Trajectory Generation for Dynamic Bipedal Walking through Qualitative Model Based Manifold Learning

Subramanian Ramamoorthy
School of Informatics
The University of Edinburgh
Edinburgh, EH9 3JZ, UK
Email: s.ramamoorthy@ed.ac.uk

Benjamin J. Kuipers
Department of Computer Sciences
The University of Texas at Austin
Austin, Texas 78712, USA
Email: kuipers@cs.utexas.edu

***Abstract*— Legged robots represent great promise for transport in unstructured environments. However, it has been difficult to devise motion planning strategies that achieve a combination of energy efficiency, safety and flexibility comparable to legged animals. In this paper, we address this issue by presenting a trajectory generation strategy for dynamic bipedal walking robots using a factored approach to motion planning - combining a low-dimensional plan (based on intermittently actuated passive walking in a compass-gait biped) with a manifold learning algorithm that solves the problem of embedding this plan in the high-dimensional phase space of the robot. This allows us to achieve task level control (over step length) in an energy efficient way - starting with only a coarse qualitative model of the system dynamics and performing a data-driven approximation of the dynamics in order to synthesize families of dynamically realizable trajectories. We demonstrate the utility of this approach with simulation results for a multi-link legged robot.**

I. Introduction

Achieving a combination of task-level flexibility and a measure of reliability is one of the most significant challenges in robotics. As researchers explore the use of innovative robot concepts in applications ranging from rescue to planetary exploration, these concerns become all the more pressing. Owing to the fact that humanoid robots represent great promise for operation in unstructured environments, just like us humans, they have captured the imagination of the researcher and the lay-person alike, and resulted in several decades of intense research activity. Yet, we are far from motion planning algorithms that allow us to replicate the efficiency and flexibility of human walking over natural unstructured terrains, e.g., while stepping over a sequence of rocks on a pond.

In our view, such operation in unstructured environments requires two major elements: (a) efficient representations for task encoding, i.e., ways to specify the task of walking that remain applicable despite variation in the environment and (b) techniques for dealing with imprecision in models of the system dynamics and the environment. In this paper, we present an approach to trajectory generation that incorporates these elements.

In recent years, there have been a number of excellent attempts to address the broad issues raised above, through innovations in planning and control techniques. One set of techniques [1], [2] has involved the use of nonlinear control theory, including feedback linearization and sliding mode algorithms, to compensate the nonlinear dynamics of the robot so that the resulting linear system can be controlled along specific trajectories. The trajectory to be followed is designed by specifying constraints on key variables, such as the center of mass, and computing the motion of all degrees of freedom to satisfy these constraints. This is achieved using numerical optimization [1], symbolic planning [2] and other related techniques. A closely related procedure is that of dynamics filtering [3], which accepts the output of purely kinematic planned trajectories [4] and 'corrects' them through a model-based optimization process. All of these techniques yield the very desirable result of task-level control. However, the user of these methods must develop detailed and accurate analytical models of the system and environmental dynamics and use them to design the high-bandwidth controllers that compensate nonlinearities and enforce desired trajectories. In practice, to the extent that robots are engineered machines, it is reasonable to assume some knowledge of the kinematic structure of the robot, and perhaps also of the coarse structure of the dynamics. However, the external environment that the robot must interact with and its dynamic properties are often poorly understood and models of these effects (e.g., friction, impact, slip) can be unreliable. As robots are expected to quickly and seamlessly fit into new environments, it is desirable to explore approaches that can accommodate significant imprecision in these models of dynamics.

An alternative is found in methods based on machine learning. Recently, there have been many attempts to autonomously learn control strategies for dynamic walking [5], [6], [7]. Many of these techniques have achieved promising success in structured environments. However, they do not yet provide sufficient leverage over task level variables, e.g., foot placement, that are crucial in unstructured environments.

Another very promising approach to walking is passive dynamic walking. Emphasizing minimalism and simplicity, these robots avoid the need for high-bandwidth control by utilizing the natural nonlinear dynamics of the robot instead of actively compensating for it. It has been shown [8], [9] that bipedal robots may be constructed to walk down slopes with no actuation or computing elements at all. Actuated versions of such robots have also been constructed to achieve level ground walking. In the same spirit, [10] describes a strategy

ICRA 2008.
http://homepages.inf.ed.ac.uk/sramamoo/pubs.html

for bipedal walking that is based on 'intuitive' local strategies. This is very desirable and represents one of the keys to replicating the efficiency and grace of natural bipedal walkers. However, it has been difficult to dictate task-level goals within the passive dynamic walking framework without resorting to significant simplifying assumptions about the task and terrain conditions. An approach to addressing this problem is presented in [11], where one finds a strategy for intermittently actuated passive walking that achieves task-level control in the context of a canonical template model, the compass gait.

The primary focus of the current paper is a solution to the problem of how a low-dimensional task-level control strategy, such as in [11], can be lifted to the more complex morphology and dynamics of a higher-dimensional legged robot - despite significant imprecision in models of its dynamics. In other words, given only qualitative insights regarding the underlying nature of a complex behavior, how does one utilize it to shape learning in a high-dimensional dynamical system? We present an algorithmic solution to this problem wherein we solve a data-driven manifold approximation problem in the high-dimensional phase space and regularize the approximation using the qualitative model. So, the low-dimensional model describes the essential nature of the motion plan - at each footstep and in response to a changing environment, decoupling this issue from the high-dimensional approximation problem which only needs to address the problem of learning dynamics from data. This approach bears an essential similarity to previous work such as [10] and [1] in its use of an abstract model to define the task. The primary difference lies in our focus on data-driven methods for acquiring and improving the control strategy.

