

**INTERACTIVELY RESPONSIVE ANIMATION OF
HUMAN WALKING IN VIRTUAL ENVIRONMENTS**

By

Shih-kai Chung

B.A. June 1988, National Taiwan University, Taiwan
M.S. May 1994, The George Washington University

A Dissertation Submitted to

The Faculty of

The Department of Computer Science of
The George Washington University in partial satisfaction
of the requirements of the degree of Doctor of Science

May 21, 2000

Dissertation directed by

James K. Hahn
Associate Professor of Engineering and Applied Science

© Copyright Shih-kai Chung 2000

Abstract

Computer animation of human locomotion has become popular in recent years because of the desire to use human beings as synthetic actors in three-dimensional simulation environments. Researchers have proposed various motion control mechanisms to simulate human-like figure locomotion. However, most of the animation systems based on these control mechanisms are only suitable for animating human walking on flat ground, without obstacles. For walking on uneven terrain, which is fundamental in our daily life and critical in virtual environment applications, current systems either require major modification and enhancement, or require significant user intervention to control the motion. The main purpose of this study is to provide a new solution to the important problems of walking in various environments.

In this dissertation, we present research into building an animation system which is capable of simulating human walking on varied terrain. Locomotion strategies and optimal control approaches are integrated into the system's motion control structure to allow the user to animate desired motion at different hierarchical levels of control. The results show that this system has three important advantages over most of the existing systems. First, the capability of simulating walking in different environments, such as uneven terrain or obstacle-cluttered environments. Second, interactive simulation can be easily achieved. Finally, a variety of walking styles can be interactively controlled. These advantages make it well suited for virtual environment applications, such as exploring new environments.

Table of Contents

Copyright.....	ii
Abstract.....	iii
Table of contents.....	iv
List of figures.....	vi
List of tables.....	vii
1. Introduction.....	1
1.1. General overview.....	1
1.2. Purpose of the study.....	2
1.3. Overview of dissertation.....	3
1.4. Contributions.....	4
2. Previous Work.....	7
2.1. Kinematics.....	7
2.1.1. Forward kinematic.....	7
2.1.2. Inverse kinematic.....	8
2.2. Hybrid (kinematics and dynamics).....	11
2.3. Dynamics ..	12
2.4. Constraint optimization.....	14
2.5. Genetic programming.....	16
2.6. Motion editing.....	17
2.7. Related studies in biomechanics and human gait analysis.....	20
3. Human Walking Model.....	23
3.1. Terminology of gait.....	23
3.2. Walking on uneven terrain.....	27
3.3. Human model representation.....	29
2.4. Motion control hierarchy.....	32
4. Hierarchical Motion Control.....	34
4.1. Locomotion parameters.....	34
4.2. Footprint planning and locomotion strategies.....	37
5. Modeling Coordinated Leg Motions.....	44
5.1. Stance foot trajectory.....	44
5.2. Swing leg trajectory.....	49
5.2.1. Collision avoidance.....	50
5.2.2. Least energy spent.....	51
5.2.3. Foot movement along the swing-foot trajectory.....	53
5.2.4. Coordinate synchronization.....	54
6. Motion Control of the Body Center of Mass.....	55
6.1. Characteristics of the pelvis trajectory in walking.....	55
6.2. Pelvis movement in the sagittal plane.....	57

6.2.1. Summit at mid-stance	58
6.2.2. Valley at middle of double support.....	58
6.2.3. Exception handling of the pelvis searching algorithm.....	62
6.2.4. Building the pelvis curve	63
6.3. Pelvis movement in the transverse plane	64
7. Motion Control at the Low Level.....	66
7.1. Gait determinants	66
7.2. Motion of the upper body.....	68
7.2.1. Motion of the torso and head	68
7.2.2. Motion of the arms	69
7.3. Personalized human locomotion	71
7.3.1. Interface	71
7.3.2. Adjusting control parameters	73
8. Results	74
9. Discussion and Conclusions.....	78
9.1. Observations and potential improvement	78
9.2. Conclusions.....	80
References	86
Appendix A. Using a Bézier Curve for editing the trajectory.....	82
Appendix B. The <i>displacement vs. time</i> patterns of foot movement in various gaits	83
Appendix C. The weighting factors and their corresponding leg-joint angular motions	84

List of Figures

1. Locomotion cycle for bipedal walking	24
2. The controlled degrees of freedom of the human model.....	29
3. Locomotion system structure	33
4. The direction change of locomotion in 2-D step planning	40
5. Footprint planning along the different 3-D body trajectories	43
6. A two-stage process for computation of the foot place-on and lift-off angles	45
7. Plane geometry associated with a two-link planar leg	47
8. A Bezier curve to represent trajectory of the swing foot	51
9. Displacements of pelvis in three planes of space	56
10. The computation of pelvis location at consecutive mid-stances	60
11. The computation of body banking angle.....	65
12. The motion curves of shoulder and Elbow during walking	70
13. The motion control interface for VWalker.....	72
14. Walking sequences in different environments	75
15. The <i>displacement vs. time</i> patterns of swing-foot movement in various gaits.....	83
16. Variation of leg-joint angles in normal straight walking on flat ground.....	84

List of Tables

1. Effect of sampling number on system's performance.....	76
---	----

Chapter 1

Introduction

1.1 General overview

The representation and display of 3-D virtual worlds has been the subject of much recent research, and the desire to put human actors in such a simulated world has made human animation become a popular research area. Still, the synthesis of human motion represents one of the most challenging areas in computer animation to date. One of the problems is that the human being possesses more than 200 degrees of freedom (DOF) (even for simplified human figure representation, there are usually more than 30 DOFs). Controlling these many hierarchical DOFs to express a certain desired motion presents a difficult problem. The other challenge is the fact that the viewers see each other's motion everyday and are very sensitive to erroneous movements (it simply doesn't look right, although isolating the factors of the incorrect movement is often difficult). Walking, being the most common means of moving about and an essential part of our daily life, naturally has been the most popular research area in human animation.

Human walking is a smooth, highly coordinated, rhythmical movement by which the body moves step by step in the desired direction. Numerous studies from various fields, such as biomechanics, robotics, and ergonomics, have provided a rich data base on "normal" straight-walking gait patterns. In the field of computer animation, because of the complex hierarchical structure of the human being, most of the research in motion control of human figures has been devoted to ways of reducing the amount of specification necessary to achieve a desired motion. In these models, motion control is

implemented through the application of a set of constraints, with different constraint sets generating different movements. The movements produced by these models can be quite fluid and natural. However, applying these models usually requires considerable computation, as well as significant expertise by the animator, to produce a desired motion. This is particularly true when the envisioned motion includes stylized movements that are deliberately objective, such as walking on uneven terrain.

1.2 Purpose of this Study

In light of the recent surge of interest in virtual environment applications, much research has been devoted to solving the problems of manipulating humans in 3-D simulated worlds, and especially to human locomotion. However, most of the animation approaches based on these studies can only generate walking on flat ground, without obstacles. Lacking the locomotion capabilities for walking on anything but flat ground, their application in virtual environments is inevitably limited.

The capability of walking over uneven terrain and cluttered environments is fundamental in our daily life (e.g. stair climbing and descending), and is critical on occasion, such as environment exploring. To date, only a few systems are capable of simulating human walking on non-flat ground. Yet, most of these systems require further user intervention, which can be a tedious job, and are usually unable to produce continuous walking over uneven terrain interactively.

The primary objective of this study was to develop an interactive system for simulating human walking. Several existing and novel techniques are integrated to generate a flexible motion synthesis tool for our research goals. The capability of

simulating human walking on uneven terrain with interactive modeling of detailed motion make our approach well suited for virtual environment applications and interactive games.

1.3 Overview of dissertation

- Chapter 1 briefly describes the problems and current status of computer simulation of human locomotion. The motivation of this study and its contributions to computer graphics character animation in virtual environment applications is stated.
- Chapter 2 reviews previous computational models of human animation, with focus on human locomotion. Related studies of human locomotion in biomechanics and human gait observation are also described.
- Chapter 3 details the timing events in the walking gait cycle and the problems of simulating human walking in virtual environments. It also outlines the hierarchical animation system structure and its controlled degrees of freedom of the articulated human figure.
- Chapter 4 presents motion control at the high level of the animation structure, with walking speed and marching direction being the two major walking attributes. It also illustrates the stepping strategies in virtual environments and footprint planning algorithms.
- Chapter 5 describes our motion control algorithms for specifying and controlling foot movements during walking. Treated as the end-effector in our inverse kinematic algorithm, the support and swing foot trajectories are computed based on terrain constraints and walking attributes.

- Chapter 6 elaborates the computational model of the figure's body center, the pelvis. Treated as the root of this articulated figure, pelvis movement is modeled using optimization techniques to ensure smooth leg joint movements, while the foot constraints are satisfied.
- Chapter 7 describes the motion control at the lowest level of the animation structure to produce different walking. Upper body and arm motions during gait cycle are described. Through a spin-button-based interface, detailed movement is controlled at each degree of freedom.
- Chapter 8 shows the results of using our algorithms in simulating human walking in virtual environments. The performance issues regarding scaling the complexity of the algorithm to achieve the best synthesized motions in real time are also addressed.
- Chapter 9 presents some observations and discussion of our motion control techniques. Some potential improvements of the system are also offered. Conclusions are included in this final chapter.

1.4 Contributions

The main purpose of this research is to provide a new solution to the important problems of simulating human walking in various environments. Studies from various fields, such as animation, robotics, biomechanics, and psychology, are integrated into a human walking model that supports real-time creation of human walking on various terrain. The contributions of these motion control techniques, as to the existing human locomotion models, include:

Capability: Much research has been devoted to simulating human-like figure locomotion, however, none of these studies is able to animate locomotion in various environments without a user explicitly detailing the motions. This study explores new techniques that adapt the gait to the terrain, compute foot placement automatically, and customize the gait, all interactively, to simulate human walking in various virtual environments:

- Footstep planning strategy that plans the footstep two steps ahead of the current stance step, adapts the gait to the terrain. It frees the animator from the laborious details of foot placement. This concept of look-ahead footstep planning has not appeared in the literature of computer animation.
- New techniques for computing body trajectories during locomotion in various environments are presented. These trajectories are calculated through a search for optimized objectives that satisfy all constraints to accommodate various terrain.

Easily controlled: Although the concept of hierarchical control is not new, motion control mechanisms with intuitive parameters are integrated into our system at each hierarchical control level to balance the control and automation. The resulting walk is dependent on both the parameters that adapt the gait to the terrain, and those that customize the animated characteristics. As the control hierarchy descends from the top, automation recedes, while more control is given to help the user direct desired motions. For example, at the high level, through a small number of intuitive parameters, such as “walking speed” and “destination” (e.g. walk to A at speed X), the system will generate the corresponding walking motion. As we move down the hierarchy, additional

locomotion attributes are provided to simulate individual walks; that is to produce walks under the same constraints (i.e. footprints along the traveling path), but with a variety of characteristics. An important feature of this approach is that the animator can adjust the possible animations generated by the system to what is really needed.

Responsive: The computations of objective-optimized body trajectories are fast. The complexities of the trajectory computations are independent of the terrain. Also, the overall reduction in complexity through the use of a spatial discretization ensures the motions are generated with minimal latency based on user's inputs. Finally, the computation-intensive search algorithms for the body trajectory are scalable. Depending on the computation platform, the algorithms can be scaled to produce "optimized" results within the interactivity requirement. Thus, the animator will be able to adjust the locomotion parameters and watch the resulting motions on the fly. This capability is important in helping the animator direct the desired motions, and also critical in virtual environment applications.

Realistic: Biomechanics knowledge of human walking and studies of motion analysis were used to aid computer animation of natural walking.

The above characteristics imply that underlying the locomotion system there exists a comprehensive structure incorporating multiple knowledge bases and a reasoning capability. There is also sufficient computational capability to implement locomotion algorithms and simulations of desired motion. From an application point of view, the motion control techniques are well suited for virtual environment applications and interactive games.

Chapter 2

Previous Work

Motion control of articulated figures has been a popular research area in computer animation for years, and there is a large body of studies focussed on the important problems of locomotion control. These techniques are roughly classified into five categories: kinematics, dynamic, hybrid, constraint optimization, and motion editing. What follows is a brief summary of these techniques, with emphasis on algorithms that have been applied to human walking, especially those used for motion control of limb trajectories.

2.1 Kinematics approaches

Kinematics approaches produce motion from positions, velocities, and accelerations, that is, all the geometrical and time-related properties of the motion. Kinematics approaches for simulating human locomotion have been described by several researchers over the years [5, 9, 10, 41, 51, 52]. These approaches for articulated figure animation generally fall into one of the two categories:

2.1.1 Forward kinematics

Forward kinematics approaches provide motion control by specifying the joint angles over time. The motion of the end-effector is determined as the accumulation of all transformations from the chain root to the end-effector. The major advantage of forward

kinematics approaches over the other motion control techniques is that they provide the animator complete control of the motions in minimal cost of computation need. However, the animator will have to deal with the following difficulties:

- When applying forward kinematics directly, obvious constraints imposed on the motions may be violated. For example, in animating human locomotion, the most fundamental constraints are that the supporting foot should not go through or off the ground, and that the global motion should be continuous (especially at the heel lift-off and strike points). Special handling will be required to satisfy these constraints. One solution for this locomotion problem with forward kinematics is to switch the root of the hierarchical structure, based on the constraint situations (i.e. make the supporting foot the root).
- Although motions generated by this technique look convincingly real, the technique is quite labor-intensive and requires considerable talent in order to get the desired results. As the complexity of the articulation increases (i.e. a more complex human model or movement), the usage of this technique will become less practical.

Because of the complexity problem of human structure, much of the research in motion control for human figures has concentrated on providing the animator with high-level control, which will reduce the amount of specification necessary to achieve a desired motion. An early work by Zelter [65] used hierarchical motor control techniques to animate locomotion of a human skeleton with a straight-ahead gait over level, unobstructed terrain. Variations of walking, such as different walking styles or walking on moderately uneven terrain, were achieved by parameterizing the generalized walk

controller and its associated motor programs. Unfortunately, this requires the user the detailed knowledge of the skeleton animation system as well as programming experience. Another drawback of this approach is that the animator must trade artistic control in return for automatic motion synthesis.

Bruderlin and Calvert [7] proposed procedural animation techniques to animate personalized human locomotion. In their system, three locomotion parameters, step length, step frequency and velocity, are used to specify the basic locomotion stride. Then, additional locomotion attributes are added at different levels of the motion control hierarchy to individualize the locomotion. The complexity of their control algorithm is simple enough to provide the animator interactive control of personalized human locomotion. Because their computation model is mainly based on normal walking on flat ground, without further modification of the model, its application is highly limited in virtual environments.

2.1.2 Inverse kinematics

Inverse kinematics for end-effector goal positioning is adopted from robotics. It computes the joint angles for each segment in the chain structure from the position and orientation of the end of the limb. The advantages of inverse kinematic approaches over the other motion control techniques are first, the animator defines the configuration of the end-effector only, and inverse kinematics will solve for the configurations of all joints in the link hierarchy. In general, specifying only the motion of the end-effector is more intuitive and easier than explicitly specifying all joints for the animator. This also implies that the quality of the motion is highly depended on how well the body trajectories are defined.

Second, constraint satisfaction, such as the feet must stay at certain positions during locomotion, can be precisely executed, using inverse kinematics. This constraint-satisfaction characteristic makes inverse kinematic method a useful tool in dealing with constraints regarding end-effector's configuration for most of the existing animation system.

Boulic et al. [5] used a generalization of experimental data based on the normalized velocity of walking. The generalization, in its direct application, could produce undesired results, such as parameters violate some of the kinematic constraints imposed on walking. Inverse kinematic was implemented to correct these problems. Among the multiple inverse kinematic solutions, the one that is the closest to the original motion is chosen to preserve the original characteristics of the walking data. Based on the *Jack* system [42] developed at University of Pennsylvania, Phillips and Badler [27] implemented an inverse kinematics algorithm to generate motions. The users have to choose properly the end-effectors and then define sets of constraints that drive the limbs to move in desired patterns. Minimization of energy described by the constraints is used to choose the set of joint angles among the multiple inverse kinematics solutions. Koga et al. [29] used a path planner to compute the collision-free trajectories for cooperating arms to manipulate a moveable object between two configurations. An inverse kinematic algorithm was utilized by the path planner for the generation of forearm and upper arm postures to match the hand position. Then, joint angle of the wrist was computed to match the hand orientation.

For systems that devote to human locomotion, Girard's PODA [17] uses a mix of kinematics and "pseudo-dynamic" methods to simulate human locomotion. A multi-pass

process is used to determine the body motion which best fits a set of footprints. The vertical body motion is computed by a fixed family of functions during support phase (in the case of running, a ballistic motion is added after the end of support phase). The horizontal motion is computed independently using a velocity-error feedback loop. As the motion of the body is defined using the kinematic constraints and simple dynamics, the legs are animated kinematically, using a pseudo-inverse Jacobian technique to make the leg angles close to the desired angles, while keeping the foot on the ground during support. Implementing the above approaches in the animation system, PODA appears to be one of the human animation systems, which attempt to combine automatic simulation and artistic control, with the later more emphasized. Using the above approaches, some of the most impressive human animations to date were produced.

2.2 Hybrid (kinematics and dynamic)

Beyond kinematics methods, some hybrid locomotion techniques have been proposed to generate walking motions by adding physical properties. The task is to find effective combinations that generate realistic motion while providing animator reasonable and intuitive control over the motion. In general, simplified dynamic models are applied to simulate some parts of articulated figure, such as the swing leg, support leg, or the body as a whole. They are responsible for the enhancement of realistic part of the animation. Kinematics, on the other hand, gives animator the flexibility to control the desired motions. Several researchers [6, 17, 26] have implemented this technique in articulated figure animation.

