Probabilistic Stability in Legged Systems: Metastability and the Mean First Passage Time (MFPT) Stability Margin

Russ Tedrake, *Member, IEEE*
Katie Byl, *Student Member, IEEE*
Jerry Pratt, *Member, IEEE*

***Abstract*— The performance of many legged robots today is limited by trajectory generation processes which use overly conservative estimates of dynamic stability. We propose that the correct way to estimate the stability margin of a legged system is by modeling the dynamics as a stochastic process and studying its underlying statistics. In particular, we propose a new stability margin based on the mean first passage time (MFPT) to a fallen down state. We show that the famous static stability margin (SSM) and zero-moment point (ZMP) margin can be viewed as approximations of this statistic. We then present computational tools based on approximate optimal control for computing stronger approximations using a simulation, and for improving the approximation online on a real robot.**

Our investigation of the MFPT for simple walking models revealed an elegant simplification. For walking with disturbances, the overall dynamics of the state distributions are well described by a fast convergence to a metastable (long-living) limit cycle, and a small leakage rate to the fallen down state. The time constants of these eigenmodes provide scalar quantifiers that succinctly answer the question, "How stable is your robot?"

***Index Terms*— stability margins, legged robots, zero-moment point (ZMP), compass gait, approximate optimal control, first passage time, first visit time, mean time to failure**

I. Introduction

When we build models of legged systems, they are always deterministic. This makes sense - it is difficult enough to understand even these systems. But at the core of many control systems for legged robots is a concept rooted in stochastic processes - the stability margin. This line of research has evolved, with many proposed stability margins that capture different aspects of the walking problem [**?**], [**?**], [**?**], [**?**], [**?**], without ever having a deeper discussion about what types of stochasticity we expect in walking systems, and quantifying the expected results in the dynamics of our machines.

The consequences of this are that the existing stability margins are easy to compute, but relatively poor descriptions of the actual stability properties of our walking systems. For this reason, they are necessarily overly conservative, and they have become one of the major limitations to the performance of trajectory following legged systems.

This paper concerns itself primarily with the evaluation of stability for an existing closed-loop walking system. We also include a brief discussion of using this system to actually improve the control system.

R. Tedrake and K. Byl are with the Computer Science and Artificial Intelligence Lab at the Massachusetts Institute of Technology. (email: {russt,katiebyl}@mit.edu). J. Pratt is at the IHMC.

THE introduction will go something like this:

- major limitation of walking robots today is that they are constrained to overly conservative trajectories (less natural, less energy efficient, slower)
- define stability margins
- stability margins are used in the trajectory generation process[1]
- existing margins are far too simple, and don't capture the real source of stability (i.e., for the Honda robot, foot placement is more important than ankle torque)
- The entire idea of a stability *margin* implies stochasticity, but there are very few stochastic models of walking robot dynamics
- if we just keep track of the statistics of falling on our robot, then we can use that information to improve our stability margin online.
- present new theory and some computational tools
- also works for walking with foot roll and running
- helps answer the question, "Which stability margin is right for me?"
- walking enters the probabilistic revolution
- Mean Time to Failure instead of FPT?
- This work suggests that by running simulations of the robot, and by collecting statistics from trials run on the real robot, we can start with a stability margin estimate like ZMP then systematically improve that estimate using statistical inference.

II. Stability for Legged Locomotion

In most cases, the stability of steady-state locomotion can be characterized as the stability of a limit cycle. This analysis can be complicated, however, as even the simplest walking models can exhibit period-doubling bifurcations and chaotic attractors on their Poincaré maps, yet still walk without falling down[2]. On a real robot, imperfections in state estimation and small changes in the walking surface (and therefore in the closed-loop dynamics) make it very difficult to analyze the actual convergence to a particular limit cycle. Obstacle avoidance and other deviations from the steady-state trajectories can complicate the analysis even further.

In practice, many researchers testing legged machines use a more practical measure of stability - they simply say that the

robot is stable if and only if it doesn't fall down. Consider the deterministic closed-loop dynamics of a robot in the general form $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, t)$, where $\mathbf{x} \in \Re^N$ is the state vector. Now define a subset $F \subset \Re^N$ of the state-space as the states where the robot has clearly fallen down (e.g., a non-foot point of the robot is in contact with the ground). For the remainder of this paper, we will denote a configuration $\mathbf{x}'$ as *deterministically stable* if and only if $\forall t, \mathbf{x}(t) \notin F$ given $\mathbf{x}(0) = \mathbf{x}'$. In other words, $\mathbf{x}'$ is stable if and only if it is outside of the basin of attraction of F.

III. Stability Margins

Stability margins are an extension of stability concepts to stochastic systems. A legged robot is vulnerable to many sources of variability and uncertainty - sensor noise, unmodeled phenomena like actuator backlash, terrain variability and random disturbances - to name a few. A stability margin describes the tolerance of the stability properties of a system to these disturbances.

For linear systems, where the entire state space is either stable or unstable, two well-defined stability margins are the gain margin and the phase margin[3]. For nonlinear systems, it is possible for different states to have different stability properties. When planning a trajectory through this state space, it becomes important to understand how stable different states are relative to each other. There are tools for quantifying local stability in nonlinear systems, including linear analysis around a fixed point, contraction analysis[4], and the Lyapunov exponent[5], but these ideas are not immediately applicable to the definition of stability introduced in the last section. Instead, the legged robotics community, particularly in domain of walking robots, has embraced a more heuristic approach to quantifying stability. The static stability margin (SSM) and the zero-moment point (ZMP) margin are the two quantities most relevant to bipedal walking, but a number of other margins have been described and compared. See [6] and [7] for brief reviews.

A. SSM and ZMP

For walking on flat terrain, define the support polygon as a region $S \subset \Re^2$, which is the convex hull of the ground contact points. We use ∂S to denote the set of all boundary points of S.

One of the first stability margins introduced for walking was the static stability margin (SSM)[8]. A robot is considered statically stable if and only if the ground projection of the center of mass is inside the support polygon, because if all of the internal degrees of freedom of the robot are stationary and the ground contacts are the only sources of external force acting on the robot then the center of mass will also remain stationary. To handle the situation where the robot is moving (which is treated as a disturbance in this simple model) and other sources of uncertainty, we use the Euclidean distance from the ground projection of the center of mass to the closest edge of the support polygon as a heuristic stability margin:

$$\phi_{ssm} = \begin{cases} \min_{\mathbf{x}' \in \partial S} \|\mathbf{x}' - \mathbf{P}_{xy}\mathbf{x}_{com}\|_2 & \text{if } \mathbf{P}_{xy}\mathbf{x}_{com} \in S, \\ -\min_{\mathbf{x}' \in \partial S} \|\mathbf{x}' - \mathbf{P}_{xy}\mathbf{x}_{com}\|_2 & \text{otherwise,} \end{cases}$$

where $\mathbf{x}_{com} \in \Re^3$ is the center of mass and $\mathbf{P}_{xy}$ is the matrix $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}$ that projects $\mathbf{x}_{com}$ onto the xy-plane of the ground.

Especially for bipedal robots, maintaining static stability throughout the walking cycle results in extremely slow and unnatural movements. To consider the stability of a dynamic gait (where the ground projection of the center of mass leaves the support polygon), we use another important dynamic variable: the zero-moment point (ZMP)[9]. For our purposes, the ZMP is equivalent to the center of pressure, and is given by

$$\mathbf{x}_{zmp} = \frac{\int_{\mathbf{x}' \in S} F_z(\mathbf{x}')\mathbf{x}' d\mathbf{x}'}{\int_{\mathbf{x}' \in S} F_z(\mathbf{x}') d\mathbf{x}'},$$

where $F_z(\mathbf{x}')$ is the vertical component of the ground reaction force at $\mathbf{x}'$. Equivalently, we can write the ZMP in terms of the internal joints accelerations[1]. Having the ZMP inside the support polygon is the true dynamic condition that guarantees that the foot will not rotate. Consequently, the ZMP stability margin is defined as

$$\phi_{zmp} = \min_{\mathbf{x}' \in \partial S} \|\mathbf{x}' - \mathbf{x}_{zmp}\|_2$$

Notice that, by definition, $\mathbf{x}_{zmp}$ is always in S. A related dynamic quantity, which *can* leave the support polygon, is the foot rotation indicator (FRI)[10].

