NOTICE WARNING CONCERNING COPYRIGHT RESTRICTIONS:
The copyright law of the United States (title 17, U.S. Code) governs the making of photocopies or other reproductions of copyrighted material. Any copying of this document without permission of its author may be prohibited by law.

Legless Locomotion for Legged Robots

Ravi Balasubramanian, Alfred A. Rizzi, and Matthew Mason

CMU-RI-TR-04-05 2

University Libraries
Carnegie Mellon University

Legless Locomotion for Legged Robots

Ravi Balasubramanian Alfred A. Rizzi Matthew T. Mason

CMU-RI-TR-04-05

April 2003

The Robotics Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213

© Carnegie Mellon University

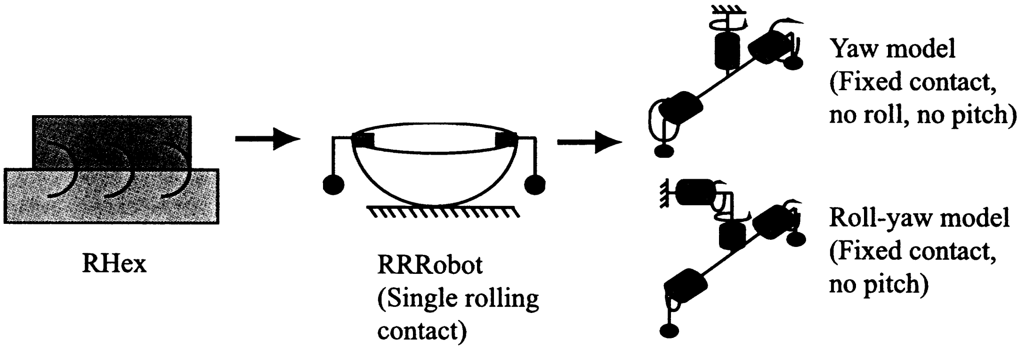
Executive Summary

Legged robots offer rough terrain mobility, but sometimes conventional legged locomotion may fail, for example, when the robot gets stuck on a rock, and no leg has contact with the environment (the robot is *high-centered*). We propose a novel locomotion technique for a high-centered legged robot. We hypothesize that by using its legs as reaction masses, the robot can rock and roll on its stomach and move incrementally. In this report, we use simple models to study how leg motions affect body attitude configuration for a high-centered robot.

Our interest in this problem arose from our experience with a highly mobile, robust hexapod robot getting high-centered in challenging terrain. One way of freeing a high-centered legged robot is to locomote the robot till it topples over. To simplify the analysis, we used *Rocking and Rolling Robot* (RRRobot). RRRobot is a hemisphere that has two short legs with small masses at their distal ends. Assuming RRRobot rolls on the flat ground without slipping, the goal is to locomote the always-high-centered RRRobot on the plane by finding suitable leg trajectories. Informal experiments with a teleoperated RRRobot suggest that a properly designed leg motion can produce attitude oscillations that result in incremental translation of the robot. The result is a form of legless, whole-body locomotion using the legs as reaction masses.

We use simulation and dynamics analysis to understand how leg motions can be used to create small body attitude changes. We study two simplified versions of RRRobot, one in which body motion is restricted to yaw and the other in which body motion is restricted to roll and yaw. We present some observations from control theory and show that simple controls exist to locally modify body attitude configuration in the simplified models. We hypothesize that such attitude oscillations when coupled with the contact constraints will produce RRRobot translation. More experiments are necessary to verify this.

Interesting future directions of work are to study the influence of body shape on the proposed locomotion technique and to control the direction of locomotion. A shorter version of this report is published in Proceedings of the International Conference on Intelligent Robots and Systems, 2003.



Progression of models used.

Contents

1 Introduction **1**
1.1 Motion options for high-centered RHex . . . 1
1.2 RRRobot . . . 3
1.3 Related Work . . . 4

2 The Yaw model **5**
2.1 The Yaw Model mathematical details . . . 6
2.2 The Yaw Model formal analysis . . . 8
2.3 Changing body yaw using leg motions . . . 10

3 The Roll-Yaw model **11**
3.1 The Roll-Yaw model details . . . 12
3.2 Roll-Yaw model simulations . . . 12
  3.2.1 Gait 1 . . . 12
  3.2.2 Gait 2 . . . 13
  3.2.3 Gait 3 . . . 15

4 Discussion **15**

5 Conclusion **16**

List of Figures

1 The RHex experimental platform. 1
2 Producing locomotion using multiple contacts. 2
3 Producing locomotion using a single rolling contact. 2
4 The RRRobot experimental platform and its similarity to RHex. 3
5 RRRobot translation using body attitude oscillation. 4
6 The snakeboard. 6
7 The roller racer. 6
8 A simplified RRRobot model allowing only body yaw. 7
9 Numerical stability of the yaw model controllability matrix 9
10 Lie bracket-inspired leg motions to change body yaw. 11
11 A simplified RRRobot model allowing only body roll and yaw. 12
12 Variation of RRRobot roll and yaw inertias with leg configuration. 13
13 Gait 1 leg motions. 13
14 Body attitude oscillations produced by Gait 1. 14
15 Gait 2 leg motions. 14
16 Body attitude oscillations produced by Gait 2. 14
17 Body attitude oscillations produced by Gait 3. 16

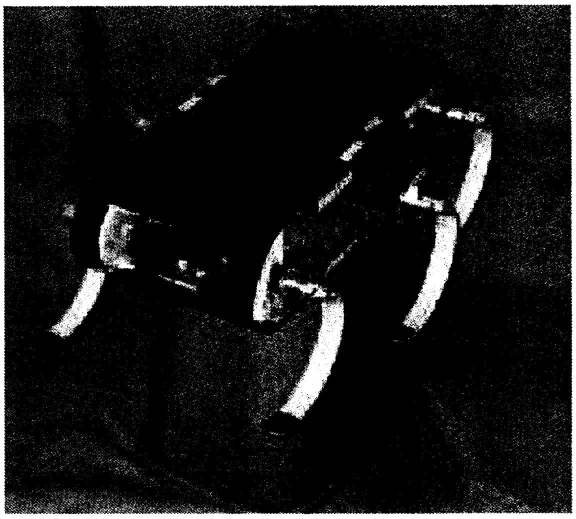


Figure 1: The RHex experimental platform high-centered on a block.

1 Introduction

Legged robots offer rough terrain mobility, evident from examples of robots exploring live volcanos [1], climbing stairs [3], and negotiating inclines [12]. However, conventional legged locomotion may fail when a leg becomes stuck in a hole, when there is not enough clearance to stand, or when no leg can engage the ground (the robot is *high-centered*). This report presents a novel locomotion strategy for situations where conventional use of legs is not possible.

Our interest in this problem arose from experience with RHex [16]. RHex is a simple, robust, and highly mobile hexapod robot with many interesting behaviors such as running at 2 m/s, flipping backwards, etc. (see Fig. 1, http://rhex.net). Each leg has a single actuated hip joint with no joint limit, allowing it to rotate all the way around. When RHex is high-centered on a block, the legs do not touch the ground. Seeing how RHex gets stuck in challenging terrain, we proceeded to look at robot recovery, i.e., how to free RHex from stuck configurations. The only possible means of locomotion when RHex is high-centered is by using the legs as reaction masses. The robot can then wiggle its legs in such a way that body motion is produced. This report presents a preliminary study on simplified kinematic and dynamic models of robot-environment interaction which, when superimposed, may provide a strategy to free a high-centered legged robot like RHex.

1.1 Motion options for high-centered RHex

There are at least two options by which a high-centered RHex can escape from its stuck configuration.

1. A rocking motion that exploits body protrusions (see Fig. 2).
2. A smooth rocking motion that exploits a rolling contact between a round RHex body and the block surface (see Fig. 3).

Option 1 is a rocking motion that exploits the non-smoothness of the robot body, allowing small body protrusions to act as "feet" (see Fig. 2). By a series of rolling and yawing motions, the rocking motion shifts the robot weight from one foot to another. Option 1, thus, uses multiple contacts between the robot and obstacle.

