A Partially Observable Hybrid System Model for Bipedal Locomotion for Adapting to Terrain Variations

Koushil Sreenath
University of Pennsylvania
3300 Walnut St.
Philadelphia, PA 19104
koushils@seas.upenn.edu

Connie R Hill Jr.
Lockheed Martin Advanced Technology Laboratories
3 Executive Campus
Cherry Hill, NJ 08002
connie.r.hill.jr@lmco.com

Vijay Kumar
University of Pennsylvania
3300 Walnut St.
Philadelphia, PA 19104
kumar@seas.upenn.edu

ABSTRACT

We propose a methodology of applying PoMDPs at a sufficiently high abstraction of a high-dimensional continuous-time partially observable hybrid system. In particular, we develop a two-layer hybrid controller, where the higher-level PoMDP-based hybrid controller learns the boundaries between various modes and appropriately switches between them. The modes partition the state-space and represent a closed-loop hybrid system with a lower-level hybrid controller. We apply this methodology onto the problem of bipedal walking on varying terrain, where the gradient change in the terrain is only partially observable (due to poor and noisy sensors.) We develop three lower-level hybrid controllers that result in robust walking on level ground, up and down ramps. The higher-level PoMDP-based hybrid controller then learns the boundary between these controllers and is used to perform appropriate controller switching. With only a coarse, discrete estimate of walking speed, the controller enables traversing terrain both with long sustained constant slopes, and with rapid changes in slope. Simulation results are presented on a 26-dimensional planar bipedal robot model that incorporates contact forces and friction.

Categories and Subject Descriptors

H.4 [**Information Systems Applications**]: Miscellaneous

General Terms

Theory

1. INTRODUCTION

Partially Observable Markov Decision Processes (PoMDPs) are good for solving problems with partial observability of

HSCC'13, April 8–11, 2013, Philadelphia, Pennsylvania, USA.

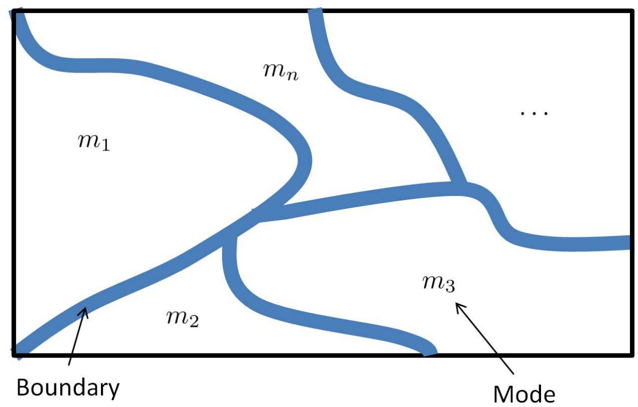


Figure 1: An abstract illustration of a two-layer hybrid controller with different modes separated by boundaries. Each mode is a closed-loop hybrid system with a lower-level hybrid controller, comprised of an inner-loop continuous-time controller and a discrete-time event-based outer-loop. Additionally, a higher-level hybrid controller learns the boundaries between the modes, by solving the postulated PoMDP problem, and then achieves switching between them.

systems with uncertainty in the states [11, 4, 1], however they are plagued by the problem of state explosion. Here we apply PoMDPs at a sufficiently high abstraction of a high-dimensional continuous-time partially observable hybrid system. In particular, we develop a two-layer hybrid controller, where the higher-level PoMDP-based hybrid controller learns the boundaries between various modes and appropriately switches between them. The modes partition the state-space and represent a closed-loop hybrid system with a lower-level hybrid controller. Similar ideas exist for simpler systems [5, 6].

Essentially, our method proposes a two-layer hybrid controller hierarchy, with multiple discrete modes that partition the state space; see Figure 1. Each mode is a closed-loop hybrid system with a *lower-level* hybrid controller, comprised of an inner-loop continuous-time controller and a discrete-time event-based outer-loop. Additionally, an *higher-level* hybrid controller learns the boundaries between the modes, by solving the postulated PoMDP problem, and then achieves switching between them. We have a nested hybrid system and the proposed methodology can be applied at any level.

