Gait Generation and Control in a Climbing Hexapod Robot

A. A. Rizzi[a], G. C. Haynes[a], R. J. Full[b], and D. E. Koditschek[c]

[a]The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA;
[b]Department of Integrative Biology, University of California, Berkeley, CA;
[c]Electrical & Systems Engineering, University of Pennsylvania, Philadelphia, PA

ABSTRACT

We discuss the gait generation and control architecture of a bioinspired climbing robot that presently climbs a variety of vertical surfaces, including carpet, cork and a growing range of stucco-like surfaces in the quasi-static regime. The initial version of the robot utilizes a collection of gaits (cyclic feed-forward motion patterns) to locomote over these surfaces, with each gait tuned for a specific surface and set of operating conditions. The need for more flexibility in gait specification (e.g., adjusting number of feet on the ground), more intricate shaping of workspace motions (e.g., shaping the details of the foot attachment and detachment trajectories), and the need to encode gait "transitions" (e.g., tripod to pentapod gait structure) has led us to separate this trajectory generation scheme into the functional composition of a phase assigning transformation of the "clock space" (the six dimensional torus) followed by a map from phase into leg joints that decouples the geometric details of a particular gait. This decomposition also supports the introduction of sensory feedback to allow recovery from unexpected event and to adapt to changing surface geometries.

Keywords: Legged Robots, Climbing Robots, Robot Locomotion, Gait Parameterization, Gait Regulation, Behavioral Control

1. INTRODUCTION

This paper addresses the design and regulation of robot climbing behaviors. Specifically, we explore how gaits can be parameterized to encode biologically inspired behavioral ideas, and how those parameters can be adjusted on-line to produce robust feedback based behavioral controllers capable of inducing an appropriately designed robotic mechanism to climb challenging surfaces. The issues that arise in acomplishing this include understanding how adjustments in gait parameters will impact behavior, and ensuring that the feedback based adjustments to behavior do not inadvertently result in invalid behaviors. To achieve this we decompose the control system into two distinct segments, the first directly controls gait parameters to regulate task level behaviors (e.g. modifying foot velocities to regulate contact forces), while the second layer of control supervises and limits the action of the first to maintain important structural qualities of the behavior are maintained (e.g. that five of the robots six feet remain on the ground during a pentapedal gait).

To focus this effort, we begin by restricting the class of behaviors that we will consider to "gaits." Broadly speaking, a gait provides an abstracted view of a particular legged locomotion strategy by specifying the cyclic motions each leg should take and how these individual leg motions should be related (in time and space) to one another. Such specifications of behavior are inevitably tuned or designed to for a relatively narrow set of operating conditions, and can only be expected to function reliably as long as actual operating conditions remain suitably near to the designed for conditions. One popular alternative approach, especially for climbing robots, would be to consider what are often referred to as "free gaits." these are more abstract specifications of behavior where specific motions or timing relationships between foot trajectories are left "free," and a higher level planning or control system is allowed to choose leg motion to produce the desired overall behavior. The result is a relatively complex constrained motion planing and control problem. Such an approach necessitates accurate models both for robot behavior and surface properties in order to properly constrain the planning and control problems, and typically finding such models can be very difficult in realistic operating conditions. As such, we prefer to focus on the slightly more generic idea of developing and controlling parameterized gaits, with feedback being used to make small behavioral adjustments in order to regulate overall behavior without being dependent on precise models for interaction between the robot and its environment.

Unmanned Systems Technology VIII, edited by Grant R. Gerhart, Charles M. Shoemaker, Douglas W. Gage,
Proc. of SPIE Vol. 6230, 623018, (2006) · 0277-786X/06/$15 · doi: 10.1117/12.666017

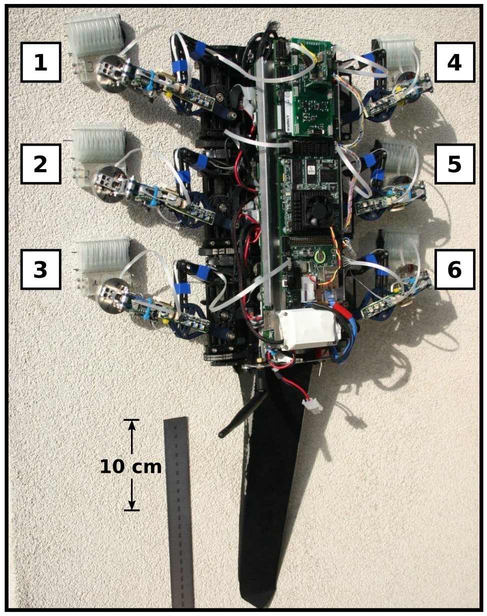


Figure 1. The RiSE robot. RiSE is a hexapedal robot capable of both climbing on a variety of vertical surfaces as well as locomoting over rough level terrain. RiSE utilizes compliant micro-spines on its feet to reliable attach to textured vertical surfaces, such as stucco, as seen here.