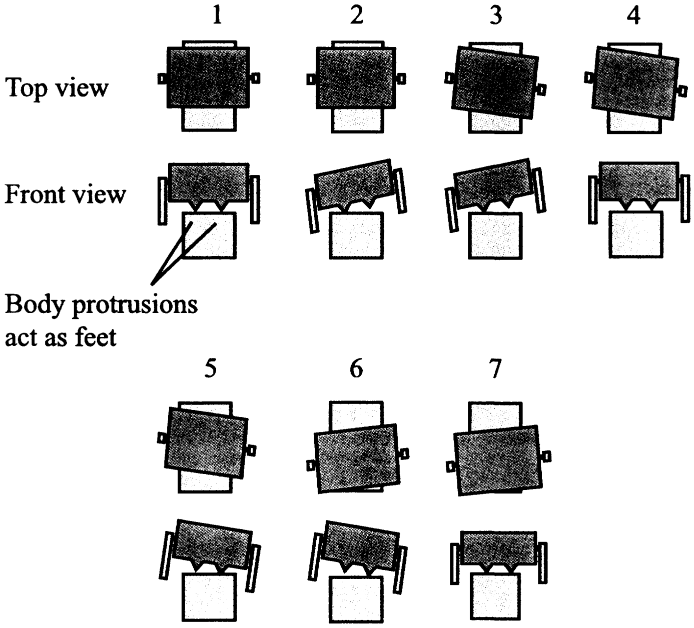

1
2
3
4
Top view
Front view
Body protrusions act as feet
5
6
7

Figure 2: Body attitude oscillations can produce locomotion for a robot with body protrusions.

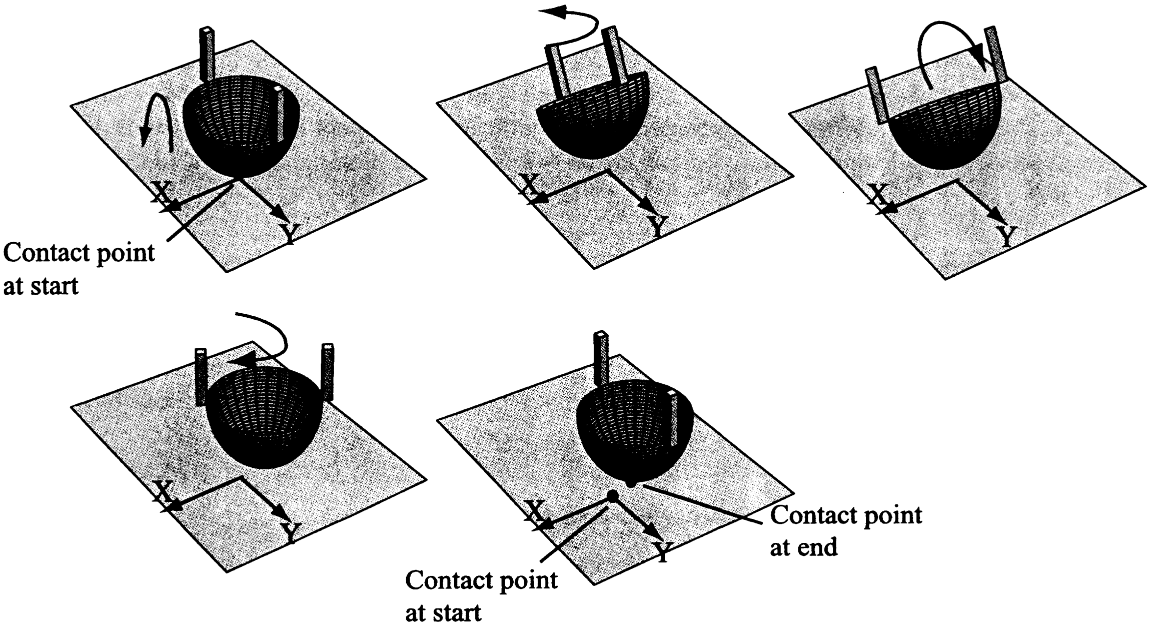
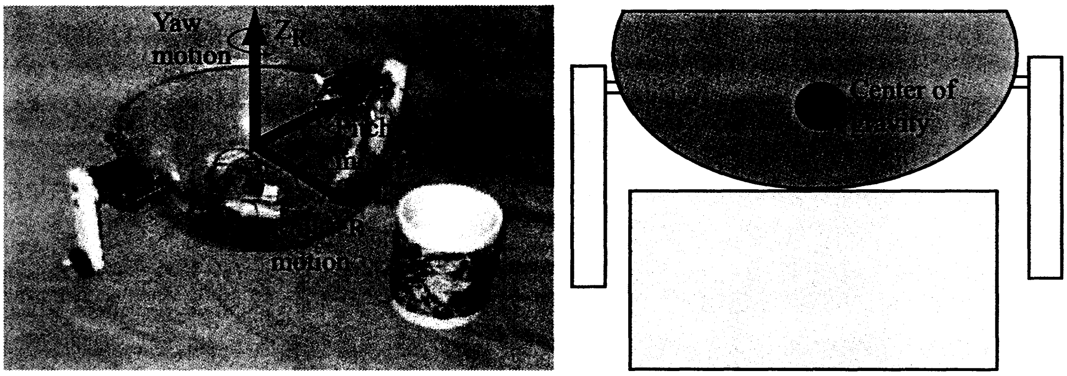

X
Y
Contact point at start
X
Y
X
Y
X
Y
X
Y
Contact point at end
Contact point at start

Figure 3: Body attitude oscillations coupled with a slip-free contact can produce locomotion for a round-bodied robot.



(a) (b)

Figure 4: (a) The RRRobot experimental platform uses halteres to induce body attitude oscillations leading to body translation and (b) a high-centered round bodied RHex is similar to RRRobot sitting on a plane.

Option 2 assumes a round-bodied RHex sitting on a smooth surface (see Fig. 3). Then, interleaved roll and yaw body motions with no slip at the contact point can locomote the robot in the plane, while taking advantage of the constraints on the velocities of the two bodies in contact (nonholonomic constraints).

We prefer to study option 2 to option 1. This is because RHex's leg masses are small when compared to the body weight, and, hence, the forces generated by leg motions may be too small to lift the robot weight onto the pivot points in option 1. Also, each time the weight shifts from one foot to another, there will be impacts between the robot body and block surface. These impacts may produce slip-related motion, which is difficult to model. Thus, option 1 may not be suitable for RHex's conditions and is also difficult to model. In contrast, with a rounded body, even small torques can cause the robot's attitude to oscillate, and these oscillations can generate translation when coupled with the nonholonomic contact constraints between the robot body and block surface. Also, since all interactions of the body with the environment are smooth, there are no discontinuities to model. Thus, option 2 may be effective in locomoting a round-bodied robot on its stomach and is also easier to model than option 1. The rest of this report will focus on option 2. The challenge will be to find a set of body attitude oscillations that produce locomotion and appropriate leg motions to achieve such body motions.

1.2 RRRobot

As a preliminary test to see if a rocking motion coupled with the nonholonomic contact constraints can produce locomotion for a round-bodied robot, we built a simple prototype called *RRRobot* (Rocking and Rolling Robot), a hemisphere with two actuated legs that cannot reach the ground (see Fig. 4a). Thus, the two legs are used only as reaction masses, and they are more appropriately called *halteres*, after the dumbbells sometimes used by athletes to give impetus in leaping. The shape of RRRobot's rounded body, its interaction with the obstacle surface, and its influence on RRRobot locomotion is not the primary focus of this report, i.e., the body shape could have been, say, oval instead of spherical. The crucial point is that RRRobot is always high-centered and has similarities to a high-centered round-bodied RHex (see Fig. 4b). The goal of building RRRobot is to find suitable leg trajectories that cause RRRobot to translate in the XY-plane. Table 1 gives a sequence of interleaved body roll and yaw motions that locomotes RRRobot in the XY-plane (see Fig. 5).

