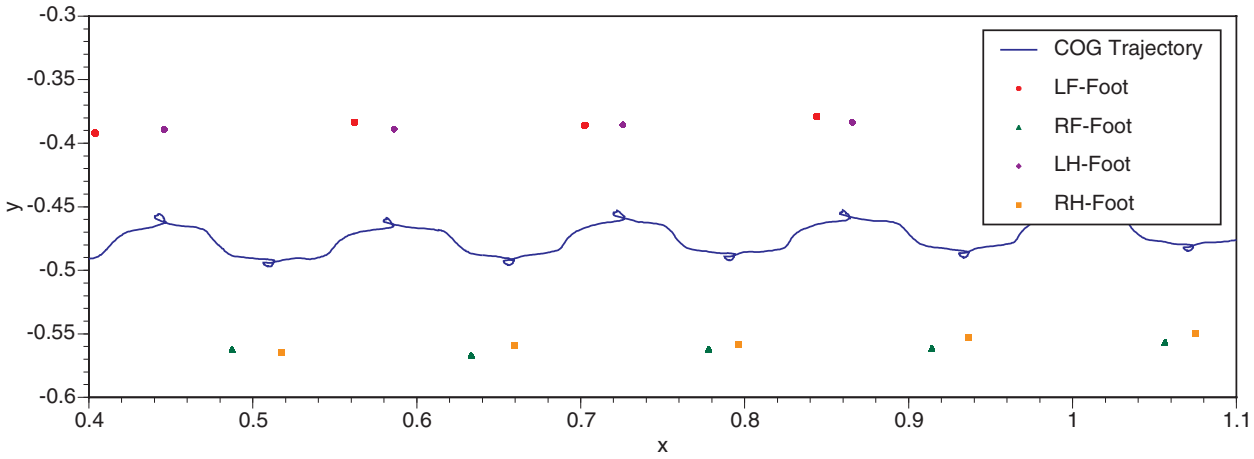

Figure 8: COG Trajectory and foot placement of Little Dog when walking on even terrain.

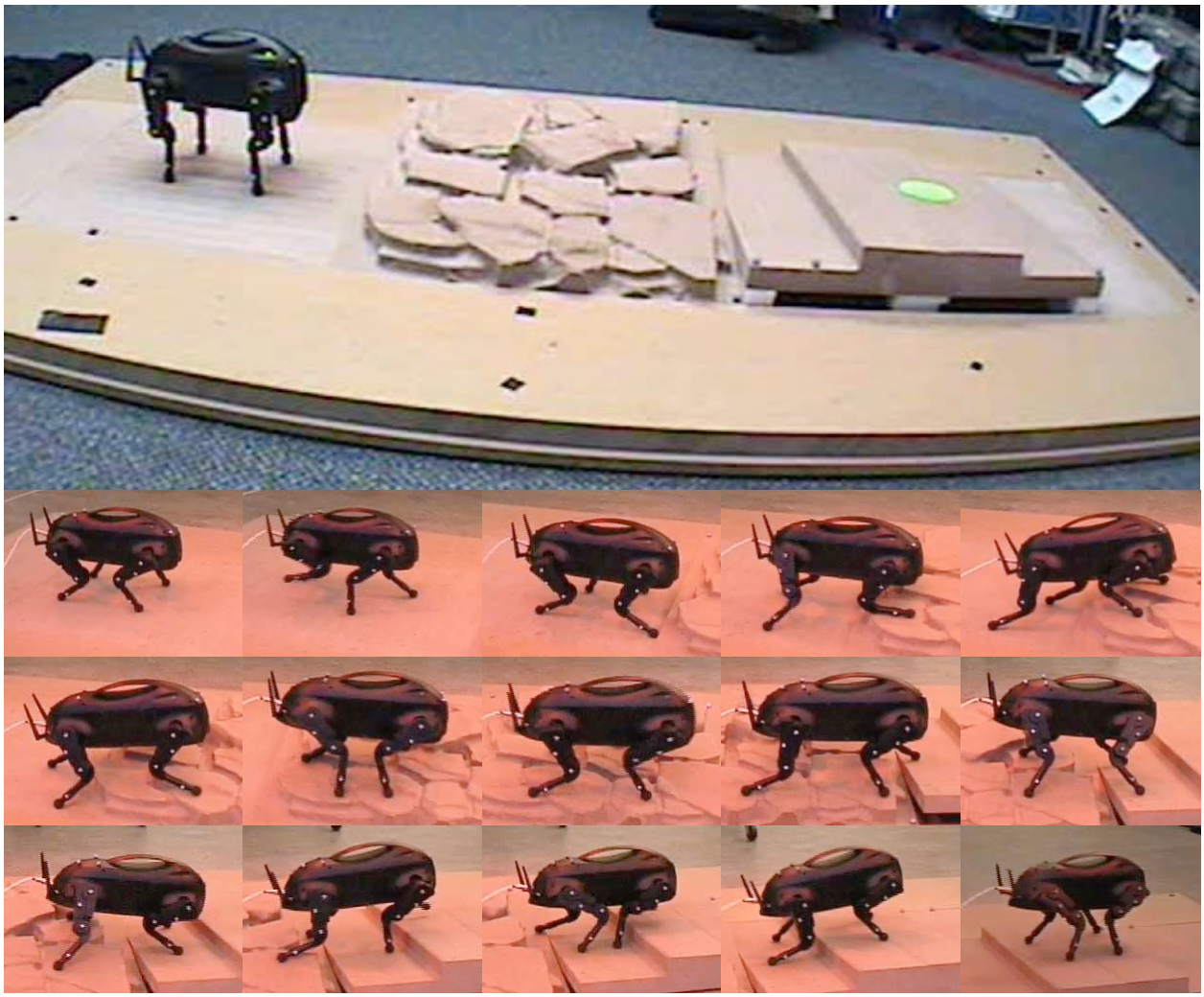

Figure 9: Top: Overview of rough terrain. Bottom: Sequence of snapshots of Little-Dog while traversing the terrain.

