Animation of Human Walking: A Survey Based on Artistic Expression Control

人物走路動畫：以藝術表現控制為主的研究論述

Chung Shih-Kai, Assistant Professor,
Dept. of Multimedia and Animation Arts, National Taiwan University of Arts

鐘世凱，助理教授
國立台灣藝術大學多媒體動畫藝術學系

ABSTRACT

Computer animation of human walking has become popular in recent years because of the desire to use human beings as synthetic actors in three-dimensional simulation environments. This paper surveys the motion control techniques developed in Computer Animation for animating human walking. It summarizes the theoretical approaches used in published work and describes their strengths, weaknesses, and relative performance. First, the problems and current status of computer animation of human walking are described. Then, we review a variety of motion control techniques of human-walking animation. Artistic expression control is especially emphasized in evaluating each method. Finally, some potential improvements of these computational models in achieving artistic expression are also offered as the conclusions.

Keywords: computer animation, human walking, motion control, artistic expression control.

摘 要

3D 場景中虛擬人物的應用需求帶動人類走路動作的模擬成為電腦動畫研究的顯學。本篇論文研究探討電腦動畫角色人物走路動作模擬的各種作法，並針對已發表的學術論文中各項走路動作控制方法簡述歸類，探討分析各類控制作法其優、缺點及相關的表現。本文首先闡述電腦動畫角色走路動畫的困難與現狀，繼而探討各類動畫模擬方法，並特別以“藝術表現性”評估分析各作法。最後結論提出人類走路動作藝術表現的建議模擬控制架構及可能的改進作法。

關鍵詞：電腦動畫，人物走路動作，動作控制，藝術表現控制。

1. Introduction

The representation and display of virtual 3-D 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 (even for simplified human figure representation, there are usually more than 30 degrees of freedom in it). Controlling these many hierarchical degrees of freedom 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). These challenges increase the difficulty of animation task. 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 database on "normal" straight-walk gait patterns. In the field of computer animation, because of the complex hierarchical structure of human being, most of the researches in motion control of human figures have 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 constraints set 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 of the animator, in order to produce a desired motion. This is particularly true when the envisioned motion includes stylized movements that are deliberately objective-oriented, such as walking on uneven terrain.

This paper surveys a variety of motion control techniques in computer animation, focusing on the solution each method brings to the control of human walking. Since several fields have many kinds of walking models, rather than provides a general review of human walking methods, this study uses an important animation criteria in motion control to survey these animation models – artistic expression control.

Whereas a general model may provide basic control of normal walking, different computational model may each have its own strength and advantage over the others in suitable applications. However, an ideal model should be able to provide animator the easy-control ways to desired motion. That is, it should give the animator an user-friendly framework to interactively generate the motion. In the mean time and more importantly, animators should have no problem to implement their artistic expression, such as Animation Principles (Thomas & Johnston, 1981), using these walking models?

This survey proceeds as follows. Section 1 provides a general overview of animation of human walking. Section 2 introduces human model and the basic framework to control human walking. Kinematics motion control techniques are described in section 3, followed by dynamics methods in section 4. Section 5 discusses hybrid methods which may combine several control approaches to simulate the motion. Motion data and motion editing are surveyed in section 6. Finally, general conclusions and observations are presented in section 7.

2. Human Walking

Animation of human walking is a crucial but difficult problem in computer animation. The challenges mainly come from the following two reasons:
(1). Geometric models used in computer graphics are not well-suited for the shape of human body.
(2). The complex movement of joints is difficult to control.
The first issue is more about the appearance of the human model, and many research fields in computer graphics have addressed these problems, for example “subdivision” technique in modeling, realistic shading of human skin, physically-based cloth and hair simulation. These techniques all improve appeal of the human model. However, these appearance-related issues are beyond the scope of this survey. The second issue addresses the problems in motion controlling the complex structure of human skeleton. Three tasks are needed in order to animate human walking: First, a proper articulated representation of the human model. Second, the mechanism of human walking. Finally, controlling the articulated model to animate human walking.

2.1. Human Model

The first problem in human animation is how to represent a human model precisely. The representation of a human model in computer graphics usually is done by an articulated figure, which is made up of a series of body segments (rigid body) connected at joints. Each body segment can be described in terms of a coordinate frame attached to the body. A rigid body can have six degrees of freedom (DOF) of motion: It can translate (its position) and rotate (its orientation) in three directions (x’s, y’s, and z’s axis). For the articulated figure which represents the human skeleton, a joint is a skeleton point where the limb which is linked to the point may move, and all joints in the body are restricted to rotary joints. That is, each body segment is connected (a positional constraint) to another one in the articulated structure. Hence, there is no positional DOF, but rotational DOF only for the joints.

The human being possesses more than 200-piece of bone segments. However, anatomical research has revealed that some of the bones, such as those from the head, are either fixed or hardly movable with respect to their adjacent neighbors. Although this phenomena can reduces a great amount of DOFs to be animated, theoretically, we still need to animate more than 200 DOFs, which is very difficult to control simultaneously, to mimic detailed human motion. In fact, most human-like characters in computer animation use a far less DOFs model. Even so, a simplified human figure representation, usually possesses more than 30 degrees of freedom in it, as shown in Figure 1.

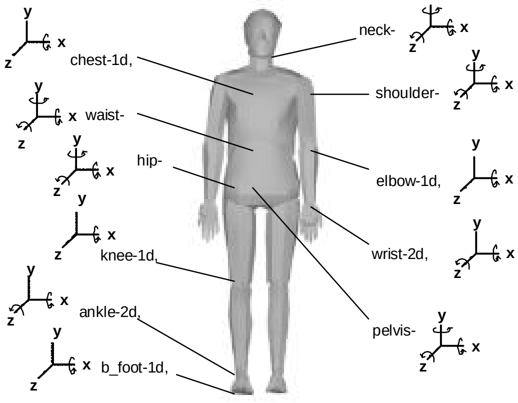

Figure 1: The controlled degrees of freedom of the human model. There are 18 body segments and a total of 36 controlled degrees of freedom.

2.2. Walking

Walking is the most common means of moving about and is the essential activity 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 people walk in the same way. From human gait observations (Murray et al., 1964), the difference 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 focuses more on the lower limb joint movements, especially in the sagittal plane, and leaves the rest of the body joints to the animator for desired movements.

2.2.1. 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 the temporal and spatial components. Some of the following terminology of gait relates to the period of time during which events take place, and some refer 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 swing forward in preparation for the next step. These phases and their timings are illustrated in Figure 2.