The high level structure of the argument in this paper is diagrammed in figure 1. Following this, we begin in section II, with a brief discussion of low-dimensional template models and what they provide to our problem. In section III, we will discuss how these low-dimensional task encodings relate to the high-dimensional dynamics of the robot. In essence, the abstract plan induces a submanifold in the phase space of the robot. This submanifold is difficult to characterize analytically but it can be learned from experimentation, as we will show in section IV. In section V, we will present simulation results for a planar multi-link legged robot, demonstrating that the approach of this paper enables the robot to achieve the difficult goal of traversing irregular terrain, while also achieving energy efficiency comparable to versions of passive walkers such as in [10]. We conclude with a brief discussion of the significance of these results in section VI.

II. Background: Template Models for Dynamic Bipedal Walking

Humanoid robots are complex mechanisms, involving many links, actuators and other elements that render it a high dimensional nonlinear system. Directly modeling this system as a general dynamical system is difficult, especially for a task such as walking in unstructured environments where agent-environment interaction effects are critical but imprecisely known. However, viewed as a specific behavior to be achieved, the task of walking has a simple underlying structure.

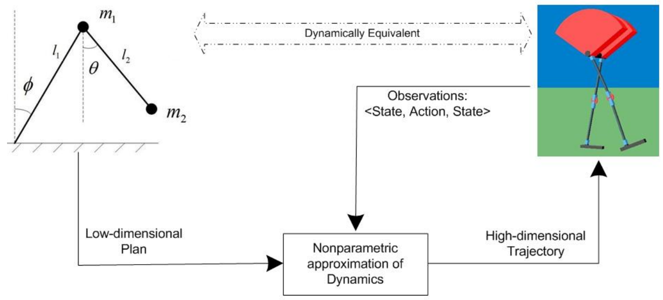

Fig. 1. Block diagram schematic representing the proposed approach. A low-dimensional template model is used to synthesize a plan. The template model is known to be equivalent to the bipedal robot and the properties of the plan can be established through systematic analysis [11]. The process of embedding this plan in a high-dimensional robot system involves a manifold approximation problem, which forms the focus of this paper.

This structure can be understood in several ways. Biologists studying the mechanics and energetics of walking [12], [13], [14], [15] have found that human walking is consistent with a simple model based on the dynamics of a pendulum, called the compass gait. In fact, it has been argued in the biological literature that many tasks involving locomotion [15], [16] and manipulation [17] are structured hierarchically in terms of a template model, involving the minimum number of variables required to express task requirements, and an embedding mechanism to anchor the resulting plan in the more complex morphology.

Even if one were not interested in these facts about biological behavior, one could perform a simple experiment with one's favorite walking robot to verify this assertion - by acquiring motion capture data of relevant kinematic and dynamic variables. Analyzing the resulting data using various nonlinear dimensionality reduction techniques, e.g., [18], [19], will show that walking involves a low-dimensional submanifold structure. This fact has been utilized in recent work [7], [20], where a specific observed trajectory is projected onto a low-dimensional space in which tracking controllers may be designed. This is a desirable direction. However, existing results are primarily focussed on local adjustments to specific observed trajectories, as opposed to the issue of handling large variations and families of trajectories that are required for walking in unstructured environments. Also, since the low dimensional space is chosen somewhat arbitrarily, as the output of a statistical algorithm, the resulting behaviors can be fragile (in terms of generalization ability).

In [11], a planning and control strategy is presented that is compatible with biological models and yields low-dimensional trajectories that are amenable to principled analysis. In essence, the strategy is based on the observation that walking involves two coupled subsystems - an inverted pendulum (the torso vaulting over a rigid stance leg) and a swinging pendulum (the retracted swing leg moving from the rear to the

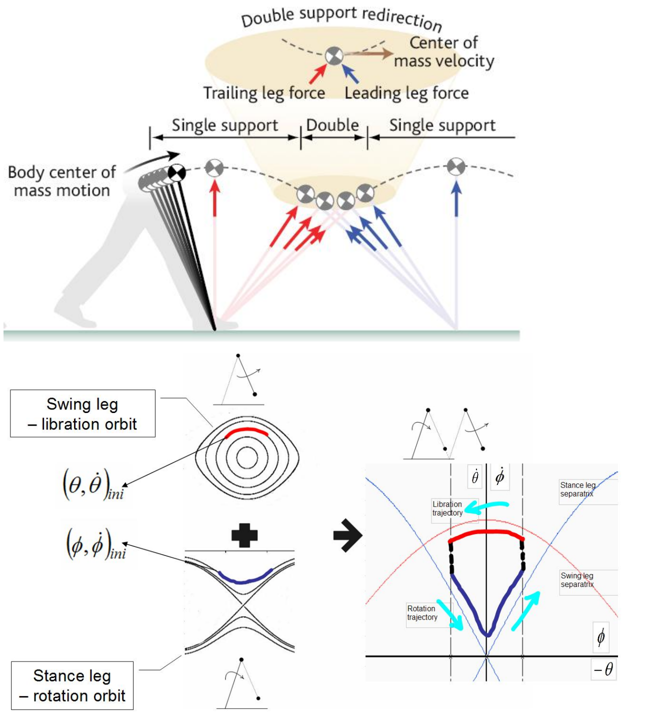

Fig. 2. *Above*: Conceptual schematic of the compass gait model of human walking (Reproduced with permission from Kuo, A.D., *Science* 309:1686-1687, 2005); *Below*: Synthesis of trajectories in the phase space of this model. Here ϕ, θ refer to the stance leg and swing leg angles (with respect to the vertical axis) respectively.

