Interactive Behaviors for Bipedal Articulated Figures

Cary B. Phillips
Norman I. Badler

Computer Graphics Research Laboratory
Department of Computer and Information Science
University of Pennsylvania
Philadelphia, Pennsylvania 19104-6389

Abstract

We describe techniques for interactively controlling bipedal articulated figures through kinematic constraints. These constraints model certain behavioral tendencies which capture some of the characteristics of human-like movement, and give us control over such elements as the figures' balance and stability. They operate in near real-time, so provide behavioral control for interactive manipulation. These constraints form the basis of an interactive motion-generation system that allows the active movement elements to be layered on top of the passive behavioral constraints.

Keywords: Interactive manipulation, inverse kinematics, articulated figures, balance, behavioral animation.

1. Introduction

In this paper, we describe techniques for interacting with two-footed articulated figures through kinematic constraints, concentrating on the movement of the feet and the center of mass. We particularly address the class of movements which are not bounded by dynamics. Such motions are typically executed at slow speed where inertial or frictional effects are minimal, and include standing, shifting the weight from one foot to the other, turning around, and taking small steps to the front, back, or to the side. In short, these motions encompass the types of movement which people act out while standing and moving but not actively locomoting from one place to another. We believe that this type of motion is of great importance to an animator, and we show how we describe these motions kinematically. We believe this approach provides superior control to dynamics techniques, particularly since these motions do not need the full complexity of a dynamic simulation.

Our movement primitives serve as the foundation for several higher level motion description mechanisms. Since they operate in near real time, they provide behavioral control for interactive manipulation. This allows the user to push, pull, and twist the figure interactively using our 3D direct manipulation interface, all while the figure maintains its balance. These primitives form the basis of an interactive animation system that allows the active movement elements to be layered on top of the passive behavioral constraints. Finally, these primitives provide the necessary interface to task level animation programs.

2. Background

Systems which provide goal directed motion have been used for the most part only on rather simple objects such as chains or mechanisms, and the available goals have been rather simplistic as well, such as point-to-nail or point-to-point constraints. Although such systems are very powerful for generating certain types of motion, they have not adequately addressed the problem of how an animator is to assemble a collection of goals which will accurately describe the intended motion [1]. Stating that such constraints will vary over time does not solve the fundamental problems of determining useful sets of constraints for human motion, negotiating their overlapping interactions, or organizing their timing for motion realism. In addition, they have not been successfully applied to highly articulated figures with expected behaviors. Flocking behavior-constraining functions have demonstrated particle motion within a global framework [5], but do not apply to articulated figures.

3. Articulated Figures and Inverse Kinematics

The techniques described in this paper are implemented as a part of *Jack*™, a multifaceted system for interactively modeling, manipulating, and animating articulated figures, principally human figures. *Jack* represents figures as collections of rigid segments connected by joints that may have arbitrary rotational or translational degrees of freedom. The model of the human figure that we use for the examples in this paper has 36 joints with a total of 88 degrees of freedom, excluding the hands and fingers. It has a torso consisting of 17 segments and 18 vertebral joints [2].

Jack uses an inverse kinematics algorithm that is based on a variable-metric optimization procedure, described in

† *Jack* is a trademark of the University of Pennsylvania. "Jack" is a nonsense name, not an acronym.

detail in [6]. This method uses the gradient descent approach to minimize the potential energy described by a set of constraints. Each constraint describes a desired geometric relationship between an *end effector* and a *goal* position or orientation in space. The algorithm is an iterative numerical procedure. At each iteration it computes the Jacobian of the input joint set, which relates the change in each joint angle to the change in total potential energy, which is weighted sum of the energy from each constraint. This determines a joint-space trajectory to follow which minimizes the total energy. The algorithm handles arbitrary numbers of constraints and arbitrary numbers of degrees of freedom. The constraints may overlap in the sense that a single joint may affect several constraints.

Generating Motion with Inverse Kinematics

Generating motion with inverse kinematics is somewhat different from constraint based systems that are based on dynamics. In particular, the only useful product of the inverse kinematics algorithm is the final position with the constraint energy minimized. The intermediate steps during the solution process should not be considered as "motion". To describe motion with inverse kinematics, we must select an appropriate set of end effectors and then describe the desired positions and/or orientations of these end effectors at each time increment. We then invoke the inverse kinematics algorithm at each time step to determine the set of joint angles that satisfies the desired relationships. The key to successfully describing motion through inverse kinematics is to choose properly the end effectors and then design sets of constraints that cause the figure to move in predictable patterns.

