Chapter 4

More General 2-D Walkers

This chapter is an updated version of a paper by Mariano Garcia, Anindya Chatterjee, and Andy Ruina entitled "Efficiency, Speed, and Scaling of Passive Dynamic Bipedal Walking." It was submitted to Dynamics And Stability of Systems on July 28, 1998.

My role in this paper was as follows: I concocted all of the models and equations, and did all of the simulation and data collection, including finding gait cycles and analyzing them. Results from Chapter 3 suggested the possibility of period-doubling, which I found. Andy Ruina and I came up with the tuning criteria during a discussion about zero-slope kneed walkers. Anindya Chatterjee thought up the explanation for the different scaling rules and the derivation of the transition slope in the short-step gait.

4.0.3 Abstract

We address some performance limits of the two-dimensional passive-dynamic walking machines discovered by Tad McGeer. Energetic *in*efficiency is measured by downhill ground-slope γ needed to sustain gait, with $\gamma = 0$ being perfectly efficient. Speed is measured by the Froude number. We present some necessary conditions on the walker mass distribution to achieve perfectly efficient (zero-slope capable) walking for both kneed and straight-legged models. Our numerical investigations indicate that, consistent with a previous semi-analytical study of a simpler model, such walkers have two distinct gaits at arbitrarily small ground-slopes, for which only the longer-step gait is sometimes stable. Energy dissipation can be dominated by a term proportional to $(\text{speed})^2$ from tangential foot velocity at heelstrike and from kneestrike, or a term proportional to $(\text{speed})^4$ from normal foot collisions at heelstrike, depending on the gait, ground-slope, and walker design. For all zero-slope capable straight-legged walkers, the long-step gaits have no tangential foot velocity at heelstrike and are hence especially fast at low power or low ground-slope. A period doubling route to chaos is also numerically demonstrated for a kneed walker.

4.1 Introduction

Since humans and some potentially-useful robot designs use legged walking motions, it is interesting to consider the limits of possible performance of bipedal walking machines.

One natural modeling approach would be to consider the optimal performance of powered and controlled machines. Because animal nerve systems are so capable, because the energetic cost of thinking is so low, and because minimizing food use is advantageous, an energy-based optimization approach is likely to capture much of how people move (for example, see Beckett and Chang (1973), Alexander (1980), and Alexander (1991)). Energetic efficiency is one obvious goal of both biological and artificial locomotion and transportation systems. Other possible optimization criteria for consideration in the description of animal motion include peak muscle force, minimum-jerk, minimum-stress, maximum speed, etc., as discussed by Collins (1995), Nelson (1983), and Hatze (1989). The results of such optimizations, like the results of animal evolution, will probably often show limited use of muscles in walking, as shown in EMG studies by Basmajian and Tuttle (1973). So, given the uncertainties and complexities of many-degree-of-freedom optimization studies and the likely prediction of small muscle-usage (see Yamaguchi and Zajac (1990) for example), there is hope for insight from simpler approaches.

Although some motor activity is needed for walking, perhaps it can be neglected in some analyses like engine power can be neglected for much of the study of airplane flight, as argued by McGeer (1990a). A simple energy source (gravity) could then be used as a proxy for the small but essential muscle use of humans (or motor use in efficient robots). It is hoped, as must be ultimately checked, that many results thus obtained will be insensitive to the choice of the energy source. However, the use of gravity as an energy source (as opposed to a simple motor approximation) eliminates some arbitrariness, and simplifies simulation and physical experimental verification.

Here, we address walking performance issues in the context of gravity-powered walking machines, also mentioning some other properties of these machines.

4.2 Passive Dynamic Walking Machines

Passive dynamic walking machines that walk on shallow ground-slopes were first designed, simulated and built by Tad McGeer, who was inspired by the "ballistic" double- and triple-pendulum leg models proposed by Mochon and McMahon (1980). The McGeer-like passive-dynamic walking machines consist of hinged rigid bodies that make collisional and rolling contact with a sloped, rigid ground surface.

The two-dimensional kneed walking machine we study here, essentially a copy of McGeer's design, is shown schematically in Figure 4.1. It consists of a *swing leg* (not in contact with the ground) and a *stance leg* (touching the ground), connected by a frictionless hinge at the *hip*. Extra mass is generally added at the hip serving as a crude model of an upper body. Each leg (assumed identical to the other) is composed of a rigid thigh and shank. The stance knee is locked. For kneed walkers, the swing knee is a frictionless hinge with a knee-stop preventing hyperextension between kneestrike and heelstrike. The knee stop also prevents the stance leg from hyperextending, but not from flexing. Straight-legged (kneeless) walkers may be viewed as obtained from kneed walkers by permanently locking the knees.

A strobe photo of one of our working kneed physical models is shown in Figure 4.2. A simulated walking cycle, using parameters measured from this model, is shown schematically in Figure 4.3. Step period and stride length data from the physical model matched our simulation results to within about 5% at this ground-slope. See Table 4.2 and Figures 4.6 and 4.12 for a comparison of simulation to experiment.

Three remarkable features that make the McGeer-like models so intriguing for both robotics and the understanding of animal gait are these:

1. **Existence of Gait.** A mechanism that resembles human legs in overall layout has an uncontrolled periodic motion that is rather anthropomorphic. A look at video recordings of McGeer's kneed machines (or our imitations of them), at passive-dynamic kneed simulations, or a comparison of passive-dynamic

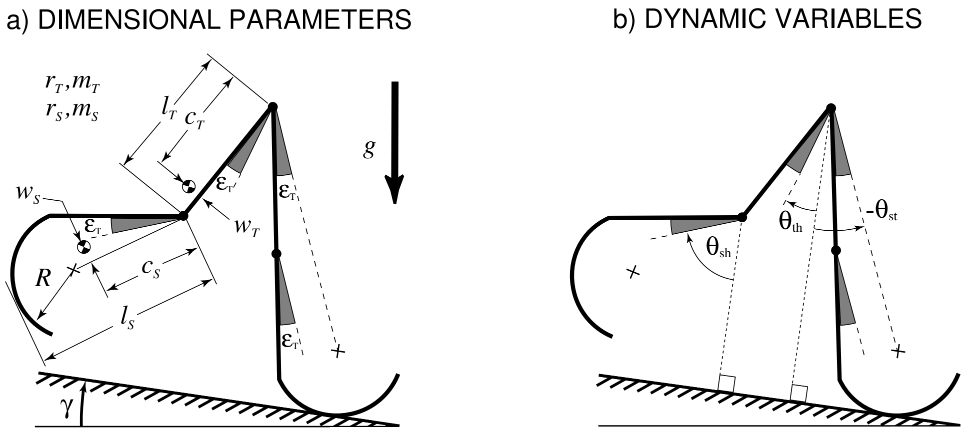

Figure 4.1: Our description of McGeer's kneed walking model. Shown above are (a) model parameters, and (b) dynamic variables. Radii of gyration and masses of thigh and shank are denoted by r_t, m_t, r_s, and m_s, respectively. The foot is a circular arc centered at the "+". ε_T is defined to be the angle between the stance thigh and the line connecting the hip to the foot center. Dynamic variable values θ_{st}, θ_{th}, and θ_{sh} are measured from ground-normal to lines offset by ε_T from their respective segments. A stop (not shown) at each knee prevents hyperextension of either knee. In straight-legged models, the knee is locked.

Figure 4.2: Strobe photo of our passive dynamic walker walking down a shallow ramp in our lab. The double leg-set constrains motions to a plane. The simulation we show in Figure 4.3 uses the parameters measured from this walker. Photo by R. Pratap.

stick-figure strobe shots with human data (lower part of Figure 4.3) certainly hint at the role of passive-dynamics in human gait.

2. **Efficiency of Gait.** These machines can walk down shallow ground-slopes, indicating small energy cost for horizontal transport. McGeer numerically found walking motions for ground-slopes as low as about 0.005 radians and we will show here predictions for walking at arbitrarily small ground-slopes.

3. **Stability of Gait.** For certain parameter combinations, McGeer found *stable* limit cycle motions for both straight-legged and kneed walkers as Goswami et al. (1996b) and Garcia et al. (1998) later repeated for some straight-legged walkers, and we repeat and extend here for kneed and straight-legged walkers.

Recent or current work is in progress by Fowble and Kuo (1996), Adolfsson et al. (1998), Coleman and Ruina (1998), and Garcia (1998) to extend McGeer's work on three-dimensional models. All of the above work hints at the possible role of passive-dynamics in producing *and* stabilizing efficient uncontrolled motion. But even unstable limit cycle motions of mechanical systems can (in principle) be stabilized with minimal energetic cost, as has been demonstrated for a three-dimensional walking model by Fowble and Kuo (1996) and implied by McGeer (1993b). Thus, although the stability feature of some passive-dynamic designs is intriguing and possibly useful, even unstable passive-dynamic motions could be relevant to animals or machines.

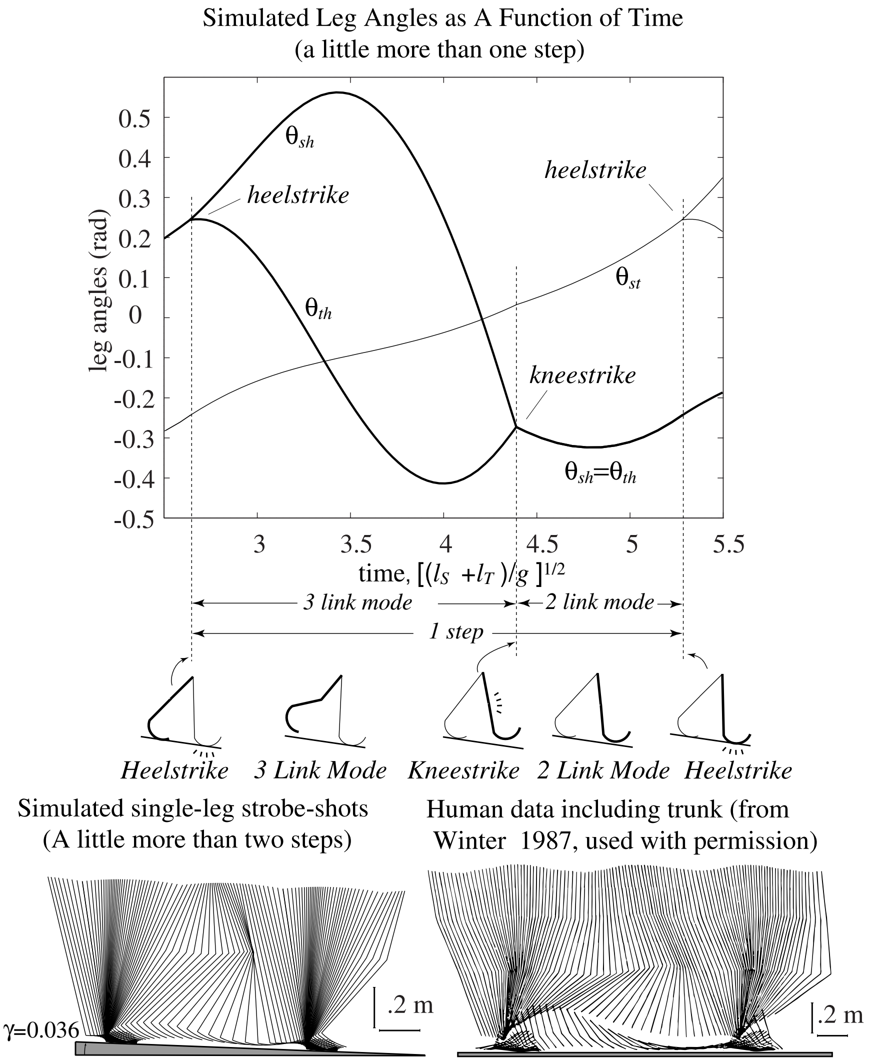

Figure 4.3: Simulated gait cycle (ours, similar to McGeer's). Angles of leg segments are shown from just before a heelstrike to just after the next heelstrike in a steady stable gait of the walker in Figure 4.1. The heavy line corresponds to the motion of the heavy-line leg on the small cartoon under the graph. At the start of the step, this is the stance leg, but it becomes the swing leg just after the first heelstrike shown, and again becomes the stance leg after the second heelstrike shown. In general, the angular velocities of the segments have discontinuities at kneestrike and heelstrike, which would appear as kinks in the trajectories above, but they do not happen to be prominent here. The strobe-like picture of the walker simulation shows the anthropomorphic nature of the gait; it was created from the simulated gait cycle in the graph. Measured human data (including trunk, with a smaller scale and a longer stride) from Winter (1987) is shown to the right. The parameters used in the simulation are those of the working physical model in Figure 4.2. The dimensional parameters are as follows: $l_t = 0.35$m, $w_t = 0$m, $m_t = 2.345$kg, $r_t = 0.099$m, $c_t = 0.091$m, $l_s = 0.46$m, $w_s = 0.025$m, $m_s = 1.013$kg, $r_s = 0.197$m, $c_s = 0.17$m, $R = 0.2$m, $\gamma = 0.036$rad$, g = 9.81$m/s$^2, \varepsilon_T = 0.097$rad.

4.3 Preliminaries

4.3.1 McGeer's Recipe

Our numerical analysis follows the program of McGeer (1990a) in treating a step as a Poincaré map. This approach is particularly well suited to the analysis of gait and is not limited to passive models. The technique is also described in some detail by Garcia et al. (1998) and Coleman et al. (1997).