Body motion	XY-plane motion
Positive roll	P_0 to P_1
Positive yaw	P_1 to P_2
Negative roll	P_2 to P_3
Negative yaw	P_3 to P_4

Table 1: A sequence of body attitude motions that translates RRRobot in the XY-plane (see Fig. 5).

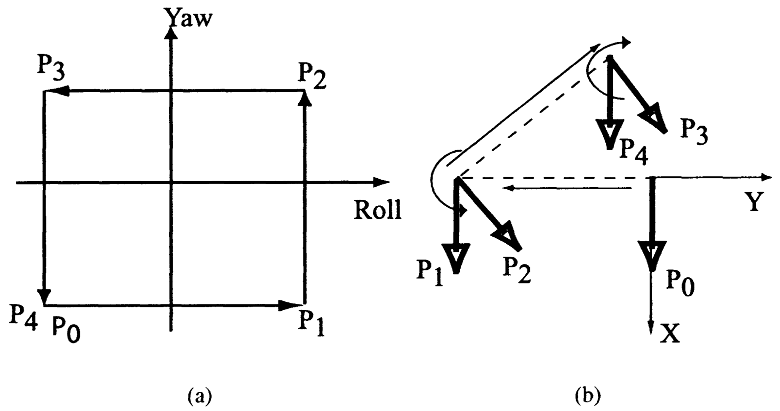


Figure 5: (a) A sequence of body attitude motions in roll-yaw space and (b) the resulting translation in the XY plane (see Table 1).

To understand the relationship between leg motion and body roll-pitch-yaw dynamics, we tried different leg motion patterns or *gaits* on a teleoperated RRRobot- patterns such as moving just one leg, moving both legs in phase, and moving the legs out-of-phase. When we drove the halteres with periodic waveforms in an interleaved fashion, the resulting motion included incremental translation, some changes in yaw, and large oscillations in pitch and roll. Thus, we succeeded in getting some translation through teleoperation, but failed in controlling the pitch oscillations. We believe translation and controlled attitude oscillations may be possible using a computer-controlled RRRobot.

Our experiments with RRRobot suggest that oscillations in body attitude might be a practical way to recover from a high-centered state. With locomotion of high-centered robots using attitude oscillations as motivation, the remainder of this paper focuses on the use of halteres to control the robot body attitude. We will consider only small attitude changes and assume that RRRobot translation has no effect on the body roll-pitch-yaw dynamics. Thus, we will fix the body center for the roll-pitch-yaw dynamics analysis. After a review of related work in Section 1.3, the body of the paper will explore two simplified models, one in which the body motion is limited to yaw (Section 2) and one in which the body motion is limited to roll and yaw (Section 3). Section 4 presents a brief discussion of how attitude oscillations, when coupled with kinematic contact constraints, can lead to RRRobot translation.

1.3 Related Work

The locomotion strategy proposed here for RRRobot involves the interplay between roll-pitch-yaw dynamics and the kinematic nonholonomic constraints associated with contact. Numerous investigators have studied dynamic systems with constraints. Lewis et al. [9] discuss the constrained mechanics of the *Snakeboard,* a modified version of the skateboard in which the wheel directions can be controlled (see Fig. 6). The snakeboard rider locomotes by twisting his body back and forth, while simultaneously moving the wheels with the proper phase relationship. Lewis et al. present numerical simulations of snakeboard locomotion using characteristic wheel motions. In studying the RRRobot dynamics, we will use the Lagrangian method, which involves computing the Lagrangian and taking its derivatives [6]. Zenkov et al. [19] discuss the energy-momentum method for control of dynamic systems with nonholonomic constraints, such as the rattleback, the roller racer (see Fig. 7), and the rolling disk.

RRRobot's configuration space may be split up into two parts- the sphere's position and orientation with respect to a spatial coordinate frame (called the *group space*) and the two joint angles (called the internal configuration or the *base space*). Note that the sphere's position and orientation are not actuated, while the base space is actuated. Ostrowski [14] presents a general framework for studying systems where the robot's position and orientation can be represented as a group. Since only the base space is actuated, Ostrowski finds a *connection* relating the base space velocities to the group space velocities; but he presents results only for wheeled systems with constant inertias such as the snakeboard and the Hirose snake. Note that RRRobot has a spherical contact with the plane, its inertias change with configuration, and the mass matrix is coupled. Further analysis of the RRRobot system using the Ostrowski approach is required.

RRRobot locomotes by rolling its round body without slip on the planar surface. The curvatures of the two surfaces, the type of contact between the two surfaces, etc., determine the kinematic constraints and, hence, the relative motion between the two bodies [11]. Li et al. [10] discuss the motion of two rigid bodies with rolling contact. Given the geometry of the two bodies, they use Chow's theorem [5] to establish the existence of a path between any two configurations and, then, propose a kinematic algorithm for finding the path, given that one of the objects is flat. Camicia [4] provides an analysis of the nonholonomic kinematics and dynamics of the *Sphericle* [2], a hollow ball driven on a planar surface by an unicycle placed inside. The Sphericle and RRRobot have similar nonholonomic contact constraints with the surface they rest on. Nonholonomic constraints involve both configuration and rate variables (such as velocities), while holonomic constraints involve only configuration variables (see [13] for more details).

If we consider RRRobot to be floating in space and ignore the surface it is resting on, controlling body attitude using halteres is similar to Fernandes et al. [7], which discusses near-optimal nonholonomic motion planning for a system of coupled bodies using Lagrangian dynamics. In particular, they plan the motion for a falling cat, considering two types of joints between the upper body and the lower body: a ball and socket joint and an universal joint. Given an arbitrary starting point, the goal is to land the falling cat on its feet, subject to the nonholonomic constraint that angular momentum is conserved.

RRRobot's body attitude is not directly actuated, but its inertias change with leg position. By repetitively wiggling the legs while exploiting the differences in angular inertia, RRRobot may be able to adjust its orientation (see Section 2.3 for more details). In contrast, in the case where spinning reaction wheels are used to control satellite attitude, the inertias of the satellite system do not change with rotation of reaction wheels. Rui et al. [15] discuss nonlinear control of spacecraft attitude with reaction wheels and present a controllability analysis and motion planning approaches for maneuvers. Using small perturbation-like motions of a six degrees of freedom manipulator on board, Suzuki et al. [17] plan the orientation of a satellite approximately and create spiral motions of the satellite-manipulator system in nine dimensional space.

2 The Yaw model

We now study the interaction between leg motions and body dynamics. Since we consider only small oscillations in RRRobot attitude, we begin by assuming that the body center does not translate much and, thus, ignore the dynamic effects of translation.

To understand the roll-pitch-yaw dynamics, we begin by exploring a simplified version of RRRobot,

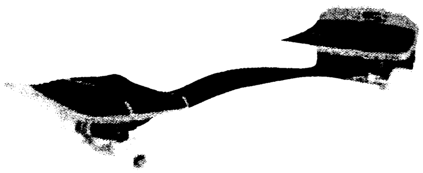


Figure 6: The snakeboard.

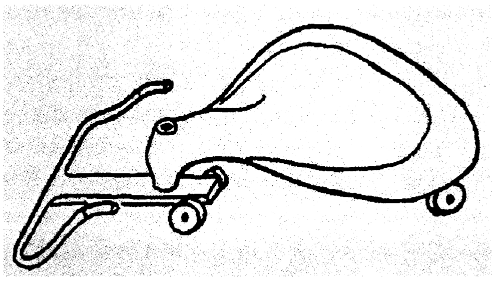


Figure 7: The roller racer.

one that only yaws (see Fig. 8). The body center B is fixed, and there is no gravitational field. We place a passive pin joint at B with its axis aligned along the yaw axis, and body yaw configuration is represented by θ_2. Body roll and pitch are fixed at zero. Leg 1 joint angle is represented by ϕ_1, leg 2 joint angle by ϕ_2, and there are no joint limits. The masses represented by black dots in Fig. 8 include three arranged symmetrically on the body and one at the distal end of each leg. Each body mass has value m_m, and each leg mass has value $m_l < m_m$. Torques τ_1 and τ_2 can be applied at leg joints 1 and 2, and all links are rigid.

