

Partial Feedback Linearization Based Control for a One-Legged Quadruped Model with Flexible Spine on Rough Terrain

Pat Terry and Katie Byl

Abstract—This paper considers control techniques for a planar model of a quadruped robot with a flexible actuated spine, with the goal of having robust gaits that can both remain stable and successfully switch trajectories to achieve different stride lengths on rough terrain. Our focus is on tracking trajectories for center of body and body angle, given a single point of support on the ground, and we focus on repeated hopping motions on the rear leg to illustrate the approach more directly. A limitation in modeling compliant components such as springs in the legs of both bipeds and quadrupeds is the inability to instantaneously control the accelerations of the links of the robots, therefore we do not include a spring or compliant device in our model, instead using the well developed and simple dynamics of a 3-link manipulator in an effort to achieve high controllability. The problem of having a control framework capable of generating and tracking precise body movements for a non-compliant model in the presence of ground collisions remains an open problem, and here we provide a robust solution by extending partial feedback linearization techniques to directly regulate the center of mass and body angle toward a trajectory that is planned to be compatible with a single point of support over time.

I. MOTIVATION

The potential benefits of legged locomotion are many-fold and have been an area of active study for many years. Legged robots are versatile because they potentially can utilize a wide range of gaits allowing them to traverse rough terrain, to operate in areas where wheeled vehicles can not. Studying the behavior of a single leg and extending this to multiple legs has been shown to be effective [1] and has led to the development and study of many hopping and bounding robots in an effort to better understand legged locomotion and develop robust, stable, and accurate control.

The inclusion of a flexible spine in quadruped modeling is motivated by observing how animals in nature utilize their spine for bounding and leaping [2] [3], and has been an active area of study for robot locomotion [4] [5]. Recent experimental results have shown that the inclusion of an active spine results in improved stability [7] and the ability of achieving faster gaits with higher reachable apex heights [8]. However, much of the recent work has been concerned with utilizing existing methods of bounding gait generation for quadrupeds such as the famous BigDog [6], and applying it to a similar quadruped model with the inclusion of a flexible spinal joint [9]. By comparison, we are interested in instead thinking of the actuated spine as an additional degree of freedom that allows us to pursue more direct reference

tracking of the center of mass location of the robot over time.

An important step in studying this type of system, before optimal trajectories can be selected and energy efficient gaits can be constructed, is the development of a robust control framework. The model we use for this system is shown in Fig. 1, and is essentially a 4 link manipulator with masses at the two hip locations. No spring is included at the foot in order to investigate control methods that allow for high controllability via instantaneously regulating the acceleration of each link, which is not possible with the inclusion of a compliant component such as a spring at the leg due to the force being instantaneously determined by the spring force. This is an underactuated control problem, and much work has been done using the method of Partial Feedback Linearization (PFL) on similarly modeled systems by Spong [10]. When modeled in this way, our system shares similarities between hopper and biped models, and extending PFL to underactuated robots modeled as manipulators has been studied in the past for biped locomotion problems such as the Compass Gait [11]. We know from Spong’s work that a number of equations of motion of the system equal to the amount of available actuators may be linearized even if they are not directly actuated [12]. Spong’s famous Acrobot work was focused on the PFL of a specific actuated or unactuated link physically on the system [13], however we are interested in utilizing PFL in order to carefully regulate both the center of mass and the body angle of the robot by generating their respective equations of motion and using PFL to linearize them with respect to our available actuators. These variables are interesting to control because they allow us to accurately set gait characteristics and provide the ability to switch between gaits to regulate stride length.

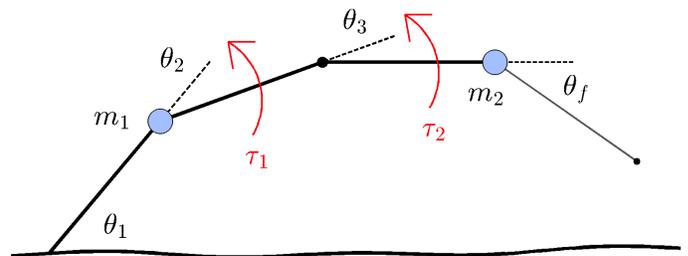


Fig. 1. System model is represented as simple chain when in stance phase. The system has three state angles with only two actuators and is therefore underactuated.

The trajectories presented in this paper are designed by

This work is supported by Army (Grant No. W911NF-08-R-0012)
 P. Terry and K. Byl are with the Robotics Lab in the ECE Dept,
 University of California, Santa Barbara, CA 93106 USA pwterry at
 u mail.ucsb.edu, katiebyl at ece.ucsb.edu

keeping the ground reaction force vector (GRF) constantly pointing directly through the center of mass, as seen on Fig. 2. This is done to simplify the trajectory generation by keeping the body angle variation during stance to a minimum, but also provides an interesting set of trajectories that the system naturally would like to follow under no perturbations. The resulting CoM motions are not bounding, but hopping gaits. The modeling and control structure presented is general enough to have results compared to both hopping robots and bipeds. Typically Hoppers only have one available control parameter that must be used to regulate the touchdown angle in order to achieve a stable gait [14], however the model presented here is interesting because with the addition of an actuated spine angle we in effect have an additional control parameter we can utilize to achieve a wider range of gait motions.