forward foot placement point). Figure 2 depicts this model and the corresponding phase space picture of the trajectories. The low-dimensional (template) space is the product space of the phase spaces of these pendulum subsystems. Each subsystem can evolve passively along a periodic orbit, requiring actuation only to switch between these families of orbits at footfall. The process of walking involves synchronized evolution of these two orbits.

For the purposes of the current paper, all we need to know about the template level strategy is that it is a predictive control algorithm that takes as input a finite-horizon goal - location of the next desired footfall, and produces trajectory segments corresponding to the template variables (angles, lengths and angular velocities, i.e., $\mathcal{S}_L \equiv \langle \phi, \theta, \dot{\phi}, \dot{\theta}, l_{st}, l_{sw} \rangle \subset \Re^6$). This algorithm can be invoked at each foot step, given the desired location of the next footstep, and a finite horizon trajectory can be obtained to achieve this goal. We would like to point out that this template-level strategy could be replaced with any equivalent strategy, perhaps based on an alternate control design framework, without significantly altering the subsequent argument of this paper.

During the finite horizon, the dynamics of the template have the structure of a mapping, $\mathcal{M}_L : \mathcal{S}_L \mapsto \mathcal{S}_L$. An important aspect of the control strategy in [11] - which mirrors biological observations - is that with the exception of the brief periods of time in double support (when a switch is made between different instantiations of $\mathcal{M}_L$) no actuation is required. In double support, the combination of actuation and impact forces have the effect of enforcing a jump in state space - which sets up the boundary conditions for the next finite-horizon template-level trajectory generation problem mentioned above.

Concretely, $\mathcal{M}_L$ (from [11]) can be represented in terms of the following phase space trajectories:

- Stance leg trajectory:
  $\dot{\phi} = \sqrt{\frac{2E_1}{m_{torso} l_{st}^2} + \frac{2g}{l_{st}}(1 - \cos\phi)}$, where E_1 is the energy which may be used as a parameter.
- Swing leg trajectory:
  $\dot{\theta} = \sqrt{\frac{2g}{l_{sw}} \cos\theta - \frac{2E_2}{m_{leg} l_{sw}^2}}$, where E_2 is the energy parameter.
- Swing leg length:
  Instantaneous extension/retraction is not physically realizable, so it is approximated by a quadratic of the form $l_{sw} = \min\{\alpha\phi^2 + l_{swmin}, l_{swmax}\}$.
- Stance leg length: Essentially a rigid pendulum of length l_{st}, perhaps with a minor correction for plantar-flexion of the form, $l_{st} = \epsilon\phi + l_{stnom}, 0 < \phi < \phi_{max}$

The above set of equations are a canonical parameterized representation of the family of trajectories that could be synthesized with the template model. A leg might begin by executing the stance leg trajectory, retract after footfall to become the swing leg, execute the swing leg trajectory and finally extend to touch down and return to the stance leg role. Given a foot placement goal for a particular step, the template algorithm instantiates specific numerical values for parameters (E_1, E_2) in the above equations. When combined with start and end poses of the legs, $(\phi, \theta)_{ini,fin}$, this constrains the space within which feasible realizations of high-dimensional trajectories must lie.

III. Embedding Template Model Based Plans in the High-Dimensional Dynamics of the Robot

The main problem being addressed in this paper is that of trajectory generation for dynamical robots operating in unstructured environments, despite significant imprecision in models of the dynamics of the environment or of the robot itself. We factor this problem into the subproblem of task variation, and the subproblem of imprecision in models of dynamics. The template strategy mentioned in the previous section addresses the problem of task variation - given a task level goal, what is the qualitative nature of the resulting trajectory. This then reduces the overall problem to that of learning models of dynamics from experimental data.

In order to learn this model, we realize that the task of walking corresponds to the restriction of the high-dimensional trajectories to a submanifold [19], [7]. Even before a robot has mastered the strategy required to execute a sophisticated behavior such as walking precisely over an irregularly spaced sequence of footholds, it can provide useful data through a process of motor babbling and other exploratory attempts to traverse irregular terrains. This provides us with a collection of short trajectory segments (in a high-dimensional phase space) that capture the possible dynamics of the robot (including specific constraints imposed on it by the environment). Given a start and end pose (based on foot placement goals), planning amounts to selecting and interpolating between appropriate segments from this set. However, given the randomized nature of the data, this often leads to spurious trajectories that correspond to some realization of the dynamics of the robot, but do not lead to achievement of the desired task. For instance, the observations include several trajectories that end up with the robot falling over on its face. A naive approach that follows the nearest observed trajectory segment can end up with the same result. This problem is handled by regularizing the approximation such that the generated trajectory $\mathcal{M}_H$ agrees with the low-dimensional plan $\mathcal{M}_L$. In the following, we will expand on these ideas and present the overall algorithm.

