Metastability for High-Dimensional Walking Systems on Stochastically Rough Terrain

Mehdi Benallegue
LPPA UMR 7152 CNRS/Collge de France
11 place Marcelin Berthelot 75005 Paris
Email: mehdi@benallegue.com

Jean-Paul Laumond
LAAS, CNRS, UPR 8001
7 Av. du Colonel Roche, 31077 Toulouse
Email: jpl@laas.fr

***Abstract*—Biped walking is a complex task, but usually with a natural limit-cycle behavior when walking on an even ground. However, perturbations during walking can make the robot fall. Several works addressed the issue of measuring the robustness to disturbances, and most methods study the effect of a single perturbation. But when walking, the disturbances can be multiple, such as walking on rough terrains. The metastability is a concept that helps studying the case of multiple disturbances. The performance measure is the expectation of the time during which the robot can keep balance. However, until today, only two methods permit to measure this metrics: the discretization of all the state-space and the Monte-Carlo sampling. The former one cannot address high dimensional state-space and the latter is too much time-consuming when the falls are rare. We propose here a solution for walkers with high-dimensional states, even when falls are very rares. The novelty of this method is to rely on the property that limit-cycle walkers may return to the limit-cycle several times before to fall. This method is even extended to the cases of bifurcations or chaos. We illustrate the performance of the approach with a simulated high-dimensional actuated walking system.**

I. Introduction

Legged walking is a complex, non-linear, and highly constrained task. Nonetheless, the simplifications made by researches in biomechanics, neuroscience and robotics enabled synthetic discoveries on elementary models. One among the most important of these results is the limit-cycle property of stable walking on flat terrain, whether for humans or artificial walking systems. The limit-cycle describes a balanced state of walking, and enables to evaluate some characteristics such as energetic efficiency.

The most prominent models for artificial energy-efficient biped walking are the passive-dynamic robots [9]. These robots are able to produce stable gaits on shallow slopes without any other source of energy but the gravity. Also, the gait of these walkers have a natural convergence to a limit cycle that depends only on the natural passive dynamics of walking. These models have been extended to flat ground by adding small actuators while keeping most of the passive dynamics and the limit cycle properties [4]. These robots are known to have the closest energy-efficiency to humans [1].

However, by making these simplifications and idealizing the environment, the roughness of real grounds is not considered. Nevertheless, the robustness to unexpected disturbances is among the most important differences in walking performances between humans and robots. For example, robots show very high sensitivity to disturbances due to ground unevenness [6], while humans show impressive capabilities to reject large disturbances. Actively controlled robot have usually specific criteria to respect to keep "balance", such as keeping the center of pressure strictly inside the support convex polygon (see [13] for an overview of balance criteria of walking systems). Violating the balance criteria leads generally to an unavoidable fall. On the contrary, for most passivity-based limit-cycle walkers, there is no formal definition of a balance criterion; because the joint positions are not actively controlled, and the natural dynamic passivity leads to an unpredictability on the outcome of the perturbation. Therefore, there is usually no way to decide whether a state is balanced or not, except to wait for a stable gait or a fall.

The improvement of limit cycle walking systems in terms of disturbance rejection requires to be able to quantify and to compare the capabilities of the robots facing perturbations. Several methods were created to evaluate these performances. Some of these methods evaluate the range of perturbations that can be handled by the walker, such as measuring the size of the basin of attraction of the limit-cycle [11], or simply quantifying the biggest perturbation that can be absorbed by the walker. Other methods evaluate the rate of convergence to the limit cycle after small perturbations (the largest Floquet multiplier [9] or the Gait Sensitivity Norm [7]).

These metrics cannot show the walker performances against repetitive disturbances such as on rough terrains. That's because the two first ones measure the effect of a single perturbation, and the latter ones are local and do not model the non-linearity of the system when getting away from the vicinity of the limit-cycle.

Byl and Tedrake proposed an alternative method to estimate the balance performance of a walker, using the concept of *metastability* [3]. It consists at considering that, on rough terrains, the probability of falling tends to 1 as time goes to infinity, for any walking system. Therefore, one good way to evaluate the robustness of a walker is to evaluate the expectation of the number of steps that the robot can achieve before to fall, which is also called the *mean first-passage time* (MFPT). However, the proposed method to compute this expectation is based on a Markov chain model, and requires a discretization of all the state space. This can be possible for low dimensional robots but becomes computationally unrealistic for complex walking systems. An alternate way to compute the MFPT is to try a Monte-Carlo sampling method, which has the advantage of a high accuracy in the evaluation MFPT and an arbitrary precision in the state description, but requires very long simulations when failures are rare [2]. Making failure as rare as possible is the objective of the optimization in the design of walking systems. The Monte-Carlo method is then very restrictive.

Our contribution is to extend the metastability study to high dimensional limit-cycle walking systems, while keeping reasonable simulation time. The novelty of our method relies on the property that limit-cycle walkers may return to the limit-cycle several times before to fall[1]. We show that this property performs noticeably well to estimate MFPT, especially for very rare falls.

In the next section, we present the metastability concept and its application to limit-cycle walking systems subject to stochastic perturbations. Then we introduce the algorithm to compute MFPT and we show how it can be extended to bifurcations and chaos. In Section III, we present an example of a high-dimensional robot that has an important variability in its limit-cycle, and we show how we measured MFPT. In SectionIV, we conclude our paper with a discussion about the advantages and the limitations of the approach.