A kneed walking *step* starts just after a heelstrike and ends after the next heelstrike. We assume that the swing knee is initially free to flex, and so we say that the walker is in *three-link mode.* Starting with initial conditions right after heelstrike, we solve the differential equations of motion for the three-link mode until kneestrike is detected. Using the angular-momentum based velocity-jump conditions that describe the knee collision (assumed to be instantaneous and sticking), we obtain new initial conditions for the start of the straight-leg or *two-link mode* (like McGeer, we use suction cups with adjustable leaks to enforce the sticking knee collision in our physical kneed models). We then solve the equations of motion for this straight-leg phase (with the knee locked) until heelstrike is detected (straight-legged walkers are always in two-link mode, and have no kneestrike during their gait). At the instant of double-support (i.e., contact at both feet), heelstrike occurs, which is also assumed to be instantaneous and sticking. The assumption of a sticking collision seems reasonable for the physical model where no macroscopic bounce or slip is observed. At heelstrike, the legs exchange stance and swing roles. Using the angular-momentum based velocity-jump conditions that describe the foot collision and renaming variables to switch legs, we obtain the initial conditions for the next three-link mode. Equations of motion for two- and three-link modes, as well as kneestrike and heelstrike calculations are given in Appendix 4.10.

Following McGeer, we assume in our simulations that during the heelstrike collision, there is no impulse on the old stance foot. We have not tested this modeling assumption with force plate measurements, but it is self-consistent and gives simulation results which correctly predict the behavior of our experimental models. The heelstrike collision causes the knee on the new swing leg to unlock and flex, and the next step begins. This yields one evaluation of the "stride function", McGeer's name for the nonlinear return map (or Poincaré map) which describes one step.

If the new initial conditions after one step are exactly the same as those of the previous step, we have found a period-one limit-cycle (also called a fixed point of the map, or a *gait cycle*). If gait cycles do exist, they might be stable, in which case they can be found by direct simulation of the system over several steps, provided the initial conditions chosen are in the basin of attraction (as in Goswami et al. (1997)). Whether or not the fixed points are stable, they can be found by root-finding algorithms applied to the return map, as in McGeer (1990a) (also explained by Garcia et al. (1998)). At the fixed point, the eigenvalues of the Jacobian determine stability (all eigenvalues inside the unit circle implies linearized stability), something that is of secondary interest.

Although the root-finding involved in finding a gait cycle involves the numerical solution of n equations in n unknowns (where n is the dimension of the return map) there is no *a priori* guarantee that any gait cycles will exist for a given passive dynamic walking machine (i.e., a given set of masses, lengths, etc.) on a given ground-slope γ. In practice, all searches with all designs have found either zero, one, or two anthropomorphic period-one solutions for given machine parameters and ground-slope. Other non-anthropomorphic solutions may exist where the leg swings forward and backward more than once or the swing leg makes full revolutions (as in Garcia et al. (1998) and Coleman (1998b)) but we do not consider them here.

Aside On The Map Dimensions

In general, one expects the dimension of the Poincaré map to be one less than the order of the system. For general straight-legged walkers, the map of the 4th-order system (two angles and two rates) is generally three-dimensional. For the superficially 6th-order kneed walker (three angles and three rates in three-link mode), the map is also only *three-dimensional* because the kneed walker is like a straight-legged walker for part of the swing-phase (after kneestrike). Thus, of the four numerically-calculated eigenvalues of McGeer (1990b) $(-0.001, 0.073, 0.261 \pm 0.363i)$, the first is actually *exactly* zero.

4.3.2 Reality Checks

Our numerical simulations are based on the assumptions above and not a general purpose rigid-body simulation code. Thus, some of the periodic solutions we find might violate various physically-relevant inequality conditions (foot clearance, etc.) as discussed in McGeer (1990b). For our purposes, we neglect those violations in order to have solutions to study over the parameter range of interest; but when building a physical model, these issues are of interest. In our simulations, we can rationalize this neglect for each item of concern as follows.

1. **Foot scuffing.** In simulations of straight legged walkers, the swing leg inevitably passes through or scuffs the ground near mid-swing. In physical realizations of straight legged walkers, McGeer overcame this scuffing either with electromechanically-retractable ankles or with tiles placed on the ground

(spaced for alternate stance-leg landings). A student group at Cornell University (Lattanzio et al. (1992)) overcame this difficulty using a passive mechanism that slightly retracted the swing leg. Kneed walkers may, but do not necessarily, avoid this scuffing by sufficiently flexing the swing knee at mid stride. For 3-D walking mechanisms (e.g., Fallis (1888)) and possibly in part for humans, foot scuffing can be avoided by side to side rocking. An unpowered scuffing solution could be changed to a non-scuffing solution by adding a small amount of actuation or a passive mechanism to slightly retract alternating legs for clearance.

2. **Positive knee-locking of the stance leg.** In our physical model, a joint-stop prevents knee hyperextension, but nothing stops knee flex, except residual suction in the knee cups just after kneestrike. In our simulations we assume that the stance knee is locked until it leaves the ground. Naturally-arising torques at the knee prevent unlocking in some but not all solutions. The simulation shown in Figure 4.3 has a slight stance-leg unlocking impulse at kneestrike which we ignore in our simulations. The corresponding physical model does not collapse, presumably because the naturally-arising torques just after kneestrike are enough to re-engage the knee stop. But even if they might not be, intermittent locking of a rotating joint can be performed with (theoretically) zero energy cost.

3. **Positive stance contact force and no slip.** The simulations assume contact between the stance leg and the ground. There is no contact tension in our simulations since all motions are well below the speed range ($v \approx \sqrt{gL}$) where tension is required to keep the stance leg in contact with the ground.

The ground reaction forces and impulses are generally well within the friction cone for the foot-ramp interaction (say $\mu = 0.8$).

4. **Unlocking of the new swing leg.** As a leg switches from stance to swing in the simulations, it is allowed to flex or extend at the knee. That this motion is not hyperextension needs to be checked. Some of our simulated gaits have a small amount of this hyperextension, especially at near-zero ground-slopes. Allowance of such hyperextension could be designed into a low-energy controller.

5. **Ground release of the new swing leg.** As a leg switches from stance to swing, it is released from the ground. We assume that it does not penetrate the ground, but ground penetration could also be avoided by a low-energy controller.

4.3.3 Measures Of Performance

The performance issues of interest here are energetic cost and speed.

Slope And Specific Cost Of Transport

Since moving sideways in a gravitational field is workless, a rational dimensionless measure of energetic efficiency is somewhat problematic for transport or locomotion on level ground. The most reasonable measure of efficiency

$$\frac{\text{(fundamental minimum energetic cost)}}{\text{(actual cost)}} \tag{4.1}$$

is zero for all but the most ideal machines, for which it is undefined (0/0).

The common dimensionless measure of (in)efficiency for locomotion is, however, the "specific cost of transport" or "specific resistance"

$$\eta = \frac{\text{(mechanical energy cost for transport)}}{\text{(weight)} \times \text{(distance travelled)}} \tag{4.2}$$

where the energy cost is also the energy dissipated. For steady walking motions with no joint or rolling friction, this reduces to (McGeer (1993b))

$$\eta = \frac{\text{(energy dissipated per step)}}{\text{(walker weight)} \times \text{(step length)}} \tag{4.3}$$

In the case of gravitationally-powered walking, the energy dissipated is (weight) $\times$ (height drop over one step). So the inefficiency measure is

$$\eta = \sin\gamma \qquad \text{or} \qquad \eta = \tan\gamma \tag{4.4}$$

depending on whether it is distance along the path or horizontal distance that is being rewarded.

Other possible measures of transport cost are equivalent to the ground-slope γ at which the gait takes place (at least for small cost of transport). For example, an almost identical measure of inefficiency is F/N where F is the propulsive force necessary to maintain a constant (average) velocity, and N is the force normal to the transport surface. This is the common measure of rolling resistance. For rolling downhill, this ratio reduces to $\tan\gamma$. The coefficient of friction μ can be defined as the work per unit (distance $\times$ weight) required to drag an object on level ground. The same object can slide steadily down a ramp if $\tan\gamma = \mu$.

Finally, borrowing from aeronautics, a measure of aircraft performance when powered by gravity is the best achievable glide ratio, $\tan\gamma$, which is also approximately the lift to drag ratio of the aircraft (as in Katz (1994)).

For downhill locomotion on small ground-slopes, with dissipation due only to sticking collisions, we can equate all of these inefficiency measures to

$$\text{inefficiency} = \gamma \tag{4.5}$$

Perfect transport efficiency is achieved by passive walking with $\gamma = 0$.

Speed Performance

Energetic efficiency does not credit speed nor penalize slowness. In practice, speed is also important. However, speed at a high energetic cost is of limited value, excepting for critical tasks such as hunting and escaping. Intuitive measures of merit, such as minimizing power for a given speed or maximizing speed at a given power are *dimensional* and thus lead to improvement by scale changes alone. For example, at a given ground-slope, the speed of all the designs we consider can be increased by a factor of ten with no increase in power by increasing the height by a factor of 100 and decreasing the mass by a factor of ten. That is, speed scales with $\sqrt{gl}$ and power scales with $mg^{3/2}l^{1/2}$, where g is the acceleration due to gravity, and l is a characteristic dimension, say the length of the walker's leg.

A simple nondimensional measure of merit that rewards speed is the square root of the Froude number,

$$\frac{v}{\sqrt{gl}} \tag{4.6}$$

at a fixed ground-slope. This measure cannot be affected by simple scaling changes. It is the natural measure recovered by redimensionalizing our non-dimensional results.

4.3.4 Kinetic Energy Lost In Plastic Collisions Of Walkers

The scalings we discuss depend essentially on the energy loss in the perfectly plastic (no-slip, no-bounce) heelstrike collision. General treatments of rigid body collisions, as well as references to the literature, may be found in works by Brach (1991), Chatterjee (1997), Chatterjee and Ruina (1998), and Brogliato (1996). We summarize below some relevant facts.

At the instant prior to heelstrike, let the velocity of the incipient contact point on the foot be $\dot{x}$ down the slope and $\dot{y}$ normal to the slope as in Figure 4.4. See also Chatterjee and Garcia (1998).

Under the presumably reasonable assumption that no impulse from the ground acts at the trailing foot, the kinetic energy lost in the collision is equal to the product

$$\frac{1}{2}\begin{bmatrix}\dot{x} & \dot{y}\end{bmatrix}\underbrace{\begin{bmatrix}M_{11} & M_{12}\\ M_{21} & M_{22}\end{bmatrix}}_{\mathbf{M}}\begin{bmatrix}\dot{x}\\ \dot{y}\end{bmatrix} \tag{4.7}$$

where $\mathbf{M}$ is a 2×2, symmetric, positive semi-definite matrix that depends on the mass distribution and geometry of the walker. $\mathbf{M}$ is also configuration dependent, i.e., it depends on θ^*_{st}. We define $\mathbf{M}(0)$ as $\mathbf{M}$ in the limit as $\theta^*_{st} \rightarrow 0$. For the discussion that follows, we need not discuss explicit (and complicated) representation of $\mathbf{M}$ in terms of the mass distribution and configuration of the linkage. The only exception is for a simple first-order approximation of $\mathbf{M}$ in Section 4.6.3; more discussion is found in a companion paper by Chatterjee and Garcia (1998).

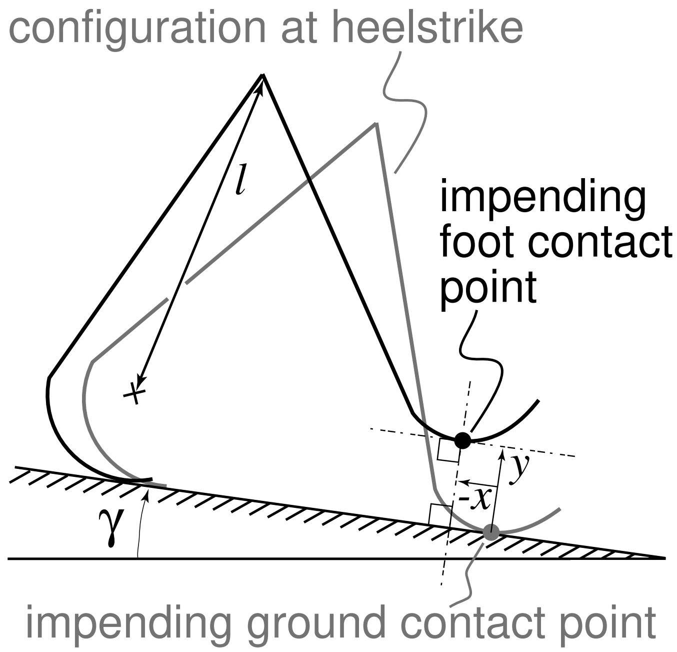

Figure 4.4: Close to heelstrike, the x and y coordinates of any point on the foot relative to any point on the ground can be used as generalized coordinates to describe the configuration of the walker. Since we are examining a known gait cycle, these points can be chosen to be the points at which contact will occur at heelstrike. l is the distance between the foot center and the hip.

4.4 The Simplest Walking Model

An extreme simplification of a straight-legged walker is the "minimal biped" of Alexander (1995) (a hip-mass on massless legs), made deterministic by the addition of infinitesimal point-masses at the feet. This "simplest walker" was studied in some detail in Garcia et al. (1998) with both perturbation methods and numerical simulation. The key results of that study are summarized below.

The simplest point-foot model has two gaits (two fixed points) at arbitrarily small ground-slopes. Of these, the long-step gait is stable at sufficiently small ground-slopes ($\gamma < 0.015$), while the short-step gait is unstable at all ground-slopes. (In Garcia et al. (1998) these two gaits are called the "long-period" and "short-period" gaits, possibly incorrectly intimating that the long-period gait is generally slower than the short-period gait.) As $\gamma \to 0$, the long-step gait motion approaches a symmetric motion that has normal heelstrike collisions ($\dot{x} = 0$ in Figure 4.4). The short-step solution does not have this symmetry and the (quasi-massless) foot has a collision with a non-zero tangential component at heelstrike.

For this walker, both gait cycles were found to have stance angle θ_{st}, step-length, and velocity proportional to $\gamma^{1/3}$ at small ground-slopes (see curves D in Figure 4.9), while the step periods tend to (different) nonzero constants as the step length goes to zero. This implies a walking power consumption proportional to the fourth power of speed (for low speeds), for *both* gaits:

$$\text{Power} \propto m v^4 g^{-1/2} l^{-3/2}, \tag{4.8}$$

where m is the walker mass, v is the average walking velocity, g is the gravitational constant, and l is the walker's leg length. Equation 4.8 makes the unintuitive prediction that cost of transport increases with decreasing gravity and leg length.