For the purposes of this paper, we consider the inverse kinematics algorithm as a black box that takes as input a set of constraints and a set of joints and returns with a set of joint angles that minimize the energy described by the constraints.

Constraints as Handles

We use the constraints as handles by which to control parts of the figure. We can make a loose analogy between this and a marionette puppet controlled by strings, except that our strings need not hang vertically, and they can twist and push as well as pull. How do we pull on the strings to get the figure to move as we want? How many strings do we need? Where should we attach them? We choose not to shape the goal-control mechanism into highly specific motion control elements to perform tasks like walking or running, but to design general purpose motion building blocks that stand by themselves as useful mechanisms of control.

We are not overly concerned here with the physical laws of nature but in capturing some of the global characteristics of human-like movement. We are willing to sacrifice some degree of Newtonian realism in order to achieve greater interactive control. We believe that a large portion of these characteristics can be captured through some simple behavioral tendencies, the most important of which are balance and stability. By phrasing these tendencies as figure *behaviors*, we can view the effect of the constraints in a more intuitive light.

4. Behaviors for Articulated Figures

The basic architecture of our system lets us treat time in one of two ways. We can "freeze time" and make postural adjustments to the figure through the real-time interaction mechanism. In this case, we can think of each iteration of the interaction as a time step. Alternatively, we can set up a series of primitive *actions* and then start the system time running from a certain point. The primitive actions cause the motion to take place.

The Feet

We begin by recognizing the importance of the support structure of the human body, i.e. its feet and legs and how they support the body's weight. As bipedal creatures, human beings have a built-in closed loop between the feet and legs that they are very good at manipulating. Unfortunately, because we model articulated figures as a hierarchy, we must take special care in modeling the connection between the feet and the ground. We do this by designating one foot or the other as dominant, and we the root the figure hierarchy through that foot. We hold the other foot in place by a constraint located at the ball of the foot. The orientation component of the foot constraint keeps the foot flat on the floor while allowing it to twist.

The Center of Mass and Balance

The center of mass is of critical importance because so many aspects of the movement through space of a human figure are dictated by the need to maintain balance. In addition, many types of movement, such as stepping and walking, involve intentional shifts in the center of mass away from the support polygon, followed by actions of the feet and legs to restore the balance. We consider balance as one of the most significant behaviors to model in a human figure, both the ability to maintain it and the ability to deviate from it.

We model balance in the figure as a constraint on the center of mass to remain vertically above a point in the support polygon. We associate the center of mass logically with the lower torso region of the figure, and we use this as the end effector of the constraint, with the ankle, knee, and hip joints of the dominant leg as the constraint variables. During the constraint satisfaction process at each time step, the center of mass is not recomputed. Since the center of mass belongs logically to the lower torso, its position relative to the torso remains fixed as the inverse kinematics algorithm positions the ankle, knee, and hip so that the previously computed center of mass point lies above the balance point. There are generally other constraints active at the same time, along with other postural adjustments, so that several parts of the figure assume different postures during the process.

After we solve the constraints, we recompute the center of mass. It will generally lie in a different location because of the postural adjustments, indicating that the figure is not balanced as it should be. Therefore, we must solve the constraints again, and repeat the process until the balance condition is satisfied. In this case the structure of the human figure helps. Most of the postural adjustments take place on the first iteration, so on subsequent iterations the changes in the center of mass relative to the rest of the body are quite minor. We measure the distance that the center of mass changes from one iteration to the next, and we accept the posture when the change is below a certain threshold. Although it is difficult to guarantee the convergence theoretically, in practice it seldom takes more than two iterations to achieve balance.

The Spine and Torso

Monheit [2] has developed a computational model for describing movements of the spine in terms of total bending angles in the forward, lateral, and axial directions. The technique uses weighting factors that distribute the total bending angle to the individual vertebrae in such as way that respects the proper coupling between the joints. Different weight distributions generate bends of different flavors, such as neck curls or motions confined to the lower back.