2.1 The Yaw Model mathematical details

The equations of motion for the yaw model take the form

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} \;=\; \begin{pmatrix} 0 \\ \tau_1 \\ \tau_2 \end{pmatrix}, \tag{1}$$

where $q = (\theta_2, \phi_1, \phi_2)^T \in \mathbb{S}^1 \times \mathbb{S}^1 \times \mathbb{S}^1$ represents the configuration of the robot at any instant,

$M(q) = \begin{pmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{pmatrix}$,

$m_{11} = 2m_m b^2 + \frac{1}{2}m_l(2(2b^2 + l^2)l^2(\cos 2\phi_1 + \cos 2\phi_2))$,

$m_{12} = m_l lb \sin\phi_1$,

$m_{13} = m_l lb \sin\phi_2$,

$m_{21} = m_l lb \sin\phi_1$,

$m_{22} = m_l l^2$,

$m_{23} = 0$,

$m_{31} = -m_l lb \sin\phi_2$,

$m_{32} = 0$,

$m_{33} = m_l l^2$,

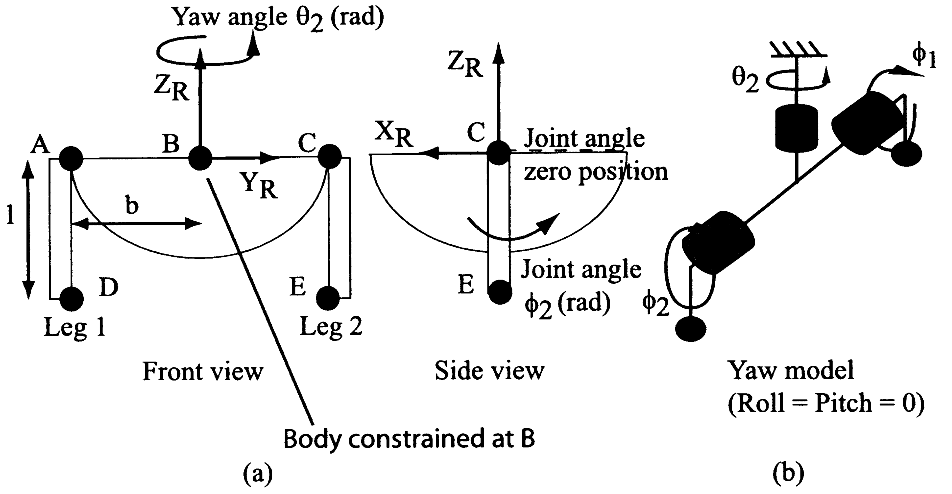

Yaw angle θ2 (rad)
ZR
A
B
C
YR
b
l
D
E
Leg 1
Leg 2
Front view
Body constrained at B
(a)
ZR
XR
C
Joint angle zero position
E
Joint angle ϕ2 (rad)
Side view
θ2
ϕ1
ϕ2
Yaw model (Roll = Pitch = 0)
(b)

Figure 8: (a) A simplified RRRobot model and (b) the yaw model as a kinematic chain.

and $C(q,\dot{q})\dot{q} = \begin{pmatrix} m_l l^2\dot{\theta}(\dot{\phi}_1 \sin 2\phi_1 + \dot{\phi}_2 \sin 2\phi_2) + m_l l b(\dot{\phi}_1^2 \sin\phi_1 - \dot{\phi}_2^2 \sin\phi_2) \\ -\frac{1}{2} m_l l^2 \dot{\theta}^2 \sin 2\phi_1 \\ -\frac{1}{2} m_l l^2 \dot{\theta}^2 \sin 2\phi_2 \end{pmatrix}$.

$M(q)$ is the Yaw Model's positive definite mass matrix, and $C(q,\dot{q})\dot{q}$ is the vector consisting of Coriolis and centrifugal terms. Row 1 of (1) implies that body yaw is not directly actuated and is influenced by leg motions only. Trivially, we infer that if the leg velocities and accelerations are zero, then the body yaw velocity and acceleration must also be zero.

Equation 1 can be rewritten as

$$\dot{x} \;=\; f(x) + g_1(x)\tau_1 + g_2(x)\tau_2, \tag{2}$$

where

$$\begin{aligned} f(x) &= \begin{pmatrix} \dot{q} \\ -M^{-1}(q)C(q,\dot{q})\dot{q} \end{pmatrix}, \\ g_1(x) &= \begin{pmatrix} 0 \\ 0 \\ 0 \\ M_2^{-1}(q) \end{pmatrix}, \\ g_2(x) &= \begin{pmatrix} 0 \\ 0 \\ 0 \\ M_3^{-1}(q) \end{pmatrix}. \end{aligned}$$

Here, $x = \begin{pmatrix} q \\ \dot{q} \end{pmatrix} \in \mathbb{S}^1 \times \mathbb{S}^1 \times \mathbb{S}^1\mathbb{R}^3$ represents the state of the system, and $\dot{x} = \begin{pmatrix} \dot{q} \\ \ddot{q} \end{pmatrix} \in \mathbb{R}^6$ represents the rate of change of the state of the system. The matrix $M^{-1}(q)$ is the inverse of $M(q)$, and $M_i^{-1}(q)$ represents the i^{th} column of $M^{-1}(q)$.

2.2 The Yaw Model formal analysis

We now present a theoretical analysis of the Yaw Model. In control systems literature, (2) is said to be in state-space form, and we can compute the rate of change of the state of the system, given its current state and control inputs. Note that the state of the yaw model x includes both configuration and velocity terms, and the rate of the change of the state of the system $\dot{x}$ includes the velocity and acceleration terms. The vectors g_1 and g_2 are called control vector fields, corresponding to the controls τ_1 and τ_2, and f is called the drift vector field. The control vector field g_i represents the acceleration of the robot when subject to an unit torque at leg joint i and zero torque at the other leg joint when the system is at rest ($\dot{q} = 0$). The drift vector field f represents the motion of the robot in the absence of the torques; it consists of the Coriolis and centrifugal terms and is zero when $\dot{q} = 0$. Note that the last three rows in (2) is the same as (1) with some terms rearranged.

We can explore the yaw model, a nonlinear system that includes a drift vector field, by studying the vector fields f, g_1, and g_2 and their filtration [18]. The filtration of the vector fields f, g_1, and g_2 is defined as the sequence $\{G_i\}$, such that

$$\begin{aligned} G_1 &= \operatorname{Span}\{f, g_1, g_2\}, \\ G_i &= G_{i-1} + \operatorname{Span}\{[G_1, G_{i-1}]\}, i > 1, \end{aligned}$$

where $[G_i, G_j]$ represents the Lie bracket between two distributions G_i and G_j. Each G_i is, thus, spanned by the input vector fields plus the vector fields formed by taking up to $i - 1$ Lie brackets of the input vector fields. For example,

$$\begin{aligned} [G_1, G_1] &= \{[f,f],[f,g_1],[f,g_2],[g_1,f],[g_1,g_1],[g_1,g_2],[g_2,f],[g_2,g_1],[g_2,g_2]\}, \\ &= \{[f,g_1],[f,g_2],[g_1,f],[g_1,g_2],[g_2,f],[g_2,g_1]\}, \\ G_2 &= \operatorname{Span}\{f, g_1, g_2, [g_1,f],[g_2,f]\} \subset \mathbb{R}^6, \end{aligned}$$

since, for vector fields u and v, $[u, u] = 0$ and $[u, v] = -[v, u]$ by definition.

Observations on the Yaw model

1. The Lie bracket $[g_1, g_2]$ is identically zero. Since $[g_1, g_2]$ is zero, we can ignore it in future analysis. But the Lie brackets $[g_i, f]$ are non-zero and give us new directions in the tangent space. Thus, interaction between the control and drift vector fields is important for yaw model motion.