Armstrong et al. [1] and Wilhelms [59] both proposed similar methods where all of the links of the articulated figures were under control of the dynamic simulation, but the animator could constrain the motion through kinematic means. For each individual link in the structure, one of the four kinematic control strategies was assigned to constrain its movement. Then, the system will generate required forces that work to exactly match the kinematically defined motions. A similar technique was proposed by Westenhofer and Hahn [57]. Different from [1] and [59], dynamic is used to enhance kinematically created motion with realistic effects, instead of exactly matching it. Considering the realism is highly depended on the kinematic specification for [1] and [59], Westenhofer and Hahn's approach provides more flexibility in achieving natural continuous motion.

Bruderlin and Calvert [6] use a similar mix of techniques to generate parameterized walking motion. The concept of step symmetry (based on the symmetry of a compass gait) is applied to find the end positions of the supporting hip, and a telescoping leg model with two degrees of freedom is used to compute the trajectory of the supporting hip during step time. Rather than using a general dynamic model, the equations of motion are tailored to suit for only a specific range of movement and time period. Proper forces and torques that drive the dynamic model of the leg are then determined by numerical approximation techniques. Kinematics, in turn, work for the cosmetics, and animates the feet, upper body, and arms kinematically to mimic the pattern observed in human walking.

2.3 Dynamic

Dynamic approaches describe motion by a set of forces and torques from which kinematic data are derived. Dynamic simulation and control algorithms [1, 15, 22, 24, 28, 30, 35, 44] have been used to generate motions of articulated figures for years, and there is also a significant body of robotics research concerning the control of bipedal locomotion, as well as biomechanics for simulating human walking motions. However, to date physical-based modeling of human locomotion still presents one of the most challenging tasks in the computer animation community. This is probably because joint contact and individual muscle forces during gait are still not well-known, and aside from the difficulties in modeling formulation, and solution, determination of limb center of mass and inertial properties add more complexities and uncertainty to the problem.

McKenna and Zeltzer [35] simulated the gait of a virtual insect by combining dynamic simulation and a walking algorithm that was based on the motion patterns observed in insect locomotion. Raibert and Hodgins [44] used a similar approach but a different motion controller. They fashioned the models from analyses of robots and real creatures. Numerical integration of the dynamic model and specific control algorithms were used to generate running and jumping (with a ballistic flight phase) motions of multi-legged imaginary creatures.

Hodgins et al. [22] introduced a dynamic approach to animate human running. The control algorithm is based on a cyclic state machine which determines the proper control actions to calculate the forces and torques that satisfy the requirements of the task and input from the user. Hodgins and Pollard [24] further extended the work of [22] to show that existing simulated motion can be adapted to new dynamic models while

maintain the important characteristics of original motion. Using their approaches, they are able to animate the running motion of a child, woman, and imaginary biped creature by modifying the control system for a man.

The results of McKenna & Zeltzer and Raibert & Hodgins have proved that dynamic approaches with proper control algorithm can produce some very life-like and experimentally validated motions. However, the motions produced to date have been limited to relatively simple creatures performing simple locomotion. For autonomous locomotion on rough terrain or cluttered environment, a more robust model with intelligent control algorithm will be required to achieve the animation goals.

2.4 Constraint optimization

Constraint optimization approaches generate animation through an optimization of the objective subject to the constraints specified by the animator. Modeling the coordinated articulated figure motion is fundamentally a problem of control, due to the nonlinear relationship between joint motions and limb movement and the need to satisfy constraints on a movement's trajectory, speed, and energy expenditure. Furthermore, empirical studies of coordinated animal motion suggest that limb trajectories and body movement seem to be formulated in terms of optimization of performance, such as minimization of jerk about the end of the limb [3].

Witkin and Kass [63] used spacetime constraints to control the motion of a jumping Luxo lamp. The implementation of *spacetime* in Witkin and Kass's work was limited by the fact that the objective functions had to be optimized over the entire span of an animation. To reduce the computational complexity of optimization and provide user

more control over the motions. Cohen [13] divided the original spacetime work into subsets or smaller spacetime windows, over which subproblems are formulated and solved. Liu et al. [32] proposed a hierarchical spacetime constraints paradigm to further lessen the computational complexity problem. Their system provides a means to add detailed motion only when it is required, thus minimizing the number of discrete variables and resulting in faster optimization iterations. These spacetime approaches, in general, are capable of producing realistic results. However, they all suffer from a number of computational difficulties when the complexity of the character or animation increases, thus, are not well suit for interactive human figure simulation.

Van de Panne [56] proposed a locomotion system to use footprints as the basis for generating animated locomotion. The foundation of his approach is to simulate the motion solely in terms of a center of mass trajectory which itself is synthesized from the footprint information. The footprint planning algorithm is formulated for bipedal characters and uses some timing information in addition to the footprint locations and orientation. Similar to the work of [6], “virtual leg” (i.e. a telescope-like leg lengthed from the foot support point to the center of mass) concept is introduced in the optimization process. The objective function is optimized to minimize the sum of the measures of “physical plausibility” and “perceived comfort” for the resultant motion which is constrained to match given footprint and timing information. Since only simple dynamic (physical plausibility) and kinematic terms (length of the virtual leg) are required for the optimization process, interactive simulation is achievable. Using this constraint optimization technique, a couple of interesting examples of a dinosaur walking on regular and spiral staircases bipedally were shown.

Aiming at simulating human walking in various virtual environments. Chung and Hahn [12] presented a hierarchical motion control system for animating human walking along predetermined paths over uneven terrain. Their method ensures that the foot remains in contact with the ground during stance and avoid collision during swing. The joint angles for the lower limbs and the trajectory of the pelvis are computed by inverse kinematic and optimization procedures. Using the proposed control algorithms, their walking model can be adjusted for ascending slopes and stairs.

Constraint optimization techniques have shown to be able to automatically generate expressive and natural limb motions that satisfy several of the basic principles of animation. Enhanced spacetime techniques, such as [13, 30], are especially suitable for complex motions. For example, locomotion on rough terrain could be broken into multiple spacetime windows to satisfy the constraints and animation goals. However, the motions generated are highly depended on the animator's ability to program the mathematical objective functions that meet the goals of a desired animation. Unfortunately, finding proper objective functions and formulating them for certain motions appeared to be a difficult task for the animators.

2.5 Genetic programming

Genetic programming uses the concepts commonly used in genetic algorithms to write programs. It has been used to provide solutions to a variety of problems in computer animation. For articulated figure motion, it defines a hyperspace containing an indefinite number of possible motions and behaviors. To direct the evolution towards a specific motion or behavior, such as walking, running, and jumping, appropriate "fitness"

evaluation functions must be used to select the desired results. The act of these fitness functions is just like the natural selection, which selects the most fit individuals to survive and prosper in real life. Not many published works have addressed the problems of animating articulated figure using genetic programming techniques. Sim [49], Gritz [19] and Hahn have developed systems to animate articulated figures' behaviors and movements in simulated virtual world.

An important issue in genetic programming is complexity vs. control. The genetic programming technique defines a hyperspace containing an indefinite number of possible behaviors, some of them might be difficult to create or design by the other animation techniques. However the advantage of automatic generation of complexity in genetic programming usually comes with the lacking of control over the motion. That is, in general, the users have to sacrifice some control when using these approaches. Similar to the objective functions in spacetime approaches, the fitness functions are the deciding factors in genetic programming animations. For articulated figure motion, especially intentional movement of complex articulated figure, such as human locomotion, determining the proper fitness measures and formulating them presents a big challenge to the animators.

2.6 Motion capturing and editing

An alternative way to obtain movements of articulated figures is capture the motions from live subjects. Postures or motion sequences can be obtained with motion capture to constitute libraries of postures/sequences. They can later be reused/modified and combined with editing tools. The complexity of human figure and the limitations of

current motion control systems, coupled with the increasing popularity and maturity (especially the hardware) of motion capturing, have made motion editing techniques [8, 39, 55, 64] become the recent trend of human animation.

Wiley and Hahn [59] showed that the range of possible motions can be greatly expanded by linear interpolation from a set of example motions that are similar to the desired motions. Similar interpolation technique was also proposed by Rose et al. [47]. In their system, non-uniform time scaling of the data sequences is used for the interpolation scheme to work. The applications of both systems are somehow limited by the fact that the desired motion is based on interpolation of similar motion sequences. This makes their approaches more appealing for periodical motion, such as human locomotion.

Spacetime constraint techniques are also broadly adopted in motion editing systems. Gleicher [20] used spacetime constraints to edit pre-existing motion for new needs. Because the goal of the system is to achieve interactive editing, many tradeoffs have been made to improve performance. For example, instead of seeking the perfect objective function to control the motion, as used in previous spacetime constraint approaches, a simpler objective function, which minimizes the amount the points on the characters which are displaced over the course of the motion, is used to make interactive performance possible.

Rose, et al. [46] combined spacetime constraints and inverse kinematic constraints to generate transition between motion sequences. The motions of the support limb and the root of the body are determined kinematically. The horizontal component of the root position is interpolated based on the horizontal velocities/accelerations at the beginning and end of transition while the vertical position is linearly interpolated from the end of

the first motion to the beginning of the second motion. Inverse kinematics constraint is enforced to ensure kinematic constraints are satisfied during transition for the support limb. As for the motion control of all the other limbs, a spacetime constraint approach which tries to minimize the torque required to transition from one motion to another while maintaining the joint angle constraints is employed.

Similar spacetime constraint techniques for motion transformation were also proposed by Gleicher [20], Lee and Shin [31]. Gleicher [21] further extends [20] to adapt the motion from one character to another character with identical structure but different limb lengths. To retarget motion from one articulated character to another, some basic features of the motion (for example, the supporting foot must stay on the ground for walking) are set to be the constraints. If the constraints are violated when the motion is applied to a different character, an adaptation to the motion must be made to re-establish the constraints in a manner that fits the motion. The retargeting method is a spacetime constraints solver that considers the entire motion simultaneously. To preserve the nature of the original motion, the magnitude of the changes is minimized to compute the adaptation to the motion. Just like his previous spacetime work [20], to make this system more practical in use, some tradeoffs are made to improve the system's performance.

One major problem of using spacetime-constraint approaches in simulating human motion is dealing with the complexity of the spacetime optimization processes. Popovic and Witkin [43] describe a character-simplification methodology for mapping a motion between characters with drastically different numbers of degrees of freedom. Spacetime editing was applied on the simplified character (less degrees of freedom) to get spacetime motions. These simplified spacetime motions are then mapped back to the original

motion to generate the final “transformed” motion. Because all dynamics computations are performed on the simplified model, the complexity of spacetime optimization can be greatly reduced. On the other hand, since no dynamics computations are done on the full character model, the transformed motion is not physically correct.

2.7 Related studies in biomechanics and human gait analysis

Research in biomechanics and human gait analysis [11, 13, 23, 24, 36, 48, 58, 61] has made extensive studies of human body motion during normal level walking. Principle results have come from careful analysis of motion patterns, such as configuration (both position and orientation) of body joints, muscles’ activities (from electromyography), and reaction of the foot with the ground (force plate). They provide a rich resource for simulating human locomotion. However, most attention has been on level walking. To date published work that addresses non-level walking is rare.

Based on the hypothesis that the behavior will be such as to minimize the amount of mechanical work done, Beckett and Chang [4] made studies of the energy expended in walking by analysis of the motion of the leg and foot in the swing phase of a step. The energy consumed is obtained by evaluating the work done in traveling a given distance. It appears that the results check reasonably well with natural gait, and indicates that for a given individual there is a natural gait at which he can travel a given distance with minimum effort.

The model of energy minimization does not, however, take into consideration the necessity of maintaining balance during gait. In Redfern and Schumann’s work [45], they proposed a model of foot placement control which provides a stable base of support. Foot

placements are chosen to minimize the sum, in terms of position and velocity with respect to the pelvis, of the supporting and swing feet. Experimental data were collected to test this model during walking trials of different speeds. Results show that the sum of the supporting and swing feet (positions relative to the pelvis) is very close to zero at heel contact, supporting the positional control hypothesis that foot placements are dependent upon location of the stance foot with respect to the pelvis in order to help maintain balance during gait.

Going up and down stairs is a common activity of daily living. From a mechanical viewpoint, it is quite different from level walking. Flynn [16] and Livingston [33] studied at the kinematics of stair walking, and detailed the joint motions of the lower limb. Through the analysis of the temporal events and the angular motion of the major joints (hip, knee, and ankle) of the legs, movements of stair climbing and stair descending were compared on different staircases.

Andriacchi et al. [2] have studied the motions, forces, and moment at the major joints of the lower extremities in subjects going up and down stairs. Their work has provided one of the most comprehensive sets of data on lower-limb mechanics in normal subjects during stair walking. The common patterns of motion, forces, and muscle activity of the lower limbs were described, as was some useful information on the strategy changes in stair walking.

An analysis which integrates kinematic and kinetic data of lower limbs in stair walking was described by McFadyen and Winter [34]. Some of the new finding in their study includes first, strategies for climbing and descending stairs may vary, but there are some basic mechanical patterns. Most variability is seen at the hip. Second, significant

progression occurs during ‘pull-up’ in early stance for ascent and ‘landing’ in late stance for descent. The knee extensors are responsible for the greatest generation of energy during these events. Third, despite the fact that the magnitude of the supporting moments for stair climbing are greater, descent, level, and ascent walking all exhibit a supporting moment patterns of similar shape. From the animator’s viewpoint, this observation might indicate the possibility of a generic motion control mechanism for all human stepping activities.

Townsend and Tsai [54] proposed a bipedal robot model for uneven terrain walking. Their approach uses a common locomotion algorithm and varies the coefficients and initials to generate a certain range of gaits. The climbing and descending gaits were synthesized according to generalized postural stability and other feasibility requirements for a kinematically constrained, articulated walking model. Although their studies are for biped machine, instead of human, many of the practical constraints and conditions are derived from human motion characteristics or to be compatible with human motion. The results show that general characteristics can be identified with the swing leg take-off or touch-down conditions for a given gait algorithm. Thus, system kinematics is such that iteration or control could utilize the initial and last terminal configuration data to define subsequent walking, and a variety of walking can be achieved by modifying the same basic gait algorithm and varying initial conditions.

Chapter 3

Human Walking Model

Walking is the most common means of moving about and is one of the essential activities of our daily life. Other locomotion methods such as running, and, less commonly, hopping and jumping, all have common patterns of movement, and by studying walking, it becomes easier to understand the rest. Human walking can be described as a smooth, highly coordinated, rhythmical movement by which the body moves step by step in the marching direction. It requires the simultaneous involvement of all lower limb joints in a complex pattern of movement.

Basically, all normal people walk in the same way. From human gait observations [36], the differences in gait between one person and another occur mainly in movements in the coronal and transverse planes. Throughout the whole body, those joint movements which occur in the sagittal plane are very similar between individuals, and if the upper limbs are unencumbered, they actually demonstrate a stereotyped pattern of reciprocal movement in phase with the lower limbs. The above observations lead our human walking system design to focus more on lower limb joint movements, especially in the sagittal plane, and leaves the rest of the body joints to the animator for desired movements.

3.1 Terminology of gait

Human walking is a complex activity, and, for the purpose of computer simulation, we need to analyze human gait and break it down into temporal and spatial components.

Some of the following terminology of gait relates to the period of time during which events take place, and some refers to the positions or distances covered by the limbs.

Gait cycle

The *gait cycle* is defined as the time interval between two successive occurrences of one of the repetitive events of walking. Although any event could be chosen to define the gait cycle, it is usually convenient to use the instant at which the heel of one foot strikes the floor as the beginning, and the moment when the same heel strikes the floor again as the ending of the gait cycle. Based on the events during the gait cycle, it can be subdivided into support, swing, and double support phases, which describe the periods of time when the foot is either in contact with the floor, or swinging forward in preparation for the next step. These phases and their timings are illustrated in Figure 1.

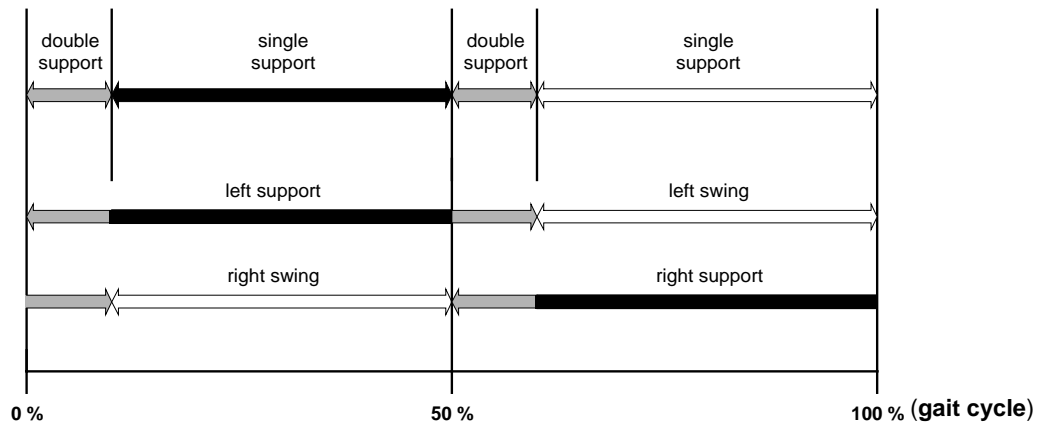


Figure 1: Locomotion cycle for bipedal walking

Support phase

The support phase is the period of time when the limb under consideration is in contact with the floor. It provides the stability of the gait, and is necessary if an accurate swing phase is to take place. Based on the spatial relationship between the supporting foot and the floor, the support phase can be further subdivided into the following stages.

Heel strike: This is the first moment of foot-floor contact for the leading limb. At the moment of heel strike the following limb is also in contact with the floor, giving a phase of double support. In normal walking, this is the moment that the center-of-mass of the body is at its lowest, and the walker is most stable.