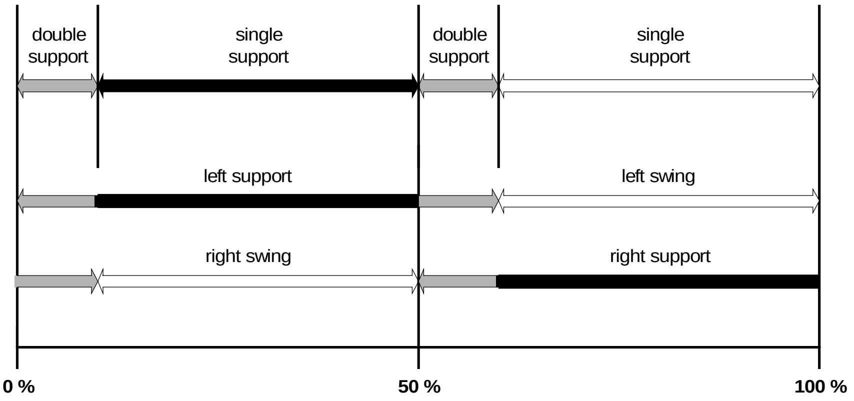

Figure 2: Locomotion cycle for bipedal walking

Step duration = Support duration + Swing duration
Duty factor = Support duration / Step duration
Double support duration = (Support duration - Swing duration) / 2

Support Phase

This 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, 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 position 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 at 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 supporting floor at about the middle of this period. This is also the position where the walker is at their least stable.

Push off: this period starts from the end of ‘flat foot’ until 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 drive forces come from the hip (major) and plantar (minor) flexors. The non-weight-bearing limb was accelerated forward in this period.

Mid-swing: this corresponds with mid-stance and at this moment the swing limb passes the supporting limb with rather steady speed.
Deceleration: 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 those times in the gait cycle when concentric activity is required to accelerate a limb (Winter, 1987).

Double Support Phase

Double support phase refers to the period of time when both feet are in contact with the ground. It is a small interval during the gait cycle that two leg events are overlapped: the final fraction of support phase from one leg, and the beginning fraction from the other leg. Its temporal length is equal to the difference between support phase and 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 running state. Human gait observations have shown that at an 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, average normal walk has a leg duty factor about 0.6.

2.3 Animating Human Walking

2.3.1 Robotics Background

Research results from robotics have proposed several kinds of walking models. Among those many robotic terms adopted in articulated figure animation, ***degrees of freedom**, **end effector***, and *state vector* are crucial in controlling human motion.

Degrees of Freedom

A virtual object can be positioned precisely in 3D space through setting six geometrical attributes (3 for position and 3 for orientation) correctly. That is, an object in free motion possesses six degrees of freedom at most. To specify the state of an articulated structure, which consists of a series of body segments, the total number of independent position variables needed is the number of degrees of freedom of that figure.

State Vector

In robotics, a state vector is the minimum number of variables needed to describe dynamic behavior of a system. Applying this to articulated figure animation, a state vector is a set of parameters defining the configuration of all joints constituting the figure. That is, an articulated figure in a particular pose is described by the state vector:

$$q = (q_1, q_2, \ldots, q_n) \qquad (1)$$

The dimension of the state vector in general is equal to the DOF of the articulated figure, and it is the algorithm/animator’s responsibility to control these many elements simultaneously in the vector to generate the desired pose in specific time.

End Effector

The end effector is the free end of a chain of links in a structure. In robotics, this usual refers to the “hand” connected to the robot’s arm. For human animation, which the human model itself is a structure of open chains, body segments, such as head, hands and feet, in general, are used as the end effector.

2.3.2 Representing Human Figures

Using computer to generate human animation, the first problem needed to be addressed is how to represent the human figure mathematically. Two representations are widely used in computer animation of articulated figure.

DH-notation

From robotics research, Denavit and Hartenberg proposed the famous DH-notation (Denavit & Hartenberg, 1955). 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*** (***a***), ***the distance between links*** (**d**), ***the twist of the link*** (a), and ***the angle between links*** (q), are used to define the linear transformation matrix between adjacent coordinate systems attached to each joint. D-H notation is designed for link structures where the joints have a single DOF. Applying in human animation, this usually means a rotational DOF about a single axis. The transformation matrix that relates coordinate frame ***i*** to frame ***i-1*** can be expressed as

$$
{}^{i-1}T_i = \begin{bmatrix} \cos q_i & \sin q_i \cos a_{i-1} & \sin q_i \sin a_{i-1} & 0 \\ -\sin q_i & \cos q_i \cos a_{i-1} & \cos q_i \sin a_{i-1} & 0 \\ 0 & -\sin a_{i-1} & \cos a_{i-1} & 0 \\ a_{i-1} & -d_i \sin a_{i-1} & d_i \cos a_{i-1} & 1 \end{bmatrix} \qquad (2)
$$

And the configuration of each body segment (*i*) in the articulated figure can be computed by

$$
{}^{0}\mathrm{T}_{i} = {}^{0}\mathrm{T}_{1}\,{}^{1}\mathrm{T}_{2} \ldots {}^{i-1}\mathrm{T}_{i} \qquad (3)
$$

where T is the transformation matrix which relates two coordinate frames, and ${}^{0}\mathrm{T}$ is the base frame which specifies the position and orientation of the root link in the articulated structure.

Axis Position (AP) Joint Representation

Adopted from robotics, DH-notation is suitable for manipulators that are a single chain structure. For “ball” joints which contain multiple rotational DOFs, such as the shoulder or ankle joints, or branching joints, such as the hip or neck joints, multiple, single DOF joints will be needed to applied at the same part in space. Sims and Zeltzer (Sims & Zeltzer, 1988) propose an intuitive system called axis-position joint representation, which records 7 parameters for a joint:

(1). x-translation, y-translation, and z-translation of the joint.
(2). x-rotation, y-rotation, and z-rotation of the joint.
(3). Pointer to the link(s) which the joint is attached to.

Compared to DH-notation which requires 4 parameters to specify a coordinate frame (child-link) with respect to the previous (parent link), AP needs 7 parameters to specify a joint in space. At first glance, DH-notation seem is a more economical relative system than AP to compute the configuration of an articulated figure. However, AP provides a more intuitive system to set a joint and its links in space. Besides, “ball” joints and “branching” links in human body both favor AP representation.

2.4 Motion Control of Human Animation

In light of the recent surge of interest in virtual environment applications, many researches have devoted to solve the problems of manipulating human in the 3-D simulated worlds, especially human walking. Different methods may integrate knowledge from various fields, such as animation, robotics, biomechanics, and psychology, into a human walking model. These animating systems can be roughly classified into the four main groups: kinematics, dynamics, hybrid and motion editing.