II. Metastability analysis of walking

A. Limit-cycle walking: a metastable system

The metastability is the property of some stochastic dynamic systems to keep a specific behavior for long periods, but being guaranteed to leave this state after a sufficiently long time [12]. The system reaches then what is considered as a *failed* state (see Figure 1). This class of systems can be found in several fields: such as chemical reactions, crystalline structures, brain wave patterns, etc. Computing MFPT is a usual way to estimate the stability of such kind of systems [10].

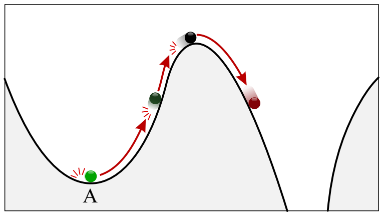


Fig. 1. The perturbations can take away the system from the metastable state ‘A’. If the system escapes its basin of attraction, it may fall in a deeper “failed” basin. When the failed state is absorbing, the system never gets back to the metastable state.

Walking on rough terrain is also a metastable system: the limit-cycle is an attractive balanced state, and the fall is considered as an *absorbing* failed state. An absorbing state is a state from which the system cannot escape.

In order to study the balance performances of a limit-cycle walker, it is usual to take profit from the periodicity of walking, by considering the time-discretization constituted by a Poincaré map. The Poincaré map describes the intersection of the state vector with a manifold while evolving in time. For walking case, the manifold is provided naturally from a cyclic event during the gait, and is generally defined by ‘the impact instant of the swing leg on the ground’. Let’s note R the set of all possible states at impact. The state dynamics between the impacts can be described with a discrete-time dynamic system:

$$x_{k+1} = f(x_k, w_k) \tag{1}$$

Where $x_k \in R$ is the state vector of the walker at the k-th impact; w_k represents the perturbations that happen during the k-th step and f is the function that describes the dynamics during the step.

This function associates to each state $x_k \in R$ and a perturbation w_k the return state x_{k+1}, which belongs to R only when the walker does not fall during the step. The fall is a state where there is a contact between a *non support* limb and the ground. Let’s define then a “fallen state” $\overset{\times}{x}$ that gathers all possible states of a fallen walker. This simplification permits to define the Poincaré manifold as $R^* = R \cup \overset{\times}{x}$, and to extend the function f by defining that

$$\forall w, f(\overset{\times}{x}, w) = \overset{\times}{x} \tag{2}$$

which means that the walker cannot get up once it has fallen, making the fall *the absorbing failed state*.

The walker has an attractive limit cycle, which is often a periodic one-step trajectory which intersects R at a given state $\breve{x}$ (pronounced “x-breve”). In contrast, in some cases, there are multiple period bifurcations and even chaotic behaviors during the limit-cycle [5]. In such cases, the limit cycle intersects R in a set of states instead of one. These cases will be discussed in Subsection II-C. For now, we consider $\breve{x}$ as the unique state, which corresponds to the fixed point of the system when no perturbations occur (see Figure 2):

$$f(\breve{x}, 0) = \breve{x} \tag{3}$$

This walking system subject to perturbations can be considered as an *infinite state machine* with two specific states which are the limit-cycle $\breve{x}$ state and the absorbing fallen state $\overset{\times}{x}$ (Figure 3).

B. Computing the mean first passage time

In our context, we define MFPT as follows: starting from a given initial state x_0, MFPT is the expectation of the number of returns that the walker can make before falling. In other terms it is $m = \mathbb{E}(K)$ where $K \in \mathbb{N}$ s.t. $x_K \neq \overset{\times}{x}$ and $x_{K+1} = \overset{\times}{x}$. The initial state x_0 may take any value, but let’s consider it, for now, as equal to the limit-cycle state $x_0 = \breve{x}$.

In fact, this definition of MFPT does *not* provide the actual *time* of the first passage, but gives the *number of steps* instead. This can seem misleading, because the steps do not necessarily

[1] Section III presents an example where this property appears.

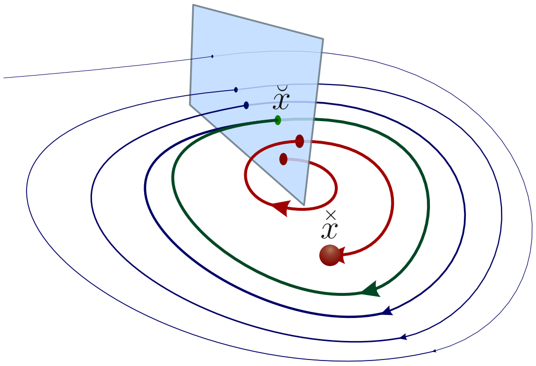


Fig. 2. A 3D line representing the state of the limit-cycle walker with no perturbations. The intersecting plane is the Poincaré manifold, the thickest ring is the limit cycle and the seperate vertex is the fall-state. When starting from a state in the basin of attraction of the limit-cycle, the state intersection with the Poincaré manifold converges to $\breve{x}$. When starting from a position outside this basin of attraction, there is a fall.

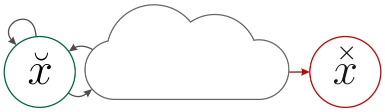


Fig. 3. The limit-cycle walker subject to perturbations as a state transition system. The cloud represents the set of states that are neither fall nor limit-cycle state.

last the same time, and it can be possible that, in some cases, few steps last longer than more ones. However, we prefer this definition because it enables to compare different robots. Indeed, a small robot has faster dynamics than a taller one, and it may have shorter MFPT while showing better (scaled) disturbances rejection.