This scaling also follows from the energy balance between collision losses and used-up gravitational potential energy, rather like for the rimless wheel described by McGeer (1990a). For this special case, $\mathbf{M}(0)$ of Equation 4.9 is singular, and the only energy dissipation term comes from the $\dot{y}$ or normal component of the foot velocity at heelstrike, for both gaits. As explained partially in Alexander (1980) and for this model in Garcia et al. (1998), the normal velocity is proportional to both stance angle and stance angle rate. Since for small motions, the period is approximately independent of amplitude and the speeds are proportional to amplitude, the kinetic energy lost is proportional to the fourth power of step amplitude. The gravitational power available is proportional to step amplitude and ground-slope. Thus, (step amplitude)^4 $\propto$ (step amplitude) $\times$ (slope), and so (step amplitude) $\propto$ (slope)$^{1/3}$.

Between ground-slopes of $\gamma \approx .015$ to $\gamma \approx .019$ a period-doubling route to chaos was observed. Aside from the period doubling route to chaos, no other non period-one gaits were sought or found. Recently, Howell and Baillieul (1998) discovered a stable period-three gait at a slope of $\gamma \approx 0.0125$ and subsequent period-doublings (period-six, period-twelve, etc.) for this model.

In the rest of this chapter we describe, as based on or at least tested by numerical integrations, which of the features above extend to more general straight-legged and kneed walkers. In some cases the equations of motion were derived with the help of symbolic algebra (Maple®) and the simulations were run using MATLAB®. The numerical methods and error checks used are similar to those described in Garcia et al. (1998).

4.5 More Complex Walking Models

4.5.1 Straight-Legged Pointfoot Walker With Finite Foot Mass

Describing mass distribution for a general straight-legged walker requires specifying center of mass position and moment of inertia. The studies of Goswami et al. (1996b) and Goswami et al. (1997) were limited to a subset where the center of mass was on the foot-to-hip line (zero fore-aft offset), which reduced the number of parameters by one. We now consider a subset of the walkers of Goswami (and co-workers) with two finite point-masses: one at the hip and one at the foot (like the Goswami walkers, this subset also has point-feet). This simplification further reduces the number of parameters by one.

For the particular cases we study, the foot mass is still substantially smaller than the hip mass, but not infinitesimally small; the swing-leg angular velocity does not contribute to the pre-collision angular momentum about the new contact point (point of foot-collision at heelstrike), and the return map is two-dimensional, as it is with the simplest walker. Despite this simplification, we still resort to numerics when studying this class of models.

Solution Families And Scaling Laws

For the cases we have tried of these walkers, there are still two solutions at arbitrarily small ground-slopes. Again, the two solutions are distinguished by the long-step solution having essentially normal heelstrike collisions and the short-step solution having heelstrike collisions with a significant tangential component (a feature also approximately observed in McGeer (1993b) and McGeer (1992) for a more complex walker).

For the simplest walker, with negligible foot mass, the only kinetic energy lost is that of the hip. When the feet have finite mass, however, the foot masses also lose energy at heelstrike. If the striking foot hits the ground with no tangential velocity, the loss scales as step length to the fourth power, giving a step length proportional to $\gamma^{1/3}$. With non-negligible-foot-mass, a tangential velocity component at heelstrike can change the energy-loss scaling.

This is demonstrated in Figure 4.5, which shows stance angle as a function of γ for two walkers as described above. (Figure 4.11 is similar but plots step velocity as a function of slope.) For the long-step solution, the stance angle remains proportional to the cube-root of the ground-slope. Unlike the simplest walker, the stance angle for the short-step solution is linearly proportional to γ at very small ground-slopes, and proportional to $\gamma^{1/3}$ at somewhat larger slopes. There is some transition region where the short-step scaling changes between the $\gamma^{1/3}$ and the γ scaling. So for the long-step gait, which is stable, power $\propto$ (speed)^4, while for the unstable short-step gait, power $\propto$ (speed)^2 below the transition region and $\propto$ (speed)^4 above the transition region. Experimental results from Vaida et al. (1981) show that total power requirement data for level walking with and without stilts can be fit to square or fourth-power scaling laws.

The Short-Step Transition Region

The ground-slope at the transition region is governed by the ratio of the foot mass to the hip mass. If the foot mass is ρ times the hip mass instead of being totally negligible, then we will show later by asymptotic arguments that the transition occurs near a ground-slope proportional to $\rho^{3/2}$. Step length $\propto$ $\gamma^{1/3}$ at ground-

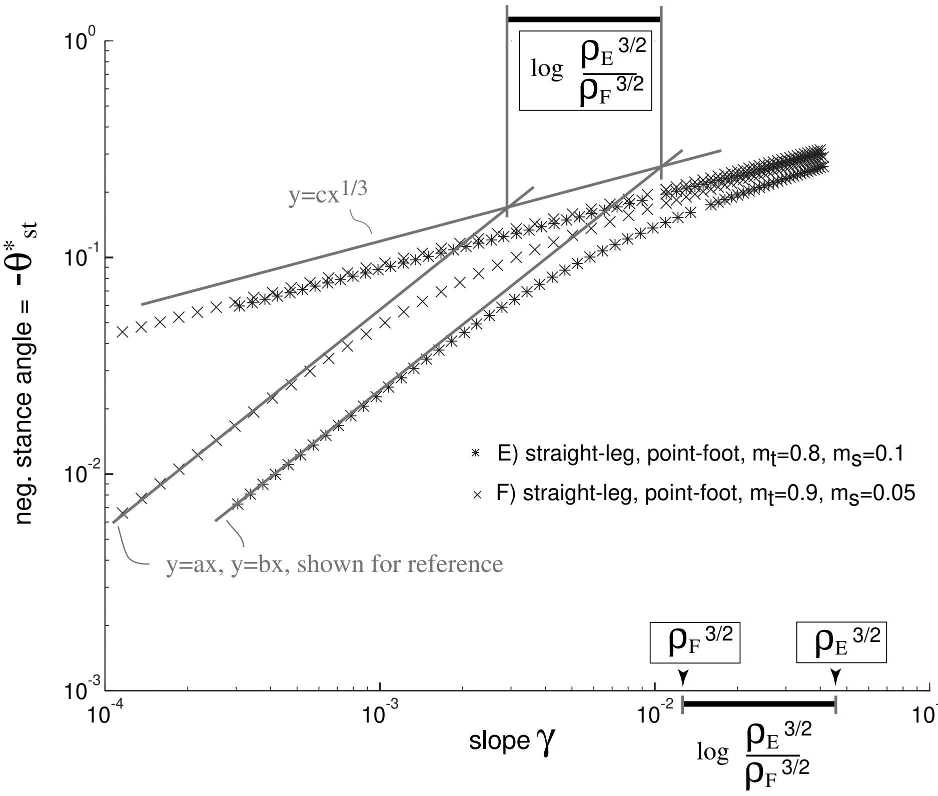

Figure 4.5: Scaling transition comparison for two point-foot walkers. Data for walker E is shown again in Figure 4.9. Parameters for walker E are shown in Table 4.1. Walker F has a foot mass of 0.05 and a hip mass of 0.9. At large ground-slopes ($\gamma \gg \rho^{3/2}$), the walker stance angles and velocities scale as $\gamma^{1/3}$. The predicted critical slopes for walkers E and F are shown at the bottom of the plot. Above these slopes, we expect the stance angle to be proportional to the cube root of the ground-slope. The ratio of the two critical slopes is nearly identical to the ratio of the intersections of the two best-fit lines in the linear scaling regimes with a cube-root scaling line. The solution curves for the "simplest" walker would show as parallel lines more or less on top of the upper curves in this figure. Note that the "×" symbol used here refers to data from a different walker than in Figure 4.9. See Figure 4.11 for a similar plot of step velocity versus slope for the above walkers.

slopes where $\gamma >> \rho^{3/2}$, and step length $\propto \gamma$ at smaller slopes where $\gamma << \rho^{3/2}$ (see Section 4.6.3).

Figure 4.5 shows two such transitions. Walker E has a foot mass of 0.1 and a hip mass of 0.8 (any units) and so $\rho_E = 0.125$. Walker F has a foot mass of 0.05 and a hip mass of 0.9 so $\rho_F = 0.0556$. Using the above rule of thumb, we expect the transition ground-slope for walker E to be proportional to ${\rho_E}^{3/2} = 0.125^{3/2} \approx 0.044$ and that of walker F to be proportional to ${\rho_F}^{3/2} = 0.0556^{3/2} \approx 0.013$ (these values are marked in Figure 4.5). The ratio of the two critical slopes is equal to the ratio of the intersection values of the extrapolated best-fit lines from each walker's linear-scaling regime with an arbitrary cube-root-scaling line, as shown in the figure.

4.5.2 Generic Kneed And Straight-Legged Models

What about walkers with more general mass-distribution, non-zero foot radii, and foot offsets? Figure 4.6 shows stance angles at gait cycles occurring at different ground-slopes γ for the kneed walker of Figure 4.3. Stable solutions are denoted by the heavy line. The dashed curve shows stance angle versus ground-slope for the straight-legged version (the same walker but with the knees always locked, $\theta_{sh} \equiv \theta_{th}$).

Some observations about the solutions in Figure 4.6 are as follows:

1. For both kneed and straight-legged walkers there are ground-slope γ regimes where there are either zero, one, or two solutions. This agrees with the observations and calculations of McGeer (1990a) and McGeer (1990b).

2. For the parameters used here, none of the straight-legged solutions and only a section of the kneed solution-locus are stable.

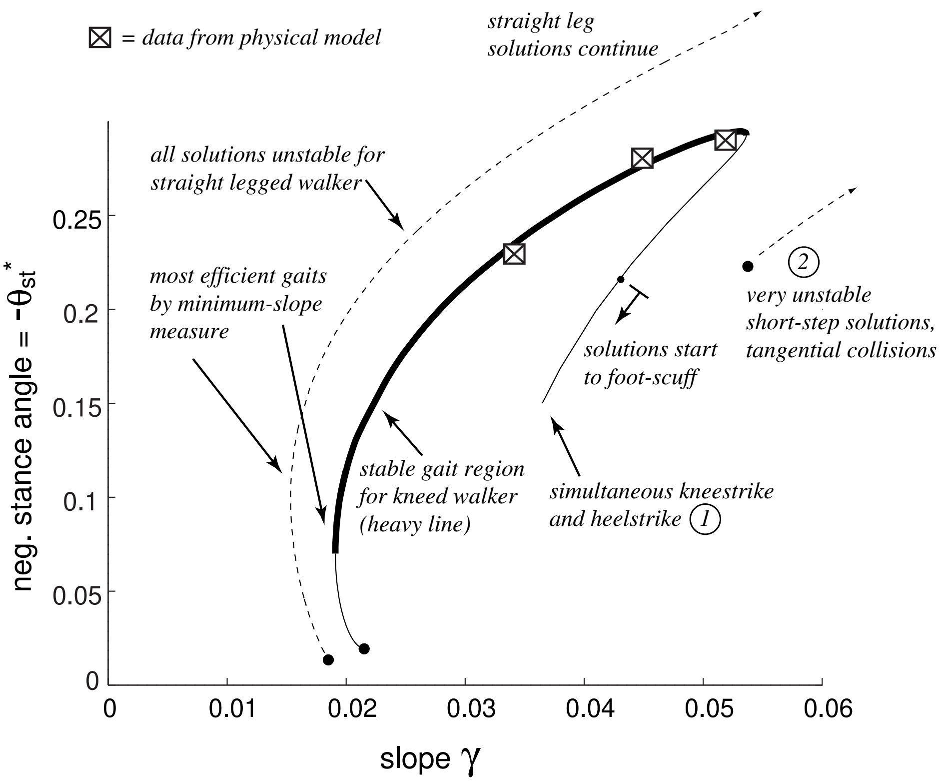

Figure 4.6: Numerically-calculated locus of solutions showing stance angle as a function of ground-slope for our physical kneed walking model (solid line) and for the same model but with the knees locked (dashed line) at fixed points. The thick portion of the solid line denotes stable solutions for the kneed walker. Figure 4.12 is a similar plot but with velocity replacing stance angle.

3. Along the kneed curve, kneestrike occurs later and later in the step, until at one end of the curve (point 1), heelstrike and kneestrike occur simultaneously. The locus of solutions terminates here since we do not investigate motions where heelstrike precedes kneestrike. In the analogous region on the dashed curve for the straight-legged walker (point 2), the heelstrike collision becomes increasingly tangential in nature, until the solution disappears. In this area, solutions and numerics become very unstable; we believe that the solution terminates at a ground-slope of about $\gamma = 0.056$ based on numerical evidence not discussed here.

4. At the other end of the long-step curve, the gait cycles are approaching initial conditions which approach falling backwards, i.e.,. the walker has just enough initial kinetic energy for the stance leg to make it past the vertical position. The same thing occurs at the analogous point on the dashed curve for the straight-legged walker. In both cases, this is the slowest gait found for the walkers.

5. All of the straight-leg solutions and some of the kneed solutions allow the foot to scuff the ground during the gait cycle.

6. Each walker needs a minimum ground-slope to sustain gait (about 0.016 rad for the straight-leg walker and about 0.02 rad for the kneed walker). So, these walkers are not perfectly efficient by the efficiency measures discussed earlier.

4.6 Walking At Near-Zero Slopes

As seen in the previous section, generic McGeer-like walking machines will typically have no steady walking motions below some nonzero ground-slope; thus, they have some nonzero inefficiency. But we have already seen that the point-foot walker with or without negligible foot mass can be perfectly efficient. Here we investigate the possibility of more general straight-legged and kneed walkers capable of zero-slope walking.