Mid-stance: This is the period that the supporting foot is flat in relation to the floor. In mid-stance, the body is carried forward over the supporting limb, and the opposite limb is in the swing phase. The whole body center-of-mass passes from behind to in front of the supporting foot during this phase. It rises to its highest position in relation to the floor at about the middle of this period. This is also the position where the walker is least stable.

Push off: This period starts from the end of 'flat foot' and ends at the end of support phase. Initially, there is 'heel off', followed by a propulsive stage that is called 'push off' which leads to the moment of 'toe off' when propulsion ends and the swing phase starts.

Swing phase

During the swing phase, the swing limb moves in front of the supporting limb so that forward progression can take place. This phase can be subdivided into three stages.

Acceleration: The driving forces come from the hip (major) extensors and plantar (minor) flexors. The non-weight-bearing limb is accelerated forward in this period.

Mid-swing: This corresponds with mid-stance. At this moment the swing limb passes the supporting limb with rather steady speed.

Propulsive breaking: In this final stage of the swing phase, the lower limb muscles work to decelerate the swing limb in preparation for heel strike. The activities of the muscles in this stage are usually eccentric and need less energy than phases of the gait cycle when concentric activity is required to accelerate a limb [61].

Double support phase

The double support phase is the period of time when both feet are in contact with the ground. It is a small interval during the gait cycle when two leg events are overlapped: the final fraction of the support phase from one leg, and the beginning fraction from the other leg. Its temporal length is equal to the difference between the support phase and the swing phase. On normal walking, this also is the period of time where the body travels through its lowest vertical height during the gait cycle.

Duty factor

Leg duty factor describes the time a foot stays on the ground as a fraction of the gait cycle. For bipedal gait, this can be used to distinguish between walking and running. If the leg duty factor exceeds 0.5, the figure is in walking mode, and if it is less than 0.5, the figure is in a running state. Human gait observations have shown that during average speed of normal walking, the support phase takes about 60% of the time of the gait cycle and the swing phase about 40%. This means that average normal walk has a leg duty factor of about 0.6.

The double support phase and leg duty factor can be computed as follows.

Step duration = Support duration + Swing duration

Duty factor = Support duration / Step duration

Double support duration = (Support duration - Swing duration) / 2

3.2 Walking on uneven terrain

Walking on uneven terrain is a common activity of our daily living. Like normal walking, there is a support phase, a swing phase, and a phase of double support in the gait cycle. It is a modified walking activity with similar patterns of joint movement and muscle action of normal walking. The differences between level and uneven-terrain walking activities are that the latter has greater ranges of motion of the different joints, joint forces and moments, during gait. Kinematic studies [33,54] have shown that in non-level walking, compared with level walking, the largest range increase of joint motion occurs at the knee joint, with no significant change at the ankle joint. For walking on uneven terrain, the ranges of hip and knee joint movement are greater than in normal walking, and there is considerable vertical translation of the center-of-mass making it an activity that requires more energy. Because the terrain may vary greatly in height, the range of movement and the vertical translation of the center-of-mass will vary according to the roughness (mainly the height difference) of the terrain.

Previous human locomotion approaches have generated convincing results in animating human normal walking. Also, studies in biomechanics [2, 34, 52] have indicated that a significant degree of similarity can be noted in the efforts required for ‘normal-appearing’ uneven-terrain and level walking for modification of the basic gait algorithms and varying initial conditions. Still, not much success has been achieved in

computer simulation of human non-level walking today, due to the following difficulties imposed by uneven terrain:

- Footstep planning is more difficult on uneven terrain than flat ground. For most step-oriented approaches, this further increases the difficulty of achieving interactive simulation, which is essential in virtual environment applications.
- Adding the extra constraints imposed by uneven terrain, specification and control of limb trajectories is more challenging. For example, the trajectory of the stance foot will have to adapt to the supporting ground, and the trajectory of swing foot must be collision-free from the uneven ground.
- More importantly, synchronizing these limb trajectories to generate smooth joint movements requires more effort. Since non-level walking requires greater range of limb movements to avoid the obstacles in the path than normal walking, finding the key limb trajectory (i.e. root of the articulated links) to ensure natural joint movements is a critical and non-trivial task.

Specification and control of limb trajectories are areas of active research in robotics, and animation techniques that adopt robotics and biomechanics knowledge should be able to generate legged straight-path walking. However, the natural clutter and constraints of a complex environment tend to restrict the usefulness of a straight-path walking control mechanism. To simulate human walking along any desired path in various environments, new stepping strategies inspired by human gait observations and a collision-free path planning algorithm are implemented in the system. Also goal-directed

inverse kinematics, combined with optimizations of limb trajectories and joint angles, are used in computing the motions of walking humans in virtual environments.

3.3 Human model representation

This section describes the kinematic structure of the human figure model. The default kinematic model used in our simulation was adopted from the 3-D geometric model of the human skeleton created by Stredney [50]. The kinematic data of our model are parameterized from joint to joint, as matched to the geometric skeleton model in its default anatomical position. Although Stredney’s model provides precise details of the human structure, controlling all of its degrees of freedom is impractical for animation purposes. Thus, a higher level of kinematic-complexity representation is included in the model.

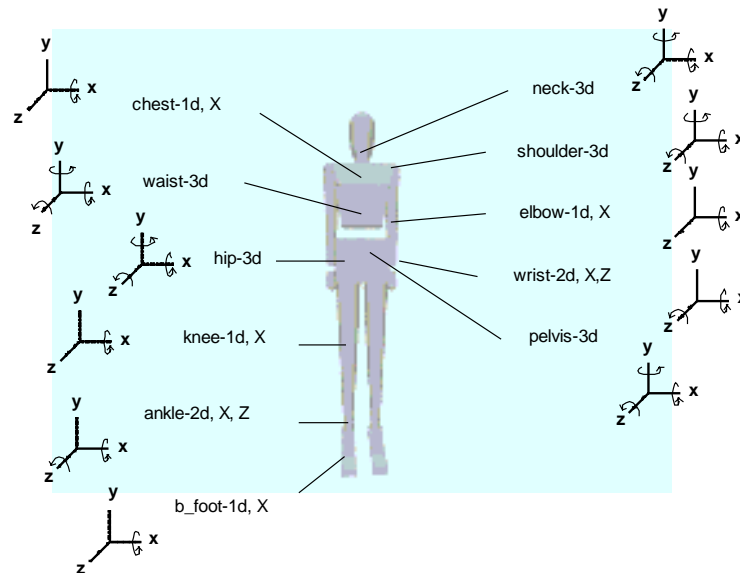


Figure 2: The controlled degrees of freedom of the human model. There are 18 body segments and a total of 36 controlled degrees of freedom.

The base state (neutral position) for our model is a standing position with the arms down. Its body hierarchy starts from the pelvis and branches down the legs, up the spine and to the head and the arms. The DOFs in our human figure are specified to capture the major ways in which the overall body moves, especially for the lower body segments. A diagram of the DOFs that are modeled in the articulated figure is shown in Figure 2. It possesses 18 joints with a total of 36 degrees of freedom.

It is difficult to say what the “major” degrees of freedom are for human walking, but some choices were more clearly defined. For example, the overall body has six degrees of freedom; three for spatial translation and three for rotation. The six DOFs are associated with the pelvis of the figure, which is the root in the hierarchical kinematic structure. For leg joints, from the toe to the hip, a 1-D hinge joint located at the toe is used to aid the modeling of foot activity on the supporting ground. The ankle allows primarily flexion/extension, but small amounts of abduction/adduction are also possible to handle locomotion events, such as turning and body lateral displacement during walking. A hinge joint is defined to model the flexion/extension activity of the knee. The hip joint is approximated by a hinge joint, but it actually has six DOFs to allow modeling the complex motions between the pelvis and the legs.

Although a complex upper-body has not been modeled, a number of degrees of freedom were included to approximate its motion. All joints at the upperbody are defined as hinge joints in the simulation. Three DOFs are included for the waist to model the motion of the trunk. Similarly, a three DOF neck joint is included to model the head movement. Three DOFs are required for the shoulder to allow the arm to rotate in any

direction with respect to the trunk. Finally, one degree of freedom is used to model flexion/extension of the elbow.

In order to define the functionality of each limb, our system has adopted the kinematic notation proposed by Denavit and Hartenberg (DH-notation [13]). This notation specifies the kinematics of each link relative to its neighbors by attaching a coordinate frame to each link. Four parameters, *the length of the link* (\mathbf{a}), *the distance between links* (\mathbf{d}), *the twist of the link* (α), and *the angle between links* (θ), are used to define the linear transformation matrix between adjacent coordinate systems attached to each joint. The transformation matrix that relates coordinate frame \mathbf{i} to frame $\mathbf{i-1}$ can be expressed as

$${}^{i-1}\mathbf{T}_i = \begin{bmatrix} \cos \theta_i & \sin \theta_i \cos \alpha_{i-1} & \sin \theta_i \sin \alpha_{i-1} & 0 \\ -\sin \theta_i & \cos \theta_i \cos \alpha_{i-1} & \cos \theta_i \sin \alpha_{i-1} & 0 \\ 0 & -\sin \alpha_{i-1} & \cos \alpha_{i-1} & 0 \\ a_{i-1} & -d_i \sin \alpha_{i-1} & d_i \cos \alpha_{i-1} & 1 \end{bmatrix} \quad (1)$$

And the configuration of each body segment (\mathbf{i}) in the articulated figure relative to ${}^0\mathbf{T}_i$ can be computed by

$${}^0\mathbf{T}_i = {}^0\mathbf{T}_1 {}^1\mathbf{T}_2 \dots {}^{i-1}\mathbf{T}_i \quad (2)$$

where \mathbf{T} is the transformation matrix which relates two coordinate frames.

3.4 Motion control hierarchy

Because of the hierarchical structure of the human figure, and its capability of expressing many complex movements, automatically generating human motions while allowing the user to specify certain movement characteristics is a challenging task for human motion control systems. Hierarchical motion control concepts are adopted in our modeling of human walking, because it provides the user a tool to balance the *automation vs. control* problem.

Ideally, A locomotion system should provide a reasonable configuration of the figure at any time as it moves along a desired path. To achieve this animation goal, intelligent stepping strategies, robust and efficient walking algorithms, and hierarchical motion control mechanisms are integrated into the system to allow the user to animate a variety of human walking in diverse environments interactively. The basic structure of this locomotion system is presented in Figure 3. From this figure, it can be seen that the motion control mechanism is hierarchical in nature. At the high level, the user can simply provide intuitive locomotion parameters, such as the body travelling path, and the desired speed along the path (e.g. “walk this route at speed x”), and the system will generate the “default” walking motion for these locomotion parameters automatically. At the middle level of control, locomotion attributes regarding the lower body, such as the weighting factors of leg joints, are provided to allow the user to animate a variety of gait characteristics. Finally, detailed movement instructions for each limb segment as a function of time are specified at the low level, so the user can animate the walking motion with various personalities and walking styles.

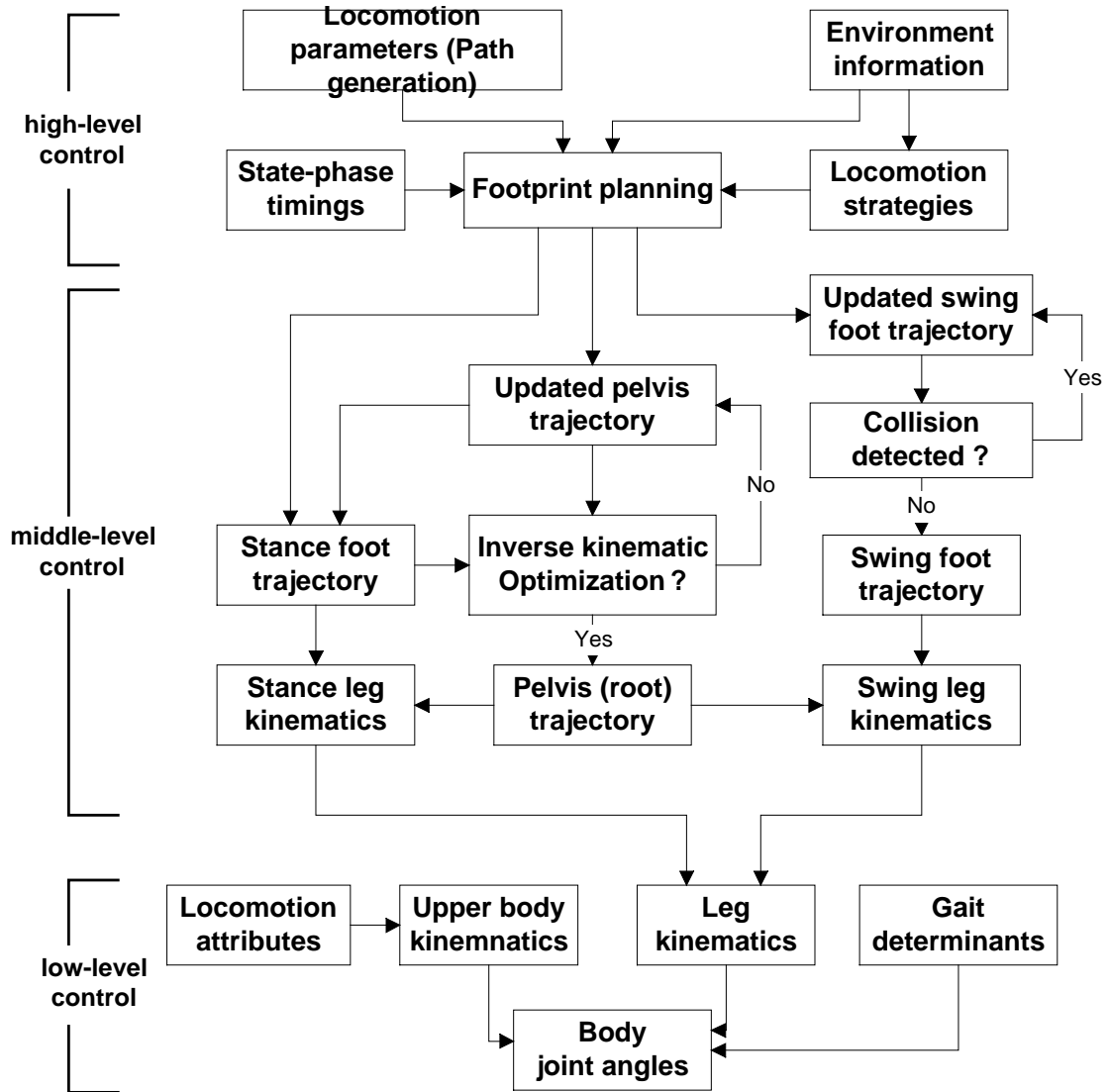


Figure 3. Locomotion system structure.

Chapter 4

Hierarchical Motion Control

Hierarchical motion control techniques have been widely used in computer animation for many years. Our motion control algorithms adopt this concept because it provides a convenient tool for a locomotion system to balance the important problems of control and automation. We think these techniques are well suited for controlling articulated figure motion, especially for structured or cyclic movements such as human walking and running.

“Hierarchical” at the high level of our locomotion control implies that the user will be able to simulate human walks with a small number of locomotion parameters. Ideally, these parameters should be simple and intuitive to the user, and be easily integrated into the task-oriented mechanisms. For example, given the desired walking speed and the traveling path (or dragging the virtual actor around the environment interactively), the system should be able to compute the 3-D path information and its corresponding locomotion strategies to generate walking motion automatically.

4.1 Locomotion parameters

Finding a safe path from a starting location to a destination in a certain environment represents a challenging task in several research fields. The path planning of walking figures is somewhat similar to the path planning of manipulators in robotics, of which many studies have been reported. However, implementing path planning for a walking figure is easier than it is for manipulators, because of the following aspects. First, it

doesn't have workspace limits in the horizontal direction. Second, the collision detection and avoidance is much simpler to implement, since no link other than the end-effector (the feet) may collide with the environment.

In the real world, when an obstacle is encountered in the path, we have two options: go around it by changing the walking direction, or go over it by modifying limb trajectories. For example, if the obstacle is too big to go over, the walker has to alter his walking direction to go around it. Research fields, such as robotics and artificial intelligence, have provided rich sources to solve this path-searching problem; however, this is beyond the scope of this study. Our system defines the traveling path by allowing the user to move the character around interactively. An alternative way of defining the traveling path is to let the user design the global path on the horizontal plane by specifying the piecewise cubic polynomial curves and their control points. While the path planning need only provide an approximate path for the virtual actor to follow, more considerations are put into the effort to solve the problems, such as a gait algorithm, collision avoidance, and computational time.

Walking speed serves as one of the most important factors in determining gait characteristics. Bruderlin and Calvert [6] showed that important gait determinants, such as step length and step frequency, can be related to the walking speed and the character's body height, as shown in the following equations.

$$\text{Step length} = \sqrt{0.004 \times V \times \text{body_height}} \quad (3)$$

$$V \text{ (m / min)} = \text{Step length (m)} \times \text{Step frequency (steps / min)} \quad (4)$$

Where V is the distance covered by the whole body in a given time;

Step length is the distance between successive foot-floor contact with opposite feet; and step frequency is the number of steps being taken in a given time.

Because equation (3) is based on body height and walking speed only, it can be applied to different characters with various walking speeds, and still produce reasonable results in general. For the purpose of various gait motions, the system allows the user to override these attributes arbitrarily. For example, in certain steps during the locomotion, we may extend (shorten) the step length to overcome obstacles along the traveling path.

An interesting issue unaddressed in [6] regarding the walking speed and gait characteristics is the adjustment of leg duty factor in the gait cycle. That is, the support phase of the gait should slightly extend as the speed of walking decreases, and reduce as the walking speed increases. From the human gait data we collected, at the customary 90 meter/minute rate of walking, the support and swing phases represent 60% and 40% of the gait cycle, respectively. Using the average body height of 1.75 meter, the system calculates the leg duty factor using the following equation.

$$Leg_duty_factor = 0.6 + 0.01 \times (90 - v) \times \sqrt{\frac{body_height}{v}} \quad (5)$$

Where v is the walking speed (m/min), and 90 represents the average walking speed.