The SSM, ZMP, and most of the related stability margins for walking only make sense when the support polygon is larger than a point. For that reason, these margins cannot be used to describe the stability of walking robots with point feet in the single support phase. More importantly, they are not defined if and when a robot leaves the ground entirely, and therefore cannot be used for running robots.

B. Nonlinear systems interpretation

The SSM and the ZMP both represent approximations of the tolerance of a walking system to uncertainties. In the SSM, the assumption is that the robot will fall down if the ground projection of the center of mass leaves the support polygon. In other words, it assumes that the set of states that put the ground projection of the center of mass directly over the edge of the support polygon is the separatrix between the stable walking configurations and the basin of attraction of the fall down set, F. A similar assumption is made with the ZMP criterion, except the center of mass term is replaced with the ZMP. Both make a very coarse estimate of the basin of attraction of the fall down set, F, and then use the Euclidean distance of the current state from this basin of attraction as the stability margin. In both cases, the assumption is that the robot will fall down if the system begins to rotate around the support polygon. This is a reasonable assumption, because foot rotation reduces the actuation authority of the system, but humans and many robots are able to walk stability with less conservative gaits.

Even with a good approximation of the deterministic stability of the system, the *Euclidean* distance from a basin of attraction does not necessarily provide an accurate estimate of the stability margin. Consider a first-order stochastic system

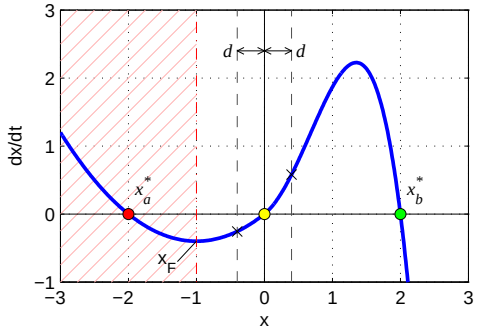

Fig. 1. Euclidean distance as a measure of relative stability.

with a single variable, $\dot{x} = f(x) + \xi$, where:

$$f(x) = \begin{cases} \frac{2}{5}x(x+2) & , \ x \leq 0 \\ \frac{1}{3}x(x-2)(x+3)\left(x+\frac{2}{5}\right) & , \ x > 0, \end{cases}$$

as plotted in Figure 1, and ξ is a random variable. When $\xi = 0$, this system has two stable fixed points (red at $x = -2$ and green at $x = +2$), and one unstable fixed point (yellow at $x = 0$) which defines the deterministic separatrix between the basins of attraction of the two stable fixed points. To motivate the problem, let us consider a neighborhood F at $x \leq x_F$ to be undesirable, analogous to the "fallen" states of a robot, and the other attractor (around x_b^*) as desirable (e.g. a stable walking cycle). The relative stability of each initial condition, e.g. the probability of visiting the region F, is plotted in Figure 2 plots for the case where ξ is drawn from a Gaussian distribution with mean zero and standard deviation σ_ξ. Notice the asymmetry around the basin of attraction (caused by the asymmetry in $f(x)$). The points $x = d$ and $x = -d$ have the same Euclidean distance from the original, deterministic separatrix at $x = 0$, but can have very different stability properties. Asymmetries like those used in $f(x)$ often arise in walking; as we will see in Section VIII, having the ankles of our robots away from the middle of the foot is one cause.

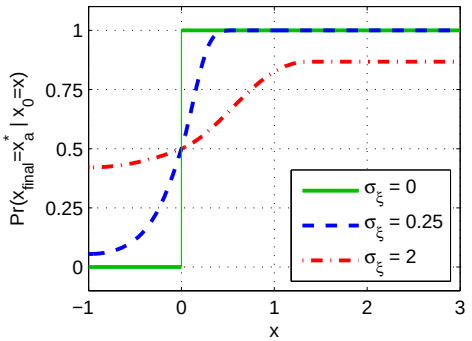

Fig. 2. Probability of survival, i.e. $x(t) \geq x_F \forall t$, given an unknown, constant offset ξ in velocity: $\dot{x} = f(x) + \xi$. Note the asymmetry about the separatrix, $x = 0$, as the noise in standard deviation in the uncertainty of ξ, σ_ξ, increases.

IV. Probabilistic Stability

In order to achieve more aggressive estimates of stability, we must be more specific about the types of uncertainty that we hope to accommodate in these systems, and we must consider the entire closed-loop dynamics of the system, including both the plant and the controller. For example, for a disturbance model that describes an impulsive force applied to our walking robot at the hip at time T, we can calculate the largest force required to push our closed-loop dynamics into the basin of attraction of F. The value of this quantity will vary over the state space of the robot, but it is a computation that is easy to perform if we have a simulation of the closed-loop dynamics. For impulsive disturbances in particular, if we have already computed the basins of attraction of the unperturbed system, then it is particularly easy to calculate this margin. Having computed these quantities, we could, for instance, restrict our trajectory generation process to select states that require more than 10 Joules to destabilize the system.

Real robots are subject to many forms of disturbances; we may wish to describe the overall robustness of the system, simultaneously considering many disturbance models. In this situation we can formulate a complete stochastic dynamic model of the robot and consider the probability and statistics of falling over, independent of the specific event (or set of events) that caused the problem. This approach, which we refer to as *probabilistic stability* analysis, has the additional advantage that it can be assessed experimentally on the real robot, using the actual stochasticity of the environment without any explicit models of the dynamics nor the disturbances.

A. The first passage time

Let us consider again the system in Figure 1, this time subjected to a time-varying disturbance, $\xi(t) = N(0, 2)$, where $N(\mu, \sigma)$ represents a normal distribution with mean z and standard deviation σ, and $\langle \xi(t), \xi(t') \rangle = 0$ for $t \neq t'$. The dynamics of this system are more difficult to quantify; as $t \to \infty$, $x(t)$ will visit every state with probability 1. For this system, the probability of entering region F as $t \to \infty$ is a poor metric of relative stability, because it is one everywhere.

Note: Need to be careful here with continuous distributions. Use a Wiener process?

Instead, to quantify the relative dynamical "distance" from each state x to the fall down region, we can describe the amount of time that it takes the dynamical system to visit F, for the first time. This time, well known as the *first passage time* (also first visit time, time-to-failure), is a random variable. The mean first passage time is a statistic that has proven useful in a number of fields.

This is a real problem, for both bounded and unbounded noise. This would be true for a real robot, too - eventually the robot will get hit by a car, or wind up in a tornado, or walk off the edge of a cliff. This is a grim, but realistic, observation about our walking machines: if we run them for long enough, they *will* eventually fall down.

V. Model-based Approximation Algorithms

In this section we will present some tools for calculating the MFPT using a stochastic dynamic model of the system. All of these algorithms are based on the observation that computing the MFPT can be reduced to policy evaluation

(computing the value function, or cost-to-go function) in an optimal control problem which penalizes falling down. Note that policy evaluation is substantially easier that solving for the optimal policy - and can be very efficient.

