2010 IEEE International Conference on Robotics and Automation
Anchorage Convention District
May 3-8, 2010, Anchorage, Alaska, USA

Constrained Convergent Gait Regulation for a Climbing Robot

Salomon Trujillo, Barrett Heyneman and Mark Cutkosky
Center for Design Research, Stanford University
Stanford, CA 94305–2232, USA
Email: sjtrujil@stanford.edu

Abstract— **The priorities of a climbing legged robot are to maintain a grasp on its climbing surface and to climb efficiently against the force of gravity. These priorities profoundly constrain the choice of gait regulation methods. We propose a gait regulation and analysis method that varies foot detachment timing, effectively modifying stride length and frequency in order to maintain gait phasing, subject to kinematic and stability constraints. The method results in linear equations, leading to straightforward tests for local and global convergence when, for example, disturbances such as foot slippage cause departures from the nominal phasing. We illustrate the procedure with an example involving a bounding gait and compare it with empirical results obtained on the RiSE climbing robot.**

I. INTRODUCTION

In comparison to walking or running robots, vertically climbing robots face severe requirements on stability and power consumption. They must maintain contact forces at all times to prevent falling and they must ensure that motors do not overheat when they are climbing rapidly or applying large forces. Therefore, we seek a gait regulation method for climbing robots that will produce convergence to a standard free gait [1], [2], where rhythmic phasing between legs is formulated as an objective, subject to constraints.

Stability typically requires that certain combinations of legs remain attached at various times during a stride. In addition, the legs are subject to kinematic constraints (e.g., on maximum and minimum joint angles). When kinematic and stability constraints conflict, it may be necessary for the robot to pause while other legs are brought into contact. In parallel with the stability and kinematic constraints, a climbing robot is subject to constraints on the maximum sustained torques that its motors can produce without overheating. In an earlier study [3], we found that the following approach would minimize motor heating:

- *Stance:* External forces should be distributed evenly between as many legs as possible, to reduce the maximum heat produced in the motors of any leg.
- *Swing:* The legs should recycle as fast as possible. This approach reduces the total heat produced, when averaged over a full stride period.

Taken together, stability and force distribution requirements govern our choice of gaits and gait regulation methods. Stance calls for a force-controlled approach in order to balance the load between legs and apply the forces needed for grip maintenance. Swing calls for a time-optimal bang-bang controller such that minimal time is spent in swing. Using these control methods during stance and swing rules out several established methods of gait regulation. However, we still have the freedom to vary attachment and detachment timing. In order to provide the longest potential stride for each leg, we maintain leg attachment at the earliest point possible in the leg's trajectory. This choice leaves us with the ability to vary leg detachment, effectively altering the stride length and period. In this way, lagging legs will experience faster stride frequencies, with slower frequencies for leading legs, until gait phasing is fully distributed.

Under nominal conditions, we want a gait with evenly spaced footfalls and no pausing. Under difficult conditions, disturbances will alter the phasing between legs and may cause the robot to stop forward movement in order to safely cycle its legs. By implementing a standard gait, the robot maximizes the phase differences between legs for which kinematic and stability constraints may conflict, granting the robot leeway for disturbance rejection. In this article, we will analytically show that our algorithm, for certain parameter choices, has global convergence and has no undesirable local minima. This analysis can both indicate if a set of gait parameters is globally convergent and give estimates to the rate of convergence. We have implemented our algorithm on a quadruped variant of the RiSE robot V2 [4] and use it to illustrate our approach.

A. Previous Work

The literature for gait regulation in the context of legged robots is extensive [5], [6], [7], [8]. It is undesirable to simply encode position trajectories into the robot, as that method would produce internal forces and reduce the ability of the robot to handle perturbations [9]. One solution is to introduce adjustable clocks [10]. This technique works well but requires the gait regulator to have control authority during recirculation. Another strategy is to use central pattern generators where the gait behavior emerges from a set of coupled oscillators [11]. These systems can be difficult to construct when a particular behavior is desired, but they have inspired a series of methods that seek to modify the behavior of legs based on behavior of neighboring legs [12], [13]. This approach modifies various parameters of the gait, including the "posterior extreme position," which is analogous to modifying detachment timing in the approach that we present here.

Unlike these previous approaches, we employ a centralized controller that monitors the phases of all legs and triggers detachments. To this end, we introduce a new controller,

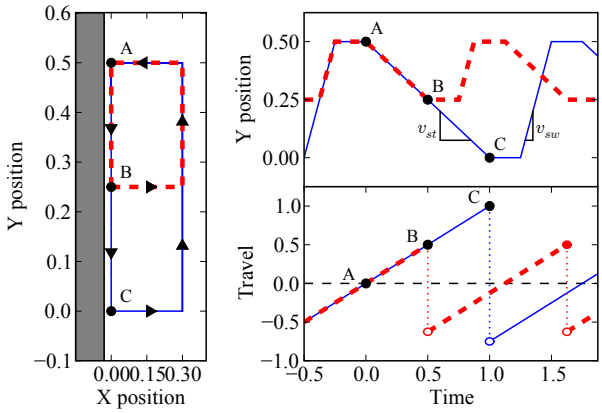

Fig. 1. Left: Two simplified leg trajectories for one leg of a climbing robot (grayed surface indicates the wall). The blue solid trajectory has an intentionally longer stride and period than the red dotted trajectory. Upper-right: Stroke position vs. time. Lower-right: Leg travel, s, vs. time. (A) Foot attachment event for both trajectories. (B) Foot detachment event for the short trajectory. (C) Foot detachment for the long trajectory. Note the constant slope of $s(t)$ and discontinuities at detachment events.

a new gait representation and a novel analog for phase called *travel*, which seeks to resolve mathematical difficulties when describing the instantaneous phase of cycles with time-varying periods.