The bipedal robot, executing a gait, is a high-dimensional nonlinear dynamical system whose behavior over time can be represented as the map, $\mathcal{M}_H : \mathcal{S}_H \mapsto \mathcal{S}_H$, where $\mathcal{S}_H \subset \Re^{12}$ is the set of all joint angles and velocities, $\{q_i, \dot{q_i}\}$.

The essence of our approach to computing high-dimensional trajectories is to establish an equivalence between the maps $\mathcal{M}_H$ and $\mathcal{M}_L$. Due to the fact that $\mathcal{M}_H$ is a much more complex higher-dimensional system, we will take this equivalence to be in the nature of an approximation. Over each finite horizon between footfalls, we require that the following diagram must commute:

$$\begin{array}{ccc} \mathcal{S}_H & \xrightarrow{\mathcal{M}_H} & \mathcal{S}_H \\ \big\downarrow \pi & & \big\downarrow \pi \\ \mathcal{S}_L & \xrightarrow{\mathcal{M}_L} & \mathcal{S}_L \end{array}$$

In practice, the maps will certainly not agree exactly, so at any point along the step, it is possible to define an error, $e_{comm} = f(\pi(\rho) - \eta), \rho \in \mathcal{S}_H, \eta \in \mathcal{S}_L$. The projection $\pi(\rho)$ can be computed from simple geometric/kinematic considerations, so this error can be evaluated along a desired path. In particular, it is clear that this error can be defined for arbitrary trajectory segments lying anywhere in phase space. If one were to fix a phase space trajectory in $\mathcal{S}_L$, then it is possible to associate an error with every trajectory segment lying in $\mathcal{S}_H$. This induces a vector field in $\mathcal{S}_H$ in terms of geodesic paths that satisfy the commutativity requirement stated above.

Now, $\mathcal{M}_L$ is explicitly defined above but $\mathcal{M}_H$ is not assumed to be known analytically. It will need to be *learned* from experience. In other words, we know $\mathcal{M}_H$ only as a set of observations of the form $\langle \rho^-, \tau, \rho^+ \rangle_i$ (where τ is a vector of torques) acquired over very small time intervals (e.g., 10 ms), as the robot performs some random sequence of actions in order to explore the phase space. So, the trajectory generation challenge is to compute the sequence of ρ_j, given the start and end poses for a foot step, such that $\sum_j e_{comm}$ is minimized. From kinematic considerations, a foot placement goal corresponds to an end pose. By iteratively solving this trajectory generation problem at each foot step, the robot is able to achieve a sequence of foot placement goals and navigate irregular terrain. Note that the core calculation involves a trajectory from a start to end pose, selecting from a family of possible trajectories. External effects such as impact and dissipation are naturally handled - as perturbations to the start pose from which the computation for the current foot step is performed.

This problem can be addressed in the general framework of manifold learning [18]. Traditionally, the manifold learning problem is posed as one of dimensionality reduction. In fact, we have the opposite goal in mind - to embed a low dimensional plan in a high-dimensional system. In order to do this, we adopt the following procedure:

1) Allow the robot to perform a sequence of random actions and acquire a collection of points $\langle \rho^-, \tau, \rho^+ \rangle_i$. In practice, it is often quicker to begin with some *ad hoc* strategy that serves as a seed for reasonable behaviors[1]. However, the procedure will work even for much more random scenarios - given sufficient trials to provide suitable coverage of state space.
2) Organize observations in the form of a graph. Weight each edge of the graph in terms of the error e_{comm} (see Algorithm 1). This endows a manifold structure on the observations with the error acting as the metric.
3) Compute shortest paths in this graph, i.e., geodesics on the manifold, to obtain a sequence of ρ_j.

We make the following remarks at this point.

- This is a general procedure so that $\mathcal{M}_L$ may be any suitably smooth vector field whose effect on the states can be evaluated in the form $\eta^+ = f_{\delta t}(\eta^-)$ (in particular, it can differ from the one described in section II, if so desired) and the main requirement on $\mathcal{M}_H$ is that we have a sufficiently distribution of observations $\langle \rho^-, \tau, \rho^+ \rangle_i$. By continually updating a database of observations, this procedure can naturally handle variations in system and environmental dynamics.
- As posed, the graph/manifold may be endowed with convergence to a desired trajectory in the sense that if the system is disturbed away from the desired trajectory (a sequence of desired ρ_j), moving it to a new point $\hat{\rho}$, then it can correct its course by following the shortest path from this new position to the goal, via a sequence of $\hat{\rho}_j$.
- The primary role of e_{comm} is to filter out spurious paths that do not result in stable walking. If the observations were obtained exclusively from an already experienced stable walker capable of performing all behaviors of interest, then this weight is redundant. However, when observations are noisy and error-prone, this term is necessary for task achievement.

[1] As a practical matter, one may adopt a sequential strategy for experiments - going from entirely randomized experiments on a simulator to a laboratory floor to an outdoor environment which may be the ultimate target for operation. This allows for sufficient data to be collected from all conditions while still minimizing the exposure of a physical robot to the harshness of exploratory behaviors.