To reduce the MFPT calculation into an optimal control problem, we need to slightly modify the dynamics of our system so that F is an absorbing region. In our 1d example, this would mean:

$$\dot{x}(t) = \begin{cases} 0 & x \in F \\ f(x) + \xi & \text{otherwise.} \end{cases}$$

In practice this can be accomplished by simply stopping the trial when your robot falls to the floor. Now the MFPT can be calculated as

$$\langle T_F(x) \rangle = \langle \int_0^\infty g(x) dt \rangle,$$

where

$$g(x) = \begin{cases} 0 & x \in F \\ 1 & \text{otherwise.} \end{cases}$$

A. Monte Carlo

Monte Carlo simulations one obvious option estimating the FPT statistics. The disadvantage of running multiple trials-to-failure is that such simulations will generally be computationally intensive. For systems where the model has many states (e.g. ≥ 5) but the initial conditions we wish to simulate occupy only a relatively small manifolds in this state-space (e.g. walking with joints at the knees and/or ankles), running trials from these few initial conditions may provide a relatively efficient and accurate method of estimating the FPT statistics.

B. Direct Computation using Markov Chains

When practical, discretizing the dynamics in both time and state space to create a Markov chain is a powerful tool in both calculating the FPT statistics and in optimizing control policies to maximize MFPT.

VI. Old algorithms section

A. Stochastic Poincaré maps and Markov chains

In general, we can estimate the "cumulative value" of a particular cost function over time given 1) a particular initial condition, 2) a (stochastic) noise definition and 3) a control policy; this can be done by performing a direct calculate using a step-to-step transition matrix, f. The transition matrix encodes the probabilistic transitions from some set of particular (discrete) states to the future state(s) that will be visiting in a given "step" (in time, or literally as a step taken by a walker). The continuous system dynamics are first "meshed" into a set of discrete sets of states. These discrete states and their transition rules comprise a Markov chain approximation of the exact system dynamics: the combination of system dynamics, noise inputs, and control policy for the model determine a (generally stochastic) set of probabilities of going from the current state to each of the other discretized states defined. The transition matrix is created such that $f(i, j)$ gives the probability of starting in state i and transitioning to state j; each row of f must therefore add to 1.

To evaluate the performance of a particular system (e.g. given some definition of system parameters, control law and/or noise), one clean approach would be to define a step-to-step cost and then integrate this cost over all possible (probabilistic) eventualities from any initial condition of interest. We can use the step-to-step transition matrix to perform this calculate efficiently (using the Bellman equation).

$$V_i = \gamma \sum_{j=1}^{N} f_{ij} V_j + r_i$$

where V_i is the cumulative value we expect to receive, given we are currently in state i, r_i is the one-step "reward" (or penalty) for implementing our control policy at state i (which incorporates all possible eventualities, if there is noise), γ is a per-step discount factor for the step-to-step value function accumulation and f is the transition matrix defined above.

Calculating the mean first passage time (MFPT) is a particular case of the Bellman equation above, where r_i is the "value" of taking *any* step from a non-failed state. In other words, $r_i = 1$ for all $i \notin F$, where F is the set of all "failed" (e.g. falling) states, since (by definition) exactly one step is taken in transition from a particular post-collision state to next step if the initial state is not a failed state. The discount factor is $\gamma = 1$, because we want to calculate the expected total number of steps taken.

We can select a desired one-step cost (such as the Euclidean ZMP margin, or a combination of error for a desired trajectory and weighted control effort, etc.) and a discount factor, γ. Then, the vector giving the cumulative value function from each discrete initial condition (given our defined system dynamics, noise and control policy) is:

$$\mathbf{V} = (\mathbf{I} - \gamma \mathbf{f})^{-1} \mathbf{r}$$

B. Computing the First Passage Time distribution

As we will see in a few sections, computing the mean first passage time is considerably easier than computing the entire first passage distribution, and more amenable to online computation on a real robot. This algorithm is included because the FPT distribution has interesting properties on it's own, and because the mean FPT might not be the best summary of the distribution for all problems.

The first passage time for a particular initial condition can be calculated by Monté Carlo methods, but this is extremely computationally intensive. If our goal is to compute the distribution over the entire state space of the robot, it becomes intractable. Instead, we can take advantage of how this distribution changes over the state space of the robot to create a surprisingly efficient algorithm.

Assume that F is an absorbing set (or equivalently for us, all states in F are absorbing states), and discretize the process in time. Then the first passage time is a random variable described by

$$p_{\mathbf{x}}(n) = \text{Pr}[\text{FPT}(\mathbf{x}) = n] = \text{Pr}[\mathbf{X}(n) \in F, \mathbf{X}(n-1) \notin F | \mathbf{X}(0) = \mathbf{x}],$$
$$P_{\mathbf{x}}(n) = \text{Pr}[\text{FPT}(\mathbf{x}) \leq n] = \text{Pr}[\mathbf{X}(n) \in F | \mathbf{X}(0) = \mathbf{x}]$$

The cumulative probability distribution, $P_{\mathbf{x}}(n)$, can be interpreted as the probability that the robot has fallen down at time step n. Working with $P_{\mathbf{x}}(n)$ has a few advantages. Taking advantage of the Markov property, $Pr[\mathbf{X}(n) = \mathbf{x}(n)|\mathbf{x}(0), \mathbf{x}(1), ..., \mathbf{x}(n-1)] = Pr[\mathbf{X}(n) = \mathbf{x}(n)|\mathbf{x}(n-1)]$, we can write the distribution recursively:

$$\begin{aligned} P_{\mathbf{x}}(n) =& \text{Pr}[\mathbf{X}(n) \in F | \mathbf{X}(0) = \mathbf{x}] \\ =& \sum_{y \in F} \text{Pr}[\mathbf{X}(n) = y | \mathbf{X}(0) = \mathbf{x}] \\ =& \sum_{\mathbf{x}'} \text{Pr}[\mathbf{X}(1) = \mathbf{x}' | \mathbf{X}(0) = \mathbf{x}] \sum_{y \in F} \text{Pr}[\mathbf{X}(n) = y | \mathbf{X}(1) = \mathbf{x}'] \\ =& \sum_{\mathbf{x}'} f(\mathbf{x}', \mathbf{x}) P_{\mathbf{x}'}(n-1). \end{aligned}$$

In words, the probability of falling down from state x in n steps can be computed by taking one step, then using the known probability of falling down in $n-1$ steps from the new state. This is a powerful observation; it means that we can improve our stability estimates online with bootstrapping. To finish the recursion, observe that if $\mathbf{x} \in F$, then $\forall n, P_{\mathbf{x}}(n) = 1$.

$$\begin{aligned} P_{\mathbf{x}}(0) =& \begin{cases} 1 & \text{if } \mathbf{x} \in F, \\ 0 & \text{otherwise.} \end{cases} \\ P_{\mathbf{x}}(n) =& \sum_{\mathbf{x}' \notin F} f(\mathbf{x}', \mathbf{x}) P_{\mathbf{x}'}(n-1) + \sum_{\mathbf{x}' \in F} f(\mathbf{x}', \mathbf{x}) \\ =& \sum_{\mathbf{x}' \notin F} f(\mathbf{x}', \mathbf{x}) P_{\mathbf{x}'}(n-1) + \text{Pr}[\mathbf{x}' \in F | \mathbf{x}]. \end{aligned}$$

Finally, to back out the probability density, observe that

$$\begin{aligned} p_{\mathbf{x}}(0) =& P_{\mathbf{x}}(0) \\ p_{\mathbf{x}}(n) =& P_{\mathbf{x}}(n) - P_{\mathbf{x}}(n-1). \end{aligned}$$

This recursion can be implemented efficiently on a computer. Describe the tricks like discretizing, but interpolating the state space, and pre-computing all of the transition probabilities.

For low dimensional systems, this recursion can be estimated effectively using the following particle filter[12] algorithm: *come up with algorithm and put it here*.