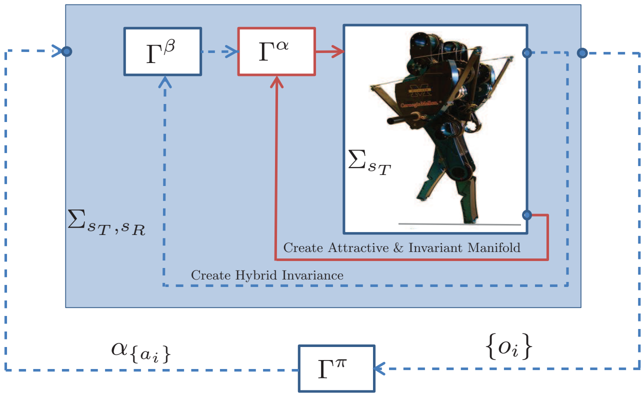


Figure 2: Feedback diagram illustrating the PoMDP-based hybrid controller structure. Continuous lines represent signals in continuous time; dashed lines represent signals in discrete time. The controllers Γ^α and Γ^β create an attractive and invariant hybrid zero dynamics [15]. Σ_{s_T}, and Σ_{s_T,s_R} are open-loop, and closed-loop partially observable hybrid systems respectively. The controller Γ^π is an hybrid controller that implements a PoMDP policy to switch the walking gait (specified by the α parameters) based on the observations that are being received from the hybrid system (marked by the blue box).

We chose to apply it at the highest level because, as we will see, this is where our state is not fully observable.

We particularize the method for studying the problem of designing feedback controllers for structured terrain where the gradient changes discretely, but is not accurately perceivable either by a tactile sensor through forward kinematics or a vision system. For instance, indoor corridors and sidewalks with gradual changes in slope are examples of such terrain. Walking on varying slopes has been considered in [14, 3], however these assume that the slope can be perfectly known or can be inferred at each step. Bipedal walking on rough terrain has been primarily addressed by developing controllers that are robust to bounded variation in step heights of the terrain [7, 2]. These controllers are typically hybrid, with a continuous-time inner-loop controller and a discrete-time event-based outer-loop controller designed to provide robustness to changes in the terrain.

However, a single controller can not be easily designed to achieve walking at multiple ground slopes, but rather multiple locally stable controllers that are robust about different specific ground slopes are sought, along with some higher-level sequential decision making to switch between the controllers. Partially observable Markov decision process (PoMDP), [1], provide a natural model for sequential decision making under uncertainty, and is particularly applicable to situations where a robot cannot reliably identify the state of the underlying environment. Although switching controllers have been demonstrated in the literature for rough terrain walking [3, 9], these approaches do not consider any uncertainty of when to switch.

We apply the two-layer PoMDP hybrid controller framework to a special case of three simple discrete modes for walking on level ground, walking on a up ramp and a down ramp of specified gradient. For each of these modes, there is a continuous-time controller for tracking, and a discrete-time event-based controller to provide additional robustness to adapt to small changes in terrain gradient. The higher-level PoMDP-based hybrid controller learns the boundaries between the discrete modes based on the stochastic distributions specified as part of the PoMDP problem, and provides a way to switch between the low-level hybrid controllers. Note that we are not considering the quasi-static case [12], but we are using a PoMDP formulation to solve a problem in which the dynamics and real-time control are critical for operation of the system.

Employing only three specific walking gaits, our proposed method is able to successfully walk on level ground and sustained up and down slopes of 10° while also working in situations where the slope changes every step. Moreover, this method can be easily extended to have additional modes (rather than the 3 illustrated here) corresponding to walking on different ground slopes.

Note that most work on rough terrain walking assumes the availability of precise ground height, and/or walking step speed. However, for physical legged robots, computing small changes in ground height at impact using forward kinematics is error-prone due to sensor and calibrations errors that add up as the kinematic chain is traversed. Moreover, relying on visual or laser sensors for estimating small changes in ground slope is also not feasible. Furthermore, accurate forward velocity, usually obtained by fusing odometry and visual information, is not available for legged robots due to the error-prone odometry. The proposed method sidesteps these issues by only requiring a coarse estimate of the walking speed (speed roughly equal to, greater than, or less than nominal speed) for estimating the terrain gradient.

The rest of the paper is organized as follows. Section 2 briefly presents the hybrid model for walking and the nominal controller used for walking. Section 3 develops and formulates a PoMDP problem as a Kronecker product of separate PoMDPs for modeling the terrain and the robot. Section 4 presents simulation results by evaluating the postulated PoMDP-based hybrid controller on a high-dimensional planar bipedal model that captures unilateral ground constraints and stick-slip friction, to demonstrate walking on stochastically varying terrain. Finally, Section 5 presents concluding remarks.

