2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)
November 3 - 5, 2015, Seoul, Korea

A Method for Dynamically Balancing a Point Foot Robot

Donghyun Kim, Gray Thomas, and Luis Sentis[1]

Abstract— **In this paper we apply a general control framework, Whole-Body Operational Space Control (WBOSC), to an under-actuated point-foot biped. We use WBOSC to reproduce the behavior of the Prismatic Inverted Pendulum (PIP) as the center of mass dynamics for our point foot biped. We present and analyze a new algorithm that dynamically balances the PIP model by choosing footstep placements. Our algorithm uses the shooting method and numerical integration to find a footstep location even when the robot's constraints do not permit an analytic solution to the model dynamics. This approach stabilizes the robot in simulation, and keeps the physical robot upright for as many as 15 steps. The primary limitation appears to be inaccurate foot positioning.**

I. INTRODUCTION

As humanoid robots move towards practical application, we are drawn to the remaining limitations in their locomotion systems, which remain decidedly less capable and less reliable than the human locomotion they seek to emulate. In particular, we see many robots that rely very heavily on the ability to apply a torque at their ankle joint. This limits them to terrain on which they can land their whole foot, and makes terrain which necessitates blade or point contact difficult. For example, rocky beaches, areas filled with tree roots, or a toy-covered floor. By addressing point foot locomotion, we hope to offer under-actuated dynamic behaviors as a practical option, even for robots which possess fully controllable alternatives.

This paper explores the idea of using Whole-Body Operational Space Control [1] (WBOSC) to establish a model-following behavior which facilitates 3D walking in a robot limited to under-actuated point contact. Whole-body Operational Space Control is a control framework which calculates a vector of joint torques consistent with a desired set of operational space tasks and a known set of contact constraints. WBOSC does not naturally exploit tilting the torso to balance above one foot, since it decouples its operational space goals. We do not attempt to stabilize the robot in that manner, rather we attempt to stabilize the robot by constantly stepping with a fixed torso orientation.

The main contributions of this study are to: (1) propose an under-actuated bipedal robot infrastructure based on WBOSC and the Prismatic Inverted Pendulum (PIP) model, and (2) adapt a previously published [2] footstep placement controller to stabilize the physical robot, and (3) assess the performance of the resulting system.

[1]Luis Sentis is with Faculty of Mechanical Engineering, University of Texas at Austin, 78712, US lsentis@austin.utexas.edu

II. RELATED WORK

We implement WBOSC in an under-actuated bipedal point-foot robot. This under-actuation means we cannot control all aspects of the center of mass motion with our continuous time controller, as an operational space controller would generally be free to do. We have chosen to explicitly "give up" on the x and y components of motion, leaving those aspects to evolve as they will while the WBOSC satisfies the remaining tasks. These liberated degrees of freedom are still controlled, but must be controlled indirectly, on a step by step basis, by choosing the footstep locations. In [3] a similar idea is pursued, for 2D walking, the controller indirectly avoids influencing the x component of the center of mass behavior using virtual model control [4], an older WBC approach which does not attempt to exploit an accurate dynamic model. The resulting center of mass system behaves much like the linear inverted pendulum model.

A. Locomotion for Point-Foot Bipeds

Point-foot biped robots similar to ours have been widely studied [4]–[9] due to their mechanical simplicity and interesting dynamics. Few have managed to walk upright without the help of a constraint mechanism, the two most notable examples being the hydraulically actuated hopper from [10] and the biped from [6].

As a hybrid dynamical systems problem, point-foot locomotion is difficult because single support motion is under-actuated and naturally unstable. While there are several ways to keep bipedal robots upright [11], the most powerful of them is to exploit the hybrid dynamics and move a foot to a new spot.

One of the most successful approaches to this problem comes from [12], to wherein a model-based controller implementation constrains the dynamics to match a numerically generated lower dimensional system of differential equations which still contains discrete footstep events, a Hybrid Zero Dynamics (HZD). This HZD is designed to be stable and efficient, and when the robot is placed under feedback control this HZD behavior is faithfully reproduced. The robot walks. This work is performed in 2D and focuses on efficient forward walking. The group also introduce a 3D walking result [13] but the robot equips on passive prosthetic feet. Another successful approach based on hybrid zero dynamics is the line of work by [14] which utilizes human inspired trajectories to generate stable periodic locomotion in 2D. These formulations are designed to achieve periodic motions, as is also the case for other works based on Poincare maps [9], [15].

