In: Biological Cybernetics Research Trends
Editor: Thomas O. Williams, pp. -
ISBN: 978-1-60021-568-1


***Chapter 2***

# BIPEDAL WALKING FROM NEUROLOGICAL AND BIOMECHANICAL PRINCIPLES

***Sungho Jo and Andreas Hofmann***
Computer Science and Artificial Intelligence Laboratory, Media Laboratory,
Massachusetts Institute of Technology

## ABSTRACT

This chapter presents neurological and biomechanical principles of human bipedal walking and their potential application to humanoid walking robots by proposing two computational models. First, a computational model of cerebrocerebello-spinomuscular interaction during sagittal planar walking provides insight into each neural system's function based on neuro-anatomy and physiology. The neural systems substantially decouple gait cycle generation and postural stabilization, with a spinal pattern generator fulfilling the former function, and a cerebrocerebellar feedback system fulfilling the latter. A muscle synergy network facilitates control of redundant muscular actuators in descending pathways. Two control variables: horizontal position of the center of mass, and trunk pitch angle, are estimated from sensed information through the ascending neural pathways. Therefore, the space of the controller is simpler than the space of the actuators and plant. In this way, a simple control strategy is constructed.

The second computational model features a hierarchical task execution architecture that is suitable for control of a 3-D biped, for a variety of challenging walking tasks, including walking on difficult terrain with foot placement constraints. This approach supports exploration of performance limits based on biomechanical structure for such challenging tasks. The model uses an enhanced multivariable feedback linearizing controller, inspired by the muscle synergy approach used in the first computational model, to transform the biped into an abstracted plant that is easier to control. As with the first computational model, key control variables are the biped's center of mass, posture, and stepping foot position. The plant abstraction linearizes and decouples these variables, but the linearization has constraints based on actuation limits, and on limits of the biped's base of support. A reachability analysis is performed, in terms of the abstracted plant, in order to generate families of trajectories that satisfy task goals while observing constraints. Use of such trajectory sets supports high performance execution of difficult

tasks such as kicking a soccer ball, or dynamic walking on a path of irregularly placed stones, while rejecting disturbances that may occur.

## 1. INTRODUCTION

Bipedal walking on level terrain is characterized by a regular pattern of rhythmic stepping movements, where each step involves coordination of multiple muscles. The control problem is one of moving the center of mass of the biped forward while maintaining balance and posture. Maintaining balance involves controlling the center of mass momentum with respect to the base of support provided by the biped's feet.

Our first model, the *hypothetical neural walking controller,* addresses the problems of generation of rhythmic movement patterns, coordination of multiple muscles, and balance control, using a computational model of cerebrocerebello-spinomuscular interaction during sagittal planar walking. Because this model is based on a plausible neural anatomy, it provides insight into the function of each anatomical structure for human walking. Additionally, it makes sense to investigate use of such neural-based approaches for artificial bipeds since such approaches leverage a design that has been perfected by nature over millions of years of evolution.

Experimental studies support the existence of rhythmic pattern generation in humans [Dimitrijevic et al., 1998; Grasso et al, 2004; Calancie et al., 1994; Dietz and Harkema 2004]. Even though no explicit neural circuitry for such rhythmic pattern generation has been confirmed, the high probability of its existence has encouraged use of such pattern generation models to describe human biped walking [Taga, 1995; Ogihara and Yamazaki, 2001].

With respect to multiple muscular activation during walking, it is useful to begin with neuro-mechanical principles. Observations of muscular activations during gaits at different speeds [Ivanenko et al., 2004] indicate that the electromyogram (EMG*)* pattern remains unchanged principally. Several investigations [Ivanenko et al., 2004; Ivanenko et al., 2005; Olree and Vaughan, 1995] have used factor analysis to extract four or five principal waveforms from the muscular activations and tried to interpret the waveforms in terms of functions. Experimental studies demonstrate that tonic stimulation of the spinal cord in frogs induces synergistic patterns of muscular activations [Tresch et al., 1999; Cheung et al., 2005; d'Avella et al., 2003; d'Avella and Bizzi, 2005]. Such a mechanism simplifies redundant muscle control by providing a lower dimensional control for each leg. The lower dimensional control can analytically construct a wealth of frog leg EMG activities and behaviors in a feed-forward manner.

Rhythmic pattern generation, and muscle activation using synergies are feed-forward mechanisms. They generate a regular pattern of muscle activation without considering the actual state of the muscles and limbs being moved. Because disturbances occur, the actual muscle and limb state will differ from that intended by the feed-forward mechanism. To correct for this difference, feedback mechanisms must be utilized. It has been shown, experimentally, that the cerebrum and cerebellum augment the muscle activations provided by simpler feed-forward mechanisms [Kandel et al., 2000]. These augmentations are critical for reacting to disturbances and maintaining balance. Furthermore, it has been experimentally shown that selective lesions of descending control from the motor cortex compromise fine control of swing leg trajectories in cats [Drew, 1993]. The motor cortex has also been shown

to contribute to the structure and timing of the step cycle during locomotion in the intact cat [Bretzner and Drew, 2005]. Malfunction of the cerebellum due to either stroke or tumor in humans causes devastating effects in postural balance and locomotion [Porter and Lemon, 1993]. Thus, normal bipedal function in humans appears to depend significantly upon activity in transcerebral pathways [Nielsen, 2003; Peterson et al., 1998; Nathan, 1994]. Moreover, most cerebral activities during locomotion seem to be generated by sensory long-loop feedback mechanisms [Christensen et al., 2000; Nielsen, 2003]. For these reasons, a cerebrocerebellar long-loop feedback system must be considered in order to completely understand the neural mechanisms of human biped walking.

While it is important to understand walking fundamentals by studying normal human walking on level terrain, it is also important to understand and exploit the unique advantages provided by legged systems. The most obvious advantage of legged systems over wheeled vehicles is the ability to traverse un-even terrain. A normal human can traverse steps, climb rocks, cross brooks by stepping on stones, and in general, perform a wide variety of locomotion tasks that require irregular stepping patterns to traverse difficult terrain. The ability to perform such tasks is the primary reason for using legs, rather than wheels, for locomotion. Furthermore, humans perform such tasks in everyday life, and therefore, the ability to perform these tasks is a requirement for robotic devices intended for use in unstructured human environments.

For this reason, our second model focuses on execution of challenging locomotion tasks. This 3-dimensional, 18 degree of freedom model is used to investigate performance limits for such challenging tasks, taking into account the state and timing requirements of the tasks, and the actuation limits of the biped. As with the first model, the control architecture is functionally hierarchical, but it is not based on neural anatomy. Nevertheless, there are interesting parallels between the second model and the first at the functional level. The second model uses a *dynamic virtual model controller* to generate multiple actuation commands based on state goals. Thus, it performs a function similar to the synergistic muscle activation component in the first model. Above this controller in the functional hierarchy, the second model uses a task executive to monitor execution state and adjust control parameters. This function is analogous to that provided by the cerebrocerebellar long-loop feedback mechanism in the first model.

By comparing, and ultimately, merging key features of these two models, we expect to gain insight into the key principles of human walking, and how such principles can be used in the design of advanced artificial bipedal walking machines.

In the next section, we provide a review of common approaches used previously to control bipedal walkers, and we also review key requirements for bipedal balance control. In Section 3, we describe the first model, and in Section 4, we describe the second. Section 5 compares important features of each model, and discusses how these features could be merged in order to leverage the best features of both.

## 2. BACKGROUND

We begin this section with a brief review of previous control algorithms for walking bipeds. We then review the biomechanics of balance control, emphasizing the key

requirements that all bipedal walking controllers must satisfy. This is important for understanding the function of the two models described subsequently.

## 2.1. Control of Walking Bipeds

Over the past decade, a number of humanoid robots capable of walking have been developed. These include the Honda P3 and Asimo robots [Hirai, 1997, 1998], the Sony SDR [Yamaguchi, 1999], and Tokyo University's H6 [Kagami, 2001]. These systems generate detailed joint trajectories offline using dynamic optimization algorithms that observe dynamic limitations. These trajectories are then tracked using simple high-impedance PD control laws. However, this approach is not very robust to disturbances, since it depends on close tracking of the joint reference trajectories. If a disturbance occurs, tracking error can easily become too large due to actuation limits related to imperfect ground contact, and the system can lose its balance [Pratt and Tedrake, 2005].

The main problem with this approach is that use of high-impedance position control to track predetermined, detailed, joint reference trajectories results in a lack of compliance and robustness to force disturbances. The tracking controller will try to follow the predetermined trajectory no matter what, even if the situation requires a completely different response, such as modifying stepping foot placement, or using non-contact limb movement. Humans, on the other hand, are compliant to force disturbances in that they yield, when necessary, and are robust in that they can take complex compensating actions.

Achieving human-like performance using the high-impedance tracking method would require it to either generate, or somehow find, a new reference trajectory, quickly, when a significant disturbance occurs. Since generation of such reference trajectories, is computationally expensive, and since a very large number of such trajectories is needed to cover a wide range of disturbances, achieving human-like compliance and robustness is an unsolved problem for this method. Thus, although the method achieves stable walking on level terrain, it is brittle in to force disturbances, and performance on rough, uneven terrain, is poor.

## 2.2. Biomechanics of Balance Control

Balance control is essential for performing walking tasks robustly. Balance control requires the ability to adjust the biped's linear and angular momentum. Due to conservation of momentum laws, such adjustment can only be achieved through force interaction with the environment. For a biped, this force interaction is comprised of gravity and the *ground reaction force*, the net force exerted by the ground against the biped. In this section, we present an analysis of physical constraints and requirements for balancing. This leads to a simple set of balance control requirements that specify coordination of control actions that adjust the ground reaction force, and therefore, the momentum of the biped.

To derive this set of requirements, we make use of a number of physical points that summarize the system's balance state. These points are the *center of mass* (CM), the *zero-moment point* (ZMP) [Vukobratovic and Juricic, 1969], and the *centroidal-moment point* (CMP) [Popovic et al., 2005]. As we will discuss in more detail, the ZMP is a point on the

ground that represents the combined force interaction of all ground contact points. The CMP is the point on the ground from which the ground reaction force would have to emanate if it were to produce no torque about the CM.

We define the biped's support base as the smallest convex polygon that includes all points where the foot or feet are in contact with the ground. The ground reaction force vector, $\mathbf{f}_{gr}$, is then defined as the integral, over the base of support, of the incremental ground reaction forces emanating from each point of contact with the ground. This is expressed as

$$\mathbf{f}_{gr} = \iint_{B.O.S} \mathbf{f}_{gr}(x, y)dxdy$$

where $\mathbf{f}_{gr}(x, y)$ is the incremental force at point x,y on the ground, and B.O.S refers to the base of support region.

The CM is the weighted mean of the positions of all points in the system, where the weight applied to each point is the point's mass. Thus, for a discrete distribution of masses $m_i$ located at positions $\mathbf{r}_i$, the position of the center of mass is given by

$$CM = \frac{\sum_i m_i \mathbf{r}_i}{\sum_i m_i}.$$

A bipedal mechanism consists of a set of articulated links, each of which is a rigid body with mass $m_i$, and center of mass $\mathbf{r}_i$. The CM represents the effective mass of the system, concentrated at a single point. This is valuable because it allows us to simplify the balance control problem by reducing the problem to keeping the CM in the right place at the right time.

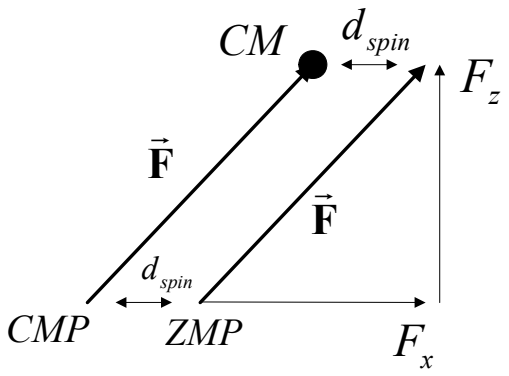
The ZMP [Vukobratovic and Juricic, 1969] also is a point that represents a combination of distributed points. It is defined as the point on the ground, where the total moment generated due to gravity and inertia is 0 [Takanishi et al., 1985]. This point has been shown to be the same as the *center of pressure* [Goswami, 1999], which is the point on the ground where the ground reaction force acts. Because the base of support is defined by the convex polygon of points in contact with the ground, and because the ZMP represents the average force contribution of these points, the ZMP is always inside the biped's base of support [Goswami, 1999].

The CMP [Popovic et al., 2005] is the point on the ground, not necessarily within the support base, from which the observed net ground reaction force vector would have to act in order to generate no torque about the CM. The relationship between the CM and CMP then indicates the specific effect that the net ground reaction force has on CM translation. Because the observed net ground reaction force always operates at the ZMP which is within the support base, whenever the net ground reaction force generates no torque about the CM, then the ZMP and CMP coincide, as shown in figure 2.2a. If the net ground reaction force generates torque, however, as shown in figure 2.2b, then the CMP and ZMP differ in location, as shown in figure 2.2c. In particular, the CMP may be outside the support base. In this case

the displacement of the CMP from the ZMP reflects the increased ability of the net ground reaction force to affect translation of the CM.

**a.**

**b.**




**c.**

Figure 2.2.a. If the ground reaction force vector points from the ZMP directly toward the CM, no moment is generated about the CM, and the ZMP and CMP coincide. b. When $\mathbf{f}_{gr}$ does not point towards the CM, a torque is generated about the CM. c. The CMP is the point where the ZMP would have to be in order for the ground reaction force vector to pass through the CM.

This capability of producing torque about the CM comes at an expense, however. While translational controllability of the CM is improved, angular stability about the CM is sacrificed. Thus, for example, the torso may deviate from its upright posture. In many situations, such a sacrifice is worthwhile if the angular instability is bounded and temporary. For example, a tightrope walker will tolerate temporary angular instability if this means that he will not fall off the tightrope.

Expressing requirements for balance in terms of CM, ZMP, CMP, and the support base is extremely useful for planning and control, due to the simplicity of this representation. Balance control is then reduced to a problem of adjusting the base of support, adjusting the ZMP within the base of support, and, if necessary, performing motions that generate angular momentum, so that the CMP can be moved, temporarily, outside the base of support, in order to exert additional compensating force on the CM.

## 3. HYPOTHETICAL NEURAL CONTROL OF BIPEDAL WALKING

The impression of most investigators is that during human locomotion, higher systems drive and modulate spinal level systems that are responsible for much of the basic patterning of muscle activity. The precise hierarchical partitioning of function has not yet been determined. However, several observations are relevant.

- Muscular activation and leg function during locomotion [Ivanenko et al., 2004] indicate that the *electromyogram* (EMG*)* pattern is consistent across movement speeds, with only duration and intensity changing.
- Experimental observations support a spinal locus for important rhythmic locomotor EMG pattern generation in humans [Dimitrijevic et al., 1998; Grasso et al., 2004; Calancie et al., 1994; Dietz and Harkema, 2004] often in response to tonic electrical stimulation [Dimitrijevic et al., 1998].
- Experimental stimulation of either the cerebellum or midbrain produces rhythmic locomotor movement in both intact and decerebrate cat with vigor and frequency that increase with stimulus intensity [Mori et al., 1999; Mori et al., 1998]. These locomotor regions both strongly recruit vestibulospinal, reticulospinal and other direct spinal efferent pathways [Shik and Orlovsky, 1976]. Either cerebellar or midbrain locomotor regions can presumably drive and potentially modulate a spinal locomotor control system [Grillner, 1975; Mori et al., 1999; Mori et al., 1998; Shik and Orlovsky, 1976; Kandel et al., 2000].
- Tonic stimulation of a frog spinal cord demonstrates synergistic patterns of muscle activities [Tresch et al., 1999; Cheung et al., 2005; d'Avella et al., 2003; d'Avella and Bizzi, 2005]. Such a mechanism collapses multiple muscle control to a lower degrees of freedom control for each leg.

These findings suggest that a rhythmic central pattern generator, modulated by the supraspinal system, may interact with muscle control synergies to provide a simple and effective walking control.

## 3.1. Methods

### *3.1.1. Musculoskeletal Model*

Walking dynamics is implemented by repetitive interaction between body and ground in the gravitational field. In this study, we used rigid body dynamics to construct a human body model to implement sagittal planar walking motions. Three joints: ankle, knee, and hip, for each leg are connected with body segments: trunk with head, upper leg, and lower leg. The interaction between the feet and the ground is modeled by a nonlinear impedance as generally used for computational modeling [van der Kooij et al., 2003]. Each leg is surrounded by a total of nine muscles, that is to say, six mono- and three bi-articular muscles: dorsiflexor, plantarflexor, knee extensor, knee flexor, hip extensor, hip flexor, biarticular knee-hip extensor, biarticular knee-hip flexor, and biarticular ankle-knee flexor as indicated in figure 3.1.