4.6.1 Necessary Mass Distribution Conditions For Efficient Walking

Necessary conditions on the mass distribution for near-zero-slope kneed walkers are as follows:

1. As demonstrated in a companion paper by Chatterjee and Garcia (1998), based on some reasonable assumptions, if walking motions do occur at very small ground-slopes, these motions will be very slow. For these motions, gravity forces will be much bigger than inertia forces. As a result, the walker must be close to static equilibrium at all times. In the limit of zero slope, the walker configuration must approach a static equilibrium, or standing solution. Thus the foot-normal at the near-zero-slope walking condition must be directed towards the body center of mass.

2. At the instant of double-support, or heelstrike, both legs are straight and simultaneously touch the ground. As the ground-slope (hence, step length) goes to zero, the spacing between the legs at this instant also goes to zero. In

the limiting case, the foot contact point must be that point on the foot which is farthest from the hip. Thus the normal to the foot contact point must pass through the hip.

3. From (1) and (2) the line from the hip through the body center of mass must intersect the foot curve normally at the place that will be the nominal contact point at zero-slope walking. For circular feet this is equivalent to the collinearity of the center of mass of the whole body, the hip, and the foot center (see Figure 4.7).

4. For the swing leg to be in static equilibrium in three-link mode and to have zero knee-locking torque, the center of mass of the shank must lie directly under the knee, in the straight-leg configuration (see Figure 4.7).

The simplification of these ideas for straight-legged walkers is described by item (3) above. These necessary statics-based conditions on the mass distribution do not guarantee that near-zero-slope walking solutions exist. Although we do not know general sufficient conditions, it is our experience that designs which meet these conditions and whose total center of mass is close to the hip do have walking motions for arbitrarily small ground-slopes.

The machines investigated by Goswami et al. (1996b) and Goswami et al. (1997) satisfy these necessary conditions, so will probably walk at arbitrarily small ground-slopes.

Walkers satisfying the necessary conditions above have an $\mathbf{M}(0)$ which is diagonal.

4.6.2 Tuning Mass-Distribution For Near Zero-Slope Walking

After adjusting the mass parameters (*tuning*) for the walkers of Figures 4.3 and 4.6 to match the conditions for zero-slope walking (Figure 4.7), we found (by simulation) that both of these *tuned* walkers do indeed have walking motions at near-zero-slopes. Also, each walker now has *two* walking motions or gait cycles at all ground-slopes of $\gamma < 0.04$. The long-step, long-period cycle is stable at small ground-slopes, while the short-step, short-period cycle is unstable. In these numerical experiments, some reality-checks from section 4.3.2 are violated, and so we cannot experimentally verify the tuning criteria.

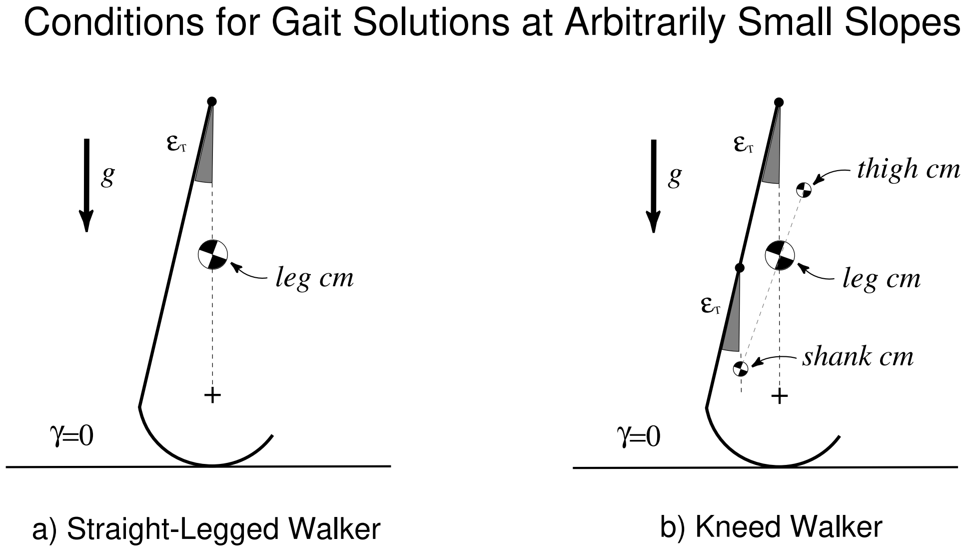

Figure 4.7: To walk at arbitrarily shallow ground-slopes, a walker must allow a static standing solution at zero slope with the stance leg locked, the swing leg unlocked, the legs parallel, and the hip directly above the foot contact. These conditions are shown graphically for (a) a straight leg, and (b) kneed walker.

Figure 4.8 shows the evolution of solution families of the tuned kneed walker

(Figure 4.7) as it is detuned into the original kneed walker of Figure 4.6. As the 0% detuned walker is detuned, the graphs change shape in the following ways.

1. The cusp at the origin breaks and the two solutions separate. The long-step solution remains stable for small slopes but shifts to the right and no longer extends to zero ground-slope. Also, an unstable region appears at very low speeds.

2. The short-step solution also shifts to the right; the point on this curve where the solution terminates (where heelstrike and kneestrike are simultaneous) shifts up the curve.

3. At higher ground-slopes, the two solutions get closer and eventually merge as the walker is de-tuned; the curve then splits into two solutions. As the walker is de-tuned further, the solutions continue to separate. The high-slope solution branch presumably continues to exist at very high slopes. This merging and splitting of solutions does not seem to occur in straight-legged walkers.

Figure 4.9 shows short-step and long-step gait solutions for the abovementioned tuned straight-leg (A) and kneed (C) walkers, the point-foot walker of Garcia et al. (1998) (D), which is tuned by definition, and two other tuned straight-leg models (B and E). The parameters for each of these tuned walkers are listed in Table 4.1. Figure 4.13 is the same plot but with step velocity replacing stance angle.

We make the following observations on their behavior.

1. All of these walkers can walk at arbitrarily small ground-slopes.

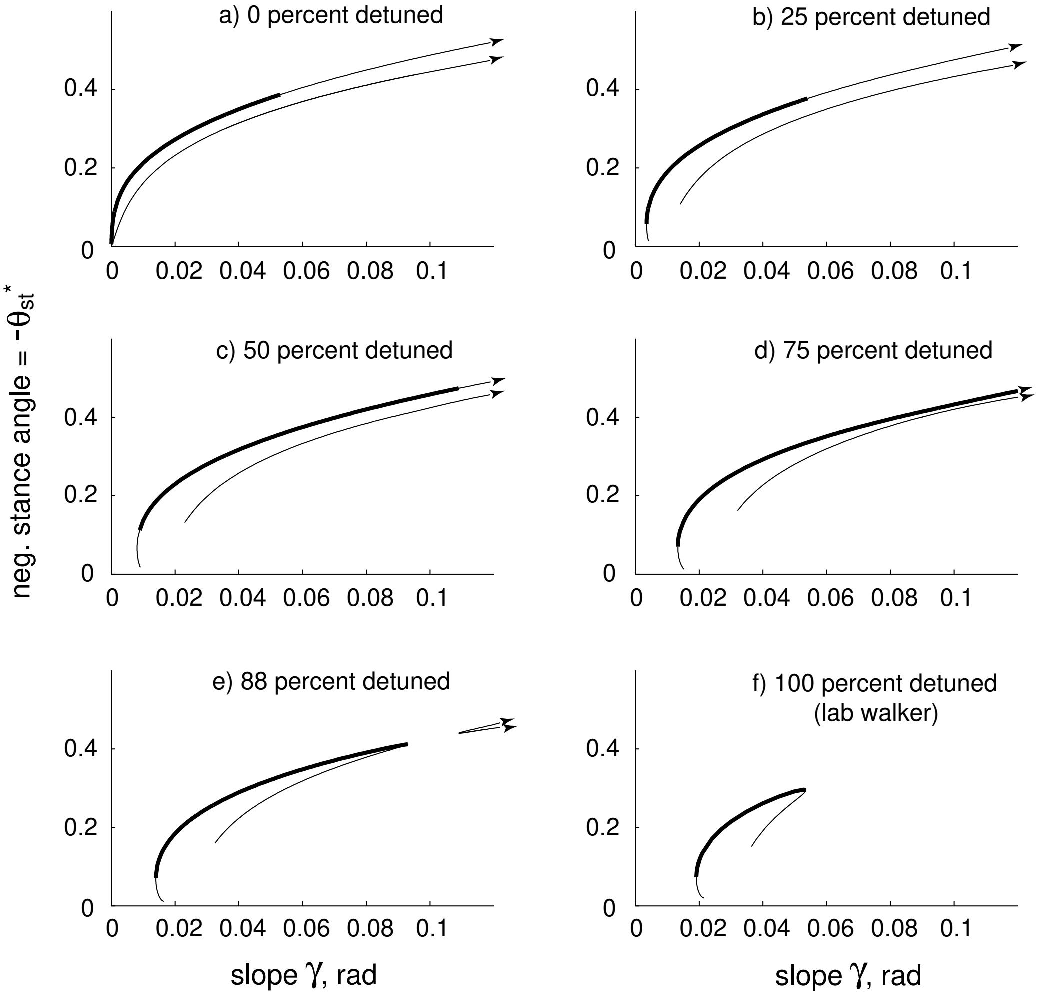

Figure 4.8: Solution families during de-tuning of the tuned kneed walker. Subplot (a) shows the perfectly tuned walker with solutions extending to zero slope. Subplots (b), (c), and (d) show the solution curves no longer meeting at low slope. Subplot (e) shows the solutions merging and splitting into two solution regions, and subplot (f) shows the (100 % detuned) original lab walker of Figure 4.6. Presumably, the higher-slope solutions are also present but not visible on subplot (f).

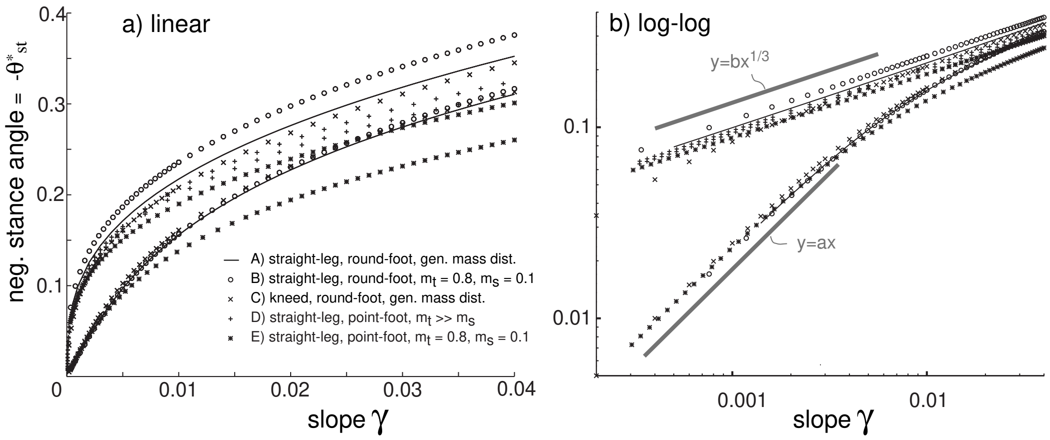

Figure 4.9: Gait families for tuned zero-slope-capable walkers on (a) a linear plot, and and (b) a log-log plot. Parameter values are listed in Table 4.1. By "gen. mass. dist." it is meant that the parameters are close to those of the kneed walker of Figure 4.3. Note (1) there are two gaits cycles at each γ for all walkers shown; (2) for the "simplest" walker (D) both step lengths are proportional to $\gamma^{1/3}$; (3) the short-step gaits of the other walkers have step lengths proportional to γ for small γ; (4) the long-step gaits for the other walkers have step lengths that are much longer than for the short-step gaits, though not necessarily exactly proportional to $\gamma^{1/3}$ for small γ; and (5) for a point-foot, straight-legged walker with non-negligible foot mass, the step length of the long-step gait *is* proportional to $\gamma^{1/3}$ for small γ. Figure 4.13 is a similar plot but with step velocity instead of stance angle.

Table 4.1: Parameters for several tuned walkers in any consistent units. Only C has knees. Straight-legged walkers A, B, D, and E have redundant parameters since the shank and thigh are rigidly connected. m_t and m_s are the thigh and shank masses. For B, D, and E m_s is a point-mass at the bottom of the foot.

Walker	A	B	C	D	E
l_t	0.35	0.5	0.35	0.5	0.5
w_t	-0.023	0	-0.023	0	0
m_t	2.345	0.4	2.345	∞	0.4
r_t	0.1882	0	0.1882	0	0
c_t	0.084	0	0.084	0	0
l_s	0.46	0.5	0.46	0.5	0.5
w_s	0.022	0	0.022	0	0
m_s	1.013	0.1	1.013	1	0.1
r_s	0.1226	0	0.1226	0	0
c_s	0.17	0.5	0.17	0.5	0.5
R	0.2	0.2	0.2	0	0
ε_T	0.097	0	0.097	0	0

2. The simplest walking model (D) is the only one with both gaits having a step length $\propto \gamma^{1/3}$.

3. At the smallest ground-slopes, the tuned straight-leg walkers with finite foot mass (A, B, and E) each have one gait solution with step length $\propto \gamma$ (the short-step gait) and one solution with step length $\propto \gamma^{1/3}$ (the long-step gait). At steeper ground-slopes ($\gamma > 0.01$), all the gait solutions have step length $\propto \gamma^{1/3}$ as seen most clearly in Figure 4.9b.

4. The tuned kneed walking solutions (C) *seem* to follow a similar pattern to the straight-leg solutions in Figure 4.9. However, the long-step lengths of the tuned kneed walker are not proportional to $\gamma^{1/3}$ at very small ground-slopes. (Note how the "$\times$" symbols trail off at the left of the plot as if they were migrating towards the linear $y = ax$ scaling.) In fact, further numerical results indicate that this solution slowly changes to a linear scaling at extremely small ground-slopes ($\gamma \approx 0.00015$). Strictly speaking then, for the kneed walker, both the long-step and short-step gaits have step lengths $\propto \gamma$, although the longer period gait changes its scaling at extremely small ground-slopes.