C. FPT stability margins

Once we have computed the entire FPT distribution for all states, there are a number of candidates for the actual stability margin:

- $\underset{t}{\text{argmin}}\ [P_{\mathbf{x}}(t) > \alpha]$ - The time at which the probability of falling down is greater than α, $0 \leq \alpha \leq 1$.
- $E[p_{\mathbf{x}}(t)]$ - Mean first passage time.
- $P_{\mathbf{x}}(t_\alpha)$ - Probability of falling down by time t_α.
- $\underset{t}{\text{argmax}}\ [p_{\mathbf{x}}(t)]$ - Maximum-likelihood (ML) time to fall down.

D. Mean First Passage Time

The mean first passage time can be calculated directly (i.e. without Monte Carlo simulations or iterative, asymptotic approximation) if the dynamics can be represented as a Markov process with discrete states. In the case of walking, a natural choice is a set of states after each leg collision (step). The accuracy of this calculation can theoretically be made arbitrarily exact through appropriately refined meshing and interpolation to represent state transitions. The practical limitations to accuracy will be speed and (more significantly) memory in representing a sufficiently large number of states to capturing states of interest.

The resulting Markov system is described by its transition matrix, $\mathbf{f}$:

$$f_{ij} = Pr\left[X(n) = j | X(n-1) = i\right]$$

We can define the variable $s(n)$ to the count the number of Markov steps taken which *did not originate* from a failed state in the course of arriving (through some particular, as yet unspecified path) from some initial state $X(0)$ to a state $X(n)$. This count is defined as $s(0) = 0$ in the initial state. Then,

$$s(n) = \begin{cases} s(n-1) & \text{if } X(n-1) \in F \\ 1 + s(n-1) & \text{otherwise} \end{cases}$$

For most real-world systems, the transition matrix inherently allows for a non-zero chance of eventually entering the absorbing failure state as $n \to \infty$. Therefore, a failure *will* eventually occur, and the finite number of steps executed before entering the failed state will be $s_f = s(n \to \infty)$.

Let m_i be the MFPT, given we begin in state $X(0) = i$. This is the expected count of steps that will be taken before arriving in an absorbing (failed) state:

$$m_i = E\left[s_f | X(0) = i\right]$$

We can use the definition of the counting variable s and the total expectation theorem to define m_i recursively:

$$m_i = \begin{cases} 0 & \text{if } i \in F \\ 1 + \sum_{j=1}^{N} f_{ij} m_j & \text{otherwise} \end{cases}$$

where N is the total number of states. We can write the equations for the unknown *non-absorbing states* in matrix form. To avoid singularities in the calculation, however, we will use sub-matrices $\bar{\mathbf{f}}'$ and $\bar{\mathbf{m}}'$, which contain *only* the non-absorbing states for which we need to solve the MFPT. (Recall we already know that $m_i = 0$ for $i \in F$; this along with the sub-matrix calculation then provide the full solution for $m_i \forall i$.)

$$\bar{\mathbf{m}}' = \bar{\mathbf{f}}' \bar{\mathbf{m}}' + \bar{\mathbf{1}}$$

where $\bar{\mathbf{1}}$ is a column vector containing all ones. Solving for the vector of mean first passage times for the non-absorbing states, we obtain:

$$\bar{\mathbf{m}}' = (\mathbf{I} - \bar{\mathbf{f}}')^{-1} (\bar{\mathbf{1}})$$

We could also choose to define the MFPT such that the last step (arriving into the failed state for the first time) is not

counted as a step, in which case we can simply subtract one from the resulting estimate for MFPT obtained through the calculation above.

The rest of this section is an older version of this section on deriving MFPT... For cases with finite state spaces, calculating the mean first passage time is simple. In this case, we can write the probability distribution $\Pr[\mathbf{X}(n) = \mathbf{x}]$ as a vector, $\bar{\mathbf{x}}(n)$, and the transition probabilities as a matrix, $\bar{\mathbf{f}}$. Then we have

$$\bar{\mathbf{x}}(n) = \bar{\mathbf{f}}\bar{\mathbf{x}}(n-1) = \bar{\mathbf{f}}^n\bar{\mathbf{x}}(0).$$

The first passage time distribution from state i (represented by the ith element of the vector $\bar{\mathbf{x}}$) is:

$$p_i(0) = \begin{cases} 1 & \text{if } i \in F \\ 0 & \text{otherwise,} \end{cases}$$

$$p_i(n) = \sum_{j\in F} \left[\bar{x}_j(n) - \bar{x}_j(n-1)\right] = \sum_{j\in F} \left[\bar{\mathbf{f}}^{n-1}(\bar{\mathbf{f}} - \mathbf{I})\delta_i\right]_j,$$

where δ_i is the vector with a one in component i and zeros everywhere else. Finally, the mean first passage time is

$$\begin{aligned} E[\text{FPT}(i)] &= \sum_{n=0}^{\infty} n p_i(n) = \sum_{j\in F}\left[\left(\sum_{n=0}^{\infty} n\bar{\mathbf{f}}^{n-1}\right)(\bar{\mathbf{f}} - \mathbf{I})\delta_i\right]_j \\ &= \sum_{j\in F}\left[(\mathbf{I} - \bar{\mathbf{f}})^{-2}(\bar{\mathbf{f}} - \mathbf{I})\delta_i\right]_j \end{aligned}$$

Can I put this into vector form?

Introduce algorithm FPT(λ).

Another derivation

Note: I just used random variable names in here. We should decide on a convention.

The first passage time, ϕ, for a discrete time system on a given trial can be written as:

$$\phi = \sum_{i=0}^{N} 1,$$

where N is the step on which the robot fell. Equivalently, this can be written as

$$\phi = \sum_{i=0}^{\infty} g(x,u), \qquad g(x,u) = \begin{cases} 0 & x \in F \\ 1 & \text{otherwise} \end{cases}.$$

For systems which also have discrete states (Markov chains), we can write g as a vector with elements $\{0, 1\}$. The mean first passage time, $\mathbf{m}$, given I've started in state i is time is

$$m_i = E\left[\sum_{i=0}^{\infty} g(x,u)\right],\ x(0) = s_i.$$

This looks like an infinite-horizon, undiscounted value function, and we can write it recursively as

$$\mathbf{m} = \mathbf{g} + \bar{\mathbf{f}}\mathbf{m}.$$

Therefore, the mean first passage time is simply

$$\mathbf{m} = (\mathbf{I} - \mathbf{f})^{-1}\mathbf{g}.$$

E. Mixing Time

If the MFPT from some subset of states is large enough, we can treat the process of walking as a metastable state: initial conditions are essentially forgotten after an appropriately long mixing time, and the inverse of the MFPT defines a leakage or "escape" rate at which failures occur on average: $r = 1/MFPT$

Using the concept of metastability implies a separation of time scales. We can test the validity of this assumption by comparing the second- and third-largest eigenvalues of the (transpose of the) transition matrix, $\mathbf{f}$. The largest eigenvalue will always be unity, $\lambda_1 = 1$. For a system with a single "metastable" state second-largest eigenvalue will be related to the escape rate, r, as $\lambda_2 = 1 - r$. In words, it is the expected probability of not failing on any given step, once initial conditions have been forgotten. The remaining eigenvalues give rates at which initial conditions will be forgotten. Any initial condition can be decomposed into a weighted average of the eigenvectors of the transpose of f, and the logarithmic rate at which each vector component "dies off" is related to the corresponding eigenvalue. (For instance, an eigenvector with $\lambda = .25$ decays to 1/4 of its previous contribution at each step...) The eigenvector corresponding to λ_1 sums to one (i.e. the total probability of existing), while each other eigenvector sums to zero. This maintains a total probability of one as initial conditions "die away".