We have an optional behavior that holds constant the global orientation of the head. To model this type of behavior, we monitor the global orientation of the neck as the body posture changes at each time step. We measure the difference in euler angles between the current and desired neck orientation, and then apply these rotations to the spine.

The Pelvis

The pelvis connects the lower part of the spine to the upper legs. This is the general area of the center mass, so its position is governed primarily by the center of mass constraint. Therefore, our constraints on the pelvis involve only its orientation. The passive behavior of the pelvis involves holding its current orientation. Because of its central location, manipulations of the pelvis provide a powerful control over the general posture of a figure, especially when combined with the balance and torso constraints.

5. Real-time Interaction

The real-time interaction mechanism is described in [3] and [4]. Using this facility, we can interactively move and rotate the goals of constraints around in space through a 3D direct manipulation technique which gets its input from a three button mouse. This mechanism provides a nice form of postural control, although it is not so good at choreographing complex *motions* interactively.

We provide the following types of interaction. Each of these corresponds to a *Jack* system command which allows the appropriate property to be manipulated interactively.

bend torso This follows the technique described in [2]. The center of mass constraint causes automatic postural adjustments in the legs. For example, if we bend the torso forward, the hips automatically shift backwards so that the center of mass remains over the same point. Figure 1 shows how the hips automatically shift backwards to maintain balance as we bend the figure forwards.

rotate pelvis This interactively changes the orientation of the constraint on the pelvis. We can rotate the pelvis forward and backward, side to side, or we can twist it vertically. The constraints on the feet keep them planted on the ground. For example, if we set up a constraint on the torso, and then rotate the pelvis forwards, the figure will automatically squat but keep its head up.

move center of mass To do this, we move the goal point for the center of mass constraint, which allows us to disturb the figure's balance. We can shift the center of forwards or backwards, side to side to concentrate the weight on one foot or the other, or up and down to make the figure squat or stand on its toes.

move foot One foot is always the dominant one, and it serves as the root of the figure hierarchy. The other foot is held in place by a constraint. We can interactively move either foot. The passive balance behavior can either hold the center of mass at a fixed point or allow it to float to a point between the feet. Figure 2 show the center of mass floating between the feet as we move the left foot backwards and to the side.

6. The Composition of Actions

The notion of *action* in *Jack* is a scripted change to a constraint controlling the body. An action has three distinct parts: its beginning, its application, and its termination. Each action has a distinct starting and ending time. Each action has its own set of constraints controlling part of the body. Its parameters control the velocity of the constraint's goal and the constraint's weight as a function of time. Through a windowed interface, we can create, modify, and delete actions and get a global picture of a movement sequence.

Our system has actions which correspond to each of the types of manipulation described in Section 5. Each action causes the appropriate body part to move to a specified position or assume a desired orientation. The user specifies the desired posture by moving the body part using the manipulation mechanisms. This may be changed interactively, so if an action does not have the desired affect it can be easily adjusted. The position and orientation of the goal of an action's constraint are interpolated between a starting and ending value. The starting value is the current position or orientation of the end effector when the constraint is activated.

Actions may overlap in time, even ones which control the same part of the body. Since each action has its own constraints, this simply means that during the period of overlap, there will be multiple constraints on that part of the body. This is handled automatically by the inverse kinematics algorithm. We must take special care to control the effect of constraints when this overlap occurs. If constraints die out abruptly, then their termination may cause discontinuities in the motion of the figure. This may happen if a constraint is pulling part of a figure in a certain direction opposed to another constraint. In this case, the constraint should be phased out gradually rather than terminated instantaneously.

We allow the weight factor of each constraint to be a function of time. The weighting function associated with a constraint may increase, decrease, ease in and then ease out, or remain constant over the lifetime of the action. In practice, constant weights suffices when there are not many active actions. However, actions controlling the same part of the