4.2 Footprint planning and locomotion strategies

It is well known that stable biped gaits can be achieved by discrete foot placement [53]. To ensure correct and natural foot placement, planning the footprints at the right places along the traveling path is critical. In building general locomotion behavior, straight-path footstep planning concepts are utilized in our system. For example, equation (3) is used as the primary process for footstep planning within our 3-D locomotion mechanism. If the type of locomotion is anything other than linear locomotion, such as curved path locomotion, or on uneven terrain, further modifications of equation (3) are required to achieve the appropriate 3-D locomotion behavior.

On a flat, obstacle-free ground, a simple and effective way to arrange the next footprint is to advance the current footprint location by the step length computed from equation (3) along the direction of travel. However, an intelligent footprint planning mechanism with flexible step length is necessary for locomotion on uneven terrain. Based on this consideration, a non-uniform step length for each step is computed as a function of *direction change* along the path, *terrain status*, and *locomotion strategies*.

Direction change

The traveling path is the 2-D body trajectory over the horizontal ground plane. If we view a straight path locomotion algorithm as the planner for a 1-D system, we can modify it to suit the 2-D locomotion behavior for curve-path locomotion. Such a generalization from 1-D to 2-D is based on the intuition that there should be a smooth transition between linear and curved locomotion. Thus, curvature is the factor determining the similarity between the 1-D and 2-D locomotion behaviors.

A simple and convenient way to generate the travelling path is to use splines to model the path of the figure. While this approach works well for conventional animation models, it is not well suited for some applications, such as interactive games and virtual worlds, where the environment is unpredictable. To solve this problem, Our system has an interactive mode of motion control, which allows the user to drag (guide) the human figure to walk along a desired path, and generates completely autonomous motion on the fly.

When requested to generate the next step, the 1-D footprint-planning algorithm with uniform step length is applied first to get the information of the estimated *next_step*. The orientation of each footprint is calculated as the tangent vector along the traveling path at the footprint location. However, as the curvature of the travelling path increases, so will the rotation of the upper-body coordinate system, so as to follow the path. If the curvature in one step exceeds certain criteria (empirical studies have shown that 60 degrees per step is a good measure), not only the upper-body coordinate system, but also the entire skeleton will rotate to account for the high curvature of path. A common way to simulate this effect in locomotion animation is to introduce so-called “foot sliding”. That is, to allow the supporting foot to rotate toward the desired direction.

In the interactively guided walking of our system, the user can drag the human figure around the environment by clicking the pointer at the desired location. The resulting directional change is calculated as the difference between the current body’s orientation and the direction of the vector from body’s center to the “clicked” location. If the direction change is greater than 60 degrees, foot sliding, coupled with step planning, are both used to adapt to the significant direction change. The stepping strategy is

designed in attempt to minimize the needed step number in handling direction change, as shown by the following algorithm.

```
if direction-change < 45 degrees
    one step: no foot-sliding;
else if direction-change < 60 degrees
    one step: foot-sliding with (direction-change – 45) degrees;
else 1st step: foot-sliding for 15 degrees;
    if direction-change < 120 degrees
        if direction-change >105 degrees
            2nd step: foot-sliding with (direction-change – 105) degrees;
        else 2nd step: foot-sliding for 15 degrees;
    if direction-change > 165 degrees
        3rd step: foot-sliding with (direction-change – 165) degrees;
```

The application of foot sliding is used to lesson the large twist of the supporting hip joint. The computation of the body orientation should not be affected by the introduction of foot sliding. Thus, if a dramatic turn is required, it will take no more than three steps for the figure to turn into the desired direction specified by the user.

As shown in Figure 4, the direction-change for each step, α , is defined to be the angular displacement from current direction vector, \mathbf{L}_1 , to the estimated direction vector, \mathbf{L}_2 , of *next_step*. The computation of the current step-length based on direction change works as follows:

Dir. of estimated *Footprint* = tangent vector of the path at *Footprint* position

Dir. change (DC) = *Dir.* of next *Footprint* - *Dir.* of current *Footprint*

new *step length* = $f(\text{DC}) \times \text{equation (1)}$ $0 < f(\text{DC}) \leq 1$ (3)

The new step length is reduced as a function of direction-change magnitude, whereas the step duration is kept the same. The decrease of step length for the same step duration during direction change indicates that the average speed of progression is reduced, thus allowing for a safe adaptation of locomotor patterns during advance direction change.

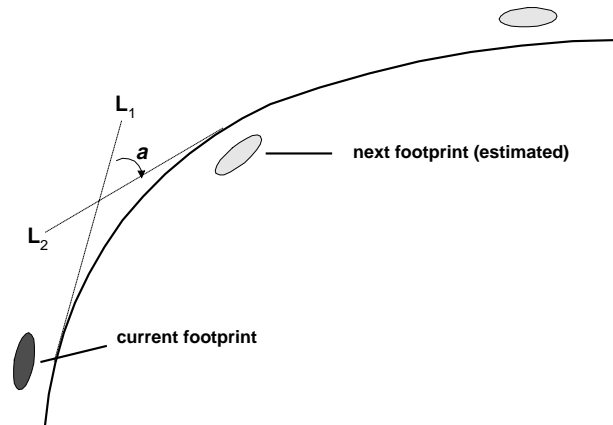


Figure 4. The direction change, a , in 2D step planning.

Once two consecutive footprints are placed, the orientational movement of the body in the horizontal plane can be determined. The intuitive way to compute body

orientational movement is using linear interpolation between two consecutive footprint orientations. The advantages of doing this are, first, when the body center passes the mid-stance position, the body center and the stance foot are at their natural standing orientations to each other. Second, the transition of orientation during the step and between steps is relatively smooth.

Since the entire process of step length adjustment is incremental at the step level, it fits neatly into the behavior simulation paradigm for human walking.

Terrain status

Environmental information provides 3-D space information of the environment. The traveling path is specified in the 2-D horizontal plane. Thus, it is necessary to map the path onto the world coordinate of the environment to get 3-D information along the path. Since usually no link other than the feet may collide with the environment for human walking, the 2-D to 3-D mapping is quite simple. The mapping process is discrete; instead of mapping the entire 2-D traveling path, each requested point in the 2-D path is checked and provided with 3-D information. Then, we can apply our locomotion strategies to plan the footprints along the traveling path, determine supporting foot trajectory, and search for an objective-optimized, collision-free trajectory of the swing leg for each gait.

Locomotion strategies

Step length adjustment represents the most important type of gait adaptation while traversing an uneven or cluttered environment. Given the 2-D traveling path, a new 3-D body trajectory, which includes the terrain status (height at a specified location), is computed by mapping the 2D traveling path onto the environment.

Experimental data show that the location of the obstacle has no effect on a human’s ability to go over the obstacle safely when the reaction time is given at least a two-step advanced notice [38]. Using this observation, with linear interpolation of the step lengths among consecutive steps, our footprint planning mechanism plans the footprints two steps ahead, and computes the step length of the current step, based on the step lengths of previous (known), current (estimated), and next (estimated) steps. The simplified step length adjustment algorithm works as follows:

```

set all steps’ step-length to unadjusted;

for step i :

if ( (obstacle_in_ step i+2) && (step i+1.adjusted == no))
{
    step i+1.adjusted = yes;

    step i+2.adjusted = yes;

    step_displacement = footprint i+2 desired - footprint i+2 default ;

    if (step_displacement < threshold)

        new_step_length = step_length + 0.5 x step_displacement;
    else

        new_step_length = step_length + 0.33 x step_displacement;

    reset footprint(i+1, new_step_length);

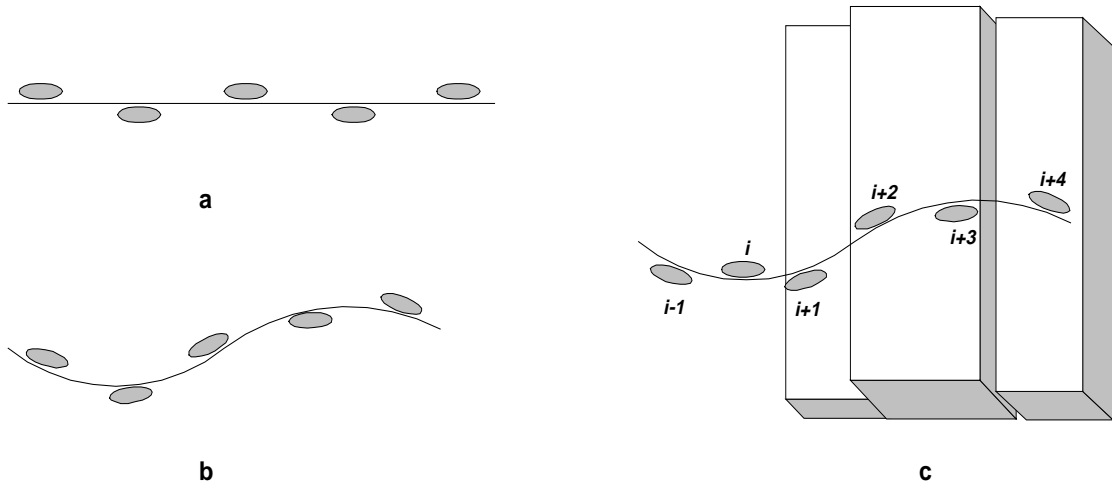
    reset footprint(i, new_step_length);

}

```

This intuitive, yet simple, footstep planning scheme works reasonably well in variant terrain as “readiness” for next step is prepared by including the estimated step length of

the next step, while “continuity” between consecutive steps is achieved by including both the predefined previous step length and the estimated step length. Figure 5 shows footprint planning along various 3-D body trajectories.



a and b: Four steps on flat terrain

c. Five steps on uneven terrain

Figure 5. Footprint planning along the 3-D body trajectory:
a. Straight free walking on even terrain with uniform step length.
b. Step length is slightly decreased in curved walking.
c. Applying flexible step length while encountering obstacles along the path.
For example, footprint i is retreated due to the obstacle encounter in step $i+1$.

Chapter 5

Modeling Coordinated Leg Motions

Modeling leg movements is the most challenging task in animating human locomotion. Simulating the leg motions over non-flat ground presents a difficult problem to the animator, due to the extra environmental constraints that need to be satisfied. Our modeling of leg movement is mainly based on constraint-optimization. First, the supporting foot constraints (footprints) generated by the footprint-planning algorithm need to be satisfied. This indicates that the states (place-on, stay flat, and lift-off) of the supporting foot needs to adapt to the terrain of the supporting ground. Second, constraints (collision-avoidance) imposed by the surrounding environment have to be satisfied. That is, the trajectory of the swing foot must be collision free. Finally, the internal constraint (pelvis) must be defined in such a way that it optimizes our simulation objectives, such as natural limb (joint) movements.

5.1 Stance foot trajectory

Stance foot trajectory represents one of the end-effector trajectories in our inverse kinematic mechanism. The support phase starts after the heel strikes the ground. For a small fraction of the support duration before “flat foot”, there is controlled plantar flexion at the ankle joint to lower the foot to the floor. Then, the foot stays flat on the ground for most of the support duration, followed by pushing off the ground at the end of the support phase.

Since state-phase timing is given, as is footprint location, the trajectory of the stance foot can be determined by two parameters: the initial-contact angle between the foot and the ground at the beginning of the support phase, and the lift-off angle at the end of the support phase. Linear interpolation scheme is used to compute the foot trajectory during non-flat-foot phases. The initial-contact and lift-off configurations of the stance foot during the support phase are illustrated at Figure 6.

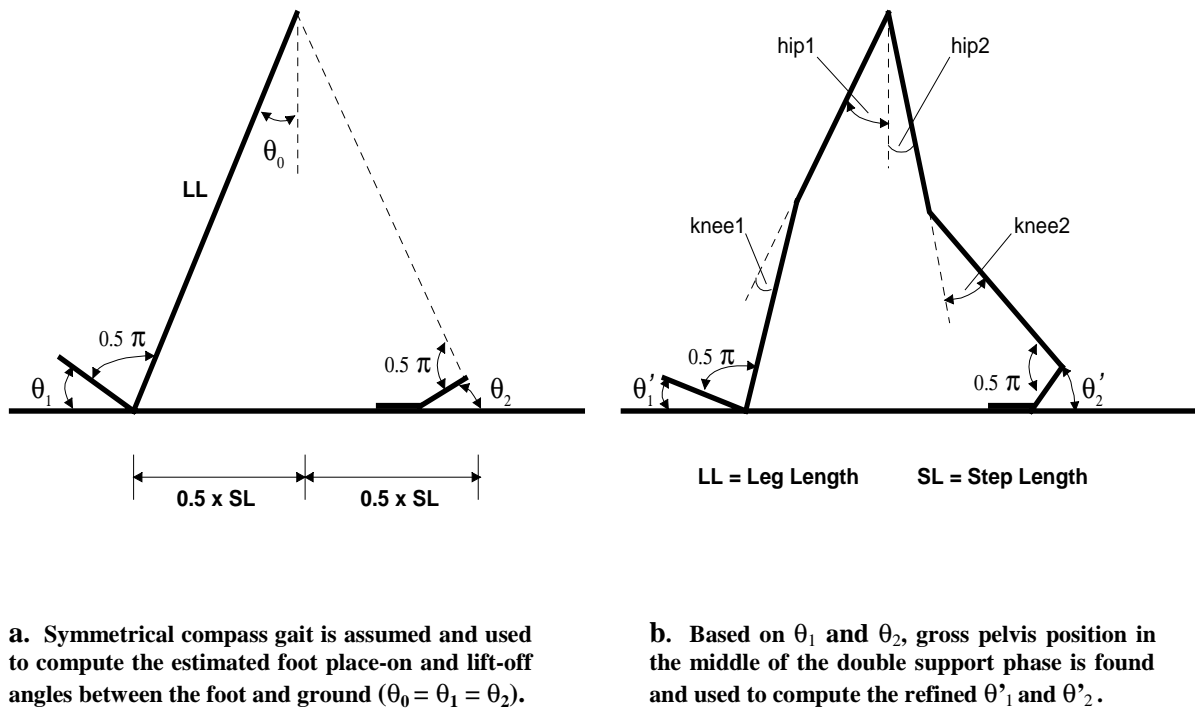


Figure 6. The two-stage process for computation of the foot initial-contact and lift-off angles.

Results from gait observation [25] have shown that the ankle joint is generally close to its neutral position in plantarflexion/dorsiflexion at the times of foot lift-off and

initial contact. This position is usually referred to as 90° . Research in gait mechanics [11] also suggests that the configuration of the supporting leg (full knee extension, and ankle at neutral) at the beginning of support phase provides optimum balance between step length and stable weight loading. Applying this theory to our computation of the initial-contact and lift-off angle parameters, we assume the ankle joint is at its natural rest configuration (90° is used as the rest joint angle for ankle. As an alternative, the user can arbitrarily adjust it.) when the foot starts to contact the ground for weight bearing, and lift-off from the ground for leg swinging. First, symmetrical compass gait is assumed and used in the computation of the preliminary foot place-on (θ_1) and foot lift-off (θ_2) angles. Using these preliminary angles, combined with the *pelvis-trajectory defining algorithm*, described in a later section, the gross pelvis position at the middle of the double support phase is found. And since the ankle positions at foot place-on and lift-off are already known, computation of the leg joint angles using inverse kinematics can be effectively done using a geometric approach. For example, as shown in Figure 7, the knee joint angle can be computed by applying the “law of cosines”:

$$\cos(\theta_k) = \frac{L_1^2 + L_2^2 - X^2 - Y^2}{2L_1L_2} \quad (6)$$

Since the knee joint angle can't be negative, we can ignore the other possible solution (the one with stripes in figure 7) computed from symmetry. To solve for the hip joint angle, equation 7 and 8 are used to computed the angles **a** and **b** as indicated in Figure 7.

$$a = A \tan 2(Y, X) \quad (7)$$

$$\cos(b) = \frac{X^2 + Y^2 + L_1^2 - L_2^2}{2L_1\sqrt{X^2 + Y^2}} \quad (8)$$

Then, the hip joint angle can be computed as “ $-90 + (a - b)$ ”, called its rest angle, which is 0 when the figure is in a natural standing position.

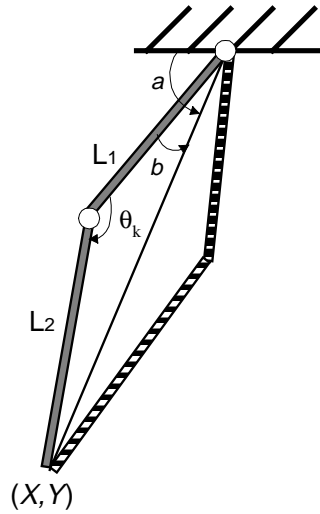


Figure 7. Plane geometry associated with a two-link planar leg

Finally, the angles at foot place-on (θ'_1) and lift-off (θ'_2) are computed from the configuration of the forefoot and the respective hip and knee joint angles. That is, finding the ankle joint angle which satisfies the constraint configuration of the forefoot. The following is the algorithm to compute these angle parameters:

Process *AnglesFootGround* (rest_angle)

{

$\theta_0 = \sin^{-1} (0.5 \times \mathbf{SL} / \mathbf{LL});$ // SL = step length, LL = extended leg length

```

ankle_at_place_on ( $\theta_0$ , ankle1);           // ankle1 = ankle position at foot place-on
ankle_at_lift_off ( $\theta_0$ , ankle2);         // ankle2 = ankle position at foot lift-off
pelvis_at_MDS (ankle1, ankle2, pelvis);    // pelvis position at middle of double support
inverse_kinematic(pelvis,ankle1, hip1, knee1); // I.K. to solve for hip and knee joints
inverse_kinematic(pelvis,ankle2, hip2, knee2);
 $\theta'_1 = (0.5 \times \pi - \text{rest\_angle}) - (\text{hip1} + \text{knee1})$ ; // set rest_angle =  $0.5 \times \pi$ 
 $\theta'_2 = (0.5 \times \pi - \text{rest\_angle}) - (\text{hip2} + \text{knee2})$ ; // user can set rest_angle arbitrarily
}

```

Once the two parameters, foot at place-on and lift-off angles, are determined, defining the trajectory of the stance foot is a relatively straightforward task. Linear interpolation is used in the time intervals from foot place-on to flat foot, and from flat foot to foot lift-off, to compute the angle between the supporting foot and the ground. In the case of “foot sliding”, the required sliding degree is linearly interpolated through the entire support phase. Knowing these angle parameters at given times with the footprint’s location, the trajectory of the stance foot is defined.

