2014 13th International Conference on Control, Automation, Robotics & Vision
Marina Bay Sands, Singapore, 10-12th December 2014 (ICARCV 2014)

Continuous Cyclic Stepping on 3D Point-Foot Biped Robots Via Constant Time to Velocity Reversal

Donghyun Kim
Mechanical Engineering
University of Texas at Austin
Austin, Texas 78712
Email: dk6587@utexas.edu

Gray Thomas
Mechanical Engineering
University of Texas at Austin
Austin, Texas 78712
Email: gray.c.thomas@gmail.com

Luis Sentis
Mechanical Engineering
University of Texas at Austin
Austin, Texas 78712
Email: lsentis@austin.utexas.edu

***Abstract*—This paper presents a control scheme for ensuring that a 3D, under-actuated, point-foot biped robot remains balanced while walking. It achieves this by observing the center of mass (COM) position error relative to a reference path and re-planning a new reference trajectory to remove this error at every step. The Prismatic Inverted Pendulum Model (PIPM) is used to simplify behavioral analysis of the robot. We use phase space techniques to plan the COM trajectories and foot placement. While obtaining a stable path using this simplified model is easy, when applied to a real robot, there will usually be deviation from the expected path due to modeling inaccuracies. Although fully-actuated robots can reduce the deviation with relatively simple feedback control loops, when working with under-actuated robots, it is challenging to design such a feedback control loop. Our approach is based on continuous re-planning. By planning the path of the next step based on the observed initial error, we can find the proper landing location of each step. For each step we allocate sufficient time to avoid disturbances from the moment induced by the moving leg, which is not modeled in the PIPM. Our control scheme relies on the PIPM instead of the Linear Inverted Pendulum Model (LIPM) to enable non-planar COM motion, which is essential for rough terrain locomotion. We show simulation results that include full multi-body dynamics, friction, and ground reaction forces.**

I. Introduction

In this paper, we introduce a new robust locomotion re-planning strategy and implement a feedback control system for generating steps in point-foot bipedal robots. It uses elements of our previous work on Phase Space Planning [1] to generate stepping events, and Whole-Body Control [2] to generate feedback control policies for coordinating the robot's behavior.

Keeping point-foot biped robots balancing is a daunting task because of the lack of actuated ankles that could otherwise continuously balance the robots. Point foot robots must keep balance by swinging their feet to reach stabilizing contact positions. Such strategies are informally called "Control of Swing-Foot Placement" and have successfully been developed in 3D point-foot robots for quite some time. Control strategies using inverted pendulum dynamics for 3D stilt-type robots were developed as early as in [3]. Simpler strategies that could generate stable swing foot trajectories as a function of center of mass velocities were developed for 3D point-foot bipeds soon afterwards by [4]. Extensions to the simple stabilizing strategies have propagated with notable work to generate reactive 3D walking by [5]. In particular, the latter work provides continuous adaptations of the foot placement controller during the locomotion cycle as opposed to once-per-cycle adaptations done in previous work. Very recently, new extensions to the latter work and their implementation using virtual constraints have been implemented by [6] on the 3D underactuated robot ATRIAS [7].

The present work is motivated by the unique challenges of extreme dynamic locomotion, including the control and implementation of dynamic gaits using multicontact strategies (see [8] for our formulation of the Multicontact/Grasp Matrix that describes simple robot dynamics during arbitrary multicontact phases) and leaping between near vertical surfaces (see [9] for our planning strategy using phase space planning techniques). Above all, our interest lies in designing and implementing extreme dynamic gaits with any number of contacts and in any type of surfaces. Our line of work, dubbed Phase Space Planning, is along the lines of the work by [3] in that we use pendulum dynamics to stabilize the gaits. However, we differ from the previous work in various respects: (1) we use a prismatic inverted pendulum to generalize the arbitrary behavior of the robot's center of mass in 3D; (2) we use numerical methods to solve the nonlinear prismatic pendulum dynamics; (3) we plan contact transitions by finding boundary conditions in the phase plane of the center of mass due to neighboring contact states; and (4), our Phase Space Planning technique has been generalized to design maneuvers of the robot in between near vertical surfaces and leveraging multicontact dynamics in [9]. Although our final aim is to implement the aforementioned extreme maneuvers in bipeds, the goal of this paper is to develop a feedback control strategy that provides robustness to our Phase Space Planning method. Overall, we believe that model-based strategies such as using prismatic inverted pendulum dynamics or the Multicontact/Grasp Matrix are more suitable to be generalized to extreme maneuvers as they provide a framework for planning with any number of contacts and in any surface. For this reason we develop here a feebdack control strategy that can stabilize our phase space planning method and we demonstrate it first on a 3D point-foot biped on flat terrain.