II. ALGORITHM IMPLEMENTATION

A. Travel

We begin our discussion by presenting our definition of travel. Due to the cyclic nature of legged locomotion, phase can be used a generalized variable for leg position along a prescribed trajectory and is subsequently treated with the same notation as rotation [14]. Since our method has a continually changing period, it becomes difficult to describe the state in terms of angles or ratios. Instead, we describe travel, s, in units of time as a measure of leg position, $\vec{\theta}$, in joint space. This approach is similar to the phase coordinates of intermittent systems [15], with the primary difference being that we do not normalize the travel metric to 2π as is common with phase. Using desired leg trajectories, f, parameterized in terms of time, we calculate travel from the inverse of these trajectory functions:

$$\vec{\theta}_{desired} = f(t) \quad \Rightarrow \quad s(t) = f^{\dagger}(\vec{\theta}_{actual}) + c \qquad (1)$$

where c is a constant used to align the travel to a fixed origin independent of time. To simplify the mathematics, we leave travel in units of time and do not divide by the stride period. The purpose of this operation is to find the estimated time of arrival for every point along a single leg's joint-space trajectory curve. During normal operation, the time-derivative of travel is constant, $\frac{ds}{dt} \approx 1$, but when a disturbance is encountered, the position of the leg and not the elapsed time determines the perturbation to travel. Since the legs are operating using methods such as force control or bang-bang control and not position control, such perturbations arise frequently.

Fig. 1 shows a schematic representation of a leg trajectory (abstracted here to a rectangular loop for ease of depiction) and the corresponding plots of Y position (stroke along wall) and travel as functions of time. We fix the origin, $s = 0$, at the attachment point, which can be achieved by choosing the appropriate c. This choice provides a convenient notation where positive travel represents a leg in stance while negative travel represents a leg in swing. In theory, the travel function experiences no discontinuity during the attachment event. The detachment point, as shown in Fig. 1, occurs at variable points in state-space and produces a large discontinuity as the travel jumps from the positive stance travel to the negative swing travel.

We model the change in travel at detachment as a linear equation solely in terms of travel. We assume the durations of detachment and acceleration to swing, as well as the deceleration from swing and attachment, are constant and thus we represent their combined time as δ. Nominal stance and swing velocities (v_{st} and v_{sw}) are fixed as a property of the gait. If a leg that attached at time 0 detaches at time t_d, it has traveled a distance of $v_{st}t_d$ and thus the time it will take to return to the attachment point is $t_d \left|\frac{v_{st}}{v_{sw}}\right| + \delta$. For ease of notation we define the ratio between stance and swing velocities as $\gamma = \frac{v_{st}}{v_{sw}}$. We can therefore express detachment as a discontinuous jump in travel:

$$s(t_{d+}) = -\gamma s(t_{d-}) - \delta \qquad (2)$$

Building upon the description of travel for a single leg, we represent leg configurations for the entire robot using a vector of travels in which the travel of the i^{th} leg is s_i.

B. Gait Regulation (GR) Rule

The Gait Regulation rule (hereafter, the GR rule) governs leg detachment. When the weighted average of all the leg travels exceeds a prescribed trigger point, the leg with highest travel should detach. For gaits such as the alternating tripod of a hexapod, a detachment event triggers simultaneous detachment of a set of legs. In order to maintain symmetry with each step, the weighted average is applied to a sorted list of leg travels. We find a permutation of travels that reorders them in descending order: $s_i^* = s_j \; st. \; s_1^* \geq s_2^* \geq \ldots \geq s_n^*$. The GR rule is written as follows:

$$\sum \omega_i s_i^* \geq T \qquad (3)$$

The GR rule is unique up to a normalization factor; the analysis in this paper uses $\sum w_i = 1$, though other algebraically equivalent normalizations exist. An appropriately chosen GR rule produces rhythmic behavior upon convergence of the gait and distributes the footfalls to maintain sufficient space for accommodating disturbances. However, while it regulates the interactions among legs, it does not prevent the robot from falling. Thus, the GR rule should be preempted by constraints that ensure the stability of the robot.

C. Configuration Constraint (CC) Rules

Under various conditions, the GR rule may fail to trigger a detachment before a leg is physically unable to continue its

trajectory, or it may trigger one that would cause the robot to lose stability. To prevent such situations, we introduce Configuration Constraint (CC) rules that take priority over the GR rule. There are two variants: rules that trigger a detachment and rules that inhibit detachment events.