The supporting foot trajectory computation is entirely based on the footprints, which are adapted to the supporting ground. The pre-assumed symmetry compass gait algorithm is quite simple and effective in providing the gross (default) ankle positions for the inverse kinematic scheme, and in general, works well in different environments, such as flat, non-level, and sloping ground.

5.2 Swing-leg trajectory during the gait cycle

The swing-foot represents another end-effector in our modeling figure. The following questions arise: What criteria should be applied in the determination of a natural and expressive trajectory of the swing-foot and how do we formalize such expressive qualities in foot movement? From the studies of natural movements of intelligent animals, it is reasonable to assume that the following correspondences will also hold for human motion:

- Coordinated goal-directed (intentional) motion => minimum jerk about limbs, especially about end-effectors.
- Relaxed swinging motion => minimum energy expenditure.

Both correspondences can be weighted in determining the motion of the swing leg in human walking. For example, while minimum energy expenditure (relaxed swing motion) is preferred in walking on flat ground, coordinated goal-directed (collision avoidance) motion is more appropriate in uneven terrain walking. A Bézier curve (the detailed expression of this spline is given in appendix A) is applied to represent the desired trajectory of the swing foot. The 4 control points are given by the position \mathbf{P}_0 of the current swing ankle at foot lift-off, \mathbf{P}_1 and \mathbf{P}_2 which control the height displacement of the swing foot trajectory, and the ankle position \mathbf{P}_3 at the next heel strike point (Figure 8). In defining the trajectory of the swing foot, the main concerns we have are, first, the trajectory must be collision free. Second, the needed energy to follow along the trajectory should be minimized.

5.2.1 Collision avoidance

Terrain roughness represents one of the primary and difficult problems in simulating human walking in virtual environments. In real or simulated worlds, in general, a severe roughness can be treated as an obstacle to the body, and can be handled by appropriate path planning, and a mild roughness can be taken care of by elevating the leg to step on or over it.

Collision-free path planning has been studied in a variety of research fields, such as robotics, artificial intelligence, and computer animation. From the global perspective, a collision-free path for an articulated figure is the path along which the figure moves from an initial configuration to a final configuration, without colliding with any obstacle residing in the same space. From the local perspective, in general, no link other than the end-effector may collide with the environment. Thus, the swing-foot trajectory is the deciding factor of planning such a collision-free path for human locomotion. That is, if the swing leg can be raised to overcome a certain obstacle, the whole body should have no problem to overcome that obstacle.

During the normal swing phase, the leg that is not in contact with the ground should clear the ground by a reasonable and safe margin. If there is an obstacle presented in the path of the swing leg, depending on the location of the obstacle and the body posture at that time frame, the priority is clearly to provide adequate elevation of the foot to overcome that obstacle. For the purpose of collision avoidance, in addition to the Bézier curve used in this system, many other curve generation schemes are also available, such as cubic splines and B-splines. For collision detection, the positions of the toe end and heel are examined against the ground surface.

5.2.2 Least energy spent

Foot trajectories have been investigated in relation to energy expenditure [23, 62]. In general, free-swing limb trajectories tended to be chosen to minimize energy expenditure. This may explain why there is low toe-clearance of the swing foot for normal walking gaits; the higher the toe clearance, the more energy is needed. In an uneven terrain environment, if a surface obstacle is presented in the path of the swing leg, the most important factors we need to be concerned with are obstacle height and location, which reflect the unevenness of the terrain that is commonly encountered. A Bézier curve, which maintains a low safe margin above the terrain surface, is applied to represent the trajectory of the swing leg. The ankle of the swing leg at lift-off and place-on, which we have computed in the stance foot trajectory section, are set to be the starting and ending points of the Bézier curve, and the in-between control points of the curve are elevated from the ground until the whole curve safely passes above the surface without collision (Figure 8).

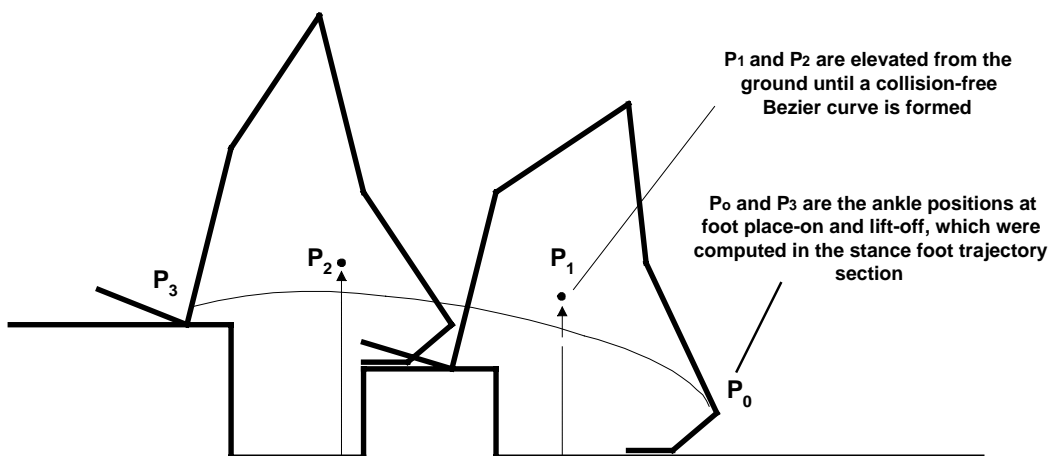


Figure 8. A Bézier curve to represent trajectory of the swing foot. Minimal toe clearance is maintained above upraised obstacle.

Not surprisingly, the results show that the Bézier curve defined by maintaining a small margin over the obstacles also optimizes the objective function expressed in expression (10). We think the possible reasons for this are:

- Maintaining minimal toe clearance while keeping collision-free, in general, produces a “flat” Bézier curve. Also, the “smoothness” characteristic of the Bézier curves results in the smoothness of joint-angle movement.
- Raising the leg more than what is necessary, from an inverse kinematic point of view, causes more flexion in the hip and knee joints, which is unfavorable to our optimization objective (equation 4). Experimental data from gait observation also shows that raising the leg entails an additional energy cost.

Among the many curve generation schemes, such as the cubic spline and B-spline, the Bézier curve is chosen to define the swing foot trajectory mainly because the curve never oscillates widely away from the designed points and unexpected results seldom occur. The cubic spline is simple, but lacks the ability to control the shape of the curve, which is essential in obstacle avoidance. On the contrary, the B-spline curves provide more control in modifying the shape of the curve, but are computationally expensive and overqualified. However, the Bézier curve has its own limitation; it is impossible to make any local change in the curve without affecting the entire curve. Fortunately, from experimental results, this limitation causes little or no problem in our applications, even for walking on relatively rough terrain.

5.2.3 Foot movement along the swing-foot trajectory

Despite the geometric shape of the swing-foot trajectory, the Bézier curve itself doesn't provide much information about foot movement along the curve. Thus, further regulation of this curve is required to improve the walking model.

From human gait observation, The speed of swing-foot's movement generally varies a little during swing phase, being fastest during the middle and slowest in the beginning and termination of the swing phase. From captured motion data we collected (human gait lab, National Institute of Health, NIH), we found that the "foot" moves in this pattern not only in normal gait, but also in stair climbing, descending, and obstacle-overcome gaits (appendix B). These observations indicate there is a generic *displacement vs. time* pattern of swing-foot motion for different gaits.

To simulate the same pattern of pelvis movement during the gait cycle, a non-uniform time-space sampling approach is applied to the Bézier curve. The parameter t of the Bézier curve is computed as:

$$t_{new} = \frac{\int_0^{t_{sampling}} f(t)dt}{\int_0^1 f(t)dt} \quad (9)$$

where $f(t)$ is the "generic" motion-captured swing-foot trajectory during the gait cycle. Setting the parameter t 's range from 0 to 1, $\int_0^1 f(t)dt$ is the whole swing-foot trajectory, while $\int_0^{t_{sampling}} f(t)dt$ represents the piece of the trajectory that ends at $t = t_{sampling}$. Thus, the parameter t of the Bézier curve is adjusted and normalized, using the swing-foot motion pattern.

5.2.4 Coordinate synchronization

The modeling of leg motion starts with footprint planning. Footprints are planned two steps ahead of the current step, based on step length and environmental information. Therefore, coordinates in relation to the environment (world-coordinate) is the natural choice to define the foot trajectories in. The supporting foot trajectory is designed to adapt to the terrain surface at the footprint. Leg configurations at step switching, and foot timing events (heel touch-down, flat foot, and foot lift-off) are used to compute the trajectory. Based on the supporting foot model, the swing-foot trajectory is then defined as a Bézier curve, which starts from the foot lift-off at footprint # n , and terminates at the heel touch-down at footprint # $n+2$, with the in-between control points to elevate the foot from colliding with any surface obstacles and to minimize energy expenditure.

Both the supporting and swing foot are treated as end-effectors of our articulated human figure. The next task in simulating the leg motion is to define the trajectory of the root, which will then be used to synchronize the movements of its end-effectors based on our simulation goal, smooth body center and leg joint movements. Like many human animation systems choosing the body center as the root of the animated figure, our system uses the pelvis, and treats it as the body center (root) of the human model in our simulations. Both foot trajectories are then mapped to the coordinate system with reference to the pelvis, to make sure that both feet (end-effectors) are reachable from the pelvis for the legs.

Chapter 6

Motion Control of the Body Center

As motion control goes deeper in the motion control hierarchy, the system relies increasingly on internal knowledge about particular movements in order to automate the motion generation. Some of the internal knowledge about movement is provided through user's preference, thus more control is given to the user to animate desired gait characteristics. Furthermore, additional animation attributes that control the lower body, such as the weighting factor of leg joints and neutral angles of ankle joints at heel strike and foot lift-off, are provided to allow the user to animate a variety of gait characteristics.

Given the hierarchical structure of the human model, modeling the “root” movement of the articulated structure is the most challenging task in simulating human locomotion. Previous work, such as Girard [18] and Bruderlin & Calvert [6], has proposed different approaches for defining the movement of the body center during locomotion. However, their models were designed for flat-ground locomotion, and can not be applied to uneven ground without major modification of their control algorithms. In this study, we present a new approach, which uses the pelvis as the root of our human model, to define the movement of the body center during walking.

6.1 Characteristics of the pelvis trajectory in walking

Walking is a smooth, highly coordinated, rhythmical movement by which the body moves step by step in a required direction. As the footprints are placed, a smooth pelvis trajectory in a walking gait can be defined by a function of the support-leg profile; in the

double support phase, both legs are quite far apart, and this will cause a lowering of the center-of-mass. In the mid-stance phase, the center-of-mass will be displaced vertically but, by having a slightly flexed knee joint during this phase, the amount of vertical displacement is reduced. Human gait observations have shown that the shape of the human pelvis trajectory in walking is similar to a smooth sinusoidal curve as shown in Figure 9.

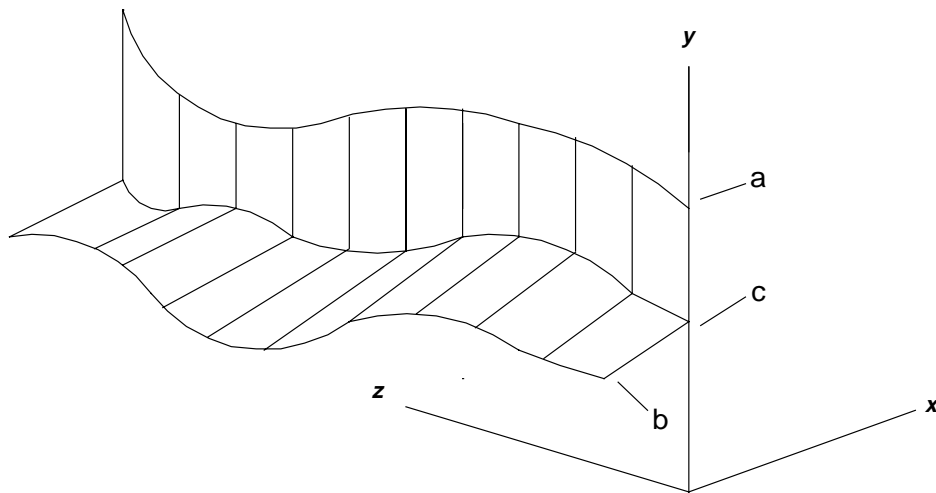


Figure 9. Displacements of pelvis in three planes of space.
a. Lateral displacement in the horizontal plane;
b. Vertical displacement in the sagittal plane;
c. *a* and *b* are projected and combined to form *c* as the 3-D pelvis trajectory.

In the sagittal plane (projection of the 3-D curve onto the YZ vertical plane), the pelvis reaches the peaks at about the middle of the stance phase of the supporting leg, and falls to the troughs during the middle of the double support interval, when both feet are in

contact with the ground. From a geometric point of view, cubic splines should be able to define the shape of the pelvis trajectory curve with minimal control points. Thus, based on the following considerations, we chose a Bézier curve to model the pelvis trajectory of human walking:

- The two end points, which the Bézier curve will pass through, are well suited to represent the vertical maximum and minimum of the sinusoidal pelvis curve.
- First-order continuity between adjacent segments is achieved by imposing the constraint that the third control point from previous segment and the second control point from the current segment be collinear.
- Due to the pelvis local activity, varied horizontal velocities of the pelvis during the gait cycle can be simulated by adjusting the in-between control points.
- It is easy to reshape the Bézier curve by adjusting one or two control points, and the computational requirements remain reasonable.

Using Bézier curves to represent the pelvic motion curve during walking, we have to position the control points properly in order to mimic natural pelvic movement observed in human walking.

6.2 Pelvic movement in the sagittal plane

In normal walking, pelvic movement is well studied. Its trajectory displays a similar pattern of a smooth sinusoidal curve along the line of progression, with little variance in the sagittal plane. For walking in virtual environments, because of the variable terrain

along the travelling path, pelvic movement in the sagittal plane can be quite lively and difficult to define using current simulation tools for human locomotion.

6.2.1 Summit at mid-stance

During the gait cycle, the pelvis passes through its vertical maximum at the middle of the leg supporting duration, also known as mid-stance (**MS**). Since the supporting foot stays flat on the ground and the ankle joint is at its neutral standing position at this moment, knee joint flexion becomes the deciding factor to compute the location of the pelvis at mid-stance. For example, given a highly extended knee joint, the summit of the pelvis trajectory is raised, as is the vertical displacement, resulting in a bouncy gait. Similarly, large flexion of the knee joint reduces the magnitude of the vertical displacement of the pelvis trajectory, and generates a smoother pelvic curve, which in general is preferred for normal walking.

6.2.2 Valley at the middle of double support

The pelvis passes through its vertical minimum at the middle of the double support phase (**MDS**). The timing of the **MDS** is right in the middle of two consecutive **MSs**, but the pelvis location at **MDS** could be anywhere in-between the two **MSs** as shown in Figure 10. Since each possible location of the pelvis at **MDS** will define a different motion curve for the pelvis in our trajectory-searching algorithm, given the pelvis location at **MDS**, we use an optimization approach, which tries to minimize the weighted sum of angular

jerkiness of supporting leg joints (expressed as the mean square of the rate change of angular acceleration) during the support phase, to evaluate the resulting curve:

$$\nabla \left[\sum_{i=1}^{i=3} \mathbf{W}_i \int_{t_{start_sup\ port}}^{t_{mid_sup\ port}} (d^3 f_i(t) dt^3)^2 dt + \sum_{j=1}^{j=3} \mathbf{W}_j \int_{t_{mid_sup\ port}}^{t_{end_sup\ port}} (d^3 f_j(t) dt^3)^2 dt \right] \quad (10)$$

Where \mathbf{W}_i is the weighting factor of joint i ;

$f_i(t)$ is the i th joint angle computed from inverse kinematic algorithm at time t .

The reasons behind using expression (10) as our optimization function for evaluating the candidate pelvic motion curves are:

1. The Supporting leg is the major limb that advances and balances the whole body during locomotion. Minimizing the jerkiness of the supporting leg joints, in general, relieving the leg muscle from sudden bursts of activity which consumes much energy, plays an important role in human locomotion.
2. Given the configurations of the end-effectors, the inverse kinematic algorithm computes each joint angle of the articulated chain automatically. Using equation (10) will eliminate candidate curves that produce jerky joint motions in consecutive frames.
3. Using a weighting factor for each joint gives the animator the flexibility to produce a variety of walking styles. Assigning different weighting factors for the leg joints will determine the “value” of the evaluated pelvic trajectory, as well as the angular movement of the leg joints. The effect of different weighting factors on the leg joint is illustrated in Appendix C.

For the supporting leg, since the foot (end-effector) trajectory is known, given the pelvic (root) trajectory during the support phase, we have both root and end-effector trajectories of the supporting limb chain. The inverse kinematics algorithm is then applied to compute the in-between joint angles of the supporting leg during the leg-support phase. In the objective function, the weighting factors define the relative importance and contribution of a given joint. Equivalent weighting factors for the leg joints are applied as the default weighting factors in the objective function. As an alternative, the user can arbitrarily adjust these weighting factors to generate various gait characteristics.