A stochastically rough terrain is a terrain with a *texture*, for which the unevenness follows a probability distribution; this is considered as a stochastic perturbation for a robot designed to walk on flat surfaces. As stated before, we assume that on a stochastically rough terrain, the probability that the robot falls tends to one as the number of steps goes to infinity:

$$\lim_{k\to\infty} P(x_k = \overset{\times}{x}) = 1, \qquad (4)$$

And if the robot does not fall, the probability that the robot comes back, at least once, to the limit-cycle tends to one as the number of steps grows. In other terms for any neighborhood $V(\breve{x})$ of $\breve{x}$ in R:

$$\lim_{k\to\infty} P(\exists k_0 \in \mathbb{N}^* \; x_{k_0} \in V(\breve{x}) | x_k \neq \overset{\times}{x}) = 1 \qquad (5)$$

The proposed algorithm relies on these two statements to consider the walking process as a Bernoulli trial: (i) either the robot falls or (ii) it comes back to the limit cycle. Coming back to the limit cycle can be approximated by checking if the state belongs to a given neighborhood $V(\breve{x})$ of $\breve{x}$ which is small enough to be considered as a good approximation. This trial has a probability of fall p_f and a probability of limit-cycle return $(1 - p_f)$. The robot stops at any falling trial. Walking then follows a geometric distribution: a Bernoulli process which stops at the first fail. The expectation of the number of returns is then:

$$r = \frac{1 - p_f}{p_f} \qquad (6)$$

The mean number of returns does not give the mean number of steps. But if we know m_r the mean number of steps of a return and m_f the mean number of steps of a fall, MFPT can be estimated by:

$$m = r\, m_r + m_f \qquad (7)$$

The estimations of p_f, m_r and m_f can be obtained by repeating trials. Each trial starts from $\breve{x}$, and stops either at $\breve{x}$ or at $\overset{\times}{x}$. We denote $\hat{p}_f$, $\hat{r}$, $\hat{m}_r$ and $\hat{m}_f$ the estimations of p_f, r, m_r and m_f respectively.

Let's evaluate the number of walking steps which are required in order to obtain a good approximation of this value. Suppose that, in order to have the desired precision, we want to make $a \in \mathbb{N}$ trials. If we use the Monte-Carlo method, we will have to wait for $n_{MC} = a.\hat{m}$ steps to obtain the evaluation of m. But if we use our method, we will have to wait only for

$$n = a\, \hat{m}_f\, \hat{p}_f + a\, \hat{m}_r (1 - \hat{p}_f) = a\, \hat{m}\, \hat{p}_f \qquad (8)$$

which can be dramatically less than n_{MC} when the falls are rare, and never more than n_{MC}. Note that when p_f is equal to zero we have $m = \infty$. In that case, Monte-Carlo sampling is unsuitable, while our method takes $n = a.\hat{m}_r$ (from (8)).

The method efficiency relies on the fact that the robot has a non-negligible probability to come back to the limit-cycle state before to fall. However, statement (5) may be wrong for special kinds of perturbations. For example, if perturbations create a new fixed point $\breve{x}'$, then the walker never reaches $\breve{x}$ when starting from $\breve{x}'$. In that case, the previous developments are still valid, but our method may have less advantages compared to Monte-Carlo sampling because the value of $(1 - p_f)$ may be too small. In fact, our method can be applied with the state $\breve{x}$ having any value, but the biggest possible return probability $(1 - p_f)$ permits to have the fastest sampling.

So, what happens if we start from another state x_0 which is not the limit-cycle state? Then, the statements (4) and (5) can still be reasonably considered true. In that case, we still can take profit from them and use both x_0 and $\breve{x}$ as starting state. That requires then a twofold sampling, the first starting from x_0 and the second starting from $\breve{x}$. Beside the values p_f, m_f and m_r, we need, starting from x_0, to estimate also (i) the probability to fall without reaching the limit-cycle $p_{f,0}$, (ii) the mean number of steps needed to fall $m_{f,0}$, and (iii) the mean number of steps needed to reach the limit-cycle $m_{r,0}$. The new MFPT is then estimated by:

$$\hat{m} = \hat{p}_{f,0}\, \hat{m}_{f,0} + (1 - \hat{p}_{f,0})(\hat{m}_{r,0} + \hat{r}\, \hat{m}_r + \hat{m}_f) \qquad (9)$$

where $\hat{p}_{f,0}$, $\hat{m}_{f,0}$ and $\hat{m}_{r,0}$ are the estimates of $p_{f,0}$, $m_{f,0}$ and $m_{r,0}$, respectively.

C. Bifurcations and chaos

For some cases of limit-cycle walking systems, there exists multiple period bifurcations which can multiply the duration of a limit cycle. The bifurcations happen for passive walkers on steep slopes [5], or for limit-cycle robots on repetitive terrains, such as treadmill belts. For some cases the bifurcation spreads until creating what is called a *chaotic behavior* [5]. In this case the fixed point is not stable anymore. Instead, there is a set of states $\{\breve{x}_1, ..., \breve{x}_l\}$ such that:

$$\forall i \in \{1, ..., l-1\}, f(\breve{x}_i, 0) = \breve{x}_{i+1} \tag{10}$$

and

$$f(\breve{x}_l, 0) = \breve{x}_1 \tag{11}$$

(see Figure 4(a)). When subject to perturbations, the steps can lead from any state to any other, and for a period long enough eventually fails (see Figure 4(b)).