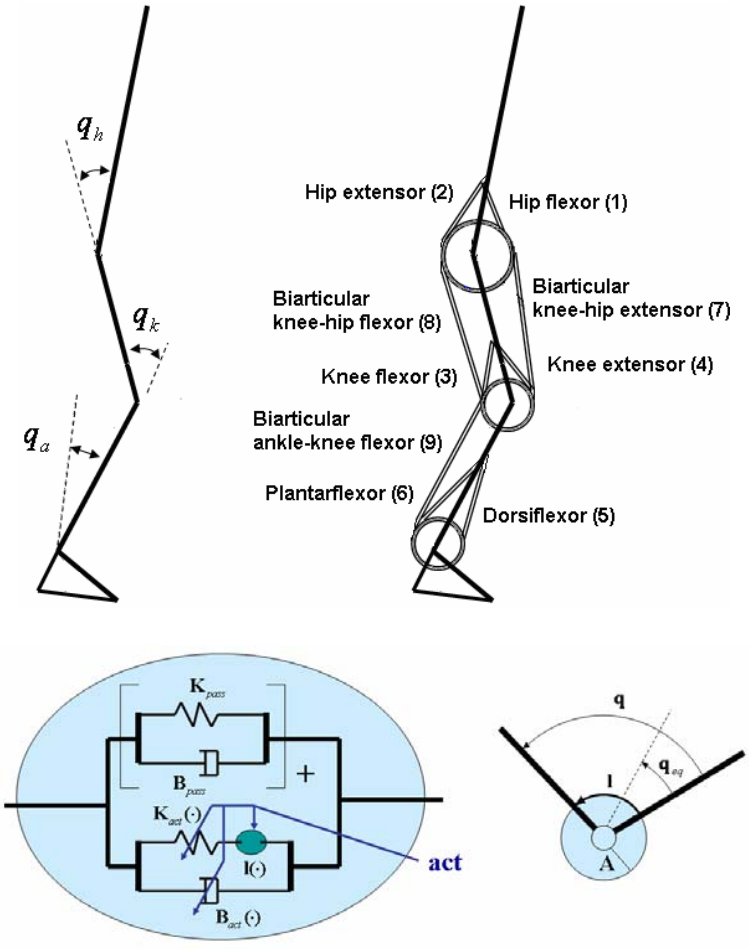



Figure 3.1. Body configuration: each leg has ankle ( $q_a$ ), knee ( $q_k$ ) and hip ( $q_h$ ) joints and are surrounded by nine muscles. For simplicity, only one leg is illustrated. Each muscle is modeled as the circuit in the oval. Geometric relationship between joint angle and muscle length is also illustrated (For simplicity, they are expressed in scalars).

Each muscle consists of passive and active force generators. Each component is described by a nonlinear impedance, but the active one is activation-dependent as follows:

$$\begin{aligned}\mathbf{F}_{pass} &= \left[\mathbf{K}_{pass}(\mathbf{l}_{eq}-\mathbf{l})-\mathbf{B}_{pass}\dot{\mathbf{l}}\right]_{+} \\ \mathbf{F}_{act} &= \mathbf{K}_{act}(\mathbf{act})\left[\mathbf{l}_{eq}-\mathbf{l}\right]_{+}-\mathbf{B}_{act}(\mathbf{act})\dot{\mathbf{l}}\end{aligned} \tag{3.1}$$

where $\mathbf{l}$ is the muscle length vector; $\dot{\mathbf{l}}$ the muscle length change rate vector; $\mathbf{l}_{eq}$ intrinsic muscle length at equilibrium position; $\mathbf{K}_{pass}$, $\mathbf{B}_{pass}$ are respectively the passive muscle stiffness and viscosity matrices; $\mathbf{K}_{act}(\mathbf{act})$, $\mathbf{B}_{act}(\mathbf{act})$ are respectively the active muscle stiffness and viscosity matrices depending on activation signal $\mathbf{act}$ [Jo, 2006]; $\mathbf{l}(\mathbf{act}) = \mathbf{l}_{eq} + \mathbf{act}$ is computationally assumed.

The joint angles and muscle lengths hold the following relation through the moment arm matrix $\mathbf{A}$ as in figure 3.1 [Jo and Massaquoi, 2004].

$$\mathbf{l} = \mathbf{l}_{eq} + \mathbf{A}(\mathbf{q} - \mathbf{q}_{eq}) \tag{3.2}$$

where $\mathbf{q}$ is joint angle vector; $\mathbf{q}_{eq}$ is joint angle vector at equilibrium. Here, the moment arms are assumed to be constants over motions for computational simplicity [Ogihara and Yamazaki, 2001; Jo, 2006].

The generation of muscular activation by neural signal is approximately described by low-pass dynamics in the form of:

$$EC(s) = \frac{\rho^2}{(s+\rho)^2} \qquad \rho = 30 \text{ rad/sec} \text{ [Fuglevand and Winter, 1993],}$$

and

$$\mathbf{act} = EC(s)\left(\mathbf{u}_{\alpha}\right) \tag{3.3}$$

where $\mathbf{u}_{\alpha}$ is the command from $\alpha$ motorneuron and $s$ is the Laplace varaible.

### *3.1.2. Spinal Pattern Generator*

For further simplicity and generality, a hypothetical pulse generator is considered to have binary output. At the level of the spinal pulse generator, the locomotor function is viewed in terms of five hypothetical *control epochs*: “loading” (LOA), “regulation” (REG), “thrust” (THR), “retraction” (RET) and “forward” (FOW) (figure 3.3). The last is almost, but not precisely coextensive with the actual swing phase. These epochs are empirically selected. If the command output of each epoch is designated $u_{PG,i}(t)$, then its periodic activation can be modeled as rectangular pulse:

$$u_{PG,i}(t) = \eta_{PG} \cdot 1\left[\cos(2\pi f_{PG} t - \phi_i) - h_i\right]_+ \quad i = 1,2,3,4. \tag{3.4}$$

where $1[x]_+ = 1$ where $x > 0$ and 0 otherwise.

$f_{PG}$ determines the pattern frequency, $\eta_{PG}$ is an activation intensity factor, $\phi_i$ is the phase shift, and $h_i$ activity discharge threshold. Specification of $\phi_i$ and $h_i$ determines the sequence and potential overlap between the pulse activations. Figure 3.2 illustrates the four pulses corresponding to the control epochs.

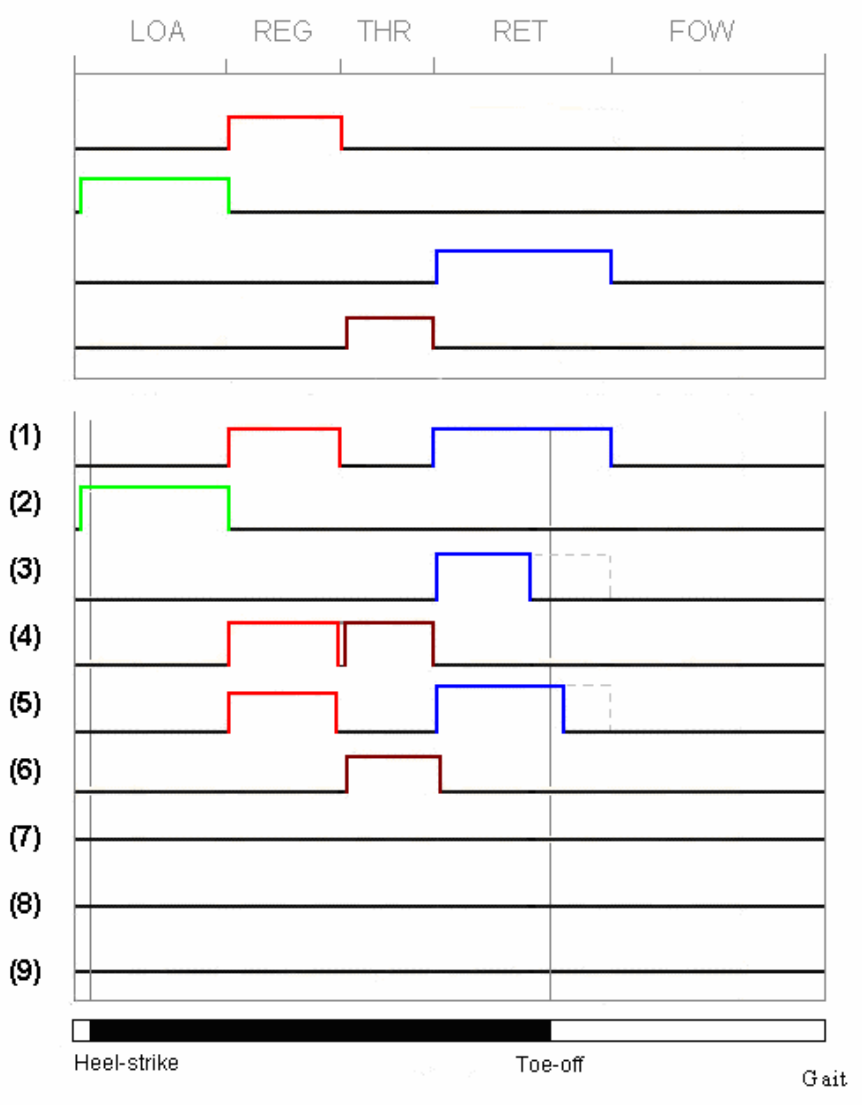



Figure 3.2. Pulses from spinal pattern generator (top) and spinal signals to muscles (bottom). Numbers indicate corresponding muscles in figure 3.1.

It is proposed that pulse generator commands are distributed to the muscles through a spinal network represented by the matrix $\mathbf{W}_{pg}$ according to five functional epochs during the gait cycle. No command is activated during the fifth epoch so that the swing phase is almost passive.

$$\mathbf{u}_{sp} = \mathbf{W}_{pg}\mathbf{u}_{pg} \tag{3.5}$$

where $\mathbf{u}_{pg}$ is the pulse command vector, and $\mathbf{u}_{sp}$ is the vector of spinal patterns to muscles.

The combination of the spinal pulse generator and the synergy distribution network can be considered a neural pattern generator (NPG). A major feature of the NPG is that only a few simple pulses are required to generate sequential activations over the whole muscles as in figure 3.3. The columns in the spinal network $\mathbf{W}_{pg}$ are comparable with synergies [Tresch et al., 1999; Cheung et al 2005; d'Avella et al., 2003] even though the concept is not exactly identical. We call each column in $\mathbf{W}_{pg}$ a locomotor synergy. For computational simplicity, synergies only related to the mono-articular muscle activities were formed for present research.

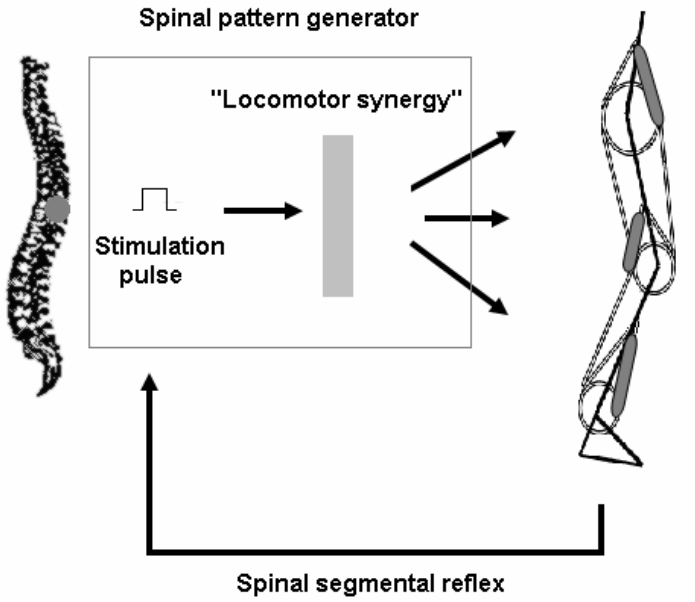



Figure 3.3. Locomotor synergy in the spinal pattern generator.

### *3.1.3. Supraspinal and Spinal Feedback Systems*

Pure feedforward systems such as synergies are not effective due to lack of robustness. Feedback systems effectively compensate for unexpected disturbances or noise. We assume that two feedback systems in the human body are significant. One is called the supraspinal (long-loop) feedback system, circulating between muscle to the cerebrocerebellar system, and

the other is called the spinal (segmental reflex) feedback system, indicating a pathway between muscle and spinal cord. Without feedback components, signals from the pattern generator have to be very accurate and sophisticated to implement stable walking; even small disturbances would easily destabilize walking.

#### 3.1.3.1. Spinal (Segmental) Reflex Feedback

In viewpoint of the system, the spinal pattern generator provides pure feedforward commands to generate the stereotyped patterns. It is known that several spinal reflexes in reality effect the neural patterns to improve walking morphology by using peripherally sensed information [Baxendale and Ferrell, 1981; Brooke et al., 1997; Grillner, 1975; Duysens et al., 2000]. Peripheral segmental reflex modifies certain synergy components. In this way, a few number of simple feedforward patterns are synergetically sufficient to construct overall muscle activations. For the present model, a possible neural circuit, presynaptic inhibition, is embedded [Baxendale and Ferrell, 1981; Brooke et al., 1997; Rossignol et al., 2006; Duysens et al., 2000]. A descending signal conveys a tonic excitation $q_{th,j}$, $j = a,k,h$ (ankle, knee, hip) that inhibits the proprioceptive afferent $q_j$ at joint $j$ until superseded. Each $q_{th,j}$ is a constant threshold value. Thereafter, the interneuron is activated and the motor neuron activity is suppressed. It is speculated herein that such a mechanism could truncate activities in early FOW to prevent excessive leg retraction. This mechanism is empirically useful to improve the timing between knee and ankle motions to prevent ground contact during swing phase. The spinal reflex action is therefore modeled as:

$$\mathbf{u}_{reflex} = -\mathbf{W}_S \cdot 1\lfloor \mathbf{q}(\mathbf{t} - \mathbf{T}_{pr}) - \mathbf{q}_{th} \rfloor_+ \tag{3.6}$$

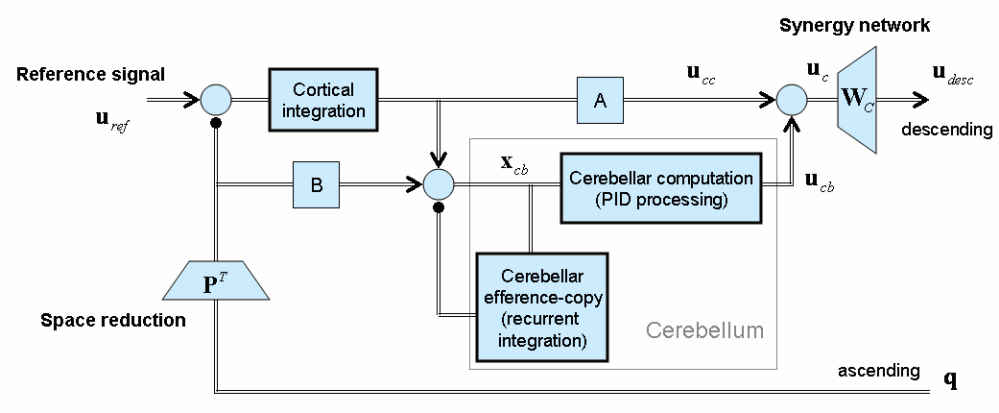
where $\mathbf{q}_{th} = \begin{bmatrix} q_{th,a} & q_{th,k} & q_{th,h} \end{bmatrix}^T$, and $\mathbf{W}_S$ is a matrix that scales and distributes joint-related signals from the uniarticular muscles to the other muscles via the vector $\mathbf{u}_{reflex}$. $\mathbf{T}_{pr}$ represents neural transmission delays from spinal cord to muscle. The inhibitive effect of the reflex action is indicated by dotted lines in figure 3.2.

#### 3.1.3.2. Supraspinal Feedback

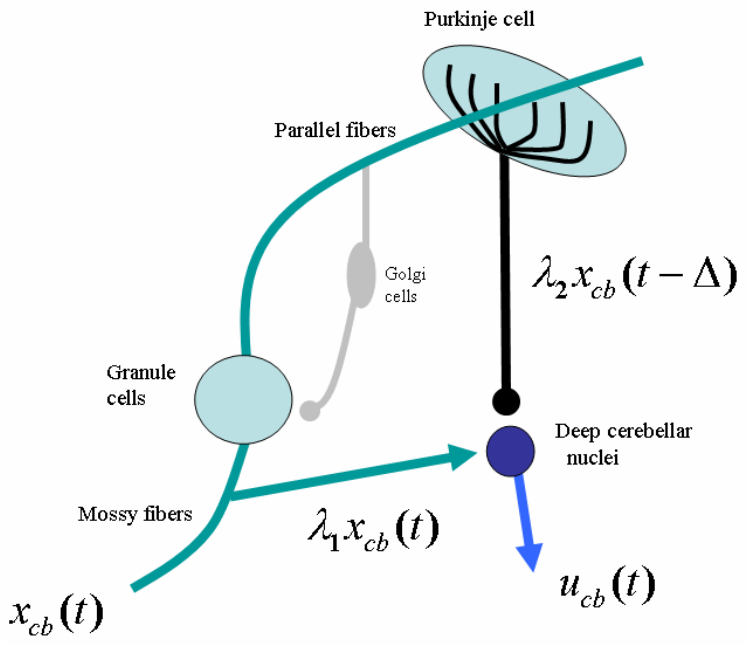
We implemented computational walking tests of our model for the case where the supraspinal system is missing. With only the pattern generator and spinal segmental reflex feedback, the computational model took several forward steps initially but soon fell down. Physiologically, an intact motor cortex is believed to be a prerequisite for bipedal walking, although conclusive data is still lacking [Nielsen, 2003]. It is observed that corticospinal cells are always modulated during walking, which means the supraspinal system participates in walking control somehow. Lesion studies support the necessity of the supraspinal feedback for integrated walking [Porter and Lemon, 1993; Nielsen, 2003; Peterson et al., 1998].

The cerebrocerebellar feedback system (supraspinal control) used in this study is illustrated in figure 3.4a. The model is adapted from a balance control model in Jo and Massaquoi, 2004. In figure 3.4a, the cortical integration represents the inherent input-output characteristic of the main neurons of the motor cortex as proposed in [Karameh and Massaquoi, 2005]. Gains A and B affect the relative balance of cortical and cerebellar

circuitries. The activity related to the recurrent feedback circuit between the cerebellar output and input is implemented by integration [Massaquoi and Topka, 2002] and the local close-loop provides phase lead compensation to the cerebellar input signal. The signal on the projection pathway may be interpreted to comparably be "efference-copy" discharge [Allen and Tsukahara, 1974].