5. Although it is not shown in the plots, all of the tuned long-step gaits have a certain ground-slope below which the gait is stable (down to zero slope).

Our numerical results indicate that a step length proportional to $\gamma^{1/3}$ is the longest step possible (and hence the smallest power consumption possible) at very small ground-slopes (i.e., low walking speeds).

As compared to walkers which do not meet the criteria in Figure 4.7 above, tuned walkers will have a longer step length at a given small ground-slope (assuming other parameters such as masses, inertias, and leg lengths are held constant). This is illustrated by comparing Figures 4.9 and 4.6 from Section 4.5.2. At any ground-slope where gaits exist for both tuned and un-tuned walkers, the tuned walkers have a larger stance angle, and hence a longer step. Because the step periods are approximately the same for similarly-scaled tuned and un-tuned walkers, tuned walkers will also be faster than non-tuned walkers at a given ground-slope.

4.6.3 Energy And Scaling For Near-Zero-Slope Walking

Here, we show that for general walking machines that walk at near-zero ground-slopes, the step length (or velocity) for small slope γ is asymptotically proportional to either $\gamma^{1/3}$ or to γ. Thus far in our numerical investigations, as seen from the numerical results in Figure 4.9, we find that all solutions for zero-slope-capable straight leg walkers apparently obey one of these two scalings.

Derivation Of Scaling Rule For Straight-Legged Walkers

Consider a straight-legged walker, with curved feet, at the instant prior to heelstrike. At this instant, $\theta_{st} = \theta_{th}$ (ignore θ_{sh} in Figure 4.1 for the straight-leg case). The matrix $\mathbf{M}$ referred to earlier (see Equation 4.7) is actually a function of θ_{st}, but since θ_{st} is small we write

$$\mathbf{M}(\theta_{st}) = \mathbf{M}(0) + \mathcal{O}(\theta_{st}). \tag{4.9}$$

For the present discussion, we mention that for zero-slope-capable walkers, $\mathbf{M}(0)$ will be diagonal when expressed in the coordinate system implicit in Figure 4.4 and Equation 4.7. In particular, its element M_{11} will be equal to the total mass of the walker, while element M_{22} will be smaller (a function of mass distribution and walker geometry). Finally, for a straight-leg, point-foot walker with a small foot mass (as considered in Section 4.5.1), M_{22} is approximately equal (asymptotic) to the foot mass.

Recall Figure 4.4. Now we express $\dot{x}$ and $\dot{y}$ in terms of $\dot{\theta}_{st}$ and $\dot{\theta}_{th}$, at the heelstrike configuration but just before heelstrike ($\theta^*_{th} = -\theta^*_{st}$). We define the length l to be the distance between the hip and the foot center.

We obtain:

$$\begin{aligned} \dot{x} &= R(\dot{\theta}_{st} - \dot{\theta}_{th}) + l\cos\theta^*_{st}(\dot{\theta}_{st} - \dot{\theta}_{th}), \\ \dot{y} &= -l\sin\theta^*_{st}(\dot{\theta}_{st} + \dot{\theta}_{th}). \end{aligned} \tag{4.10}$$

Assume that as $\gamma \to 0$, the gait cycle step length θ^*_{st} is asymptotically of $\mathcal{O}(\gamma^p)$ for some $p > 0$. During the walking step, the angular rates $\dot{\theta}_{st}$ and $\dot{\theta}_{th}$ must also be $\mathcal{O}(\gamma^p)$. Moreover, at the instant of heelstrike, one or both of these rates must also be of the same order of magnitude $\mathcal{O}(\gamma^p)$ (these claims are proved by contradiction in Chatterjee and Garcia (1998)). Note that one would expect, from eq. 4.10, that $\dot{x} = \mathcal{O}(\gamma^p)$, while $\dot{y} = \mathcal{O}(\gamma^{2p})$.

Typically, at heelstrike, $\dot{x} \neq 0$ in Eq. 4.10, i.e., $\dot{\theta}_{st} - \dot{\theta}_{th} \neq 0$. As a result, for small γ, we find that the energy dissipation per step is proportional to $\dot{x}^2$ (see Eq. 4.7), or $\mathcal{O}(\gamma^{2p})$. Since the potential energy available per step is $\mathcal{O}(\gamma^{p+1})$ (step length $\times$ slope), we find that energy balance requires that $2p = p + 1$, or $p = 1$, i.e., the step length is proportional to γ.

On the other hand, for some gaits of some walkers, it may happen that $\dot{x} = 0$ at heelstrike, or $\dot{\theta}_{st} - \dot{\theta}_{th} = 0$. In those cases, the energy dissipation per step is $\mathcal{O}(\dot{y}^2)$ or $\mathcal{O}(\gamma^{4p})$. As before, the potential energy available per step is $\mathcal{O}(\gamma^{p+1})$. Energy balance therefore requires $4p = p + 1$, or $p = 1/3$, i.e., the step length *can* be proportional to $\gamma^{1/3}$. The same result holds true if $\dot{x}$ is not exactly zero, but a higher-order quantity, such that $\dot{\theta}_{st} - \dot{\theta}_{th} = \mathcal{O}(\gamma^{2p})$, or smaller. Thus, it is seen that the only two possibilities for step lengths proportional to γ^p are $p = 1$ and $p = 1/3$. Since step periods are approximately constant at small slopes, the above arguments also hold true for velocity as a function of slope.

Predicting The Short-Step Transition Slope for Point-Foot Walkers

Short-step gaits for straight-legged walkers with non-negligible foot masses and point-feet will typically exhibit a transition region between the linear and cube-root scalings, as discussed in Section 4.5.1. Here, we consider the effect of adding a small but non-negligible foot mass of ρ times the total mass (here assumed nondimensionalized to unity), to the simplest point-foot walker.

Assuming θ_{st} is small, the kinetic energy dissipated in the collision is simply ($\mathbf{M}(0)$ is now diag[1 ρ]; see discussion of $\mathbf{M}(0)$ above also)

$$\frac{1}{2}\left(\dot{y}^2 + \rho\dot{x}^2\right) .$$

Assuming a step length comparable to γ^p as before, we find that energy balance requires

$$\gamma^{p+1} = \mathcal{O}(\gamma^{4p} + \rho\gamma^{2p}) ,$$

or

$$\gamma^{3p-1} + \rho\gamma^{p-1} = \mathcal{O}(1) .$$

For ρ any nonzero positive constant, this expression implies $p = 1$ as γ becomes arbitrarily small (i.e., step length $\propto \gamma$ at very small ground-slopes). On the other hand, for sufficiently small but fixed γ, taking smaller and smaller ρ, the scaling eventually reverts to $p = 1/3$. The transition occurs when the two left hand side terms above are comparable, or $\rho = \mathcal{O}(\gamma^{2p})$, in which case clearly $p = 1/3$. Thus, the transition occurs when $\rho = \mathcal{O}(\gamma^{2/3})$, or $\gamma = \mathcal{O}(\rho^{3/2})$ (see Section 4.5.1).

Extension Of Scaling Argument To Kneed Walkers

Kneed walkers dissipate kinetic energy in collisions at both heelstrike as well as at kneestrike. For heelstrike, the energy loss calculations described above still hold: the pre-collision velocities are determined from the straight-leg or 2-link configuration, but the matrix $\mathbf{M}$ of Equation 4.7 has to be calculated for the walker in three-link mode since the new swing leg is not constrained at the knee.

For kneed walkers, if the step length is proportional to γ^p, then the collision losses at kneestrike as well as heelstrike are each proportional to either γ^{2p}, or γ^{4p}, or perhaps even higher order, as $\gamma \to 0$. Of these, the heelstrike losses are already known to be at least $\mathcal{O}(\gamma^{4p})$. Thus, the total losses per step are either $\mathcal{O}(\gamma^{2p})$ or $\mathcal{O}(\gamma^{4p})$, leading to $p = 1$ or $p = 1/3$, respectively.

In our simulations zero-slope-capable kneed walkers also have one solution that nearly obeys the high-efficiency ($p = 1/3$) scaling over the range of ground-slopes depicted in the figures. At even smaller ground-slopes, the knee collision eventually dominates and $p = 1$ scaling is recovered.

4.7 Chaos In Tuned Kneed Walkers

As shown for the point-foot walker in Garcia et al. (1996) and Garcia et al. (1998), and the walkers of Thuilot et al. (1997) and Goswami et al. (1996b), tuned kneed walkers can also exhibit period doubling and chaotic gait, as shown in Figure 4.10.

At small enough ground-slopes, the period-one gaits cycles are stable. The

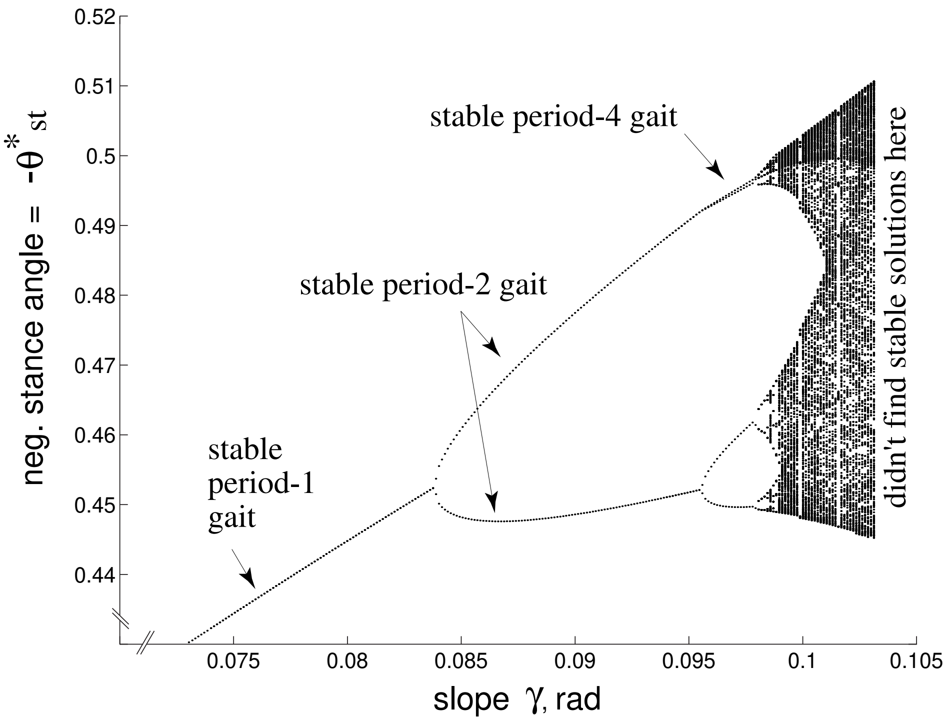

Figure 4.10: Period doubling of stable kneed walking motions. Only stable walking motions are shown, although all periodic gaits persist as unstable gait cycles after they undergo period-doubling. The parameters are those of the tuned kneed walker (C) in Table 4.1.

period-one motion becomes unstable at a slope of about 0.084, but a stable period-*two* gait appears, followed by a stable period-four gait, and so on (see Figure 4.10). The walker continues to have stable (although possibly chaotic and not period-one) solutions at ground-slopes of up to about 0.103. At higher slopes we could not find stable walking motions for this kneed walker. For reasons we do not understand, the period-doubling occurs at higher ground-slopes than for the point-foot walker in Garcia et al. (1998) where the first period-doubling occurs at $\gamma \approx 0.015$ and the chaotic gait is at about $\gamma = 0.019$.

The existence of chaotic motions suggests that the "stepping stone" problem addressed by McGeer (1993b) with active torque control might have passive or nearly passive solutions (as also mentioned in Garcia et al. (1998)).

4.8 Concluding Remarks

We have found necessary conditions for arbitrarily small-slope passive dynamic walking. Remarkably, all the designs of this type that we have numerically tested have one solution that is stable and which is highly efficient (has no tangential velocity component at heelstrike and obeys the scaling rule 4.8). For the simplest walker of Garcia et al. (1998), the non-tangential collisions come from a time-reversal symmetry in the long-step gaits at very small ground-slopes. We do not know why the more general zero-slope-capable walkers also seem to have these efficient solutions.

Since the details of bipedal locomotion are intimately related to the mechanics of freely-swinging limbs it seems possible that the efficiency of human and efficient robot legs is related to the efficiency of similar, passive-dynamic mechanisms. Since we have described clear mechanical criteria for efficient passive-dynamic walking machines, similar criteria may apply for efficient human and robotic legs. Might it be more than coincidence that human legs nearly satisfy the necessary zero-slope conditions we have described? It remains to be seen if any of the scalings we have found are obeyed by an optimized but actively-powered bipedal design.

A quantitatively inclined reader will have noticed that the slopes and specific-costs of transport in our calculations are below those directly applicable to animals and physical machines. At higher ground-slopes or costs of transport other terms become important but the ideal effects we have described may still largely describe the scalings in passive walking.

Our results also imply that small parameter adjustments (in particular, fore-aft movement of the thigh and shank centers-of-mass) can influence biped gait characteristics rather remarkably.

4.9 Appendix Of Additional Plots And Tables

Figures 4.12 and 4.11 are similar to Figures 4.6 and 4.5, except that step velocity replaces stance angle.

One interesting feature of these figures is that for almost all of the walkers, the long-step (long-period) gait is the faster gait at lower slopes, while the short-step (short-period) gait is the faster gait at higher slopes. The only exception to this rule is the simplest walker, for which the short-step gait is always faster than the long-step gait at a given slope.