Bifurcations may be handled by three different methods.

Method 1- To consider only one limit state: Let's consider, without loss of generality, that $\breve{x} = \breve{x}_1$. In this case, Property (5) holds, and we can apply the method shown in the previous subsection. However, the return probability $(1 - p_f)$ can be very low for high number of limit-cycle states. Therefore the advantage to Monte-Carlo method is less interesting.

Method 2- The Markov chain model: Another solution is to consider the walking system as a stochastic finite-state machine (a Markov chain model) with an absorbing state $\overset{\times}{x}$ and the other states $\breve{x}_1, ..., \breve{x}_l$. When the walker is not in any of these states, it is considered *in transition* (see Figure 4(c)). Compared to the Markov chain model presented by Byl and Tedrake [3], one transition does not represent *one* walking step but potentially several ones.

In this case, we need to sample differently: starting from each states $\breve{x}_i$ and $\breve{x}_j$ we estimate (i) the transition distribution (the probability $t_{ij} = P(x_{k+1} = \breve{x}_j | x_k = \breve{x}_i)$) and (ii) the mean number of steps m_{ij} required to go from $\breve{x}_i$ to $\breve{x}_j$.

For notation simplification, we denote the fall-state $\overset{\times}{x}$ by $\breve{x}_0$. This gives the the matrix $M = (m_{ij})$ and the probability transition matrix $T = (t_{ij})$ whose first line is $\begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}$ because of the absorbing fall-state. Let's define $\mathbf{m} = \begin{bmatrix} m_1, \cdots, m_l \end{bmatrix}^t$ being the vector of all MFPTs of the limit-cycle states $\breve{x}_1, ..., \breve{x}_l$. Vector $\mathbf{m}$ is easy to compute, by considering the relationship:

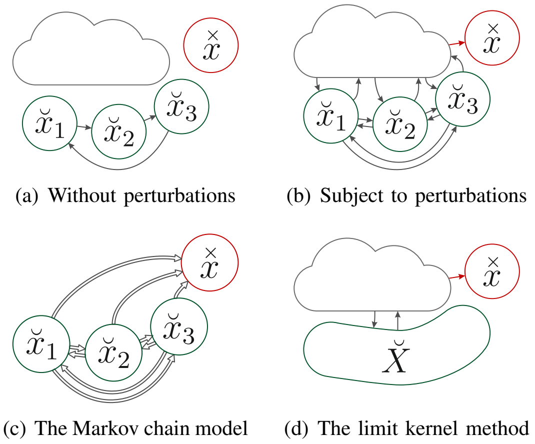
$$m_i = \sum_{j=0}^{l} t_{ij}(m_{ij} + m_j) \tag{12}$$

with $m_0 = 0$ being MFPT for the fall-state. This provides the following single-shot computation:

$$\mathbf{m} = (I - \mathring{T})^{-1} \begin{bmatrix} \sum_{j=0}^{l} t_{1j} m_{1j} \\ \vdots \\ \sum_{j=0}^{l} t_{lj} m_{lj} \end{bmatrix} \tag{13}$$

where $\mathring{T}$ is T with removed first row and first column.

This model does not rely on the bifurcation properties of the limit-cycle of the walker. In fact, it holds for any set of states $\breve{x}_1, ..., \breve{x}_l$. The method can be, for example, adapted to create partial discretization of the state space, e.g. around a limit-cycle. This would enable to reduce the fall probability p_f, but with a loss of accuracy because of the discretization. In that sense, our method constitutes a compromise between the method of Byl and Tedrake [3] (full state-space discretization) and Method 1.



(a) Without perturbations (b) Subject to perturbations

(c) The Markov chain model (d) The limit kernel method

Fig. 4. A simple example of three-states bifurcation. The cloud represents the set of states that are neither fall nor limit-cycle state. The doubled arrows represent transitions of the Markov chain model, and each transition between $\breve{x}_i$ and $\breve{x}_j$ takes the mean number of steps m_{ij}. The shape labeled '$\breve{X}$' represents the approximation of the set of limit-cycle states.

Method 3- The limit kernel method: Method 2 shows several advantages in the case of bifurcations. However, if the number of the limit-cycle states is too large, for example in chaotic behaviors, the required matrix can take a lot of memory (theoretically infinite). We then suggest to make an approximation $\breve{X}$ of the set of all state-space which is reached by this limit behavior, and to use Method 1 with $\breve{X}$ as limit state.

We call $\breve{X}$ the *limit kernel*. The method creates a loss in the accuracy of MFPT estimation, which is likely an overestimation of its actual value. That is because a non limit-cycle state x could lie inside $\breve{X}$, but non limit-cycle states are usually less balanced than limit-cycle ones; thus the balance of the state x is overvalued. Note that a too big kernel leads to overestimate MFPT while too small kernel to a higher fall-without-return probability p_f and makes MFPT longer to be estimated. In addition, $\breve{X}$ must be defined in such a way that we must be able to know whether a given state x is inside $\breve{X}$ or not.

In the following section we present an example of a limit kernel approximated by a multidimensional ellipsoid using the covariance matrix of the limit-states set. The example does not show a true chaotic behavior (a chaos due to the dynamics of the system itself), but the rounding and the time-discretization of our numerical simulations create a big set of limit-cycle states. Then we show that it is possible to reach high accuracy in MFPT estimation with a relatively loose-fitting approximation.