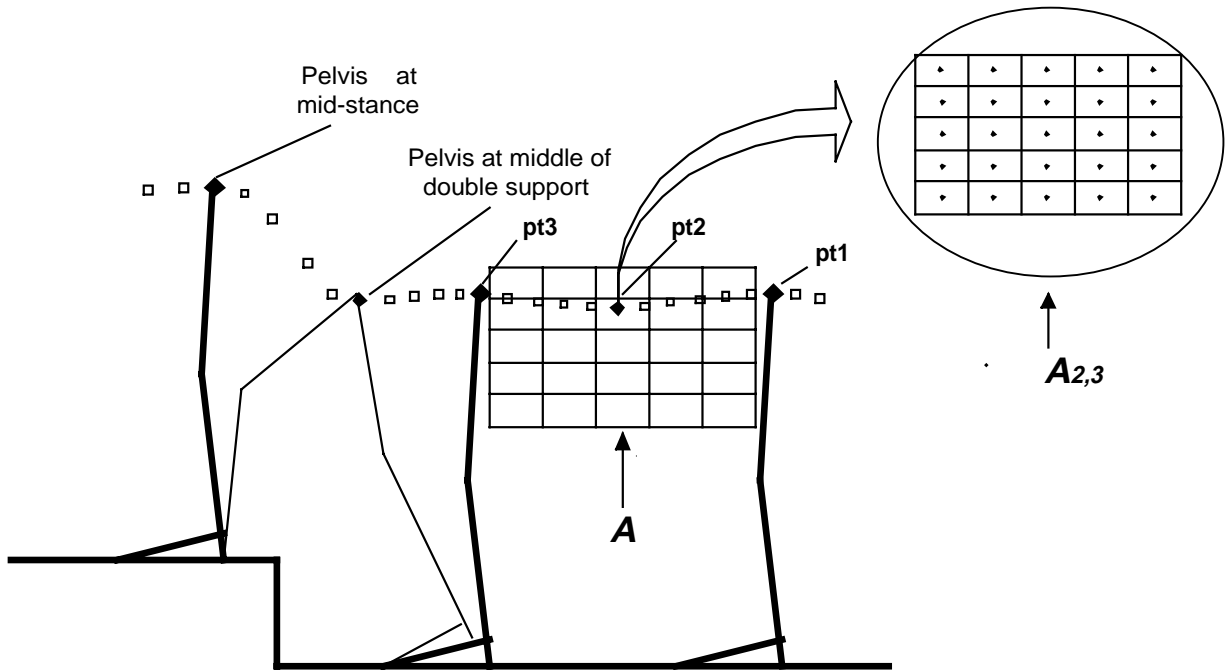


Figure 10. Pelvis location at consecutive MSs. An optimal objective function is used to evaluate each cell of the 5x5 grid area. Then, the qualified cell is further subdivided into another 5x5 grid, and evaluation is applied again to find the best pelvis location at MDS.

As shown in figure 10, the pelvis location at **MDS** (pt2) is somewhere in area **A** between pelvis locations at **MSs** (pt1 and pt3). First, area **A** is subdivided into a 5x5 grid. The centroid of each cell is then used to form a curve to be evaluated by our objective function (equation 10). Then, the grid with the best-scored curve is further subdivided into another 5x5 grid. Like the previous process, each centroid of the new subdivided grid is used to form a curve to be evaluated by the same objective function. The best-scored centroid of the second-level grid is then selected to be the pelvis location at **MDS**, and forms the pelvic (root) curve in our inverse kinematics mechanism. By using this two-level spatial subdividing approach, area **A** is subdivided into 625 (25x25) sub-areas. However, instead of evaluating all 625 curves, less than 1/12 ((25+25) / 625) of them (and most of these 50 curves can be eliminated at a very early stage of evaluation, due to being out of working space of the articulated chain) are actually evaluated to find the most satisfying curve. The algorithm to search for the pelvis location at **MDS** works as follows:

```

at each level of subdividing // 2 levels, each is a 5x5 grid
for ( i=1; i ≤ 25; i ++ ) {
    build the pelvis curve;
    t = 0;
    while ( t ≤ 1.0 ) {
        t = t + Δt;
        find heel locations at t by linear interpolation;
        find pelvis locations at t by spline interpolation;
    }
}

```

```

        inverse kinematics to get in-between joint angles;
        score(i) = score(i) + equation (10);
    }
    if ( score(i) < min_score )
        min_score = score(i);
    }
    return (i);

```

6.2.3 Exception handling of the pelvis searching algorithm

Using the leg configuration information and the space-subdivision searching algorithm for pelvis locations at specific times during the gait cycle, a pelvis trajectory (a two-segment Bézier curve) can be formed to produce smooth pelvis and leg-joint movements. However, when the roughness of the terrain increases to a certain level, the Bézier curve which satisfies all the constraints may not exist for the searching algorithm, as some of the sampling point(s) interpolated from the evaluated curve may be unreachable from the foot constraint point for the supporting leg. Our solution to this problem is to assign a relatively large value for equation (10) for the sampling point which is out of the workspace of the foot constraint point, and “force” the searching algorithms to determine the “optimized” pelvis location at MDS.

The potential problem of forcing the searching algorithm to choose the pelvis location at MDS is that during the simulation, there might exist moment(s) (image frames) that the inverse kinematic algorithm will be unable to compute the proper leg configuration. A solution to this problem is to have the leg with fully extended knee joint

pointed from the foot constraint point to the pelvis location interpolated from the Bézier curve. An alternative way is to interpolate the leg joint angles from its neighbor frames which are inverse-kinematically solvable. Either way, the pelvis location derived from the leg configuration (kinematic solution from the foot constraint point) will deviate from the one that is interpolated from the Bézier curve. Our strategy is to adopt the approach with the least weighted spatial displacement of the two locations.

6.2.4 Building the pelvis curve

After defining the pelvis location at the middle of double-support and single-support (the end points of the Bézier curve, P_0 and P_3), determination of the two in-between points represents the next challenging task in building the pelvis curve.

Initially, the control points (P_1 and P_2) are positioned in-between the two end points (P_0 and P_3) so that P_0 through P_3 are equal-distantly spaced along the axis of progression. Then, to achieve first-order continuity between adjacent segments of Bézier curves, P_1 and P_2 are vertically adjusted to ensure that the third control point of the first segment, the shared end point, and the second control point of the second segment are collinear.

A smooth Bézier curve can be formed using the control points derived from the above scheme. However, an interesting observation from human gait analysis has shown that in normal walking, the pelvis moves forward with a small variance in speed throughout the gait cycle, being slowest during the middle of the double-support and single-support phases [25]. Also from the captured motion data we collected (Biomechanics Laboratory, National Institutes of Health, Bethesda, MD), we found this

pelvis movement pattern to be rather generic: A similar pattern is also shown in stair climbing, descending, and obstacle-overcome gaits. To simulate this “pattern”, our approach is to slightly align the in-between control points horizontally toward their respective adjacent end points. By doing so, the advantage is two fold. First, the simulated pelvis movement along the Bézier curve matches closer to the one gathered from gait observation. Second, the discontinuity between the segments, if it exists, can be reduced (or made imperceptible to the viewer).

6.3 Pelvis movement on the transverse plane

From human gait observation of normal straight walking, by projecting the body center onto the horizontal plane, the body center passes forward along the medial borders of the feet, oscillating some to either side of the line of progression. The horizontal kinematics of the body center reveals that balance on the transverse plane is regulated from the middle of double-support to the next middle of double-support, with the peak lateral excursion occurring during the middle of single-support.

To simulate curved walking, our system models the angular motion of the body center to rotations about the yaw (turning) and roll (banking) axes. Since the orientation of the body center is close to the footprint’s orientation at the middle of single-support, linear interpolation between two successive footprint angles, in general, gives a good result in determining the angular movement of the body center about the yaw axis, with its rotation at the mean of the angles of the feet during the middle of double-support.

For banking effects, consider the human figure moving at a speed s along a circle of radius r . The centrifugal force, by Newton’s equations, will be a vector perpendicular

to the circle having magnitude ms^2/r . Since the system allows the user to interactively drag the human figure in the desired direction, we generalize the radius r of the curved path by the direction-difference between steps and the step-length

$$r = \frac{1}{\sin(\text{direction_difference})} \times \text{step_length}$$

Therefore, the banking angle (Figure 11) can be computed as

$$\theta = A \tan\left(\frac{s^2}{rg}\right)$$

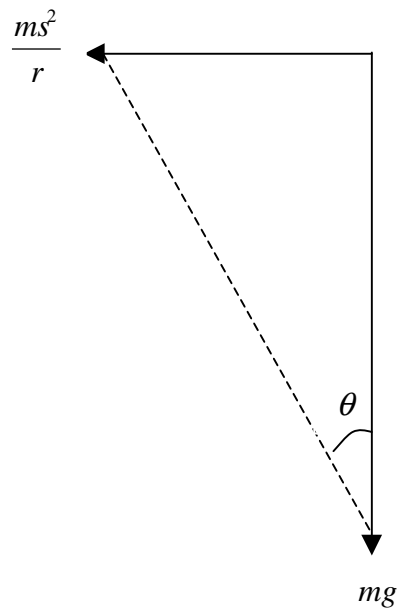


Figure 11. The banking angle is computed in terms of the speed of walking, the path curvature, and the gravitational acceleration.

Chapter 7

Low level Motion Control

One of the advantages of our motion control mechanism is that it allows the user to fine-tune the appearance of the motion at the final stage of motion control process. At this stage, since the major joint configurations of the legs are known, as is the root configuration of the articulated body structure, it is possible to fine-tune the other locomotion attributes to achieve different motions without violating kinematic constraints, such as foot penetration or sliding on the ground, or colliding with objects. Among these adjustable animation attributes, gait determinants are the most important factors affecting locomotion.

7.1 Gait determinants

Gait determinants in our system include three activities of the pelvis during the gait cycle.

They are **pelvic rotation**, **pelvic tilt**, and **lateral pelvic displacement**:

- **Pelvic rotation:** The pelvis rotates about the body torso alternatively to the left and to the right, relative to the line of progression. Pelvic rotation reaches its maximum at heel-strike which further flattens the pelvis trajectory, and at minimum at mid-stance, where both hips are aligned horizontally. Saunders *et al.* [48] have quoted $\pm 3^\circ$ for the amplitude in a normal walking gait. Our system adopts Saunders' numbers and further extends the range of rotation to accommodate the speed of walking; the faster the speed, the larger the range ($\pm 3^\circ \sim \pm 6^\circ$).

- **Pelvic tilt:** In a normal walking gait, due to pelvic tilt ($\pm 5^\circ$ about the walking direction axis), the hip of the swing leg falls slightly below the hip of the supporting leg. Pelvic tilt reaches its maximum at the end of double support, where the hip of the swing leg drops below the supporting leg hip, and its minimum at heel-strike, where both hips are aligned vertically. Introducing pelvic tilt needs to be carefully implemented, because it creates the possibility that the swing foot may penetrate the ground surface. A natural way to avoid this undesired effect is to introduce larger knee flexion of the swing leg to compensate the pelvic tilt.
- **Lateral body displacement:** Normal walking involves displacement of the pelvis from side to side. This is caused by the fact that the body always shifts slightly over the weight-bearing leg. For each stride, the maximal lateral displacement (to the supporting side) reaches its maximum halfway through the support phase at an amplitude of approximately 4 to 5 cm, and its minimum (no displacement) at heel-strike. In general, the magnitude of lateral pelvic displacement is a function of stride width and velocity; greater stride width causes more displacement, and faster locomotion decreases displacement.

All these default maximum and minimum values for gait determinants can be controlled interactively as locomotion attributes. And from the extreme values of these determinants, the configuration of the pelvis is “adjusted” by linear interpolation. To ensure the constraints between the stance foot and supporting ground are satisfied, the articulated structure of the human figure is slightly modified; a sub-set of the supporting leg is set independent from the whole body. The order of the chain structure of the supporting leg is reversed; changing from hip-to-toe to toe-to-hip, and the hinge-joint

constraint between the hip and pelvis is broken loose to allow further manipulating of the pelvis.

7.2 Motion control of the upper body

Although it is possible to walk with little movement of the trunk and no movement in the upper limbs, the walks in these situations look tired and awkward. Whereas the patterns of movement in the lower limbs are similar between individuals, they are more varied in the upper limbs. Even for the same individual, the range of movement in the upper limbs will vary according to the velocity of walking, or walking activities, such as walking on ascent and descent [25]. This is not surprising, as the movements of the upper limbs are likely to be affected by the movements of the lower limbs and rotation of the trunk [37]. First, the arm swing may impart momentum through the trunk to the lower limbs. Second, the arm swing may be used to correct over-rotation at the lower spine.

7.2.1 Motion of the torso and head

From a kinematic point of view, the locomotion attributes for the upper body, in general, will not affect the function of the lower limbs. All of these attributes are initially set to default values and can be adjusted interactively to individualize walks, that is, to produce walks with the same step length, step frequency and velocity. Several degrees of freedom of the upper body are animated. List compensation of the torso is implemented to keep the “body-center” close to the supporting base of the stance foot, in reference to the line of progression. Torso tilt is interpolated between maximum forward and backward lean.

The forward and backward lean values are offset to adapt the height difference among the steps. For example, if the figure is walking upward, the lean values are offset “forward” toward the progression direction. Similarly, in descent walking, they are offset “backward” in the opposite direction.

We try to simulate head movement according to the interested line-of-sight of the figure. For example, the head is kept pointing toward the marching direction, thus attributes from the chest segment and neck joint are both used to compensate the pelvis and torso actions. To simulate head motion in the sagittal plane, the degree of freedom on the sagittal plane of the neck joint is animated to keep the head up-straight during normal walking, and lean it down when walking on uneven terrain.

7.2.2 Motion of the arms

There are 6 attributes for varying the movements of the arms. Three for the shoulder: shoulder rotation, shoulder swing (sagittal plane), arm out (coronal plane), one for elbow flexion, and two for the wrist: wrist list (sagittal plane), and wrist flexion (coronal plane). The swing of the arms provides a purposeful counter-force to minimize the rotatory displacement of the body by the locomotor mechanics of the legs. Timing between the two arms is a 50% offset in the gait cycle, with the maximal displacement in either direction occurring at the beginning of the double-support phase.

In walking, the major activities of arm swing occur at shoulder swing and elbow flexion. In normal free walking, the shoulder flexes about 24° by the end of single-support, then it slowly extends throughout the swing phase. The elbow flex moves in the same direction as the shoulder swing, and has an almost equivalent arc (30°) of flexion

and extension swing during each stride. Figure 12 shows the rotational motions of shoulder and elbow joints during normal walking.

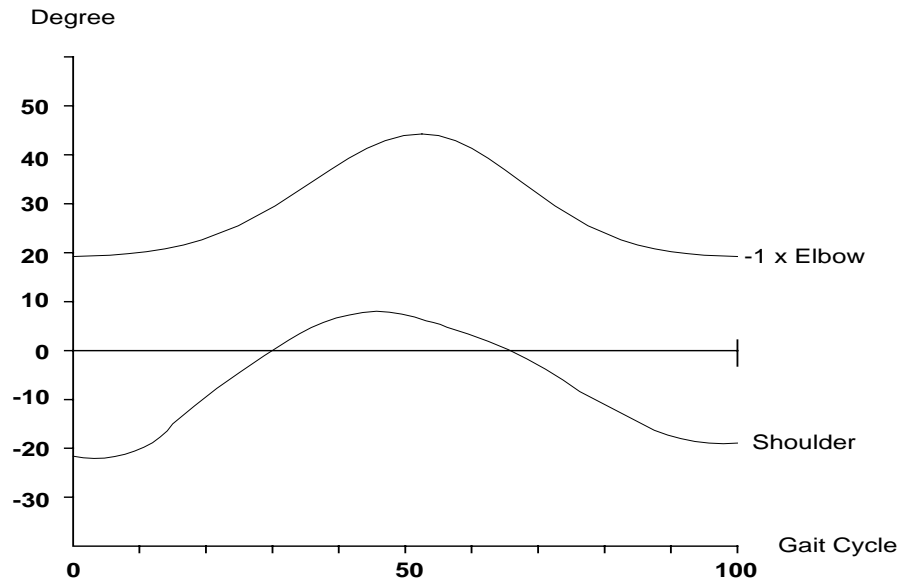


Figure 12. Shoulder and elbow motions during walking. Horizontal scale indicates % gait cycle beginning with double-support.

Since locomotion is a cyclic activity of recurring patterns with a basic unit of one locomotion stride or cycle consisting of two symmetric steps, interpolation between extreme values of the attributes is capable of generating convincing results. Cubic spline interpolation is used to compute each attribute of the arms. The ranges of arm attributes will automatically adjust and adapt to the walking speed and terrain status. As the speed of walking increases, so do the ranges of the arm attributes. In walking on uneven terrain, ascent walking will increase ranges, while descent walking will reduce them.

7.3 Personalized human locomotion

Although the main effort of this study is to provide an animation scheme with regard to leg motions of human walking in virtual environments, a variety of walks with different personalities and moods can still be obtained by modifying the basic motion generated from the algorithms. One of the advantages of our motion control mechanism is that after the motion control passes to the lowest level, an instance of particular walking motion is generated, and this leaves the animator the option to further direct a desired motion by altering the parameters.

7.3.1 Interface

A system called Vwalker has been implemented in Visual C++ according to the motion control principles described in the previous sections. An illustration of the interface is given in Figure 13. All of the locomotion parameters and attributes are initially set to default values, and can be adjusted interactively while the motion of the human walking is displayed on the screen. Although many more parameters are conceivable to further personalize the locomotion, study has indicated that too many variables lead to confusion and make it difficult for the user to predict the outcome. Our solution to this problem is mainly based on the motion control hierarchy concept. That is, the system allows the user to specify an instance of a particular locomotion by adjusting the parameters that are used in the control algorithm. Then, after locking down the parameters which are determinants of the selected stride, the system allows the user to further specify the desired motion at

the lower control levels. Thus, while giving more control to the user, constraint satisfaction is still ensured.