The overall goal is to develop a relatively intuitive system for describing and controlling behavior. The basic gait structure captures the important behavioral components of the behavior, describing how legs should move in order to cause feet to attach to the surface, produce traction forces, and detach from the surface. Simple feedback controllers then use the parameters that describe these gaits to provide a layer of feedback based control.

We will explore these concepts on the RiSE robot[1] (see Figure 1), where sensory feedback and gait regulation are used while climbing textured walls. The RiSE robot has twelve active degrees of freedom distributed evenly over its six legs, and makes use of passive compliant degrees of freedom in its legs and feet to help better direct and distribute the ground reaction forces generated while climbing. The feet consist of a large collection of micro-spines each compliantly anchored to the foot to facilitate attachment and load balancing.[2] Early empirical evidence suggested that the robot climbs more effectively when traction forces are evenly distributed among not only the spines on a single foot, but also when the loads are evenly balanced between the various feet that are in contact with the surface – a foot that carries too large a load is in danger of slipping (or damaging catastrophically its ground support material), whereas a foot with too little load is not likely to properly engage with the surface. So our focus will be on how we can use gait-based feedback strategies to evenly balance force among the robot's feet, while at the same time ensuring that the the robot continues to execute a stable pentapedal gait, suitable for climbing on these stucco surfaces.

1.1. Related Research

Ever since the development of the first walking mechanisms, robotics researchers have sought to create robots that exhibit complex, animal-like locomotion capability. At the same time many biologists have attempted to "reverse-engineer" the neuronal bases of locomotion.[3,4] One line of robotics development has attempted to realize these biological ideas by utilizing networks of simple reflexes and coordination schemes to locomote.[4–7] While these efforts have yielded complex behaviors, they have proven difficult to analyze or generalize to platforms other than the ones they were initially developed for. Thus it is difficult to conclude that these ideas have afforded a practical understanding of robot locomotion in general.

An alternative approach to locomotion has utilized deliberate and careful planning of every footfall that the system may make.[8,9] These methods are quite attractive as they explicitly attempt to take into consideration physical constraints and robot capabilities to find robot motions that will produce the desired overall behavior. However, to accomplish this, these methods require accurate models of the robot, accurate models of the environment, and accurate models that describe how the robot will interact with the environment. These models are required to allow the underlying reasoning or search systems to explore possibilities. The resulting systems tend to be computationally expensive, and their dependence on precise interaction models tends to make them extremely sensitive to small perturbations. Furthermore implementation of these types of strategies will additionally require relatively accurate sensors to reliably report robot state. All in all the implication is that these strategies will tend to result in systems that exhibit a level of fragility that will be problematic for deployment in realistic application settings.

The idea of using gait patterns to encode locomotion, however, has a number of features that make it attractive. The fundamental idea is, at its core simple, yet has been demonstrated to be suitable for encoding complex dynamics behavior.[10–12] A significant amount of research has focused on trying to understand good techniques for parameterizing gaits and selecting good behaviors from families of parameterized gaits. Much of this research has focused on artificial learning of effective gaits through automated tuning.[7, 13–15] Unfortunately, the results are again typically very robot specific, and little effort has been dedicated to finding parameterizations that are portable across multiple legged robots.

The question of how feedback can be utilized to control and coordinate legged locomotion has received less focus. However there are results that explore leg coordination mechanisms[16] and applications of these ideas have been used to make local adjustments to feedforward gait patterns[17] in order to improve performance. The work presented here goes beyond these ideas to actively adjust the gait itself, resulting in a large set of possible gait patterns that result in desirable overall robot behavior.

2. GAITS AND BEHAVIORS

2.1. Defining and Parameterizing Gaits

Throughout this paper we will use the term gait to indicate a cyclic motion pattern that produces locomotion through a sequence of foot contacts with the ground. in general gaits will be designed both to ensure support of the robot body, as well as producing ground contact forces that help propel the robot forward (or up) over a surface.

More specifically, gaits are motion patterns that can be described by functions that map from a phase space, $\mathbb{P}$, to the configuration space of a robot, $\mathbb{Q}$. Here we use the term phase in much the same way that we would typically use time, the major distinction being that time always advances with a constant velocity, while phase velocity can be altered by the system to control the rate at which a gait is executed. Gaits are required to be *cyclic*, so the domain, $\mathbb{P}$, is topologically equivalent to the unit circle, $\mathbb{S}^1$. Throughout this presentation we will use $\phi \in \mathbb{P}$ to denote the phase angle of the *master clock*, a centralized clock that we use to drive the generation of motion patterns in the RiSE robot.

If $\mathbb{G}$ is the space of all possible gaits, then a gait $g \in \mathbb{G}$ is a periodic, continuous, and injective function from phase angle to desired robot configuration, $g : \mathbb{P} \rightarrow \mathbb{Q}$. By requiring that g be injective we ensure that g is invertible over its range. On a robotic hexapod with actuators located only on the legs, the configuration of the robot is naturally thought of as the Cartesian product of individual configuration spaces for each leg. If $\mathbb{Q}_i$ is the configuration space of leg i then $\mathbb{Q} = \mathbb{Q}_1 \times \mathbb{Q}_2 \times \mathbb{Q}_3 \times \mathbb{Q}_4 \times \mathbb{Q}_5 \times \mathbb{Q}_6$. Frequently, it is useful to think of a gait as a collection of separate functions, $g_{1-6}(\phi)$, where $g_i(\phi)$ captures the motion of leg i.