The exclusion of a spring on the leg of the robot prevents us from ignoring problem of ground collisions. Impact with the ground occurs with every legged robot to some degree and causes some energy loss proportional to the unsprung mass. As a result of this, hardware implementations of real robots have potentially significant losses from the ground impact, one common example being the unsprung portion of the leg of a Hopper [15]. Human legs exhibit similar behaviour in that they are not perfectly lossless springs, which has been observed in actual energy recovery experiments [16], with some studies suggesting gait selection and transition speed has a significant impact on energy recovery [17]. In this work the collision with the ground is assumed to be completely inelastic with no slipping at the point of contact, and is calculated by applying conservation of angular momentum. While we are not focused on the specific issue of energy loss and optimizing for efficiency, we are interested in the problem of maintaining accurate control in the presence of this large perturbation from the foot impacting the ground at every step. Our control provides a robust way of maintaining stability despite the large linear and angular velocity instantaneous perturbations caused by ground collisions.

There are specifics of the system we do not model in this study, and leave open for future work. We are not focused on finding optimal trajectories for the system to follow, but rather having a precise and stable control framework that allows the input of CoM stance phase trajectories and allows for switching between them under rough terrain conditions to achieve different strides. Energy efficiency and limitations of real-world actuators are also not considered in this work, and sensor accuracy is assumed such that full state information is available.

The remaining sections of this paper are organized as follows. Section II introduces our system model's equations of motion, and derives the partial feedback linearization control laws. Section III recasts the problem of ground collision in terms of our control, and provides a solution that involves splitting up the stance phase into two segments each with unique PFL control. Section IV describes our use of asymmetric CoM trajectories in order to switch between

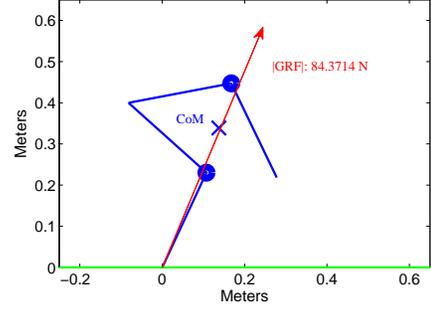


Fig. 2. Frame of simulation during stance phase showing the ground reaction force pointing through the center of mass of the system. If a trajectory is designed such that this can be maintained for all time, the body angle will remain constant.

different stride lengths, and lastly Section V provides simulation results on several rough terrain levels.

II. SYSTEM DYNAMICS AND PFL CONSTRUCTION

Our planar system model is composed of two equal point masses of 2kg, and four equal links of approximately 25cm. There are two actuators, one at the hip and one at the spine. The feet are assumed to be massless, therefore only three angles are needed to model the continuous time dynamics. During the flight phase an additional state r_1 is used to describe the displacement off ground, however while in stance the system forms a simple kinematic chain.

Assuming the foot makes contact with the terrain at the origin, the locations of the three body points are defined as follows.

$$\begin{aligned} x_b &= L_1 * \cos(\theta_1) \\ y_b &= L_1 * \sin(\theta_1) \end{aligned} \quad (1)$$

$$\begin{aligned} x_h &= x_b + L_2 * \cos(\theta_1 + \theta_2) \\ y_h &= y_b + L_2 * \sin(\theta_1 + \theta_2) \end{aligned} \quad (2)$$

$$\begin{aligned} x_f &= x_h + L_3 * \cos(\theta_1 + \theta_2 + \theta_3) \\ y_f &= y_h + L_3 * \sin(\theta_1 + \theta_2 + \theta_3) \end{aligned} \quad (3)$$

Next we define the kinetic co-energy T and potential energy V . The velocities of the three body points are also required, and can be easily computed using product and chain rules.

$$T = 0.5(m_b(\dot{x}_b^2 + \dot{y}_b^2) + m_f(\dot{x}_f^2 + \dot{y}_f^2) + J_b\dot{\theta}_b^2 + J_f\dot{\theta}_f^2) \quad (4)$$

$$V = m_bgy_b + m_fgy_f \quad (5)$$

By utilizing the Lagrangian method, non-linear equations of motion are calculated and defined in terms of the state vector X containing six states. The equations for link accelerations are collected into non-linear matrices $M(X)$ and $C(X)$.

$$X = [\theta_1, \theta_2, \theta_3, \dot{\theta}_1, \dot{\theta}_2, \dot{\theta}_3]^T \quad (6)$$

$$\begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \\ \ddot{\theta}_3 \end{bmatrix} = M(X)^{-1} \left(C(X) + \begin{bmatrix} 0 \\ \tau_1 \\ \tau_2 \end{bmatrix} \right) \quad (7)$$

The center of mass coordinates and body angle of the system are the three control variables we are interested in. Using partial feedback linearization we can control two of these three because we have two actuators. These variables are defined as

$$x_{cm} = \frac{1}{m_f + m_b} * (m_b x_b + m_f x_f) \quad (8)$$

$$y_{cm} = \frac{1}{m_f + m_b} * (m_b y_b + m_f y_f)$$

$$\theta_B = \text{atan}\left(\frac{y_f - y_b}{x_f - x_b}\right) \quad (9)$$

We then take derivatives to define \ddot{x}_{cm} , \ddot{y}_{cm} and $\ddot{\theta}_B$, and by substituting Eq. 7 for the state acceleration variables we re-write the acceleration equations as follows

$$\begin{aligned} \ddot{x}_{cm} &= \beta_x \tau_2 + \gamma_x \tau_1 + \epsilon_x \\ \ddot{y}_{cm} &= \beta_y \tau_2 + \gamma_y \tau_1 + \epsilon_y \\ \ddot{\theta}_B &= \beta_\theta \tau_2 + \gamma_\theta \tau_1 + \epsilon_\theta \end{aligned} \quad (10)$$