Besides [3], there are other simple model-based control methods such as Preview Control [10] or Capture Point [11] which could potentially be leveraged to stabilize the 3D gait of

978-1-4799-5199-4/14/$31.00 ©2014 IEEE

Fig. 1. Hume a point-foot Bipedal Robot

point-foot biped. However, our goal is to develop a feedback controller for our general Phase Space Planning technique which ultimately aims at producing multi-contact and extreme gaits.

II. State Machine

The proposed control system uses a state machine to guide the robot through the discrete phases of stepping. Once per step, the state machine activates the online footstep re-planner, which decides the placement of the upcoming footstep. The state machine then determines the goals and constraints needed by the whole body controller to determine the motor torque. As can be seen in Fig. 2, the state machine divides the stepping

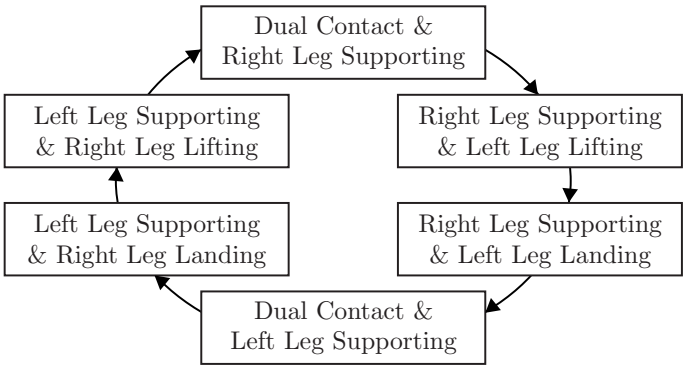

Fig. 2. State Machine for Continuous Stepping Motion

motion into six states of three types. The states are symmetric with respect to the supporting leg being either right or left. Thus, for every state, there exists a state with the leg roles reversed. The three types of states are dual contact, beginning of swing (lifting), and end of swing (landing).

There is only one path which transitions between the states. Beginning, as the simulation does, in either of the dual contact phases, a timer initiates the transition to the same-support-leg lifting phase, in which the swing leg follows a pre-defined lifting trajectory. At the halfway point of this trajectory, the planner determines a landing location and time of contact for the next footstep. The transition to the landing phase follows the completion of the lifting trajectory. In the landing phase, the foot moves toward the goal placement with a velocity based on the expected landing time. When the foot contacts the ground, the state machine switches leg roles, and the state transitions to the dual contact state opposite the starting state.

Summary of Variables	
$()_l$, $()_s$	Variables in the lateral, Sagittal planes
$i \in \{l, s\}$	Index to denote planes of motion
$\zeta_{i,p}$	Planned Foot Placement
$\zeta_{i,st}$	Current Stance Foot Location
x_i, $\dot{x}_i$	COM position, velocity
t_{sw}	The instant in time when a foot lands (switches)
t'_i	The time to velocity reversal
$\dot{x}_{i,apex}$	Minimum speed of COM during a skipping step
A	The set of kinematically reachable foot positions

TABLE I
Summary of Planner Variables

III. Online Planning Process

The footstep planner looks ahead one step into the future to determine the foot placement, and chooses that foot placement separately in the lateral and sagittal planes, which are defined with respect to the body's local frame at the beginning of each foot lift. Two problems – one in the lateral and one in the sagittal planes – are simultaneously solved assuming decoupled PIPM dynamics for each step [12]. The goal of the planner is to drive the COM into a cyclic motion, or similarly to select a foot placement, defined by $\zeta_{l,p}$, and $\zeta_{s,p}$, such that at some pre-defined constant times t'_l, and t'_s since the touchdown time, t_{sw}, the COM velocity will become zero (Figure. 3).

To enable dealing with variable height COM trajectories, the Prismatic Inverted Pendulum Model (PIPM) is used rather than the Linear Inverted Pendulum Model (LIPM). The use of this more complicated model determines the algorithmic and computational choices of the proposed controller as it does not permit analytic solutions or approximations with first order capture point dynamics as the LIPM does. The PIPM thus motivates the use of numerical integration.