To estimate the rate of mixing, we can look at the eigenvalue with the third-largest magnitude. The time constant for mixing (at which point the initial condition contribution has decayed to 1/e of its initial value) is $\tau_3 = 1/log(1/\lambda_3)$, while that of the metastable process is $\tau_2 = 1/log(1/\lambda_2)$. To model the system as metastable, we require $\tau_2 >> \tau_3$, which can also be written as the requirement:

$$log(1/\lambda_3) >> log(1/\lambda_2)$$

or, using the approximation $log(1/(1-\delta)) \approx log(1+\delta) \approx \delta$,

$$(1 - \lambda_3) >> (1 - \lambda_2)$$

so that the rate of mixing $(1 - \lambda_3)$ is significantly faster than the leakage rate $(1 - \lambda_2)$.

(Note that we can solve for the quasi-stationary distribution by elimination the failure state from f; renormalizing the each row to add to one; and then solving for the eigenvector of the *transpose* of this submatrix which corresponds to an eigenvalue of one.)

F. Simple 1D dynamic system example

We will now return to the simple first-order dynamic system described in Section III-B to illustrate what the FPT statistics tell us about system stability. Figure 3 shows both the mean (solid line) and median (dashed) of the first passage time distribution. Near the desirable attractor at $x = 2$, the process can be approximated as metastable state with constant leakage rate, r. Thus, the failure statistics near $x = 2$ are described by an exponential probability distribution of failure. For an exponential process, the mean is $MFPT = 1/r$, and the median is $ln(2)/r$). The value ratio of the median to the mean

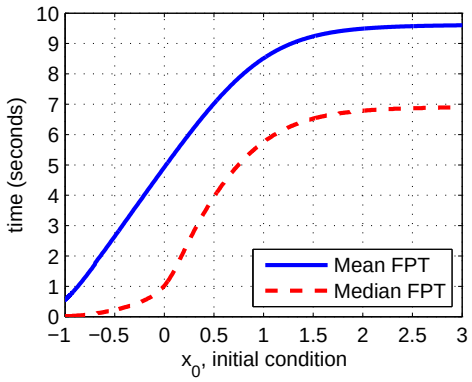

Fig. 3. Mean FPT and *median* FPT given a time-varying disturbance in velocity: $\dot{x} = f(x) + \xi(t)$

at $x = 2$ is close to what you would expect: $6.8/9.5$, or $1.03ln(2)$. By contrast, the dynamics near $x = 0$ are more complex. They involve a substantial risk of early death, but after initial conditions are forgotten, the distribution of states for the survivors approaches the metastable basin shown in Figure 4. To picture the risk, imagine the following scenario: you are presented with two fuses which look identical. However, one is a slow-burning fuse that takes 9 minutes to burn, while the fast one fails after 1 minute. If you tend to pick the quick fuse about (but slightly more often than) 50% of the time, the mean burning time will be about 5 minutes, but if your chance of selecting the faster one is just slightly over 50/50, the median will be only one minute. This is analogous to what is happening at the separatrix, $x = 0$. The mean is about 5 seconds, while the median is about one second. The process involves a high "infant mortality" which is not captured adequately by looking solely at the mean FPT for that initial condition.

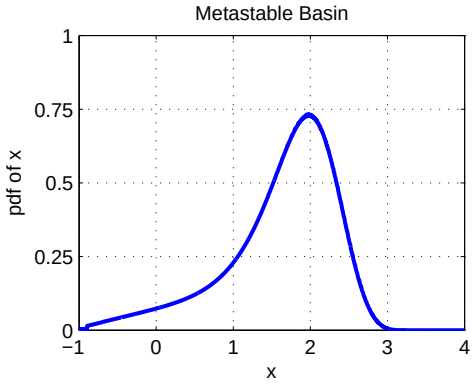

Fig. 4. Quasi-static probability distribution function for the value of x given a time-varying disturbance in velocity: $\dot{x} = f(x) + \xi(t)$

A more direct way of identifying the magnitude of the short-term risk is to calculate the probability of surviving until at least some particular time threshold, T, has passed: $Pr(x(T) \in F|x(0))$. Figure 5 shows curves of this statistic for 3 different values of T. After one second, for instance, there is only about a 25% chance of survival for $x(0) = -0.5$ and about a 50% chance if we began at $x(0) = 0$. Correspondingly, the median in Figure **??** goes through one second at $x(0) = 0$. This short-term risk is analogous to the SSM and ZMP margin, which attempt to predict the risk of falling in the immediate future.

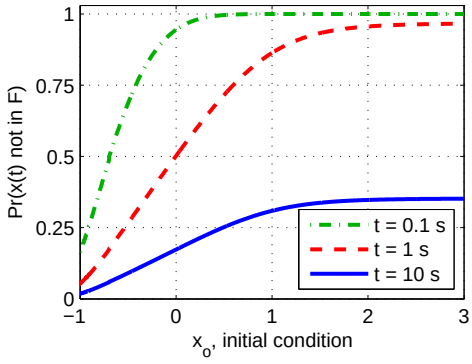

Fig. 5. Probability of survival after a given time has elapsed, as a function of initial condition, x_0, $Pr(x(T) \in F|x(0))$.

VII. The Compass Gait on Rough Terrain

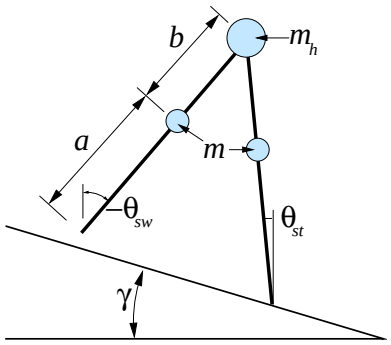

Fig. 6. The compass gait walker.

One of the simplest models of walking is the 2D passive compass gait[**?**], depicted in Figure 6. It consists of two massless legs, with a frictionless pivot between them and resembles a drafting compass. Our compass gait model includes three point masses: one on each leg and one of the pivot or "hip" joint where the legs meet. Impacts during walking are modeled as inelastic, and we assume the (massless) toe will retract automatically during the swing phase to avoid foot-scuffing. There is no actuation in the walker; stable limit cycles exist where the energy lost at impact exactly balances the change in potential energy as the device walks down an incline.

Given appropriate parameter values and initial conditions, a 2D compass gait walker will walk down a particular constant, gradual slope, converging deterministically to a stable limit cycle. We now consider the same system with added noise to the terrain. For most non-trivial noise descriptions [1], any such walker must eventually fall. However, if particular initial

[1] COMMENT: e.g.Gaussian, but also certainly many bounded noise models as well.

conditions exist for one walker which result in a MFPT of thousands or millions of steps, this walker is clearly more successful than one for which no initial condition results in better than a dozen or so expected steps on average. We are also interested in the size of the stochastic basin of attraction for such a walker. Even if one particular initial condition results in a large MFPT, we would generally like the walker to perform well given reasonable perturbations in initial condition.

A. Deterministic basin of attraction

For the deterministic compass gait on even terrain, we can numerically model the dynamics and calculate whether the walker will converge to a stable limit cycle or eventually fall. This defines two clear regions in state space: the basin of attraction of stable walking ($MFPT = \infty$), and the basin of attraction to failure. Figure 7 shows a 2D slice of 3-dimensional basin of attraction of stable walking for a walker where $a = .6$, $b = .4$, and $m_h = 2m$, traveling down a 4 degree slope. The basin is plotted for post-collision states, so that the constant slope at impact forces a linear relationship between the stance angle, θ_{st}, and swing angle, θ_{sw} at each impact. This leaves the inter-leg angle (33 degrees for the slice shown) and the two angular velocities as independent degrees of freedom.