Where β , γ , and ϵ are functions of the state X . We now define our feedback control law to track references for two desired control variables. For example, if we wish to track x_{cm} and y_{cm} , then:

$$\tau_2 = \frac{1}{\beta_y - \beta_x \frac{\gamma_y}{\gamma_x}} (\epsilon_x \frac{\gamma_y}{\gamma_x} - \epsilon_y - v_1 \frac{\gamma_y}{\gamma_x} + v_2) \quad (11)$$

$$\tau_1 = \frac{1}{\gamma_x} (-\epsilon_x - \beta_x \tau_2 + v_1) \quad (12)$$

Where v_1 and v_2 are calculated via PD feedback, to drive the system to desired references:

$$\begin{aligned} v_1 &= K_p(x_{ref} - x_{cm}) + K_d(\dot{x}_{ref} - \dot{x}_{cm}) \\ v_2 &= K_p(y_{ref} - y_{cm}) + K_d(\dot{y}_{ref} - \dot{y}_{cm}) \end{aligned} \quad (13)$$

We may also choose a different combination if desired, such as y_{cm} and θ_B , but we are interested in direct controllability of the center of mass location in order to accurately control the gait displacement of the robot. If a spring is included in the leg of the model, the lower body point accelerations \ddot{x}_b and \ddot{y}_b cannot be both instantaneously made to track references due to the radial force being that of a spring. This manifests in the PFL construction by β evaluating to zero for all X , thus to maintain our desired control of the CoM location we do not include a spring in the leg of our model.

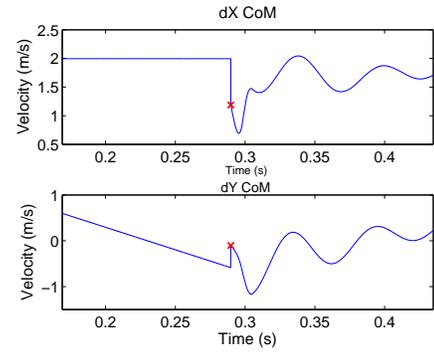


Fig. 3. Collision with the ground causes the CoM velocities to instantaneously change, misaligning the trajectory.

To control the linear and angular positions and velocities of the system, the GRF is carefully controlled. The trajectories we present in this work are generated such that the total net torque w.r.t. the CoM is zero, and thus under ideal conditions with no perturbations there would be no change in the angular momentum of the system, and the body angle would remain constant. We can set gait characteristics such as stride length by choosing a desired x_{cm} displacements and \dot{x}_{cm} take-off velocities.

III. DIVISION OF THE STANCE PHASE

By using our PFL controller we can specify references for the center of mass coordinates as a function of time, however the inelastic collision with the ground perturbs the trajectory at every step. Rather than thinking of the ground impact in terms of energy loss, we re-cast the problem from the point of view of our PFL controller. The ground impact causes only two major problems from this point of view:

- (1) Addition of large negative body angular velocity to the system caused by the velocity loss radially along the leg
- (2) Misalignment of the two initial conditions for velocity trajectories. As Fig. 3 shows, \dot{x}_{cm} and \dot{y}_{cm} are perturbed off the trajectory significantly by the impact, while positions are not affected by the impact.

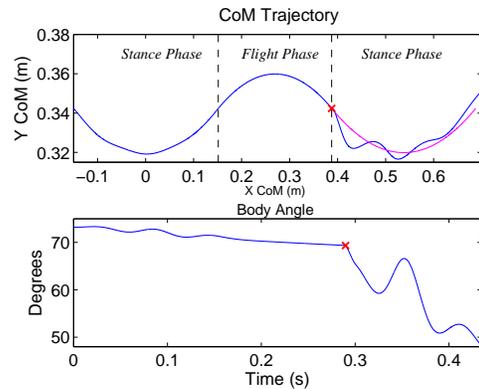


Fig. 4. Resulting CoM and body angle trends after a ground impact when attempting to simply use PFL control on x_{cm} and y_{cm} trajectories.