A. Continuous Stepping Hypothesis

Let $z(x)$ represent a predefined center of mass height surface parameterized by the center of mass horizontal position, x, and let g be the magnitude of the acceleration due to gravity. Notice that to simplify notation, we are temporarily dropping the step index, $i \in \{l, s\}$. The following equation

holds according to the PIPM model,

$$\ddot{x} = \frac{g + \ddot{z}}{z}(x - \zeta_{st}). \tag{1}$$

In practice, it is inconvenient to handle the dynamics of z and x at the same time. Instead, parameterizing the dynamics by one variable (e.g. $z(x)$ instead of $z(t)$ and $x(t)$) is more reasonable when applying the PIPM in a control scheme. In particular, because the height surface is a function of x, we can write

$$\frac{dz}{dt} = \frac{dz}{dx}\frac{dx}{dt}, \tag{2}$$

$$\frac{d^2z}{dt^2} = \frac{d}{dt}\Big(\frac{dz}{dx}\Big)\frac{dx}{dt} + \frac{dz}{dx}\frac{d}{dt}\Big(\frac{dx}{dt}\Big), \tag{3}$$

$$\ddot{z} = \frac{d^2z}{dx^2}\dot{x}^2 + \frac{dz}{dx}\ddot{x}. \tag{4}$$

By plugging Equation (4) into Equation (1), we obtain,

$$\ddot{x} = \frac{g + \frac{d^2z}{dx^2}\dot{x}^2 + \frac{dz}{dx}\ddot{x}}{z}(x - \zeta_{st}), \tag{5}$$

$$z\ddot{x} = \Big(g + \frac{d^2z}{dx^2}\dot{x}^2\Big)(x - \zeta_{st}) + (x - \zeta_{st})\frac{dz}{dx}\ddot{x}, \tag{6}$$

$$\ddot{x} = \frac{g + \frac{d^2z}{dx^2}\dot{x}^2}{z - (x - \zeta_{st})\frac{dz}{dx}}(x - \zeta_{st}). \tag{7}$$

Now, the term $\ddot{z}$ is removed.

By numerically integrating the above equation, phase portraits can be obtained for the state $(x, \dot{x})$ and foot placements (ζ_{st}). In Figure 6 we depict various phase portraits both from our predictions using the above equations and from the actual simulations of a stepping behavior.

To show that the proposed planner enables the robot to enter a cyclic locomotion pattern, we observe the velocity,

$$\dot{x}(t'|\zeta) = \int_0^{t_{sw}+t'} \ddot{x}\, dt, \tag{8}$$

where the notation $\dot{x}(t'|\zeta)$ means the value of $\dot{x}$ at time t' for a foot placement value ζ, either consisting of the current stance foot, ζ_{st} or the candidate foot placement ζ_p being planned.

Proposition 3.1 (Cyclic Stepping): *If the phase plane trajectory of the COM always returns to the zero velocity line, and the planned foot location remains in the kinematically reachable set, then the stepping behavior is cyclic.*

Assumption 3.2 (Approximate PIPM Behavior): *If the swing foot moves slowly enough with respect to the COM, the robot dynamics can be approximated by the PIPM.*

With the above statements, we pose a hypothesis supporting the core idea of our planner.

Hypothesis 3.3 (Constant Time to Velocity Reversal): *A constant time t' can be found for all steps in a sequence such that a stepping cycle as defined in Proposition 3.1 is achieved.*

B. Skipping Steps

It is not always guaranteed that foot placements will be in the reachable workspace, A. When this problem happens we take a strategy of placing the foot such that the rule $\dot{x}_{apex} = 0.3\ \dot{x}(t_{sw}|\zeta_{st})$ is attained, thus effectively bounding the velocity. The speed $\dot{x}_{apex}$ is the minimum speed when the COM passes the apex of the stance foot, i.e. at $\dot{x}$ at $x = \zeta_{st}$ as defined in Table I.

C. Numerical Algorithms

In each plane of motion the planner solves a simple shooting problem. Again, if the foot placement is unreachable, we use a different velocity goal. When it is reachable, we use the method described in Algorithms 1 and 2 which aim at reversing the velocity at time t'. When unreachable, we use Algorithms 3 and 4 which aim at bouding the velocity $\dot{x}_{apex}$ as described above. The initial limits for bisection, a and b, are chosen to be close to the minimum and maximum reachable step locations.