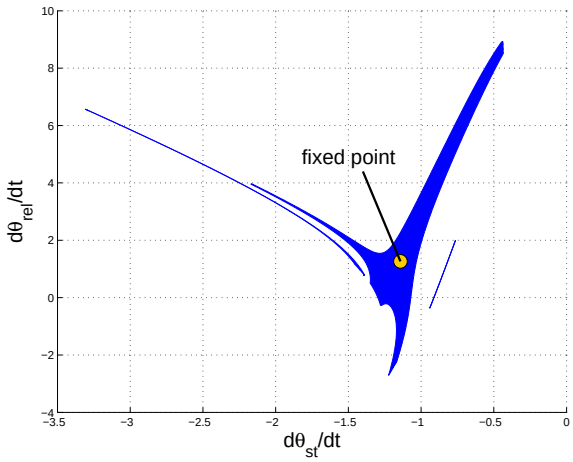

Fig. 7. A 2D slice of the deterministic basin of attraction for a passive compass gait walker on an even slope. This slice corresponds to an interleg angle of about 33°, which is the interleg angle of the stable limit cycle. The fixed point of the limit cycle is located at the center of the dot shown.

B. Stochastic basin of attraction

The deterministic basin in Figure 7 gives only a binary classification of stability (on whether it will converge to the limit cycle). We can attempt to judge the robustness of a system to stochastic noise by measuring the smallest distance in each degree of freedom to the separatrix between the stable and unstable regions, but it does not allow us to analyze the stochastic stability to particular noise distribution.

We propose mapping the MFPT over the state space provides a much richer description of stability. Figure 8 shows a 3D plot giving MFPT as a function of state over the same 2D slice in state space depicted in Figure 7. Here, the terrain is modeled as having a Gaussian distribution in *slope* for each step. For Figure 8, $\mu = 4$ deg, $\sigma = 0.5$ deg.

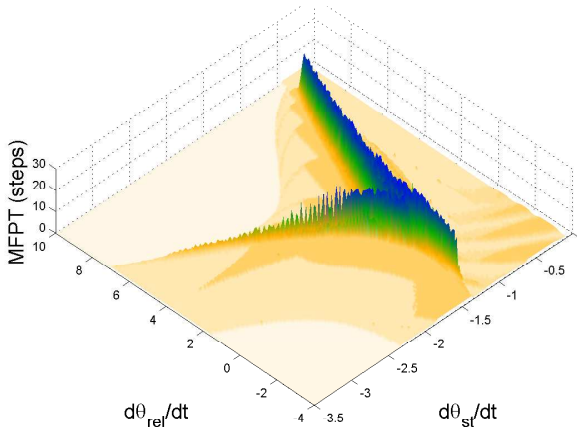

Fig. 8. The stochastic basin of attraction, showing MFPT for a passive compass gait walker on rough terrain.

Data were obtained by running Monte Carlo simulations[2]. We have run Monte Carlo simulations for particular initial conditions (rather than over a mesh of initial conditions) which show significant and unexpected differences in stability between different mechanical walker designs. Approximating the MFPT clearly provides a useful tool for optimizing both mechanical system dynamics and control policies. For instance, modifying the walker such that $a = .7$, $b = .3$, and $m_h = 0.3m$, the MFPT on the same terrain is close to 3 million steps (compared with 20 in the previous walker).

VIII. Inverted Pendulum With Base

The simplest model for comparing the FPT to the ZMP is the inverted pendulum with a base which is connected to the ground via two points. I assume that friction is large enough that the base never slips on the floor.

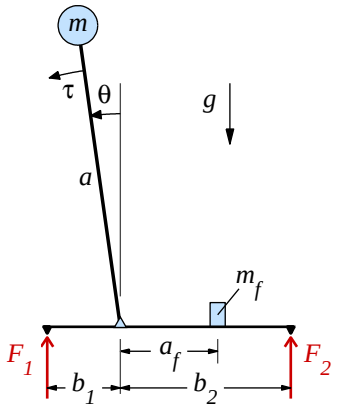

Fig. 9. Inverted Pendulum With Base

Given the state $(\theta, \dot{\theta})$ and the controller output (τ), we can compute the ground reaction forces for the regime when both

[2]Comment: 25,000 points for the 2D slice in Fig. 8.

feet are on the ground and $F_1, F_2 \geq 0$:

$$F_y = mg - \frac{\tau}{a}\sin\theta - mg\sin^2\theta - ma\dot{\theta}^2\cos\theta$$
$$F_1 = \frac{-\tau + b_2 F_y + (b_1 + a_f)m_f g}{b_1 + b_2}$$
$$F_2 = \frac{\tau + b_1 F_y + (b_2 - a_f)m_f g}{b_1 + b_2}$$

The system uses torque-limited PD control to regulate the pendulum about the angle which locates the center of mass directly about the center of the foot (maximizing SSM for a noiseless system). The noise input consists of impulses (step changes in pendulum velocity). At each time step, there is a small but finite chance of receiving an impulse, whose magnitude is selected from a Gaussian distribution. The dynamics are simulated over a smaller time step, using the finite difference method, and the Gaussian noise distribution is approximated by a finite set of discrete possible values.

A. The ZMP Margin: ϕ_{zmp}

The ZMP location, referenced from the back of the foot, is simply

$$x_{zmp} = \frac{(b_1 + b_2)F_1}{F_1 + F_2}.$$

The ZMP stability margin is:

$$\phi_{zmp} = |\min[x_{zmp}, (b_1 + b_2) - x_{zmp}]|.$$

B. The mean first passage time: ϕ_{fpt}

To calculate the MFPT for this system, we approximated the dynamics as a discrete Markov chain and iterative calculated the approximate transition matrix and then remeshed to capture regions where the gradient in MFPT is largest. This involved discretizing the state space over θ and $\dot{\theta}$, selecting a discrete time step for transition, discretizing the noise as a finite set of possible values at each step, and then simulating dynamics for the necessary time step for each discrete noise value. Each new state was approximated by barycentric weighting to an appropriately enclosing set of $N+1$ nodes in the particular 2^N-node rectangular mesh element which surrounds this desired new states. States which do not fall within any existing mesh elements are considered part of the absorbing failed (fallen) state.

C. How does ϕ_{zmp} compare to ϕ_{fpt}?

Figures **??** and**??** compare ZMP and MFPT for a symmetric pendulum ($b1 = b2$) and an asymmetric one. Comparing the two figures, we note: (1) the plots are is predictably asymmetric for the latter and (2) the MFPT of the former system is approximately 100 times longer that than of the asymmetric one. The ZMP captures the relative short-term stability of possible states in a given system; however, it gives no information about the expected performance of one system versus the other. The MFPT provides a more meaningful metric for optimizing system performance, since it captures the overall behavior of the resulting dynamic system.