The errors caused by the impact are extremely large, and attempting to simply use the PFL control law on x_{cm} and

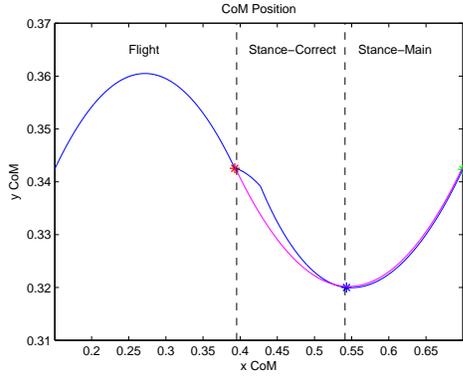


Fig. 5. The stance phase is divided into two phases. The Correction phase removes injected angular velocity and brings the system to the desired CoM (x, y) trajectory in space. Actual simulation data is shown in blue, and compared to Fig. 4 the system is significantly more stable.

y_{cm} is not robust and often unstable, as illustrated by Fig. 4. To address this problem, we divide the stance phase into two parts, *Correction* and *Main*, as shown in Fig. 5. The Main phase is the PFL implementation presented in Eq. 13, while the Correction phase instead controls y_{cm} and θ_B using the same PFL setup. This is done to robustly remove the extra angular velocity introduced into the system by the ground impact, which solves problem (1) above. Additional disturbances to the system due to ground level offsets (i.e., rough terrain) may be also treated as misalignments of the trajectory initial conditions at impact, and so this correction strategy is also design to improve robustness to disturbances, more generally.

After the correction phase, when body angular velocity is essentially zero, keeping angular velocity at zero is equivalent to ensuring the ground reaction forces point exactly at the center of mass. The reference trajectories for the nominal gait are designed specifically such that they are compatible with this constraint, ensuring the reference x - y trajectory is compatible with the body angle remaining constant.

A. Stance-Correction Phase

At the instant of ground impact, the body angle of the system may be above or below the desired system value. Therefore, two families of body angle trajectories are designed, so that each can be handled appropriately. These trajectories, shown in Fig. 6, are constructed using piece-wise exponentials, and constructed such that the initial positions and velocities are aligned with the values at impact. Trajectory A is chosen if the body angle is detected to above the desired value, and Trajectory B is chosen if the body angle is detected to be below the desired value.

After the body angle trajectory is chosen, we construct a valid trajectory for y_{cm} . The trajectory is selected such that x_{cm} , the uncontrolled variable, converges as closely as possible to the Stance-Main trajectory space. In order to design the y_{cm} trajectory, the x_{cm} trend during the Correction phase is estimated from the designed body angle trajectory. Both \dot{x}_{cm} and $\dot{\theta}_B$ are proportional to the x -dir

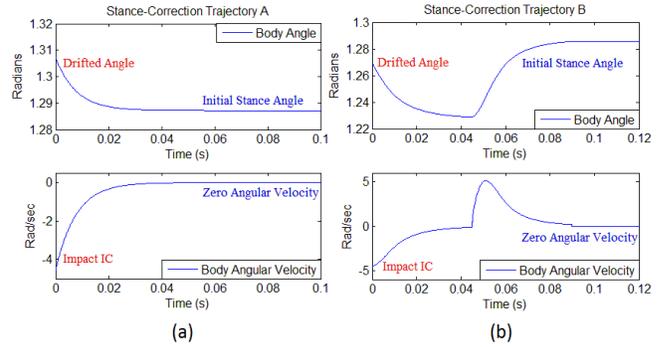


Fig. 6. Correction phase body angle trajectories for the two cases when the body angle has drifted (a) up and (b) down.

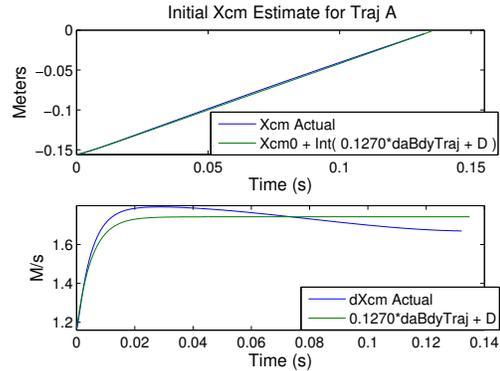


Fig. 7. The \dot{x}_{cm} trend during the Correction phase is estimated from the designed body angle trajectory. An estimate of the ending x_{cm} value is obtained via integration.

ground reaction force, F_x , so that \ddot{x}_{cm} and $\ddot{\theta}_B$ are strongly coupled during the Correction phase. It was experimentally determined that a good first order estimate of \dot{x}_{cm} can be obtained by scaling the designed $\dot{\theta}_B$ trajectory by roughly ten percent and aligning it to the initial \dot{x}_{cm} value. The overlay of these curves can be seen in Fig. 7, and the approximate x_{cm} value at the end of the Correction phase is obtained by integration. This estimate is significantly more accurate if Trajectory A is used.

We define the CoM trajectories as a function of x_{cm} by simply fitting the trajectories to polynomials as a function of x_{cm} , and define $y_{ref}(x_{cm})$, $\dot{y}_{ref}(x_{cm})$, and $\dot{x}_{ref}(x_{cm})$. The trajectory for y_{cm} is constructed by building a 3rd order polynomial such that the position and velocities are aligned to the values at impact, and the end values are calculated by simply evaluating $y_{ref}(x_{cm})$ and $\dot{y}_{ref}(x_{cm})$ for the estimated x_{cm} value at the end of the Correction phase from Fig. 7. A set of example trajectories constructed at runtime are shown in Fig. 8. The chosen trajectories for y_{cm} and θ_B are then implemented using PFL.