Algorithm 1 Estimate of Foot Placement for Velocity Reversal

Require: $t', x(t_{sw}|\zeta_{st}), \dot{x}(t_{sw}|\zeta_{st}), a, b$
while error $> \epsilon$ **do**
  $c \leftarrow \frac{(a+b)}{2}$
  $\dot{x}(t') \leftarrow$ **Integration**$(c, t', x, \dot{x})$
  if $\dot{x}(t') > 0$ **then**
    $a \leftarrow c$
  else
    $b \leftarrow c$
  end if
  error $\leftarrow abs(\dot{x}(t'))$
end while
return $\zeta_p = c$

Algorithm 2 Time Based Integration $(\zeta, t', x, \dot{x})$

while $t < t'$ **do**
  $\ddot{x} \leftarrow \frac{g + \frac{d^2z}{dx^2}\dot{x}^2}{z - (x-\zeta)\frac{dz}{dx}}(x - \zeta)$
  $\dot{x} \leftarrow \dot{x} + \ddot{x}\Delta t$
  $x \leftarrow x + \dot{x}\Delta t + \frac{1}{2}\ddot{x}\Delta t^2$
end while
return $\dot{x}$

In both cases, the bisection search converges thanks to the monotonic relation between the foot location and velocity change. Using the equations,

$$\dot{x}(t'|\zeta_p) = \int_0^{t'} \frac{g + \frac{d^2z}{dx^2}\dot{x}^2}{z - (x - \zeta_p)\frac{dz}{dx}}(x - \zeta_p)dt,$$

we can easily derived the partial derivative,

$$\frac{\partial \dot{x}(t'|\zeta_p)}{\partial \zeta_p} = -\int_0^{t'} \frac{z(g + \frac{d^2z}{dx^2}\dot{x}^2)}{\Big(z - (x - \zeta_p)\frac{dz}{dx}\Big)^2}dt. \tag{9}$$