III. Example: A 4-DOF 2D robot

A. Robot geometry and actuation

The walking robot model we consider is depicted in Fig 5. It moves in the sagittal plane. The robot bodies consist of a rigid upper body with two masses, one around the middle of the segment (torso) and another at the top (head); and two rigid legs with a mass on the segment. Each leg is equipped with a prismatic spring-damper located between the mass and the ground. We call *toe*, the bottom of the mobile part of the leg. The hip-toe length l_p at the rest position of the spring is denoted by $l_{p,0}$. The masses distribution and limbs proportions taken for the simulation are anthropometric (Table I).

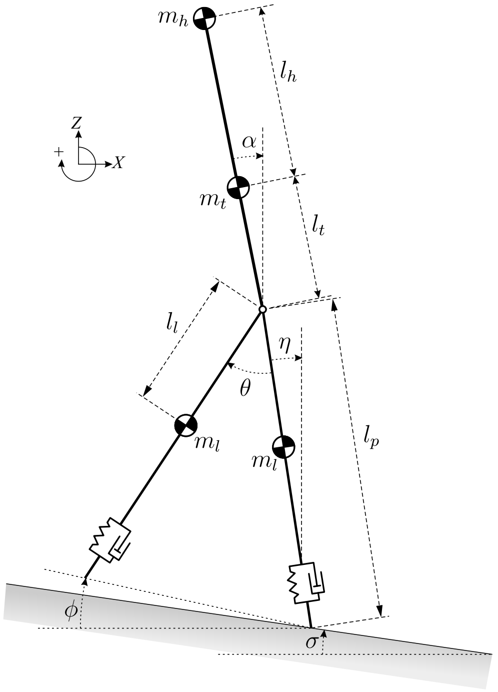


Fig. 5. A sketch representation of the actuated 4 DOF model

l_h = 0.535 m	l_t = 0.375 m	$l_{p,0}$ = 1 m	l_l = 0.40 m
m_h = 0.5 kg	m_t = 4.5 kg	m_l = 1.5 kg	

TABLE I
The parameters values of the simulated robot

The robot is considered to be always walking, that means that there is always at least a contact between a toe and the surface of the ground. If this condition is not satisfied, the robot is considered as fallen. The so-called *stance leg* is the leg in contact with the ground. The other leg is called the *swing leg*. When a leg is in a swing phase, its toe comes back instantly to the rest position of the spring, so the length of the swing leg is constant with value $l_{p,0}$. We denote by l_p the length of the stance leg.

The state vector of the robot is: $x = \left(\theta, \dot{\theta}, \dot{\eta}, \phi, \alpha, \dot{\alpha}, l_p, \dot{l_p}\right)$ where $\dot{\theta}$ is the time-derivative of θ.

The simulations are performed using the dynamic simulator ODE with a time discretization of 1 KHz. A special simulation time sample is set to 0.1 ms at the double support instant in order to increase the physical realism of the impacts. Impacts are considered inelastic and contacts are considered perfect with no slipping.

The toe of the stance leg follows a passive spring-damper behavior. The force applied to the ground has a proportional-derivative (PD) expression:

$$f_t = -K_{toe,p}(l_p - l_{p,0}) - K_{toe,d}.\dot{l_p} \tag{14}$$

where $K_{toe,p}$ is the elasticity of the spring and $K_{toe,d}$ is the damping factor. At the instant of raise-up of the stance toe, a velocity controlled impulsion is applied to the ground to give a propulsion to the robot (Figure 6). The force is automatically computed by the dynamic simulator to provide the desired velocity:

$$f_t = h(\dot{l}_{p,r}) \tag{15}$$

where $\dot{l}_{p,r}$ is the given desired velocity and h is the speed-referenced controller. The force f_t is applied during one time-step of simulation.

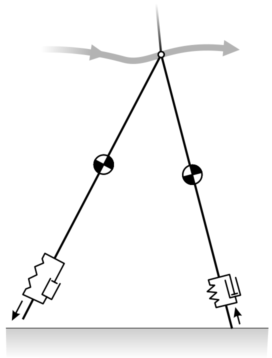


Fig. 6. The actuation and passivity of toes

The inter-leg joint is also controlled. The actuator is a pure torque generator controlled by a simple PD that generates a torque toward a reference angle:

$$\tau_h = -K_{hip,p}(\theta - \theta_r) - K_{hip,d}.\dot{\theta} \tag{16}$$

where $K_{hip,p}$ is the proportional gain, θ_r is the reference angle and $K_{hip,d}$ is the derivative gain. In order to preserve the natural dynamics of the legs the values of the gains are small.

The trunk is stabilized to keep an upright orientation. Among the several methods for controlling the upper body, we choose the solution that generates a torque between the *stance* leg and the trunk. By doing so, we leave the passivity of the swing leg creating a more efficient motion [8]. The trunk torque is actuated by a pure torque generator, controlled by a PD that brings back the torque to vertical:

$$\tau_t = -K_{trunk,p}.\alpha - K_{trunk,d}.\dot{\alpha} \tag{17}$$

where $K_{trunk,p}$ is the proportional gain and $K_{trunk,d}$ is the derivative gain.

The values of all these parameters appear in Table II.