To characterize how well the system converges to the ideal CoM trajectory and switches to Stance-Main, we construct three error signals

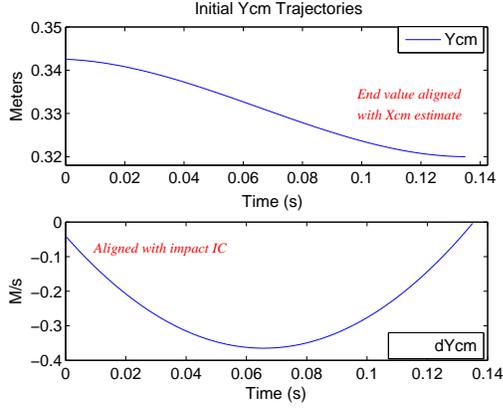


Fig. 8. Trajectories are generated for y_{cm} and \dot{y}_{cm} based on the body angle trajectory and the x_{cm} estimate.

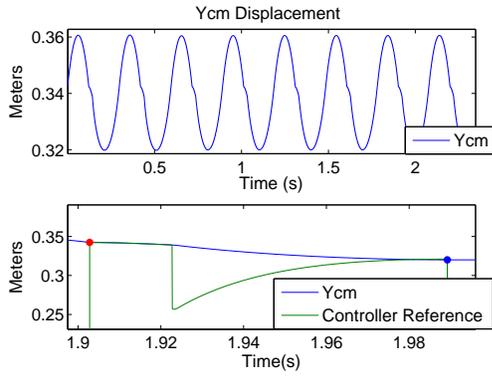


Fig. 9. After the controller has had time to correct for the angular velocity injected at impact, additional feedback terms activate based on the error generated from x_{cm} to improve convergence.

$$\begin{aligned}
 e_{y_{cm}} &= y_{ref}(x_{cm}) - y_{cm} \\
 e_{\dot{y}_{cm}} &= \dot{y}_{ref}(x_{cm}) - \dot{y}_{cm} \\
 e_{\dot{x}_{cm}} &= \dot{x}_{ref}(x_{cm}) - \dot{x}_{cm}
 \end{aligned} \tag{14}$$

If all three error signals are within given tolerances, the system is on the CoM trajectory and ready to switch to Stance-Main. We can further increase the accuracy of convergence by using two of these errors in feedback on the references relating to y_{cm} for v_2 in Eq. 13 as follows

$$\begin{aligned}
 y_{ref'} &= y_{ref} + K_1 e_{y_{cm}} \\
 \dot{y}_{ref'} &= \dot{y}_{ref} + K_2 e_{\dot{y}_{cm}}
 \end{aligned} \tag{15}$$

Convergence to Stance-Main is improved by using the additional feedback terms in Eq. 15, however Fig. 9 illustrates that an impulsive jump occurs in the reference signal. It is important to note that there is no feedback correction for $e_{\dot{x}_{cm}}$ in the Correction phase as for the other two errors seen in 15, and this will cause a subsequent error in the take-off velocity that must be later corrected in Stance-Main.

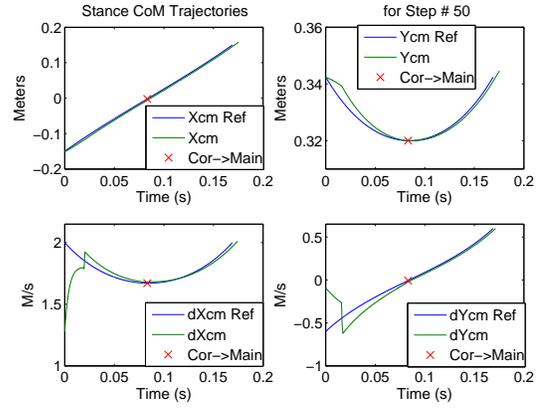


Fig. 10. Example Stance phase data taken after several steps showing convergence from Stance-Correction to Stance-Main. The velocities are misaligned at time zero from the ground impact.

B. Stance-Main Phase

Once the undesired angular velocity due to the ground impact has been essentially eliminated, the controller is capable of switching to the Stance-Main phase. The switch occurs when the Correction phase trajectory has ended, or the error signals in Eq. 14 are below a threshold, after which x_{cm} and y_{cm} are controlled via PFL techniques earlier introduced. Figure 10 shows example data taken from a simulation of the CoM states during the Stance phase, where it is clear that much of the error introduced by the ground impact is attenuated. However, the residual error left in the system will still cause the body angle to drift slightly during Stance-Main. Although the magnitude of the drift is small enough for the next Correction phase to remove, performance can be significantly impaired when operating on rough terrain.

To address this, the angular drift can be significantly reduced by adding an additional feedback term on $\dot{\theta}_B$ that penalizes the angular velocity. We can then add an additional term to have an effect on setting the body angle, which is beneficial because recall our initial correction phase estimate seen in Fig. 7 is more accurate when Trajectory A is used, therefore we encourage the system to keep the body angle slightly higher than the desired value at the end of Stance-Main by setting a higher reference. Figure 11 illustrates the effect of including this correctional control. A weighting value w can be used to set how much control effort to commit to the angular drift correction, as shown below in Eq. 16.

$$v_{1'} = v_1 + w(K_p(\theta_{ref} - \theta_B) + K_d(\dot{\theta}_{ref} - \dot{\theta}_B)) \tag{16}$$