Kinematics: procedural approaches which based on empirical/biomechanical knowledge kinematics

Dynamics: approaches which make use of knowledge of dynamic properties to specify the natural movement.

Hybrid: techniques which integrate different methods, mainly kinematics and dynamics.

Motion editing: methods which enable edition of captured /synthetic walking motion.

Automatism vs. control is a classical trade-off in motion control of computer animation. Unlike a classical survey, which provides a general review of previous work based on human motion. This paper focuses on methods for generating the human walking. The most important control criteria, “artistic expression”, in computer animation are especially emphasized in surveying the previous work. While performance, especially efficiency in generating animation, is always in surveying a method, capability, which refers to the human figure’s various motions, such as walking on uneven terrains or walking in different styles, is also evaluated in this paper. As performance has a method’s automatism covered, capability and controllability are more about animator’s artistic expression.

The next question comes up with artistic expression is how do we measure the degree of it. Our solution comes from the classical Disney’s Animation Principles, which contains twelve guidelines for animators to generate quality animation. Among these twelve principles, ten of them are about animator’s control over the motion: Squash and Stretch, Timing, Anticipation, Follow through and overlapping action, Straight ahead action and pose-to-pose action, Slow in and slow out, Arcs, Exaggeration, Secondary action, Appeal. In this paper, the above motion-related animation principles are used as the tools for artistic expression in surveying the methods.

3. Kinematics

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 (Calvert & Chapman, 1982; Sturman, 1986; Boulic et al., 1990; Thalmann & Thalmann, 1990; Phillips et al, 1993; Bruderlin & Calvert, 1996). These approaches for articulated figure animation generally fall into one of the two categories:

3.1 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. Mathematically, each link in the articulated figure can be computed by the following equation:

$$X = f(q) \qquad (q_1, q_2, \ldots, q_n) \in q \tag{4}$$

where X represents the pose of link in space, and $(q_1, q_2, \ldots, q_n)$ represent the ordered degree of freedom in the link structure respectively.

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 walking, 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. at each foot-strike, the new supporting foot becomes the root, and its position is fixed on the ground).
- 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 (Zelter, 1982) embedded biomechanical knowledge of locomotion and used hierarchical motor control techniques to animate a human skeleton walking with a straight-ahead gait over level, unobstructed terrain. Key postures are set through specified state values, and these values are then linearly interpolated to generate in-between postures. Variations of walking, such as different walking styles or walking on moderately uneven terrain, are achieved by parameterizing the generalized walk controller and its associated motor programs. Unfortunately, this will require 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 (Bruderlin & Calvert, 1989) 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. Note that the same kind of mechanism was later used to simulate human running (Bruderlin & Calvert, 1993). 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.

3.2 Inverse Kinematics (IK)

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 relation between the end-effector and joint angles in the chain structure takes the form:

$$\Delta X = J \Delta q \tag{5}$$

where r X represents motion of the end-effector, r q represents the angular displacements of the joint angles in the articulated structure, and $\boldsymbol{J}$ is the Jacobian matrix of the kinematics system.

Due to the different dimensions of X and q, for example, for redundant links, the joint-space is of higher dimension than the six-dimensional (6 DOFs) Cartesian space of the end-effector, $\boldsymbol{J}$ is not directly invertible. This implies that there is an infinite number of joint-space rates that will lead to the same Cartesian space of the end-effector. A widely used solution is give by:

$$\Delta q = J^{+} \Delta X + a (I - J^{+} J) \Delta z \tag{6}$$

where $\boldsymbol{J}^{+}$ is the pseudo-inverse of the Jacobian matrix $\boldsymbol{J}$, a is the penalty value between zero and one, $\boldsymbol{I}$ is an $\boldsymbol{n}$ x $\boldsymbol{n}$ identity matrix, and $\boldsymbol{z}$ is a constraint, such as energy consumed, to minimize.

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 solves 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. (Boulic et al., 1990) used a generalization of experimental data based on the normalized velocity of walking. Forward kinematics was used to generate key postures that are interpolated. The generalization, in its direct application, could produce undesired results, such as parameters violate some of the kinematics constraints (i.e. a foot penetrates the ground.) imposed on walking. Then, inverse kinematics was implemented to correct these problems. Among the multiple inverse kinematics solutions, the one that is the closest to the original motion is chose to preserve the original characteristics of the walking data. Based on the ***Jack*** system (Phillips & Badler, 1991) developed at University of Pennsylvania, Ko and Badler (Ko & Badler, 1993) implemented inverse kinematics algorithm to generate motions. The users will 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. (Koga et al., 1994) used a path planner to compute the collision-free trajectories for cooperating arms to manipulate a moveable object between two configurations. Inverse kinematics 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 is computed to match the hand orientation.

For systems that devote to human locomotion, Girard's PODA (Girard & Maciejewski. 1985) uses a mix of kinematics and "pseudo-dynamic" methods to simulate human locomotion. A multi-pass process is used to determine the body motion that 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 kinematics 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.

For style-based simulation, the recent work from Grochow et al. (Grochow et al., 2004) proposed an inverse kinematics system, which generates motion based on a learning probability model of human poses. The IK system is capable of generating any static poses, but the strength of their system is to create stylistically consistent animations based on previously observed natural poses. A Probability Distribution Function (PDF), which indicates the "probability" function over character poses, is applied to pick poses that are most similar to the space of poses in the existing motion data. The major limitation of the system is it requires suitable training data; if the existing motion data does not contain poses which match the desired poses well, more constraints will be needed to select new poses during inverse kinematics simulation.

3.3 Discussion

Traditional keyframing technique provides animator complete control over the motion animated. The quality of the result totally depends on animator's skill, spending effort, and time. Since the animator control every state of the articulated figure in any desired key time, it provides the best tool to display animator's artistic expression. Theoretically, holding the animator's capability, any type of walking motion is possible. This overwhelming advantage over the other motion control techniques make key framing the major, even the only, tool in production house which own skilled animators, to produce film quality character animation. For example, the production house of "Toy Story" and "The Incredibles", Pixar, uses keyframing only to animate character motion. The major drawback from keyframing techniques is the enormous effort and time the animator has to spend. Thus, from functionality point of view, real time simulation of autonomous human walking is impossible.

From the prospect of functionality, due to the light kinematics computation needed, real-time simulation of human walking can be easily achieved through today's computation hardware. This advantage makes kinematics techniques well suited for time critical applications, such as virtual reality and its related applications. As for variations of walking, previous works have shown parameterizing the generalized walk controller might have the solution. However, animator's full understanding of the skeleton animation system is required. This includes detailed knowledge about how the human skeleton controller responds to the added constraints from different walking styles or various environments. More importantly, what parameters and how to tune them to meet the new requirements or constraints?