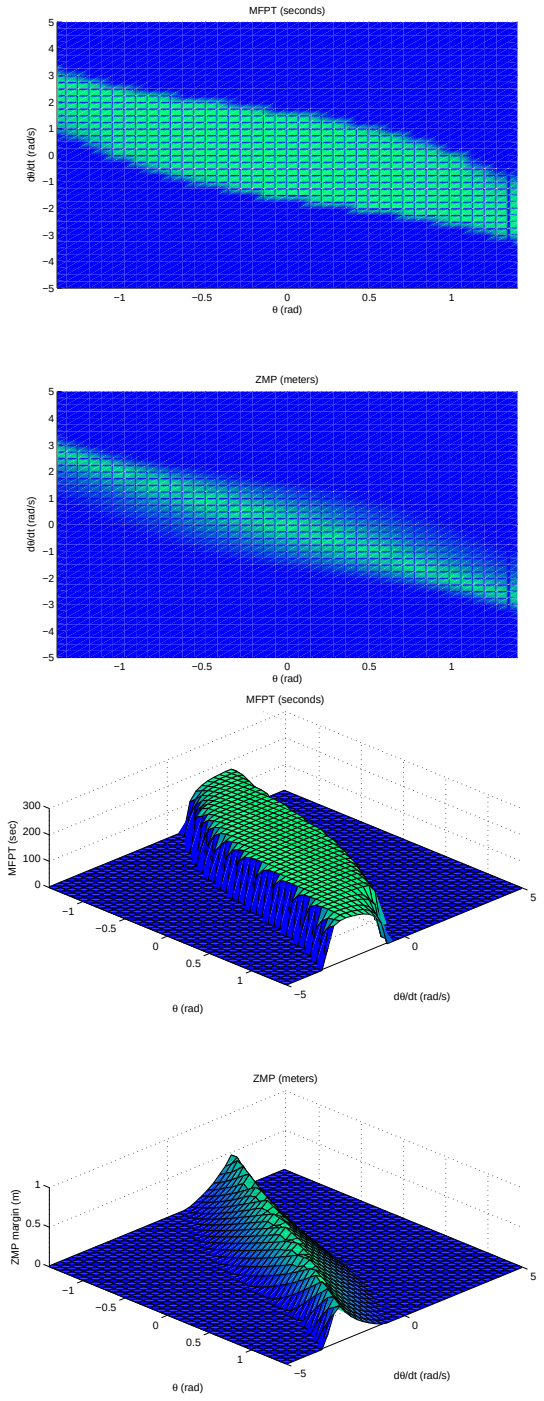

Fig. 10. MFPT (top) versus ZMP, plotted over the state space for the inverted pendulum with base. (Note the top two subplots give an overhead view, while the bottommost two show the same data from a 3D perspective.) The MFPT plot has an essentially flat top: pendulums which begin within this region tend to converge rapidly to a subset of this large region. Because initial conditions are quickly forgotten, there is little difference in MFPT within this larger region of "likely recovery" back toward equilibrium.

D. Improving ϕ_{zmp}

Most walking systems involve asymmetries in the shape of the support polygon, actuator authority, expected noise disturbance and/or sensor information (among other things). We would therefore expect that we can find stability metrics that

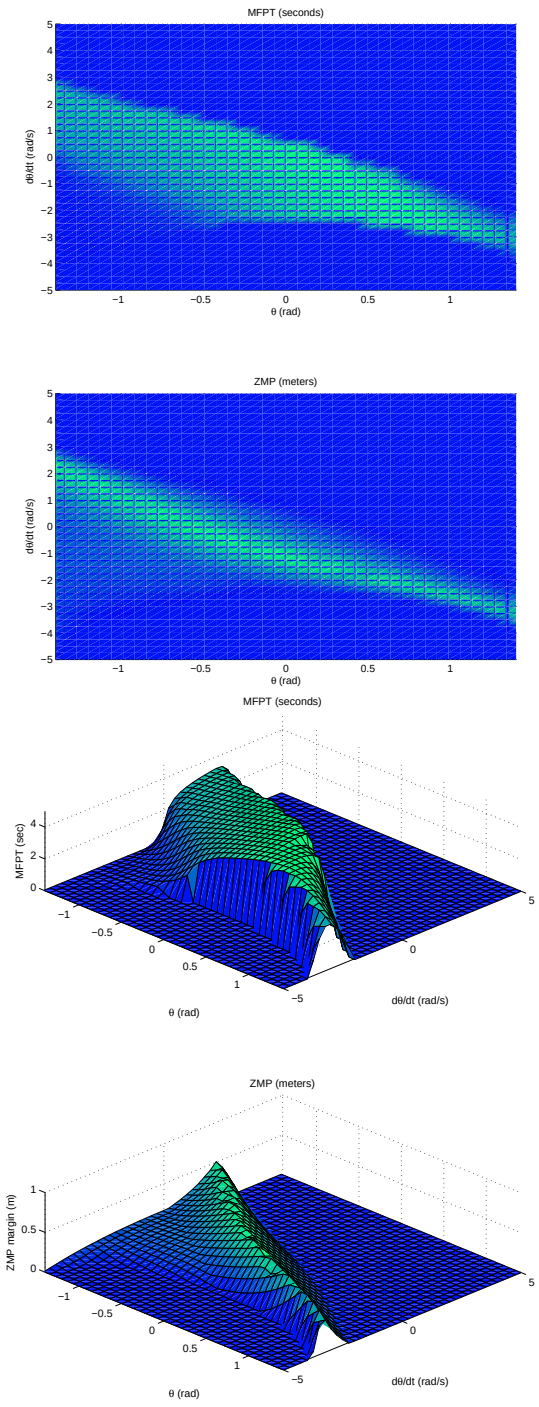

Fig. 11. MFPT (top) versus ZMP for an asymmetric position of pendulum pivot. Note that the system-wide performance of this system is much worse than that of the symmetric system corresponding to Figure **??**: the MFPT of the symmetric system is about 100 that of the asymmetric system shown here.

are better-correlated with robust walking than the ZMP margin. One candidate stochastic stability metric is the probability of having failed over some short, prescribed time interval in the future: $Pr(x(T) \in F|x(0))$, for some time scale, T. We can often assume well-designed walkers will converge back to metastable walking gaits over longer time scales, and then valuing our immediate ability to avoid failure heavily makes sense.

IX. The Planar One-Leg Hopper

The planar one-leg hopper shown in Figure 12 is based on the Raibert hopper[**?**], [**?**]. It uses an adjustable spring stiffness and torque actuation between the leg and body to regulate hopping height, body attitude and forward velocity. The state space describing this 2D model involve 10 states variables: x and y positions of the foot, the adjustable undeflected length of the leg spring, the angles of the leg and the body with respect to global coordinate, and the time derivatives of each of these fives states.

Because of the size of the state space, solving for the MFPT via the meshing strategy employ in the inverted pendulum model in section VIII is probably impractical[3], and Monte Carlo simulations of the sort performed in section 6 would be computationally intensive.

We tentatively propose using a hybrid technique to approximate MFPT for higher-order systems such as this, involving two basic modeling steps. First, the dynamics would be simulated using Monte Carlo methods to identify particular states which are most often visited. The quasi-stationary metastable distribution can be approximated by selecting a finite number of unmeshed (scattered) nodes in state space and weighting each approximate the observed probability density function observed from simulations. Next, estimate the single-step probability of transitioning to failure from each of the discrete states selected. This involves selecting a discrete time-step for the transition matrix, identifying the magnitude of the disturbance(s) that result in inevitable failure, and integrating to find the probability that a disturbance of such magnitude would happen during the time step chosen for simulation. The system-wide leakage rate (inverse MFPT) can be approximated as the sum of the probability of each discrete state being visit during observed trial runs times the probability of immediate failure in the chosen time step, given we are in this particular state. We are currently developing tools to implement this method and to estimate its accuracy.

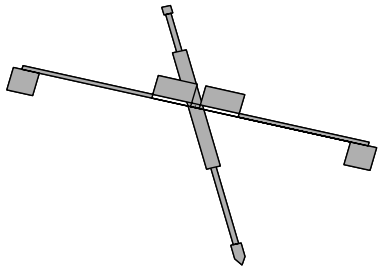

Fig. 12. The planar one-leg hopper

X. Estimating First Passage Time Statistics Online

- Formulate as maximum time optimal control problem, $C = -\int dt$. Value function is stability margin.

[3]Comment: Our attempts at iteratively remeshing even a reduced-order version of the model result in transition matrices which become too large to compute in MATLAB before any metastable modes begin to emerge.

- Value iteration for simple systems.
- Online value learning algorithms for real systems
- ZMP as an initial guess.... only do better
- Reduced-order approximations

Monte carlo. Particle filters.