The fundamental hybrid behavior of these systems can be captured by simple models, and the above approaches generate lower dimensional models automatically, and not necessarily ones that are intuitive. We deviate from the above works in that we prefer models which do not presuppose footstep locations in order to allow aperiodic locomotion. Since the center of mass can be controlled independently of the swing foot, such models need only describe center of mass behavior.

Frameworks such as the capture point [16], and the more general Divergent Component of Motion (DCM) [17] use such a model: the analytically tractable linear point mass model. In this model, the center of mass experiences a force proportional to its distance from the stance foot, pushing it directly away from the stance foot. Since gravity is the only other force acting on the center of mass, there exists a Virtual Repellent Point (VRP) located a constant height above the point foot. Technically, if we permit a non-zero angular momentum rate of change this point can move. Relative to the VRP the center of mass has unstable, linear dynamics in point foot robots. By exploiting the fact that the analytic solution to this model contains a stable (convergent) and an unstable (divergent) component, it is possible to control only the divergent component of motion, which has first order unstable linear dynamics. DCM-based methods are generally designed to stabilize plate-foot robots with a pre-defined path of known footstep locations, but the DCM could easily be applied to the problem of stabilizing point-foot robots by choosing footstep locations. However, this has not been pursued before nor in this paper.

The DCM is a powerful and intuitive concept, but the linear model is overly restrictive given the ample time available for computing the next footstep location. A numerical technique, phase space planning [18] extends the linear pendulum model to consider the case where the center of mass height is a first order continuous function of the horizontal position—the Prismatic Inverted Pendulum (PIP) model [2]. Our controller attempts to indirectly restrict the center of mass dynamics to follow the PIP model, and one objective of this paper is to assess the performance of our whole body controller by observing the degree to which it can recreate these dynamics. Previously [2] we presented an earlier, somewhat incomplete, version of the trajectory generation algorithm considered in this paper. This paper adds 3D experimental results, a simulation which uses a non-constant height surface, and a method of optimization which is more generally applicable.

III. Stabilizing the PIP Model

Given that the task-set we command through WBOSC is designed to replicate the behavior of the PIP model, we need only to choose footstep locations that stabilize the PIP model in order to stabilize the robot itself. The PIP model does not depend on the trajectory the swing foot, so we can specify the landing point and time freely, within sensible kinematic limits.

We have developed an algorithm for finding an upcoming footstep location which stabilizes the PIP model which we refer to as the "phase space constant time to velocity reversal planner". In every step, when the lifting phase reaches 60% completion, this planner runs once to compute the next footstep location. This is a processor intensive task, and must be run outside of the real-time thread; it finishes just before the lifting phase ends. The operational space set-point trajectory for the swing foot is then defined parametrically based on this desired landing position, with the trajectory ending once ground contact is sensed. If the ground has the height we expect and the position tracking is ideal, then the footstep will land after exactly the nominal swing duration. If the planned step is outside the mechanical limits of the robot, the planner chooses the closest reachable step.

1) Velocity Reversal: Our planner attempts to stabilize the robot by causing the center of mass to reverse direction every step. In its simplified planning model, the feet change swing-stance roles instantaneously, and the center of mass must be made to reverse its velocity a set time after this instant. This duration between the time of transition and the time of reversal, t' determines the behavior of the planner. If the duration is too small the planner will take larger and larger steps until the robot reaches its kinematic limits. If the duration is large, the robot's feet will move closer together over time, until its center of mass behavior is better described by noise than by the PIP model. We chose larger values of t', but implement a strategy which prevents the feet from overlapping each other. When we plan foot locations for 3D walking we consider forward and lateral velocity separately, and we assign a different time to velocity reversal parameter for each, t'_x and t'_y respectively. The two must naturally share a cycle time, but allowing a different time to velocity reversal allows them to have a different propensity to take wide steps. This is used to prevent the y-component of foot separation from shrinking too quickly.

While other planning frameworks have demonstrated their practicality, the authors find this framework interesting because it allows for dynamics of the center of mass that are not analytically solvable in closed form. This is a large class of behaviors which are not possible using capture point or divergent component of motion methods. Since the PIP model is not always analytically tractable, numerical search is necessary. This leads to the primary computational element in the planning procedure: a shooting method search over the possible footstep locations to find a center of mass trajectory which accomplishes both velocity reversal goals. In the one degree of freedom case, and in our previous paper, we used a bisection search over footstep locations. However bisection search is not possible in 3D and we used a Newton's method search with numerical derivative approximations.