Obedience to travel limits is an example of the former variant. Various physical constraints can lead to leg travel limits, such as joint limits in an articulated leg or leg trajectories in which only restricted sections produce useful climbing motions. Triggering detachment earlier than indicated by the GR rule allows the robot to continue climbing even if a leg runs out of travel. Such a CC rule can be written as

$$s_i \geq L_i \tag{4}$$

where i is the index of an arbitrary leg and L_i is that leg's travel limit in travel space. Hereafter, we will assume all travel limits L_i are equal to L. The travel limits define the boundaries of the n-dimensional workspace of the robot, $\mathcal{W}$:

$$\mathcal{W} \equiv \{\vec{s} \mid L \geq s_i \geq -\gamma L - \delta\} \tag{5}$$

The second type of CC rule inhibits detachment events that would remove legs essential for support. While the conditions for stability typically depend on the details of the robot and the climbing surface, the gait regulation algorithm only requires that they can be represented by a set of linear inequalities in terms of travels:

$$\sum c_i s_i > c_0 \tag{6}$$

One common measure of stability for legged robots is the support polygon [14]. While this is more readily applicable to walking than climbing, it can be used to form CC rules under this framework. Regions in travel space for which detachment is prohibited under a support polygon measure of stability are not necessarily linear and therefore cannot directly be expressed in the form of (6). However, conservative piecewise linear approximations can be formed which meet the linear CC rule requirement and allow the analysis in the following section to be performed.

The region of the robot workspace, $\mathcal{W}$, in which no inhibitory CC rule is active is referred to as $\mathcal{W}_s$ and includes all configurations in which detachment is allowed.

The inhibitory CC rules clearly override the GR rule, as stability is essential. However, more complex cases can also arise, as when a leg hits a joint limit, but its support is needed for stability. If two CC rules produce a conflict, the robot cannot continue to produce forward motion; all legs in stance should halt and wait for the legs in swing to reattach such that the offending leg(s) are not required for support.

III. TWO-LEGGED EXAMPLE

We illustrate the approach for gait regulation in the context of a climbing robot with two virtual legs [16], for which the travel space is conveniently represented with two-dimensional diagrams. For the quadrupedal RiSE robot variant used in the subsequent empirical examples, this simplification corresponds to a bound gait in which the front and rear legs move together. The algebraic method extends straightforwardly to larger numbers of legs, but the travel diagrams become difficult to visualize.

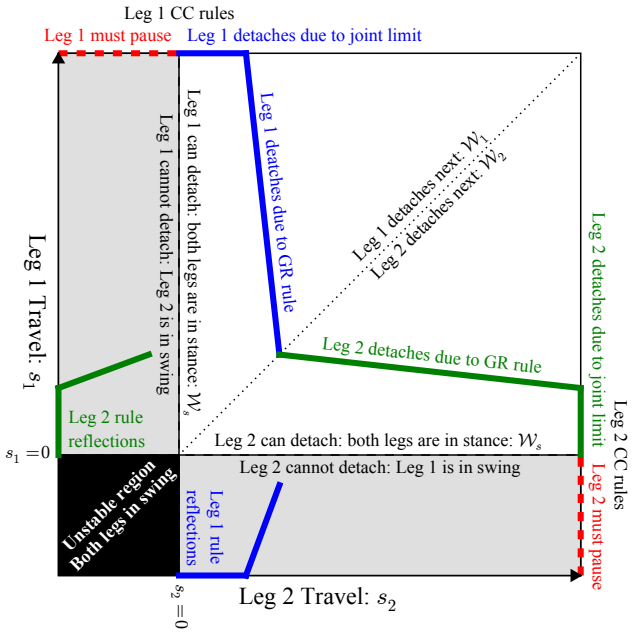

Fig. 2. Annotated travel space for a two-legged robot. The thick solid lines trigger detachment, which instantly shifts the travel to their respective reflections, produced by application of (2).

Because the RiSE robot has a tail, either the front or rear pair of legs can hold it against the wall, pulling inward and upward against gravity. Therefore the stability requirement is that at least one virtual leg be in stance. In travel space, the stable region formed is equivalent to the Stance Complex presented in [17]:

$$\mathcal{W}_s \equiv \{\vec{s} \in \mathcal{W} \mid s_1 \geq 0 \ , \ s_2 \geq 0\} \tag{7}$$

The workspace, $\mathcal{W}$, for the two-legged robot is shown in Fig. 2, along with the bounding GR and CC rules. These rules manifest as line segments in $\mathcal{W}$ (in higher dimensions, they would appear as $n-1$ polygonal facets.) The rules are given as linear inequalities, thus they can be used to describe polygonal regions where the rule will trigger detachment [18]. The interior regions are unreachable except by an initial configuration, thus we will only concentrate on their leading edges.

The workspace is split into two regions: $\mathcal{W}_1 \equiv \{\vec{s} \in \mathcal{W} \mid s_1 > s_2\}$ and $\mathcal{W}_2 \equiv \{\vec{s} \in \mathcal{W} \mid s_2 > s_1\}$ which are the only orderings of two legs. Since the GR rule operates on the ordered travels, it manifests as symmetric around the boundary between $\mathcal{W}_1$ and $\mathcal{W}_2$. The slope of the GR rule is defined by the weights ω_i. The CC rule appears on the joint-limit boundary of $\mathcal{W}$.