$K_{toe,p} = 5000$ n/m	$K_{toe,d} = 200$ n.s/m	$\dot{l}_{p,r} = 1$ m/s
$K_{hip,p} = 1$ n.m/rad	$K_{hip,d} = 0.15$ n.m.s/rad	$\theta_r = 0.3$ rad
$K_{trunk,p} = 30$ n.m/rad	$K_{trunk,d} = 15$ n.m.s/rad	

TABLE II
THE ACTUATION/PASSIVITY PARAMETERS OF THE ROBOT

B. The limit kernel

When starting the simulation on a flat ground from a state in the basin of attraction, the robot reaches a limit behavior within few steps. Figure 7 shows a phase plot of the inclination of the left leg over the time during the limit-cycle. The limit cycle seems to be unique, but there is a variability in the state vectors x_k when meeting the Poincaré manifold: the thickness of the curve is the overlay of many curves. This variability is probably due to the time-discretization and rounding errors. This means that there is no real asymptotically stable fixed-point of the system, and a we have very large set $\{\breve{x}_1, ..., \breve{x}_l\}$.

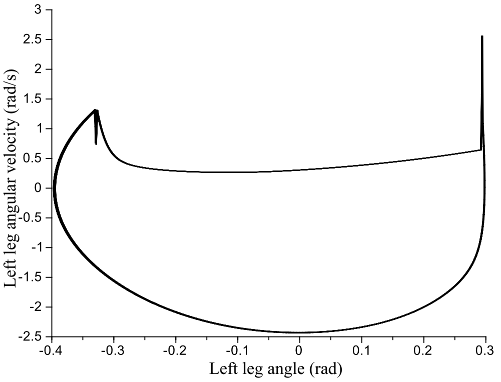
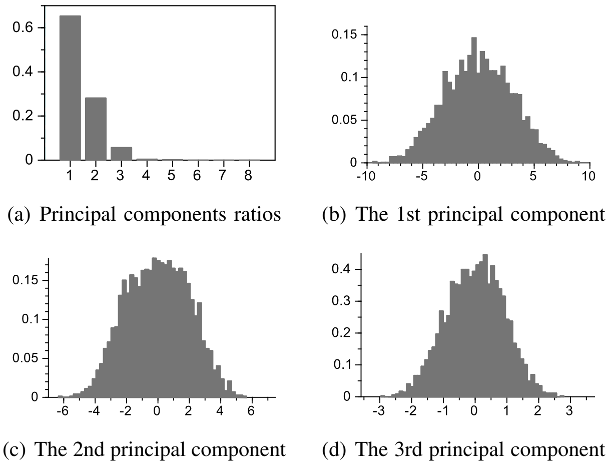


Fig. 7. Phase plot of the limit-cycle of the left leg inclination during several walking steps, each cycle here has a duration of 2 steps.

Our idea is to study the distribution of these states in the multidimensional space. We made a rough sampling of the state space (8000 initial states) and we ran a simulation from each of them. The simulations for the states that lie into the basin of attraction reached the limit cycle. In order to be sure to catch most of the possible dynamics of the limit-states, we recorded the states for each successful simulation, between the 400th and the 500th step, obtaining 3000 values of $\breve{x}_i$.

To analyze the obtained values of the limit cycle, we have used principal components analysis (PCA). In Figure 8, we see that the three first vectors carry the most important part of the variability. Within the directions of these vectors, the distribution is regular and reaches its maximum around its mean value. This suggests that a multidimensional ellipsoid can fit well the points cloud. An ellipsoid is an easy shape to define: a state is inside the ellipsoid if and only if its distance to the center is under a threshold d_0. What we mean by distance here is an anisotropic metrics defined by a semi positive definite matrix.



(a) Principal components ratios (b) The 1st principal component

(c) The 2nd principal component (d) The 3rd principal component

Fig. 8. The PCA of the limit-cycle states values. We see in (a) that the three first components carry most of the information. In (b), (c) and (d) we see histograms of their distribution.

We propose to use the distance defined by the inverse of the covariance matrix of the set of states $(\mathrm{cov}(\{\breve{x}_1, ..., \breve{x}_l\}))^{-1}$. However, the covariance matrix can be non-invertible. For example, the value ϕ is always at zero during double support, creating a zero column/row in $\mathrm{cov}(\{\breve{x}_1, ..., \breve{x}_l\})$. So, the solution is to use the Moore-Penrose pseudo-inverse of the covariance matrix $C = (cov(\{\breve{x}_1, ..., \breve{x}_l\}))^{+}$. The function of distance to the center is then defined by:

$$d(x) = (x - \bar{x})^t C (x - \bar{x}) \qquad (18)$$

where $\bar{x}$ is the mean value of $\{\breve{x}_1, ..., \breve{x}_l\}$.

Note that even if the pseudo-inverse solves the problem of non-invertibility, this new distance ignores completely the null-space of the covariance matrix. That means that whatever the value of ϕ, the distance is constant. As a consequence, a special treatment must help to decide whether a state is in the Kernel or not. We choose to follow the same choice as Byl and Tedrake [3]: we ignore ϕ in the MFPT estimation. This makes our limit kernel approximation oversized.

Figure 9 shows the distribution of the values of $d(\breve{x}_i)$. Most of the distances to the center are less than 20 while all of them are less than 40. In order to show that a relatively loose-fitting shape can be enough to estimate correctly MFPT, we take the threshold $d_0 = 1000$.