Given this decomposition, natural parameters emerge from g_i. One natural distinction is made between a leg that is in contact with the ground, termed *in stance*, and a leg that is recirculating in the air, *in flight*. The phase angle at which a given leg begins stance is called the *stance phase offset*, which we denote $\rho_i \in \mathbb{P}$. For a given ρ_i, $g_i(\phi)$ can be normalized, producing a related function $(g_{n,i})$ for which $\phi = 0$ corresponds to a leg beginning stance, $g_i(\phi) = g_{n,i}(\phi - \rho_i)$. This normalization is useful since typically, the $g_{n,i}$ mappings are similar or identical for multiple legs, and the stance phase offsets are readily accessible parameters, decoupled from the geometric structure of the gait.

Stance phase offsets are temporal parameters, and suggest a structure to the timing of a gait. Multiple legs with the same stance phase offset make contact with the ground at the same time, whereas differing offsets dictate a particular order in which legs make contact. It is worth noting that gaits are actually defined by the *phase differences* between legs, so representing the gait for a hexapod like RiSE with six phase distinct phase offsets actually over-parameterizes the behavior. Two gaits with phase offsets a fixed value apart are in fact equivalent.

2.2. Gaits as a Basis for Behaviors

Unlike gaits, which are constrained to producing only periodic motion, behaviors can include non-cyclic motion and thus may be arbitrarily complex. While there have been a variety of approaches to encoding complex behaviors on legged robots, the use of gaits as a basis allows the layering of relatively simple feedback systems on top of existing gait patterns, allowing the feedback system to inherit capabilities that are developed at the gait level. Essentially, we are asserting that a behavior that uses a specific gait as its basis builds upon that same style of locomotion, provided the behavior only makes small local adjustments to the gait parameters, keeping the now time evolving gait relatively "close" to the original gait in the space of all gaits, $\mathbb{G}$.

To encode a gait-based behavior, we design controllers (feedback strategies) that output a differential change of gait. Formally, this is a continuous function that maps from some set of inputs, $\mathbb{U}$, to the tangent space of gaits $b : \mathbb{U} \to T\mathbb{G}$. As an example, we will use the stance phase offsets, $\rho_i \in \mathbb{P}$, as one relatively simple method of parameterizing gaits. A set of six phases offsets, therefore, provides a useful low dimensional parameterization of the full space of gaits.

The gait-based behavioral controller we will explore in this paper seeks to control interaction forces at the feet of RiSE by controlling the phase offset velocities in a particular gait that is well adapted to climbing hard rough surfaces such as stucco. The specific controller takes the form

$$\dot{\rho}_i(t) = b_{f,i}(t) + b_{c,i}(t), \tag{1}$$

where $b_{f,i}$ encodes the desire to balance forces between feet that are in stance, while $b_{c,i}$ encodes the desire to regulate the phase relationships between the legs to ensure that the structure of the particular gait is maintained. The details of these two terms will be developed in Sections 3 and 4 respectively.

3. FORCE-BALANCING CONTROLLER

Locomotion necessitates the development of forces between the robot and the ground. Controlling the temporal structure of these interaction forces is the central problem in producing desirable locomotion behavior. If we presume that we have available to us a specification of desired interaction forces, either from analysis, empirical exploration, or possibly taken from biological inspiration, then we can begin to ask the question of how we can best control the specific actions of the robot in order to best produce such a desired set of interaction forces. By comparing a particular foot's actual measured force against such a desired force profile, we can understand how effectively a foot is being used in a gait. The control system developed in this section seeks to modify parameters in a gait to achieve a better match between actual and desired forces in stance feet.

Specifically, for RiSE executing climbing gaits we will seek to balance the traction load evenly among the feet that are making contact with the surface. This has the effect of minimizing the risk of slipping and improving overall progress up the surface.

3.1. Desired Forces

Before we can consider how to balance forces among feet, we need to develop some intuition about what desirable foot forces are for climbing. In a nominal steady state, each gait has a characteristic set of ground reaction forces. In the case of a climbing robot, we are most interested in foot traction, the force felt in the fore-aft direction, or the robot's direction of travel, as this gives us an idea of the weight that each foot is actually carrying. Furthermore, this force vector is aligned with the direction that legs move while in stance move as they propel the robot up a surface.