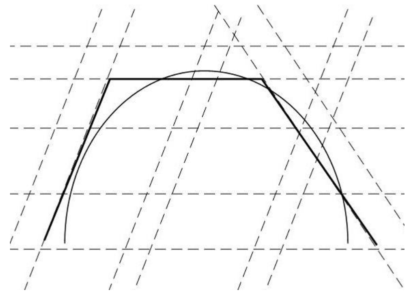

Fig. 3. Conceptual structure of the approximation problem. The observations are depicted as thin dashed lines - each dash representing a segment of experience. The desired trajectory is the circular arc. In order to achieve the desired trajectory, the system makes use to observed trajectories along the heavy line. Also, note the presence of spurious observed trajectories that could connect start and end points, but do not follow the curve.

IV. Algorithm for Qualitative Model Based Manifold Learning

The nature of the nonparametric approximation problem being solved by the manifold learning algorithm may be understood in terms of the conceptual picture in figure 3. In this picture, the plan trajectory, i.e., the sequence of η_j ($\in \mathcal{M}_L$), is the circular arc. The observed trajectories lie along the dashed lines and, by themselves, none of them is sufficient to execute the desired behavior. However, suitable portions of these observed trajectories may be synthesized into a trajectory that approximates the desired plan. As the density of these observations goes up, quality of approximation improves. Moreover, the database can contain many spurious trajectories - possible in some realization of the dynamics of the robot but incompatible with the specifications of the task. In our scheme, they will be penalized (i.e., higher e_{comm}) more than any of the other observations and will not feature in the geodesics - in effect, spurious observations will be rejected.

Based on this scheme, we present the procedure (see Algorithms 1, 2 and 3) for nonparametric approximation of the high-dimensional dynamics and its use in generating the trajectory for each foot step.

Figure 4 visually depicts the key steps in algorithm 1. Note that this depiction is really a caricature in that the dimensionality of the problem being solved is much higher.

Lastly, we note that the generated geodesic sequence ρ_j is a dynamically realizable trajectory that not only accounts for kinematics but also the dynamics and constraints applicable to the robot - due to the fact that it is synthesized as a composition of actually observed motions. These observations also included torques (recall that we have $\langle \rho^-, \tau, \rho^+ \rangle_i$). This data can be utilized to generate a corresponding profile of torques as outlined in algorithm 4.

V. Experiments

The motivation for this work comes from the task of achieving a specified sequence of footholds, using trajectories that mimic the passive nature of human walking. Towards this end, we began with a template model and a corresponding low-dimensional plan that is known to be sufficient to achieve specified footholds using an intermittently actuated passive gait. Then the focus of this paper has been on algorithms that generate high-dimensional trajectories that may be realized on a multi-link legged robot. To demonstrate the utility of this algorithm, we will show the result of this computation for different final poses. At the beginning of a step, when the robot has both feet on the ground and is in an initial pose ρ_{ini}, the location of the upcoming foothold is equivalent to the specification of a corresponding end pose ρ_{fin}. So, we will show that it is possible to achieve a *set* of ρ_{fin} using our procedure. Iteratively computing such trajectories at the beginning of each foot step endows the robot with the ability to navigate the desired sequence of footholds. In particular, since the high-dimensional trajectory is synthesized from observed data, the resulting trajectory is realizable without additional corrections.

Our experiment is based on a physically realistic simulated

Algorithm 1 Nonparametric approximation of $\mathcal{M}_H$

INPUT: Observations of $\{\rho^-, \tau, \rho^+\}_i, i = 1, \ldots, N$, Specific realization of $\mathcal{M}_L$
OUTPUT: Weighted graph representing $\mathcal{M}_H$
Project the observed points ($\eta = \pi(\rho)$) to obtain $\{\eta^-, \eta^+\}_i$
Compute inverse covariance matrix, Σ_H^{-1} using the set $\{\rho_i^-\}$
Define connectivity graph, $G = (V, E)$.

for all i **do**
  Add vertex $V_i \equiv \rho_i^-$
end for

for all V_i **do**
  Determine k-nearest neighbors V_j
  using the Mahalanobis distance $\left(D = \sqrt{(\rho_i^+ - \rho_j^-)'\Sigma_H^{-1}(\rho_i^+ - \rho_j^-)}\right)$
  for all V_j **do**
    Add edge $E_k(V_i, V_j)$
  end for
end for

for all E_k **do**
  Determine $\sum_j e_{comm} = \| \eta_{des\,j} - \pi(\rho^+)_j \|$
  where $\eta_{des\,j}$ is computed using $\mathcal{M}_L$
  Set the weight of edge E_k to e_{comm}
end for

Algorithm 2 Generate geodesic sequence ρ_j

Given initial and final poses, ρ_{ini}, ρ_{fin}
Given weighted connectivity graph $G = (V, E)$
Compute $\rho_j = \textit{ShortestPath}(\rho_{ini}, \rho_{fin})$

Algorithm 3 Generate geodesic sequence ρ_j (Handling Disturbances)

Given initial and final poses, ρ_{ini}, ρ_{fin}
Given weighted connectivity graph $G = (V, E)$
Once for each step: Compute All-Pairs Shortest Paths

while ρ_{fin} has not been reached **do**
  Decode and execute *Path*$(\rho_{current}, \rho_{fin})$
end while