In this example, we mark segments of the rules that trigger detachment with a solid line. The dashed lines represent boundaries where the joint limit CC rule and the stability CC rule conflict, forcing the robot to pause instead of detaching a leg. For rules that do trigger detachment, we can

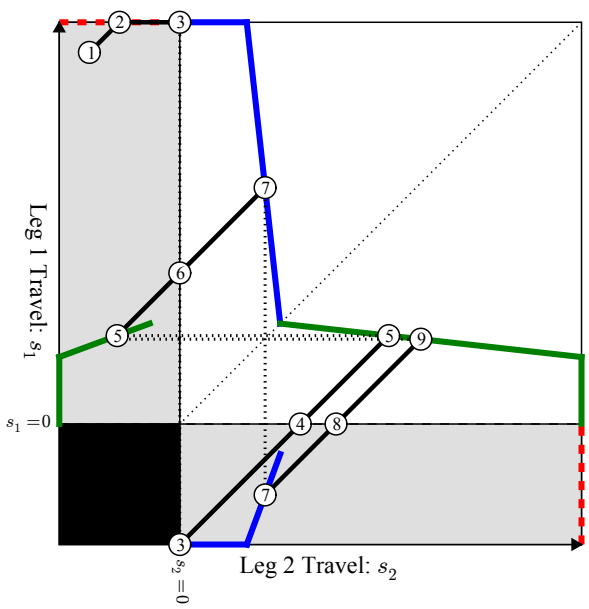

Fig. 3. A sample sequence of events in a converging gait: (1) The initial configuration of the robot. (2) Leg 1 strikes a joint limit, but cannot detach because leg 2 is in swing. (3) Leg 2 attaches, triggering detachment in Leg 1. (4) Leg 1 attaches. (5) GR rule triggers detachment in Leg 2. (6) Leg 2 attaches. (7) GR rule triggers detachment in Leg 1. (8) Leg 1 attaches. (9) GR rule triggers detachment in Leg 2. Events 5 and 9 are very close together indicating that they are near the equilibrium path of the gait.

geometrically show the result by applying the detachment transformation (2) to the line segments.

Fig. 3 shows several steps of the robot whose gait converges quickly. The robot's path in travel space is almost always in the same direction since $\frac{ds_i}{dt} = 1$, except when CC rules conflict or a detachment creates a discontinuous jump.

IV. CONVERGENCE ANALYSIS

We present two analysis techniques. The algebraic method, which extends easily to multiple legs, will determine whether the gait will converge locally to an equilibrium path. The geometric method can be used to test for global convergence from any starting configuration. It also extends to larger numbers of legs, although the diagrams become increasingly complex.

A. Algebraic Method

For algebraic analysis we use the sorted travels, s_i^* and express them in homogeneous coordinates [19]:

$$\vec{s^*}(t) = \begin{bmatrix} s_1^*(t) & s_2^*(t) & \dots & s_n^*(t) & 1 \end{bmatrix}^T \tag{8}$$

We construct matrix operators for $\vec{s^*}$ that compute subsequent states given an initial state. The sorted vector allows us to examine the effect of a single footstep, but requires operators that maintain the sorting.

In a gait that detaches m legs at a time, the one with the highest travel, s_1^*, detaches first, along with an additional $m-1$ legs that are determined by the gait. We assume that upon convergence these m legs are synchronized and thus have the highest travels $s_i^* \dots s_m^*$. The travel is modified by application of (2), and $\vec{s^*}$ is reordered to maintain its sorting. These two operations are performed simultaneously by the m-leg Detachment Matrix

$$D_m = \begin{bmatrix} 0 & I_{n-m} & 0 \\ -\gamma I_m & 0_{m\times n-m} & -\delta \vec{1_m} \\ 0 & 0_{1\times n-m} & 1 \end{bmatrix} \tag{9}$$

We represent the locomotion of the legs as a constant translation in homogeneous travel coordinates along the vector $\begin{bmatrix} 1 & 1 & \dots & 1 & 0 \end{bmatrix}$. This translation continues until the next GR rule transition, given by (3). The difference between T and the current weighted sum of the leg travels tells us how far each leg will advance, yielding

$$s_i^*(k+1) = s_i^*(k) + \left(T - \sum \omega_i s_i^*(k)\right). \tag{10}$$

This operation can be performed on all the legs by the Translation Matrix

$$L = \begin{bmatrix} I_n - \vec{1}_n \vec{w}^T & T\vec{1}_n \\ \vec{0}_n^T & 1 \end{bmatrix} \tag{11}$$

An example of the construction of matrices (9) and (11) is shown later in (15). Given these two matrices, we construct the Recurrence Matrix, $R_m = LD_m$, which maps an augmented leg-travel vector from one detachment to the next. This matrix performs an operation analogous to the Poincaré map. This analogy should not be taken too far however, as some of the assumptions for Poincaré analysis are violated in this case; notably that the travel direction is not necessarily perpendicular to the surface defined by the GR rule and the underlying system is not continuous.