**a.**




**b.**

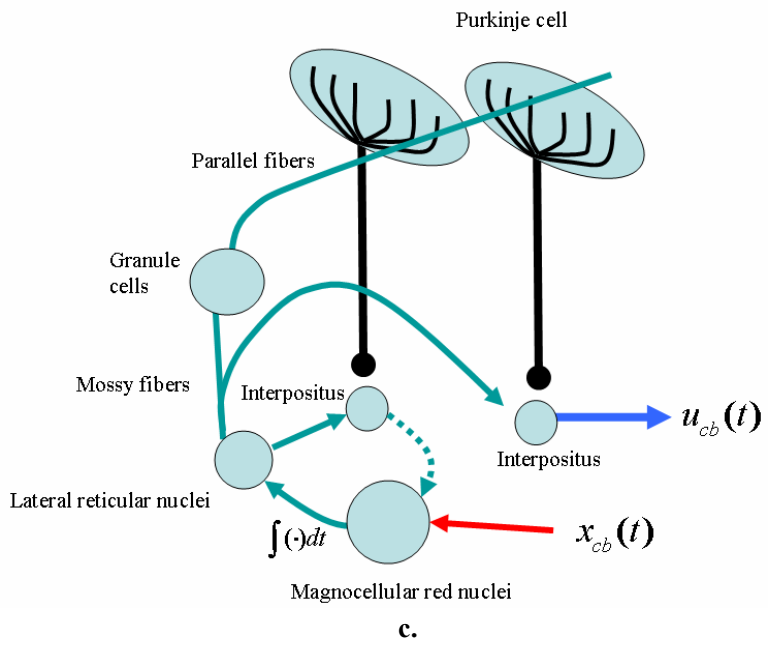



Figure 3.4. a. A model of the cerebrocerebellar feedback system (a filled circle indicates inhibition), b. the neural circuit of the cerebellar computation, c. integration by precerebellar nuclei.

Ito (1997) proposed that cerebellar processing is performed by functional corticonuclear microcomplexes. Under the framework, the cerebellum can be regarded to compute proportional scaling, integration, and differentiation [Massaquoi and Topka, 2002]. Figure 3.4b and c illustrate cerebellar neural circuitry. It does not include the climbing fiber-related neural circuitry known to process adaptation because the adaptation is beyond this chapter interest. The deep cerebellar nuclei conveys the output signal $u_{cb}(t)$. The output signals from the cerebellar computation are sent to other areas through different deep cerebellar nuclei depending on the regions of the cerebellum. Medial, intermediate and lateral regions, respectively, project via fastigial, interpositus, and dentate deep nuclei. It is known that the fastigial conveys information on upright stance and gait, and interpositus on reaching movements or alternating agonist and antagonist muscles, and dentate on coordinating movements [Thach, 1998]. The cerebellar input signal $x_{cb}(t)$ is transmitted through mossy fibers. The input excites the deep cerebellar nuclei directly and the Purkinje cell through parallel fibers. Then, the Purkinje cell inhibits the deep cerebellar nuclei. Therefore, there are two neural pathways between the input and output signals (figure 3.4b). The neural activities through the pathways are modeled as follows:

$$\lambda_1 x_{cb}(t) - \lambda_2 x_{cb}(t-\Delta) = \lambda_1 (x_{cb}(t) - x_{cb}(t-\Delta)) + (\lambda_1 - \lambda_2) x_{cb}(t-\Delta) \qquad (3.7)$$

where $\Delta$ is neural transmission delay and constants $\lambda_1$, $\lambda_2$ represents intensity of activity .

The first term above on the right hand side is interpreted as a differential operation in the continuous domain, and the second as a scaling operation. As a result, the input and output signals of the cerebellar computational circuitry are in the relation of

$$u_{cb}(t) = g_d \dot{x}_{cb}(t) + g_p x_{cb}(t) \tag{3.8}$$

where $g_d$ represents the cerebellar derivative gain, and $g_p$ the cerebellar proportional gain.

In addition, it is suggested that signal processing from magnocellular red nuclei to lateral reticular nuclei is interpreted as integration [Jo and Massaquoi, 2004; Jo, 2006]. Then, the cerebellar input signal through mossy fibers now becomes $\int x_{cb}(t)dt$ where input $x_{cb}(t)$ is fed to the signal processing from magnocellular red nuclei to lateral reticular nuclei, and then transmitted to deep cerebellar nuclei (especially, interpositus in medial cerebellum) and also conveyed to granule cells through mossy fibers as in figure 3.4c [Allen and Tsukahara, 1974]. When Eq. (3.8) is applied to this new input, the output signal is in the form of

$$u_{cb}(t) = g_p x_{cb}(t) + g_i \int x_{cb}(t)dt \tag{3.9}$$

Therefore, the overall cerebellar computation, summation of Eq. (3.8) and (3.9), is represented as Proportional-Integral-Derivative (PID) signal processing.

$$u_{cb}(t) = g_p x_{cb}(t) + g_i \int x_{cb}(t)dt + g_d \dot{x}_{cb}(t) \tag{3.10}$$

A different set of PID gains will be selected to represent different regional signal processings for different behaviors. As mentioned previously, cerebellar computation is regionally distinguished. For example, arm motions are usually at higher frequencies than leg motions. Therefore, the derivative component will be more effective for arm motion control. For this walking model problem, it turns out empirically that proportional control mainly operates. Considering the lead compensation of the local closed-loop, the signal related to proportional control is velocity-like in fact.

The cerebellar PID gains are also adaptively tunable. Plasticities such as Long Term Depression (LTD) or Long Term Potentiation (LTP) etc [Ito, 2001] in the cerebellar cortex are computationally comparable with the gain tuning. This chapter does not go deeply into the adaptation but the manipulation of walking behavior. However, the second model may provide insights into automatic gain tuning.

It may be reasonable to assume that the supraspinal system manages the sensorimotor activities in specified coordinates rather than fully concerns the whole redundant sensory information though it is not completely verified yet. In this way, the control problem gets simplified because control variable dimensionality is reduced. Each columnar assembly in sensorimotor area 3a is expected to contain a specific feature presentation of sensory information [Huffman and Krubitzer, 2001]. Presumably it is the spot for such control space reduction. Further research is required for confirmation, however. Two control variables, the body's CM ($x_{com}$) and trunk pitch angle ($q_{com}$), are empirically chosen for the cerebrocerebellar control channels. Recent experimental study indicates the two variables are critical to regulate human postural balance [Freitas et al., 2006]. Presumably the supraspinal feedback system is still in charge of balance control during normal walking as it is during postural balance. It may be possible to regard walking as extension of postural balance in

similar mechanical principle. In both walking and postural balance, the upper body desirably remains close to the vertical line so that the posture helps control the body's CM within stable supporting area [Gilchrist and Winter, 1997]. Walking requires a sequential postural balance control in dynamical manner. Precisely speaking, $x_{com}$ indicates the forward position of the body's CM relative to the stance foot. It is assumed that the two control variables are approximately estimated from sensed information and represented in the form of:

$$\begin{aligned} \hat{x}_{com} &= \mathbf{p}_1^T \mathbf{q}(\mathbf{t} - \mathbf{T}_{spr}) \\ \hat{q}_{com} &= \mathbf{p}_2^T \mathbf{q}(\mathbf{t} - \mathbf{T}_{spr}) \end{aligned} \tag{3.11}$$

where $\mathbf{T}_{spr}$ represents neural transmission delays from muscle to supraspinal system.

A weighted linear combination of sensory information as above is the inner product of a population vector $\mathbf{p}$ and a vector of sensory information [Georgopoulos, 1988]. Their tonic reference signals $x_{com,ref}$ and $q_{com,ref}$ are specified in cortical cortex. For a steady walking, constant offset $x_{com,ref}$ and verticality of head-trunk segment, $q_{com,ref} = 0$, are desired.

The cerebrocerebellar signal processing is applied separately to each control variable, therefore, two long-loop feedback pathways are coexistent. Still, it is possible to walk or even run with the trunk bent forward or backward, and during running the trunk is maintained erect even while there is no ground contract. Therefore, it is plausible that regulations of trunk pitch and relative CM position are managed by separate neural circuits. The interpositus nucleus projects to the cortex and could be involved in the cerebrocerebellar coordination. The interpositus also is known to be involved in arm and leg control. Therefore, it may be related to the CM relevant control loop. The fastigial nucleus receives direct connections from spinal cord connects the vestibular nucleus and then back down to spinal cord, therefore, perhaps it is involved with the trunk pitch control loop.

When $\mathbf{u}_c$ represents the neural output from the cerebrocerebellar system, $\mathbf{u}_{cc}$ the output from parietal or motor cortices that bypass the cerebellum, and $\mathbf{u}_{cb}$ the output from the cerebellar signal processing,

$$\mathbf{u}_c = \mathbf{u}_{cc} + \mathbf{u}_{cb}$$

and

$$\mathbf{u}_{desc} = \mathbf{W}_c \mathbf{u}_c(\mathbf{t} - \mathbf{T}_{sp}) \tag{3.12}$$

where $\mathbf{W}_c$ is a matrix to represent the distribution network presumably in cerebral cortical area 4, and $\mathbf{u}_{desc}$ the descending signal from the supraspinal system to muscles.

The distribution network functions as if it is the inverse of the space reduction $\mathbf{p}^T = \begin{bmatrix} \mathbf{p}_1 & \mathbf{p}_2 \end{bmatrix}^T$ so that it spreads out neural signals to the whole muscles. We call it the

supraspinal synergy network. $\mathbf{T}_{sp}$ represents neural transmission delays from supraspinal to spinal systems.

The alpha motor neuronal signal $\mathbf{u}_{\alpha}$ is obtained by summing the overall neural signals from supraspinal and spinal systems (Eq. (3.5), (3.6) and (3.12)):

$$\mathbf{u}_{\alpha} = \mathbf{u}_{desc} + \mathbf{u}_{sp} + \mathbf{u}_{reflex} \tag{3.13}$$

The alpha motor command is applied to Eq. (3.3) to produce muscular activations.

## 3.2. Result

Neural systems explained in the previous section are integrated as in figure 3.6 to build the whole biped walking model. The model is designed under the following assumptions for tractability of the initial study and simple requirements.

1. Movements will be mainly driven by monoarticular muscle activations.
2. Spinal rhythmic patterns are represented by a sequence of on-off pulses.
3. Inter-leg coordination is achieved by 180 degree phase difference between the pattern generators in each leg.

With a set of initial conditions, model parameters are determined to achieve stable walking simulations. At the first step, neural control only with feedforward spinal pattern generation is simulated. The parameters in the pattern generator are tuned for the model to take a couple of steps before falling. Then, the spinal segmental feedback is added to achieve smooth swing motions. Finally, the supraspinal feedback is added to maintain stable walking motions. No optimization for tuning parameters is intended because the goal of the study is the explanation of the principles behind the system and its performance, rather than achieving optimal performance.

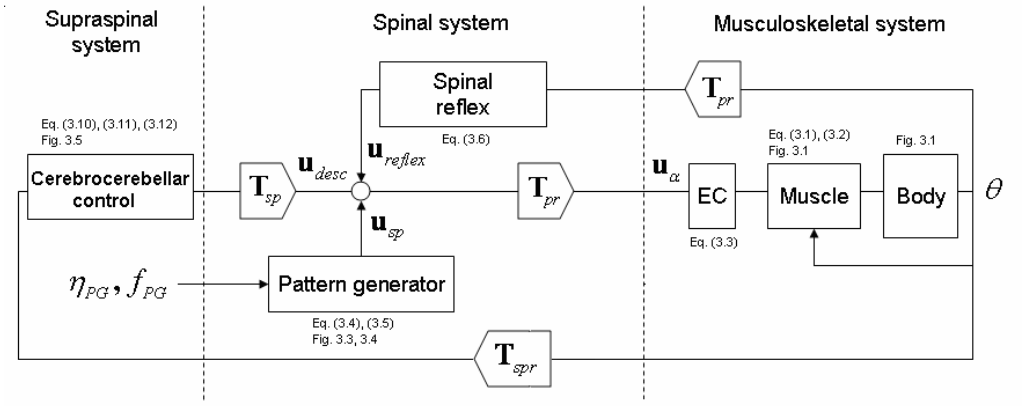
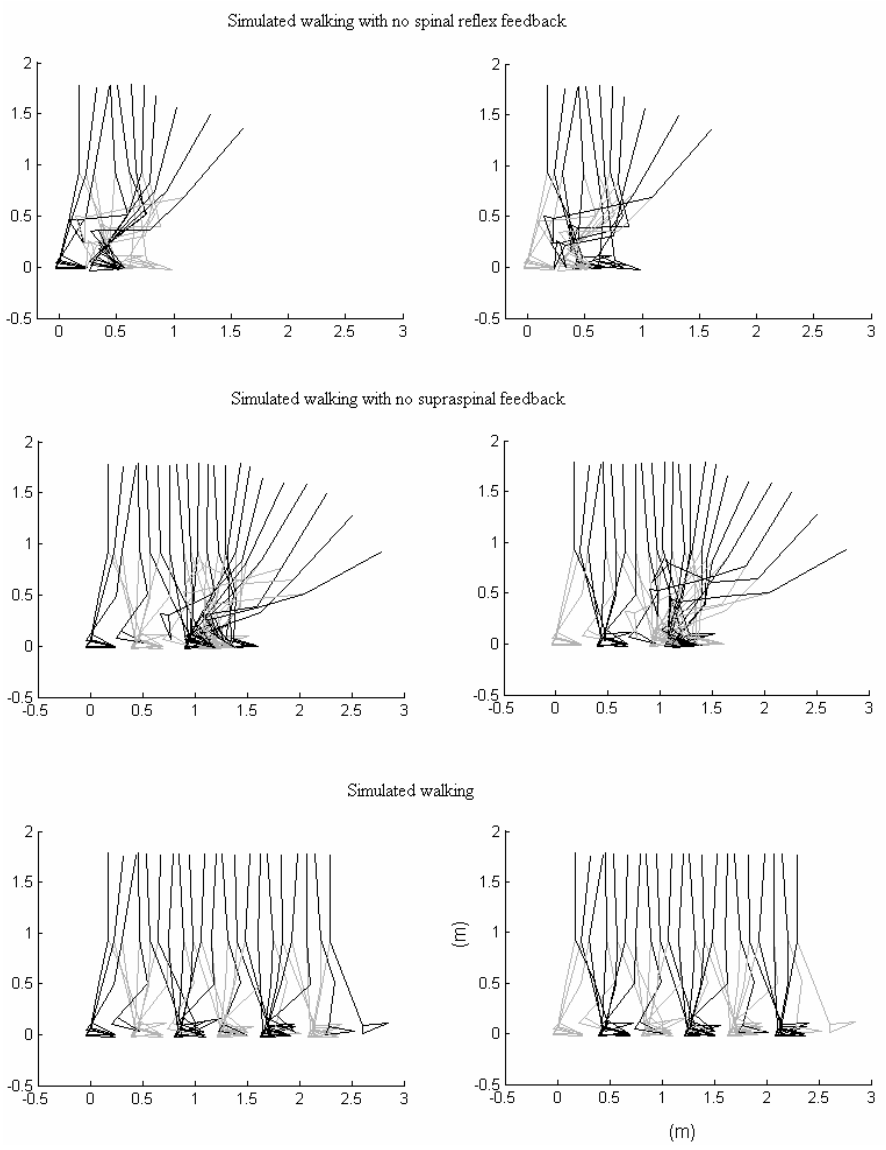
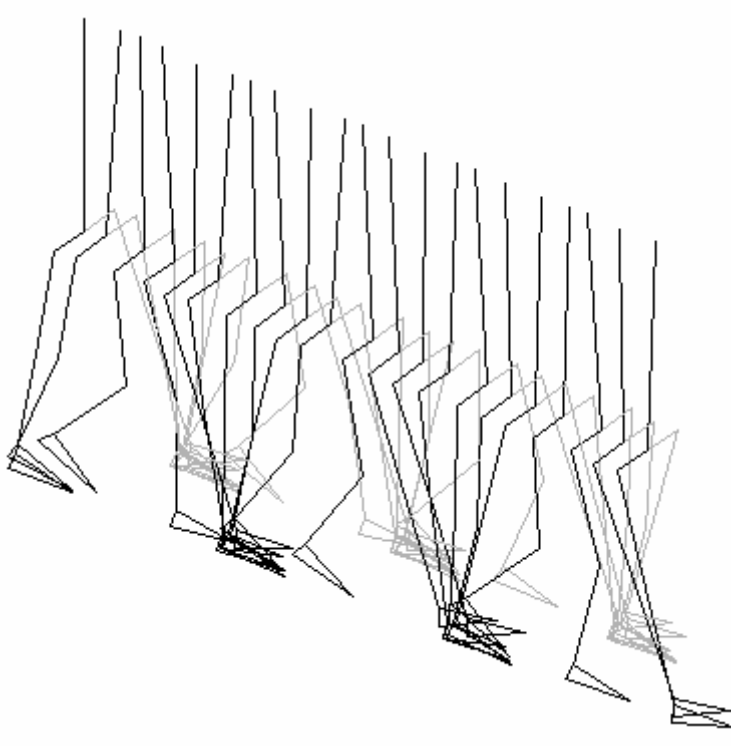



Figure 3.5. Integrated neural system for walking simulation.

The overall system for walking simulation is illustrated in figure 3.5. Principally, the neural system consists of two feedback controls and one feedforward command generator to control behavior of the body. Figure 3.6 demonstrates simulated walking tests using the proposed model. Elimination of either spinal or supraspinal neural feedback results in a fall. Unstable walking with no spinal reflex feedback is caused because the left leg swings too excessively. In comparison, walking with no supraspinal feedback failed postural balance to support the body. As for normal walking pattern, the steady state motion at speed of 1.21m/s is simulated after initial transient responses even though the model includes phase lag components such as neural delays and EC coupling. Moreover, the model does not require any detailed dynamic information like an internal model but a few rudimentary feedforward pulses.




**a.**



**b.**

Figure 3.6. a. stick plot of biped walking. (top): simulation with no spinal reflex feedback, (middle): simulation with no supraspinal feedback, and (bottom): simulation of nominal walking. Left columns highlights left leg motions, and right columns right leg motions, and b. three-dimensional illustration.