We implemented our walking gait pattern generator on Little-Dog, and tested it in flat terrain forward walking, and also on a complex tilted rocky terrain walking task. The foot placement planer was generated manually, and is not described in this paper.

Figure 8 illustrates the COG trajectory and the foot placements of a straight walking pattern in the top-down view (*x*-*y* plane) of the robot. This data was obtained with a 3 second period for one full walking cycle, a duty factor of 0.8, $\alpha_{lr} = 0.5$, $\alpha_{fb} = 0.1$, and 0.14m ground clearance dur-

ing walking. As can be seen in Figure 8, the COG trajectory swings appropriately sideways, and from the tiny loops of the trajectory, the forward-backward swing at double frequency can be recognized. Subjectively, the walking pattern looks very smooth and natural.

Figure 9 demonstrates snapshots of the robot walking over rough terrain. The artificial terrain boards used in this test were generated from a laser scan of actual terrains. With the help of a VICON motion capture system, the pose of the robot can be determined with high accuracy. Each terrain board has a special combination of VICON markers that allow inferring its identity, its location and orientation, and to associate an *x-y-z* raster file that provides precise information about the terrain height at every location (on a 1mm grid). Thus, the perceptual component for walking over rough terrain is highly simplified in this setup, such that maximal emphasis can be given to the motor control research components. We developed a foot placement planner that selects the next foothold for each swing leg according to several features of the terrain, e.g., slope, evenness of terrain, potential impacts of the knees, etc. About 500 potential foothold are evaluated for each swing leg, and the one receiving the highest score is selected, using a heuristic scoring system. Subsequently, the walking pattern of the robot is adjusted to hit the chosen foothold, which also entails the need to adjust the COG trajectory. The series of images in Figure 9 demonstrates that our adjustable walking pattern generator performs well on this rough terrain. The video attached to this paper provides a more life-like impression of the robot's motion.

VI. Conclusions

This paper presented a flexible walking pattern generator for quadruped walking over rough terrain. The key features of our gait generator were that is automatically and smoothly adjusts the COG trajectory of the robot according to the current foot placement pattern, such that a foot placement planner can select footholds with minimal consideration of the walking pattern. The smoothness, i.e., double differentiability, of the COG trajectory allowed us to obtain continuous acceleration and velocity profiles for the desired joint angle trajectories, which, in turn, enabled us to employ advanced model-based controllers to keep the feedback gains of the robot low and to achieve a high level of compliance. We believe that such compliant control is crucial in future legged locomotion approaches in order to allow robots to operate safely in human environments.

We evaluated our approach on an actual quadruped robot, the Little-Dog robot, for walking over rough terrain. Our walking pattern proved to generate very smooth and elegant motion with excellent balance control.

Future work will address replacing the spline-based movement approach with an operational space COG controller that simultaneously controls the COG trajectory and the trajectory of the swing leg. This approach will allow having continuous control of the robot balance, instead of only intermittent control as current realized by the spline method.

VII. Acknowledgements

This research was supported in part by National Science Foundation grants ECS-0325383, IIS-0312802, IIS-0082995, ECS-0326095, ANI-0224419, the DARPA program on Learning Locomotion, a NASA grant AC#98-516, an AFOSR grant on Intelligent Control, the ERATO Kawato Dynamic Brain Project funded by the Japanese Science and Technology Agency, and the ATR Computational Neuroscience Laboratories.

VIII. Literature

[1] T. McGeer, "Passive dynamic walking," *International Journal of Robotics Research*, vol. 9, pp. 633-643, 1990.

[2] S. Collins, A. Ruina, R. Tedrake, and M. Wisse, "Efficient bipedal robots based on passive-dynamic walkers," *Science*, vol. 307, pp. 1082-5, 2005.

[3] M. Vukobratovic and J. Stepanenko, "On the stability of anthropomoprhic systems," *Journal of Mathematical Bioscience*, vol. 15, pp. 1-37, 1972.

[4] M. Vukobratovic and B. Borovac, "Zero-moment point -- Thirty five years of its life," *International Journal of Humanoid Robotics*, vol. 1, pp. 157-173, 2004.

[5] S. Kajita and K. Tani, "Experimental study of biped dynamic walking," *Control Systems Magazine, IEEE*, vol. 16, pp. 13-19, 1996.

[6] F. Hardarson, "Stability analysis and synthesis of statically balanced walking for quadruped robots," in *Mechatronics Lab, Department of Machine Design*. Stockholm: Royal Institute of Technology (KTH), 2002.

[7] O. Khatib, L. Sentis, J. Park, and J. Warrent, "Whole-body dynamic behavior and control of human-like robots," *International Journal of Humanoid Robotics*, vol. 1, pp. 1-15, 2004.

[8] J. Nakanishi, M. Mistry, and S. Schaal, "Inverse dynamics control with floating base and constraints," presented at International Conference on Robotics and Automation (ICRA2007), submitted.

[9] K. Yoneda and S. Hirose, "Dynamic and static fusion gait of a quadruped walking vehicle on a winding path," *Advanced Robotics*, vol. 9, pp. 125-136, 1995.

[10] V. Hugel and P. Blazevic, "Towards efficient implementation of quadruped gaits with duty factor of 0.75," 1999.

[11] R. Featherstone, *Robot dynamics algorithms*: Kluwer Academic Publishers, 1987.