Lastly, we must correct $e_{\dot{x}_{cm}}$ from the Correction phase, which causes a slow drift in the x_{cm} take-off velocity. Figure 12 shows example take-off velocity errors from a simulation on rough terrain, and illustrates that indeed the error in take-off velocity is directly proportional and nearly identical to $e_{\dot{x}_{cm}}$, and we account for this drift by simply adding a PD feedback term on the reference for \dot{x}_{cm} to Eq. 13, where n is the current step.

$$\dot{x}_{ref'} = \dot{x}_{ref} + K_3 e_{\dot{x}_{cm}}[n] - K_4 e_{\dot{x}_{cm}}[n-1] \tag{17}$$

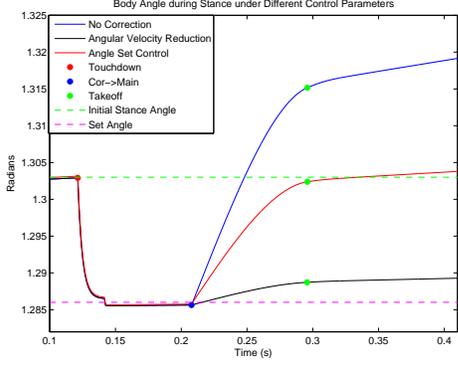


Fig. 11. Adding a feedback term on the body angle allows us to both reduce angular velocity drift during stance-main, shown above between the blue and green markers, and also set the ending body angle during stance to some unmeasured degree of accuracy. We do this to prevent drift and encourage the use of Trajectory A during Stance-Correction.

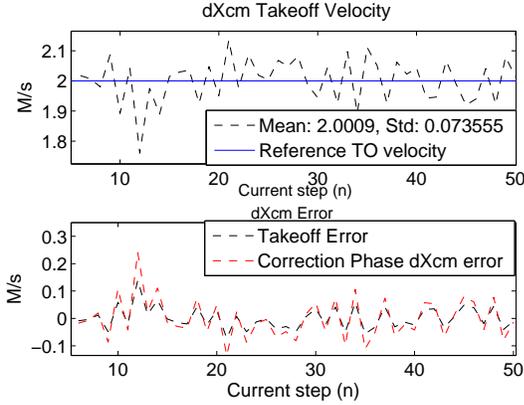


Fig. 12. Above is an example of takeoff velocity and internal Correction phase convergence errors operating on rough terrain. Since the errors have an almost identical profile, we use the error from the Correction phase to implement the takeoff velocity drift correction in order to prevent the control from being delayed by a step.

Figure 13 shows that the drift is successfully corrected with this feedback term. During the flight phase, only actuator τ_2 has any effect on the configuration of the masses during the ballistic trajectory, and it is used simply to drive the spine angle θ_3 toward the initialized stance value.

IV. TRAJECTORY SWITCHING

Since our controller needs only CoM stance phase trajectories to operate, we can control the gait step length by choosing trajectories with different terminating velocities. Given two trajectories with different \dot{x}_{cm} terminating values, we construct asymmetric trajectories that switch to or from the desired trajectory set. To illustrate the ability to switch between different gaits, we designed three CoM trajectory sets for "Small", "Medium", and "Large" strides, where the difference in step length between each set is roughly ten percent. Figure 14 shows the trajectory set for the "Large" strides and the asymmetric trajectories to switch from "Medium" to "Large".

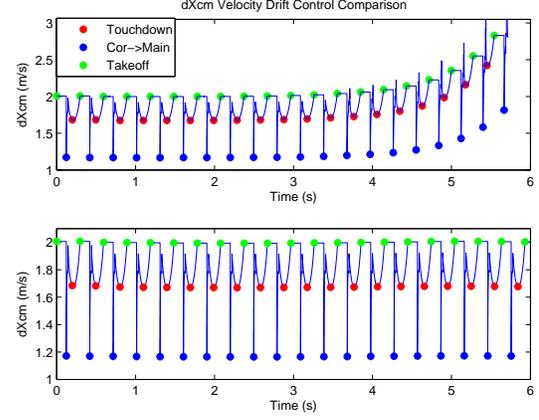


Fig. 13. The take-off velocity for x_{cm} will slowly drift over time eventually causing failure, but can be fixed with an additional feedback term.

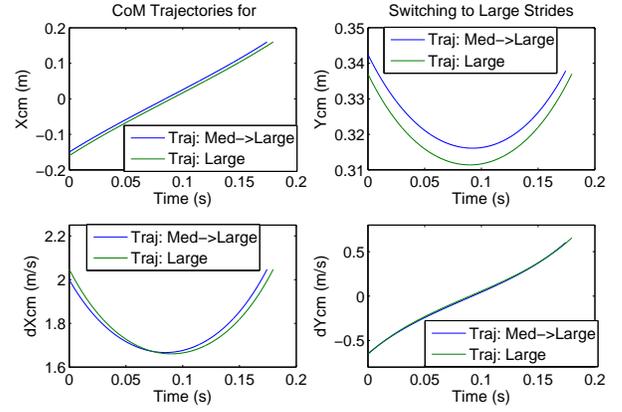


Fig. 14. Example CoM trajectories used for the "Large" strides. The magnitude of the x_{cm} velocity at take-off is larger than the "Medium" trajectories shown in Fig. 10, which results in larger strides. Note the asymmetry in the switching trajectory shown in blue.