As illustrated in Fig. 1, the planner begins calculating the landing location when 70% of the lifting phase is reached in the robot state machine's progression. Using the current estimate of the center of mass velocity and position, it numerically integrates forwards in time to predict its COM position and velocity when its stance foot and swing foot

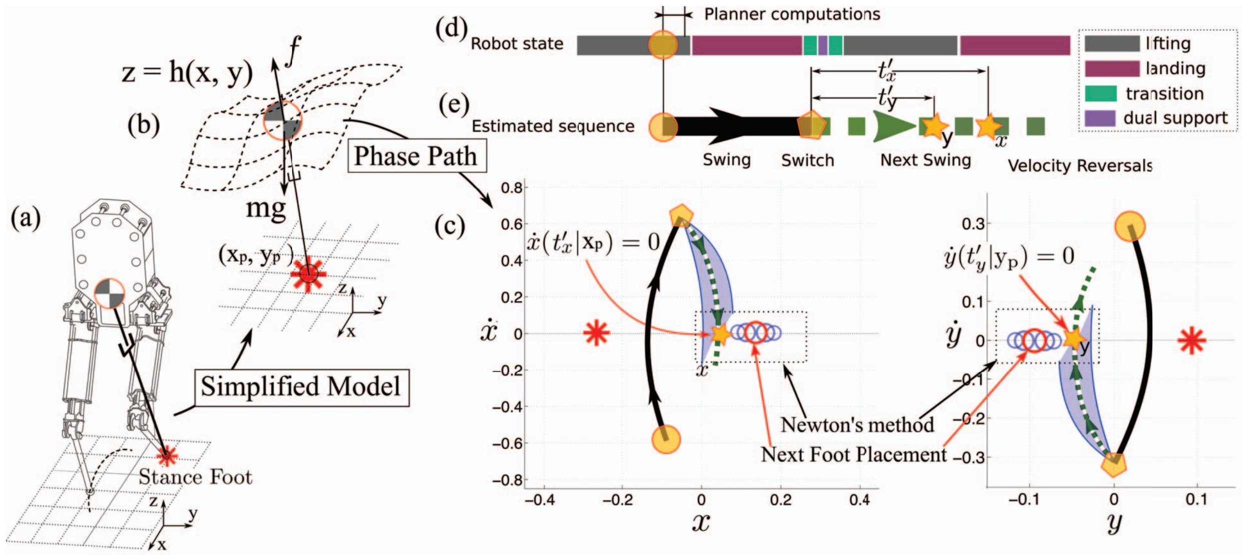

Fig. 1. **Constant Time to Velocity Reversal Algorithm.** As shown in (a), we approximate the dynamics of the robot with the prismatic inverted pendulum, shown in (b). This model predicts the dynamics of the horizontal center of mass position x, y given the stance foot location $(x_p,\ y_p) =$ ∗ and the height surface $z = h(x,\ y)$. This can be integrated forward in time via the numerical integration procedure shown in (c). When the planner starts operating it records the initial state and integrates this state forward to determine the switching state. As shown in time-lines (d) and (e), the "Estimated sequence" of the planner has an analogue in the "Robot states" of the state machine. In particular, the switching state roughly corresponds to the dual support phase of the walking state machine. This state represents the planner's guess at the time and state $(x, \dot{x}, y, \dot{y})$ values immediately after the switch. The goal of the planner is ultimately to stabilize the robot, but this is implemented by choosing the next footstep ○ such that x and y velocity equal zero t'_x and t'_y seconds, respectively, after the foot switch every step. For sufficiently smooth height surfaces, the relationship between the next footstep location and the velocity is monotonic, so only a single solution exists. We use Newton's method with numerical differentiation to identify this solution. There exist two velocity reversal states: $\star_x$ and $\star_y$ as shown in (c).

will switch roles. This time, position, and velocity is known as the switching state, in Fig. 1.

The implementation of the planner enforces the choice of a value for the reversing time, t'. This time value remains constant for every step. As of now, t' is manually chosen and as we show in the simulations it is able to stabilize the biped for an arbitrary long number of steps. The planner then finds and returns the footstep location which causes the robot's COM velocity to reach zero, t' seconds after the foot switch, starting from the post-impact state. For each potential footstep location considered, the planner integrates forward in time starting from the post-impact state as suggested in Fig. 1, returning the velocity after t' seconds. This integration can be viewed as a function mapping footstep location to a future velocity, and it is this function over which we search for a zero crossing via bisection. Since we use bisection, the number of integrations actually performed is very low, however the process relies on the monotonicity of the relationship between footstep location and the velocity after t' seconds of integration. If the height surface is planar, then this relationship is linear.