C. Simulation results

The model of perturbations we propose is to change at each step the slope of the ground (see Figure 10). The slopes follow a Gaussian white noise. The variance of the Gaussian defines the *texture* of the ground. This kind of perturbations has the advantage to not require any scaling, regarding the size of the robot, because this perturbation is equivalent to a step down or a step up with a height which is proportional to the walking-step length.

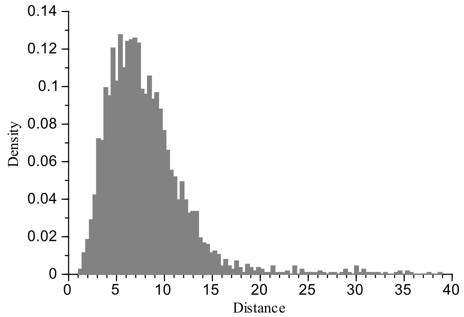


Fig. 9. A histogram of the repatition of $d(\breve{x}_i)$. There is no state at almost 0 distance because the multidimensional space is very small for small distances.

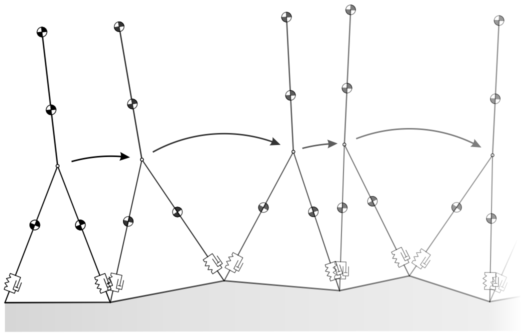


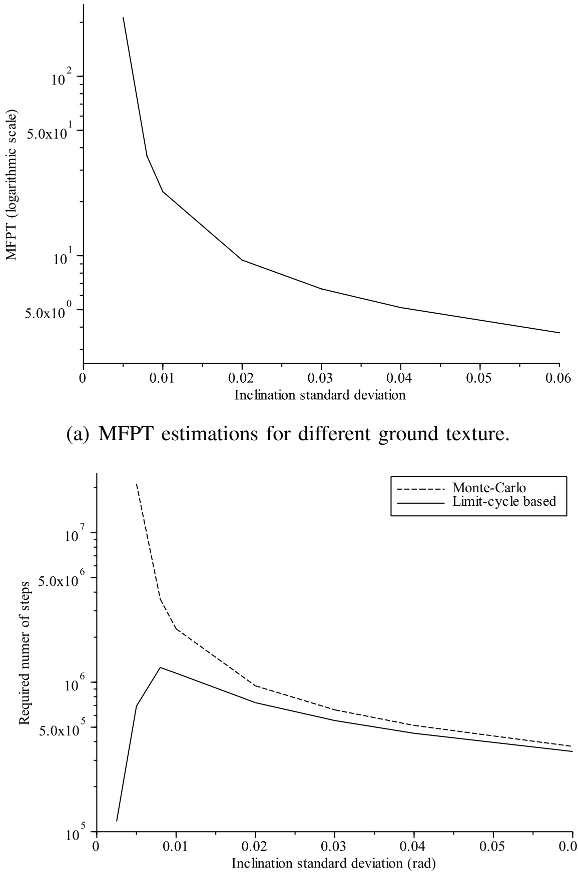
Fig. 10. The perturbations model. Each step has a different slope inclination. The inclinations follow a Gaussian white noise.

We ran the algorithm described in Section II-C with the ellipsod as a limit kernel and starting with $x_0 = \bar{x}$ and $a = 100000$ trials were performed for each slope variance. The results are shown in Figure 11(a). The robot has infinite estimated MFPT until 0.05rad of standard deviation for the ground slope. We know that this estimation is inaccurate, but it suggests that the true MFPT is much higher than the biggest MFPT we can evaluate with simulation techniques. We qualify this value as being *virtually infinite*. While the variance increases, MFPT decreases rapidly to reach less than 10 steps.

More importantly, Figure 11(b) compares the number of steps required by Monte-Carlo sampling and by our method. Considering that we simulate 100 steps in 4 seconds, the total time required to get these results when using our method is of 48 hours, while using Monte-Carlo sampling takes 319 hours. Such time costs have been estimated without considering the 0.05 rad case. including this case is out of the scope of Monte-Carlo method, while it takes only 1.3 hour for our method to estimate the virtually infinite MFPT.

In order to evaluate the accuracy of our method, we also ran a classical Monte-Carlo simulation for the case 0.1rad. MFPT estimation was 22.66 steps, while it was 22.75 steps with our method. This shows an overestimation of 0.37% only but with 50% less simulation time.

We have seen through this example a relatively loose fitting limit kernel which enabled an accurate and a much faster estimation of MFPTs, especially for rare fall cases. Beside the gain over Monte-Carlo, our method has several advantages compared to other measures of robustness to disturbances, as discussed below.



(a) MFPT estimations for different ground texture.

(b) The number of simulated steps required for the estimation.

Fig. 11. MFPT estimation and the number of steps required for the estimation. Both are in logarithmic scale.

IV. Discussion

This section presents the benefits of our method over the existing approaches as well as its limitations.

In contrast to the previous robustness estimators, the metastability based estimator enables to analyze the effect of repetitive disturbances for which Property (5) holds. Indeed, the Floquet multiplier and the gait sensitivity norm methods study the convergence to the limit-cycle after a small perturbation. These approaches can be extended to the presence of repetitive disturbances but they remain local around the limit-cycle. They cannot show under which conditions the robot would fall. Furthermore these methods become less suitable in the case of bifurcations, and unable to address the case of chaos.