2. DYNAMICAL MODEL AND NOMINAL CONTROL DESIGN FOR WALKING

We have seen several methods for handling rough terrain in the previous section, with researchers considering simple systems for demonstrating their method. We will illustrate the PoMDP-based control design on a dynamical model of a real-life complex experimental system called MABEL, a planar bipedal robot at The University of Michigan, which has an underactuated, compliant transmission [15].

A dynamical model for walking can be developed by modeling the single support phase, when one foot is assumed to be pinned to the ground, and the subsequent double support phase that occurs when the swing foot makes contact with the ground. The single support phase is modeled as the continuous-time dynamics of a pinned, planar, kinematic chain with revolute joints and rigid links, while the double support phase is modeled as an instantaneous impact. The hybrid model for walking is then as follows,

$$\Sigma_{s_T}: \begin{cases} \dot{x} = f(x) + g(x)u, & x^- \notin \mathcal{G}_{s_T} \\ x^+ = \Delta(x^-), & x^- \in \mathcal{G}_{s_T}. \end{cases} \tag{1}$$

where $x = \begin{bmatrix} q & \dot{q} \end{bmatrix} \in \mathcal{X}$ is the state of the system, f, g are vector fields that captures the continuous-time dynamics, Δ represents the impact map that maps the pre-impact state x^- to the post-impact state x^+, and $\mathcal{G}_{s_T} = \{x \in \mathcal{X} \mid H_{s_T}(x) = 0\}$ is the guard surface representing contact of the swing foot with the ground, with $s_T \in \mathcal{S}_T := \{L, U, D\}$ representing if the terrain is level, a up ramp, or a down ramp. In the absence of sensors for precisely measuring small variations in the terrain either through forward kinematics, or through vision, the terrain state is only partially observable, making this a *partially observable hybrid model*.

To coordinate and synchronize the links of a robot to achieve the objectives of walking, such as keeping the torso upright, advancing the swing foot to take a step, maintaining a foot clearance for a specific slope of the terrain, etc., we use *virtual* constraints [15]. Virtual constraints are holonomic constraints on the robot's configuration variables that are asymptotically imposed through feedback control. One virtual constraint is typically chosen per actuator in the form of an output, such that when a feedback controller drives the output to zero, the constraint is enforced. Virtual constraints can be written as,

$$y = h(q, \alpha) = H_0 q - h_d(\theta(q), \alpha), \tag{2}$$

where, H_0 is a selection matrix for choosing the variables to be controlled, $h_d(\theta, \alpha)$ are the desired trajectories, expressed as Bézier polynomials, parametrized by θ, a monotonous function of the joint variables, and α the Bézier polynomial coefficients. The α-parameters are chosen through a constrained nonlinear optimization to obtain a periodic gait for walking. For the purpose of this paper, we design a set of periodic walking gaits, represented by the parameters α_{s_R}, with $s_R \in \mathcal{S}_R := \{WL, WU, WD\}$, designed for walking on level ground, and up, down ramps of a 10° gradient respectively.

Next we present the lower-level hybrid controller, comprised of a continuous-time inner-loop controller and a discrete-time outer-loop controller, that locally, exponentially stabilizes the periodic walking gaits. Assuming the output (2) has vector relative degree two, the continuous-time controller Γ^α is given by the input-output linearizing controller,

$$\Gamma^\alpha: \quad u = -L_g L_f h(q, \alpha)^{-1} \left(L_f^2 h + K_p y + K_d \dot{y} \right). \tag{3}$$

This controller locally drives the output to zero exponentially and creates the invariant manifold $\mathcal{Z} = \{x \in \mathcal{X} \mid y = 0, L_f y = 0\}$.

The discrete-time event-based controller Γ^β serves to perform step-to-step parameter updates by adding an additional term to the virtual constraint defined in (2) to get a new output,

$$\Gamma^\beta: \quad y_b = h(q, \alpha, \beta) = H_0 q - h_d(\theta(q), \alpha) - h_b(\theta(q), \beta). \tag{4}$$

The event-based control is used to select β so as to perform corrections to the virtual constraint to obtain hybrid invariance, i.e., $x^- \in \mathcal{Z} \implies x^+ \in \mathcal{Z}$ [15, Sec. IV-B], at each step, thereby smoothly handling transients.