While it is relatively natural to think about the desired traction force a foot should generate during stance, we must also take into account the period of loading before stance, *attachment*, and unloading after stance,

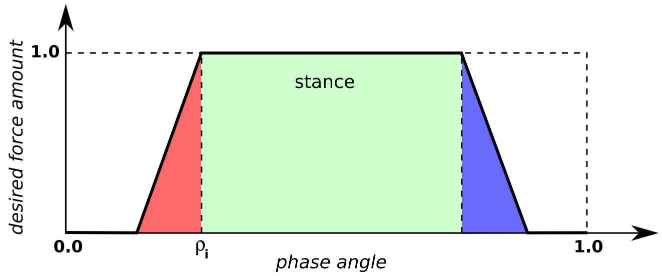


Figure 2. The desired force profile, $d_i(\phi)$. During stance, the area lightly shaded, a foot should carry 100% of its desired force, whereas during flight, it should not carry any traction load. Attachment and detachment involve adding and removing load, over some finite period.

detachment. Thus, there is a profile that describes the desired percentage of total force that a foot should experience while in contact with the surface. Figure 2 shows the relatively simplistic force profile, denoted $d_i(\phi)$, we use to capture this idea. Each foot carries no load during flight, and begins to carry full load starting at its stance phase offset angle, ρ_i.

3.2. Computing an Average Force

In order to determine if a foot is carrying too much or too little force, we must understand how to calculate a total force load that can be apportioned among the stance feet. This must take into account the fact that feet load and unload at different times in a gait pattern. Specifically, when calculating an "average foot force," we will only consider forces generated by feet that are in contact with the surface. If $f_{m,i}$ is the measured traction force for a given foot, then

$$f_i = \begin{cases} 0 & \text{if leg } i \text{ is in flight} \\ f_{m,i} & \text{if leg } i \text{ is not in flight} \end{cases} \tag{2}$$

simply ignores data from the foot while it is in flight.

Using (2) to sum forces over all feet yields the total traction force being generated by the robot. We must now decide how to apportion this load. A foot that is in stance counts as a full unit – if a robot has three feet in stance, the total force should be divided by three. A foot that is in attachment or detachment, however, is only meant to be carrying a partial amount of force, therefore it counts as only a partial unit. A robot with three feet in stance, and one foot that is halfway done attaching should be averaged over 3.5 feet, not four. The function $d_i(\phi)$ provides us with the perfect method of weighting the feet to compute the "expected average" foot load.

$$a = \frac{\sum_{i=1}^{6} f_i}{\sum_{i=1}^{6} d_i(\phi)} \tag{3}$$

We can now compute the force we expect from a foot in full contact with the ground. To compute the desired force, in Newtons, we simply multiply (3) by $d_i(\phi)$, yielding a desired force for each foot, $f_{d,i} = d_i(\phi)a$.

3.3. Gait-Level Force Control

Given measurements of a foot's traction force and a specification for the desired traction force, as determined by the average force calculation above, we can now begin to construct a gait-based behavioral controller that computes parametric changes to the current gait in order to regulate the traction force generated at each foot.

For a tradition feedforward gait, the phase offsets are all constant, thus as a consequence the phase offset velocities are always zero. Our feedback strategies will modify this by actively modifying the phase offsets by commanding non-zero phase offset velocities. Given our conventions, a positive phase offset velocity results in delaying a given leg relative to the other legs, whereas a negative phase offset velocity speeds up the leg relative to its peers.

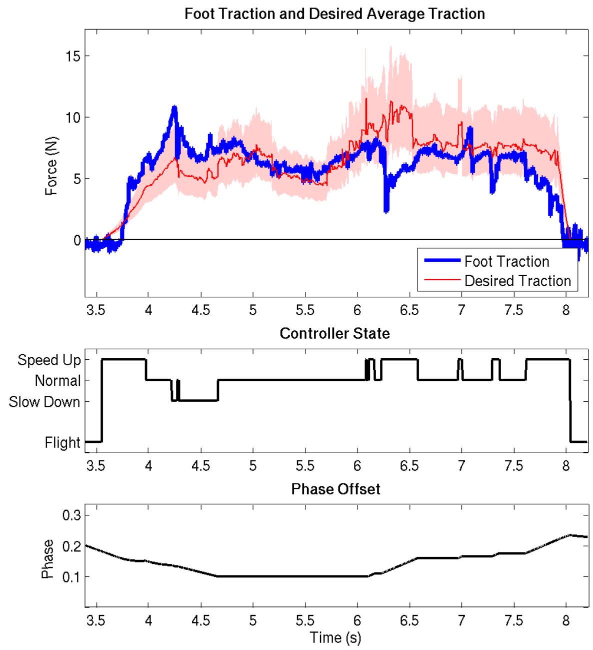


Figure 3. Results with the gait-based force controller from (4). The top plot shows the actual traction force for a given foot, as well as the desired force, computed from the average traction force, with a shaded deadband region. Note how the force follows a profile similar to that in Figure 2. The middle plot shows the action taken by the controller, and the bottom plot shows the resulting change to the leg's stance phase offset, ρ_i.

the obvious simple control strategy to consider would simply compare the actual traction force to the desired traction force, and speed up or slow down a leg as necessary to align the two. This relies on the tacit assertion that we understand the relationship between phase offset velocity and traction force – for a leg that is in stance on RiSE a negative phase offset velocity will cause the leg accelerating through stance and carrying more force, while a positive phase offset velocity will slow the legs progress through stance and reduce the traction force it produces. Using this idea we can construct a controller, with gain k_p,

$$b_{f,i}(i) = k_p \tilde{f}_i, \tag{4}$$

where $\tilde{f}_i = f_i - f_{d,i}$.