The major feature of the model is the functional decoupling over the hierarchical neural control circuits even though the functions are not perfectly separate. The NPG executes gaits, and supraspinal system controls dynamic postural balance. Segmental reflex in the level of spinal cord helps regulate interlimb movement. The supraspinal system concerns the processing of two sensed variables, i.e., the location of CM and trunk pitch angle. The control channel relevant to the location of CM plays a main role as the postural control of the lower limb, and trunk pitch angle as that of the upper limb. The decoupling scheme makes it possible to avoid the overall complicated recomputation of motor program when the body responses to environment or elicit a new behavior. A simple adjustment of the relevant local neural system may enable simply the achievement of behavioral adaptation.

## 4. EXECUTION OF CHALLENGING BIPEDAL LOCOMOTION TASKS

A particularly challenging example of an autonomous robot in an unstructured environment is a bipedal walking machine. An example task for such a system is to walk to a moving soccer ball and kick it. Stepping movement must be synchronized with ball movement so that the kick happens when the ball is close enough. More generally, such tasks require that the biped be in the right location at an acceptable time. This implies spatial and

temporal constraints for such tasks. There are also important dynamic balance constraints that limit the kinds of movements the biped may make without falling down.

If the system encounters a disturbance while performing a task, it will have to compensate in some way in order to successfully carry out the task. In the soccer example, the disturbance may cause a delay, allowing another player to kick the ball, or it may interfere with movement synchronization. For example, a trip causes disruption of synchronization between the stepping foot, and the overall forward movement of the system's CM. This synchronization must be restored in order for the task to be completed. A second example task is walking on a constrained foot path, such as stones across a brook, or on a balance beam. As with the soccer ball example, this task has spatial, temporal, and dynamic constraints, but in this case, the spatial constraints are more stringent; the biped must reach its goal using foot placements that are precisely constrained. A biped walking over blocks also constrains foot placement in a similar manner. When foot placement is constrained, the stepping pattern cannot be changed arbitrarily to compensate for a disturbance. For example, if a lateral push disturbance occurs, rather than stepping the leg out to the side, other compensating techniques, such as angular movement of the body and swing leg must be used.

In these examples, and others like them, the key challenge is to move a complex, dynamic system to the right place, at the right time, despite actuation limits, and despite disturbances. The system should be able to recover from disturbances such as slips, trips, pushes, and ground contact instability due to soft terrain, even when foot placement is constrained.

## 4.1. Locomotion Task Execution Problem Statement

A control system for tasks such as those discussed in the previous examples must be capable of guiding a robotic biped through a series of walking task goals, in the presence of disturbances. The system must accept as input a high-level specification of where it should be, and by what time, and then automatically figure out the details of how to move to accomplish these goals. If a disturbance occurs during execution of the task, the system should attempt to compensate in order to avoid a fall, and should still try to complete the task on time. If this is not possible, the system should detect this as soon as possible after the disturbance, and issue a warning to a higher-level control authority.

### *4.1.1. Specification of Task Goals*

There are two kinds of task constraints: state and temporal. We specify state constraints as bounds on key position and velocity variables that summarize the state of the biped, such as the CM, and foot placement positions. For example, to specify state constraints for the soccer example, we specify a set of constraints on foot placement locations, corresponding to a series of steps leading up to the soccer ball, as shown in figure 4.1. To ensure that the body is in an appropriate position, we additionally specify a constraint on the CM so that it is close to the soccer ball.

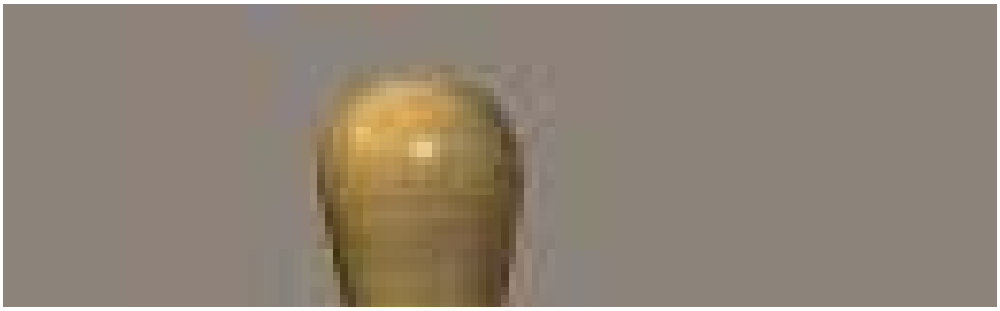


Figure 4.1. State constraints for foot placement and CM for kicking a soccer ball.

We specify temporal constraints as bounds on durations of tasks. For example, we specify a lower and upper bound on the duration allowed for the biped to reach the ball. This synchronizes movement of the biped with movement of the ball.

Note that state and temporal constraints specified for a task are distinct from state and temporal limitations inherent in the biped itself. The latter are due to actuation and dynamic limitations of the biped; they arise from the fact that the biped is a complex, articulated mechanism, where movement is achieved by applying torques to the joints, which accelerate the segments in the mechanism. Acceleration is limited by segment inertias, limitations of the joint actuators, and by the fact that the support base on the ground is limited. The state and temporal constraints specified for a task interact with, and often conflict with, the state and temporal limitations of the biped; a key challenge of the control system is to resolve this fundamental tension between task requirements and robot capabilities.

#### *4.1.2. Task Execution*

The task execution problem can be stated in the following way. Given a set of task constraints, and given a particular biped with actuation limits, generate a sequence of control actions that are within the biped's actuation limits, and that result in state trajectories that satisfy the task constraints. Thus, the generated state trajectories must pass through the goal regions at acceptable times. If it is not possible to generate such a control sequence, the system should issue a warning indicating that the task is not feasible. For example, if the soccer ball is moving too quickly or is too far away, it may not be possible to kick it.

## 4.2. Challenges

There are two key challenges to solving the task execution. First, a biped is a high-dimensional, highly-nonlinear, tightly coupled system, so computing control actions that achieve a desired state is a challenging problem. Second, a biped is under-actuated and has significant inertia, so future state evolution is coupled to current state through dynamics that limit acceleration, and the executive must consider how current state and actions may limit achievement of future desired state.

Bipeds have multiple articulated joints, and therefore, have a large number of degrees of freedom. For example, the biped model used here has 12 actuated joints, and 18 degrees of freedom, resulting in a system with 36 state variables, and 12 control input variables.

Furthermore, movement dynamics are highly nonlinear and tightly coupled, so computing control actions that achieve a desired state is a challenging problem.

The second challenge, under-actuation, is due to the fact that a biped has significant inertia, and limited ability to generate force against the ground. This results in limits on the biped's ability to accelerate its center of mass.

The principal actuation limit is the horizontal force that can be applied to the biped's CM. This is due to its high center of mass and limited base of support. Because the contact surface of the feet with the ground is limited, particularly in single support, the feet may slip or roll if inappropriate actuation forces are used. To avoid this, horizontal force exerted by the feet against the ground must be limited, but this also limits acceleration of the center of mass, and thus, the ability to control its position when there are disturbances. Thus, unlike manipulators, walking machines are under-actuated because they do not have a firm base of support, and therefore, are very sensitive to balance disturbances.

A further complicating factor is due to the inherent nature of walking tasks. Bipeds operate in a sequence of discrete modes defined by contact of the feet with the ground. At transitions between these modes, the base of support changes discontinuously. Thus, at toe-off, the base of support is instantly reduced because the biped transitions from double to single support. At heel-strike, the base of support is instantly enlarged because the biped transitions from single to double support. These discontinuous changes in support base imply discontinuous changes in actuation limits. The control system must take these discontinuous changes in actuation limits into account when generating control actions and projecting evolution of state trajectories over the future time horizon.

The support base limitations imply that walking is an inherently unstable process, where the system is constantly in a state where it is about to fall. The mode transitions due to stepping, which change the base of support, simply defer the fall; for fast walking, these modes do not have a stable equilibrium point. This is a direct consequence of the actuation constraints, and the fact that the biped has significant momentum during fast walking. This means that the biped cannot stop instantly, in the middle of a gait cycle. Rather, the system must first slow down to a slower walk, and then come to a rest in a standing position, a mode that does have a stable equilibrium point. Thus, for such walking plans, the executive must guide the system through a sequence of inherently unstable states to get to a goal state that is stable.

## 4.3. Approach and Innovations

To address the difficulty of determining the effect of control actions on biped state, we use a model-based approach, where a model of the biped is used to predict this effect. Thus, we use a *model-based executive* [Williams and Nayak, 1997; Leaute, 2005] to interpret task goals, monitor biped state, and compute joint torque inputs for the biped, as shown in figure 4.2.

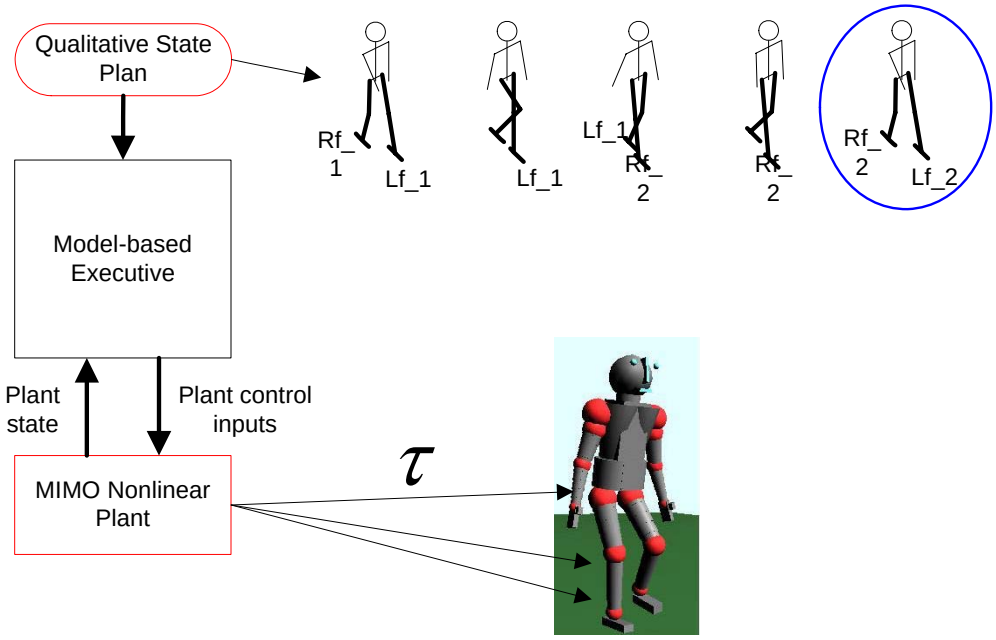



Figure 4.2. A model-based executive computes a sequence of joint torques for the biped that results in the achievement of the successive qualitative state goals.

We address the challenges described in the previous section with three key innovations. To address the first challenge (nonlinearity, high dimensionality, and tight coupling), we linearize and decouple the biped system into a set of independent, linear, single-input single-output second-order systems, resulting in an abstraction of the biped that is easier to control. We accomplish this through a novel controller called a *dynamic virtual model controller* [Hofmann, et. al., 2004]. The linearization and decoupling provided by this controller allows points of interest on the biped, such as the center of mass and the stepping foot, to be controlled directly, as if the biped were a puppet, as shown in figure 4.3.

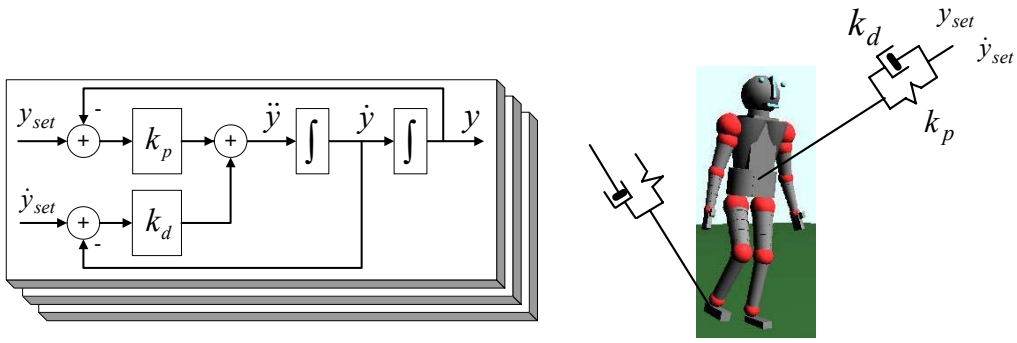



Figure 4.3. A dynamic virtual model controller provides a linear abstraction so that the reaction points move as if they were independent, linear second-order systems, controlled by the virtual elements.

To address the second challenge (actuation limits and sensitivity to balance disturbances), we perform a reachability analysis that defines sets of feasible trajectories for the state

variables and control parameters in the abstracted biped. These trajectory sets, called flow tubes, represent bundles of state trajectories that take into account dynamic limitations due to under-actuation, and also satisfy plan requirements. Additionally, our system uses a novel strategy that employs angular momentum to increase the horizontal force that can be applied to the system's center of mass, and thus, to enhance its balance controllability. This strategy is particularly useful for tasks where foot placement is constrained.

## 4.4. Control Architecture and Implementation

To achieve correct plan execution, the model-based executive must generate a control trajectory that satisfies all future goal region and temporal constraints specified in the qualitative state plan. We accomplish this by using a two-part architecture, consisting of a *hybrid task-level executive*, and a *dynamic virtual model controller* (DVMC), as shown in Fig. 4.4.

The hybrid executive controls the biped indirectly, by setting control parameters for the dynamic virtual model controller, rather than directly, by generating joint torques for the biped. Thus, it leverages the linear abstraction provided by the DVMC so that it need only consider the evolution of independent linear systems, rather than a tightly-coupled high-dimensional nonlinear system.

In order to project the feasible future evolution of the biped's state, the hybrid executive computes *flow tubes* that define valid operating regions in terms of the abstracted biped. The flow tubes represent bundles of state trajectories that satisfy plan requirements, and also take into account dynamics and actuation limitations. The flow tubes observe the discontinuous changes in actuation constraints due to ground contact events. Once the flow tubes have been computed, the hybrid executive executes the plan by adjusting control parameters in the abstracted biped in order to keep trajectories within the tubes.

Because computation of flow tubes is time consuming, and because the hybrid executive must run in real time, we perform this step off-line, as a compilation. Thus, the hybrid executive consists of two components: a *plan compiler*, and a *dispatcher*. The plan compiler outputs a *qualitative control plan*, which contains the flow tubes for all state variables in the abstracted biped.

The dispatcher monitors the state of the abstracted biped, adjusting control parameters based on the specifications in the qualitative control plan to keep trajectories in their flow tubes. The dispatcher monitors plan execution by monitoring the abstracted biped's state relative to the plan. In this way, it checks whether each trajectory is in its tube. If a disturbance occurs, the dispatcher attempts to adjust the SISO control parameter settings in order to compensate, so that the trajectory remains inside the tube, as shown in figure 4.5a. If the disturbance has pushed the trajectory outside its tube, as shown in figure 4.5b, then the dispatcher aborts, indicating to a higher-level planner that plan execution has failed.

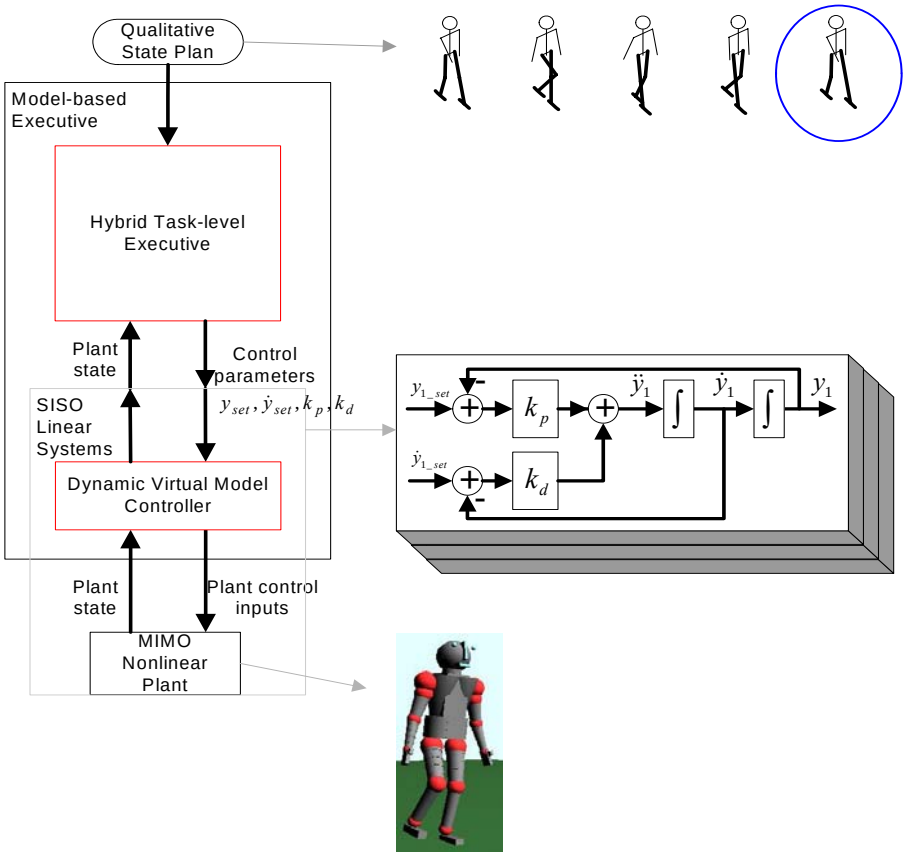
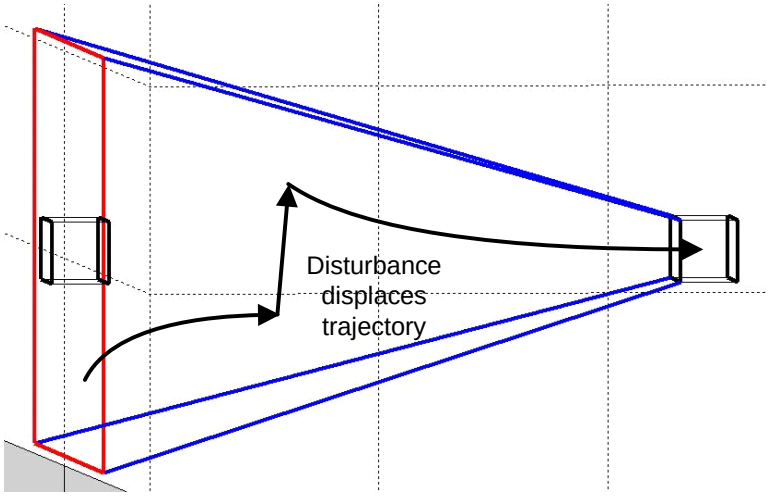
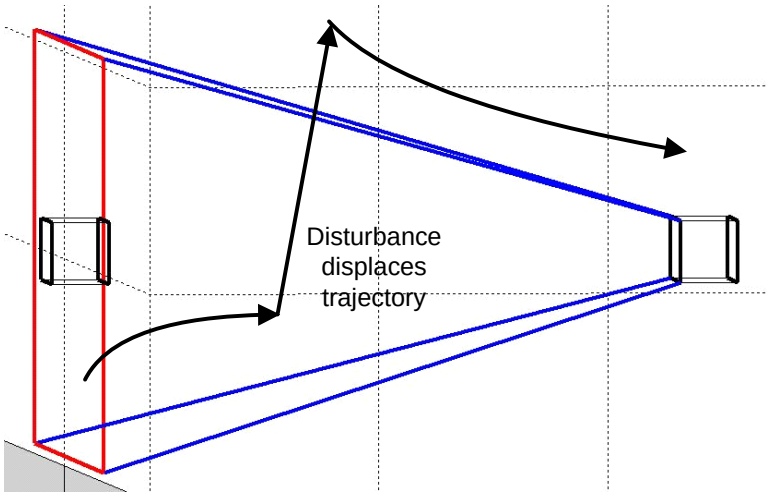



Figure 4.4. The model-based executive consists of a hybrid task-level executive and a dynamic virtual model controller. The hybrid executive controls the biped by adjusting control parameters of the linear virtual element abstraction provided by the controller. The hybrid executive sets control parameters to guide state variables through successive goal regions, while satisfying timing and balance constraints.