Compared to direct keyframing, the kinematics methods presented above have a common goal to alleviate animator's load in detailing the motion. Thus, animator can invest his time and effort to achieve more important artistic expression. This in general is done through a hierarchical control structure. At the top level, setting several key parameters, such as trajectory and velocity of walk, a general walking motion, which satisfies the constraints of key parameters, should be generated. To get further specification of the motion, such as stylized walking with personality, controlling will have to go deeper in the hierarchy and more parameters need to be set. At the lowest level of the control hierarchy, detailed movements, such as fine-tune individual DOF in the articulated structure, for each limb segment as a function of time are accessible to the animator. So, theoretically, animator will have the full freedom to implement each animation principle to fulfill his/her artistic expression in the character animation.

4. Dynamics

Dynamic approaches describe motion by a set of forces and torques from which kinematics data are derived. Basic dynamic simulation can be implemented in a few easy steps. (Winter, 1987) First, setting physical description of the objects to be animated. Second, formulating the dynamic equations of motion, which relate how objects move under the influence of forces and torques. Third, giving known forces and torques, the equations are solved for accelerations. Fourth, a numerical integration is performed to find new velocities and positions. Finally, these new positions are used to update graphical objects in the scene.

The dynamics of an object's movement is guided by Newton's laws. That is, given the presence of external forces (torques) and the mass and moment of inertia of the object, dynamics defines the relationship between the forces (torques) applied and the acceleration, velocity, and position of the object. As shown in the following equations.

$$f = m \cdot \ddot{x} \qquad (7)$$

$$t = i \cdot \ddot{q} + \dot{q} \cdot i\dot{q} \qquad (8)$$

where ***f*** is the external force applied to an object, ***m*** represents it's mass and $\ddot{x}$ the 2nd derivative with respect to time of the position vector ***x***, t is the torque, ***i*** the inertia matrix, $\ddot{q}$ the angular acceleration, and $\dot{q}$ the angular velocity.

Controllers are often used in dynamics simulations of human motion, especially for cyclic motion such as locomotion. In these frameworks, the animator animate the desired motion by specifying key states, just like setting key poses in keyframing techniques. Based on the current state and the desired state of the figure, the dynamics controller used in computing the needed forces is also responsible for controlling the direction and the speed of the motion, regulating the movement of joints, and stabilizing posture for maintaining balance. The force that leads the joint from the current state to the next state can be expressed as:

$$f = -k_p(q - q_d) - k_v(\dot{q} - \dot{q}_d) \qquad (9)$$

where k_p and k_v are proportional and derivative gains, $(q, \dot{q})$ and $(q_d, \dot{q}_d)$ are the current and next states (position and velocity) of the model.

Giving the force (torque), ***forward dynamics*** applies the above laws to calculate acceleration, thus motion of the object. The use of forward dynamics usually has to couple with some control strategies, if coordinated goal-directed movement is desired. Conversely, ***inverse***

dynamics uses a reference trajectory of the movement, then calculates the needed force and torque to generate such a motion.

4.1 Previous Works

Dynamic simulation and control algorithms (Armstrong & Green, 1985; Featherstone, 1988; McKenna & Zeltzer, 1990; Raibert & Hodgins, 1991; Van de Panne et al., 1994; Hodgins et al., 1995; Ko & Badler, 1996; Laszlo et al., 1996; Hodgins & Pollard, 1997; Komura et al., 2004) 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 the determination of joint and muscle forces in gait is 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.

Solutions to the dynamics problems for articulated structures are well understood in robotics. Featherstone (Featherstone, 1988) presented a rather comprehensive study of robot dynamics. Armstrong and Green (Armstrong & Green, 1985) proposed a recursive dynamics formulation, which is much faster than the earlier nonrecursive methods, for use in graphical simulation. Based on the basic Euler dynamics equations, they developed ***O(n)*** forward dynamics algorithms that propagate values from the base of the linked structure and advance to the end link recursively. The method is linear in the number of degrees of freedom, thus provides a fast solution for dynamics simulation.

Ko and Badler (Ko & Badler, 1996) applied inverse dynamics to calculate the force (torque) needed to perform a given motion. Motion is later corrected if it fails the verification of “balance” or “comfort” control. For example, in balancing control, the inverse dynamics control module will calculate the angle difference between initial joint and inverse dynamics result. Then add it to the kinematics initial trajectory, in order to retain balance.

Using inverse dynamics with functional control modules, their system is able to simulate virtual actor automatically adjust his posture if external force applied, such as bending the pelvis while hold a heavy weight.

Since normal walking can be represented as a cyclic motion, the concepts of using cyclic pose control graphs to generate the movement of walking have been adopted in several research works, such as (Raibert & Hodgins, 1991; Van de Panne, 1994; Hodgins et al., 1995; Hodgins & Pollard, 1997). The animator uses traditional keyframing or kinematics method to set a series of poses for a desired motion. Later these poses serve just like key-frames in keyframing. A dynamics controller is then used to compute the force and torques needed to drive the synthetic figure from pose to pose. Finally, forward dynamics is applied to generate the motion that results from the set of applied forces and torques.

Hodgins et al. (Hodgins et al., 1995) 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 (Hodgins & Pollard, 1997) further extended the work of (Hodgins et al., 1995) 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.

McKenna and Zeltzer (McKenna & Zeltzer, 1990) 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 (Raibert & Hodgins, 1991) used a similar approach but different motion controller. They fashioned the models from analyses of robots and real creatures. Numerical integration of the dynamic model and specific control algorithms are used to generate running and jumping (with a ballistic flight phase) motions of multi-legged imaginary creatures.

Van de Panne et al. (Van de Panne et al., 1994) used a two-layered controllers system to simulate human walking. The top level controller consists of a finite state machine, which associates states to poses of the human figure. At the bottom level, the proportional-derivative controllers (PD controllers), which are placed at each joint, are used to compute the required forces and torques to lead the figure to the next pose (state) in the graph. Based on (Van de Panne et al., 1994), Laszlo et al. (Laszlo et al., 1996) later proposed a “limit cycle control” method, which will maintain the balance from given cyclic pose controllers. An extra control module computes the required torques to keep balance by regulating some variables according to the observed perturbations on motion. Komura et al. (Komura et al. 2004) animates new motions for bipedal locomotion by modifying the existing motion to maintain balance in response to external influence. All these three works (Van de Panne et al., 1994; Laszlo et al., 1996; Komura et al. 2004) focus on balanced motion and show limited types of interactions. They generate rather stable walking motions, however the result motions, in general, are not realistic enough for visual quality.