2. Rank(G_2) is 5, and the constraint direction on the distribution G_2 is

$$N(x) = \begin{pmatrix} 0 \\ -\frac{l}{b}\dot{\theta}\sin 2\phi_1 - \dot{\phi}_1 \sin 2\phi_1 \\ -\frac{l}{b}\dot{\theta}\sin 2\phi_2 + \dot{\phi}_2 \sin 2\phi_2 \\ \frac{m_l l^2(\cos 2\phi_1 + \cos 2\phi_2) + 2m_l l^2 + 4b^2(m_l + m_m)}{2m_l l b} \\ \sin\phi_1 \\ -\sin\phi_2 \end{pmatrix}. \tag{3}$$

   $N(x)$ is a vector that is not in the span of G_2 and is perpendicular to each vector field in G_2. Note that the fourth element of $N(x)$ is never zero, implying that there is always a constraint along the yaw acceleration direction.

   Fig. 9 shows the variation of the condition number [8] of the distribution G_2 when the legs move between extremes in alternating time intervals. We note that the condition number of the matrix G_2

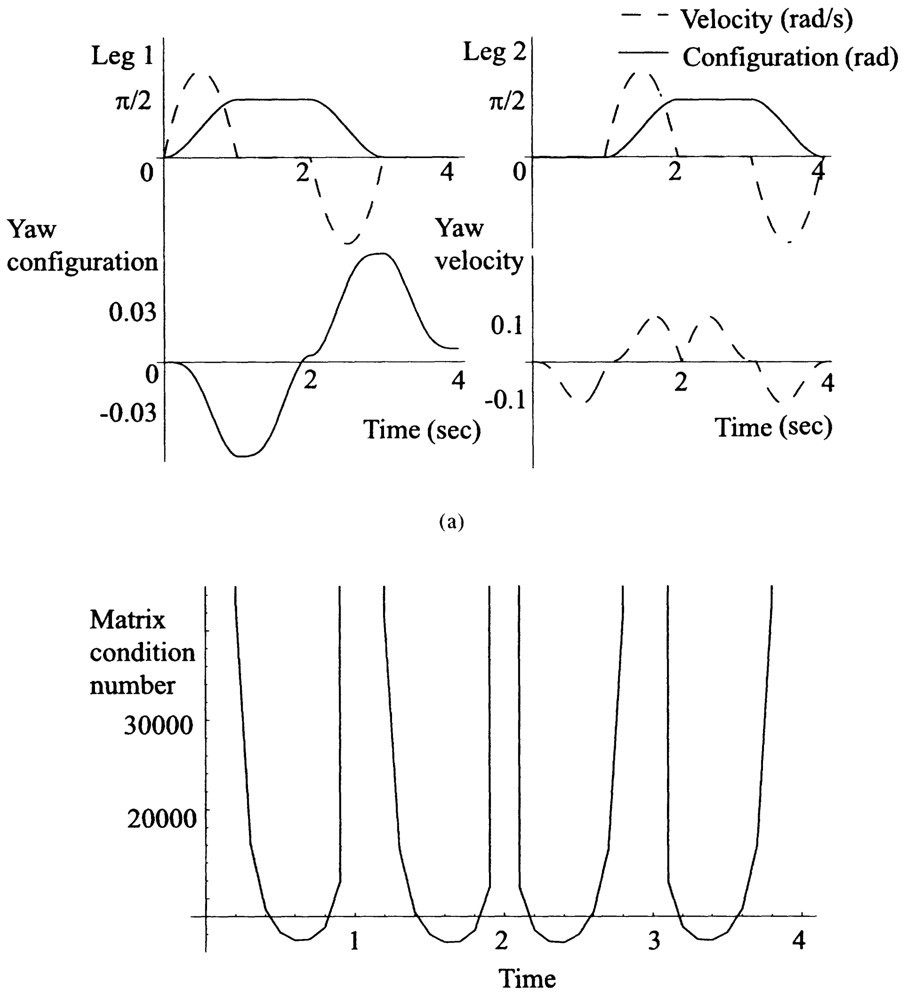


Figure 9: (a) Interleaved leg motions and (b) the variation of the condition number of the controllability matrix G_2.

increases as the velocity decreases (e.g., during the interval $t = 0.5$ to $t = 1.0$ when leg 1 decelerates to a halt at $\pi/2$), indicating that the yaw system is closer to dropping rank from five to four as the velocities decrease. When the system is at rest, the condition number is infinity, and Rank(G_2) is 4.

3. It can be shown that all the second degree Lie brackets in G_3 (e.g., $[f, [g_1, f]]$) are perpendicular to $N(x)$ by verifying that their dot products with $N(x)$ is zero. This implies that the second degree Lie brackets do not have any component along $N(x)$ and, thus, are spanned by G_2. Thus, the span of G_2 is the same as the span of G_3, and we can do all analysis of the yaw model using just the distribution G_2.

4. From 3, we deduce that the tangent space of the yaw model T_x is the space spanned by G_2. Since

$\dot{x} \in T_x$,

$$N(x) \cdot \dot{x} = 0. \tag{4}$$

Note that (4) is in Pfaffian form [13] and is identical to row 1 in (1). This constraint gives a relationship between the yaw velocity, yaw acceleration, the joint configurations, and their derivatives that must be satisfied.

Interestingly, since the dimension of the filtration is never greater than five, and the yaw system is embedded in a six-dimensional space, we can conclude that the yaw system has an algebraic constraint. For example, when the legs are horizontal and have zero velocity ($\phi_1 = \phi_2 = 0$), using (3), we compute

$$N(\theta_2, \phi_1 = 0, \phi_2 = 0, \dot{\theta}_2, \dot{\phi}_1, \dot{\phi}_2) = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \frac{2ml^2+2b^2(m_l+m_m)}{m_l lb} \\ 0 \\ 0 \end{pmatrix}.$$

Note that the only non-zero element in $N(x)$ is along the yaw acceleration direction; we infer that yaw velocity is fixed when the legs are horizontal.

2.3 Changing body yaw using leg motions

In this section, we develop an intuitive physical understanding of how the body yaws when the legs are moved. In the next section, we will use this insight to create body roll-yaw motions in a simplified version of RRRobot that is constrained to only roll and yaw.

The yaw model is so simple that we can understand its motion using the law of conservation of angular momentum. Suppose the system is at rest at $t = 0$, and the yaw angular momentum of the system about the body center is zero. We keep leg 2 fixed and apply a torque to leg 1 to move it from $\phi_1 = 0$ (horizontal position) to $\phi_1 = \pi/2$ (vertical position) so that the leg mass possesses negative yaw angular momentum about the body center (positive yaw angular momentum direction is vertical and upwards) . Since angular momentum must be conserved in the absence of externally applied moments about the yaw axis, the body starts yawing in a positive sense. Now, suppose the leg is decelerated while it approaches the vertical configuration by applying a torque at the joint. Then, the magnitude of the leg's yaw angular momentum about the body center decreases. Again, since angular momentum is conserved, the body yaw velocity also decreases, and the body comes to a halt when the leg stops in the vertical configuration. Thus, we understand how the body moves when one leg is moved, while the other leg is stationary. But this leg motion does not produce net body yaw; when the leg is moved back to original position, the body also returns to its original position.

We will now explore an important property of the yaw model that will help us produce net body yaw: the yaw inertia of the yaw model changes with leg configuration. For example, the yaw inertia is maximum when the legs are horizontal and is minimum when the legs are vertical. Since yaw inertia influences yaw motion, leg configuration affects yaw motion. Suppose the body yaw is ϵ_1 when leg 1 is kept fixed in the horizontal position and leg 2 is moved from a horizontal to vertical position. Similarly, let the body yaw be ϵ_2 when leg 1 is kept fixed in the vertical position and leg 2 is moved from a horizontal to vertical position. The magnitude of ϵ_1 will be lesser than ϵ_2, since the yaw inertia of the system is greater when leg 1 is horizontal. This dependence of yaw inertia on the leg configuration may be exploited while designing leg motions or gaits to modify yaw configuration.