V. RESULTS ON ROUGH TERRAIN

By utilizing the Correction phase to remove erroneous body angle perturbations, our control system is not only able to correct the trajectory errors from the ground impact, but is also fairly robust when operating on rough terrain. The effect of the terrain being uneven is essentially the same as an additional unknown misalignment of the y_{cm} trajectory, since the ground level is not zero. The terrain noise is generated by a Gaussian distribution of height offsets per step, and classified based on the maximum disturbance value for a given simulation. The effect of the rough terrain can be observed by viewing x_{cm} , y_{cm} , and θ_B at touchdown and the switch from Correction to Main. When the terrain is even, there is very little variation of y_{cm} for a given stride length, but x_{cm} has significant variance due to $e^{\dot{x}_{cm}}$ drift. When rough terrain is added, the value of y_{cm} at touchdown has significant variance, and this is clearly seen in Fig. 15. Since the Correction phase is built to correct for misalignments, the variance is significantly reduced at the time at which the controller switches to Stance-Main, illustrated in Fig. 16.

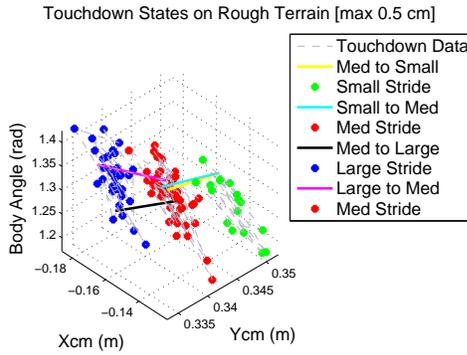


Fig. 15. Touchdown CoM states for a simulation on rough terrain. The controller switches to different strides in the following order: med-small-med-large-med. The states are significantly scattered due to uncertain ground level at impact.

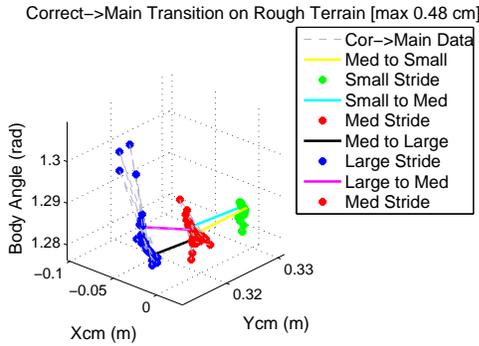


Fig. 16. Correction to Main CoM transition states for rough terrain. The controller is able to consistently correct for the large errors at touchdown, and bring the states to more localized points.

Additionally, the body angle is very consistent at the time of the switch, which can be noted by observing that the scale of the body angle axis in Fig. 16 is significantly more confined than at touchdown.

The system is stable when switching to different stride lengths for up to 1cm of rough terrain, after which failures can occasionally occur. Figures 17 - 19 show step length simulation results for varying terrain level, where the variance of each stride length is clearly proportional to the terrain noise level, but still stable. The roughest terrain stably traversed in a switching simulation was 2.25cm, roughly 10 percent of the leg length of the robot, and is shown in Fig. 20.

VI. CONCLUSIONS AND FUTURE WORK

We have presented control methods for an underactuated quadruped model with a flexible spine that allows for direct CoM trajectory planning. We provide a robust control method for correcting errors caused by the inelastic collision with the ground at each step. Our system is capable of switching to different trajectories at runtime, resulting in variable stride lengths, and remains stable operating on rough terrain. We focus on tracking a sequence of trajectories for a

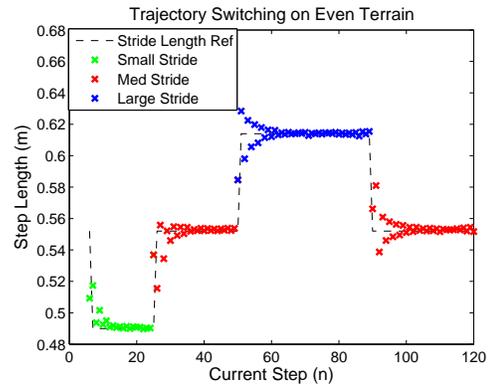


Fig. 17. Trajectory switching simulation on even terrain.

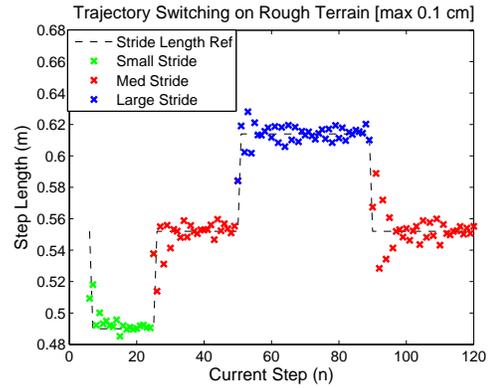


Fig. 18. Trajectory switching simulation on 0.1cm rough terrain.

single leg in support with the ground, robust to variations in ground height and other variability, which enables switching among gaits to vary foothold selection and speed.

Future work includes designing optimal trajectories for the system to follow that include bounding gaits that alternate between front and rear stance feet, as well as trajectories that consider energy efficiency by attempting to plan ahead for the upcoming ground impact. We also plan to revisit the possible inclusion of a spring in the leg to further improve energy efficiency while maintaining high controllability. Lastly, considering real-world problems such as actuator limitations, sensor accuracy, and state estimation are also areas for further study.