Once we have R_m we can analyze its eigenvalues to determine convergence of the GR rule. Because we are using homogeneous coordinates, the construction of R_m guarantees there will always be at least one eigenvalue $\lambda = 1$, and the corresponding eigenvector, $\vec{\xi}(\lambda = 1)$, will be the only eigenvector with a nonzero final entry. When $\vec{\xi}$ is scaled so that its final element is 1, it gives the equilibrium detachment state of the leg travels.

This analysis provides two necessary conditions for the gait to converge to $\vec{\xi}$. First, all other modes must decay [20]: $|\lambda_i| < 1$. Second, $\vec{\xi}$ must be achievable: $\vec{\xi} \in \mathcal{W}_s$. If $\vec{\xi} \notin \mathcal{W}_s$, then the GR Rule will cause the travel to decay such that it will try to leave $\mathcal{W}_s$. A CC Rule will then cause a transition, thereby preventing the travel from converging to the equilibrium, $\vec{\xi}$.

B. Geometric Method

If the gait satisfies the necessary conditions generated by the algebraic analysis, a more involved geometric and graph theory analysis will allow us to establish global convergence. $\mathcal{W}$ is segmented into distinct regions, so that each region transitions due to a single rule, progressing to a single new region. The regions can be viewed as nodes in a directed graph, where every node has an out-degree of 1.

TABLE I

GAIT PARAMETERS USED FOR THE TWO-LEGGED EXAMPLE.

	γ	δ	ω_1	ω_2	T	L
Example	0.3	0.15 sec	0.3	0.7	0.16 sec	0.5 sec

The first step in the geometric analysis is to create the boundary of $\mathcal{W}$. Because GR rules operate on the ordered travel vector, $\vec{s^*}$, but points in $\mathcal{W}$ are unordered travels, $\mathcal{W}$ is divided into $n!$ regions corresponding to each possible leg ordering. Next any inhibitory CC rules are outlined, as well as any unstable regions, to yield $\mathcal{W}_s$

At this point we begin construction of the transition boundary B, composed of facets associated with GR or CC rules that trigger detachment. First, facets for the GR rules are added in each of the $n!$ regions. Any GR rule facets that lie outside $\mathcal{W}_s$ are removed, since an inhibitory CC rule would prevent detachment. Then facets for any detachment CC rules are added to complete B. A complete transition boundary transects $\mathcal{W}_s$; i.e. the projections of B and $\mathcal{W}_s$ onto the plane perpendicular to $\begin{bmatrix} 1 & 1 & \dots & 1 \end{bmatrix}$ are equal.

Given B, $\mathcal{W}$ can be segmented to satisfy the conditions outlined above. The full details of this segmentation process are not presented here due to space limitations. During segmentation, the directed graph, G, is constructed and each node is categorized according to which rules govern transition from it: *GR nodes* transition due to the GR rule only; *CC nodes* transition due to CC rules.

For the gait to be globally convergent, the robot must be able to start in any configuration and eventually enter a cycle in which it always transitions due to a GR rule. Equivalently, requirements for G are for all cycles in G to include only GR nodes, and for every other node in G to be able to reach one of these cycles. These properties can be checked by inspection or through methods outlined in [21], [20].

If these requirements are satisfied, the robot will reach a GR region within a finite number of steps from any starting region. After reaching the GR region, the detachment state will asymptotically converge to the equilibrium path, as shown by the algebraic analysis. If the requirements are not satisfied, the robot can enter a cyclic gait in which at least one transition is due to a CC rule, and the detachment state will not converge to $\vec{\xi}(\lambda = 1)$.

C. Two-Legged Example Analysis

The parameters for a GR rule are listed in Table I along with the joint limit L. We empirically determine γ and δ by first selecting a desired forward velocity and measuring the various velocities and time durations exhibited by the robot. The weights ω and trigger point T are also determined empirically. The synthesis of gaits is beyond the scope of this article, but experience indicates that the the heaviest weights for an n legged robot that lifts m legs at a time should be ω_n and ω_{m+1} (which for the two legged case are the same leg) with a light weight for ω_1. This approach couples the actions of each leg to the next and previous neighbors in travel.

Using the parameters in Table I, we form the R matrix:

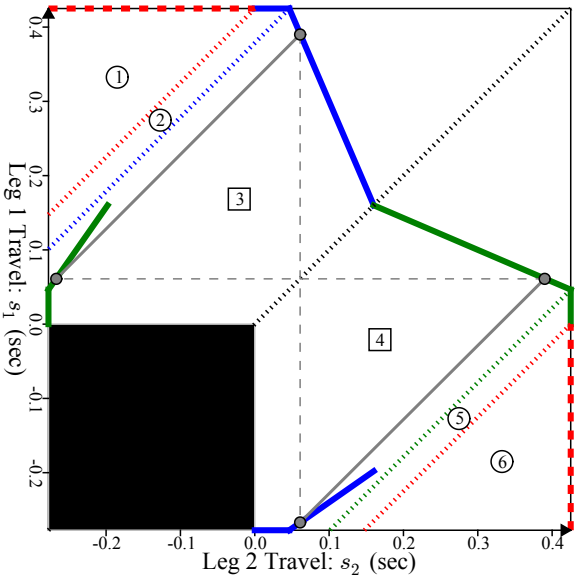

Fig. 4. The 2D gait space using parameters from Table I. Five dotted lines, each angled at 45°, divide $\mathcal{W}$ into six regions. Each region is indicated by a number which is either circled or squared. Four small gray-filled circles and their connecting lines indicate the equilibrium path.