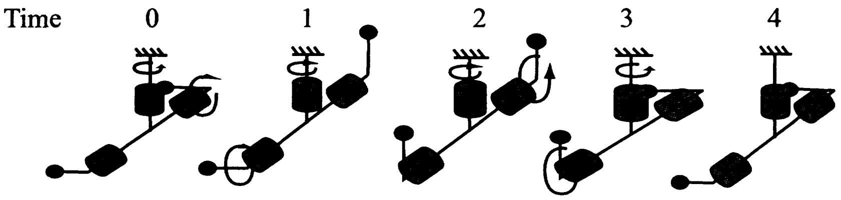


Figure 10: Lie bracket-inspired leg motions to change body yaw.

Time interval	$\phi_1(t)$	$\phi_2(t)$	Change in yaw
0–1	$0 \rightarrow \pi/2$	0	ϵ_1
1–2	$\pi/2$	$0 \rightarrow \pi/2$	$-\epsilon_2$
2–3	$\pi/2 \rightarrow 0$	$\pi/2$	$-\epsilon_2$
3–4	0	$\pi/2 \rightarrow 0$	ϵ_1
Net change in yaw			$2(\epsilon_1 - \epsilon_2)$

Table 2: Incremental motion of the yaw model.

A simple strategy to achieve net yaw is to use interleaved leg motions. We will move each leg back and forth between extremes of 0 and $\pi/2$ rad. Each leg will dwell at the extreme for one second, and will take one second to transition between angles following a cubic spline, achieved using a simple proportional-derivative controller. The result is a smoothed square wave, with the two legs $\pi/2$ rad out-of-phase (see Fig. 10).

This Lie bracket-inspired out-of-phase smoothed square wave trajectory produces net yaw, as shown in Table 2. This result can be easily confirmed by studying the table and thinking about the angular inertia of the system. Now, suppose the magnitude of body yaw change is ϵ_1 during the interval $t = 0$ to $t = 1$ and is ϵ_2 during the interval $t = 1$ to $t = 2$. The net yaw during the two motion segments is different, because the angular inertia varies depending on whether the leg is stretched out or tucked in, and this difference produces net yaw at the end of the motion sequence. We get maximal yaw motion when the legs move between configurations of minimal and maximal yaw inertias. Since the legs come to rest after each motion, the kinetic energy of the system after each motion segment is zero. Also, we get reversed yaw motion if leg 2 is moved before leg 1.

3 The Roll-Yaw model

In Section 1.2, we presented a set of roll-yaw body oscillations that produce RRRobot locomotion in the plane. To help understand how to create such body attitude oscillations, the previous section addressed the yaw model to find leg motions that can effect change in the yaw configuration, while the pitch and roll configuration were constrained to be zero. In this section, we will study another simplified version of RRRobot, one that can only roll and yaw (see Fig. 11), and find leg motions that produce roll-yaw oscillations. As in the yaw model discussed in Section 1.2, we will ignore the effect of the body translation on the body roll-yaw dynamics.

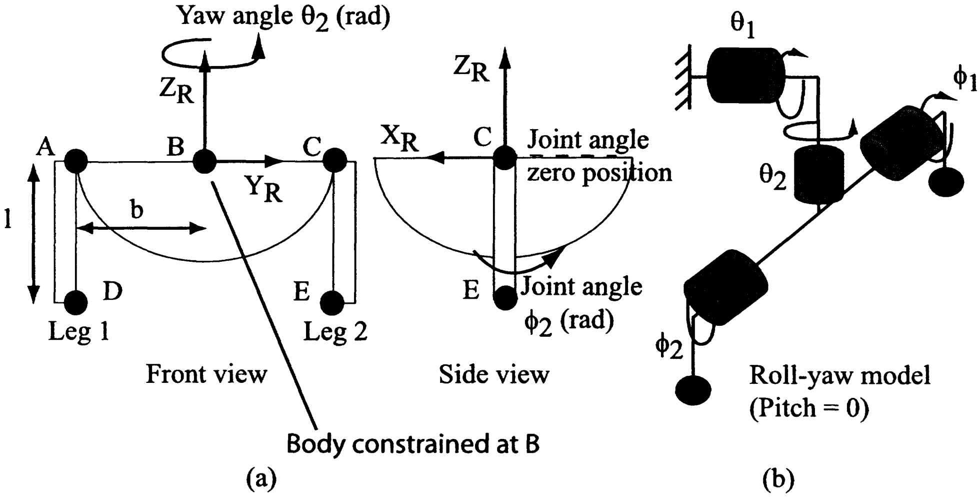


Figure 11: (a) The simplified RRRobot model and (b) the roll-yaw model as a kinematic chain.

3.1 The Roll-Yaw model details

The roll-yaw model (see Fig. 11) treated in this section is the same as the yaw model except for the additional roll freedom. Configuration is now given by $q = (\theta_1, \theta_2, \phi_1, \phi_2)^T \in \mathbb{R}^4$, where θ_1 and θ_2 represent the roll and the yaw of the system. We constrain the body by placing an universal joint at B with axes aligned along the roll and yaw directions. The body pitch configuration is constrained to be zero, and we will assume small roll and yaw motions. The mass matrix of the system is $M(q) \in \mathbb{R}^{4\times4}$ and the vector containing velocity-related terms is $C(q,\dot{q})\dot{q} \in \mathbb{R}^4$. Note that we can follow a similar analysis to that used in Section 2.3 to conclude that the robot can locally adjust $\theta_1, \theta_2, \phi_1$, and ϕ_2.

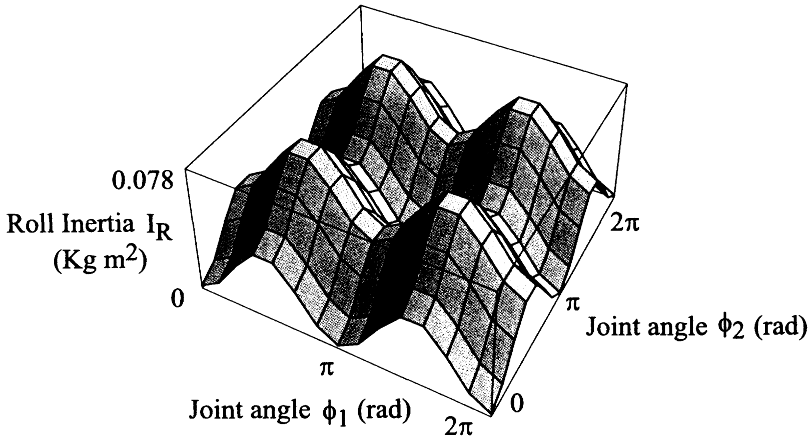
In all simulations of the roll-yaw model, the robot starts from rest. To approximate RHex's specifications, we set $m_m = 3$ kg, $m_l = 0.2$ kg, $l = 0.1$ m, and $b = 0.2$ m. We approximate the universal joint at B by using two passive single degree-of-freedom joints: one along the roll axis, and the other along the yaw axis. Since all attitude motions are small, this approximation is valid. Note that the roll inertia $I_R(\phi_1, \phi_2)$ is minimal when the legs are stretched out and maximal when the legs are tucked in; this is opposite for the yaw inertia $I_Y(\phi_1, \phi_2)$ (see Fig. 12).

3.2 Roll-Yaw model simulations

We present three patterns of leg motions or *gaits* that produce roll-yaw motions. Gaits 1 and 2 are interleaved leg motions similar to those shown in Fig. 10, and Gait 3 uses sinusoidal leg trajectories.

3.2.1 Gait 1

In Gait 1, the legs move between angles of 0 and $\pi/4$ rad (see Fig. 13), and this produces predominantly positive yaw motion (see Fig. 14a). Note that the kinetic energy of the system at the end of each gait cycle is zero. If the order of leg motions is swapped, we get roll-yaw motion along the opposite direction; call this variation Gait 1′ (see Fig. 14b).