In practice on the RiSE platform, due to actuator constraints and sensing limitations, we chose to implement a discrete version of this controller, using deadbands and saturation functions to determine when to speed or slow individual legs, and limiting the rate at which they are advanced or retarded. Figure 3 shows the action of this controller from a typical experiment on the RiSE robot. As can be seen, when the traction force exceeds the desired range, the controller slows the leg down until the force is suitably reduced. Later in stance, when the traction force is below the target range, the controller advances the lag to take more climbing load.

4. GAIT REGULATION

The control system developed in 3.3 adjusts the gait, by modifying the stance phase offsets, to better control the balance of traction forces produced among the legs. However the controller proposed does this with out regard to the actual physical configuration of the legs, and risks adjusting the legs such that the resulting behavior no longer climbs reliably. For example you can imagine legs being advanced and retarded in such a way that the resulting gait specifies that the robot should simultaneously lift all the legs on one side of the robot, or worse yet lift all the legs at once. Obviously these are unacceptable gait patterns, and the control system should actively avoid driving the system to such configurations.

In this section, we introduce the concept of *gait regulation*, and will construct a coordination mechanisms that acts between legs to ensure that these undesired situations do not occur.

4.1. Understanding Pentapedal Gaits

For simplicity we will limit our discussion here to one particular class of gaits, that is pentapedal gaits. For a climbing robot like RiSE, this class of gaits is particularly attractive for a number of reasons. Foremost, pentapedal gait patterns keep all six legs of a hexapod maximally out-of-phase with one another, and with a suitably high duty factor will always having five legs in contact with the ground while only recirculating a single leg at a time. Thus nominally the stance phase offsets will be evenly spaced out, by sixths, on the circle. This leaves free the choice of leg sequencing, combinatoric analysis allows us to conclude that there are $(6-1)! = 120$ different ways to order the legs, resulting in 120 distinct pentapedal gaits, with no appreciable difference in performance between them.

We care about this aspect of the gait structure, since as a leg's phase offset is modified by the control strategy in (4), it is possible for one leg to leapfrog another, switching their order in the gait. Thus the gait being executed actually changes from the original gait to another of the 120 possible pentapods.

4.2. Leg Coordination

Our goal is to construct a second simple control strategy that will result in making all of the 120 possible pentapedal gaits stable attractors for the robot. While previous work in leg coordination has dealt primarily with sets of decentralized clocks,[16,17] our use of a single master clock and six phase offsets is actually equivalent. So, rather than coordinating multiple clocks, we coordinate the phase offsets that are used to derive the individual clocks from the single central clock.

Our goal to keep the legs maximally out-of-phase with one another suggests the use of a fully-connected coordination network. Applying a virtual "force" derived from a network of cosines is one natural strategy to realize this behavior. The resulting relationship takes the form

$$b_{c,i}(t) = \sum_{j=1}^{N} c_i(j) \tag{5}$$

where

$$e_{i,j} = (\rho_i(t) - \rho_j(t)) \mod 1.0 \tag{6}$$

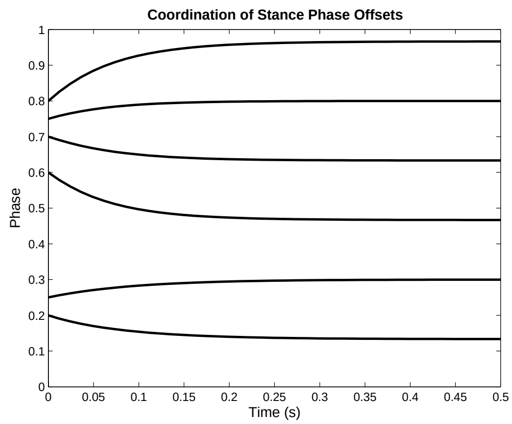
$$c_i(j) = \begin{cases} 0 & \text{if } i = j \\ cos(\frac{1}{2}(2\pi e_{i,j})) & \text{if } i \neq j. \end{cases} \tag{7}$$

$$\tag{8}$$

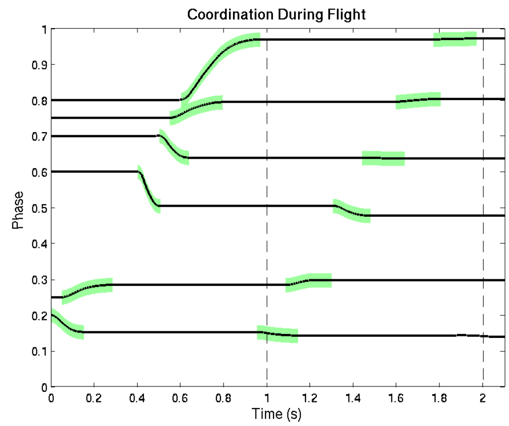
As above, a positive phase offset velocity will retard the progress of a given leg, while a negative value speeds it up. The coordination strategy defined by (5) basically seeks to force an arbitrary number of entities to be maximally out-of-phase with one another, and is zero-valued at stable configurations, including all of the $(N-1)!$ possible orderings. Figure 4(a) shows an example of this coordination strategy operating on six simulated legs.