Finally, we present a coarse speed sensor that indicates if the walking step speed is greater than, equal to, or lesser than the nominal speed, by providing an observation $o_R \in \mathcal{O}_R := \{Sp, S0, Sm\}$, according to the following rule,

$$o_R = \begin{cases} Sp, & v > (1+\delta)\ v_{s_R} \\ S0, & (1-\delta)\ v_{s_R} < v < (1+\delta)\ v_{s_R} \\ Sm, & v < (1-\delta) v_{s_R}, \end{cases} \tag{5}$$

where v_{s_R} corresponds to the steady-state speed of the walking gait α_{s_R}, for $s_R \in \mathcal{S}_R$, and δ a small positive number used as a threshold for discretization by the sensor.

$A_T = \{\}$	$s' = L$	U	D
$s = L$	0.8	0.1	0.1
U	0.35	0.6	0.05
D	0.35	0.05	0.6

(a)

$A_T = \{\}$	$O_T = \{\}$
$s = L$	1
U	1
D	1

(b)

Table 1: (a) State transition, $T_T(s, \{\}, s')$, and (b) observation functions, $Z_T(\{\}, s, \{\})$, for the terrain PoMDP, $\mathcal{P}_T$.

3. POMDP FORMULATION FOR WALKING ON SLOPES

The hybrid system Σ_{s_T} in (1) under the action of the lower-level hybrid controller (3), (4) is another (closed-loop) hybrid system Σ_{s_T, s_R}, represented by the blue box in Figure 2. In this section, we will develop a higher-level hybrid controller for this system. Note that we are able to analyze the lower-level controller developed in the previous section, and the higher-level hybrid controller developed here separately. This is primarily due to the loose coupling between the two that occurs through the incoming discrete speed observations (5), which in turn indirectly depend on the continuous states and interactions with the ground.

Here, we will assume that we neither have access to precise foot height at contact, nor accurate estimates of step speed (due to poor and noisy sensors.) Rather, we will assume we have a coarse estimate of step speed that only indicates if the speed is roughly equal to, greater than, or lesser than nominal speed as in (5). It must be noted that just looking at the speed for a single step is not sufficient to immediately determine the state of terrain so as to switch to an appropriate walking controller. Instead, we will need to pose the problem as a PoMDP, that will enable us to learn the boundaries between when each of the three walking controllers, developed in Section 2, for walking on level ground, up and down ramps, are active and when to switch to the appropriate one, based on a sequence of observations of the coarse speed estimate.

A partially observable Markov decision process (PoMDP) is a Markov decision process that does not make an assumptions that the states are directly observable [11, 4, 1]. A PoMDP can be characterized by a 6-tuple $\mathcal{P} = (\mathcal{S}, \mathcal{A}, \mathcal{O}, T, Z, R)$, where $\mathcal{S} = \{s_1, s_2, ..., s_n\}$ is a set of states that the system can be in, $\mathcal{A} = \{a_1, a_2, ..., a_m\}$ is a set of actions that can be applied to affect the system, $\mathcal{O} = \{o_1, o_2, ..., o_l\}$ is a set of observations that describe the perception of the system which partially reflects the current state of the system, $T: \mathcal{S} \times \mathcal{A} \to \Pi(\mathcal{S})$ is the state transition function that maps each state-action pair into a probability distribution over the state space, such that $T(s, a, s') = Pr(S_{t+1} =$

$a = WL$	$s' = WL$	WU	WD
$s = WL$	1	0	0
WU	1	0	0
WD	1	0	0

$a = WU$	$s' = WL$	WU	WD
$s = WL$	0	1	0
WU	0	1	0
WD	0	1	0

$a = WD$	$s' = WL$	WU	WD
$s = WL$	0	0	1
WU	0	0	1
WD	0	0	1

(a)

$a = WL$	$o = Sp$	$S0$	Sm
$s = WL$	0.1	0.8	0.1
WU	0.7	0.2	0.1
WD	0.1	0.2	0.7

$a = WL$	$o = Sp$	$S0$	Sm
$s = WL$	0.04	0.06	0.9
WU	0.1	0.7	0.2
WD	0.02	0.03	0.95

$a = WL$	$o = Sp$	$S0$	Sm
$s = WL$	0.9	0.06	0.04
WU	0.95	0.03	0.02
WD	0.2	0.7	0.1

(b)

Table 2: (a) State transition, $T_R(s, a, s')$, and (b) observation functions, $Z_R(a, s, o)$, for the robot PoMDP, $\mathcal{P}_R$.