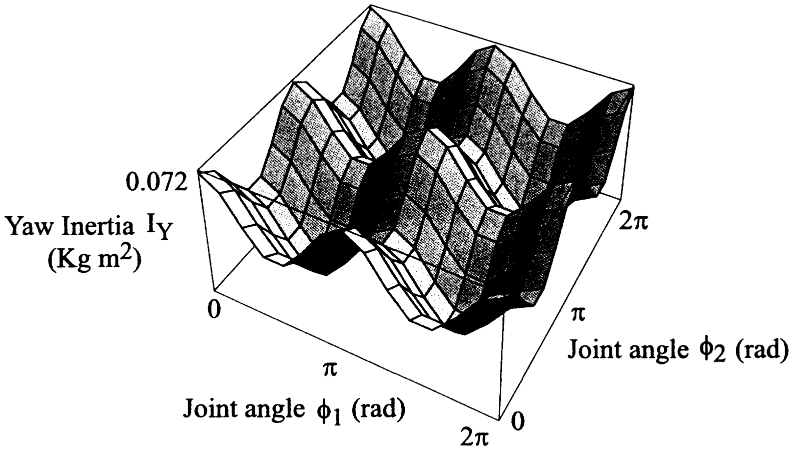


Figure 12: Variation of RRRobot roll and yaw inertias with leg configuration.

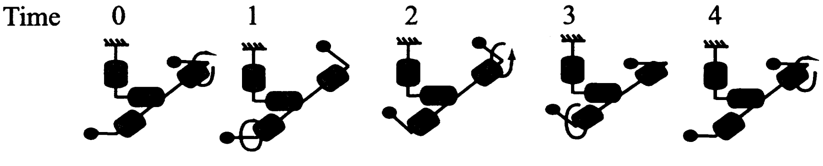


Figure 13: Gait 1 leg motions.

3.2.2 Gait 2

In Gait 2, the legs move between angles of $\pi/2$ and π rad (see Fig. 15), and this produces positive roll and negative yaw motion (see Fig. 16a). Call the variation obtained by swapping start and extreme positions Gait 2′ (see Fig. 16b).

In the absence of gravity, we can incrementally build up RRRobot's roll and yaw using many cycles of Gaits 1, 2, 1′, and 2′. We can interleave these gaits to create RRRobot body attitude motions similar to the oscillations in Fig. 5, which cause RRRobot to translate in the XY-plane. (Although some leg motion is required to paste the gaits together, the net effect on roll and yaw is zero.)

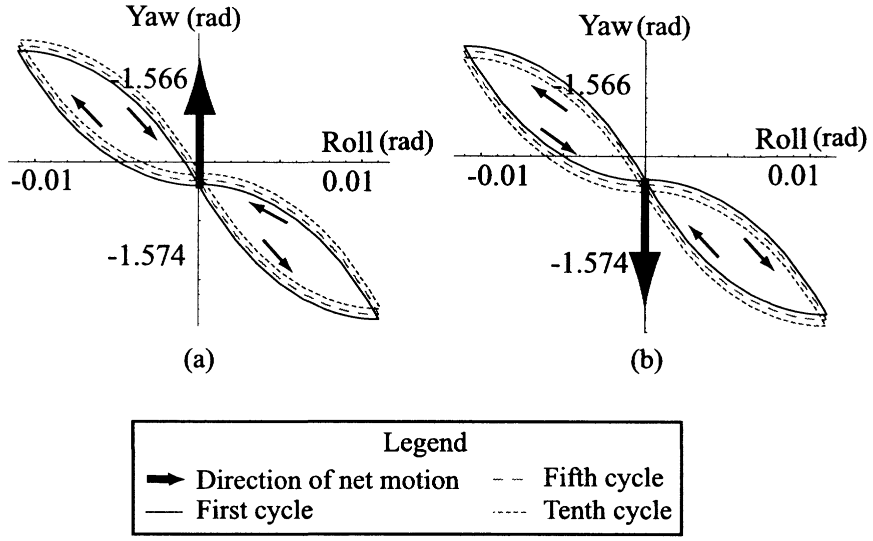


Figure 14: Body attitude oscillations and net roll and yaw produced by (a) Gait 1 and (b) Gait 1′ over ten cycles.

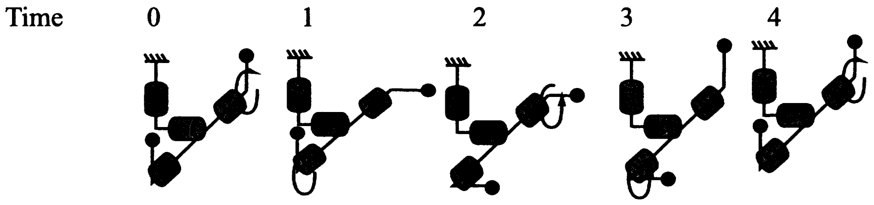


Figure 15: Gait 2 leg motions.

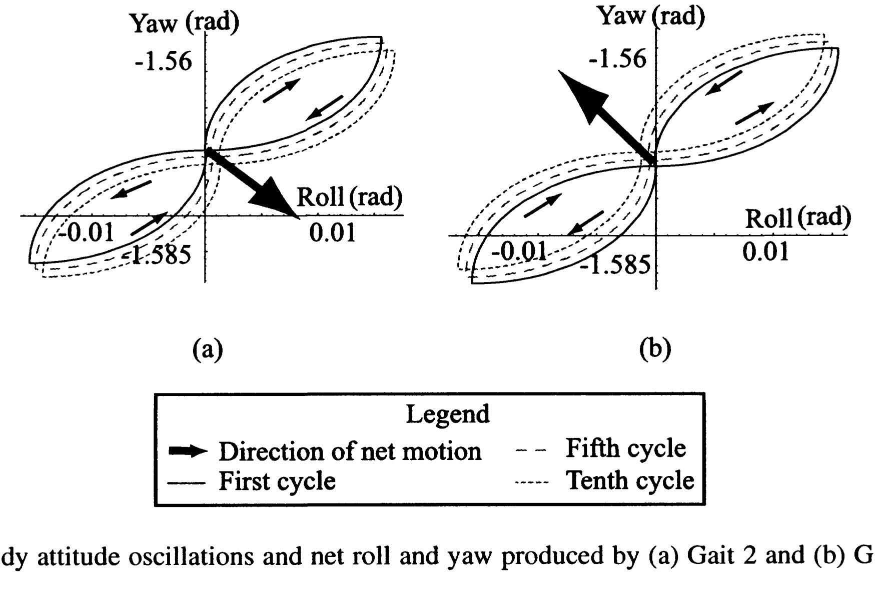


Figure 16: Body attitude oscillations and net roll and yaw produced by (a) Gait 2 and (b) Gait 2′ over ten cycles.

	Leg 1	Leg 2
Amplitude a	$\frac{\pi}{8}$	$\frac{\pi}{8}$
Frequency ω	5	5
Phase β	0	$\frac{\pi}{2}$
Offset angle γ	0	$\frac{\pi}{2}$
Damping k	-1	-1

Table 3: Gait 3 parameters.

3.2.3 Gait 3

Interleaving Gaits 1, 2, $1'$, and $2'$ produces roll-yaw oscillations in the roll-yaw model by incrementally building up large roll or yaw angles. But this is not practical for RHex or RRRobot, because, after each leg motion, the body would come to rest. It is impossible to maintain a fixed roll configuration in the presence of gravity, since the body would rock back to its equilibrium configuration. A more practical approach would be to use small continuous oscillations in roll and yaw.