Policy evaluation for the maximum-time problem. If we discretize the problem in time, then the cost-to-go function, or value function, for following policy π is given by:

$$V^\pi(\mathbf{x}) = E\{-1dt + V^\pi(\mathbf{x}')\},$$

where $\mathbf{x}'$ is the next state visited after $\mathbf{x}$ following π. Given a noise model, we can define this transition probability as

$$f(\mathbf{x}', \mathbf{x}) = Pr\{\mathbf{X}' = \mathbf{x}' | \mathbf{X} = \mathbf{x}\}.$$

If we also discretize the state space, then the value function can be written as

$$V^\pi(\mathbf{x}) = -dt + \sum_{\mathbf{x}'} V^\pi(\mathbf{x}') f(\mathbf{x}', \mathbf{x}).$$

Notice that $V^\pi(\mathbf{x}) = 0$ iff $\mathbf{x} \in F$ and those terms can be eliminated from the sum. Therefore, $V^\pi(\mathbf{x})$ can be approximated numerically using an iterative algorithm

$$\hat{V}^\pi(\mathbf{x}) \leftarrow -dt + \sum_{\mathbf{x}' \notin F} \hat{V}^\pi(\mathbf{x}') f(\mathbf{x}', \mathbf{x}).$$

Continuous time and continuous space versions of this algorithm can also be defined[**?**].

$$\phi_{fpt}(\mathbf{x}) = -J^\pi(\mathbf{x}).$$

Use $-\phi_{zmp}$ as an initial guess for $J^\pi(\mathbf{x})$, then improve with experience. There is some evidence that this can be accomplished in real-time on a real robot[13].

XI. Meta-Stability

For most walking robots that we consider to be walking well, the evolution from $p(\mathbf{x}, 0)$ to $p^*(\mathbf{x})$ has multiple time constants: initial conditions are rapidly forgotten as walkers stochastically either fail or converge to a quasi-stationary (metastable) distribution $p^0(\mathbf{x})$, which slowly leaks (via eventual failure) into $p^*(\mathbf{x})$. These dynamics are similar to the ferromagnetic systems studied in physics [**?**].

Something like a Kalman filter. Except the stable dynamics shrink the state distribution instead of observations.

A. Metastable Limit Cycles and the First Passage Time

The statistic that I propose as a metric of dynamic stability for walking systems subject to time-varying noise is the mean first passage time to the set F. This quantity, also called the first visit time or the mean time to failure (MTTF), has been used in many applications in engineering, physics, and chemistry[**?**], including modeling the time at which the membrane potential of a neuron passes threshold. The first passage time of a stochastic process,

$$\text{FPT}(\mathbf{x}) = \underset{t}{\text{argmin}}\ [\mathbf{x}(t) \in F | \mathbf{x}(0) = \mathbf{x}],$$

is a random variable over t that depends on the initial conditions $\mathbf{x}_0$.

XII. Discussion

- Chicken-and-the-egg problem? Answer: Policy iteration.
- ZMP is definitely an important dynamic variable, but the ZMP criterion leaves much to be desired.
- Maybe we don't just use a fall-down state F. What about asking when it's possible to enter a state S? Like Alec's idea about Jerry's work - evaluating the ability to come to a standstill.
- still need better methods for dealing with large dimension systems (see Sec. IX on the hopper).

A. Gain- and Band-limited noise

Under what conditions does this change our discussion of ferromagnets-? The FPT time is still useful - it just goes to ∞.

A hidden implication of these methods is a suggestion on how to walk over rough terrain. Although the dynamics of the ground change with every step, planning some optimal motion for each step is tiresome and potentially intractable. The suggestion here is that the thing we monitor about the ground, at least for mild terrain, is the statistics of the terrain, and we adjust our walking strategy according to these statistics. It is not until we encounter particularly harsh terrain that we need to explicitly model and plan[4].

For control. Scalar stability metric - 2nd eigenvalue.

B. Stability and Value Functions

SSM is a value function ZMP is a Q function.

XIII. Conclusions

Appendix I
A weak bound on the cumulative distribution

Here is a weak bound on the cumulative first passage time distribution based on the Markov inequality. Since $\tau \geq 0$, we have

$$\Pr[\tau \leq a] \leq aE\left[\frac{1}{x}\right].$$

Proof:

$$E[1/x] = \int_0^\infty f(x)/x dx = \int_0^a f(x)/x dx + \int_a^\infty f(x)/x dx$$
$$\geq \int_0^a f(x)/x dx \geq \int_0^a f(x)/a dx = \frac{1}{a}\Pr[x \leq a].\text{QED}.$$

Acknowledgment

The authors wish to thank H. Sebastian Seung, Dan Paluska, Yonatan Lowenstein, Uri Rokni, Martijn Wisse, Nick Roy and Andy Ruina for helpful discussions. This work was partially supported by NSF grant CCR-0122419.

[4]Comment: Yikes! Not very eloquent, but an important idea.

REFERENCES

[1] Q. Huang, K. Yokoi, S. Kajita, K. Kaneko, H. Arai, N. Koyachi, and K. Tanie, "Planning walking patterns for a biped robot," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 3, pp. 280–289, June 2001.

[2] A. Goswami, B. Thuilot, and B. Espiau, "Compass-like biped robot part I : Stability and bifurcation of passive gaits," INRIA, Tech. Rep. RR-2996, October 1996.

[3] N. S. Nise, *Control Systems Engineering*, 4th ed. John Wiley & Sons, Inc, 2004.

[4] W. Lohmiller and J. Slotine, "On contraction analysis for non-linear systems," *Automatica*, vol. 34, no. 6, pp. 683–696, June 1998.

[5] S. H. Strogatz, *Nonlinear Dynamics and Chaos: With Applications to Physics, Biology, Chemistry, and Engineering*. Perseus Books, 1994.

[6] S. Hirose, H. Tsukagoshi, and K. Yoneda, "Normalized energy stability margin and its contour of walking vehicles on rough terrain," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, vol. 1, 2001, pp. 181–186.

[7] E. Garcia, J. Estremera, and P. G. de santos, "A classification of stability margins for walking robots," in *Proceedings of the International Conference on Climbing and Walking Robots (CLAWAR)*, 2002.

[8] R. McGhee and A. Frank, "On the stability properties of quadraped creeping gaits," *Mathematical Biosciences*, vol. 3, pp. 331–351, 1968.

[9] M. Vukobratovic and B. Borovac, "Zero-moment point - thirty five years of its life," *International Journal of Humanoid Robotics*, vol. 1, no. 1, pp. 157–173, 2004.

[10] A. Goswami, "Postural stability of biped robots and the foot rotation indicator (FRI) point," *International Journal of Robotics Research*, vol. 18, no. 6, 1999.

[11] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Prentice Hall, December 2001.

[12] S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for on-line non-linear/non-gaussian bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, feb 2002.

[13] R. Tedrake, T. W. Zhang, and H. S. Seung, "Stochastic policy gradient reinforcement learning on a simple 3D biped," in *Proceedings of the IEEE International Conference on Intelligent Robots and Systems (IROS)*, vol. 3, Sendai, Japan, September 2004, pp. 2849–2854.

[14] P. Hanggi, P. Talkner, and M. Borkovec, "Reaction-rate theory: fifty years after kramers," *Reviewsof Modern Physics*, vol. 62, no. 2, pp. 251–342, Apr 1990.

PLACE PHOTO HERE

Russ Tedrake received his Ph.D. degree from the Computer Science and Artificial Intelligence Lab at the Massachusetts Institute of Technology in 2004. He is currently a Postdoctoral Associate with the Department of Brain and Cognitive Sciences at MIT. He really needs to come up with more things to say about himself here.

PLACE PHOTO HERE

Katie Byl received her B.S. (1999) and M.S. (2003) degrees in Mechanical Engineering at MIT. She is currently pursuing a doctorate in M.E., developing efficient and robust biped robots in the Computer Science and Artificial Intelligence Lab at MIT.