$$R_1 = LD_1 = \begin{bmatrix} 0.21 & 0.7 & 0.265 \\ -0.09 & -0.3 & 0.115 \\ 0 & 0 & 1 \end{bmatrix} \tag{12}$$

The eigenvalues and associated eigenvectors are:

$$\begin{aligned} \vec{\xi}(\lambda = 1) &= \begin{bmatrix} 0.390 & 0.061 & 1 \end{bmatrix}^T \\ \vec{\xi}(\lambda = -0.09) &= \begin{bmatrix} 7 & -3 & 0 \end{bmatrix}^T \\ \vec{\xi}(\lambda = 0) &= \begin{bmatrix} 10 & -3 & 0 \end{bmatrix}^T \end{aligned} \tag{13}$$

This system is locally convergent because we have a single eigenvalue equal to 1 while the rest have absolute values less than one and $\vec{\xi}(\lambda = 1) \in \mathcal{W}_s$. This eigenvector describes the equilibrium travels of the legs before detachment, while the others describe transient modes that decay.

The algebraic analysis has demonstrated convergence in the neighborhood of equilibrium; we now check for global convergence. Fig. 4 shows the gait space divided into six regions. The boundaries between regions are lines parallel to the travel direction $\begin{bmatrix} 1 & 1 \end{bmatrix}^T$. The boundaries coincide with the boundaries between the various GR and CC rules such that each region meets the requirement of detaching due to a single rule. In addition, each region only maps to a single distinct region, thus we do not need to further segment the gait space. Regions 2 and 3 both transition to region 4 and likewise for 4 and 5 to 3. Regions 1 and 6 encounter the conflicting CC rule boundary, but in these cases, the travel slides along the boundary until it reaches the corners of regions 2 and 5, respectively, which instantaneously triggers detachment. Using these regions, we obtain the directed graph:

$$G: \ (1) \rightarrow (2) \rightarrow \boxed{4} \leftrightarrow \boxed{3} \leftarrow (5) \leftarrow (6) \tag{14}$$

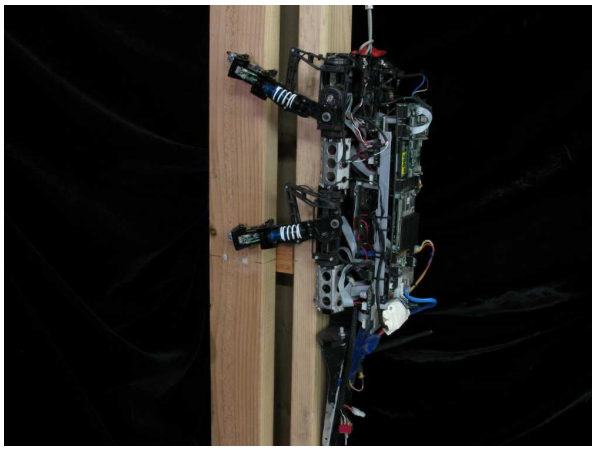

Fig. 5. Four-legged variant of the RiSE robot climbing a wooden substrate.

TABLE II

GAIT PARAMETERS USED FOR THE THE STABLE AND UNSTABLE GAIT.

	γ	δ	ω_1	ω_2	ω_3	ω_4	T
Convergent	0.3	0.15 s	0.3	0.3	0.2	0.2	0.27 s
Non-Convergent	0.3	0.15 s	0.5	0.5	0.0	0.0	0.38 s

GR nodes are represented by boxes and CC nodes represented by circles. By inspection, the only cycle involves GR regions 3 and 4, and all other nodes lead to that cycle. Therefore, the gait is globally convergent.

V. RESULTS

While this gait regulation technique is extendable to various forms of locomotion, it was designed in response to challenges faced by vertical climbing. These trials were performed on a vertical wooden substrate as shown in Fig. 5. The robot climbed six feet, but the data only show a small fraction, in order to highlight the details of the plots.

We implemented two gaits, one convergent and one non-convergent, on the quadrupedal RiSE robot and compared the observed behavior with predictions from the analysis. For these tests, the robot climbed using a bound gait where the front and rear pair of legs are out of phase with each other. We considered the four legs separately to see the effects of disturbances when a single leg slips. The algebraic analysis extends easily to the four-legged description using a 5×5 matrix but the geometric analysis requires a 4D representation of the travel space and is not shown here. The GR rule, with parameters from Table II, governs the bounding gait and synchronizes the left and right legs.