IV. Experimental Setup

The experimental conditions play a large role in explaining the final result of our robot's walking. As with all robots, the control system is built upon the sensors in a highly dependent way. The control system provides the foundation for the planner. Each component is only validly designed if the component below it behaves perfectly, and to understand the full system we must introduce the imperfections in the assumptions it has made.

A. End-to-End Controller Architecture

The feedback control system is spread over six joint-level controllers and a centralized high level controller which runs WBOSC (Fig. 2). WBOSC [1] is a feedback control strategy based on Operational Space Control [20], which extends it to floating base robots in contact with the environment. It allows the user to specify multiple task objectives and their impedance in operational space. It additionally subdivides the torques applied to the robot into orthogonal spaces which affect either the motion of the robot or the internal forces which do not.

The purpose of the joint level controllers is to achieve good torque tracking given the series elastic actuators. This type of control architecture falls into the category of a distributed control system which allows the joint controllers to focus on high speed actuator dynamics while the centralized controller does not need to deal with this nuance. Yet the feedback at the high level is necessary in order to create the coupling between joints implied by operational space impedance tasks.

B. Series Elastic Actuators

The robot's joint controllers are based on the passivity torque controller described in [21]. We kept this controller's structure while changing the gains of the feedback controller to enhance the performance of the high level controller, and this ultimately entailed reducing the low level torque gains. In order to tune the torque gains we leveraged our findings

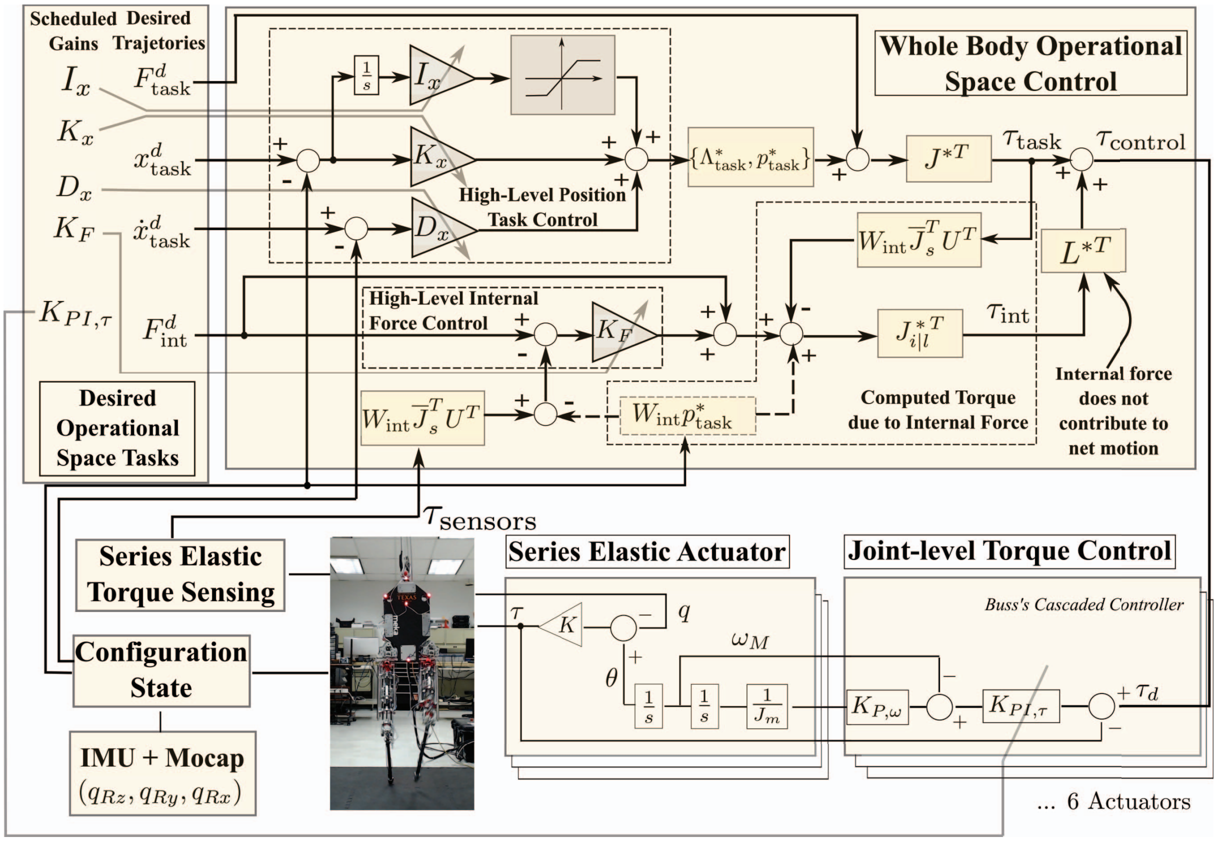

Fig. 2. **Overall Control Diagram.** This figure illustrates the whole body operational space control process (WBOSC) and the joint-level torque controllers. The feedback control of internal forces and gain scheduling are characteristic features of the feedback control system. Please see [19] to get details of our system.