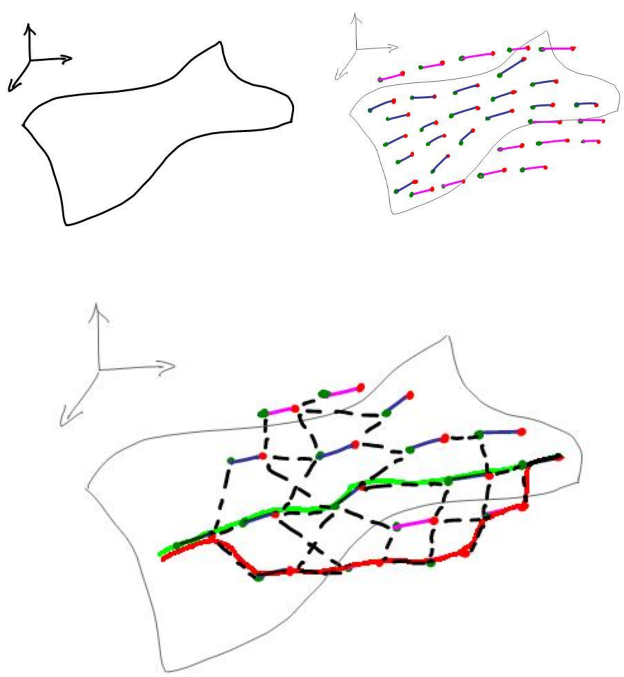

Fig. 4. *Top*: The nature of the approximation problem. The surface on the left depicts the manifold that needs to be approximated. What we actually have are short trajectory segments, some of which lie on the task manifold (shown in blue) and some that do not (shown in magenta). *Bottom*: The approximate geodesics generated using algorithm 1. The observed trajectory segments are organized into a nearest-neighbors graph (depicted here using only a few edges). By weighting these edges with e_{comm} and computing weighted shortest paths, we are able to restrict shortest paths to the desired geodesics (shown in green) as opposed to spurious paths (shown in red).

robot based on [10][2]. We begin by collecting data from this simulated robot as it executes an *ad hoc* strategy for walking. This strategy is highly noisy (deliberately injected) and not capable of achieving more than 3-4 foot steps before stumbling and falling over. However, this strategy is sufficient to acquire relevant data (quicker than random search in state space). Further, for the purposes of computational convenience and to demonstrate that our proposed algorithm does not need to begin with very dense or sequentially ordered sets of

[2]We use the *Yobotics Simulation Construction Toolset* for this experiment. The simulated robot is based on a variant of the planar *Spring Flamingo* robot - with rotational joints at human-like hips, knees and ankles. The simulated robot has a heavy torso (12 kg) on light legs (1 kg distributed between three links).

Algorithm 4 Generate Torque Profile (for each joint)

INPUT: Trajectory sequence $\{\zeta_j\}, j = 1, ..., N, \zeta_j \in S_H$ and a set of observations $\mathcal{O} = \{\rho_i^-, T_i, \rho_i^+\}, \rho_i \in S_H, T_i \in A_H \subset \Re^6, i = 1, ..., M$ (where T is a 6-dim vector of torques and ρ is a 16-dim vector of states)
OUTPUT: Sequence of torques $\{\tau_j\}$, time-integrated torque $\mathcal{T}_{int}$ and energy, $\mathcal{E}$
for $j = 1 : N - 1$ **do**
  Find K (e.g., = 20) points from the set $\mathcal{O}$ defined by the distance $\| \zeta_j - \rho_i^- \|$, including a set of K torques for the chosen joint, $\{\hat{\tau}_k\}$
  for $k = 1$ to K **do**
    With $d^- = \| \zeta_i - \rho_k^- \|$ and $d^- = \| \zeta_{i+1} - \rho_k^+ \|$, define the average distance, $d_{Mk} = \frac{d^- + d^+}{2}$ (where $\| . \|$ is the Mahalanobis distance)
  end for
  $\tau_j = \frac{\sum_1^K exp(-d_{Mk})\hat{\tau}_k}{\sum_1^K exp(-d_{Mk})}$
end for
To compute time-integrated torque:
  $\mathcal{T}_{int} = \sum_1^{N-1} \tau_j.\delta t$
To compute energy:
  Apply Savitzky-Golay filter to de-noise velocity profile
  Compute $\mathcal{E} = \int \tau_j.\dot{z_j}$ (e.g., using Simpson's 3/8 rule).

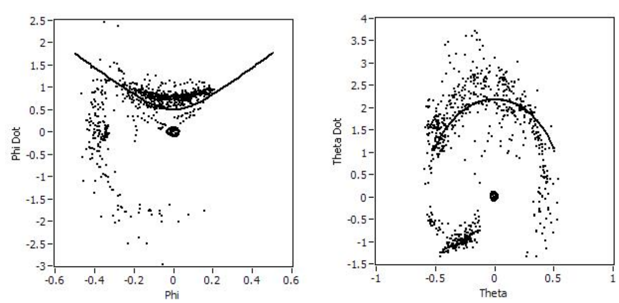

Fig. 5. Coverage obtained by the data collection process adopted in our experiment. Solid squares correspond to individual points, η^-, in the angle - angular velocity phase space corresponding to the stance and swing legs. A typical plan trajectory (corresponding to the conceptual scheme in figure 2) is superposed. Also, note the data points corresponding to inter-step transitions, leading in the vertical (increasing velocity) direction.

observations from perfectly stable walking, we subsample the data (by low-discrepancy uniform sampling in time) down to a scrambled database of 1000 points. The coverage obtained with this process is indicated in figure 5.