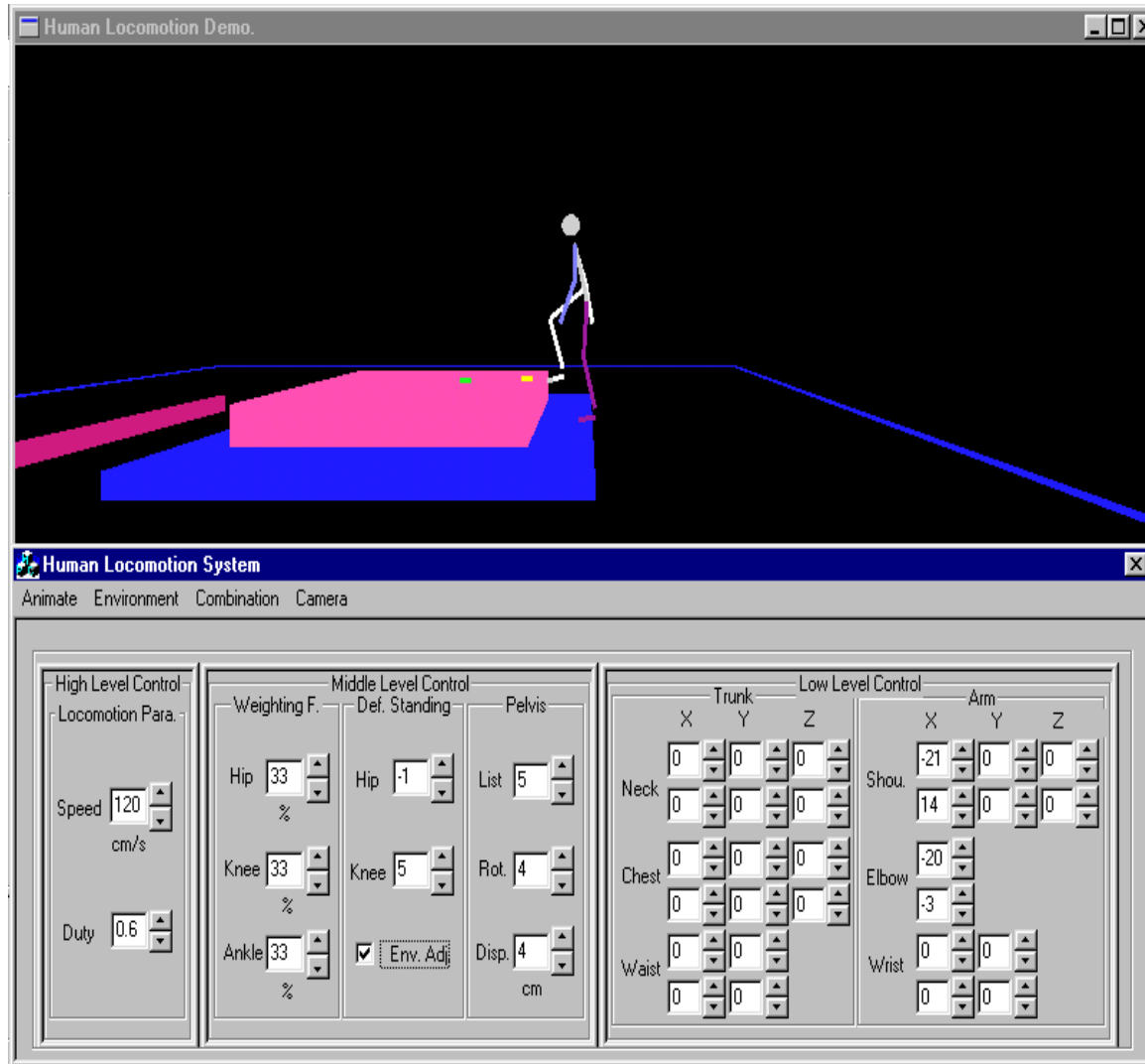


Figure 13. Motion control interface for Vwalker.

7.3.2 Adjusting control parameters

The controlling parameters are the desired angles and positions for the joints of the animated human figure. At the lowest control level, the user can easily adjust the control parameters, except for those locked by the system, (changing them may result in certain constraint violations, such as the feet penetrating the floor). generated from the algorithms, and watch the resulting motion immediately. Various walks can be animated by tweaking one or more of the control parameters. For example, the supporting knee joint angle at mid-stance will affect the bounciness of the walk, while lateral displacement of the pelvis during the support phase can be adjusted to simulate walks of different characteristics, if we scale down the default lateral displacement, the resulting motion looks more stiff and robotic. Similarly, by scaling up the value, more “life-like” characteristics are added to the walk. Giving another example, some parameters can be tweaked to simulate certain walking styles; a “proud” or “happy” looking walk can be obtained by adjusting arm swing and shoulder rotation to their maximum, while setting torso and head tilt to their minimum. A “tired” or “frustrated” looking walk is achieved by tweaking the same parameters in the opposite direction.

Chapter 8

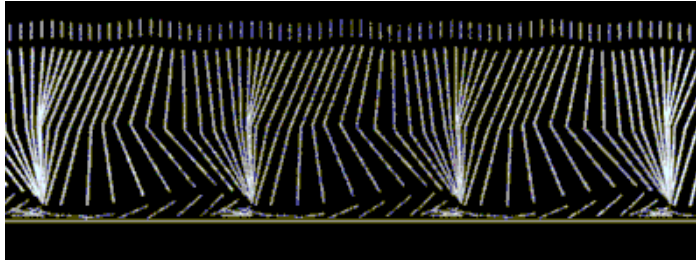
Results

We have successfully applied our motion control algorithms for animating human walking. Using Visual C++, a system called VWALKER was built to simulate human walking in different environments. The results show that this walking model allows animators to specify desired walking styles in various environments interactively, and represents an important initial step toward meeting the locomotion requirements in virtual environment applications. First, our approach is broadly capable; stepping strategies observed in human gait behaviors and constraint optimization approaches are integrated into the motion control mechanism to simulate walking in different environments. Second, it is responsive. Since relatively simple inverse kinematic mechanisms and optimal search algorithms are widely used in the computation, interactivity can be easily achieved. Finally, a variety of walking styles and personalities can be simulated through the motion control hierarchy. More control over the resulting motions is given to the animator as we move down the control hierarchy.

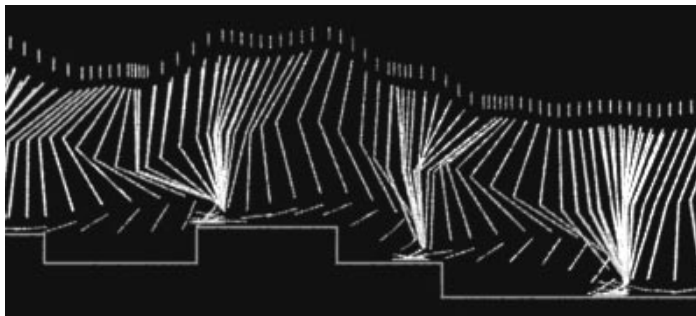
Functionality

Besides normal walking on flat ground, the motion control algorithms are capable of animating human walking on uneven terrain. The two-step lookahead footstep planning approach plans the footprints to adapt to a variety of environments, and frees the user from the laboring task of detailed foot placement, which is critical in the application area of virtual environments and interactive games. Smooth motions of the lower extremities are ensured by the objective function which attempts to minimize joint-angle differences

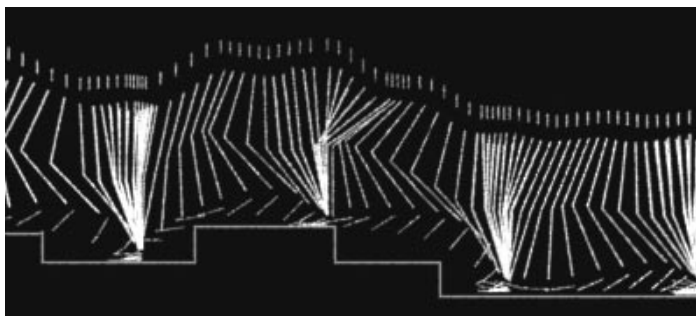
between consecutive time frames through the gait duration. Foot placement and trajectory are precisely controlled to prevent negative animation artifacts, such as the stance-foot sliding during the support phase, and collision with obstacles. Figure 14 illustrates sequences of walking with various gaits in different environments.



a. Leg motion in normal walking gait.



b. Walking sequence on uneven terrain. Left leg in motion.



c. Same walking sequence as **b.** Right leg in motion.

Figure 14. Walking sequences in different environments

Runtime complexity

The major computation in our algorithms is defining the pelvis trajectory through the gait cycle. A subdividing-in-level approach is applied to minimize the number of possible curves that the objective function has to evaluate. For example, among the n^m possible curves, $m \times n$ curves are actually evaluated, where n is the number of uniform cells, and m is the number of levels.

For each candidate curve, 20 equally temporally spaced sampling points in the gait cycle are used for equation 5 to evaluate the curve. Implementing the program on a PC platform (Pentium II 300 MHz, 128 MB RAM), the Table 1 shows the effect of the number of samples on system performance (measured in steps generated per second).

Number of samples	5	10	20	30	40
Average steps/sec	7.69	5.52	3.75	2.84	2.03

Table 1: Effect of sampling number on system's performance.

The subdividing-in-level approach and our relatively computationally-inexpensive inverse kinematics approach are used in computing the joint angles of the lower extremities. Thus, interactive simulation can be easily achieved. Considering that the cadence of normal walking is about 100-120 steps/minute, the algorithm is capable of simulating human walking at interactive rates, even on a relatively low-end platform.

From our experimental results, we notice that while increasing the number of samples does not automatically guarantee a better curve, reducing the sampling number to a certain threshold will improve the performance without significantly sacrificing too much in realism. We think this is important as the primary applications of our algorithms are for virtual environments and games; we can scale the algorithm to optimize for the best curve, under the constraint of the required interactivity.

Flexibility

By using a hierarchical motion control mechanism, the desired motions can be flexibly directed and controlled. At the high level, only minimal locomotion parameters, such as “destination” and “speed”, are required to generate the corresponding basic motions. While at the middle level, additional locomotion parameters, such as state-phase and gait determinants, are used to achieve different gait characteristics. Finally, at the low level, animation attributes affecting the upper body are fine-tuned to simulate different locomotion styles and personalities.

All these animation features are controlled by the user through a spin-button-based interface. The interactivity our algorithm provides allows the user to tweak the animation parameters and see the resulting animation on the fly. This interactivity is important in helping the animator gain experience manipulating the animating attributes, in order to generate desired movements.

Chapter 9

Discussion and Conclusions

The problems of simulating human walking have received considerable attention in computer animation, and many published works have produced convincing results. However, most of these motion control algorithms only take into account particular aspects of the many problems in animating human walking. They fail to provide a general, robust, and efficient solution for simulating highly articulated human figures in an interactive virtual environment. The research goal of this work is to find a solution to these problems that will provide interactive simulations of human walking in various virtual environments.

9.1 Observations and potential improvements

The difficulties in simulating human walking in virtual environments mainly come from the complexity of the terrain model, as additional constraints are required to be satisfied in defining the limb trajectories, such as those of end-effectors. In our work, we developed several unique approaches to defining these trajectories. Some interesting observations from the experimental results may need to be further addressed, as they have been important factors in finalizing the system design, and may be further studied to improve the walking model:

- Footprint planning strategies: Footstep planning plays an important role in walking on uneven terrain. Based on the spatial information from step length and terrain status,

the two-step lookahead approach works reasonably well in planning the footprints. It provides the spatial information of the foot placement, and frees the user from detailing the placement of footsteps, which is critical for building an interactive system. A potential improvement in computing the foot placement will be to factor-in the internal (duty factor) and external (step period) temporal information of the steps based on factors such as walking attributes and terrain status. This will need further experimentation and study.

- Constructing the limb trajectories: Bézier curves are used to represent the trajectories of the swing foot and pelvis. For the curve of the pelvis, the in-between control points are adjusted to match the pelvic movement observed during the gait cycle. For the swing-foot curve, the swing-foot speed profile, which exhibits strongly similar pattern in different modes of walking, is used to simulate the foot movement along the curve. While two curves are joined together to define the pelvis trajectory, a Bézier curve is used to simulate the swing-foot movement. This single-curve representation of the swing-foot trajectory could be a problem as the complexity of the terrain increases. Our current solution to this problem is mainly based on footprint planning. A potential improvement of the swing-foot model would be to add additional curve(s) to adapt to rough terrain.
- Realism improvement: Knowledge of human gait analysis and motion-capture data were used to improve the walking model. The motion control algorithms used to simulate the body movements are kinematic-based. These approaches, in general, are capable of generating reasonable results at interactive rates. However, there should be noticeable improvement if certain “dynamic” attributes can be added to

the walking model. For example, in the simulation of walking on uneven terrain, as the roughness of the terrain increases, the relatively static characteristics (due to the optimized objective, smoothness of leg joint movements) of the body movement become more noticeable. If some dynamic factors, such as momentum, can be quantitized to fit into the evaluation of body trajectories, more convincing results could be produced.

- Controllability at the lower levels: As motion control goes deeper in the motion control hierarchy, more control is given to the user, until each individual degree of freedom is accessible through the input by the user. This feature, coupled with efficient control algorithms, provides an interactive tool to animate desired motions; it allows the user to adjust the animation attributes and examine the motion on the fly. A potential improvement to this would be to relate sets of animation attributes to a variety of walks (personalities, styles). The user then can animate the desired motion by giving intuitive commands, such as “happily walk to X at speed Y,” instead of determining these attributes through trial and error. We are looking at the possibility of deriving the walking attributes automatically from motion capture data.

9.2 Conclusions

A hierarchical motion control mechanism has been introduced which allows the animator to generate a large variety of human walks interactively. Motion control at the high level allows the user to animate basic locomotion by specifying minimally-required parameters; the traveling path and locomotion speed. The path is generated based on the

piecewise cubic polynomial curves and their control points, so that curve shape can be easily adjusted to avoid obstacles. The walking speed is used to determine gait characteristics. This provides the user with a simple way to generate a desired walking motion by giving commands, such as “walk this route at speed X”. Like most human animation systems, the major problem in animating human locomotion is to define the link trajectories, especially the root (usually the pelvis or the supporting limb) trajectory. The computation of the limb trajectories in our system is based on observations of human walking collected by motion-capture, and our “energy optimization” approach. Using this technique, incorporated with human gait behaviors observed in experiments, such as foot placement strategies, natural human walking within an obstacle-ridden environment can be successfully simulated interactively.

At the lowest level of control, the user can add further varied motions by adjusting animation parameters, such as joint angles. At this level, satisfying motions are already generated by the system. For a specific desired motion, all the user has to do is override the basic motion by modifying necessary parameters. Because of the simplicity of overriding the existing motion, a variety of walks can be created interactively.

As an initial step in building a system that is capable of simulating human locomotion in a variety of virtual environments, our current system still requires a lot of exploration and improvements. We think the system can be greatly improved by integrating knowledge from various research fields, such as artificial intelligence and human psychology (for path and footprint planning), biomechanics, and human gait observation (motion capturing).

Appendix A: Using Bézier curve for editing the trajectory

The trajectory is generated using Bézier curve. The Bézier curve can be viewed as a representation that uses the two end points (P_0, P_3) and two other points (P_1, P_2) to indirectly specify the strength and orientation of the tangent vectors to the curve. The starting and ending tangent vectors are determined by the vectors P_0P_1 and P_2P_3 and are related to R_1 and R_4 by

$$R_1 = Q'(0) = 3(P_1 - P_0)$$

$$R_4 = Q'(1) = 3(P_3 - P_2)$$

These derivatives are in fact the tangent vectors to the curve at the end points.

The matrix formulation for specifying a Bézier curve can expressed as

$$Q(t) = TM_B P = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{bmatrix}$$

Where T is the power vector.

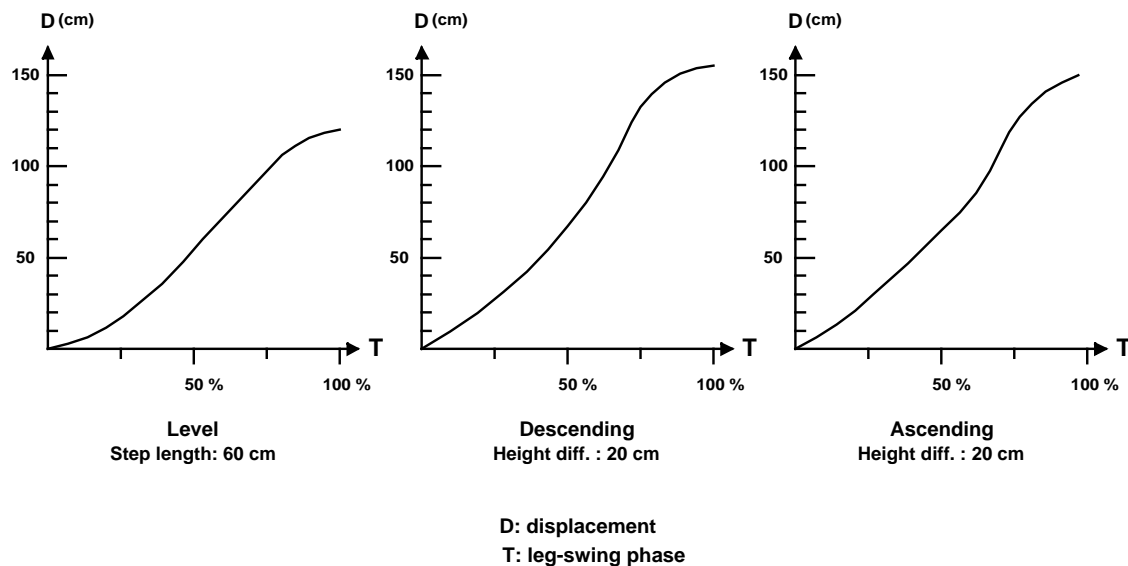
M_B is the Bézier basis matrix.