**a.**




**b.**

Figure 4.5.a. If a disturbance is not too large, the trajectory remains inside the tube. This means that the dispatcher will be able to adjust control parameters so that the trajectory reaches the goal at an acceptable time. b. If a disturbance is too large, it pushes the trajectory outside its tube, and the plan fails.

We next describe the qualitative state plan, the input that specifies the locomotion task to be performed. We then discuss the dynamic virtual model controller, which provides the abstracted biped. Plan compilation of the qualitative state plan and dispatching of the resulting qualitative control plan are then discussed.

### *4.4.1. Qualitative State Plan*

For most practical applications, a precise specification of state and temporal goals is not necessary. Rather, a loose specification, in terms of state space regions and temporal ranges, is preferable in that it admits a wider set of possible solutions. This may be exploited, for example, to improve optimality or to adapt to disturbances. An example state space goal is for the biped's center of mass position to be within a particular region. An example temporal goal is that this state space goal be achieved after 5 seconds, but before 6.

Reaching a goal location may require the biped to take a sequence of steps. Such steps represent transitions through a sequence of fundamentally different states, defined by which feet are in contact with the ground. Thus, a stepping sequence consists of alternating between double support phases, where both feet are on the ground, and single support phases, where one foot (the stance foot) is in contact with the ground, and the other foot (the swing foot) is taking the step. These phases represent qualitatively different system states, with correspondingly different behaviors.

We define a qualitative state as an abstract constraint on desired position, velocity, and temporal behavior of the biped. A qualitative state indicates which feet are on the ground, and includes constraints on foot position. It may also include state constraints on quantities like the biped's center of mass, and temporal constraints specifying time ranges by which the state space goals must be achieved. Thus, a qualitative state is a partial specification of desired behavior for a portion of a walking gait cycle. A sequence of qualitative states represents intermediate goals that lead to the final overall task goal. Such a sequence forms a qualitative state plan.

For example, a plan for a biped walking cycle is a sequence of qualitative states representing single and double support gait phases. Such a plan is shown in figure 4.6. In this plan, the first qualitative state represents double support with the left foot in front, the second, left single support, the third, double support with the right foot in front, and the fourth, right single support. The fifth qualitative state repeats the first, but is one gait cycle forward.

A qualitative state plan has a set of *activities* representing constraints on desired state evolution of workspace state variables. Activities are indicated by horizontal arrows in figure 4.6, and are arranged in rows corresponding to their associated state variables. In figure 4.6, the activities *left foot ground 1* and *left foot step 1* are for the left foot, *right foot ground 1*, *right foot step 1*, and *right foot ground 2* are for the right foot, and CM1 – 4 are for the center of mass.

Every activity starts and ends with an *event*, represented by a circle in figure 4.6. Events in this plan relate to behavior of the stepping foot. Thus, a *toe-off* event represents the stepping foot lifting off the ground, and a *heel-strike* event represents the stepping foot landing on the ground. Events define the boundaries of qualitative states. Thus, the right toe-off event defines the end of the first qualitative state (double support), and the beginning of the second qualitative state (left single support).

The qualitative state plan in figure 4.6 has a temporal constraint between the start and finish events. This constraint specifies a lower and upper bound, $[lb, ub]$, on the time between these events. Such temporal constraints are useful for specifying bounds on tasks consisting of sequences of qualitative states. The temporal constraint in figure 4.6 is a constraint on the time to complete the gait cycle, and thus, can be used to specify walking speed.

In addition to temporal constraints, qualitative state plans include state constraints. These are associated with activities, and are specified as rectangles in position/velocity state space. Such rectangles can be used to specify required initial and goal regions. In figure 4.6, the goal region constraint $CM \in R_1$ represents the requirement that the CM trajectory must be in region $R_1$ for the CM movement activity to finish successfully.

In addition to initial and goal regions, an activity may also have operating region constraints that specify valid regions in state-space where the trajectory must be over the entire duration of the activity. These are of the form $g(y_i, \dot{y}_i) \leq 0$, and they may be linear or nonlinear. Such constraints are used to express actuation limits. For example, CM movement in the plan of figure 4.6 is represented by four separate activities: CM1 – CM4. Only CM4 has a goal region. However, each of these activities have different operating regions. This is due to the discontinuous changes in the base of support resulting from the foot contact events; the base of support in double support is very different from the one in single support. Thus, for CM1, the base of support is the polygon defined by r1 and l1 in figure 4.6. For CM2, it is the polygon defined by l1 only.

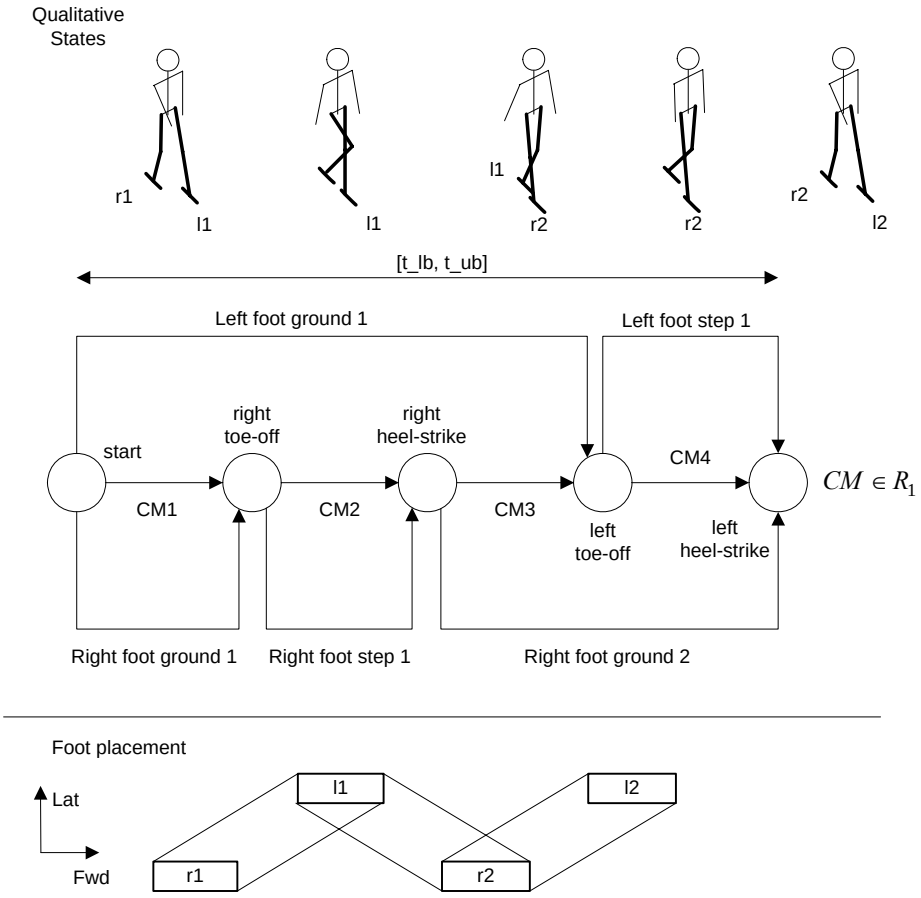



Figure 4.6. Example qualitative state plan for walking gait cycle. Circles represent events, and horizontal arrows between events represent activities. Activities may have associated state space constraints, such as the goal region constraint $CM \in R_1$, which specifies a goal for CM position and velocity. Foot placement constraints are indicated at the bottom; for example, rectangle r1 represents constraints on the first right foot.

The base of support has a strong effect on the maximum force that can be exerted on the CM. This is why these operation constraints must be defined; they represent actuation limits for the CM activities. This is why CM movement in the plan of figure 4.6 is represented by four activities instead of only one.

### *4.4.2. Dynamic Virtual Model Controller*

The *dynamic virtual model controller* (DVMC), the component of the model-based executive that interacts directly with the biped. The DVMC simplifies the job of the hybrid executive component of the model-based executive by providing the linear abstraction.

The DVMC provides the task executive with a simple way to control the biped's forward and lateral CM position. This allows the executive to maintain the system's balance, and to move the biped forward during walking, by specifying desired CM movement. Additionally, the DVMC provides the hybrid executive with a simple way to control movement of the stepping foot, and to maintain the upright posture of the torso.

The abstraction provided by the DVMC allows the hybrid executive to control the biped like a puppet, using virtual linear spring-damper elements. These virtual elements are attached at key reaction points, like the center of mass, and the stepping foot, as shown in figure 4.3. The virtual elements don't actually exist. However, the biped will move as if they did. In particular, the hybrid executive can assume that the motion of a reaction point will be that of a linear system with spring and damping parameters corresponding to the virtual elements. This greatly simplifies the planning and control functions of the hybrid executive because it does not have to be concerned with the nonlinear dynamics of the actual biped, or with computing joint actuator torques. The hybrid executive lets the dynamic virtual model controller worry about these details.

Our use of virtual elements is similar, in concept, to the one used in a virtual model controller [Pratt et al., 1997]. An important difference is that our dynamic virtual model controller takes dynamics into account, while a virtual model controller does not. A virtual model controller uses a Jacobian transformation to translate the desired forces at the reaction points, specified by the virtual elements, into joint torques that produce these forces. This works well for static or slow-moving mechanisms, but can break down as movements become faster because the controller does not take into account the dynamics of the system. Therefore, movement of the reaction point is not necessarily in line with the desired virtual force. In contrast, our dynamic virtual model controller uses a dynamic model to account for the biped's dynamics. This results in a linear system, where reaction points move as if they were simple linear second order systems, controlled by the virtual elements, as shown in figure 4.3.

The DVMC performs four key functions. First, it uses a model-based input-output linearization algorithm [Slotine and Li, 1991] to linearize and decouple the plant. Second, the controller performs a geometric transform from joint space to workspace coordinates in order to make state variables relevant to balance control, such as center of mass position, directly controllable, as the state variables of a simple linear system. Third, the controller prioritizes goals in order to accommodate actuation constraints. Such actuation constraints may cause the overall system to become over-constrained. To address this problem, our controller automatically sacrifices lower-priority goals when the system becomes over-constrained in this way. For example, the system may temporarily sacrifice goals of maintaining upright posture in order to achieve balance goals. Fourth, the DVMC addresses the problem of model inaccuracy by incorporating a sliding control algorithm [Slotine and Li, 1991].

The linearization and goal prioritization approach of our controller is similar, in concept, to the recently developed whole-body control algorithm [Khatib et al., 2004]. However, the whole-body controller relies heavily on an accurate model; it does not account for model inaccuracy. Our controller accounts for this inaccuracy using the sliding control approach. Recently, the whole-body control approach has been used to regulate a biped's linear and angular momentum [Kajita et al., 2001; Yokoi et al., 2001; Sugihara et al., 2002; Nishiwaki et al., 2002]. A key point of difference is that our model is able to purposely sacrifice angular momentum control goals in order to achieve linear control goals when both cannot be met.

#### 4.4.2.1. Linearization and State Transformation

A geometric transform, $\mathbf{h}$, is used to convert from the joint state to the workspace state representation, according to

$$\left[\mathbf{y}^T, \dot{\mathbf{y}}^T\right]^T = \mathbf{h}\left[\mathbf{x}^T, \dot{\mathbf{x}}^T\right]^T \tag{4.1}$$

where $\left[\mathbf{x}^T, \dot{\mathbf{x}}^T\right]$ is the joint state vector, and $\left[\mathbf{y}^T, \dot{\mathbf{y}}^T\right]$ is the workspace state vector. Elements of $\mathbf{x}$ include joint angle positions, such as left knee joint angle, and elements of $\mathbf{y}$ include forward and lateral CM position. The controller uses a feedback linearizing transformation to convert desired workspace variable accelerations, $\ddot{\mathbf{y}}$, into corresponding joint torques, $\boldsymbol{\tau}$, as shown in figure 4.7. Application of these torques results in a new joint state, $\left[\mathbf{x}^T, \dot{\mathbf{x}}^T\right]$. The multivariable controller then uses the transformation, $\mathbf{h}$, to convert from joint to workspace state.

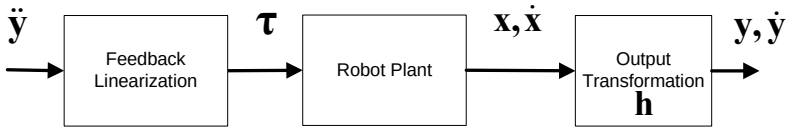



Figure 4.7. Feedback linearization and output transformation.

If we draw a black box around the series of transforms in figure 4.7, the *multiple-input multiple-output* (MIMO) nonlinear plant appears to be a set of decoupled SISO linear 2nd-order systems, as shown in figure 4.8. Each element, $\mathbf{y}_i$ of position vector $\mathbf{y}$, can be viewed as the output of one of the SISO systems, with the corresponding acceleration element, $\ddot{\mathbf{y}}_i$, being the input. Each SISO system can be controlled by a simple linear control law, such as the proportional-differential (PD) law shown in figure 4.8. The set of SISO systems, with associated linear control laws, forms the linear virtual element abstraction. The solution trajectory for each SISO system is defined by a linear second-order differential equation, so the trajectory value at any time can be computed analytically.

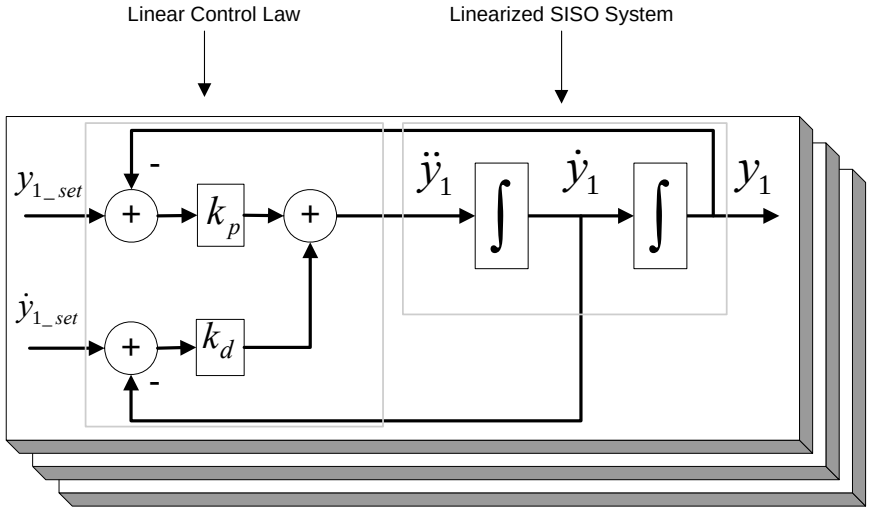



Figure 4.8. Linear virtual element abstraction consisting of a set of SISO systems with associated linear control laws.

Linearization of the plant dynamics is accomplished using the standard dynamics equation for an articulated body [Craig, 1989]:

$$\mathbf{H}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} \tag{4.2}$$

Thus, for a particular joint state $\left[\mathbf{q}^T, \dot{\mathbf{q}}^T\right]$, this gives joint torques as a linear function of joint accelerations:

$$\mathbf{H}_1\ddot{\mathbf{q}} + \mathbf{G}_1 = \boldsymbol{\tau} \tag{4.3}$$

Plant outputs are expressed as a nonlinear function of plant joint state, using Eq. (4.1). A linear relation between workspace accelerations, $\ddot{\mathbf{y}}$, and joint accelerations, $\ddot{\mathbf{q}}$, is obtained by computing the second derivative of $\mathbf{y}$.

$$\ddot{\mathbf{y}} = \frac{\partial^2 \mathbf{h}(\mathbf{q}, \dot{\mathbf{q}})}{\partial(\mathbf{q}, \dot{\mathbf{q}})^2} \tag{4.4}$$

resulting in an equation of the form

$$\ddot{\mathbf{y}} = \boldsymbol{\Psi}\ddot{\mathbf{q}} + \boldsymbol{\Psi}_{const} \tag{4.5}$$

It is assumed that this linear system can be solved for $\ddot{\mathbf{q}}$ given $\ddot{\mathbf{y}}$, at least for the region of state space in which the controller is operating.