Using this data, and applying the algorithms in section IV, we are able to generate the trajectories depicted in figure 6. Note that we are able to generate trajectories with various end poses corresponding to different step lengths. This is an improvement over many existing machine learning based results that are limited to mimicry of specific temporally ordered observed trajectories, or to local smoothing. Instead, we are able to utilize a wide variety of noisy trajectories, representing many different gaits, to synthesize a parameterized family of

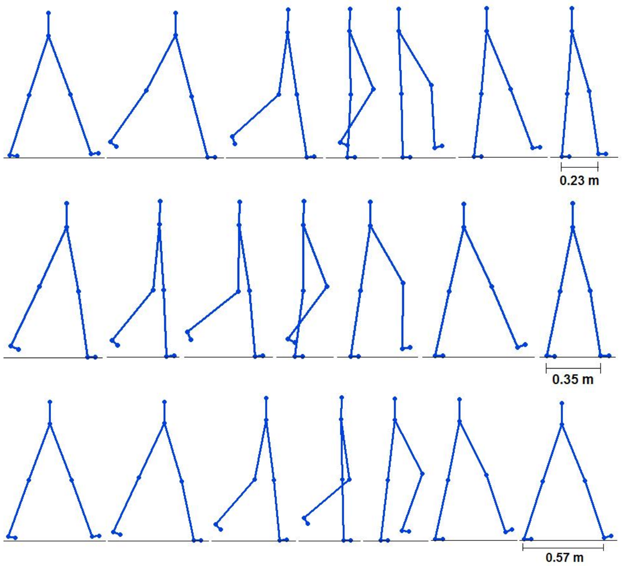

Fig. 6. Trajectories generated using the proposed algorithm. We have depicted captures from three different gaits corresponding to different foot placement, and hence end pose, goals.

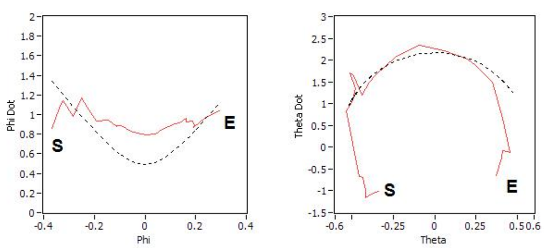

Fig. 7. Comparison (in the $\mathcal{M}_L$ phase space) between the plan (dotted line) and actual (solid line) trajectories corresponding to the longest step instance in figure 6. Start and End points (denoted S and E respectively) are chosen so as to also illustrate the process of inter-step transition ("vertical segments away from the plan trajectory").

gaits. Figure 6 depicts three specific instances from this family.

Lastly, we point out that the algorithm presented above is not only able to handle the irregular footholds requirement, but also it does this *without using much more energy than other actuated versions of passive walkers*. In order to make this point, we compare the energy consumption of our strategy against three variations of the strategy outlined in [10]. We compare against the basic periodic gait (Passive), a perturbed version of this gait that yields quasi-periodic behavior (Passive Period-2) and a variant that involves adaptive control of the swing leg (Fast). As shown in table I, the manifold approximation procedure yields trajectories that are comparable to (and slightly better than) the variants that are capable of achieving anything other than a purely periodic gait. In fact, we emphasize that *only* the manifold approximation strategy is capable of achieving the precise sequence of irregularly spaced footholds - so it is all the more reassuring that this is being achieved with comparable energy efficiency to the other passive strategies.

TABLE I

COMPARISON OF *cost of transport* - COMPUTED AS TOTAL EXPENDED ENERGY, $\mathcal{E}$, NORMALIZED BY ROBOT MASS, m, AND DISTANCE TRAVELED, x.

	Manifold Appr.	Passive	Passive Period-2	Fast
$cot = \left(\frac{\mathcal{E}}{mx}\right)$	1.636	1.367	1.682	1.652

VI. DISCUSSION

A. Extension to on-line and continuous learning

In this paper, we described an off-line learning algorithm using a batch of observational data. However, the representation and algorithmic procedure can be adapted to on-line applications. To be precise, what we are assuming is that we are able to observe the dynamics of the robot as it performs randomized trials in an environment that induces a variety of different foothold conditions. The set of observations are being collected over many trials, each one ending in failure after some amount of stumbling around. The goal of the algorithm presented in this paper is to go from this level of (lack of) skill to a robust strategy that can achieve any foot placement goal within a suitably large interval. So, collectively, the observations provide a wide variety of foot placement conditions from which the algorithm generalizes to a strategy that can reliably achieve any other foot placement goal within a suitable interval. Moreover, the proposed algorithm is not affected by the presence of spurious data. This means that in a changing environment, a higher level process could easily discard stale data without impacting performance. Once the robot has a suitable corpus of data, sufficient to get it going, the set can be continually updated by logging all subsequent experiences as the robot actually walks - and continually updating all data structures at appropriate time intervals. Also, we use a nonparametric representation of the task manifold, as a graph. This has the benefit of enabling future improvements, such as multi-resolution algorithms that make tradeoffs between data density (hence approximation quality) and computational complexity.