Applying (9), we get the detachment matrix, LD_2:

$$\begin{bmatrix} 1-\omega_1 & -\omega_2 & -\omega_3 & -\omega_4 & T \\ -\omega_1 & 1-\omega_2 & -\omega_3 & -\omega_4 & T \\ -\omega_1 & -\omega_2 & 1-\omega_3 & -\omega_4 & T \\ -\omega_1 & -\omega_2 & -\omega_3 & 1-\omega_4 & T \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ -\gamma & 0 & 0 & 0 & -\delta \\ 0 & -\gamma & 0 & 0 & -\delta \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (15)$$

A. Convergent Gait

We substitute the "convergent" values from Table II and form $R_2 = LD_2$ to yield the eigenvalues: $\begin{bmatrix} 0.55i & -0.55i & 0 & -0.48 & 1 \end{bmatrix}$ and the eigenvector $\vec{\xi}(\lambda = 1) = \begin{bmatrix} 0.41 & 0.41 & 0.07 & 0.07 & 1 \end{bmatrix}^T$. According to our criteria in Section IV-A, this gait is locally convergent. As the equilibrium detachment travel vector shows, the gait keeps the front and rear out of phase with each other and keeps the legs within each pair in phase with each other.

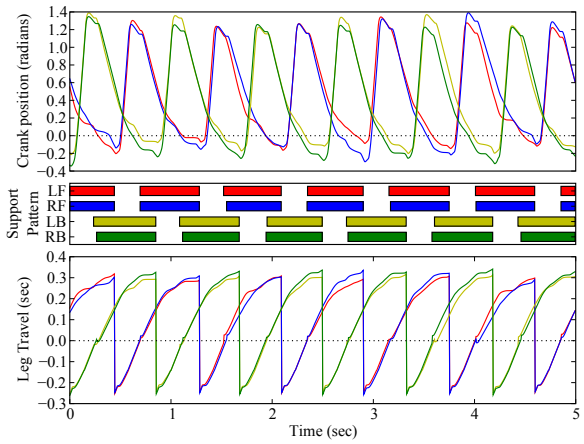

Fig. 6. The robot begins climbing using a convergent GR rule. The top subplot shows each leg's crank angle which pulls the robot up the wall. The middle subplot shows the support pattern for the robot. The bottom subplot shows the travel of each leg. The discontinuities occur when the controller decides to lift each leg. The robot maintains positive body velocity throughout the gait.

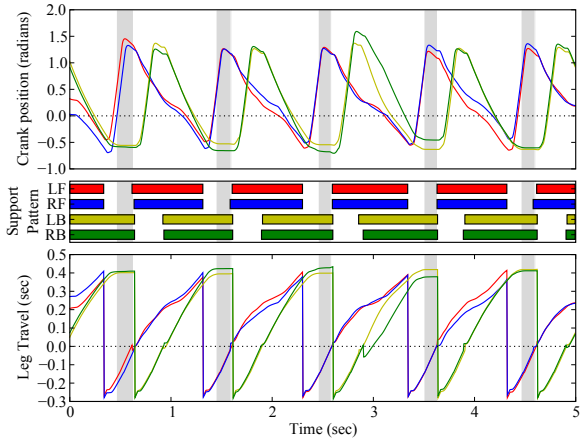

Fig. 7. The robot begins climbing using a non-convergent GR rule with the same subplots as Fig. 6. The grayed sections indicate where the robot's body velocity drops due to a CC pause condition: one pair reaches end of travel while the other pair is in swing.

Fig. 6 shows the gait at steady-state. There are a few disturbances in the travel curve during attachment because the legs do not always reattach at the same spot. Also, as the robot climbs it experiences different loading conditions for each step, so every footfall looks different. There may also be predictable but unmodeled discrepancies between the internal model and the physical leg trajectories. Despite these errors, the gait converges readily.

B. Non-Convergent Gait

The eigenvalues for R_2 using the non-convergent parameters are $\begin{bmatrix} -1 & 0.55i & -0.55i & 0 & 1 \end{bmatrix}$. The eigenvector associated with the -1 eigenvalue is $\vec{\xi}(\lambda = -1) = \begin{bmatrix} 0 & 0 & 1 & 1 & 0 \end{bmatrix}$. Since there are two eigenvalues with absolute values of one, the system will not converge to a single equilibrium path. Fig. 7 shows the results of this gait, which, as predicted, does not produce a symmetric gait. However, since the algorithm protects the robot from entering a fatal configuration, the robot can still climb.

These parameters correspond to a gait in which the front and back legs are uncoupled. Further, because RiSE is non-ideal, different legs move with slightly different speeds during swing. Because the GR rule does not couple the front and back legs, the faster pair is able to catch up to the slower pair and a stability CC rule forces them to stay in stance until the slower pair has attached.

VI. CONCLUSIONS AND FUTURE WORK

We presented an algorithm for gait regulation for a climbing legged robot that operates under practical considerations of vertical climbing imposed by stability and motor thermal optimization. Subject to these considerations, the algorithm regulates the gait by varying leg detachment timings. We present algebraic and geometric tests for local and global convergence to a standard free gait [1], which yields an even rhythm and is robust against disturbances. The algorithm is designed to ensure the robot remains stable at all times, even when a gait is unable to converge. We presented the method in the context of a robot with two virtual legs and tested it on a four-legged variant of the RiSE robot.