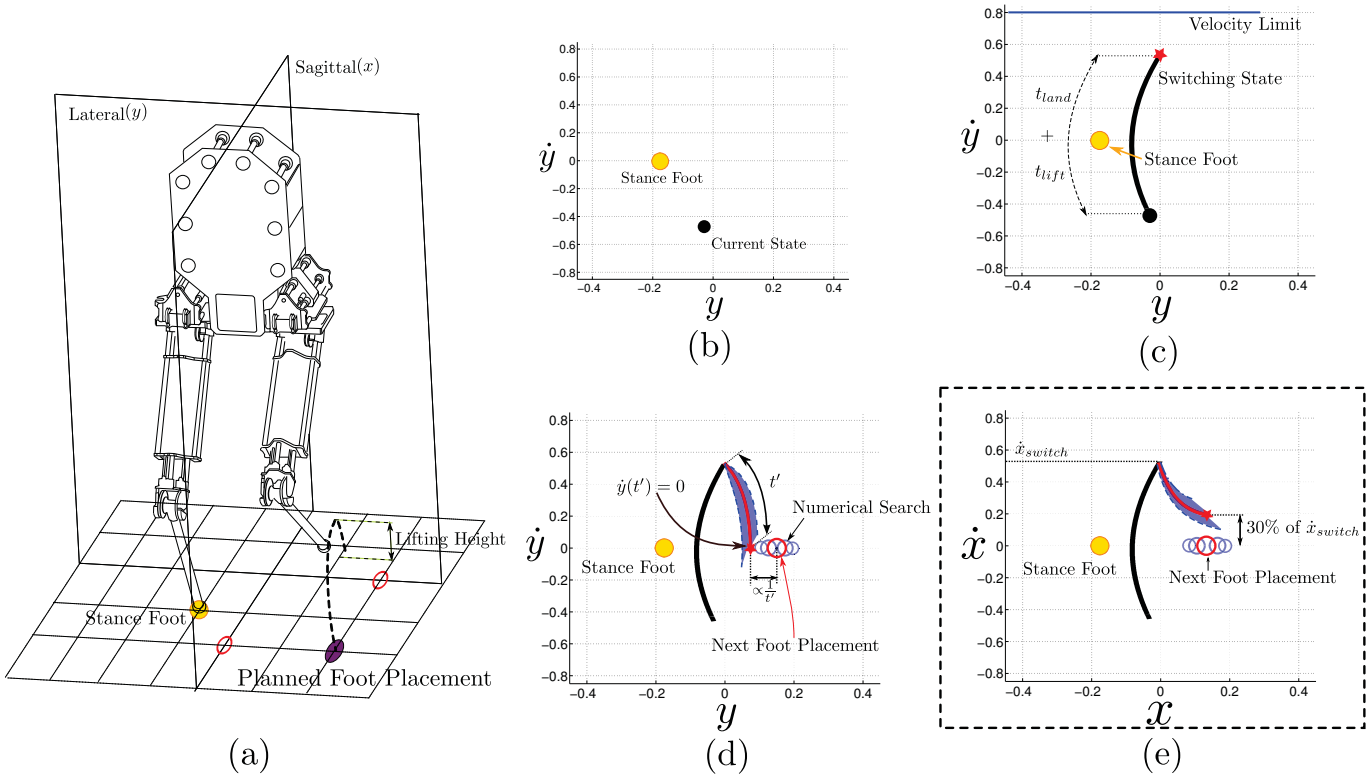

Fig. 3. Center of mass phase space diagrams illustrating the footstep placement algorithm. (a) The planner searches for the foot landing placement for each plane (lateral and Sagittal). When the foot reaches half of the swinging trajectory, the COM state and the position of the stance foot location are delivered to the planner. (b) We depict the current COM state and stance foot locations in phase space. The planner searches the landing location first in the lateral plane, y. (c) Based on PIPM dynamics, the position and velocity of the center of mass are projected into the future. (d) The planner searches for the foot placement that will make the COM velocity equal to zero at time t'. (e) If the planned foot placement is out of range, the robot makes one more step. This strategy is only allowed in the sagittal plane, x, to avoid hitting abduction/adduction joint limits on the hip.

Algorithm 3 Estimate of Foot Placement for Skipping Steps

Require: $\dot{x}_{apex}, x(t_{sw}|\zeta_{st}), \dot{x}(t_{sw}|\zeta_{st}), a, b$

while error $> \epsilon$ **do**
  $c \leftarrow \frac{(a+b)}{2}$
  $v \leftarrow$ **Integration**$(c, x, \dot{x})$
  if $v - \dot{x}_{apex} > 0$ **then**
    $a \leftarrow c$
  else
    $b \leftarrow c$
  end if
  error $\leftarrow abs(v - \dot{x}_{apex})$
end while
return $\zeta_p = c$

Because the denominator of Equation (9) is a square value, if the numerator is always positive, i.e. $\frac{d^2z}{dx^2} > -\frac{g}{\dot{x}^2}$ and $z > 0$, then $\frac{\partial(\dot{x}(t'|\zeta_p)}{\partial\zeta_p} < 0$ for all ζ_p. This inequality implies that $\dot{x}(t'|\zeta_p)$ will always decreased with ζ_p which implies monotonicity. In the robot simulation, the proposed method converges within ten iterations.

Algorithm 4 Position Based Integration $(\zeta, x, \dot{x})$

while $x \neq \zeta$ **do**
  $\ddot{x} \leftarrow \frac{g + \frac{d^2z}{dx^2}\dot{x}^2}{z - (x-\zeta_p)\frac{dz}{dx}}(x - \zeta)$
  $\dot{x} \leftarrow \dot{x} + \ddot{x}\Delta t$
  $x \leftarrow x + \dot{x}\Delta t + \frac{1}{2}\ddot{x}\Delta t^2$
end while
return $\dot{x}$

IV. Whole-Body Control

We implement a Whole-Body Compliant Controller [2], to accomplish several Cartesian position tasks while obeying contact constraints. The controller is tasked with achieving a constant COM height, a constant body pitch, a constant body roll, and trajectory tracking of the swing foot while in single support. The contact constraints we use covers only Cartesian interaction with the ground, so all three modes of rotation are freely allowed.