Now, suppose we add a linear controller that computes a control input of the form

$$\ddot{\mathbf{y}}_{des} = \mathbf{f}_{controller}(\mathbf{y}_{des}, \mathbf{y}, \dot{\mathbf{y}}_{des}, \dot{\mathbf{y}}) \tag{4.6}$$

Eq. 4.5 is then used to convert to desired joint accelerations:

$$\ddot{\mathbf{q}}_{des} = \boldsymbol{\Psi}^{-1}(\ddot{\mathbf{y}}_{des} - \boldsymbol{\Psi}_{const}) \tag{4.7}$$

These are then substituted into Eq. (4.2), with $\ddot{\mathbf{q}} = \ddot{\mathbf{q}}_{des}$, in order to get the desired control torques

The result is a a two-stage linearization, where setpoints are specified in terms of the desired output variables, as shown in the figure 4.9. In this two-stage linearization, output variable accelerations are converted to joint accelerations by the first linearization, and then joint accelerations are converted to torques by the second, inverse dynamics, linearization.

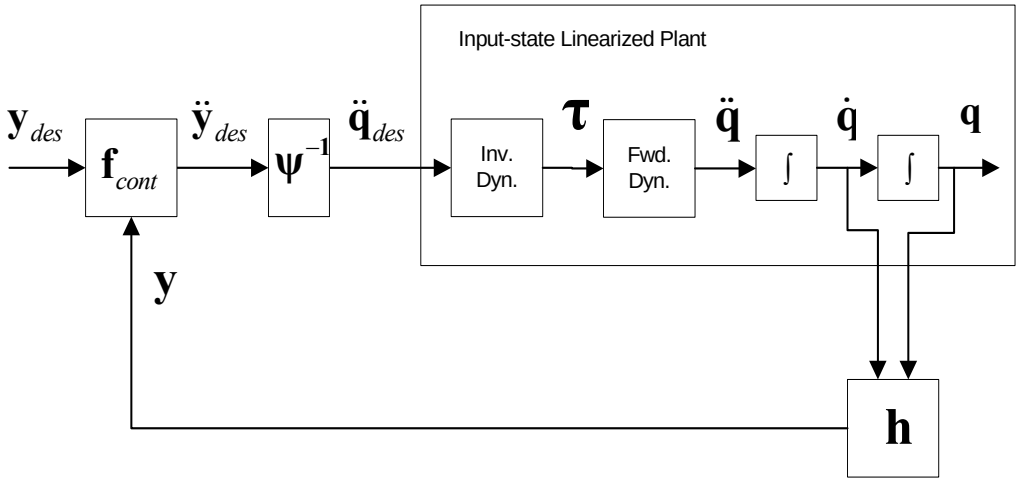



Figure 4.9. Two-stage Linearization.

Further details of the derivation of the feedback linearization and output transformation can be found in [Hofmann, 2005] and [Hofmann et al., 2004].

#### 4.4.2.2.Goal Prioritization

We chose the following outputs to be elements of the $\mathbf{y}$ vector.

- forward CM position
- lateral CM position
- stance knee joint angle
- torso roll angle
- torso pitch angle
- torso yaw angle
- forward swing foot position
- lateral swing foot position
- swing knee joint angle
- swing foot roll
- swing foot pitch
- swing hip joint yaw angle

The forward and lateral CM positions are important variables to control for balancing, so it makes sense to include these in the output vector. Similarly, swing foot placement determines the shape of the support polygon, and is therefore also crucial for balance control. Thus, it makes sense to include swing foot forward and lateral position in the output vector. It is desirable to maintain an upright torso position, so torso orientation should also be included. Note that vertical CM and swing foot position are not included in the output vector, in order to avoid singularities that may occur with these quantities. Instead, stance and swing knee joint angles are controlled, and vertical CM and swing foot positions emerge from these.

With this choice of outputs, the system given by Eq. (4.5) is square, because there are 12 inputs (the torques to the 12 joints), and there are 12 outputs. As long as $\boldsymbol{\Psi}$ is invertible, the

system will appear to be completely linearized and decoupled to the outer controller $\mathbf{f}_{cont}$. Thus, the outer controller can specify elements of $\ddot{\mathbf{y}}_{des}$ as if they were independent, and a corresponding $\ddot{\mathbf{q}}_{des}$ can always be computed.

Next, consider what happens when bounds on plant inputs due to saturation limits are added. If $\mathbf{f}_{cont}$ does not take these bounds into consideration, it could generate values for $\ddot{\mathbf{y}}_{des}$ that cause the bounds to be violated.

To avoid this type of infeasibility, slack variables, $\ddot{\mathbf{y}}_{slack}$, are introduced for each element of $\ddot{\mathbf{y}}_{des}$, so that the new controller output, $\ddot{\mathbf{y}}_{cont\_out}$, is

$$\ddot{\mathbf{y}}_{des} = \ddot{\mathbf{y}}_{cont\_out} + \ddot{\mathbf{y}}_{slack} \tag{4.8}$$

Use of these slack variables provides flexibility in that $\ddot{\mathbf{y}}_{des}$ conforms to the $\mathbf{f}_{cont}$ linear control law, without regard to the actuation bounds, while $\ddot{\mathbf{y}}_{cont\_out}$, the true output of the controller, does obey actuation bounds. The goal of the overall control system is then to minimize $\ddot{\mathbf{y}}_{slack}$, taking into account the relative importance of each element.

This minimization is accomplished by formulating the control problem as a quadratic program (QP), and then using a QP optimizer to solve it. The relative importance of the slack variables is expressed in the cost function for the QP. Slack variables associated with important outputs are given higher cost than slack variables for less important outputs. This causes the optimizer to prioritize goals by minimizing the slack variables for the most important outputs first, and therefore, setting $\ddot{\mathbf{y}}_{cont\_out}$ to be as close as possible to $\ddot{\mathbf{y}}_{des}$, in Eq. (4.8), for these outputs. For example, slack variables associated with the CM position output are given higher cost than those associated with torso orientation.

Details of the formulation are provided in [Hofmann, 2005; Hofmann et al., 2004].

The linearization described previously can be problematic for real plants because it assumes a perfect plant model. The sliding control algorithm, [Slotine and Li], addresses this problem with additional corrective control terms used to compensate for model inaccuracy. Details of our incorporation of this algorithm are provided in [Hofmann, 2005; Hofmann et al., 2004].

### *4.4.3. Qualitative Control Plan and Flow Tubes*

A QCP augments the input QSP by adding flow tubes which represent feasible trajectory sets [Bradley and Zhao, 1993; Frazzoli, 2001]. Flow tubes are associated with QSP activities; they represent feasible trajectories that result in successful execution of the associated activities. Thus, a flow tube depends on the activity's constraints, and the dynamics of the activity's SISO system.

An example flow tube is shown in figure 4.10a. A flow tube can be characterized as a set of cross-sectional regions in position-velocity phase space, one for each time in the interval $\lfloor t_0, t_g \rfloor$. Thus, such a cross section, $r_{cs}$, is a function of the flow tube, and of time: $r_{cs} = CS(Y,t): t_0 \le t \le t_g$. Figure 4.10b depicts cross sections for times $t_0$, $t_1$, and $t_g$.

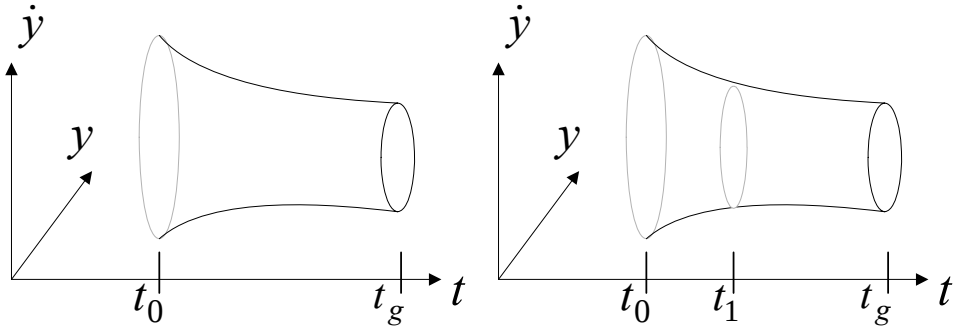



Figure 4.10.a. Example flow tube over interval $\lfloor t_0, t_g \rfloor$; b. cross sections at $t_0$, $t_1$, and $t_g$.

Next, consider the set, $R_{cs}$, of all cross sections in an interval $[t_0, t_1]$, where $t_0 \le t_1 \le t_g$. We use this set to investigate conditions under which the associated activity can be executed successfully for any start time in the interval $[t_0, t_1]$. If an allowed initial region, $r_1$, is a subset of every cross section in $R_{cs}$, then the duration of the activity is controllable over the interval $[l, u]$, where $l = t_g - t_1$, and $u = t_g - t_0$. Conversely, each cross section of $Y$ of which $r_1$ is a subset corresponds to a controllable duration of the activity.

In this way, if the initial state of the system is in $r_1$, the controllable duration of the activity is known. This controllable duration is the temporal constraint due to the plant dynamics. It is added to the temporal constraints specified explicitly in the QSP.

### *4.4.4. Plan Compilation*

Flow tubes have a complex geometry. Therefore, any tractable flow tube representation will be an approximation of the feasible set. In order to ensure that any trajectory chosen by the dispatcher leads to plan execution success, we require our flow tube representation to include only feasible trajectories; the representation may include a subset of all feasible trajectories, but not a superset [Kurzhanski and Varaiya, 1999].

Our flow tube approximation uses polyhedral cross sections at discrete time intervals [Vestal, 2001]. The time interval chosen matches the control increment of the dispatcher. Therefore, the dispatcher will always be able to access flow tube cross sections for exactly the correct time. Figure 4.11 shows a flow tube cross-sectional region in position-velocity phase space, and its polyhedral approximation. Note that the approximation is a subset of the true region; the approximation does not include points in state-space that do not belong to feasible trajectories.

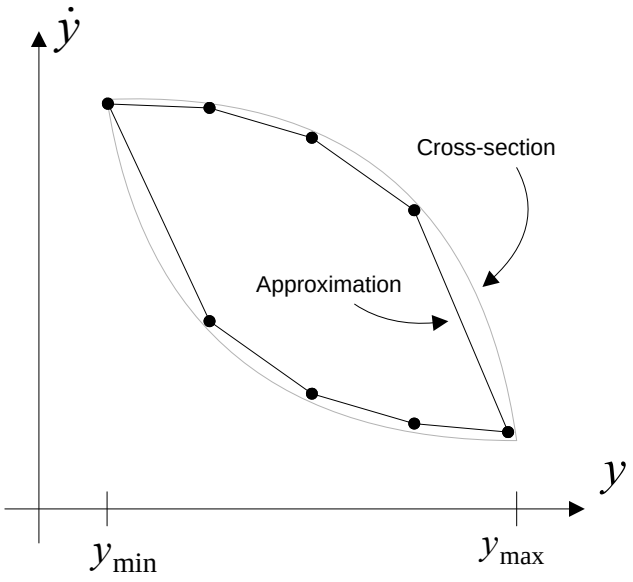



Figure 4.11. Flow tube cross section and approximation.

In order to generate the polyhedral cross-sections, the plan compiler performs a *reachability analysis* that, for every vertex position, computes extreme corresponding velocities such that the resulting polygon contains only feasible trajectory points for the time associated with the cross section. We accomplish this reachability analysis by formulating constraints on cross section vertices as a linear program (LP).

The LP formulation is based on the analytical solution of a 2nd-order linear differential equation, with analytic solution:

$$y = e^{\alpha t}\left(K_1 \cos\ \beta t + iK_2 \sin\ \beta t\right) + u / c$$

$$\dot{y} = e^{\alpha t}\left(\beta\left(-K_1 \sin\ \beta t + iK_2 \cos\ \beta t\right) + \alpha\left(K_1 \cos\ \beta t + iK_2 \sin\ \beta t\right)\right)$$

where

$$K_1 = y(0) - u / c,\ K_2 = (i / \beta)(\alpha K_1 -$$

$$\alpha = -k_d / 2, \beta = \left(-i\sqrt{{k_d}^2 - 4k_p}\right) / 2, u = k_p y_{set} + k_d \dot{y}_{set}$$

This is the solution for an SISO system in figure 4.3. If we set the time, $t$, in Eq. (4.9) to a particular duration, $d_i$, corresponding to a particular cross section of interest, and if we fix gains $k_p$ and $k_d$, then Eq. (4.9) can be expressed as

$$y = f_1(y(0), \dot{y}(0), y_{set}, \dot{y}_{set})$$
$$\dot{y} = f_2(y(0), \dot{y}(0), y_{set}, \dot{y}_{set}) \quad (4.10)$$

where $f_1$ and $f_2$ are linear for a particular setting of $t$, $k_p$, and $k_d$. Eq. (4.10) forms a set of equality constraints in the LP formulation. We also include a set of inequality constraints of the form

$$y_{set\_\min} \le y_{set} \le y_{set\_m}$$
$$\dot{y}_{set\_\min} \le \dot{y}_{set} \le \dot{y}_{set\_m}$$

to represent the actuation limits, specified for the activity in the QSP. Further, we use a set of equality constraints to express

$$\langle y, \dot{y} \rangle \in R_{goal} \quad (4.12)$$

to ensure that state at the end of duration $d_i$ is in the goal region, specified for the activity in the QSP. The formulation of (4.12) as a set of linear inequalities is straightforward because $R_{goal}$ is required to be convex.

To compute a cross section for a particular $R_{goal}$ and $d_i$, the plan compiler uses the formulation described by Eqs. (4.10 - 4.12), and sets the LP cost function to minimize $y(0)$. Solving this formulation yields the minimum initial position, $y_{\min}$, shown in figure 4.11. Repeating this process with the cost function set to maximize $y(0)$ yields the maximum initial position, $y_{\max}$. The compiler then establishes vertex positions at regular increments between $y_{\min}$ and $y_{\max}$. For each such vertex position, the compiler solves the LP formulation with the cost function set to first minimize, and then maximize, $\dot{y}(0)$, in order to find the minimum and maximum velocities for that position. This results in a set of vertices in position-velocity state space, which form the polyhedral approximation, as shown in figure 4.11.

The compiler computes cross section approximations for every $d_i$ in the temporal range $[l, u]$, where this range is given for each activity by the minimum dispatchable graph. This set of cross sections approximates a flow tube, such as the one shown in figure 4.10.

### *4.4.5. Dispatcher*

In order to execute a QCP, the dispatcher must successfully execute each activity in the QCP. The dispatcher accomplishes this by setting control parameters for each activity such that the associated trajectory reaches the activity's goal region at an acceptable time.

In order to execute an activity, the dispatcher performs three key functions: initialization, monitoring, and transition. Initialization is performed at the start of an activity's execution, monitoring is performed continuously during the activity's execution, and transition is performed at the finish of the activity's execution.

For initialization, assuming that all trajectories begin in the flow tube of their activity, the dispatcher chooses a goal duration for the control activity that is consistent with the activity's temporal constraints, and sets control parameters such that the state trajectory is predicted to be in the activity's goal region after the goal duration. The initialization function formulates a small *quadratic program* (QP) and solves it in order to determine these control parameters. This formulation is given in figure 4.12. Key to this formulation's simplicity is the fact that the analytic solution of Eq. (4.10) (functions $f_1$ and $f_2$) is used to predict the future state of the SISO system associated with the activity, and the fact that the formulation is guaranteed to produce a feasible solution, because the trajectory is within its flow tube. Further, presence of the trajectory in the flow tube guarantees that there exists a set of feasible control settings for all remaining activities in the plan, if there are no further trajectory disturbances.

FormulateControlQP( $R_{goal}, y_{curr}, \dot{y}_{curr}, t_s, t_f$ )

Parameters to optimize: $y_{pred}$, $\dot{y}_{pred}$, $y_{set}, \dot{y}_{set}$
Equality constraints: Eq. (3)
Inequality constraints: Eqs. (4, 5)
(Eq. (5) requires that trajectory prediction be within goal region)
Cost function

$$y_{goal} = \left(y_{\min}\left(R_{goal}\right) + y_{\max}\left(R_{goal}\right)\right)/2$$
$$\dot{y}_{goal} = \left(\dot{y}_{\min}\left(R_{goal}\right) + \dot{y}_{\max}\left(R_{goal}\right)\right)/2$$
$$cost = \left(y_{goal} - y_{pred}\right)^2 + \left(\dot{y}_{goal} - \dot{y}_{pred}\right)^2$$

Figure 4.12. Dispatcher QP formulation.

After initializing an activity, the dispatcher begins monitoring execution of that activity. To monitor execution, the dispatcher continually checks whether the state trajectory remains in its flow tube. If this is not the case, then plan execution has failed, and the dispatcher aborts to a higher-level control authority. Such a control authority might issue a new plan in response to such an abort. For example, the biped trying to kick the soccer ball may give up on this goal if it is no longer feasible.

If the state trajectory is in its flow tube, the dispatcher checks whether it is on track to be in the goal region at the end of the goal duration. This check is accomplished by evaluating Eq. (4.10) for the current state, and checking whether the predicted state is within the goal region. If this is not the case, the dispatcher corrects this situation by adjusting control parameters using the QP formulation of figure 4.12. Note that because the state trajectory is in its flow tube at this point, such a correction will always be possible.

As part of the monitoring function, the dispatcher also continually checks whether an activity's completion conditions are satisfied. Thus, it checks whether the state trajectory is in the activity's goal region, and whether the state trajectories of other activities whose completion must be synchronized are in their activity's goal regions. If all completion conditions for a control activity are satisfied, the dispatcher switches to the transition function.

If the activity being executed has a successor, the transition function invokes the initialization function for this successor. When all activities in the QCP have been executed successfully, execution terminates.