Some other methods, such as searching the largest admissible perturbation and measuring the size of the basin of attraction, are more global and do not need the presence of a stable limit-cycle. However, they cannot be adapted to repetitive disturbances.

MFPT is then a good metrics to measure the tolerance to rough terrains. The method by Byl and Tedrake [3] to estimate MFPT needs a discretization of all the state-space. Let's take the example of 8-dimensional state, which may be reduced to 7-dimensions if we ignore ϕ. Suppose that we want to discretize each dimension with 20 values, which is very rough, the discretization would take $20^7 = 1.28$ billion of states. Again, the computation requires the inversion of a square matrix of $20^7 \times 20^7 = 1.6384 \times 10^{18}$ cells. Our method does not have any scaling problem with the increase of the complexity of the walking system. Furthermore, our method permits an arbitrary accuracy in MFPT estimation. That's because the only states which are approximated are the limit-cycle states, and we can choose any tight approximation, with, in return, a loss in the performances of the method, but still being better than Monte-Carlo method.

However, as acknowledged, our method is not relevant if the limit-cycle changes because of the perturbations. To overcome this drawback, a solution can be to identify the new limit-cycle and to change $\breve{x}$ for the new one. Also our method fails when important chaos cannot be approximated correctly enough. Nevertheless, we keep in mind that our method can never be worse than a Monte-Carlo approach that remains the only alternative.

Finally, the biggest disadvantage of our method is that it is more time-consuming than approaches based on Floquet multipliers and gait sensitivity norm. But there is virtually no other solution for such nonlinear system than trying several times or multiple possibilities in order to have a synthetic, global estimator about walking performances on textured terrains.

V. Conclusion

This paper has shown how to take advantage of the existence of the limit cycle to efficiently estimate metastability properties of a walking robot. The method considers the walker as a Markov chain, with a special absorbing state representing the fall, and other states representing the limit-cycle behavior. When the walker is neither fallen nor in a limit-cycle state, it is considered in transition. With this model, the computation of MFPT becomes straightforward.

On the other hand, the concept of metastability is particularly adapted to the model of stochastically rough terrain. The given example have shown that the assumption which is the basis of our approach is nicely verified with this model of perturbations: the robot comes back regularly to the limit cycle, and this may happen even for high variances of the terrain inclinations.

Finally, MFPT is a good criterion to measure the robustness of walking systems to repetitive disturbances. It can be used either (i) to estimate the tolerance of a walking system to given disturbances and to give an estimation of the distance that can be traveled before falling, (ii) to use it for comparing different robots, even if they have different dynamics, (iii) to optimize the dynamical structure of a robot in order to maximize MFPT, or (iv) to compare different control policies for a same robot.

Acknowledgments

This work is partially supported by grants from the EU Project CLONS (Closed-Loop Neural Prosthesis for Vestibular Disorders, Grant Agreement 225929). We deeply thank Alain Berthoz for his support and our stimulating discussions.

References

[1] R McNeill Alexander. Walking made simple. *Science (New York, N.Y.)*, 308(5718):58–59, April 2005. URL http://www.ncbi.nlm.nih.gov/pubmed/15802589.

[2] K. Byl. *Metastable Legged-Robot Locomotion*. PhD thesis, Massachusetts Institute of Technology, 2008.

[3] K. Byl and R. Tedrake. Metastable Walking on Stochastically Rough Terrain. In *Proceedings of Robotics: Science and Systems IV*, Zurich, Switzerland, 2008.

[4] S. Collins, A. Ruina, R. Tedrake, and M. Wisse. Efficient bipedal robots based on passive-dynamic walkers. *Science*, 307(5712):1082–1085, February 2005.

[5] A. Goswami, B. Thuilot, and B. Espiau. A study of the passive gait of a compass-like biped robot: symmetry and chaos. *International Journal of Robotics Research*, 17(12):1282–1301, December 1998. URL http://dx.doi.org/10.1177/027836499801701202.

[6] D. G. E. Hobbelen and M. Wisse. Limit cycle walking. In *Humanoid Robots: Human-like Machines*, pages 277–294. 2007.

[7] D. G. E. Hobbelen and M. Wisse. A disturbance rejection measure for limit cycle walkers: The gait sensitivity norm. *IEEE Transactions on Robotics*, 23(6):1213–1224, Dec 2007. URL http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4359260.

[8] T. McGeer. Stability and control of two-dimensional biped walking. Technical report, Simon Fraser University, Centre for Systems Science, Burnaby, B.C., Canada, 1988.

[9] T. McGeer. Passive dynamic walking. *The International Journal of Robotics Research*, 9(2):62–82, 1990.

[10] R. Muller, P. Talkner, and P. Reimann. Rates and mean first passage times. *Physica. A*, 247(1-4):338–356, 1997. URL http://cat.inist.fr/?aModele=afficheN&cpsidt=10329138.

[11] A.L. Schwab and M. Wisse. Basin of attraction of the simplest walking model. *Proceedings of ASME Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, 21363, 2001.

[12] P Talkner and P Hänggi. Discrete dynamics and metastability: Mean first passage times and escape rates. *Journal of Statistical Physics*, 48(1/2):231–254, 1987.

[13] P.-B. Wieber. On the stability of walking systems. In *Proceedings of the International Workshop on Humanoid and Human Friendly Robotics*, 2002. URL http://hal.inria.fr/inria-00390866/PDF/article.pdf.