Note that there is no task for yaw control. Controlling the yaw torque on a point-foot bipeds is an over-constrained problem, and in the interest of simplicity we have left it uncontrolled. The robot plans its steps in a free yawing local frame. It may be possible to anticipate the effect of footstep

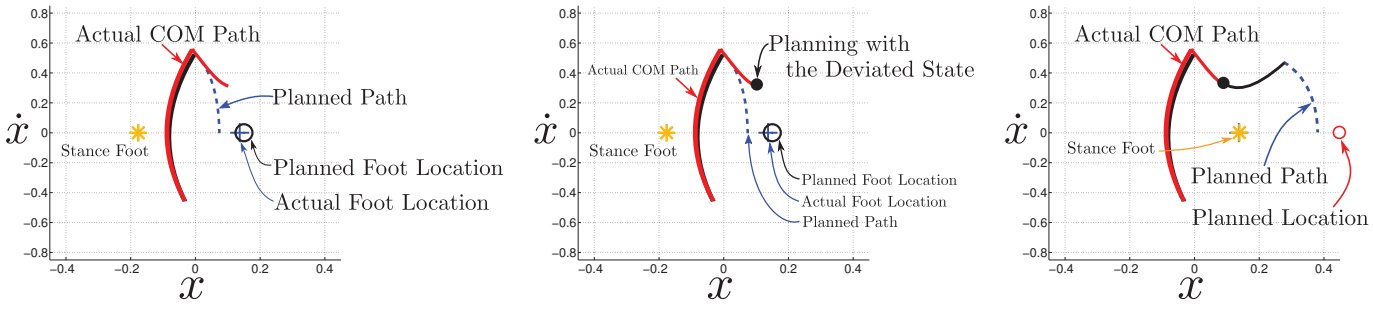

Fig. 4. **Online Planning Process**. The observed COM path generally deviates from the planned path because of uncertainties and simplifying assumptions, as shown in subfigure (a). (b) Periodically, the planner plans the next trajectory and the associated foot placement. When the planner kicks in, it starts with the currently deviated trajectory. (c) As previously described in Fig. 3, the planner then attempts to find the next foot placement such that it reverses the COM velocity.

location on yaw and account for it in the future through a more complex planner.

V. Evaluation with Dynamic Simulator

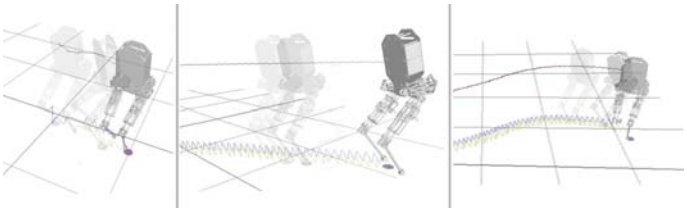

Fig. 5. Simulated Stepping on Multi-Body Dynamic Simulator srLib with Multiple Contacts. The video can be found in https://www.youtube.com/watch?v=Xz0027fjejw

Using the controller described above on a simulated model of the Hume robot, we achieved 40 seconds of simulated walking, which is enough time to show that the motion had stabilized (Figure 5). This section presents an introduction to the simulation environment and parameters used, analyzes the robot's simulated phase space trajectories, and presents the simulated joint data.

To simulate the robot we used srLib[1], which simulates impulse and friction effects as well as multi-body dynamics. The model in the simulation imitates the real robot in terms of kinematic and dynamic properties and uses a friction coefficient of 0.7 to approximate the rubber on Hume's feet. Various simulation and controller parameters are presented in table II. In the table II, we show that t's for lateral and sagittal motion are different. Increasing t' decreases step length for velocity reversal steps, but steps which are too small risk a failure to reverse velocity which forces the next step to be an emergency skipping step. In the Sagittal plane, we design large t' values to ensure that the feet stay in the reachable region.

[1]Seoul National University Robotics Library. Open-source http://robotics.snu.ac.kr/srlib/

Name	Value	Name	Value
Body Height	0.882 m	K_p (Height)	200
Pitch	0.223 rad	K_d (Height)	20
Lifting Time	0.2 sec	K_p (Orientation)	200
Switching Time	0.2 sec	K_d (Orientation)	20
t' (lateral)	0.18 sec	K_p (Foot)	200
t' (sagittal)	0.3 sec	K_d (Foot)	20
Coefficient of Friction	0.7		