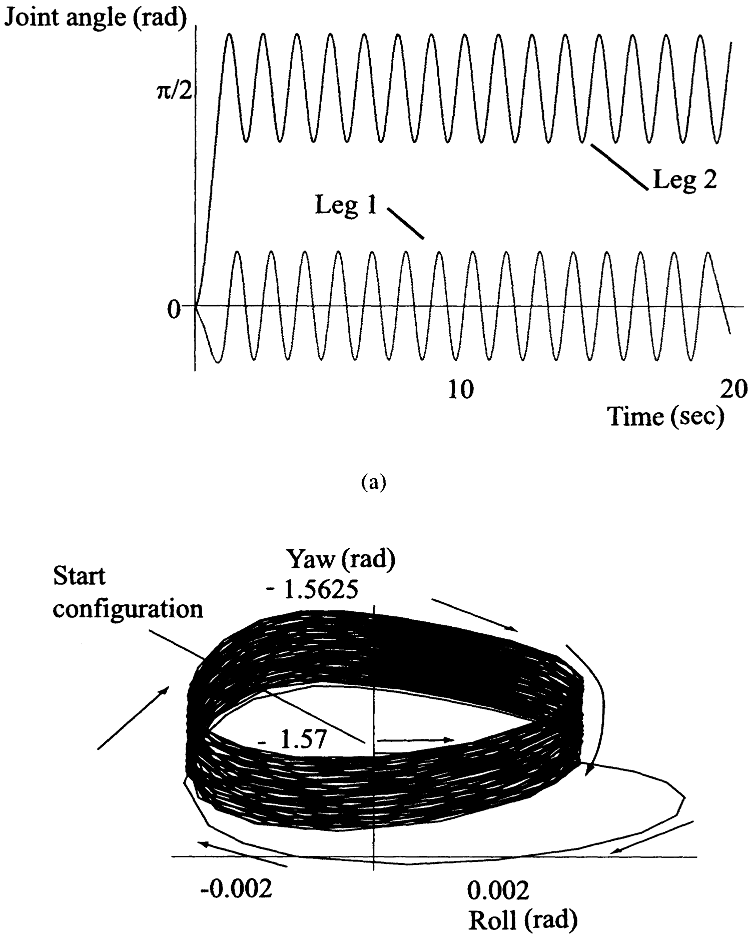
Table 3 shows the parameters of Gait 3's sinusoidal leg trajectories of the form $a\sin(\omega t + \beta) + \gamma$. Note that the amplitude and frequency of oscillations is the same for both legs, but the legs are $\pi/2$ out-of-phase with each other. The 'home' position for leg 1 is the horizontal position, while the 'home' for leg 2 is the vertical position. An envelope function ensures continuity in $(\dot{\phi}_1, \dot{\phi}_2)$ space, and the joint angles follow sinusoidal paths after ramping from zero velocity in 1 sec. To eliminate any drift in roll and yaw due to initial transients, we add damping of the form $-k_i\dot{\theta}_i, (i = 1, 2; k_i \in \mathbb{R} > 0)$ to the roll and yaw axis equations. Under Gait 3, the body settles down into a steady *rocking* pattern after the initial transient phase (see Fig. 17). This motion is similar to the roll-yaw attitude oscillations that produce RRRobot translation (see Fig. 5 and Section 1.2).

4 Discussion

Suppose RRRobot's pitch configuration changes are negligible compared to roll-yaw changes. Then, RRRobot on a block is similar to a unicycle on a plane, with yaw corresponding to the unicycle direction and body roll corresponding to the unicycle forward motion. We can approximately compute RRRobot XY-plane translation kinematically using:

$$\Delta X = \int \nu \sin\theta_2 dt,$$
$$\Delta Y = \int -\nu \cos\theta_2 dt, \qquad (5)$$

where $\nu = b\dot{\theta}_1$ is the body center velocity. We hypothesize that roll-yaw attitude oscillations produced using Gait 3 (see Section 3), when coupled with the nonholonomic contact constraints, will produce translation as in Fig. 5. More experiments are necessary to verify this on RRRobot and RHex. If true, eighty cycles of Gait 3 will produce a few millimeters of translation after about 100 secs (measured using (5)). Note that there is still some drift in the yaw direction under Gait 3. This may mean that RRRobot does not travel along a straight line, but along curved paths. Further analysis is required to control the direction of translation.



(a)

(b)

Figure 17: Gait 3 over eighty cycles: (a) Leg trajectories and (b) looping curves in roll-yaw space.

5 Conclusion

We have presented a set of models which, when superimposed under suitable conditions, may produce locomotion for high-centered round-bodied legged robots. Using a high-centered RHex as motivation, we studied simplified dynamic models of a prototype, RRRobot, to create body attitude oscillations using leg motions. Then, we hypothesized that these attitude oscillations under slip-free contact constraints may result in locomotion. More experiments are necessary to verify this on RRRobot and RHex. We also presented simple thought experiments and control theory analysis to understand motion in the analytical models. Although we have given just a few patterns of leg motions or gaits for creating attitude oscillations, we believe that more optimized body motion through automatic gait generation is possible. Interesting future work could include analyzing how the shape and properties of the robot body affect this locomotion technique. Also, we have discussed using just two legs to change body attitude; using all six legs of RHex may produce richer body motion.

Acknowledgments

This work was supported under NSF IIS 0082339, NSF IIS 0222875, and DARPA/ONR N00014-98-1-0747 contracts. We thank Devin Balkcom, George Kantor, and Uluc Saranli for insightful discussions on the dynamics and nonholonomy of mechanical systems.

References

[1] J. Bares and D. Wettergreen. Dante ii: Technical description, results and lessons learned. *International Journal of Robotics Research*, 18(7):621–649, July 1999.

[2] A. Bicchi, A. Balluchi, D. Prattichizzo, and A. Gorelli. Introducing the sphericle: an experimental testbed for research and teaching in nonholonomy. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 2620–2625, 1997.

[3] M. Buehler, R. Battaglia, A. Cocosco, G. Hawker, J. Sarkis, and K. Yamazaki. Scout: A simple quadruped that walks, climbs and runs. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 1998.

[4] C. Camicia, F. Conticelli, and A. Bicchi. Nonholonomic kinematics and dynamics of the sphericle. In *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, pages 805–810, 2000.

[5] W. Chow. Uber systeme von linearen partiel-len differentialgleichungen ester ordnung. *Math. Ann.*, 117:98–105, 1940.

[6] J. J. Craig. *Introduction to Robotics*. Addison Wesley, 1989.

[7] C. Ferandes, L. Gurvits, and Z. Li. Near optimal nonholonomic motion planning for a system of coupled rigid bodies. *IEEE Transactions on Automatic Control*, 1994.

[8] Golub and V. Loan. *Matrix Computations*. Johns Hopkins University Press, 1993.

[9] A. Lewis, J. Ostrowski, R. Murray, and J. Burdick. Nonholonomic mechanics and locomotion: The snakeboard example. In *Proceedings of the International Conference on Robotics and Automation*, pages 2391–2397, 1994.

[10] Z. Li and J. Canny. Motion of two rigid bodies with rolling constraint. *IEEE Transactions on Robotics and Automation*, 6(1):62–72, Feb. 1990.

[11] A. Marigo and A. Bicchi. Rolling bodies with regular surface: Controllability theory and applications. *IEEE Transactions on Automatic Control*, 45(9):1586–1599, Sept. 2000.

[12] T. McGeer. Passive dynamic walking. *International Journal of Robotics Research*, 1990.

[13] R. M. Murray, Z. X. Li, and S. S. Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, 1994.

[14] J. P. Ostrowski. *The Mechanics and Control of Undulatory Robotic Locomotion*. PhD thesis, California Institute of Technology, 1996.

[15] C. Rui, I. V. Kolmanovsky, and N. H. McClamroch. Nonlinear attitude and shape control of spacecraft with articulated appendages and reaction wheels. *IEEE Transactions on Automatic Control*, 45(8):1455–69, Aug. 2000.

[16] U. Saranli, M. Buehler, and D. E. Koditschek. Rhex: A simple and highly mobile hexapod robot. *International Journal of Robotics Research*, 20(7):616–631, July 2001.

[17] T. Suzuki and Y. Nakamura. Planning spiral motion of nonholonomic space robots. In *Proceedings of the International Conference of Robotics And Automation*, 1996.

[18] V. S. Varadarajan. *Lie Groups, Lie Algebras, and their Representations.* New York Springer Verlag, 1984.

[19] D. Zenkov, A. Bloch, and J. Marsden. The energy-momentum method for the stability of nonholonomic systems. Technical report, California Institute of Technology, 1997.