4.2 Discussion

Understanding the dynamics of limb motion give us the foundation to simulate human motion realistically. However, as we move from modeling the movement of limbs toward animating the entire body of an articulated figure, the complexity of the task is dramatically increased. This is because source of complexity of the whole body is much more than the sum of the parts, and controlling each limb to achieve coordinated goals or expressive qualities is extremely difficult.

The results of McKenna & Zeltzer, Raibert & Hodgins, and Van de Panne et al. 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.

Dynamics techniques may be used either for directly simulating the walk, or for adding constraints that enhance realism to a predefined motion. From the prospect of functionality, although recursive algorithms dramatically improve (for a 6-DOF limb, about 100 times faster) the computational efficiency over the old non-recursive ones for dynamics simulation, real-time dynamics simulation of complex articulated figures, such as human model still is a challenge to today’s computing hardware and control algorithms. Thus, real time simulation of human walking is difficult to achieve. This certainly will hamper animator’s artistic control of the motion.

For animator's artistic expression, since dynamics approaches are physical-based simulations, in the classical trade-off between "control" and "automation", these approaches sacrifice user's control for animated figure's automation. That is, once the simulating rules are set, the animated figure executes the task and reacts to the environment according to these rules, and the animator has no control over the motion. On the other hand, since physical laws are applied to animate the objects (characters), under proper setting of the controlling rules, realistic motion can be guaranteed. Thus, animator's desired motion, in realistic perspective, can be reasonably fulfilled by these methods. Animation principles regarding realistic issues, such as "arcs", "slow in and slow out", "follow through and overlapping", "timing", "appeal", and "secondary action", will come free with dynamics simulation. However, some important principles, which can't be done by dynamics simulation alone, must have user's intervening to break the physical limits for drama effects. For example, "anticipation", which is viewed as the pre-action of the main motion, will require the animator to explicitly add it in before the dynamics-simulated main motion; "Exaggeration", which is the primary drama factor in animation, might need the animator to scale up the magnitude of certain parts of the simulation.

5. Hybrid Methods

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.

5.1 Combination of Kinematics and Dynamics

In general, simplified dynamic models are applied to simulate some parts of articulated figure, such as the swing leg, supporting 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 (Armstrong & Green, 1985; Girard & Maciejewski, 1985; Issacs & Cohen, 1987; Bruderlin & Calvert, 1989; Wiley & Hahn, 1997; Shapiro et al., 2003) have implemented this technique in articulated figure animation.

Armstrong & Green (Armstrong & Green, 1985) and Wilhelms (Wilhelms, 1988) 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 kinematics means. For each individual link in the structure, one of the four kinematics control strategies was assigned to constrain its movement. Then, the system will generate required forces that work to exactly match the kinematically defined motions. Similar approach was proposed by Westenhofer and Hahn (Westenhofer & Hahn, 1996). Different from (Armstrong & Green, 1985) and (Wilhelms, 1988), dynamic is used to enhance kinematically created motion with realistic effects, instead of exactly matching it. Considering the realism is highly depended on the kinematics specification for (Armstrong & Green, 1985) and (Wilhelms, 1988), Westenhofer and Hahn's approach provides more flexibility in achieving natural continuous motion.

On the contrary to the hybrid systems above, which use kinematics methods to define the trajectory of the limbs, and then apply dynamics to calculate the result motion under the kinematics constraints, Girard's PODA (Girard & Maciejewski, 1985) applies simple dynamics to compute the motion of the body as a whole. Then the limbs are animated, using inverse kinematics technique, to make sure the foot stay on the ground during support, and the leg angles close to the desired angles.

Bruderlin and Calvert (Bruderlin & Calvert, 1989) 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, works for the cosmetics, and animates the feet, upper body, and arms kinematically to mimic the pattern observed in human walking.

5.2 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 (Badler et al., 1991).

Witkin and Kass (Witkin & Kass, 1988) 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 (Cohen, 1992) divided the original spacetime work into subsets or smaller spacetime windows, over which subproblems are formulated and solved. Liu et al. (Liu et al., 1994) proposed a hierarchical spacetime constraint paradigm to further lessens 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 suited for interactive human figure simulation.

Van de Panne (Van de Panne, 1997) 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 (Bruderlin & Calvert, 1989), "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 kinematics 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.

Safonova et al. (Safonova et al., 2004) parameterized motion within a low-dimensional space acquired from an existing motion capture database. Inverse kinematics is utilized to satisfy the constraints while the optimization function computes the motion that is close to the low-dimension space, satisfies the physical and user specified constraints and minimizes the user specified criteria such as energy consumption. Rather than using a database of motions in the same style, a single example motion is required to define a style in the model of Liu et al. (Liu et al., 2005). Their dynamic model incorporates several criteria of human locomotion derived from the biomechanical knowledge. Spacetime optimization framework with various parameters is used to define a range of human motion styles.

Aiming at simulating human walking in various virtual environments. Chung and Hahn (Chung & Hahn, 1999) 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 kinematics 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 (Cohen, 1992; Laszlo et al., 1996), 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.

5.3 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. Sims (Sims, 1994), Gritz and Hahn (Gritz & Hahn, 1995) 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.

5.4 Discussion

The combination of kinematics and dynamics provides a useful tool to augment realism for each simulation technique alone. Applying kinematics first to define the trajectories makes sure the articulated figure will follow the trajectories, but a successful simulation will highly depend on the animator's talent and experience to specify these trajectories. On the contrary, using dynamics method to define the trajectories, such as the overall trajectory of the body, and then applying kinematics to match these trajectories provides a more stable and flexible solution for animating human motion.

From the prospect of real-time simulation, the computation time needed for methods of kinematics-defined trajectories (kinematics-first) is not too much difference from pure dynamics approaches. This somehow makes these methods unfavored in time-critical applications, such as game, or virtual reality. Contrast to kinematics-first methods, dynamics-first approaches apply dynamics computation on simplified model to get the overall body motion. Simplification of human representation dramatically reduces dynamics computation, thus makes real-time simulation possible.

For the functionality of interactivity, the success of animated figure's interaction with the environment for kinematics-first approaches solely depends on the how well the biomechanical knowledge concerning human locomotion is embedded and parameterized in the kinematics part of simulation. For example, kinematics constraints coupled with biomechanical knowledge that lead to various styles of walking; Parameterizing these constraints and knowledge to achieve animated figure's automation. On the other hand, since dynamics-first approaches use simplified model, the inherent properties of physical-based simulation make it easy to achieve animated figure's automation in the environment. However, dynamics-first approaches most suffer the difficulties in simulating various walking styles. This is because dynamics-defined trajectories set new constraints to the later kinematics simulation, which is responsible to generate the desired poses in specific times.