in [22]. In this study we describe a trade-off between torque gains and position gains in a distributed control architecture. Specifically, we explore the observation that raising the torque controller's proportional gain limits the maximum stable position gains and vice versa. To respond to this observation we implemented a gain scheduling strategy: in the joints of the stance leg we lowered the torque gains so we could raise the position gain and reduce error. In the joints of the swing leg we raised the torque gains to produce less friction dominated behavior.

C. Whole Body Operational Space Control

At the implementation level, WBOSC worked – provided that latencies were sufficiently small. Achieving a $1\ \mathrm{ms}$ latency required significant software work. We reduced the basic computational cost of our WBOSC algorithm by bypassing recursive dynamics software and instead using a closed form expression to calculate the mass matrix. In order to reduce the tracking error, we added an integral term to all position tasks, which helps alleviate the friction difficulties involved in lowering the torque gains at the DSP level. This also reduces error due to inaccuracy in the gravity estimation term and other steady state errors in our dynamics model.

V. Experimental Results and Assessment

Our experimental results include a simulation and a physical experiment. While the physical experiment is not yet a fully succesful control result, physical results are more compelling than simulation results. The 3D walking simulation is critical to demonstrate non-flat height surface following, and walking with a non-flat ground surface. These are capabilities added to our planning algorithm since our last paper [2], which, while it was designed to allow this behavior, actually contained a bug. We then assumed, incorrectly, that bisection in two separate searches would be sufficient for tracking a 3D height surface. The simulation result demonstrates that by switching to a newton search in a coupled problem, this issue can be resolved.

A. Simulated Walking

As shown in Fig. 3, our planner allows our rigid body simulation robot to climb up on top of a 7 cm platform, using a height surface which is not flat. Specifically, this height surface is defined, piece-wise, as a function of a global x coordinate, with three constant height pieces connected by two sinusoidal segments. The middle constant height piece is 7 cm above the others, to account for the platform, and is located directly above it. The surface maintains first order continuity. In this experiment the robot's planner biases it towards a moving goal point. As this goal point passes

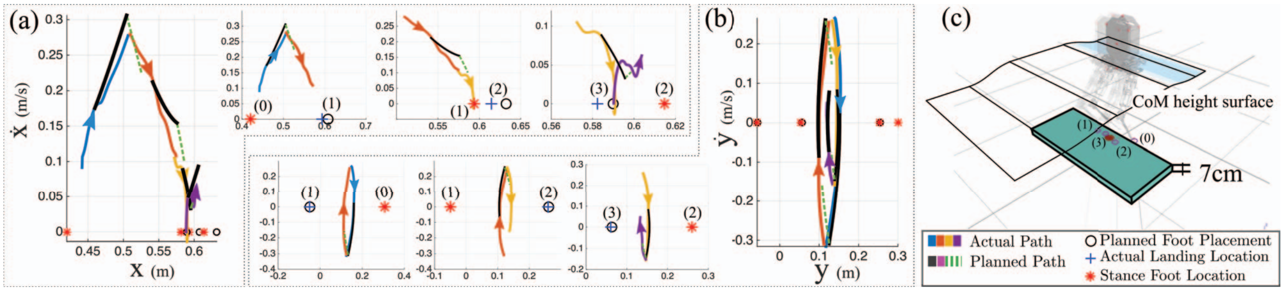

Fig. 3. **3D Stabilization in Simulation.** Subfig. (a) displays three steps of walking in the x component of theCoM perspective, while (b) displays the lateral COM phase portrait. In both figures the smaller axes highlight the planned versus actual trajectory for each of the three steps. The steps are shown in (c) using the simulation graphics provided by the SrLib Multi-Body Dynamic Simulation Environment.

over the platform the robot ascends and descends it, while constantly stepping.

As the data in Fig. 3 makes clear, the simulation robot is not able to perfectly recreate the prismatic pendulum behavior that we expect. This is because the simulation has some elements in it which are designed to simulate real world. Tiny slip and impact exist when landing occurs. The controller ignores impact transients which will not have died out by the time the planner makes its next prediction. What the simulation makes clear is that these defects alone are not enough to prevent the robot from walking over non-flat terrain.