B. Merging the benefits of analytical modeling and machine learning

Bipedal walking is one instance of a larger class of dynamically dexterous behaviors in high-dimensional, constrained, nonlinear systems. There are a number of other behaviors that we would like to achieve - ranging from dynamic object manipulation to full-body locomotion in tightly constrained spaces. In these general problem spaces, it is often the case that we do not understand all the dynamic effects well enough to be able to synthesize motion strategies using traditional model-based control design techniques. At the same time, model-free machine learning methods are still quite far from addressing such demanding problems, partly because the search spaces are too large and unconstrained. In this work, we have explored one approach to finding a middle ground, by merging a hand-crafted template strategy with a data-driven learning algorithm. In fact, our specific choices for template strategy and implementation of the learning algorithm may be modified without significantly affecting this larger argument.

VII. Conclusions

We propose a technique for trajectory generation that simultaneously addresses two goals - (i) incorporation of biologically inspired principles such as the passive/ballistic nature of human walking and (ii) planned walking with foot placement goals, such as on irregular terrain. We solve this motion planning problem without requiring detailed analytical models of the multi-link legged robot dynamics. Instead, we rely on a combination of data-driven learning and planning with a simpler abstract model. We demonstrate the utility of this approach using simulation results from a multi-link legged robot.

Acknowledgements

This work has taken place in the Intelligent Robotics Lab at the Artificial Intelligence Laboratory, The University of Texas at Austin. Research of the Intelligent Robotics Lab is supported in part by grants from the National Science Foundation (IIS-0413257 and IIS-0713150) and from the National Institutes of Health (EY016089).

References

[1] E. R. Westervelt, G. Buche, and J. W. Grizzle, Experimental validation of a framework for the design of controllers that induce stable walking in planar bipeds, *The International Journal of Robotics Research*, 24(6):559-582, 2004.

[2] A.G. Hofmann, M.B. Popovic, H.M. Herr, A sliding controller for bipedal balancing using integrated movement of contact and non-contact limbs, In *Proc. IEEE International Conference on Intelligent Robots and Systems*, 2004.

[3] K. Yamane, Y. Nakamura, Dynamics filter - concept and implementation of on-line motion generator for human figures, *IEEE Trans. Robotics and Automation*, 19(3):421-432, 2003.

[4] J.J. Kuffner, S. Kagami, K. Nishiwaki, M. Inaba, H. Inoue, Dynamically-stable motion planning for humanoid robots. *Autonomous Robots (special issue on Humanoid Robotics)*, 12:105-118, 2002.

[5] R.L. Tedrake. *Applied Optimal Control for Dynamically Stable Legged Locomotion*. PhD thesis, Massachusetts Institute of Technology, 2004.

[6] J. Morimoto, J. Nakanishi, G. Endo, G. Cheng, C.G. Atkeson, G. Zeglin, Poincare-map-based reinforcement learning for biped walking, In *Proc. IEEE International Conference on Robotics and Automation*, 2005.

[7] R. Chalodhorn, D.B. Grimes, G. Maganis, R.P.N. Rao, M. Asada, Learning humanoid motion dynamics through sensory-motor mapping in reduced dimensional spaces, In *Proc. IEEE International Conference on Robotics and Automation*, 2006.

[8] T. McGeer, Dynamics and control of bipedal locomotion, *J. Theoretical Biology*, 163:277-314, 1993.

[9] S.H. Collins, A. Ruina, R. Tedrake, M. Wisse, Efficient bipedal robots based on passive-dynamic walkers, *Science*, 307: 1082-1085, 2005.

[10] J. Pratt, C-M. Chew, A. Torres, P. Dilworth, G. Pratt, Virtual model control: An intuitive approach for bipedal locomotion, *Int. J. Robotics Research*, 20(2): 129-143, 2001.

[11] S. Ramamoorthy, B.J. Kuipers, Qualitative hybrid control of dynamic bipedal walking, In G. S. Sukhatme, S. Schaal, W. Burgard and D. Fox (Eds.), *Robotics : Science and Systems II*, pp. 89-96, MIT Press, 2007.

[12] G.A. Cavagna, P.A. Willems, M.A. Legramandi and N.C. Heglund, Pendular energy transduction within the step in human walking, *J. Experimental Biology*, 205:3413-3422, 2002.

[13] S. Mochon, T.A. McMahon, Ballistic walking: An improved model, *Mathematical Biosciences*, 52:241-260, 1980.

[14] A.D. Kuo, Energetics of actively powered locomotion using the simplest walking model, *J. Biomechanical Engineering*, 124:113-120, 2002.

[15] M.H. Dickinson, C.T. Farley, R.J. Full, M.A.R. Koehl, R. Kram and S. Lehman, How animals move: An integrative view, *Science*, 288:100-106, 2000.

[16] R.J. Full, D.E. Koditschek, Templates and anchors: neuromechanical hypotheses of legged locomotion on land, *J. Experimental Biology*, 202(23): 3325-3332, 1999.

[17] Y. Gutfreund, T. Flash, Y. Yarom, G. Fiorito, I. Segev, B. Hochner, Organization of octopus arm movements, *J. Neuroscience*, 16(22): 7297-7307, 1996.

[18] J. B. Tenenbaum, V. de Silva, J. C. Langford, A global geometric framework for nonlinear dimensionality reduction, *Science*, 290(5500):2319-2323, 2000.

[19] R. A. Peters, O. C. Jenkins, Uncovering manifold structures in robonaut's sensory-data state space, In *Proc. IEEE International Conference on Humanoid Robotics*, 2005.

[20] K. Tatani, Y. Nakamura, Reductive mapping for sequential patterns of humanoid body motion, In *Proc. Intl. Symp. Adaptive Motion of Animals and Machines*, 2003.