$s' \mid S_t = s, A_t = a)$, $Z : \mathcal{A} \times \mathcal{S} \to \Pi(\mathcal{O})$ is an observation function that maps the current state and the previous action to a distribution over the observations, such that $Z(a, s, o) = Pr(O_t = o \mid S_t = s, A_{t-1} = a)$, and finally $R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the immediate reward function.

At any given point in time the system is in state s_t, which is partially observable through observation o_t, with probability $b_t = Pr(s_t \mid o_t, a_t, o_{t-1}, a_{t-1}, \cdots, o_0, a_0)$. This *belief* distribution represents the entire history of the interaction of the system. The goal of a PoMDP is to learn an optimal policy describing action selection, that maximizes the expected discounted cumulative reward, i.e.,

$$V_t^*(s) = \max_{a \in \mathcal{A}} \left[R(s, a) + \gamma \sum_{s' \in \mathcal{S}} T(s, a, s') V_{t-1}(s') \right], \quad (6)$$

where γ is the discount, with V being the value function. The *policy* is defined as a mapping from the belief state to action state.

We model two PoMDPs, one for the terrain, $\mathcal{P}_T = (\mathcal{S}_T, \mathcal{A}_T, \mathcal{O}_T, T_T, Z_T, R_T)$, and the other for the robot, $\mathcal{P}_R = (\mathcal{S}_R, \mathcal{A}_R, \mathcal{O}_R, T_R, Z_R, R_R)$, and then form a composite PoMDP by taking the Kronecker product of the two, $\mathcal{P} = \mathcal{P}_T \otimes \mathcal{P}_R$. We model the terrain PoMDP as follows, $\mathcal{S}_T = \{L, U, D\}$ as defined earlier represents the state of the terrain to be either level, a up ramp, or a down ramp, and with no actions or observations, i.e., $\mathcal{A}_T = \{\}, \mathcal{O}_T = \{\}$. Further, we choose the state transition and observation functions as given in Table 1. These probabilities were chosen arbitrarily and do not reflect true values of terrain distributions occurring in the real world. However, it would be pretty straightforward to collect real-world data and compute these probabilities.

Next, we model the robot PoMDP as follows, $\mathcal{S}_R = \{WL, WU, WD\}$ as defined earlier represents the state of the robot controller to be either a level walking controller, a up ramp walking controller, or a down ramp walking controller. In a similar way, we choose the actions $\mathcal{A}_R = \{WL, WU, WD\}$ to represent the transition to these controllers (Note that we have abused notation and have used the same symbols for states and actions. The state represents the current controller being used on the robot, while the action represents what controller should be used next.) The observations, $\mathcal{O}_R = \{Sp, S0, Sm\}$ as defined earlier, indicate if the speed is greater than (Sp = speed plus), equal to, or lesser than (Sm = speed minus) nominal walking speed. These observations provide an indirect estimate of the change in the terrain and do not require precise accurate sensors for measuring the speed of walking. Further, we choose the state transition and observation functions as given in Table 2. To keep the reward structure simple, we choose unit reward for both PoMDPs, i.e., $R_T \equiv 1, R_R \equiv 1$.

Remark 1. *Decomposing our PoMDP into the terrain and robot PoMDPs makes presentation of the state transition and observation functions in Tables 1, 2 compact. The state transition and observation functions for the composite PoMDP are then easily obtained from these. Moreover, in the future, we can easily model a visual sensor as part of the observations in $\mathcal{P}_T$, to provide a direct estimate of the terrain. This enables, easily fusing observations from a visual sensor providing direct estimates of the terrain in $\mathcal{P}_T$, and a discrete speed sensor providing an indirect estimate of the terrain in $\mathcal{P}_R$.*

The composite PoMDP thus formed is then solved using a solver such as ZMDP, Perseus, Cassandras, or Pegasus [13, 1, 8]. Figure 4 illustrates the resulting policy graph. The obtained policy is essentially a finite state controller that describes what actions to take based on the observations received. The received observations are used to update the belief, and when the belief of a particular state is high enough, an appropriate action is issued. Thus, the outer-loop control Γ^π is specified as,

$$\Gamma^\pi : \quad \alpha = \alpha_{\pi(o_t)}, \quad (7)$$

where o_t is the observation received at discrete time instant t, and $a_t = \pi(o_t)$ is the action specified by the policy, and the α-parameters corresponding to the action are picked.