We have boldly used a discontinuous step as our ground profile, however on closer inspection it may be possible that the search will not converge because of this choice. Since a foostep on the cusp of the platform might reverse direction too soon, and a step at the very base too late, the planner will be forced to choose between two incorrect results in some cases. We therefore expect good planner performance only if all reachable footstep locations are reasonable, i.e. not vertical.

B. 3D Walking

The 3D walking experiment demonstrates the real world applicability of our approach. In this experiment, the robot is held as it performs a rise to normal height maneuver. Once it reaches its starting height, the experimenter balances the robot carefully and lets it go as it takes its first step. Once free, the robot continuously steps until it falls over. There is a harness rope, slack when the robot is at its starting height, which catches it and prevents major damage. The power and Ethernet tether hang slack from another rope.

Since Hume's feet are points, there is no way to resist yaw motion during single support, given our existing constraints. To compensate for this, we use the time the robot spends in dual support to address this q_{Rz} task. In dual support, x^d_{task} is $[\text{COM}_z,\ q_{Rz},\ q_{Ry},\ q_{Rx}]^T$. In single support, task set is $[\text{COM}_z,\ q_{Ry},\ q_{Rx},\ foot_x,\ foot_y,\ foot_z]^T$, where COM_z is the center of mass height and q_{Rz}, q_{Ry} and q_{Rx} are body yaw, pitch and roll angles, respectively.

Fig 4 shows some walking snapshots and phase paths for this experiment. Since our PIP model following is open-loop, rather than enforced by some model reference controller, we expect that error between our initial guess and the result should grow with time. This is because the model exhibits naturally unstable escape behavior. The phase space data is corrupted by the high frequency noise because the noisy joint encoder and IMU sensor signals are used to compute CoM velocity.

We correct yaw error during the transitions and dual support, but this opportunity is very brief. Our swing foot trajectory following is rather inaccurate, and this limits the success of our planner perhaps even more than the model following. Since there is error in the foot position, there is a limit on how large we can set t' before the controller risks accidentally stepping somewhere unrecoverable. Yet despite all these major sources of error our robot has been able to stay upright for 15 steps using only point foot contacts.

VI. Conclusion

Point-foot robots like ours cannot independently control their center of mass in the single support mode. Therefore, it is difficult for them to implement locomotion strategies based on center of mass tracking such us those using the Zero Moment Point or the Divergent Component of Motion. Our approach attempts to stabilize the robot with only footstep selection. But without taking advantage of centroidal momentum, we must deal with a model following error which is quite large. Our controller for the physical robot is not yet stable, as the robot remains upright for 15 steps. Since the planner is designed with the simple PIP model and the assumption that WBOSC satisfied the Zero Dynamics of the PIP model, which is to keep a height surface, constant pitch and roll, the robot fails to maintain a balance when the controller fails to satisfy the assumption. There are many reasons why the real robot cannot sustain the requirement including imperfect modeling and sensing noise. We are currently investigating techniques to eliminate these problems to enable our point-foot biped robot to remain upright indefinitely.

When we started this research we were motivated to attain non-periodic gaits. We were driven by applications such as rough terrain walking, jumping between vertical walls, and push recovery. At that time there were no algorithms suited for those type of behaviors in point-foot robots. We devised a new rule-based algorithm, which we dubbed Phase-Space