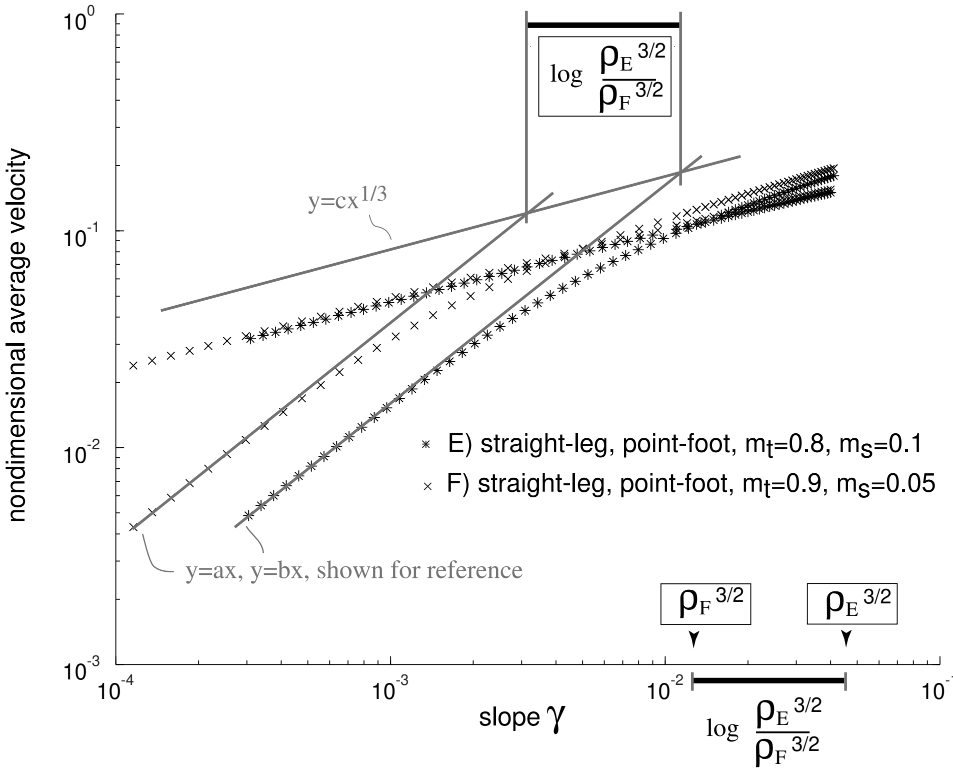

Figure 4.11: Scaling transition comparison for two point-foot walkers. Data for walker E is shown again in Figure 4.9. Parameters for walker E are shown in Table 4.1. Walker F has a foot mass of 0.05 and a hip mass of 0.9. At large ground-slopes ($\gamma \gg \rho^{3/2}$), the walker stance angles and velocities scale as $\gamma^{1/3}$. The predicted critical slopes for walkers E and F are shown at the bottom of the plot. Above these slopes, we expect the velocity to be proportional to the cube root of the ground-slope. The ratio of the two critical slopes is nearly identical to the ratio of the intersections of the two best-fit lines in the linear scaling regimes with a cube-root scaling line. The solution curves for the "simplest" walker would show as parallel lines more or less on top of the upper curves in this figure. Note that the "×" symbol used here refers to data from a different walker than in Figure 4.9.

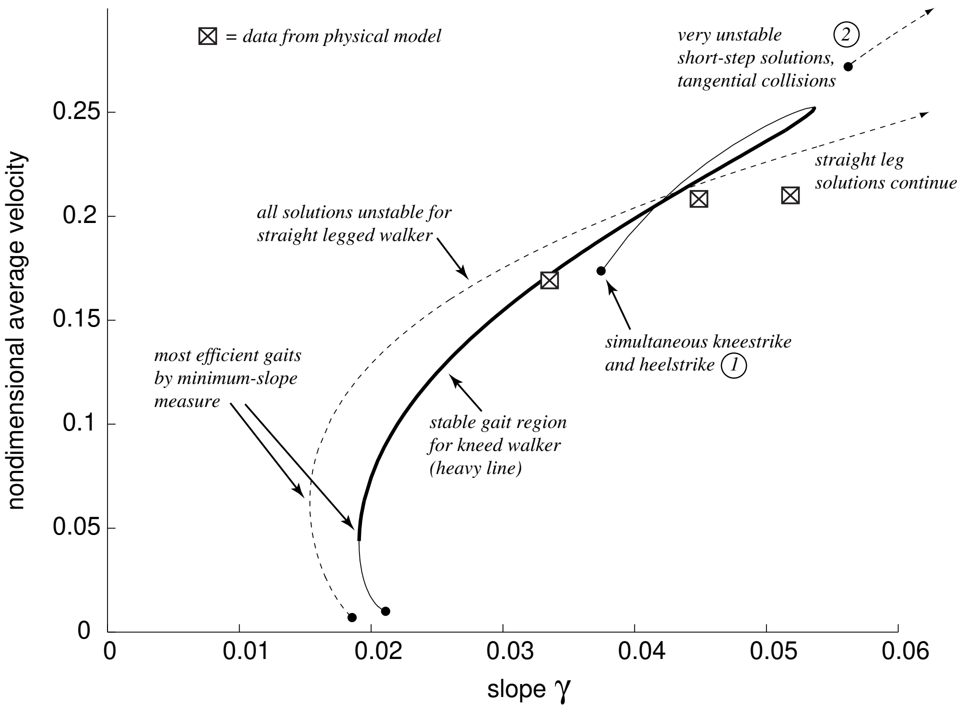

Figure 4.12: Numerically-calculated locus of solutions showing step velocity as a function of ground-slope for our physical kneed walking model (solid line) and for the same model but with the knees locked (dashed line) at fixed points. The thick portion of the solid line denotes stable solutions for the kneed walker.

Table 4.2: Predictions from simulation shown compared to data from videotape. Data was taken from successful walks down the entire length of a sixteen-foot ramp; generally the walker was allowed about three steps to converge onto its limit cycle and data from the last five or six steps were recorded and averaged. Below slopes of about 0.035, no gaits were found due to foot scuffing.

$\gamma = 0.034$

Characteristic	Simulation	Experiment
Step Period (s)	0.77	0.78
Step Length (m)	0.38	0.37
Step Velocity (m/s)	0.49	0.47
Nondimensional Velocity	0.17	0.17
Stance Angle (rad)	0.24	0.23

$\gamma = 0.045$

Characteristic	Simulation	Experiment
Step Period (s)	0.72	0.77
Step Length (m)	0.44	0.45
Step Velocity (m/s)	0.61	0.58
Nondimensional Velocity	0.22	0.21
Stance Angle (rad)	0.27	0.28

$\gamma = 0.052$

Characteristic	Simulation	Experiment
Step Period (s)	0.68	0.77
Step Length (m)	0.46	0.46
Step Velocity (m/s)	0.68	0.60
Nondimensional Velocity	0.24	0.21
Stance Angle (rad)	0.29	0.29

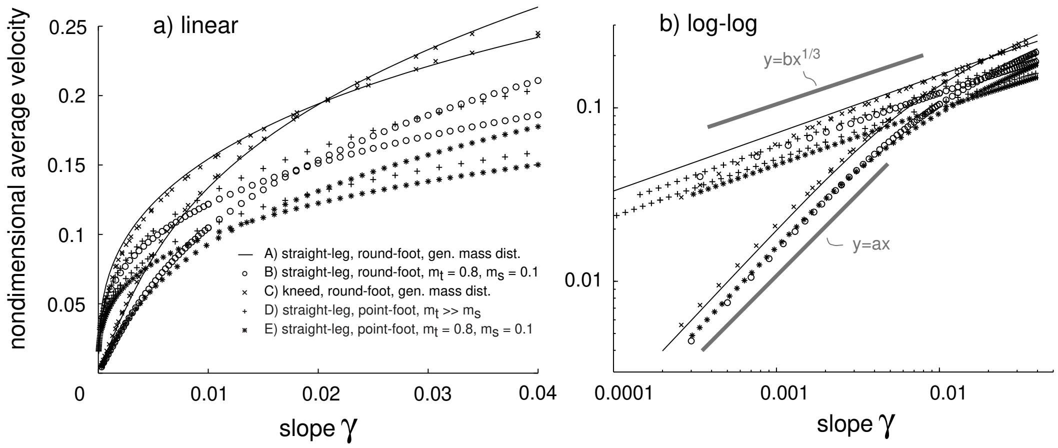

Figure 4.13: Gait families for tuned zero-slope-capable walkers on (a) a linear plot, and and (b) a log-log plot. Slope is plotted against average step velocity. Parameter values are listed in Table 4.1. By "gen. mass. dist." it is meant that the parameters are close to those of the kneed walker of Figure 4.3. Note that there are two gaits cycles at each γ for all walkers shown.

4.10 Appendix Of General 2-D Equations

This appendix did not appear with the original article as submitted.

4.10.1 Defining Parameters

The initialization file is run to set the initial conditions and to translate McGeer's parameters into those used by the derivative file. McGeer's parameters are shown in Figure 4.1. The derivative file parameters are shown in Figure 2.2. The numbers used below are those for the physical lab walker of Figure 4.2.

```
% Thigh parameters (m,kg,rad)
% without extra masses added
wt=0;         % McGeer's thigh cm offset
lt = 0.35;    % McGeer's thigh length (hip to knee)
ctold = 0.116;  % McGeer's thigh cm location down thigh,
```

```
                % before extra mass added
m(2) = 0.9868+0.8; % Thigh mass (0.8 kg added at cm of thigh)
emthigh = 0.1718;  % More extra mass added to thigh,
                   % at location
cem = 0.029845;    % cem down from hip axis

% masses added on thigh (calculated for 1 thigh)
% adjust thigh params to include extra (non-cm) mass added

% compute new cm location
% if wt was non-zero, wt should be re-computed also
ct = (ctold*m(2) + cem*emthigh)/(emthigh+m(2))

Izz2noem = 0.0181;  % I of thigh without extra mass.

% I of thigh with extra mass
Izz2wem = Izz2noem + emthigh*(cem-ct)^2+m(2)*(ctold-ct)^2;

% Compute radius of gyration (not really necessary)
rt = sqrt(Izz2wem/(m(2)+emthigh));

m(2) = m(2)+emthigh; % increment thigh mass

% Do the same thing with a hip mass
mh=0.773/2; % Add half the hip mass to each thigh
cth=(ct*m(2))/(m(2)+mh); % Recompute cm location of thigh
                         % cth is a dummy variable
% Recompute I about cm
Izz2whm=Izz2wem + mh*cth^2 + m(2)*(ct-cth)^2;

m(2)=m(2)+mh;  % Increment thigh mass
rt = sqrt(Izz2whm/m(2));  % Compute radius of gyration
ct=cth;   % Reset ct to new value

%Shank Parameters
m(3) = 1.013;  % mass of shank
rs = sqrt(0.03923/m(3)); % radius of gyration of shank.
  % This no. ^^^ is the desired shank Icm.

ls=0.46; % length from knee to bottom of foot through
         % center of foot ("ls line").

csold = 0.171; % cm offset down back of shank
```

```
wsold =0.014;  % cm offset perpendicular to shank
% This was John Camp's coordinate system. They will
% be translated to the McGeer frame below

m(1) = m(2)+m(3); % mass of whole leg (thigh + shank)

% slope, foot radius,  and gravity
gam=0.0495;  % slope
g = 9.81;   % gravitational constant
R=0.2;      % radius of foot
% since gam always appears as g*sin(gam) or
% g*cos(gam), use sgam and cgam to save some flops
sgam = sin(gam)*g;
cgam = cos(gam)*g;

% Epsilon values which define foot offset from leg
% eta is used instead of epsilon because of early
% misinterpretations of Greek alphabet characters
etak = 0.228;
etat = atan((ls-R)*sin(etak)/((ls-R)*cos(etak)+lt));
etaf = etak-etat;

cs=csold*cos(etak)+wsold*sin(etak); % cm dist down ls line
ws=csold*sin(etak)-wsold*cos(etak); % cm offset from ls line

% fixed point initial conditions
% fp for labwnks gam=0.036 em=0.8 and 0.1718 mh=0.773
theta1= 0.24395121657117;
theta2 = pi-etat-2*theta1;
theta3 = etak;
theta1dot= -1.04911493167539;
theta2dot= 0.74757611863057;
theta3dot= -1.18161338996657;

% store them in y0
y0 = [theta1 theta2 theta3 theta1dot theta2dot theta3dot];

% vectors from origin of frame to next frame
p01x = R;  % dist from bottom of foot to center of foot
p12x = lt*cos(etat)+(ls-R)*cos(etaf);
           % dist from center of foot to hip
p23x = lt; % dist from hip to swing knee
```

```
%vectors from origin of frame to cm of leg bit
pc2x = ct;  % cm x loc of thigh
pc2y = -wt; % cm y loc of thigh
pc3x = cs;  % cm x loc of shank
pc3y = -ws; % cm y loc of shank

%compute cm of thigh in coordinates of stance leg
p1c2x = -ct*cos(etat)+wt*sin(etat)+p12x;
p1c2y = ct*sin(etat)+wt*cos(etat);

%compute cm of shank in coordinates of stance leg
pcsx = -cs*cos(etaf)-ws*sin(etaf)+(ls-R)*cos(etaf);
pcsy = -cs*sin(etaf)+ws*cos(etaf)+(ls-R)*sin(etaf);
pcs=[pcsx pcsy]';

% compute cm of composite stance leg (thigh+shank)
pc1x = (m(2)*p1c2x+m(3)*pcsx)/m(1);
pc1y = (m(2)*p1c2y+m(3)*pcsy)/m(1);

% coordinates of stance knee from frame 1
p1kx = (ls-R)*cos(etaf);
p1ky = (ls-R)*sin(etaf);
p1k = [p1kx p1ky]';

%coordinates of hip from stance knee,
% used in stance knee torque check
pkct1x = lt*cos(etat);
pkct1y = -lt*sin(etat);

%moments of inertia of leg bits about centers of mass
Izz2 = m(2)*rt^2;
Izz3 = m(3)*rs^2;
Izz1 = Izz2+m(2)*((pc1x-p1c2x)^2+(pc1y-p1c2y)^2)+Izz3+m(3)*...
((pc1x-pcsx)^2+(pc1y-pcsy)^2);

% pc2lock is cm of swing leg when knee is locked
pc2lock(1)=(-pc1x+p12x)*cos(etat)+pc1y*sin(etat);
pc2lock(2)=(-pc1x+p12x)*sin(etat)-pc1y*cos(etat);

Izz = [Izz1 Izz2 Izz3]';
pc1 = [pc1x pc1y]';
pc2 = [pc2x pc2y]';
pc3 = [pc3x pc3y]';
```

```
tol = 1e-10;    % numerical tolerance
tfinal =10;     % final time for integration

% set structural constants
global K KK
initializeK_not;
initializeKK;
```