4. RESULTS

We now evaluate the proposed controller, comprising of the continuous-time controller with the discrete-time outer-loop controller as described in Section 2, and the PoMDP hybrid controller developed in Section 3, on the complex model of MABEL developed in [10]. This model is 26-dimensional and captures unilateral constraints through a compliant ground and stick-slip friction model. This captures the robot-ground interaction more realistically and is important for testing controllers especially on terrain that is varying, where foot slippage is bound to happen and is an important source of failure. This is in contrast to papers that do not model stick-slip friction while evaluating proposed rough terrain controllers.

As an initial test of validity of the control design, we consider a deterministically generated terrain that consists of segments of a level ground, a up ramp at 10°, a level ground and a down ramp at 10°; see Figure 3a. This sample terrain will test if the PoMDP policy correctly switches to an appropriate walking controller.

In this simulation, the robot is able to take over 65 steps to traverse the entire terrain, with the PoMDP correctly switching the controller at appropriate instants in time. The PoMDP policy is able to transition from $WL \rightarrow WU$, $WU \rightarrow WL$, $WL \rightarrow WD$, $WD \rightarrow WL$. Step speeds, observations received, which are computed using the rule in (5), and the actions generated by the PoMDP policy-based controller Γ^{π} are also illustrated. Specifically, the coarse discrete speed measurement indicating if the speed is approximately equal to, greater than or lesser than the nominal speed, is used to build a belief in the state of the partially observable hybrid model and take an appropriate action based on the belief, enabling switching to an appropriate controller, thereby adapting to and traversing the terrain.

Next, we test the controller on a suite of stochastically generated terrains based on the terrain state transition in Table 1. It must be noted that the PoMDP only captures the transitions of the terrain. Neither the length of terrain segments nor the specific slope to choose for the up/down ramps are specified by the PoMDP. Instead of picking just one particular value, we randomly choose a segment length from a uniform distribution in $\{2m, 4m, 6m\}$ and a slope for up ramp and another slope for down ramp from $\{5°, 7.5°, 10°\}$. With the segment length and the slopes for the up / down ramps fixed, we generate a terrain based on the terrain transition function. This forms one sample path to test the controller on, and we repeat the above procedure to obtain additional sample paths. The proposed PoMDP controller is able to successfully complete walking on over 85% of the generated terrain, some of which involve over 100 steps of walking down at a 10° incline. The most common reason for failure is an inability of the controller to accommodate changes in terrain slope of over 10°, for instance when the up ramp changes to a down ramp involving a change in slope of 20°. A small percentage of failures can be attributed to the foot slipping while going up a ramp, which by itself does not cause the robot to fall, but instead causes the swing foot to advance rapidly to catch the robot from falling, causing subsequent steps to be taken quickly, which increases the speed of walking. This results in the PoMDP estimating a wrong terrain gradient thereby not transitioning to, the correct controller.

Finally, we test the controller on a stochastically generated terrain, where the slope of the terrain changes every 1m with the terrain state transition as in Table 1. The slopes of ramp for each segment are selected from an uniform distribution of slopes in $\{2°, 4°, 6°, 8°, 10°\}$. Simulation results shown in Figures 3b, illustrate that the robot is able to traverse terrain not only when the slope is constant for a long duration, but also when the slope changes rapidly.

5. CONCLUSION

A two-layer hybrid controller methodology for applying PoMDPs at a sufficiently high abstraction of a high-dimensional continuous-time partially observable hybrid system has been presented. This has been particularized for application onto planar bipedal walking on stochastically varying slopes, where neither an assumption of requiring an estimate of step height, nor requiring an accurate estimate of the step speed at each step is made. With only a coarse estimate of step speed, the presented PoMDP-based hybrid controller is able to appropriately switch between three hybrid controllers designed for robust walking on level ground, up and down ramps of 10°. Simulations carried out on a model of MABEL illustrate the controller being able not only to traverse terrain with long and sustained constant slopes, but also ones where the slope changes more rapidly. The gradient of the ramps addressed in this paper can easily be increased. The method itself can easily be extended to multiple ground slopes and the controllers can be stitched together by appropriate switching. The only assumption is that a periodic walking gait at that slope is available. As future work, we are exploring extending this framework to address rough and difficult terrain in addition to the presented gentle ramps.