4.3. Combining Gait Regulation with Feedback

At this point we have two independent strategies, one for regulating traction forces, and the other for regulating the structure of the gait in the face of phase offset disturbances that may be injected by the traction load balancing control system. We now seek to integrate these two ideas into a unified control system that both balance traction forces while stabilizing the overall gait structure. At the highest level, we will achieve this by using the behavior from section 3 during stance, and the coordination controller described above only during flight. The intent is to ensure force balance between the feet during stance to improve climbing performance while



(a)



(b)

Figure 4. (a) A network of cosines based coordination strategy repels all phase offsets away from each other, quickly stabilizing in a pentapedal gait.

Figure 5. (b) When allowing coordination to occur only during flight, the shaded regions, multiple strides are required to return to a pentapedal gait. Note that recirculation still proceeds one leg at a time.

using the flight phases to resynchronize the feet and guarantee that they at least begin stance well separated from one another. Using a simple weighting function to reflect whether a leg is in flight or stance, we can construct a modified gait regulation controller,

$$b_{c,i}(t) = (1 - d_i(\phi)) \sum_{j=1}^{N} c_i(j). \tag{9}$$

This makes use of the weighting function shown in Figure 2, which, when subtracted from 1 is maximal during flight and zero during stance. Figure 4(b) shows the effect of this strategy in simulation.

5. EXPERIMENTAL RESULTS

We have applied these gait-based behavioral control strategies on the RiSE robot. Through these experiments we have observed that the proposed strategies do in fact improve control over the traction forces generated during climbing behaviors when compared to just open-loop gait execution strategies.

5.1. The RiSE Robot

The RiSE robot, shown in Figure 1, has a total of twelve degrees of freedom, two in each of the six legs. The *wing* joint lifts and lowers a foot onto the climbing surface, while the *crank* joint moves a foot up and down along the surface and is primarily responsible for lifting the robot as it climbs. RiSE has a body length, not including the tail, of 31 cm and a total weight of 2.6 kg. RiSE is computationally and power autonomous, runs on a 266 MHz processor, and is capable of operating without a tether for extended periods of time. Motor level control is performed at a frequency of 300 Hz, while a human operator issues task-level commands to the robot via a laptop computer, connected over an 802.11b wireless connection.

in order to climb on flat textured surfaces (such as stucco), RiSE incorporates feet that contain large collections of individually-compliant microspine toes, as shown in Figure 6, each toe is capable of carrying only a small fraction of the total robot weight weight, however when a sufficient number of spines engage the surface the robot is able to climb.[2] Furthermore, given the total weight of the robot it is important that the total climbing force be distributed among the stance feet to avoid slip, failure of spines, or localized failure of the climbing surface itself. To facilitate the control strategies described here, each leg incorporates a set of force sensing elements, early prototypes of which can be seen in Figure 6. These sensors allow the robot to measure the traction force felt by each foot while climbing.

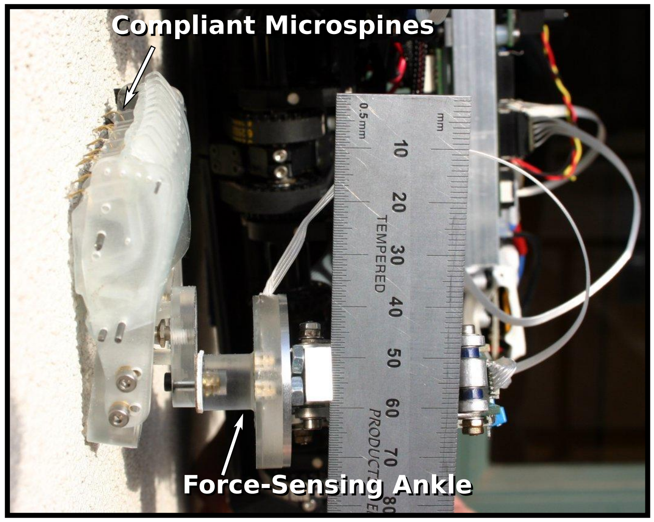


Figure 6. Photo of a RiSE foot and ankle engaged on a stucco surface. Each microspine is individually compliant, to catch onto surface asperities. A force sensing device measures the forces felt at the ankle.

The spatial trajectory of the gait we will use for all experimentation was tuned empirically prior to our layering of feedback capabilities onto it. During attachment, each foot is presented flat onto the surface, then pulled downward to allow the spines to engage on surface asperities, the foot then slows slightly as it continues through stance, slowly pulling the robot upwards on the wall, until detachment, when the foot reverses direction to unload the spines before being lifted and recirculated to begin a new attachment sequence.

5.2. Experimental Methodology

All experimental data reported here was acquired while the robot was climbing on a stucco-like surface, painted onto a wall-mounted board using supplies found at most hardware stores. The wall surface allows for easy modifications of the climbing angle, between 45 degrees and vertical. To minimize variation between experiments due to degradations in either spine sharpness or in surface quality, the feedforward and feedback behaviors were run alternately for a total of six trials, three runs apiece. Each experiment lasted approximately 1 minute, and involved the robot climbing a distance of approximately 1 body length.