Dynamics techniques pursue realism and automation on the expense of users' control. The performance of artistic expression mostly relies on animator's kinematics setup. Kinematics-first approaches allow the users define trajectories kinematically before dynamics simulation. But it's difficult for the animator to foresee how his artistic control can be preserved through the entire motion synthesis process. However, this 2-level control hierarchy mechanism provides the advantage of later on realism enhancement. For example, animation principle, secondary action, can be added on free body parts to achieve convincing auxiliary motion, such as limb swing. Dynamics-first approaches apply kinematics methods as the motion makeup. Before kinematics is actually applied, the result from dynamics simulation gives the template motion of articulated figure. Based on the template motion of main body, animator's artistic control is then added kinematically after, through user's intervening, or pre-defined postures corresponding to the current kinematics states.

In constraint optimization methods, the animator specifies what the virtual actor has to do. These requirements are mathematically expressed in formulas, together with Newton's laws, form the problem of constraint optimization. The solution to this problem is a physically valid motion satisfying the constraints. Since synthesized motion is based on dynamics simulation, computation time needed for constraint optimization methods is always a concern in time critical applications. The introducing of new constraints, expressed as formulas, further demands computation resources. Thus, real time simulation presents a difficult challenge for constraint optimization methods.

For artistic expression control, the animator sets the constraints which the embedded dynamics simulation has to satisfy and optimize. Based on proper constraints setup. Constraints optimization approaches are able to generate physically valid motion. However, to succeed in transmitting animator's artistic expression, two difficulties have to be overcome. First, what constraints to be applied to get the desired motion? For various walking styles, such as walking in frustrate mood or kidly walking, which constraints can lead to these dramatic motions respectively? Second, how to formulate the selected constraints to couple with embedded dynamics model? Taking these two difficulties into consideration, constraint optimization methods are rarely used in character animation, which in general demands animator's effort to tune-in the artistic expression.

Genetic programming defines a hyperspace containing an indefinite number of possible motions. The simulation of human walking on complex articulated figures is the most time-consuming among various motion control techniques. Real time simulation, so far, is impossible to accomplish. Compared to other methods, the most advantage of genetic programming comes from the ability to generate motions under harsh constraint situations. Among dynamics-related approaches, animated figure's motion is under the constraint of physics limits, and sometimes there is no solution can be attained to satisfy the severe condition. Genetic programming, being able to generate indefinite possible motions and behaviors, provides an alternative robust tool for constraint-driven articulated figure animation, such as locomotion in different environments.

Similar to the important role constraints play in directing the desired motion in constraint optimization techniques, fitness functions dominate the controlling of motion. The major challenge for animator to transmit their artistic expression using genetic programming is to determine the proper fitness measures and formulating them into evaluation functions. Furthermore, in order to automatically generate and select the wanted motions, the users have to sacrifice control for automation when using these approaches. Thus, make genetic programming rarely the animator's choice for character animation.

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 (Bruderlin & William, 1995; Perlin, 1995; Unuma, 1995; Witkin & Popovic, 1995) become the recent trend of human animation.

6.1 Motion Editing

Wiley and Hahn (Wiley & Hahn, 1997) 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. (Rose et al., 1998). 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 periodically motion, such as human locomotion.

Motion transition methods reuse motion clips and provide a smooth transition between two motion sequences. Commercial software, Endorphin (2005) (NaturalMotion, 2005), supports general transition from motion capture to simulation. Matching the final pose from simulation with motion capture, it provides a smooth transition for simulation to return to motion capture. Presuming dynamics simulation is only needed for a short time during which the forces change the character's state, Zordan et al. (Zordan et al., 2005) proposed a system which picks motion clip from the existing motion data after the interaction and computes new motion which tries to meet the physical states of the upcoming motion sequence. The selected motion clip not only is the animation following up but also serves as a reference for the interaction. Hence a smooth transition is assured.

Spacetime constraint techniques (Rose et al., 1996; Gleicher, 1997; Gleicher, 1998; Lee & Shin, 1999; Safonova et al., 2004; Liu et al., 2005) are also broadly adopted in motion editing systems. Gleicher (Gleicher, 1997) 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 constraints 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. (Rose et al., 1996) combined spacetime constraints and inverse kinematics constraints to generate transition between motion sequences. The motions of the supporting 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 kinematics constraints are satisfied during transition for the supporting 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 technique for motion transformation was also proposed by Gleicher (Gleicher, 1997), Lee and Shin (Lee & Shin, 1999). Gleicher (Gleicher, 1998) further extends (Gleicher, 1997) 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 retargetting method is a spacetime constraint 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 (Gleicher, 1997), 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 (Popvic & Witkin, 1999) describe a character-simplification methodology for mapping a motion between characters with drastically different numbers of degrees of freedom. Spacetime editing is 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.

6.2 Discussion

All the previous techniques manage mechanical, biomechanical or empirical knowledge of human locomotion to animate walking motion. Although dynamics simulation, based on physical laws, provides a realistic model for motion simulation, the complexity of the simulation model, coupled with unknown knowledge of human motion mechanism, makes motion capture of the real human actor the mainstream in simulating natural human motion. The acquiring of captured motion data usually has to go through a series of pipeline work, such as detailed planning, hardware setup, staff performing, and finally data editing. Logically make reusing of the collected motion data become the most popular research area in character animation.

From the prospect of real-time animation, since the simulation is based on either dynamics or spacetime methods, the computation time needed for motion editing approaches usually is expensive and unsuited for time-critical applications, such as virtual reality, or game. On the contrary, to generate realistic motion, captured motions are often used in pre-rendered animation appeared in games for visual delight

Artistic expression control has been always the controversial subject in character animation. This issue may origin from the fact that result motion mostly relies on actor's performance, instead of animator's skill and effort. The main function of motion capture is not only the exact data captured but also the spirit of motion performed by the actor. Once the motion is captured, there is not much room for the animator to work on without infecting the original motion properties. As a matter of fact, motion capture is the option for relieving animators' overload of control effort. Thus, it is rarely used in artistic character animation.

7. Discussion and Conclusions

In light of the recent surge of interest in virtual environment applications, many researches have devoted to solve the problems of manipulating human in the 3-D simulated worlds, especially human locomotion. However, most of the animation approaches based on these studies can only generate normal walking. For different walking style, it will require further user intervention, which can be a tedious job for the user, and are usually unable to produce continuous walking in real-time.