TABLE II
Parameter Settings

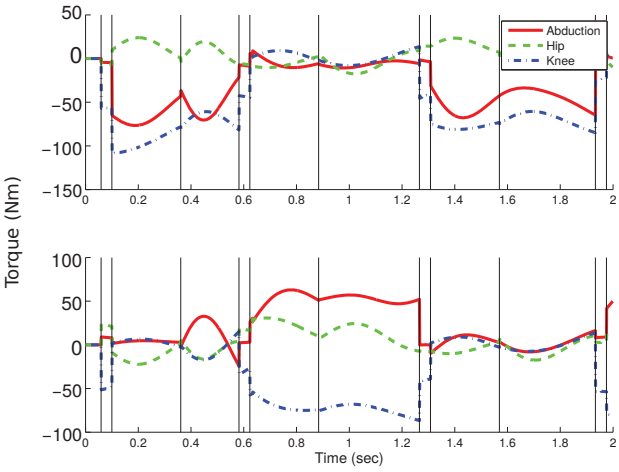

Fig. 7. Joint Torque (Top: Right Leg, Bottom: Left Leg). Black vertical lines mark the phase transitions

We pick four steps in a sequence to exemplify the operation of the planner as shown in Fig. 6. Here, we can see the deviation from the planned path due to uncertainties and the ability of the planner to recover in the following step. Online re-planning, enables the robot to produce cyclic stepping motions when possible or recover from deviations of the planned trajectories. In Figure 7, we show the torque and velocity curves of each joint to illustrate the feasibility of this strategy for a real robot. The torque limits of our Hume biped robot are $140Nm$ for

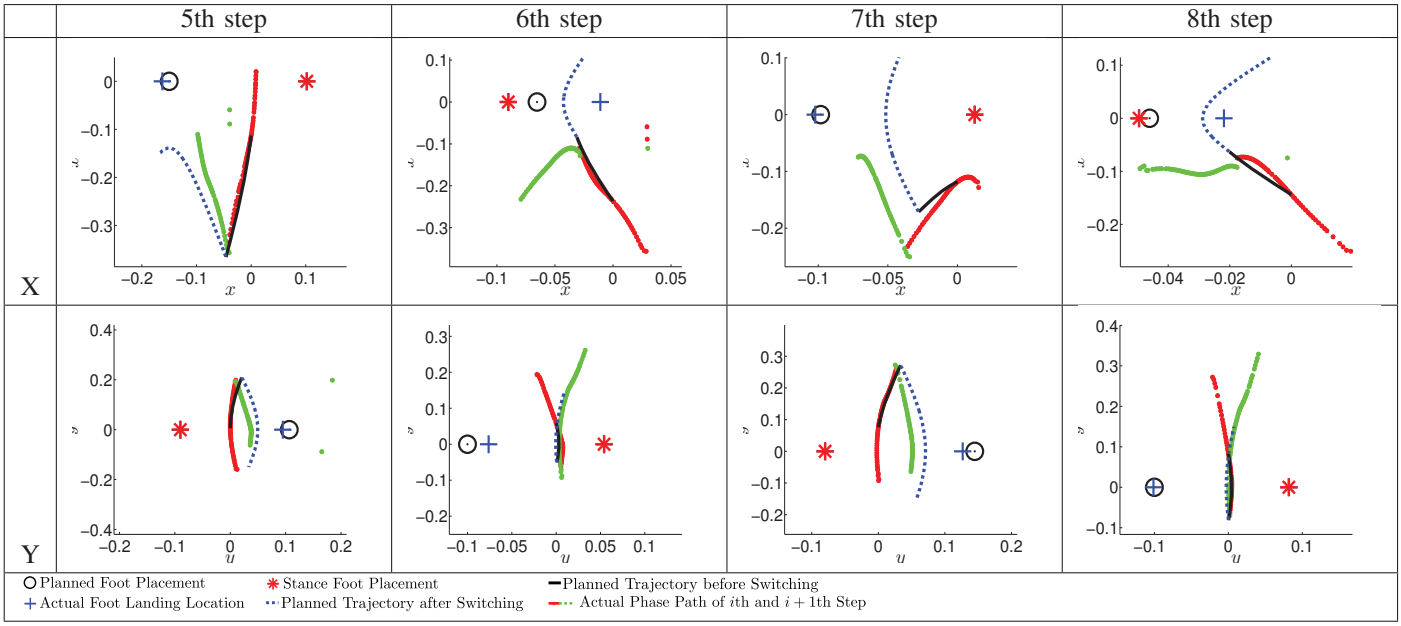

Fig. 6. **Phase Portrait in the x (Sagittal) and y (Lateral) Planes.** Black solid lines are the planned trajectories. Blue dashed lines are the planned trajectories after changing the stance foot. We demonstate that we achieve cyclic motion in the lateral plane, while we recover from deviations on the Saggital plane by continuously replanning.