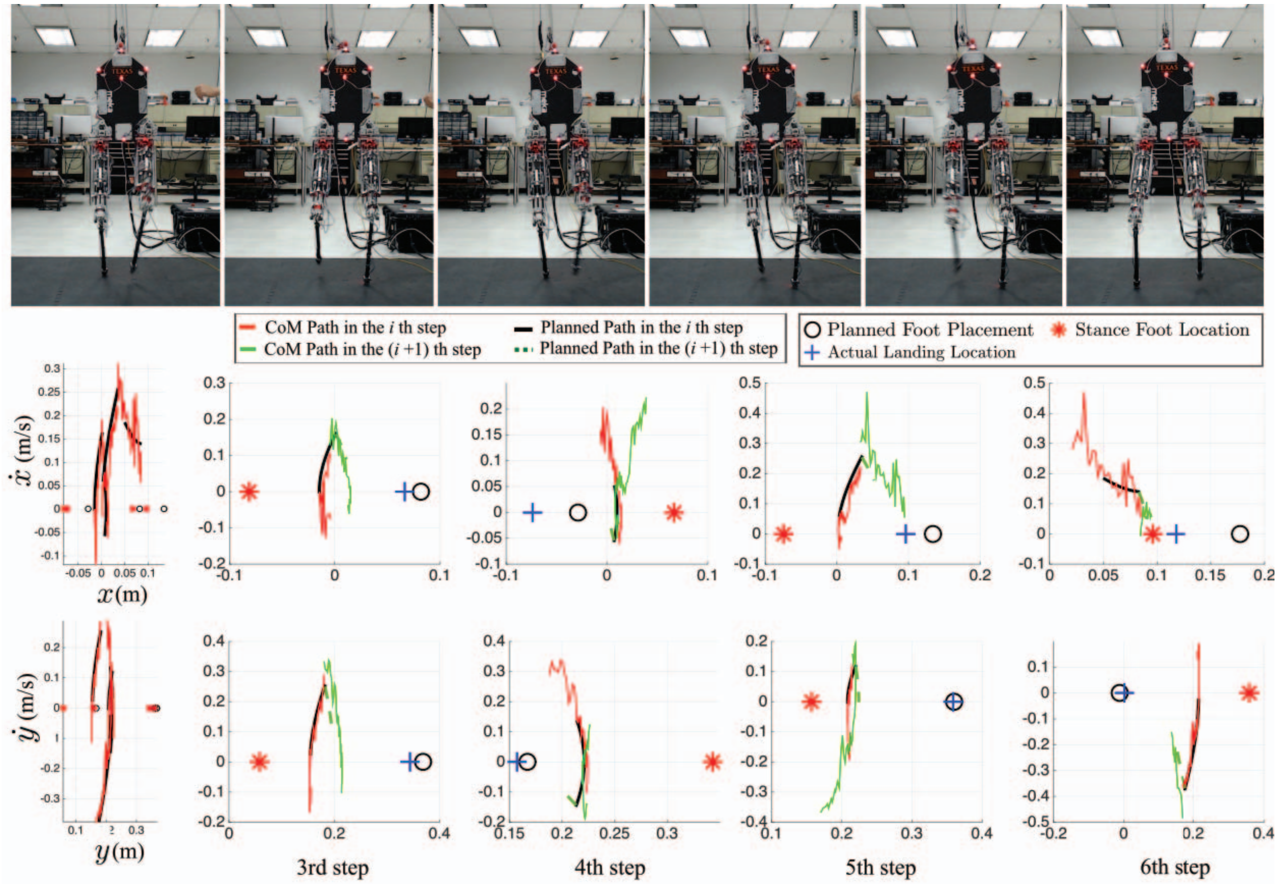

Fig. 4. **3D Walking.** The snapshots and data of steps 3-6 are presented. The phase path of steps 3-6 are expanded into individual step planning plots. In each step planning plot, a red line marks the actual COM path up to the switching state, and a green line continues the trajectory after the switch. The robot's initial stance foot in the step planning plot is denoted with a ✳, the planned second footstep with a black circle, and the achieved second stance foot location with a blue cross. Therefore, a green line in the ith step plot is the same path as the red line in the $(i+1)$th step plot. A black line and a dark green dotted line are, respectively, the PIP model's predicted paths before and after the switching state.

Planning [23], in which foot positions and apex velocities were known a priori. Phase space techniques were then employed to find transition states between the steps. For this new work, we decided to extend phase space techniques to the general case of continuously stepping such that the robot's center of mass velocity could be quickly reversed. The overall effect is an undirected walk that stabilizes the robot through continuous re-planning capabilities. Although such walking does not necessarily stay in place, our experiment has two goals: to create an environment for future push recovery, and to create an algorithm for 3D untethered balancing.

One of the primary benefits of this locomotion method is still unexplored: since we use a numerical search that chooses a footstep position before it tests the height surface, we can choose a height surface which is parameteric in the footstep position without a major change to the algorithm. So long as the reversal velocity varies continuously with the footstep, we will be able to use our search method as is. This could be extended to plan straight legged walking, and also to allow continuous searching even over discontinuous ground.

In summary, we have verified the practicality of using WBOSC to reproduce simple model behavior in an under-actuated system which permits discrete time footstep control. We have developed a discrete time controller which can stabilize point-foot robots in 3D simulation, and we have shown that it can keep our physical robot from immediately falling over. There are many reasons why the real robot falls over after 15 steps, including imperfect modeling and sensing noise. We are currently investigating techniques to eliminate these problems to enable our point foot biped robot to remain upright indefinitely

References

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

[2] D. Kim, G. Thomas, and L. Sentis, "Continuous cyclic stepping on 3d point-foot biped robots via constant time to velocity reversal," in *Control Automation Robotics & Vision (ICARCV), 2014 14th International Conference on*. IEEE, 2014.