Data was collected from the force-sensors and was simultaneously used as an input to the behavioral controller, as well as logged for analysis of the experiments. Key quantities were computed from each experiment, allowing easy comparison between feedforward and feedback behaviors. Box-and-whisker plots are used to describe visually the distribution of these calculated values.

The particular gait used in these trials was just one of the 120 possible pentapods, and the legs necessarily followed the same periodic motions as it climbed. The feedback behavior used the same spatial trajectory as the feedforward gait and began at the same particular version of the pentapod, but it was able to modify the stance phase offsets both to balance forces and regulate the structure of the gait to maintain a pentapedal structure. The end result is that the feedback behavior may actually switch between particular instances of the 120 pentapods as it climbs.

5.3. Analysis of Average Traction Forces

The first quantity we analyzed was the average force felt during stance, per foot and per experiment. Studying the distribution of these forces allows us to estimate the variation between legs and experiments. Since it is desired for legs to not carry too much or too little load, a tighter distribution implies better performance.

After collecting data from each experiment, these average values were pooled together, split up by leg type, and grouped for all. Figure 7 shows the results. It is useful to note that the overall average traction force, among all legs in both feedforward and feedback cases, is about the same, 6.5 N. This is roughly equal to body mass, in

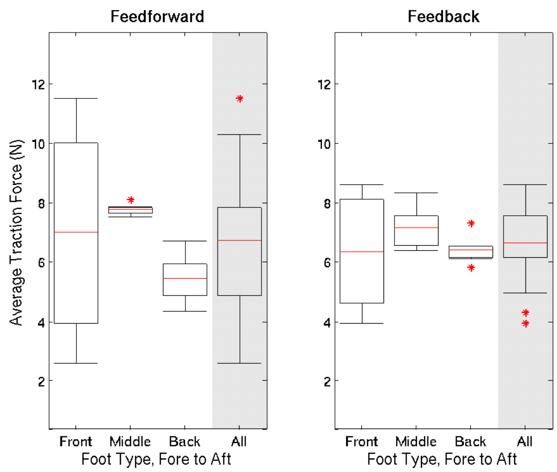


Figure 7. Average forces during stance felt by feet. The left diagram shows box-and-whisker plots describing the variation among front foot, middle, and back foot measurements. Shaded is the overall distribution of measurements. The diagram on the right shows the same for the feedback behavior. Note that the distribution under the influence of the feedback behavior, is tighter than that of the feedforward gait.

Newtons, divided among an average of 5 legs. The distribution of these two sets, however, is the more interesting quantity to examine. In the feedforward case, the variation of forces among the legs is about twice as large as the feedback behavior, suggesting larger disparities in force between the various legs of the robot.

5.4. Analysis of Force Signals

While studying the variation of the average stance traction force is useful, the quantity we really want to note is the difference from a nominal force profile. The goal of the coordinated feedback behavior is to drive a foot's force closer to some nominal force profile. We compute a single value to note this difference by means of the root-mean-square, shown for discrete measurements in (10). Values closer to 0 indicates a closer correlation between the two signals.

$$e_{rms,i} = \sqrt{\frac{\sum_{k=1}^{n} \tilde{f}_i^2[k]}{n}} \tag{10}$$

$$\tilde{f}_i[k] = f_i[k] - f_{d,i}[k] \tag{11}$$

Figure 8 shows the results of this analysis, collected from multiple experiments, and again pooled by leg type. As can be seen, the root-mean-square error observed under the control of the feedback behavior is approximately 30% less than that of the open-loop behavior. The implications is that the feedback strategies are in fact significantly improving that ability of the robot to track a specific desired traction force profile during difficult climbing.

6. CONCLUSION

Given an intuitive understanding of the basic properties of a class of legged robot gaits, we have presented a method for applying feedback control directly to the space of gaits, while building upon existing feedforward gaits. The resulting task-level feedback allows the robot to control ground reaction forces while locomoting over difficult and poorly modeled terrain. A gait regulation strategy is then used to coordinate the motion of the legs and keep the robot within a suitable range of possible gaits.

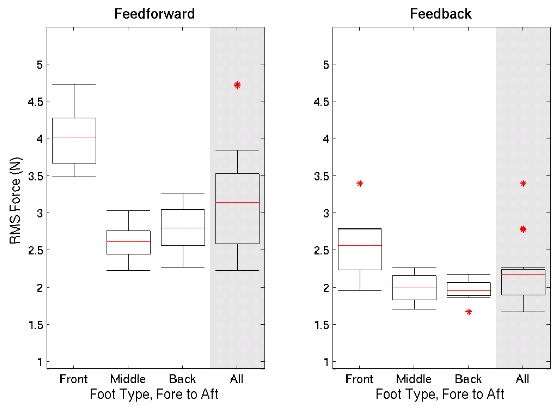


Figure 8. Analysis of the root-mean-square traction force error. In the feedback case, error is dramatically reduced, by approximately 30%, suggesting that the feedback behavior is better at following a desired force profile.