P is the Bézier geometry vector.

t is the parameter in the range $[0,1]$.

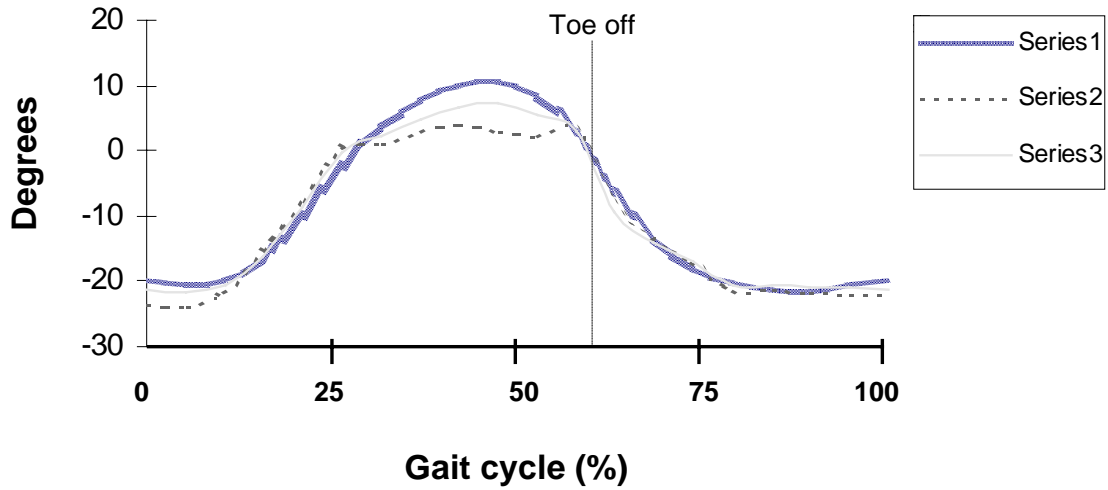
Appendix B: The *displacement vs. time* patterns of foot movement in various gaits

The speed of swing-foot varies during swing phase, generally being fastest during the middle and slowest in the beginning and terminating of the swing phase. The following charts show the *displacement vs. time* patterns of the swing-foot movement in various gaits (level, descending, and ascending). All three charts have a quite similar *displacement vs. time* pattern, thus, a “generic” *displacement vs. time* pattern is formed and used in our swing-foot model for all gaits.

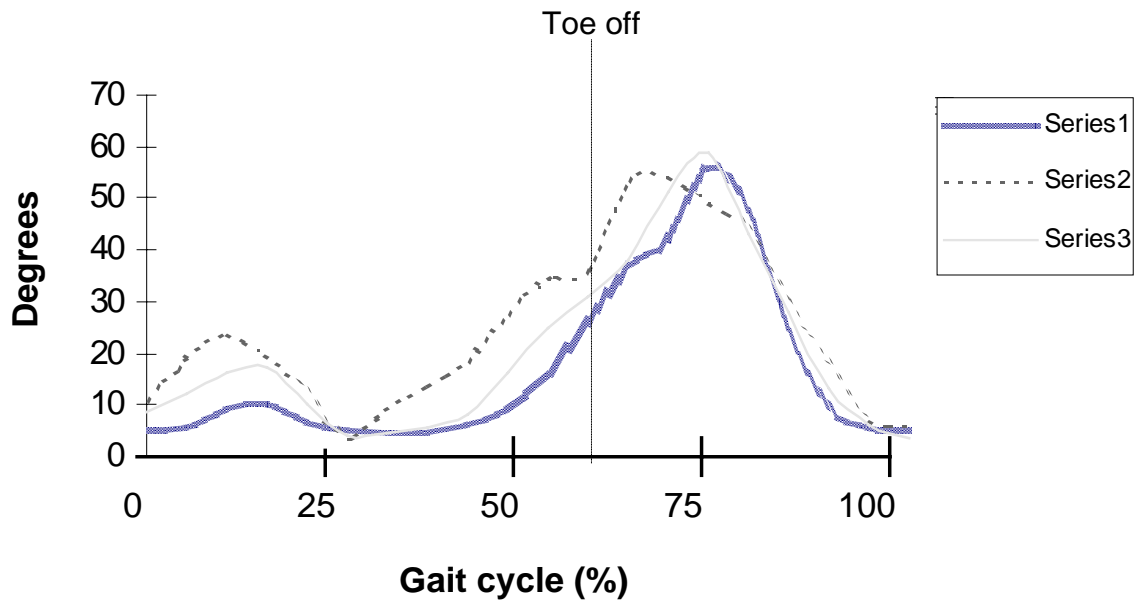


Note: Data were collected from the motion capturing work at NIH.

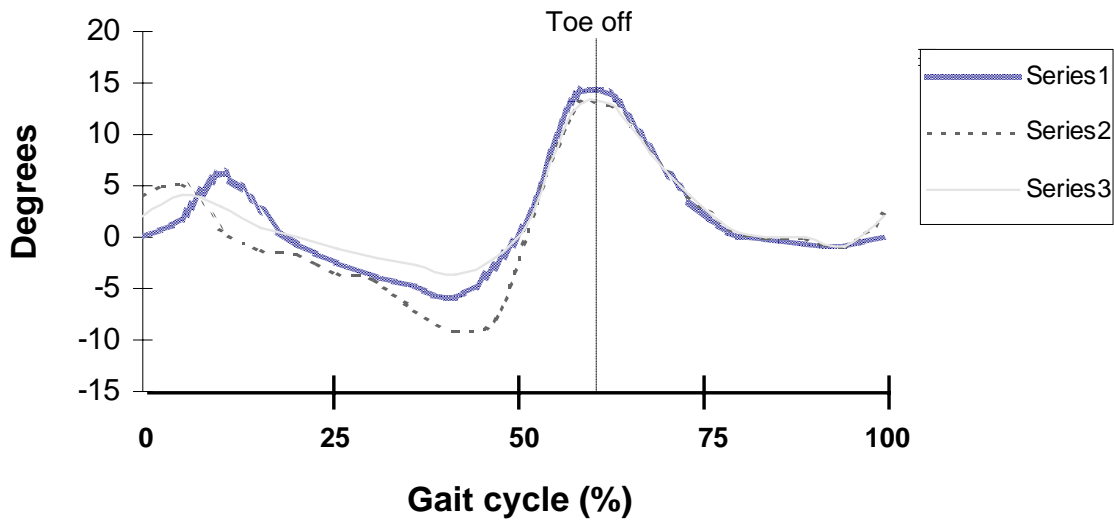
Appendix C: The weighting factors and their correspond leg-joint angular motions



I. Hip joint



II. Knee joint



III. Ankle joint

Variation of leg-joint angles in normal straight walking on flat ground

I. Hip joint

II. Knee joint

III. Ankle joint

Body height: 1.7 m

Step length: 60 cm

Duty factor: 0.6

Series 1: captured motion

Series 2: simulated walking with equal weighting factor for each leg joint

Series 3: simulated walking with weighting factor $\mathbf{W}_{\text{hip}} : \mathbf{W}_{\text{knee}} : \mathbf{W}_{\text{ankle}} = 1 : 1 : 3$

Reference

1. Armstrong, W. W. and M. W. Green. "The Dynamics of Articulated Rigid Bodies for Purposes of Animation." *The Visual Computer* 1(4), pp. 231-240, 1985.
2. Andriacchi, T., G. Anderson, R. Fermier, D. Stern, and J. Galanta. "A Study of Lower-Limb Mechanics during Stair-Climbing." *J. Bone and Joint Surgery*. 62A, pp. 749-757, 1980.
3. Badler, N. I., B. A. Barsky, and D. Zeltzer. "Making Them Move: Mechanics, Control, and Animation of Articulated Figures." Morgan Kaufmann Publishers, Inc. CA, USA. 1991.
4. Beckett R. and K. Chang. "An Evaluation of the Kinematics of Gait by Minimum Energy." *J. Biomechanics*, Vol. 1, pp. 147-159, 1968.
5. Boulic, N., M. Thalmann and D. Thalmann. "A Global Human Walking Model with Real-time Kinematic Personification." *The Visual computer*, pp. 344-358, 1990.
6. Bruderlin A. and T. W. Calvert. "Goal-Directed Dynamic Animation of Human Walking." *Computer Graphics* 23(4), pp. 233-242, 1989.
7. Bruderlin A. and T. W. Calvert. "Interactive Animation of Personalized Human Locomotion." *Graphics Interface* 1993, pp. 17-23, 1993.
8. Bruderlin A. and L. William. "Motion Signal Procession." *Computer Graphics* 29(4), pp. 97-104, 1995.
9. Bruderlin A. and T. W. Calvert. "Knowledge-Driven, Interactive Animation of Human Running." *Graphics Interface* 1996, pp. 213-221, 1996.
10. Calvert T. W. and J. Chapman. "Aspects of the Kinematic Simulation of Human Movement." *IEEE Computer Graphics and Applications*, pp. 41-50, November 1982.
11. Chao Y. S. *Biomechanics of Human Gait*. New York: Springer-Verlag, 1985.
12. Chung S. and J. Hahn. "Animation of Human Walking in Virtual Environments." *Computer Animation* 99, pp. 4-15, 1999.
13. Cohen F. "Interactive Spacetime Control for Animation." *Computer Graphics* 26(4), pp. 293-302, 1992.
14. Denavit, J. and R. S. Hartenberg. "A Kinematic Notation for Lower-Pair Mechanism Based on Matrices." *ASME Journal of Applied Mechanics*, Vol. 23, pp. 215-221, 1955.

15. Featherstone R. Robot Dynamics Algorithms. Kluwer Academic Publishers, Norwell, MA, 1988.
16. Flynn J. "Kinematical Variability of Normal Climbing and Descending Stairs." M. Sc. Thesis, Vanderbilt University, Nashville, TN, 1977.
17. Girard M. and A. A. Maciejewski. "Computational Modeling for the Computer Animation of Legged Figured." Computer Graphics 19(3), pp. 263-270, 1985.
18. Girard M. "Interactive Design of 3D Computer-Animated Legged Animal Motion." IEEE Computer Graphics and Applications 7(6), pp. 39-51, 1987.
19. Gritz L. and J. Hahn. "Genetic Programming for Articulated Figure Motion." J. of Visualization and Computer Animation, Vol. 6, pp. 129-142, 1995.
20. Gleicher M. "Motion Edition with Spacetime Constraints." Proc. 1997 Symposium on Interactive 3D Graphics, pp. 139-148, 1997.
21. Gleicher M. "Retargetting Motion to New Characters." Computer Graphics, 32(4), pp. 33-42, 1998.
22. Hodgins J. K., W.L. Wooten, A.C Brogan and J.F. O'Brien. "Animating Human Athletics." Computer Graphics, 29(4), pp. 71-78, 1995.
23. Holt K. G., J. Hamill, and R. O. Andres. "Predicting the minimal energy costs of human walking." Med. Sci. Sports Ex. 23, pp. 491-498, 1991.
24. Hodgins J. K., N. S. Pollard. "Adapting Simulated Behaviors for New Characters." Computer Graphics, 31(4), pp. 153-162, 1997.
25. Inman V. T., H. Ralston, and F. Todd. Human Walking. Williams and Wilkins, Baltimore, 1981.
26. Issacs P. and R. Cohen. "Controlling Dynamic Simulation with Kinematic Constraints, Behavior Functions and Inverse Dynamics." Computer Graphics, 21(4), pp. 215-224, 1987.
27. Ko H. and N. I. Badler. 1993. "Straight-line Walking Animation based on Kinematic Generalization that Perserves the Original Characteristics." Graphics Interface 1993, pp. 9-16, 1993.
28. Ko H. and N. I. Badler. "Animating Human Locomotion with Inverse Dynamics." IEEE Computer Graphics and Applications, 16(2), pp. 50-59, 1996.
29. Koga Y., K. Kondo, J. Kuffner and J. Latombe. "Planning Motions with Intentions." Computer Graphics 28(4), pp. 395-408, 1994.

30. Laszlo J., M. van de Panne and E. Fiume. "Limit Cycle Control and Its Application to the Animation of Balancing and Walking." *Computer Graphics* 30(4), pp. 155-162, 1996.
31. Lee and Shin. "A Hierarchical Approach to Interactive Motion Editing for Human-Like Figures." *Computer Graphics* 33(4), pp.39-48, 1999.
32. Liu Z., S. J. Gortler, and M. F. Cohen. "Hierarchical Spacetime Control." *Computer Graphics* 28(4), pp. 35-42, 1994.
33. Livingston L. "Kinematic Analysis of Stair Climbing." M. Sc. Thesis. Queen's University, Kingston, Ontario, 1984.
34. McFadyen B. J. and D. A. Winter. "An Integrated Biomechanical Analysis of Normal Stair Ascent and Descent." *J. Biomechanics*, 21(9), pp. 733-744, 1988.
35. McKenna M. and D. Zeltzer. "Dynamic Simulation of Autonomous Legged Locomotion." *Computer Graphics* 22(4), pp. 29-38, 1990.
36. Murray M. P., A. B. Drought, and R. C. Kory. "Walking Patterns of Normal Men." *J. of Bone and Joint Surgery* 46A, pp. 335-359, 1964.
37. Murray M. P., S. B. Sepic, E. J. Barnard. "Patterns of Sagittal Rotation of the Upper Limbs in Walking." *J. of American Physical Therapy Association* 47, pp. 272-284, 1967.
38. Patla A. E., S. D. Prentice, C. Robinson, and J. Neufeld. "Visual Control of Locomotion: Strategies for Changing Direction and for Going over Obstacles." *J. Exp. Psychol. Human Percep. Perform.* 17, pp. 603-634, 1991.
39. Perlin K. "Real-Time Responsive Animation with Personality." *IEEE Trans. on Visualization and Computer Graphics*, Vol. 1, pp. 5-15, Mar. 1995.
40. Perry J. *Gait Analysis*. McGraw-Hill, New York. 1992.
41. Phillips C. B., N. I. Badler, and B. L. Webber. *Simulating Humans*. Oxford University Press, 1993.
42. Phillips C. B. and N. I. Badler. "Interactive Behaviors for Bipedal Articulated Figures." *Computer Graphics* 25(4), pp. 359-362, 1991.
43. Popvic Z. and A. Witkin. "Physically Based Motion Transformation." *Computer Graphics* 33(4), pp.11-20, 1999.

44. Raibert M. H. and J. K. Hodgins. "Animation of Dynamic Legged Locomotion." *Computer Graphics*, 25(4), pp. 349-358, 1991.
45. Redfern M. S. and T. Schumann. "A Model of Foot Placement during Gait." *J. Biomechanics*, 27(11), pp. 1339-1346, 1994.
46. Rose C. F., B. Guenter, B. Bodenheimer, and M. F. Cohen. "Efficient Generation of Motion Transition Using Spacetime Constraints." *Computer Graphics* 30(4), pp. 147-154, 1996.
47. Rose C. F., M. F. Cohen, and B. Bodenheimer. "Verbs and Adverbs: Multidimensional Motion Interpolation." *IEEE Computer Graphics and Applications*, pp. 32-40, Sept. 1998.
48. Saunders J. B., V. T. Inman, and H. D. Eberhart. "The Major Determinants in Normal and Pathological Gait." *J. Bone Joint Surgery*, 35A, pp. 543-558.
49. Sims K. "Evolving Virtual Creatures." *Computer Graphics* 25(4), pp. 15-22, 1994.
50. Stredney D. "The Representation of Anatomical Structures Through Computer Animation for Scientific, Educational and Artistic Applications." M.A. thesis. The Ohio State University, 1982.
51. Sturman D. "Interactive Keyframing Animation of 3-D Articulated Models." *Graphics Interface '86, Tutorial on Computer Animation*, 1986.
52. Thalmann N. M. and D. Thalmann. *Computer Animation: Theory and Practice*. Springer-Verlag, New York, 1990.
53. Townsend A. "Biped Gait Stabilization via Foot Placement." *J. Biomechanics* Vol. 18. No. 1, pp. 21-38, 1985.
54. Townsend M. A. and T. C. Tsai. "Biomechanics and Modelling of Bipedal Climbing and Descending." *J. Biomechanics*. Vol. 9, pp. 227-239, 1976.
55. Unuma M., K. Anjyo, and R. Tekeuchi. "Fourier Principles for Emotion-Based Human Figure Animation." *Computer Graphics* 29(4), pp. 91-96, 1995.
56. Van de Panne M. "From Footprints to Animation." *Computer Graphics Forum*, 16(4), pp. 211-213, 1997.
57. Westenhofer B. and J. K. Hahn. "Using Kinematic Clones to Control the Dynamic Simulation of Articulated Figures." *Computer Graphics International*, 1996.
58. Whittle W. *Gait Analysis: an introduction*. Butterworth-Heinemann, Oxford. 1996.

59. Wiley J. and J. K. Hahn. "Interpolation Synthesis for Articulated Figure Motion." Proc. Virtual Reality Annual Int'l Symp., IEEE CS Press, pp. 156-160, Mar. 1997.
60. Wilhelms J. "Using Dynamic Analysis for Realistic Animation of Articulated Bodies." IEEE Computer Graphics and Applications 7(6), pp. 12-27, 1987.
61. Winter D. The biomechanics and motor control of human gait. Waterloo, Canada: University of Waterloo Press, 1987
62. Winter D. "Foot Trajectories in Human Gait: a Precise and Multifactorial Motor Control Task." Phys. Therapy 72(1), pp. 45-53, 1992.
63. Witkin A. and M. Kass. "Spacetime Constraints." Computer Graphics 22(4), pp. 159-168, 1988.
64. Witkin A. and Z. Popovic. "Motion Warping." Computer Graphics 29(4), pp. 105-108, 1995.
65. Zelter D. "Motor Control Techniques for Figure Animation." IEEE Computer Graphics and Applications, 2(9), pp. 53-59, 1982.