[3] J. Pratt, P. Dilworth, and G. Pratt, "Virtual model control of a bipedal walking robot," in *ICRA*, Albuquerque, USA, April 1997.

[4] J. Pratt, C.-M. Chew, A. Torres, P. Dilworth, and G. Pratt, "Virtual model control: An intuitive approach for bipedal locomotion," *The International Journal of Robotics Research*, vol. 20, no. 2, pp. 129–143, 2001.

[5] E. R. Westervelt, J. W. Grizzle, C. Chevallereau, J. H. Choi, and B. Morris, *Feedback control of dynamic bipedal robot locomotion*. CRC press Boca Raton, 2007.

[6] B. G. Buss, A. Ramezani, K. A. Hamed, B. A. Griffin, K. S. Galloway, and J. W. Grizzle, "Preliminary walking experiments with underactuated 3d bipedal robot marlo."

[7] C. Chevallereau, A. Gabriel, Y. Aoustin, F. Plestan, E. Westervelt, C. C. De Wit, J. Grizzle *et al.*, "Rabbit: A testbed for advanced control theory," *IEEE Control Systems Magazine*, vol. 23, no. 5, pp. 57–79, 2003.

[8] T. Yang, E. Westervelt, A. Serrani, and J. P. Schmiedeler, "A framework for the control of stable aperiodic walking in underactuated planar bipeds," *Autonomous Robots*, vol. 27, no. 3, pp. 277–290, 2009.

[9] A. Hereid, S. Kolathaya, M. S. Jones, J. Van Why, J. W. Hurst, and A. D. Ames, "Dynamic multi-domain bipedal walking with atrias through slip based human-inspired control," in *Proceedings of the 17th international conference on Hybrid systems: computation and control*. ACM, 2014, pp. 263–272.

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

[11] S.-H. Lee and A. Goswami, "A momentum-based balance controller for humanoid robots on non-level and non-stationary ground," *Autonomous Robots*, vol. 33, no. 4, pp. 399–414, 2012.

[12] K. Sreenath, H.-W. Park, I. Poulakakis, and J. W. Grizzle, "A compliant hybrid zero dynamics controller for stable, efficient and fast bipedal walking on mabel," *The International Journal of Robotics Research*, vol. 30, no. 9, pp. 1170–1193, 2011.

[13] B. G. Buss, A. Ramezani, K. A. Hamed, B. A. Griffin, K. S. Galloway, and J. W. Grizzle, "Preliminary walking experiments with underactuated 3D bipedal robot MARLO," in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, 2014, pp. 2529–2536.

[14] H.-H. Zhao, W.-L. Ma, M. Zeagler, and A. Ames, "Human-inspired multi-contact locomotion with amber2," in *Cyber-Physical Systems (ICCPS), 2014 ACM/IEEE International Conference on*, April 2014.

[15] J. W. Grizzle, C. Chevallereau, R. W. Sinnet, and A. D. Ames, "Models, feedback control, and open problems of 3d bipedal robotic walking," *Automatica*, 2014.

[16] J. Pratt, T. Koolen, and e. De Boer, T., "Capturability-based analysis and control of legged locomotion, part 2: Application to m2v2, a lower-body humanoid," 2012.

[17] J. Englsberger, C. Ott, and A. Albu-Schaffer, "Three-dimensional bipedal walking control based on divergent component of motion," *Robotics, IEEE Transactions on*, vol. 31, no. 2, pp. 355–368, 2015.

[18] 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*. IEEE, 2012, pp. 726–733.

[19] D. Kim, Y. Zhao, G. Thomas, and L. Sentis, "Assessing Whole-Body Operational Space Control in a Point-Foot Series Elastic Biped: Balance on Split Terrain and Undirected Walking," *arXiv.org*, p. 2855, Jan. 2015.

[20] O. Khatib, "A unified approach for motion and force control of robot manipulators: The operational space formulation," *IJRR*, vol. 3, no. 1, pp. 43–53, 1987.

[21] H. Vallery, R. Ekkelenkamp, H. Van Der Kooij, and M. Buss, "Passive and accurate torque control of series elastic actuators," in *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*. IEEE, 2007, pp. 3534–3538.

[22] Y. Zhao, N. Paine, and L. Sentis, "Feedback parameter selection for impedance control of series elastic actuators," in *Humanoid Robots (Humanoids), 2014 14th IEEE-RAS International Conference on*. IEEE, 2014, pp. 999–1006.

[23] L. Sentis, B. Fernandez, and M. Slovich, "Prediction and planning methods of bipedal dynamic locomotion over very rough terrains," in *The International Symposium of Robotics Research*, August-September 2011.