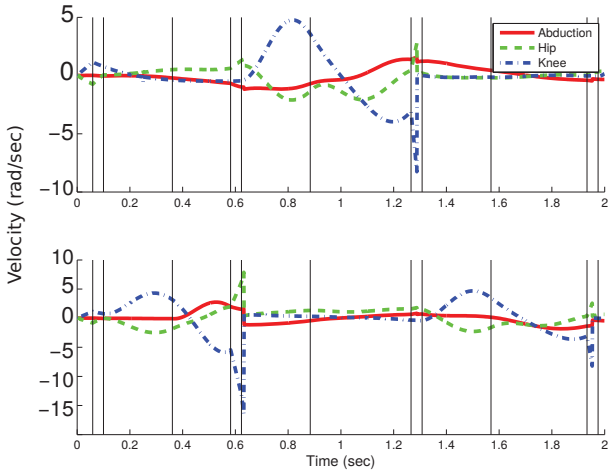

Fig. 8. Joint Velocity (Top: Right Leg, Bottom: Left Leg). Black vertical lines mark the phase transitions

the hip and knee joints, and $55Nm$ for the abduction joint. Simulation results presented in Fig. 7 show that the torque commands for the abduction joint is outside of the allowed range, but the other joints are within their limits.

Joint velocity is another important factor for continuous stepping since fast leg motion introduces large model disturbances. Hume is capable of achieving the computed velocities shown in Fig. 8.

VI. Conclusion

In this paper, we present a control scheme for maintaining stepping balance of an under-actuated point-foot biped robot that is based on a constant time between foot transition and COM velocity reversal, implemented through online replanning of footstep locations. A numerical method is introduced to handle varying height movement. Simulation results verify the feasibility of the proposed control scheme by showing continuous stepping for 40 seconds and more than 80 steps. The success on the simulation indicates the eventual and tantalizing possibility of running the proposed control algorithm in a real robot. In the future, we will investigate the body yaw motion. In terms of hardware validation, we will implement the proposed planning and control scheme in Hume.

References

[1] L. Sentis, B. Fernandez, and M. Slovich, "Prediction and planning methods of bipedal dynamic locomotion over very rough terrains," in *The 15th International Symposium on Robotics Research (ISRR 2011)*, 2011.

[2] L. Sentis, "Synthesis and Control of Whole-body Behaviors in Humanoid Systems," Ph.D. dissertation, Citeseer, 2007.

[3] H. Miura and I. Shimoyama, "Dynamic walk of a biped," *The International Journal of Robotics Research*, vol. 3, no. 2, pp. 60–74, 1984.

[4] M. Raibert, *Legged Robots that Balance*. MIT Press, Cambridge, Ma., 1986.

[5] K. Yin, K. Loken, and M. van de Panne, "Simbicon: Simple biped locomotion control," in *ACM Transactions on Graphics (TOG)*, vol. 26, no. 3. ACM, 2007, p. 105.

[6] A. Ramezani, J. W. Hurst, K. A. Hamed, and J. Grizzle, "Performance analysis and feedback control of atrias, a 3D bipedal robot," *Journal of Dynamic Systems, Measurement, and Control*, 2013.

[7] J. A. Grimes and J. W. Hurst, "The design of atrias 1.0 a unique monopod, hopping robot," in *International Conference on Climbing and Walking Robots*, 2012.

[8] L. Sentis, J. Park, and O. Khatib, "Compliant control of multi-contact and center of mass behaviors in humanoid robots," vol. 26, no. 3, pp. 483–501, June 2010.

[9] L. Sentis and M. Slovich, "Motion planning of extreme locomotion maneuvers using multi-contact dynamics and numerical integration," in *Humanoid Robots (Humanoids), 2011 11th IEEE-RAS International Conference on*. IEEE, 2011, pp. 760–767.

[10] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa, "Biped walking pattern generation by using preview control of zero-moment point," in *Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on*, vol. 2. IEEE, 2003, pp. 1620–1626.

[11] J. Pratt, J. Carff, S. Drakunov, and A. Goswami, "Capture Point: A Step toward Humanoid Push Recovery," in *Humanoid Robots, 2006 6th IEEE-RAS International Conference on*, 2006, pp. 200–207.

[12] Y. Zhao and L. Sentis, "A three dimensional foot placement planner for locomotion in very rough terrains," in *Humanoid Robots (Humanoids), 2012 12th IEEE-RAS International Conference on*, 2012, pp. 726–733.