REFERENCES

- [1] M. H. Raibert, M. Chepponis, and H. Brown, "Running on Four Legs As Though They were One," *IEEE Journal of Robotics and Automation*, vol. 2, no. 2, pp. 70-82, 1986.
- [2] M. S. Fischer, N. Schilling, M. Schmidt, D. Haarhaus, and H. Witte, "Basic limb kinematics of small therian mammals", *J Exp Biol*, vol. 205, no. 9, pp 1315-1338, 2002
- [3] B. Boszczyk, A. Boszczyk, and R. Putz, "Comparative and functional anatomy of the mammalian lumber spine", *Anatomical Record*, vol. 264, no. 2, pp 157-168, 2001
- [4] Karl F. Leiser, "Locomotion Experiments on a Planar Quadruped Robot with Articulated Spine", *Master's Thesis*, MIT, 1996.
- [5] B. M. Hauelsen, "Investigation of an Articulated Spine in a Quadruped Robotic System", *PhD Thesis*, University of Michigan, 2011.
- [6] R. Player, M. Beuhler, and M. Raibert, "BigDog", in *Soc. of Photo-Optical Instr. Eng. (SPIE) Conf. Series*, vol. 6230, Jun. 2006.

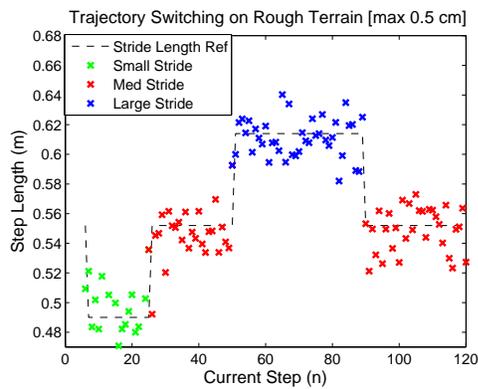


Fig. 19. Trajectory switching simulation on 0.5cm rough terrain.

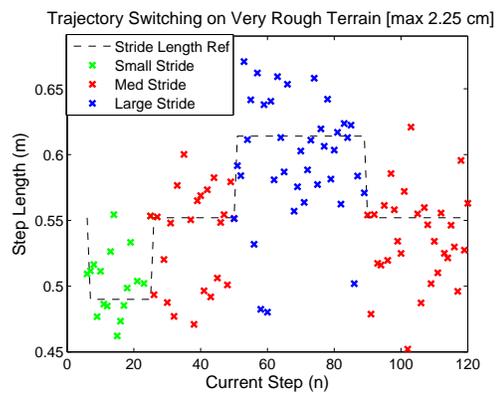


Fig. 20. Trajectory switching simulation on extremely rough terrain. The system is not always stable for terrain levels of this magnitude.

- [7] M. Kani, M. Derafeshian, H. Bidgoly, and M. Ahmadabadi, "Effect of flexible spine on stability of a passive quadruped robot: Experimental results", *Robotics and Biomimetics (ROBIO), IEEE International Conference*, 2011.
- [8] M. Khoramshahi, A. Sprowitz, A. Tuleu, M. N. Ahmadabadi, A. J. Ijspeert, "Benefits of an Active Spine Supported Bounding Locomotion With a Small Compliant Quadruped Robot", *IEEE International Conference on Robotics and Automation*, 2013.
- [9] U. Culha, U. Saranlı, "Quadrupedal Bounding with an Actuated Spinal Joint", *IEEE International Conference on Robotics and Automation*, 2011.
- [10] Mark W. Spong, "Underactuated Mechanical Systems", *Control Problems in Robotics and Automation*, pp. 135-150, 1997.
- [11] S. Kochuvila, S. Tripathi, S. T. S. B., "Control of a Compass Gait Biped Robot Based on Partial Feedback Linearization", *Joint Proceedings of the 13th Annual TAROS Conference and the 15th Annual FIRA RoboWorld Congress*, 2012.
- [12] Mark W. Spong, "Partial Feedback Linearization of Underactuated Mechanical Systems", *Intelligent Robots and Systems (IROS)*, pp. 314-321, 1994.
- [13] Mark W. Spong, "Swing Up control of the Acrobot Using Partial Feedback Linearization", *Proc. American Control Conference*, pp. 2158-2162, Baltimore, MD, 1994.
- [14] G. Piovan and K. Byl, "Enforced Symmetry of the Stance Phase for the Spring-Loaded Inverted Pendulum", *Proc. IEEE Int. Conf. Robotics and Automation (ICRA)*, 1908-1914m 2012.
- [15] K. Byl, M. Byl, M. Rutschmann, B. Satzinger, Lous van Blarigan, G. Piovan, Jason Cortell, "Series-Elastic Actuation Prototype for Rough Terrain Hopping," in *Proc. IEEE International Conference on Technologies for Practical Robot Applications (TePRA)*, pp. 103-110, 2012.
- [16] Cavagna GA, Thys H, Zamboni A., "The sources of external work in level walking and running", *J Physiol*, 262:639-57, 1976.
- [17] K. Sasaki, R. R. Neptune, "Muscle mechanical work and elastic energy utilization during walking and running near the preferred gait transition speed", *Gait and Posture* 23, 383-390, 2006.