4.10.2 Defining Structural Constants

Constant vectors `K` and `KK` are defined in this file. They are used by the closed-form (meaning not on-line, with explicit formulas for **M**, **V**, and **G**) derivative files. They are computed at initialization and then stored in memory for future use; they are not state-dependent. They are constructed by deriving equations in Maple ® and then picking out all of the computations which only involve combinations of parameters.

```
pc1sq = pc1(1)^2+pc1(2)^2;
pc2sq = pc2lock(1)^2+pc2lock(2)^2;
pc3sq = pc3(1)^2+pc3(2)^2;

p12xsq = p12x*p12x;
p23xsq = p23x*p23x;
Rsq = R*R;

K(1)  = m(3)*pc3(1)*p23x;
K(2)  = m(3)*pc3(2)*p23x;

K(3)  = m(3)*pc3(1)*p12x;
K(4) = m(3)*pc3(2)*p12x;

K(5) = m(3)*pc3(1)*R;
K(6) = m(3)*pc3(2)*R;

m2pc21p12x = m(2)*pc2(1)*p12x;
K(7) = m(2)*pc2(2)*p12x;

m2pc21R = m(2)*pc2(1)*R;
```

```
K(8) = m(2)*pc2(2)*R;

m3p23xR = m(3)*p23x*R;
m3p23xp12x = m(3)*p23x*p12x;

K(9) = m3p23xp12x+m2pc21p12x;
K(10) = m3p23xR+m2pc21R;

m1pc11R = m(1)*pc1(1)*R;
K(11) = m(1)*pc1(2)*R;

twom1pc11R = 2*m1pc11R;
K(12) = 2*K(11);

twom2plusm3Rp12x = 2*(m(2)+m(3))*R*p12x;

K(13) = twom2plusm3Rp12x+twom1pc11R;
K(14) = K(13)/2;

K(15) = Izz(3)+m(3)*pc3sq;

K(16) = Izz(2)+pc2sq*m(2)+p23xsq*m(3);

K(17) = Izz(1)+pc1sq*m(1)+Rsq*(m(1)+m(2)+m(3)) ...
        +p12xsq*(m(2)+m(3));

K(18) = 2*K(1);
K(19) = 2*K(2);

K(20) = 2*K(3);
K(21) = 2*K(4);

K(22) = 2*m(3)*pc3(1)*R;
K(23) = 2*m(3)*pc3(2)*R;

twom2pc21p12x = 2*m2pc21p12x;
K(24) = 2*K(7);

twom2pc21R = 2*m2pc21R;
K(25) = 2*K(8);

twom3p23xR = 2*m3p23xR;
twom3p23xp12x = 2*m3p23xp12x;
```

```
K(26) = twom3p23xp12x+twom2pc21p12x;
K(27) = twom3p23xR+twom2pc21R;

K(28) = m(3)*(pc3(1)*sgam-pc3(2)*cgam);
K(29) = m(3)*(pc3(1)*cgam+pc3(2)*sgam);

K(30) = (pc2(1)*sgam-pc2(2)*cgam)*m(2)+p23x*m(3)*sgam;
K(31) = (pc2(1)*cgam+pc2(2)*sgam)*m(2)+p23x*m(3)*cgam;

K(32) = (pc1(1)*sgam-pc1(2)*cgam)*m(1)+p12x*sgam*(m(2)+m(3));
K(33) = (pc1(1)*cgam+pc1(2)*sgam)*m(1)+p12x*cgam*(m(2)+m(3));
K(34) = R*sgam*(m(1)+m(2)+m(3));

KK(1) = Izz(1)+m(1)*pc2sq;

KK(2) = m(1)*pc2lock(1)*R;
KK(3) = m(1)*pc2lock(2)*R;

KK(4) = m(1)*pc2lock(1)*p12x;
KK(5) = m(1)*pc2lock(2)*p12x;

KK(6) = m(1)*2*R*(pc1(1)+p12x);
KK(7) = m(1)*2*R*pc1(2);

KK(8) = Izz(1)-KK(1)+(2*Rsq+pc1sq+p12xsq)*m(1);

KK(9) = 2*KK(2);
KK(10) = 2*KK(3);

KK(11) = KK(6)/2;
KK(12) = KK(7)/2;

KK(13) = (pc2lock(1)*sgam-pc2lock(2)*cgam)*m(1);
KK(14) = (pc2lock(1)*cgam+pc2lock(2)*sgam)*m(1);

KK(15) = (pc1(1)*sgam-pc1(2)*cgam+p12x*sgam)*m(1);
KK(16) = (pc1(1)*cgam+pc1(2)*sgam+p12x*cgam)*m(1);
KK(17) = 2*R*sgam*m(1);
```

4.10.3 Defining Some Functions

The functions defined here are for taking cross products in 2-D.

```
function crossprod = ddcross(a,b);
% expect a to be third element of 3vector
% of the form [0 0 a], b=[b(1) b(2) 0]
% computes 2D crossproduct, returns scalar

crossprod = [-a*b(2) a*b(1)]';


function crossprod = crpr(a,b);
% computes 2D crossproduct, returns scalar
% a and b = [something1 something2 0]

crossprod = a(1)*b(2)-a(2)*b(1);
```

4.10.4 Kneed Closed-Form Equations

This function returns the state derivative `ydot` as a function of the state `y`. "Closed-form" means that there are expressions for **M**, **V**, and **G** that do not require any kind of recursion, as with the on-line approach. All of the equation derivation is done in Maple ®. The advantage of this form is that it involves less computation (fewer floating point operations) per integration step than the on-line approach. The disadvantage is that the code is more cryptic and less intuitive.

```
function ydot = yderivs_new_not(y,K);

% set sines and cosines
c01 = cos(y(1));
s01 = sin(y(1));

c02 = cos(y(2));
s02 = sin(y(2));

c03 = cos(y(3));
s03 = sin(y(3));
```

```
y12 = y(1)+y(2);
y123 = y(1)+y(2)+y(3);
y23 = y(2)+y(3);

c012 = cos(y12);
s012 = sin(y12);
c0123 = cos(y123);
s0123 = sin(y123);
c023 = cos(y23);
s023 = sin(y23);

v=zeros(3,6);
M=zeros(3,3);
g=zeros(1,3);

% set mass matrix M
M(3,3) = K(15);

t0 = K(1)*c03-K(2)*s03;

M(3,2) = M(3,3)+t0;
M(2,2) = M(3,2)+K(16)+t0;

t1 = K(3)*c023-K(4)*s023+K(5)*c0123-K(6)*s0123;

M(3,1) = M(3,2)+t1;

t3 = K(9)*c02-K(7)*s02+K(10)*c012-K(8)*s012;

M(2,1) = M(3,1)+K(16)+t3+t0;

t6 = K(13)*c01-K(12)*s01;

M(1,1) = M(2,1)+K(17)+t6+t3+t1;
M(1,2) = M(2,1);
M(1,3) = M(3,1);
M(2,3) = M(3,2);

% set centrifugal and coriolis coefficients
v(3,4) = K(18)*s03+K(19)*c03;
v(2,6) = -v(3,4);
v(2,5) = v(2,6);
```

```
v(2,4) = v(2,5);

t8 = K(20)*s023+K(21)*c023;
t9 = K(22)*s0123+K(23)*c0123;

v(1,6) = -t8-t9;
v(1,5) = v(1,6);

t10 = K(26)*s02+K(24)*c02;
t11 = K(27)*s012+K(25)*c012;
t13 = t10+t11;

v(1,4) = v(1,6)-t13;
v(3,2) = v(3,4)/2;
v(2,3) = -v(3,2);
v(2,2) = v(2,3);
v(3,1) = v(3,2)+t8/2;
v(2,1) = v(2,2)+t10/2;
v(1,3) = v(1,6)/2;
v(1,2) = v(1,3)-t13/2;

t15 = K(14)*s01+K(11)*c01;

v(1,1) = v(1,2)-t15;

% set gravity terms
g(3) = K(28)*c0123-K(29)*s0123;
g(2) = K(30)*c012-K(31)*s012;
g(1) = K(32)*c01-K(33)*s01+K(34);

products = [y(4)*y(4);y(5)*y(5);y(6)*y(6);
           y(4)*y(5);y(5)*y(6);y(4)*y(6)];
% set centrifugal and coriolis terms
vdiff = v*products+g';

vg(3) = vdiff(3);
vg(2) = vdiff(2)+vg(3);
vg(1) = vdiff(1)+vg(2);
vg=vg';

% compute ydot
% M is symmetric
X=chol(M);
```

```
z = X' \ (-vg);
Oddot=X \ z;

ydot =[y(4) y(5) y(6) Oddot(1) Oddot(2) Oddot(3)]';
```

4.10.5 Kneed On-Line Equations

These equations are equivalent to the ones above in subsection 4.10.4. They are included as an example of on-line equations. After rotation matrices are assigned, the equations more or less follow Equations 2.2 through 2.13 in Section 2.3, except that some steps are skipped. There is more complete correspondence in the 3-D code in Section 5.4.

```
function ydot = yderivs_turbo(y,m,R,sgam,cgam, ...
                Izz,p12x,p23x,pc1,pc2,pc3);
% the parameters are defined above.
% assign sine and cosine constants

cO1 = cos(y(1));
sO1 = sin(y(1));

cO2 = cos(y(2));
sO2 = sin(y(2));

cO3 = cos(y(3));
sO3 = sin(y(3));

% define rotation matrices

R01=[cO1 -sO1 ;sO1 cO1];
R12=[cO2 -sO2 ;sO2 cO2];
R23=[cO3 -sO3 ;sO3 cO3];

R10=R01';
R21=R12';
R32=R23';

% first construct [V+G] (all thetadotdots = 0)
% accelerate base frame upwards to account for gravity
```

```
vdot1= R10*[cgam sgam]'; % partial acceleration of frame 1

% m a of stance leg
F1 = m(1).*(ddcross(y(4),ddcross(y(4),pc1))+vdot1);

% angular velocity of frame 2
omega2 = y(4)+y(5);

% accel of frame 2
vdot2= R21*[-y(4)^2*p12x+vdot1(1) vdot1(2)]';

% m a of frame 2
F2 = m(2).*(ddcross(omega2,ddcross(omega2,pc2))+vdot2);

% angular velocity of frame 3
omega3 = y(4)+y(5)+y(6);

% accel of frame 3
vdot3= R32*([-omega2^2*p23x+vdot2(1) vdot2(2)]');

% m a of frame 3
f3 = m(3).*(ddcross(omega3,ddcross(omega3,pc3))+vdot3);

% assign V components
vg(3)=crpr(pc3,f3);
f2=R23*f3+F2;
vg(2)=vg(3)+crpr(pc2,F2)+(f3(1)*sO3+f3(2)*cO3)*p23x;
f1=R12*f2+F1;
vg(1)=vg(2)+crpr(pc1,F1)+(f2(1)*sO2+ ...
      f2(2)*cO2)*p12x-crpr([-R*cO1 R*sO1]',f1);
vg=vg';

% last term in vg(1) is torque due to rolling contact point
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% now construct M, matrix of thetadotdot coeffs
% construct first column by taking
% O1dotdot=1, O2dotdot=O3dotdot=0,
% all thetadots = 0
M=zeros(3,3);

% accel of frame 1
vdot1 = [R*sO1 R*cO1]';
```

```
% m a of stance leg
F1 = m(1).*([-pc1(2) pc1(1)]'+vdot1);

% accel of frame 2
vdot2 = R21*[vdot1(1) p12x+vdot1(2)]';

% m a of swing thigh
F2 = m(2).*([-pc2(2) pc2(1)]'+vdot2);

% accel of frame 3
vdot3 =  R32*[vdot2(1) p23x+vdot2(2)]';

% Sum (F) = m a of swing shank
f3 = m(3).*([-pc3(2) pc3(1)]'+vdot3);

% Sum(T) = I w for swing shank
M(3,1)=Izz(3)+crpr(pc3,f3);

%Sum(F) = m a of swing thigh
f2=R23*f3+F2;

% Sum(T) = I w for swing thigh
M(2,1)=Izz(2)+M(3,1)+crpr(pc2,F2)+ ...
       (f3(1)*sO3+f3(2)*cO3)*p23x;

% Sum(F) = m a of stance leg
f1=R12*f2+F1;

% Sum(T) = I w for stance leg
M(1,1)=Izz(1)+M(2,1)+crpr(pc1,F1)+ ...
       (f2(1)*sO2+f2(2)*cO2)*p12x-...
crpr([-R*cO1 R*sO1]',f1);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% now get second column of M in the same way
% theta2dot = 1, other thetadots = 0

% m a of swing thigh
F2 = m(2).*[-pc2(2) pc2(1)]';

% accel of frame 3
vdot3 =  [p23x*sO3 p23x*cO3]';
```

```
% Sum(F) = m a for swing shank
f3 = m(3).*([-pc3(2) pc3(1)]'+vdot3);

% Sum(T) = I w for swing shank
M(3,2)=Izz(3)+crpr(pc3,f3);

% Sum(F) = m a of swing thigh
f2=R23*f3+F2;

% Sum(T) = I w for swing thigh
M(2,2)=Izz(2)+M(3,2)+crpr(pc2,F2)+ ...
       (f3(1)*sO3+f3(2)*cO3)*p23x;

M(1,2)=M(2,1);  % Symmetry gets M(1,2) for free

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% now get third column of M
% theta3dot = 1, other thetadots = 0

% Sum(F) = m a for swing shank
f3 = m(3).*[-pc3(2) pc3(1)]';

% Sum(T) = I w for swing shank
M(3,3)=Izz(3)+crpr(pc3,f3);

% Symmetry gets the rest
M(2,3)=M(3,2);
M(1,3)=M(3,1);

% now solve for ydot

X=chol(M);
z = X' \ (-vg);
Oddot=X \ z;

ydot =[y(4) y(5) y(6) Oddot(1) Oddot(2) Oddot(3)]';
```