7.1 Observation and Potential Improvements

The difficulties in animating human walking mainly come from the complexity of the articulated human model and viewers' sensitive identification of erroneous motion. From observation and experiment results, several interesting observations may need to be further addressed, as they have been the important factors in finalizing our system design, and may be further studied to improve the walking model for better functionality and artistic expression:

- Footprint planning strategies: footstep planning plays an important role in walking. Based on the spatial information from step length and terrain status, stepping strategies provide the spatial information of the foot placement and free the user from detailing the placement of footsteps, which is critical toward building an interactive system. A potential improvement in computing the foot placement will be to factor-in the internal (duty factor) and external (gait period) temporal information of the steps based on factors such as walking attributes and terrain status. This will need further experiment and study.
- Hierarchical control: at the high level of control, template motions should be automatically generated based on several key parameters, such as walking route, locomotion speed...etc. As motion control goes deeper in the motion control hierarchy, more control is giving to the user, until each individual degree of freedom is accessible through the slider by the user. This feature, coupled with the efficient control algorithms, provides an interactive tool to animate desired motions, as it allows the user to adjust the animation attributes and examine the motion on the fly. A potential improvement to this will be to relate the animation attributes as a set with 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 scrambling these attributes through trial and error. We are looking at the possibility of deriving the walking attributes automatically from motion captured data.
- Realism improvement: the knowledge of human gait analysis and motion-capture data are brought 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 in real time. 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 relative static characteristic (due to the optimized objective- smoothness of leg joint movements) of the body movement becomes more noticeable. If some dynamic factors, such as momentum, can be quantitized to fit into the evaluation of body trajectories, more convincing results should be produced.

Functionality, such as automation, and artistic control is the classical trade-off in computer animation. In general, most of the research works in academia have focused on providing solution for specific functionality. On the other hand, entertainment applications always demand artistic expression, which naturally becomes the most wanted feature in mainstream commercial 3D software. Since human animation is a highly artistic task, an important research is to embed more artistic expression control in the animation framework. Hierarchical motion control, under such a perspective, provides a promising direction for future study.

7.2 Conclusions

The problems of simulating human walking have received considerable attention in computer animation and many published works have proposed different solutions. In this paper, we describe and survey the issues associated with motion control of human walking. We organize a wide range of approaches into four main categories: kinematics animation, dynamically-based animation, hybrid animation and motion-data edition.

First, kinematics methods which rely on a certain understanding of the basic walking motion mechanisms are discussed. The quality of the motion mainly depends on the quality and the quantity of knowledge and animator's effort necessary to reproduce the desired effect. Then, techniques based on controllers and dynamics simulation are described. Based on embedded dynamics simulation, these methods provide better solution for generating realistic motion. Nevertheless, the high computational cost makes them difficult to use in interactive animation. Hybrid animation techniques try to combine the strength of kinematics and dynamics approaches. This, when coupled with constraints optimization method, provide a general way for producing multi purposes animation. Finally, motion capture and editing techniques are described. Motion can be obtained with motion capture to constitute libraries of motion data. They can later be reused/modified and combined with editing tools to create a variety of specific animations.

Unlike most survey papers in computer animation field, which usually put weight on realistic or computational efficiency issues, this paper surveys motion control methods of human walking emphasized on artistic expression control. Table 1 summarizes the main advantages and shortcomings of the four categories of motion control techniques based on survey of artistic expression control and its related functionality.

Table 1. Comparison between the techniques.

	Artistic expression	Functionality
Kinematics	The highest control of artistic expression	Low computation cost -> real time simulation Styled walking Non-realistic Interactivity with environment
Dynamics	Low animation control of artistic expression	High computation cost -> non-real-time simulation Hard styled walking Interactive realistic motion with environment
Hybrid	Artistic control comes from either kinematics part or adding constraint.	Possible real-time simulation Styled walking on kinematics parts Interactive motion with environment
Motion editing	Artistic expression relies on motion performer. No artistic expression control from the animator.	Low computation cost, but hard interactive control. Difficult styled walking Constraint-based interactivity with environment

In conclusion, we believe that each control technique presented above has its advantages and shortcomings, and there is no single solution for all-purpose human walking. Each method should not be used alone, but rather combined according to the specific application which is under developed. A promising way to do this is to adapt the hierarchical level of motion control. Different control method is applied in specific level according to the motion requirements of that level. Thus, strength from various methods can be combined to produce desired animation.

References

1. Armstrong, W. W. and M. W. Green. 1985. “The Dynamics of Articulated Rigid Bodies for Purposes of Animation.” The Visual Computer 1(4): 231-240.

2. Badler, N. I., B. A. Barsky, and D. Zeltzer. 1991. “Making Them Move: Mechanics, Control, and Animation of Articulated Figures.” Morgan Kaufmann Publishers, Inc. CA, USA.

3. Boulic, N., M. Thalmann and D. Thalmann. “A Global Human Walking Model with Real-time Kinematic Personification. The Visual computer, 1990, 6, 344-358.

4. Bruderlin A. and T. W. Calvert. 1989. “Goal-Directed Dynamic Animation of Human Walking.” Computer Graphics 23: 233-242.

5. Bruderlin A. and T. W. Calvert. 1993. “Interactive Animation of Personalized Human Locomotion.” Graphics Interface 1993, pp: 17-23.

6. Bruderlin A. and L. William. 1995. “Motion Signal Procession.” Computer Graphics 29(4): 97-104.

7. Bruderlin A. and T. W. Calvert. 1996. “Knowledge-Driven, Interactive Animation of Human Running.” Graphics Interface 1996, pp: 213-221.

8. Calvert T. W. and J. Chapman. 1982. “Aspects of the Kinematic Simulation of Human Movement.” IEEE CG&A November 1982, pp: 41-50.

9. Chung S. and J. Hahn. 1999. “Animation of Human Walking in Virtual Environments.” Computer Animation’99: 4-15.

10. Cohen F. 1992. “Interactive Spacetime Control for Animation.” Computer Graphics 26(4): 293-302.

11. Denavit, J. and R. S. Hartenberg, 1955. “A Kinematic Notation for Lower-Pair Mechanism Based on Matrices.” ASME Journal of Applied Mechanics, Vol. 23, pp. 215-221.

12. Featherstone R. 1988. Robot Dynamics Algorithms. Kluwer Academic Publishers, Norwell, MA.

13. Girard M. and A. A. Maciejewski. 1985. “Computational Modeling for the Computer Animation of Legged Figured.” Computer Graphics 19(3): 263-270.

14. Gleicher M. “Motion Edition with Spacetime Constraints.” Proc. 1997 Symposium on Interactive 3D Graphics. Apr. 1997, pp. 139-148.

15. Gleicher M. 1998. “Retargetting Motion to New Characters.” Computer Graphics, 32(4): 33-42.

16. Gritz L. and J. Hahn. 1995. “Genetic Programming for Articulated Figure Motion.” J. of Visualization and Computer Animation, vol. 6: 129-142.