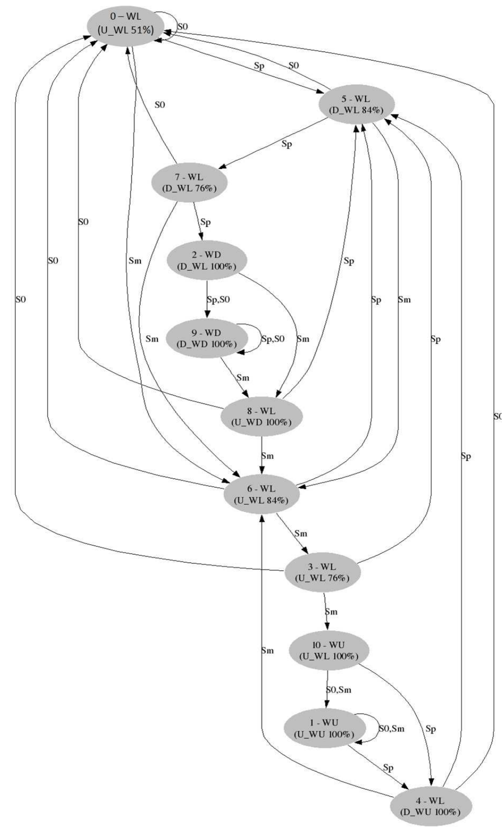
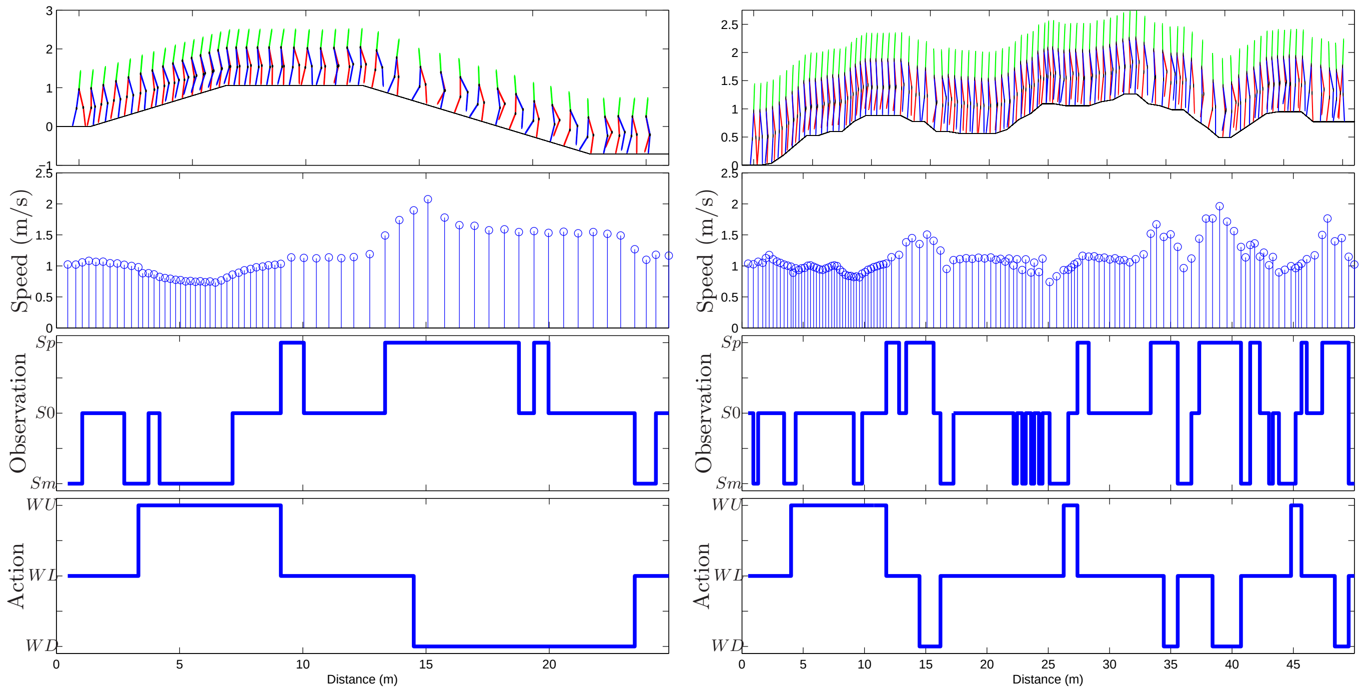