In empirical tests, the two separate control systems, one to balance traction forces between the feet and the other to perform gait regulation, have been demonstrated in operation on the robotic hexapod RiSE. Preliminary experiments suggest that the feedback behavior exhibits a large improvement over a simple feedforward gait at performing an identical task, climbing a stucco-textured wall.

From here there are a number of open problems to address. First we would like to generalize the gait coordination strategies to apply across a range of gaits, rather than only apply to pentapods. For Rise we are specifically interested in tetrapedal and tripod gaits which can afford significantly faster climbing. Furthermore, we are continuing to explore parameterizations of gaits to perform feedback on more than just traction force, to more explicitly make use use of lateral and normal force measurements in an effort to better understand and utilize the ground reaction force patterns that we observe in biology. Lastly, in an effort to demonstrate the generic nature of these ideas, we are interested in applying the strategies outlined in this paper to robots of different morphologies, including quadrupeds.

ACKNOWLEDGMENTS

This work is supported in part by the Defense Advanced Research Projects Agency within the DSO Biodynotics Program under contracts DARPA/SPAWAR N66001-03-C-8045 and DARPA/SPAWAR N66001-05-C-8025.

REFERENCES

1. K. Autumn, M. Buehler, M. Cutkosky, R. Fearing, R. J. Full, D. Goldman, R. Groff, W. Provancher, A. A. Rizzi, U. Saranli, A. Saunders, and D. E. Koditschek, "Robotics in scansorial environments," *Unmanned Ground Vehicle Technology VII* **5804**(1), pp. 291–302, SPIE, 2005.
2. A. T. Asbeck, S. Kim, M. R. Cutkosky, W. R. Provancher, and M. Lanzetta, "Scaling hard vertical surfaces with compliant microspine arrays," in *Proceedings of Robotics: Science and Systems*, (Cambridge, USA), June 2005.
3. D. M. Wilson, "Insect walking," *Annual Review of Entomology* **11**, pp. 103 – 122, 1966.
4. H. Cruse, "What mechanisms coordinate leg movement in walking arthropods?," *Trends in Neurosciences* **13**, pp. 15 – 21, 1990.
5. R. A. Brooks, "A Robot That Walks; Emergent Behaviors from a Carefully Evolved Network," memo 1091, MIT AI Lab, February 1989.
6. K. S. Espenschied, R. D. Quinn, H. J. Chiel, and R. D. Beer, "Leg coordination mechanisms in stick insect applied to hexapod robot locomotion," *Adaptive Behavior* **1**(4), pp. 455 – 468, 1993.
7. M. Lewis, A. Fagg, and G. Bekey, "Genetic algorithms for gait synthesis in a hexapod robot," 1993.

8. D. Wettergreen, H. Thomas, and C. Thorpe, “Planning strategies for the ambler walking robot,” in *IEEE Int. Conf. on Systems Engineering*, pp. 198 – 203, August 1990.
9. J. Chestnutt, M. Lau, G. Cheung, J. Kuffner, J. K. Hodgins, and T. Kanade, “Footstep planning for the honda asimo humanoid,” in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, April 2005.
10. R. Altendorfer, N. Moore, H. Komsuoglu, M. Buehler, H. B. Brown, D. McMordie, U. Saranli, R. Full, and D. Koditschek, “Rhex: A biologically inspired hexapod runner,” *Autonomous Robots* **11**, p. 207, 2001.
11. S. Schaal and C. G. Atkeson, “Open Loop Stable Strategies for Robot Juggling,” in *International Conference on Robotics and Automation*, **3**, pp. 913–918, (GA, Atlanta), 1993.
12. J. G. Cham, S. A. Bailey, J. E. Clark, R. J. Full, and M. R. Cutkosky, “Fast and robust: Hexapedal robots via shape deposition manufacturing,” *International Journal of Robotics Research* **21**(10), 2002.
13. J. D. Weingarten, G. A. D. Lopes, M. Buehler, R. E. Groff, and D. E. Koditschek, “Automated gait adaptation for legged robots,” in *IEEE Int. Conf. on Robotics and Automation (ICRA)*, **3**, pp. 2153 – 2158, 2004.
14. S. Chernova and M. Veloso, “An evolutionary approach to gait learning for four-legged robots,” in *In Proceedings of IROS’04*, September 2004.
15. V. Zykov, J. Bongard, and H. Lipson, “Evolving dynamic gaits on a physical robot,” in *Late Breaking Papers at the 2004 Genetic and Evolutionary Computation Conference*, M. Keijzer, ed., (Seattle, Washington, USA), 26 July 2004.
16. E. Klavins and D. Koditschek, “Phase regulation of decentralized cyclic robotic systems,” *The International Journal of Robotics Research* **21**(3), 2002.
17. J. D. Weingarten, R. E. Groff, and D. E. Koditschek, “A framework for the coordination of legged robot gaits,” in *IEEE Int. Conf. on Robotics, Automation and Mechatronics (RAM)*, **2**, pp. 679 – 686, 2004.