17. Grochow, K., Martin, S. L., Hertzmann, A., and Popovis, Z. 2004. “Style-Based Inverse Kinematics.” Computer Graphics 38(4): 522-531.

18. Hodgins J. K., W.L. Wooten, A.C Brogan and J.F. O’Brien. 1995. “Animating Human Athletics.” Computer Graphics, 29(4): 71-78.

19. Hodgins J. K., N. S. Pollard. 1997. “Adapting Simulated Behaviors for New Characters.” Computer Graphics, 31(4): 153-162.

20. Issacs P. and R. Cohen. 1987. “Controlling Dynamic Simulation with Kinematic Constraints, Behavior Functions and Inverse Dynamics.” Computer Graphics, 21(4): 215-224.

21. Ko H. and N. I. Badler. 1993. “Straight-line Walking Animation based on Kinematic Generalization that Perserves the Original Characteristics.” Graphics Interface ‘93, pp: 9-16.

22. Ko H. and N. I. Badler. 1996. “Animating Human Locomotion with Inverse Dynamics.” IEEE Computer Graphics and Applications, 16(2): 50-59.

23. Koga Y., K. Kondo, J, Kuffner and J. Latombe. 1994. “Planning Motions with Intentions.” Computer Graphics 28(4): 395-408.

24. Komura, T., Leung, H.. and Kuffner, J. 2004.”Animating Reactive Motions for Biped Locomotion.” Proc. Acm Symp. on Virtual Reality Software and Technology.

25. Laszlo J., M. van de Panne and E. Fiume. 1996. “Limit Cycle Control and Its Application to the Animation of Balancing and Walking.” Computer Graphics 30(4): 155-162.

26. Lee J. and S. Y. Shin. 1999. “A Hierarchical Approach to Interactive Motion Editing for Human-Like Figures.“ Computer Graphics 33(4):39-48.

27. Liu Z., S. J. Gortler, and M. F. Cohen. 1994. “Hierarchical Spacetime Control.” Computer Graphics 28(4): 35-42.

28. Liu, C. K. Hertzmann A. and Popovic Z. 2005. “ Learning Physics-Based Motion Styles with Inverse Optimization.” Computer Graphics 39(4).

29. McKenna M. and D. Zeltzer. 1990. “Dynamic Simulation of Autonomous Legged Locomotion.” Computer Graphics 22(4): 29-38.

30. Murray M. P., A. B. Drought, and R. C. Kory. 1964. “Walking Patterns of Normal Men.” J. of Bone and Joint Surgery 46A: 335-359.

31. NaturalMotion, 2005. Endorphin. www.naturalmotion.com.

32. Perlin K. “Real-Time Responsive Animation with Personality.” IEEE Trans. on Visualization and Computer Graphics, Vol. 1, Mar. 1995, pp. 5-15.

33. Phillips C. B., N. I. Badler, and B. L. Webber. Simulating Humans. Oxford University Press, 1993.

34. Phillips C. B. and N. I. Badler. 1991. “Interactive Behaviors for Bipedal Articulated Figures.” Computer Graphics 25(4): 359-362.

35. Popvic Z. and A. Witkin. 1999.”Physically Based Motion Transformation“ Computer Graphics 33(4):11-20.

36. Raibert M. H. and J. K. Hodgins. 1991. “Animation of Dynamic Legged Locomotion.” Computer Graphics, 25(4): 349-358.

37. Rose C. F., B. Guenter, B. Bodenheimer, and M. F. Cohen. 1996. “Efficient Generation of Motion Transition Using Spacetime Constraints.” Computer Graphics 30(4): 147-154.

38. Rose C. F., M. F. Cohen, and B. Bodenheimer. “Verbs and Adverbs: Multidimensional Motion Interpolation.” IEEE Computer Graphics and Applications, Sept. 1998, pp. 32-40.

39. Safonova, A., Hodgins, J. K., and Pollard, N. S. 2004. “Synthesizing Physically Realistic Human Motion in Low-Dimensional Behaviors-Specific Spaces.” Computer Graphics 38(4): 514-521.

40. Shapiro, A., Pighin, F. and Faloustsos, P. 2003. “Hybrid Control for Interactive Character Animation.” Pacific Graphics 2003, pp.455-461.

41. Sims, K. 1994. “Evolving Virtual Creatures.” Computer Graphics 25(4): 15-22.

42. Sims, K. and Zeltzer, D. 1988 “A Figure Editor and Gait Controller for Task Level Animation.”, SIGGRAPH Course Notes, 4, pp.161-181.

43. Sturman D. “Interactive Keyframing Animation of 3-D Articulated Models.” Graphics Interface ‘86, Tutorial on Computer Animation, 1986.

44. Thalmann N. M. and D. Thalmann. Computer Animation: Theory and Practice. Springer-Verlag, New York, 1990.

45. Thomas F., Johnston O., “The Illusion of Life.” Abbeville Press, New York, 1981.

46. Unuma M., K. Anjyo, and R. Tekeuchi. 1995. “Fourier Principles for Emotion-Based Human Figure Animation.” Computer Graphics 29(4): 91-96.

47. Van de Panne, R. Kim, and E. Fiume. “Virtual Wind-up Toys for Animation.” Graphics Interface, pp. 208-215, Banff, Alberta, Canada, May 1994.

48. Van de Panne M. 1997. “From Footprints to Animation.” Computer Graphics Forum, 16(4): 211-213.

49. Westenhofer B. and J. K. Hahn. 1996. “Using Kinematic Clones to Control the Dynamic Simulation of Articulated Figures.” Computer Graphics International.

50. Wiley J. and J. K. Hahn. “Interpolation Synthesis for Articulated Figure Motion.” Proc. Virtual Reality Annual Int’l Symp., IEEE CS Press, Mar. 1997, pp. 156-160.

51. Winter A. 1987. The biomechanics and motor control of human gait. Waterloo, Canada: University of Waterloo Press.

52. Wilhelms J. 1988. “Dynamics for Computer Graphics: A Tutorial”, Computing Systems, Winter, pp.63-93, 1988.

53. Witkin A. and M. Kass. 1988. “Spacetime Constraints.” Computer Graphics 22(4): 159-168.

54. Witkin A. and Z. Popovic. 1995. “Motion Warping.” Computer Graphics 29(4): 105-108.

55. Zelter D. 1982. “Motor Control Techniques for Figure Animation.” IEEE Computer Graphics and Applications, 2(9): 53-59.

56. Zordan V. B, Majkowska A., Chiu B., and Fast M., 2005. “Dynamic Response for Motion Capture Animation.” Computer Graphics 39(4).