Figure 4: The policy graph obtained by solving the postulated PoMDP. The obtained policy has 11 nodes, with each node specifying the action to be taken, the state of the composite PoMDP, $\mathcal{P}$, that has the highest belief, along with the belief expressed as a percentage. An appropriate node transition occurs when the specified observation is received from the system. From the policy it can be seen that to switch from a level walking gait to a walking up (or down) controller requires three consecutive observations of S_p, speed up (or S_m, speed down), as seen in the node transitions $0 \rightarrow 5 \rightarrow 7 \rightarrow 2$ (or $0 \rightarrow 6 \rightarrow 3 \rightarrow 10$). This sequence is optimally required for the belief to build-up, and is specific to the chosen transition, observation, and reward functions.

6. REFERENCES

[1] A. R. Cassandra, "Exact and Approximate Algorithms



(a) (b)

Figure 3: Stick figure plot, step speeds, observations received, and the actions taken by the PoMDP-based hybrid controller Γ^{π} for (a) 65 step walk over a sample deterministic terrain with the up/down ramp slopes set at 10°, (b) a 125 step walk over a sample stochastically generated terrain. The slopes for the up and down ramp gradients are uniformly selected from $\{2^{\circ}, 4^{\circ}, 6^{\circ}, 8^{\circ}, 10^{\circ}\}$.

for Partially Observable Markov Decision Processes," Ph.D., Brown University, 1998.

[2] H. Dai and R. Tedrake, "Optimizing Robust Limit Cycles for Legged Locomotion on Unknown Terrain," in *International Conference on Robotics and Automation*, 2012.

[3] T. Erez, W. D. Smart, and S. Louis, "Bipedal Walking on Rough Terrain Using Manifold Control," in *Intelligent Robots and Systems*, 2007, pp. 1539 – 1544.

[4] A. Foka and P. Trahanias, "Real-time hierarchical POMDPs for autonomous robot navigation," *Robotics and Autonomous Systems*, no. January, 2007.

[5] G. Grudic, V. Kumar, and L. Ungar, "Using policy gradient reinforcement learning on autonomous robot controllers," *Robotics: Science and Systems*, pp. 406–411, 2003.

[6] M. Lahijanian, S. B. Andersson, and C. Belta, "A probabilistic approach for control of a stochastic system from LTL specifications," in *IEEE Conference on Decision and Control*, December 2009, pp. 2236–2241.

[7] I. R. Manchester, U. Mettin, F. Iida, and R. Tedrake, "Stable Dynamic Walking over Rough Terrain Theory and Experiment," in *Proceedings of the International Symposium on Robotics Research*, 2009, pp. 1–16.

[8] A. Y. Ng, "PEGASUS : A policy search method for large MDPs and POMDPs," in *Proceedings of the Sixteenth Conference on Uncertainty in Artificial Intelligence*, 2000.

[9] H.-W. Park, A. Ramezani, and J. W. Grizzle, "A Finite-state Machine for Accommodating Unexpected Large Ground Height Variations in Bipedal Robot Walking," *IEEE Transactions on Robotics*, to appear, 2013.

[10] H.-W. Park, K. Sreenath, J. W. Hurst, and J. W. Grizzle, "Identification of a Bipedal Robot with a Compliant Drivetrain," *IEEE Control Systems Magazine (CSM)*, vol. 31, no. 2, pp. 63–88, April 2011.

[11] J. Pineau and S. Thrun, "Hierarchical POMDP decomposition for a conversational robot," *ICML Workshop on Hierarchy and Memory*, 2001.

[12] G. Singh and A. Ng, "Quadruped robot obstacle negotiation via reinforcement learning," in *IEEE International Conference on Robotics and Automation*, 2006, pp. 3003–3010.

[13] T. Smith and R. Simmons, "Heuristic search value iteration for POMDPs," *Proceedings of the 20th conference on Uncertainty*, 2004.

[14] M. Spong and G. Bhatia, "Further results on control of the compass gait biped," *Proceedings 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 2, pp. 1933–1938, 2003.

[15] 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, Sep. 2010.