It should be noted that the robot can still climb without reaching convergence, but the robot will generate CC pause conditions when the legs interfere, as shown in Fig. 7. For most legged robots, this would be of little concern, but for climbing robots, pausing is energy intensive as the robot must station-keep against the force of gravity and lose forward momentum. A properly convergent GR rule will synchronize the legs and prevent such conditions even if the legs temporarily slip out of synchronization. While we recognize the variety of gait controllers that can effectively maintain synchronization, most cannot do so without violating the conditions set forth in Section I. An event-based variable-period controller, such as the one described here, does not violate the conditions with the added benefit of mathematically demonstrable convergence.

We plan to extend this algorithm to different gaits with larger numbers of legs. The approach should be also suitable for different types of legged robots, including walking robots that are not as thermally constrained. We plan to extend the geometric analysis tools to larger dimensions, for which the representation of the travel space is no longer easy to visually inspect. We are also interested in dynamically switching between different gaits and sets of parameters. Finally, while the approach presented here provides for gait analysis, the synthesis of new gaits is an important and more open-ended problem for future work.

VII. ACKNOWLEDGMENTS

The authors thank the members of the Biomimetics and Dexterous Manipulation Laboratory for their advice during this project and Robin Deis Trujillo for support and editing. The RiSE project was supported by the DARPA BioDynotics Program. Salomon Trujillo is partially supported by the Alfred P. Sloan Foundation.

REFERENCES

[1] S. Hirose, Y. Fukuda, and H. Kikuchi, "The gait control system of a quadruped walking vehicle," *Advanced robotics*, vol. 1, no. 4, 1986.

[2] S. Hirose and O. Kunieda, "Generalized Standard Foot Trajectory for a Quadruped Walking Vehicle," *The International Journal of Robotics Research*, vol. 10, no. 1, pp. 3–12, 1991.

[3] S. Trujillo and M. Cutkosky, "Thermally constrained motor operation for a climbing robot," *Robotics and Automation, 2009. ICRA '09. IEEE International Conference on*, pp. 2362 – 2367, Apr 2009.

[4] M. Spenko, G. Haynes, and J. Saunders, "Biologically inspired climbing with a hexapedal robot," *Journal of Field Robotics*, vol. 25, pp. 223–242, April 2008.

[5] S. Kajita and B. Espiau, "Legged robots," *Springer Handbook of Robotics*, pp. 361–389, 2008.

[6] A. Calvitti and R. Beer, "Analysis of a distributed model of leg coordination," *Biological Cybernetics*, vol. 82, no. 3, pp. 197–206, Feb 2000.

[7] H. Chiel, R. Beer, R. Quinn, and K. Espenschied, "Robustness of a distributed neural network controller forlocomotion in a hexapod robot," *IEEE Transactions on Robotics and Automation*, vol. 8, pp. 293–303, Jan 1992.

[8] T. Roggendorf, "Comparing different controllers for the coordination of a six-legged walker," *Biological Cybernetics*, vol. 92, no. 4, pp. 261–274, Apr 2005.

[9] K. Waldron, "Force and motion management in legged locomotion," *Robotics and Automation, IEEE Journal of*, vol. 2, no. 4, pp. 214 – 220, Dec 1986.

[10] G. C. Haynes and A. Rizzi, "Gait regulation and feedback on a robotic climbing hexapod," *Proceedings of Robotics: Science and Systems*, Aug 2006.

[11] M. Frasca, P. Arena, and L. Fortuna, *Bio-inspired emergent control of locomotion systems*. World Scientific Publishing Co. Pte. Ltd., 2004.

[12] H. Cruse, T. Kindermann, M. Schumm, and J. Dean, "Walknet—a biologically inspired network to control six-legged walking," *Neural Networks*, vol. 11, no. 7–8, pp. 1435–1447, 1998.

[13] K. Wait and M. Goldfarb, "A biologically inspired approach to the coordination of hexapedal gait," *Robotics and Automation, 2007 IEEE International Conference on*, pp. 275–280, Apr 2007.

[14] S. Song and K. Waldron, "An analytical approach for gait study and its applications on wave gaits," *The International Journal of Robotics Research*, vol. 6, no. 2, pp. 60–71, Jun 1987.

[15] E. Klavins and D. Koditschek, "Phase regulation of decentralized cyclic robotic systems," *The International Journal of Robotics Research*, vol. 21, no. 3, pp. 257–275, Jan 2002.

[16] M. Raibert, M. Chepponis, and H. Brown, "Running on four legs as though they were one," *Robotics and Automation, IEEE Journal of*, vol. 2, no. 2, pp. 70 – 82, Jun 1986.

[17] G. C. Haynes, F. R. Cohen, and D. E. Koditschek, "Gait transitions for quasi-static hexapedal locomotion on level ground," *International Symposium of Robotics Research*, Aug 2009.

[18] J. E. Goodman and J. O'Rourke, *Handbook of discrete and computational geometry*. CRC Press LLC, 2004.

[19] D. H. Ballard and C. M. Brown, *Computer vision*. Prentice-Hall Inc., 1982.

[20] G. Strang, *Linear algebra and its applications*. Thomson Learning, 2006.

[21] J. A. Bondy and U. S. R. Murty, *Graph Theory with Applications*. American Elsevier Publishing Co., Inc., 1976.