## 4.5. Results

### *4.5.1. Biped Simulation*

The controller was tested using a high-fidelity biped simulation, serving as the plant to be controlled. This simulated biped, shown in figure 4.6, is three-dimensional with 18 degrees of freedom (6 representing the position and orientation of the trunk, and 12 corresponding to the leg joints, which can exert torques). Each leg was modeled with a ball-and socket hip joint (3 degrees of freedom), a pin knee joint (one degree of freedom), and a saddle-type ankle joint (two degrees of freedom). The upper body (head, arms and torso), upper leg and lower leg were modeled with cylindrical shapes, and the feet were modeled with rectangular blocks.

The total mass was divided among the segments according to morphological data from the literature [Clauser et al., 1969; Brown, 1987]. The dimensions of each model segment were obtained by considering morphological data that describe average human proportions [Tilley and Dreyfuss, 1993; Winters, 1990]. Ground contact force was modeled using a nonlinear spring-damper system at four points per stance leg, located at each corner of the rectangular foot. Further details of this simulated biped are given in [Hofmann, 2005].

### *4.5.2. Balance Tests*

The ability to balance on one leg is an important prerequisite for walking, expecially when foot placement is constrained. Therefore, a series of experiments was performed to evaluate the DVMC's ability to stabilize the biped in single-support mode, that is, standing on one leg. Thus, these tests exercise the DVMC component of the execution system, independently of the task executive.

Balance recovery was tested by initializing the simulated biped in a motionless position, but with the horizontal position of the center of mass (CM) outside the support polygon, defined by the stance foot. For such an initial condition, stance ankle torque alone is insufficient for restoring balance. The maximum stance ankle torque that can be exerted without having the foot roll places the ZMP point at the edge of the support polygon, but not beyond it. Since the CM is beyond this point, this is insufficient for generating an appropriate corrective horizontal component of the ground reaction force, as explained previously. The biped is sufficiently out of balance that it becomes necessary to perform dynamic movement of non-contact segments in order to generate spin torque about the CM. This augments the horizontal ground reaction force provided by the stance ankle torque, by moving the CMP outside the edge of the support polygon. This action can help the system restore balance, by bringing the horizontal position of the CM back to the center of the support polygon, but it also causes a disturbance in the angular stability (upright posture) of the biped. The controller, therefore, must judiciously sacrifice angular stability temporarily, in order to bring the CM back under control, after which, it corrects for the angular disturbance.

The initial condition used here for testing results in an instability that is similar to the one that occurs when the system is pushed near its CM. Thus, it is a good indicator of how the system will perform when disturbed in this way.

Figure 4.13 shows the system's response to a lateral disturbance, that is, where the lateral position of the CM starts beyond the left-most limit of the system's support polygon, while standing on level ground. The counter-clockwise rotation of the upper body and right (non-stance) leg results in spin angular momentum about the CM. By conservation of angular momentum, this results in an orbital angular momentum of the CM about the support point,

which augments the angular momentum produced by stance ankle torque. This pushes the CM toward the system's right.

Figure 4.14 shows the ZMP (in blue), and the CMP (in red) during this maneuver. The left-most limit of the support base is at 0.05 meters. The ZMP stays within this limit, so that the foot does not roll, but the CMP starts outside of it. The angular movement of the upper body and non-stance leg provides enough equivalent horizontal ground reaction force to bring the horizontal position of the CM back to the center of the support base. Note that after about 0.8 seconds, the ZMP is no longer pegged at the limit. This is an indication that the CM is under control, and that the controller can turn its attention to correcting the angular disturbance.

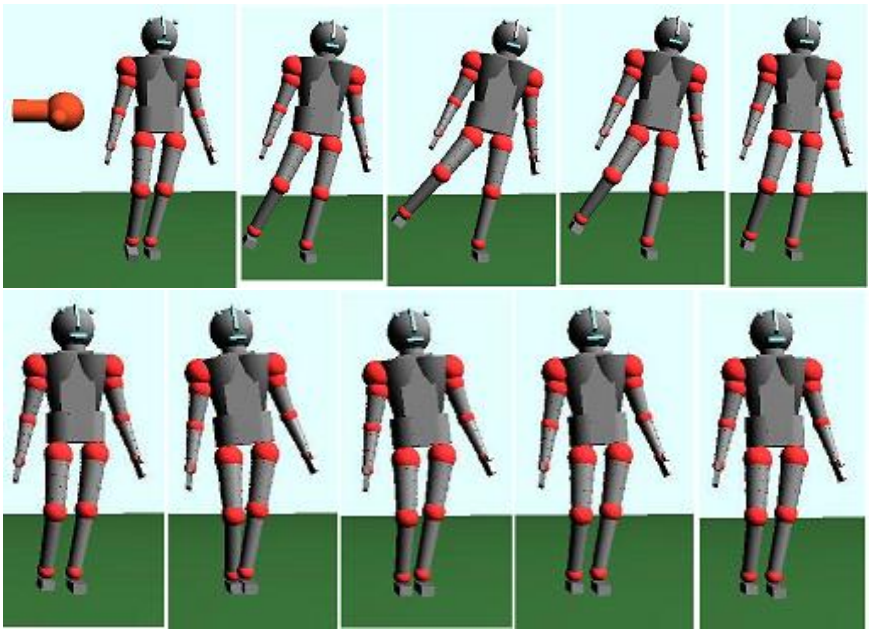


Figure 4.13. Motion sequence of biped for lateral disturbance while standing on left leg.

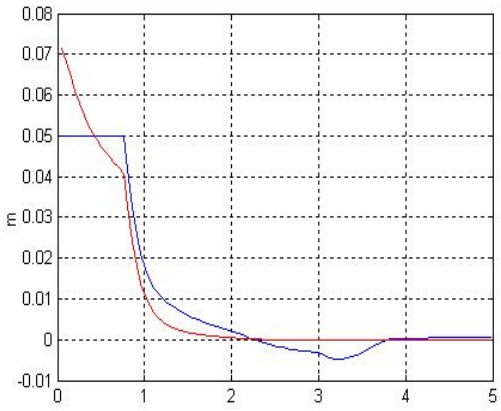



Figure 4.14. Lateral (leftward) direction ZMP in blue, CMP in red.

### *4.5.3. Kicking a Moving Soccer Ball*

Consider the previously described example task of walking to a moving soccer ball and kicking it. Such a task exercises the execution system's ability to synchronize movement of the biped with an externally applied temporal constraint; the biped must be in the right place at the right time to kick the ball.

Figure 4.15 shows the result of a test execution of this type of task. The QSP is similar to that for walking, except that there is a kick at the end. Figure 4.16 shows the corresponding flow tubes for forward and lateral CM movement, and the state trajectories within these tubes. Figure 4.17 shows initial cross sections for all flow tube sets for the task shown in figure 4.15. If the initial state of the biped is in the controllable initial region, then the task can be achieved in any duration ranging from 0.8 to 1.2 seconds. Such controllability can be used to adjust biped speed depending on the speed of the moving soccer ball, so that the biped arrives at the ball at the right time. If the initial state of the biped is in a subset of the cross sections, then this temporal controllability is more restricted; it is restricted to a tighter range than [0.8, 1.2], and synchronization with the moving soccer ball may be more difficult.

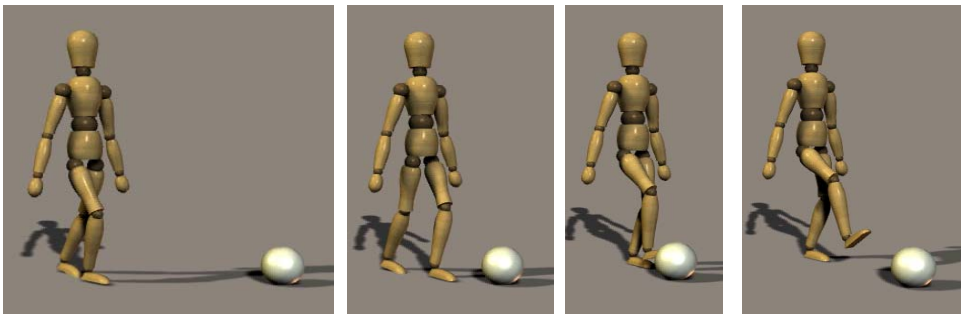
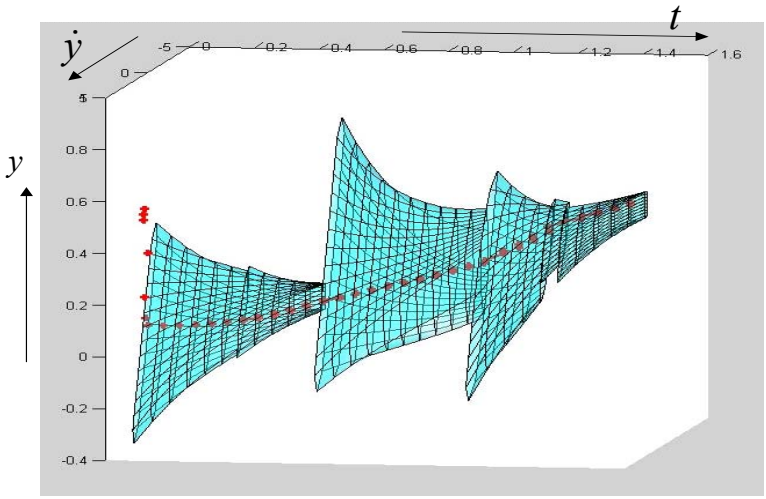
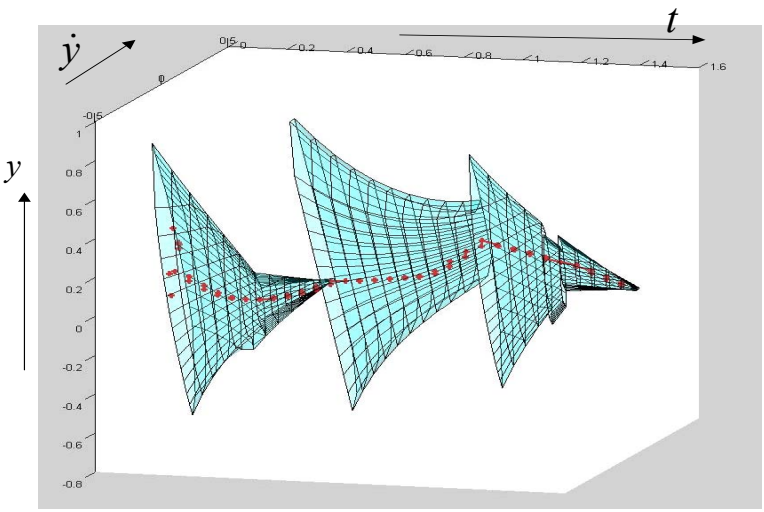


Figure 4.15. Walking to a moving soccer ball and kicking it.

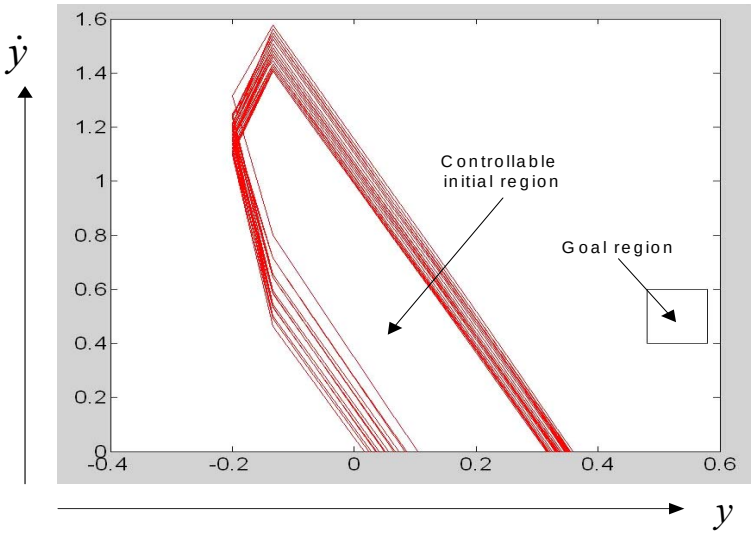



**a.**

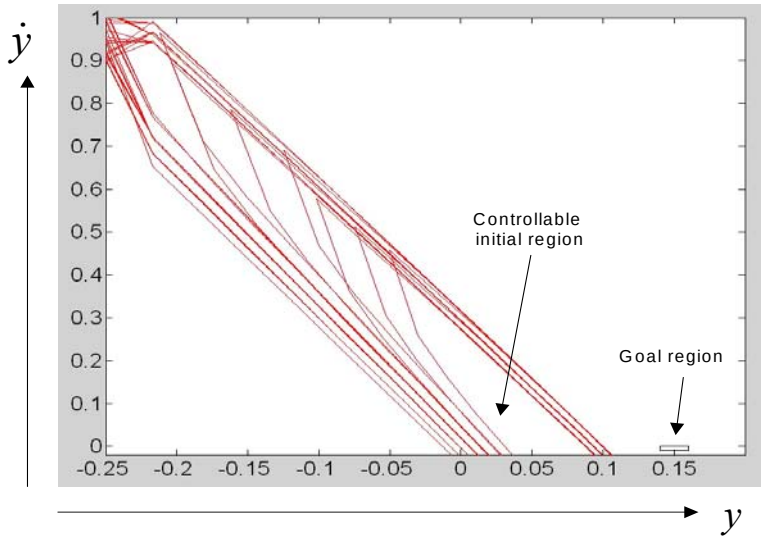



**b.**

Figure 4.16. Flow tubes and state trajectories for forward CM (a.) and lateral CM (b.).




**a.**



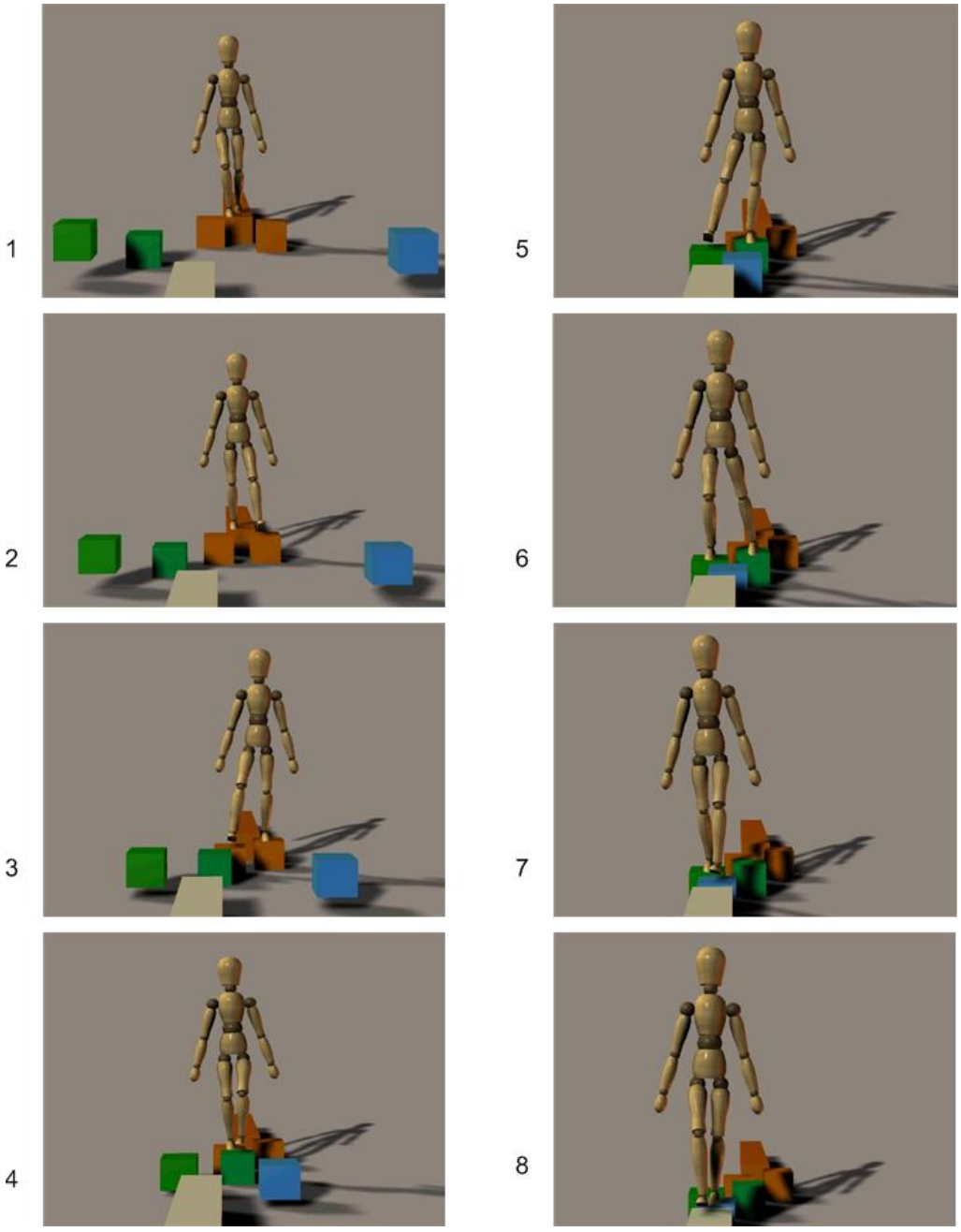

**b.**

Figure 4.17. Initial flow tube cross sections for soccer ball kicking task.