4.10.6 On-Line Kneestrike Equations

This function is applied when the swinging knee hits the knee stop. It returns the new angular velocities after the knee collision. See Section 2.5 for some derivation of the equations below.

```
function ynew = kneestrike_turbo(y,m,R,Izz,...
p12x,p23x,pc1,pc2,pc3);

% define sines and cosines

c01 = cos(y(1));
s01 = sin(y(1));

c02 = cos(y(2));
s02 = sin(y(2));

c03 = cos(y(3));
s03 = sin(y(3));

% define rotation matrices

R01=[c01 -s01 ;s01 c01];
R12=[c02 -s02 ;s02 c02];
R23=[c03 -s03 ;s03 c03];

R10=R01';
R21=R12';
R32=R23';

% angular velocities of swing thigh(2)
% and swing shank(3)
omega2 = y(4)+y(5);
omega3 = y(4)+y(5)+y(6);

% velocities of contact point(0),
% foot center(1), hip(2), and swing knee(3)
v0=R*y(4);
v11=[v0*s01 v0*c01]';
v22=R21*[v11(1) y(4)*p12x+v11(2)]';
v33=R32*[v22(1) omega2*p23x+v22(2)]';

% velocities of centers of mass of
% stance leg(1) swing thigh(2) and
% swing shank(3)
v1cm1 = v11+ddcross(y(4),pc1);
v2cm2 = v22+ddcross(omega2,pc2);
```

```
v3cm3 = v33+ddcross(omega3,pc3);

% sum angular momentum about hip before kneestrike
rhcm3 = [p23x*cO3 -p23x*sO3]'+pc3;  % goes from hip
% to cm of swing shank
hpre(2)=crpr(rhcm3,m(3).*v3cm3)+Izz(3)*omega3+...
 crpr(pc2,m(2).*v2cm2)+Izz(2)*omega2;

% sum angular momentum about floor before kneestrike
r0cm1 = [R*cO1 -R*sO1]'+pc1; % goes from contact point
% to center of mass of stance leg
r0cm2 = R21*[R*cO1+p12x -R*sO1]'+pc2;% goes from contact point
% to center of mass of swing thigh
r0cm3 = R32*(R21*([R*cO1+p12x -R*sO1]')+[p23x 0]')+pc3;
% from contact pt to cm of swing shank
hpre(1) = crpr(r0cm1,m(1).*v1cm1)+Izz(1)*y(4)+...
 crpr(r0cm2,m(2).*v2cm2)+Izz(2)*omega2+...
 crpr(r0cm3,m(3).*v3cm3)+Izz(3)*omega3;

% Now get matrix elements for hpost
% Do column 1 first, O1dot = 1, O2dot = 0

% set velocities of frames and cms assuming that
% O1dot = 1, others = 0
v11=[R*sO1 R*cO1]';
v22=R21*[v11(1) p12x+v11(2)]';
v33=R32*[v22(1) p23x+v22(2)]';

v1cm1 = v11+[-pc1(2) pc1(1)]';
v2cm2 = v22+[-pc2(2) pc2(1)]';
v3cm3 = v33+[-pc3(2) pc3(1)]';

% sum angular momentum about hip after kneestrike
rhcm3 = [p23x*cO3 -p23x*sO3]'+pc3;
hpost(2,1)=crpr(rhcm3,m(3).*v3cm3)+Izz(3)+...
 crpr(pc2,m(2).*v2cm2)+Izz(2);

% sum angular momentum about floor after kneestrike
r0cm1 = R10*[R 0]'+pc1;
r0cm2 = R21*(R10*[R 0]'+[p12x 0]')+pc2;
r0cm3 = R32*(R21*(R10*[R 0]'+[p12x 0]')+[p23x 0]')+pc3;
hpost(1,1) = crpr(r0cm1,m(1).*v1cm1)+Izz(1)+...
 crpr(r0cm2,m(2).*v2cm2)+Izz(2)+...
```

```
 crpr(r0cm3,m(3).*v3cm3)+Izz(3);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% get second column of hpost
% now O1dot = 0, O2dot = 1
% repeat procedure

% angular velocities
y(4)=0;
omega2=1;
omega3=1;

% frame and cm velocities
v33=R32*[0 p23x]';
v2cm2 = [-pc2(2) pc2(1)]';
v3cm3 = v33+[-pc3(2) pc3(1)]';

% sum angular momentum about hip after kneestrike
hpost(2,2)=crpr(rhcm3,m(3).*v3cm3)+Izz(3)+...
 crpr(pc2,m(2).*v2cm2)+Izz(2);
hpost(1,2) = hpost(2,1);
Odot = hpost \ hpre';

ynew=[y(1) y(2) y(3) Odot(1) Odot(2) 0];
```

4.10.7 Straight-Legged Closed-Form Equations

This function returns the state derivative for two-link mode (when the knee is locked). This would also be the derivative file used for straight-legged simulations.

```
function ydot = yderivstwo_new(y,KK);
% file is for 2D locked knees, with pc2  set
% for locked knee

% set sines and cosines

cO1 = cos(y(1));
sO1 = sin(y(1));

cO2 = cos(y(2));
```

```
s02 = sin(y(2));

y12 = y(1)+y(2);

c012 = cos(y12);
s012 = sin(y12);

v=zeros(1,3);
M=zeros(2,2);

% get M

M(2,2) = KK(1);

t0 = KK(2)*c012-KK(3)*s012;
t1 = KK(4)*c02-KK(5)*s02;

M(2,1) = M(2,2)+t0+t1;

t3 = KK(6)*c01-KK(7)*s01;

M(1,1) = 2*M(2,1)+t3+KK(8);
M(1,2) = M(2,1);

t4 = KK(4)*s02+KK(5)*c02;

t5 = KK(9)*s012+KK(10)*c012;

v(3) = -2*t4-t5;
v(2) = v(3)/2;

t6 = KK(11)*s01+KK(12)*c01;

% get V and G

v(1) = v(2)-t6;
g2 = KK(13)*c012-KK(14)*s012;

products = [y(4)*y(4);y(5)*y(5);y(4)*y(5)];

vg(2) = t4*products(1)+g2;
g1 = KK(15)*c01-KK(16)*s01+KK(17);
```

```
vg(1) = vg(2)+v*products+g1;
vg = vg';

% compute ydot

Oddot=M \ (-vg);

ydot = [y(4) y(5) 0 Oddot(1) Oddot(2) 0]';
```

4.10.8 On-Line Heelstrike Equations

This file swaps legs and computes the new angular rates just after heelstrike. This function is called at the instant when both legs are touching the ground. See Section 2.5 for derivation and explanation of the equations below.

```
function ynew= heelstrike_turbo(y,m,R,Izz,...
p12x,p23x,pc1,pc2,pc3,pcs,p1k,etat,ls);
% parameters have been defined above

% define sines and cosines
cO1 = cos(y(1));
sO1 = sin(y(1));

cO2 = cos(y(2));
sO2 = sin(y(2));

cO3 = cos(y(3));
sO3 = sin(y(3));

% define rotation matrices

R01=[cO1 -sO1 ;sO1 cO1];
R12=[cO2 -sO2 ;sO2 cO2];
R23=[cO3 -sO3 ;sO3 cO3];

R10=R01';
R21=R12';
R32=R23';

% angular velocities of frames 2 and 3
```

```
omega2 = y(4)+y(5);
omega3 = y(4)+y(5)+y(6);

% velocities of frames 0 through 3
v00=[0 R*y(4)]';  % contact pt
v11=R10*v00;      % foot center
v22=R21*(ddcross(y(4),[p12x 0])+v11);   % hip
v33=R32*(ddcross(omega2,[p23x 0])+v22); % swing knee

% velocities of cms
v1cm1 = v11+ddcross(y(4),pc1);  % stance leg
v2cm2 = v22+ddcross(omega2,pc2);% swing thigh
v3cm3 = v33+ddcross(omega3,pc3);% swing shank
v1k1 = v11+ddcross(y(4),pcs);   % stance shank

% sum angular momentum of stance shank about
% stance knee before heelstrike
% assume angular momentum conserved-
rkcs = -p1k+pcs; % from stance knee to cm of swing shank
hpre(3) = crpr(rkcs,m(3).*v1k1)+Izz(3)*y(4);

% sum angular momentum about hip before heelstrike
rhc1 = -[p12x 0]'+pc1;  % from hip to cm of stance leg
hpre(2)=crpr(rhc1,m(1).*v1cm1)+Izz(1)*y(4);

% sum angular momentum about new floor contact (cp)
% before heelstrike
r4cm1 = R10*[R 0]'+R12*((R23*[R-ls 0]')- ...
        [p23x 0]')-[p12x 0]'+pc1; % from new cp
  % to cm of old stance leg
r4cm2 = R21*R10*[R 0]'+(R23*[R-ls 0]')-[p23x 0]'+pc2;
  % from new cp to cm of old swing thigh
r4cm3 = R32*R21*R10*[R 0]'+[R-ls 0]'+pc3;
  % from cp to cm of old swing shank
hpre(1) = crpr(r4cm1,m(1).*v1cm1)+Izz(1)*y(4)+...
 crpr(r4cm2,m(2).*v2cm2)+Izz(2)*omega2+...
 crpr(r4cm3,m(3).*v3cm3)+Izz(3)*omega3;


% now swap angles. Old stance becomes new swing.
% Old swing becoes new stance.

y(1)=-y(1);
```

```
y(2)=pi-etat-2*y(1);

% redefine sines and cosines
s01 = -s01;

c02 = cos(y(2));
s02 = sin(y(2));

c03 = cos(y(3));
s03 = sin(y(3));

% new rotation matrices
R01=[c01 -s01 ;s01 c01];
R12=[c02 -s02 ;s02 c02];

R10=R01';
R21=R12';

% get hpost matrix of coefficients
% first, y(4)=1, y(5)=0 y(6)=0
y(4)=1;
omega2 = 1;
omega3 = 1;

% velocities of frames
v00=[0 R*y(4)]';
v11=R10*v00;
v22=R21*(ddcross(y(4),[p12x 0])+v11);
v33=R32*(ddcross(omega2,[p23x 0])+v22);

% velocities of cms
v1cm1 = v11+ddcross(y(4),pc1);
v2cm2 = v22+ddcross(omega2,pc2);
v3cm3 = v33+ddcross(omega3,pc3);
v1k1 = v11+ddcross(y(4),pcs);

% angular momentum about swing knee
hpost(3,1)=crpr(pc3,m(3).*v3cm3)+Izz(3)*omega3;

% angular momentum about hip
rhcm3 = R32*[p23x 0]'+pc3;
hpost(2,1)=crpr(pc2,m(2).*v2cm2)+Izz(2)*omega2+...
 crpr(rhcm3,m(3).*v3cm3)+Izz(3)*omega3;
```

```
% angular momentum about new contact point
r0cm1 = R10*[R 0]'+pc1;
r0cm2 = R21*(R10*[R 0]'+[p12x 0]')+pc2;
r0cm3 = R32*(R21*(R10*[R 0]'+[p12x 0]')+[p23x 0]')+pc3;
hpost(1,1)=crpr(r0cm1,m(1).*v1cm1)+Izz(1)*y(4)+...
 crpr(r0cm2,m(2).*v2cm2)+Izz(2)*omega2+...
 crpr(r0cm3,m(3).*v3cm3)+Izz(3)*omega3;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% now y(4)=0 y(5)=1 y(6)=0, get second column

% angular velocities of bits
y(4)=0;
omega2 = 0;
omega3 = 1;

% velocities of frames 0 - 3
v00=[0 R*y(4)]';
v11=R10*v00;
v22=R21*(ddcross(y(4),[p12x 0])+v11);
v33=R32*(ddcross(omega2,[p23x 0])+v22);

% velocities of cms
v1cm1 = v11+ddcross(y(4),pc1);
v2cm2 = v22+ddcross(omega2,pc2);
v3cm3 = v33+ddcross(omega3,pc3);
v1k1 = v11+ddcross(y(4),pcs);

% angular momentum about swing knee
hpost(3,3)=crpr(pc3,m(3).*v3cm3)+Izz(3)*omega3;

% angular momentum about hip
rhcm3 = R32*[p23x 0]'+pc3;
hpost(2,3)=crpr(pc2,m(2).*v2cm2)+Izz(2)*omega2+...
 crpr(rhcm3,m(3).*v3cm3)+Izz(3)*omega3;

% angular momentum about new contact point
r0cm1 = R10*[R 0]'+pc1;
r0cm2 = R21*(R10*[R 0]'+[p12x 0]')+pc2;
r0cm3 = R32*(R21*(R10*[R 0]'+[p12x 0]')+[p23x 0]')+pc3;
hpost(1,3)=crpr(r0cm1,m(1).*v1cm1)+Izz(1)*y(4)+...
 crpr(r0cm2,m(2).*v2cm2)+Izz(2)*omega2+...
```

```
 crpr(r0cm3,m(3).*v3cm3)+Izz(3)*omega3;

% solve for new angular velocities

Odot = hpost \ hpre';

ynew = [y(1) y(2) y(3) Odot(1) Odot(2) Odot(3)]';
```