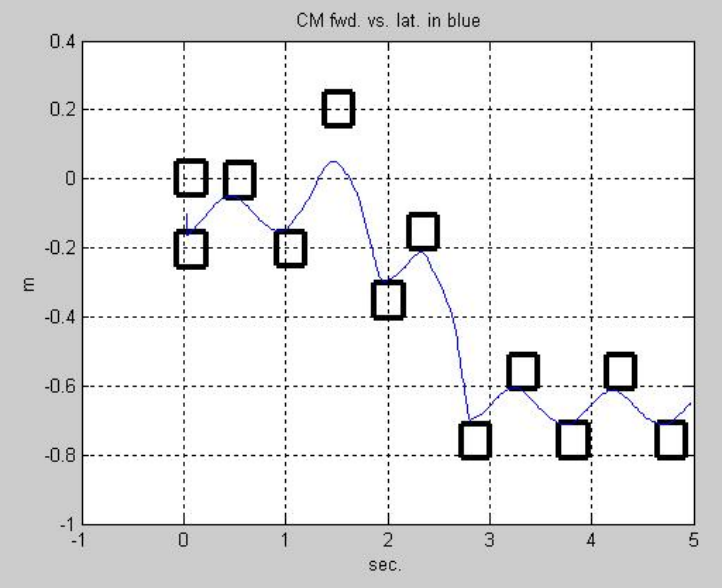
### *4.5.4. Walking on Restricted Terrain*

Figure 4.18a shows the result of a test where the task requires walking on terrain where foot placement is restricted. The irregular stepping pattern is necessary due to the irregular foot placements required by the blocks that the biped is walking on. These blocks move slowly, so the timing of foot placement, as well as the positioning is important. Timing requirements force the biped to move at a relatively fast speed, of about 0.8 m/s. At this speed, the biped can't just balance statically on each block; the fast speed requires dynamic balancing and coordination of the center of mass trajectory. Figure 4.18b shows the CM trajectory and foot placements for this test. The dynamic nature of this task is indicated by the fact that the CM trajectory barely touches the foot placement polygons, and in one case, is 0.1 m away. This indicates that the system is not statically stable in this pose, and is relying on the subsequent foot placement sequence to maintain balance.

In a subsequent series of tests, we subjected the simulated biped to lateral force disturbances as it was executing this task. Due to the limited base of support in the lateral direction, such a test exercises the system at its weakest point. These tests showed recovery from disturbances as high as 40 N, even if they occurred at in-opportune times. For example, a push from the right to the left when the biped is in left single support is a worst-case test because the right (stepping) foot cannot be placed further to the left to catch the biped. The right foot must be placed on the next available block, which is to the right. Thus, the biped must compensate by moving the ZMP and CMP appropriately, using the techniques demonstrated in Section 5.2.




**a.**




**b.**

Figure 4.18.a. Walking by stepping on slowly moving blocks: 1) biped starts on long, narrow path; 2) steps with left foot onto the brown block; 3) steps with right foot onto the other brown block; 4) steps with left foot onto the green block; 5, 6) steps with right foot onto the other green block; 7) steps with left foot onto blue block; 8) finished. b. Foot placement and CM trajectory for irregular foot placement test.

## 5. CONCLUSION

This chapter introduced two computation control strategies of biped walking that potentially provide either neuron- or bio- mechanical principles. The first model is to propose a human walking mechanism based on the functional and anatomical plausibilities of neurobiological systems, and the second to propose a hierarchical execution algorithm for biped walking under disturbance by achieving sequential goal tasks.

Each model provides useful insights into biped walking control. The first model proposes a functionally decoupled system of gait execution and balance control. A feedforward pattern generator constructs gait execution while feedback neural systems control balance. The spinal feedback system provides activation synergies that facilitate control by reducing control dimensionality. This allows the gait pattern generator to provide only a few neural command signals, which are then expanded, according to the synergies, into muscle activation signals. The viscoelasticity of the muscles smoothes the effects of command inputs and enables stable and compliant contact with the ground without explicit computation of joint torques or foot forces.

The second model uses an advanced multivariable controller to transform a tightly coupled nonlinear system into a loosely coupled system of local controllers that is much easier to control. This decoupling allows for definition of plan success in terms of synchronized presence of key control variables in goal regions at key points in the gait cycle. Compilation of the qualitative state plan into a qualitative control plan results in a precise

specification of operating regions and synchronization requirements that result in successful execution of the plan.

The two models have several common points which we consider to be principal elements for controlling high-performance biped walkers.

First, a gait is represented in terms of distinct control epochs: segmentation of neural command signals into five distinct epochs over a gait cycle in the first model (figure 3.2) is comparable with the five qualitative states in the qualitative state plan in the second model (figure 4.3). This suggests that a human gait is achieved through a sequence of distinct control actions that guide the biped through a corresponding sequence of control epochs. For the first model, this control sequence is generated by the neural pattern generator, for the second model, it is generated by the dispatcher sequencing through the qualitative states. Augmenting the control epochs for a standard gait cycle with new ones enables modified gait behaviors, including ones for kicking moving soccer balls, and for traversing difficult terrain using irregular stepping patterns. The control epochs are defined by events, such as ground contact events, and represent qualitative regions in which important parameters and model structures remain constant. In particular, piecewise linearization according to the control epochs is a powerful concept, utilized by both models.

A second common point between the two models is their hierarchical organization. For the first model, muscle activation is controlled by the spinal feedback system, which provides a layer of control abstraction. This abstraction is used by the neural pattern generator, allowing for generation of fewer control signals. At the highest level, the cerebro-cerebellar (supra-spinal) feedback system ensures that the key balance variables, CM position and body orientation, are close to their desired values. For the second model, the DVMC provides a control abstraction similar to the one provided by the spinal feedback system in the first model. The DVMC provides synergies by automatically converting control signals for key balance variables like CM position and body orientation into joint control signals. The task executive in the second model performs the control epoch sequencing and long loop feedback functions performed by the neural pattern generator and supra-spinal feedback system in the first model. Thus, a key feature of both models is the simplification provided by the lower-level synergistic control layers. In both cases, these synergistic control layers are not dominantly demanding because they are not concerned with gait pattern generation; they simply map higher level gait generation signals into muscle activation or joint control signals. Nevertheless, they provide an important simplification for the higher control levels. Thus, the decoupling provided by the lower levels splits the task of walking control into manageable parts.

A third common point between the two models is that they both focus on control of two important variables: horizontal CM position, and body orientation. Horizontal CM position is important for achieving translational balance control, whereas body orientation is important for maintaining angular stability. In both models, the higher control layers monitor these variables, and issue control actions that directly affect these variables. These control actions are translated into muscle activation or joint control signals by the lower control layers. An important point of difference between the two models is that the second model adds explicit control of the stepping foot, whereas in the first model, stepping foot behavior emerges indirectly. Such an indirect approach is appropriate for level ground walking, but the direct approach used by the second model is necessary for challenging tasks like kicking a moving soccer ball, or working on terrain with constrained foot placements.

A fourth common point between the two models is extensive use of linearization. The first model uses a simple linearization about a small number of possible operating points. The linearization is simple, and as a result, has some error over the entire range of operation. Nevertheless, this simple linearization is adequate for stable walking on level terrain, even in the presence of disturbances, as discussed previously. The DVMC in the second model uses an input-output linearization approach, which is more precise, but which is subject to model error, and can be computationally intensive. From a neuro-physiological standpoint, it is unlikely that humans perform such precise computations [Georgopouls, 1988]. These considerations suggest that it may be worth investigating a more approximate, biologically compatible table-lookup linearization approach for the DVMC. This would replace the computationally intensive input-output linearization, making the DVMC computation faster, but possibly less accurate. The results from the first model suggest that such a loss of accuracy is not crucial for adequate performance.

## 6. REFERENCES

Dietz, V. and Harkema, S.J. (2004) Locomotor activity in spinal cord-injured persons, *Journal of Applied Physiology, 96,* 1954-1960.

Dimitrijevic, M.R. and Gerasimenko, Y. and Pinter, M.M. (1998) Evidence for a spinal central pattern generator in humans, *Annals of the New York Academy of Sciences, 860*, 360-376.

Calancie, B. and Needham-Shropshire, P. and Jacobs, P. and Willer, K and Zych, G. and Green, B.A. (1994) Involuntary stepping after chronic spinal cord injury: Evidence for a central rhythm generator for locomotion in man, *Brain, 117,* 1143-1159.

Grasso, R. and Ivanenko, Y.P. and Zago, M. and Molinari, M. and Scivoletto, G. and Castellano, V. and Macellari, V. and Lacquaniti, F. (2004) Distributed plasticity of locomotor pattern generators in spinal cord injured patients, *Brain, 127(5),* 1019-1034.

Olree, K.S. and Vaughan, C.L. (1995) Fundamental patterns of bilateral muscle activity in human locomotion, *Biological Cybernetics, 73*, 409-414.

Ivanenko, Y.P. and Poppele, R.E. and Lacquaniti, F. (2004) Five basic muscle activation patterns account for muscle activity during human locomotion, *Journal of Physiology, 56*, 267-282.

Ivanenko, Y.P. and Cappellini, G. and Dominici, N. and Poppele, R.E. and Lacquaniti, F. (2005) Coordination of locomotion with voluntary movements in humans, *Journal of Neuroscience, 25(31)*, 7238-7253.

Ogihara, N and Yamazaki, N. (2001) Generation of human bipedal locomotion by bio-mimetic neuron-musculo-skeletal model, *Biological Cybernetics, 84,* 1-11.

Taga, G. (1995) A model of the neuron-musculo-skeletal system for human locomotion I. Emergence of basic gait, *Biological Cybernetics 73,* 97-111.

Freitas, S. and Durate, M. and Latash, M.L. (2006) Two kinematic synergies in voluntary whole-body movements during standing, *Journal of Neurophysiology 95,* 636-645.

Tresch, M.C. and Saltiel, P. and Bizzi, E. (1999) The construction of movement by the spinal cord, *Nature Neuroscience, 2*, 162-167.

d'Avella, A. and Saltiel, P. and Bizzi, E. (2003) Combinations of muscle synergies in the construction of a natural motor behavior, *Nature Neuroscience, 6(3),* 300-308.

d'Avella, A. and Bizzi, E (2005) Shared and specific muscle synergies in natural motor behaviors, *Proceedings of the National Academy of Sciences of the United States of America, 102(8),* 3076-3081.

Cheung, V. and d'Avellar, A. and Tresch, M, and Bizzi, E. (2005) Central and sensory contributions to the activation and organization of muscle synergies during natural motor behaviors, *Journal of Neuroscience, 25(27)*, 6419-6434.

Kandel, E.R. and Schwartz, J.H. and Jessell, T.M. Principles of neural science, fourth edition, McGraw-Hill, 2000.

Drew, T. (1993) Motor cortical activity during voluntary gait modifications in the cat. I. Cells related to the forelimbs, *Journal of Neurophysiology, 70,* 179-199.

Bretzner, F. and Drew, T. (2005) Contribution of the motor cortex to the structure and the timing of hindlimb locomotion in the cat: A microstimulation study, *Journal of Neurophysiology, 94,* 657-672.

Porter, R. and Lemon, R. Corticospinal function and voluntary movement, New York, Oxford University Press, 1993.

Peterson, N.T. and Christensen, L.O. and Nielsen, J. (1998) The effect of transcranial magnetic stimulation on the soleus H reflex during human walking, *Journal of Physiology (London) 513(Pt 2)*, 599-610.

Nathan, P.W. (1994) Effects on movement of surgical incisions into the human spinal cord, *Brain,117(Pt2)*, 337-346.

Nielsen, J.B. (2003) How we walk: central control of muscle activity during human walking, *Neuroscientist 9(3)*, 195-204.

van der Kooij, H. and Jacobs, R. and Koopman, B. and van der Helm, F. (2003) An alternative approach to synthesizing bipedal walking. *Biological Cybernetics, 88(1),* 46-59.

Fuglevand, A.J. and Winter, D.A. (1993) Models of recruitment and rate coding organization in motor-unit pools, *Journal of Neurophysiology 70(6)*, 2470-2488.

Jo, S. (2006) Hierarchical neural control of human postural balance and bipedal walking in sagittal plane, Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, USA.

Jo, S. and Massaquoi, S. (2004) A model of cerebellum stabilized and scheduled hybrid long-loop control of upright balance, *Biological Cybernetics, 91,* 188-202.

Baxendale, R.H. and Ferrell, W.R. (1981) The effect of knee joint afferent discharge on transmission in flexion reflex pathways in decerebrate cats, *Journal of Physiology (London), 315*, 231-242.

Brooke, J.D. and Chen, J. and Collins, D.F. and Mcilroy, W.E. and Misiaszek, J.E. and Staines, W.R. (1997) Sensori-sensory afferent conditioning with leg movement: gain control in spinal reflex and ascending paths, *Progress in Neurobiology, 51,* 393-421.

Rossignol, S. and Dubuc, R. and Gossard, J.-P. (2006) Dynamic sensorimotor interactions in locomotion, *Physiological Review, 86*, 89-154.

Duysens, J. and Clarac, F. and Cruse, H. (2000) Loading-regulating mechanisms in gait and posture: comparative aspects, *Physiological Review 80(1)*, 83-133.

Karameh, F.N. and Massaquoi, S.G. (2005) A model of nonlinear motor cortical integration and its relation to movement speed profile control, *Proceeding of 2005 IEEE Engineering in Medicine and Biology*, 27th Annual conference, Shanghai, China.

Massaquoi, S.G. and Topka, H. (2002) Models of cerebellar function. The cerebellum and its disorders, M. Pandolfo and M. Manto, Cambridge, U.K., Cambridge University Press**,** 69-94.

Ito, M. (1997) Cerebellar microcomplexes, in The cerebellum and cognition. J. D. Schmahmann, Academic Press, 41, 475-487.

Thach, W.T. (1998) What is the role of the cerebellum in motor learning and cognition? *Trends in Cognitive Science, 2,* 331-337.

Massaquoi, S.G. (1999) Modelling the funcito of the cerebellum in scheduled linear servo control of simple horizontal planar arm movements, PhD thesis, Electrical Engineering and Computer Science, MIT, Cambridge, MA.

Allen, G.I. and Tsukahara, N. (1974). Cerebrocerebellar communication systems, *Physiological Review, 54,* 957-1006.

Gilchrist, L.A. and Winter, D.A. (1997) A multisegment computer simulation of normal human gait, *IEEE Transaction on Rehabilitation Engineering 5(4)*, 290-299.

Huffman, K.J. and Krubitzer, L. (2001) Area 3a: Topographic organization and cortical connections in marmoset monkeys, *Cerebral Cortex, 11(9)*, 849-867.

Georgopouls, A.P. (1988) Neural integration of movement: role of motor cortex in reaching, *FASEB J, 2,* 2849-2857.

Brown, G.A. (1987) Determination of body segment parameters using computerized tomography and magnetic resonance imaging MIT Master of Science in Mechanical Engineering Thesis.

Clauser, C.E. and Mcconville, J.T. and Young, J. W. (1969) Weight, Volume, and Center of Mass Segments of the Human Body. Technical Report AMRL Tech. Report 69-70, Wright-Patterson Air Force Base, OH.

Goswami, A. (1999) Postural stability of biped robots and the foot rotation indicator (FRI) point. *International Journal of Robotics Research*, *July/August.*

Hirai, K. (1997) Current and future perspective of Honda humanoid robot *Proceedings of the 1997 IEEE/RSJ International Conference on Intelligent Robot and Systems* Grenoble, France:IEEE, New York, NY, USA. 500-508.

Hirai, K. and Hirose, M and Haikawa, Y. and Takenaka, T. (1998) The development of Honda humanoid robot. *IEEE International Conference on Robotics and Automation (ICRA)*

Hofmann, A. and Massaquoi, S. and Popovic, M. and Herr, H. (2004) A sliding controller for bipedal balancing using integrated movement of contact and non-contact limbs. *Proc. International Conference on Intelligent Robots and Systems* (IROS). Sendai, Japan

Hofmann, A. (2005) Robust execution of bipedal walking tasks from biomechanical principles. Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, USA.

Kagami, S. and Kanehiro, F. and Tamiya, Y. and Inaba, M. and Inoue, H. (2001) AutoBalancer: An online dynamic balance compensation scheme for humanoid robots, in "Robotics: The Algorithmic Perspective", Donald, B. R., Lynch, K. M., and Rus, D., editors, A. K. Peters Ltd. pp. 329 – 340

Khatib, O. and Sentis, L. and Park, J. and Warren, J. (2004) Whole body dynamic behavior and control of human-like robots, *International Journal of Humanoid Robotics, 1(1),* 1-15.

Kurzhanski, A. and Varaiya, P. (1992) Ellipsoidal techniques for reachability analysis: Internal approximation, *Systems and Control Letters.*

Leaute, T. (2005) Coordinating agile systems through the model-based execution of temporal plans. Master's Thesis, MIT

Popovic, M. and Goswami, A. and Herr, H. (2005) Ground reference points in legged locomotion: definitions, biological trajectories, and control implications. *International Journal of Robotics Research, 24(12)*, 1013-1032.

Pratt, J. and Dilworth, P. and Pratt, G. (1997) Virtual model control of a bipedal walking robot, *Proc. International Conference on Robotics and Automation (ICRA)*

Slotine, J. and Li, W. (1991) Applied Nonlinear Control. Ch. 6, Prentice Hall, NJ, USA

Takanishi, A. and Ishida, M. and Yamazaki, Y. and Kato, I. (1985) The realization of dynamic walking by the biped robot WL-10RD. In *International Conference on Advanced Robotics, Tokyo,* 459-466.

Tilley, A.R. and Dreyfuss, H. (1993) The measure of man and woman. Whitney Library of Design, an imprint of Watson-Guptill Publications, New York.

Vestal, S. (2001) A new linear hybrid automata reachability procedure. *Hybrid Systems: Computation and Control (HSCC).*

Vukobratovic, M. and Juricic, D. (1969) Contribution to the synthesis of biped gait. *IEEE Transactions on Bio-Medical Engineering, BME-16(1),* 1 – 6.

Williams, B. and Nayak, P. (1997) A reactive planner for a model-based executive. *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*

Winters, D.A. (1990) Biomechanics and Motor Control of Human Movement. John Wiley and Sons, Inc., New York.

Yamaguchi, J. and Soga, E. and Inoue, S. and Takanishi, A. (1999) Development of a bipedal humanoid robot -control method of whole body cooperative dynamic biped walking, *ICRA '99, IEEE,* 368 – 374.