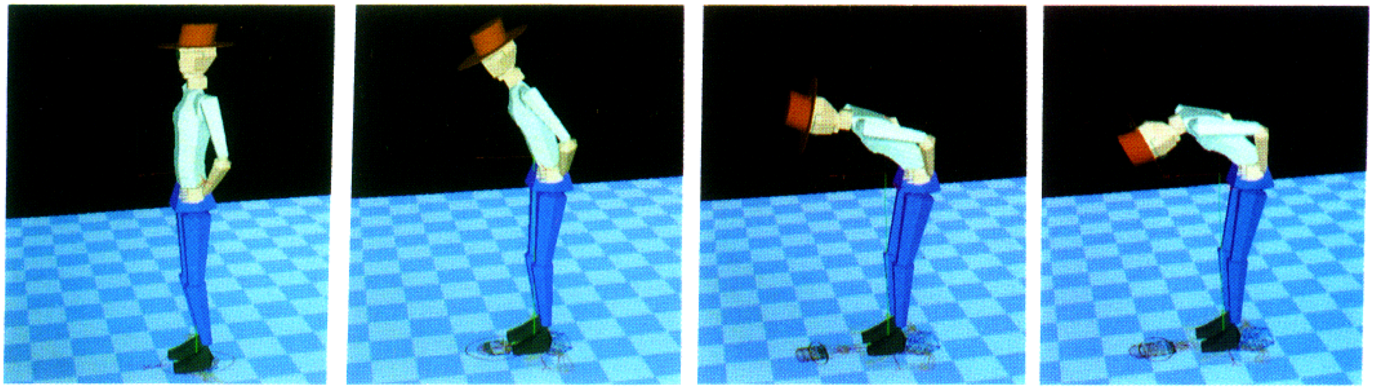

Figure 1: Interactively bending the spine

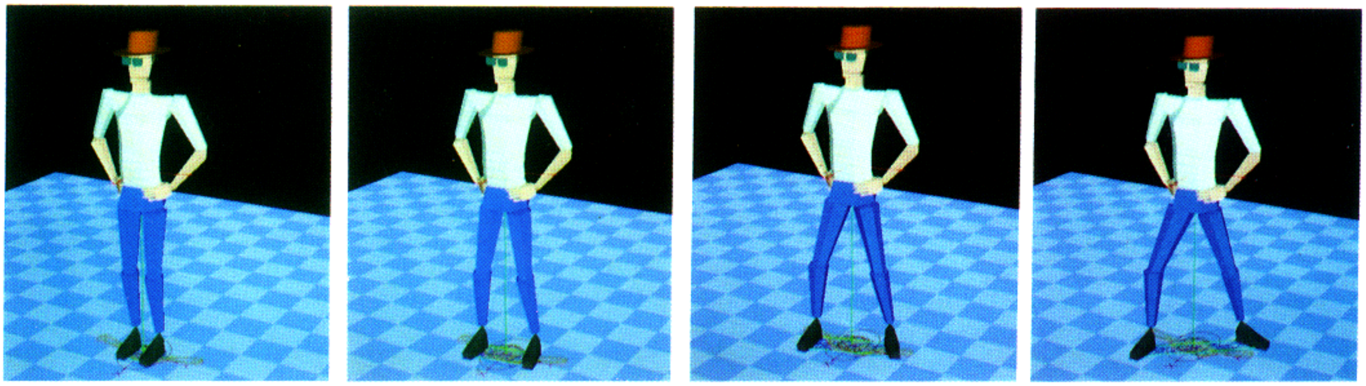

Figure 2: Interactively moving a foot

body that overlap in time should generally have the weight of the first action decay towards the end of its lifetime instead of remaining constant. It the current implementation of our system, it is up to the user to recognize this situation and set the weight functions accordingly.

7. Conclusions

We do not expect these elements alone to automatically generate realistic-looking human movement. Our approach has been to develop a general purpose set of movement elements which have specific effects. Some effects are local, such as moving a foot or raising a heel, while others are global, like maintaining balance or keeping the torso vertical. Taken together, these elements allow us to compose movement sequences of a quite general nature. In the future, we will consider how to automatically generate sequences of these actions to provide more "macro-like" control.

Acknowledgments

This research is partially supported by Lockheed Engineering and Management Services (NASA Johnson Space Center), NASA Ames Grant NAG-2-426, NASA Goddard through University of Iowa UICR, FMC Corporation, Martin-Marietta Denver Aerospace, Deere and Company, Siemens Research, NSF CISE Grant CDA88-22719, and ARO Grant DAAL03-89-C-0031 including participation by the U.S. Army Human Engineering Laboratory and the U.S. Army Natick Laboratory.

References

[1] Alan H. Barr, "Teleological Modeling," in *Making Them Move: Mechanics, Control, and Animation of Articulated Figures.* N. Badler, B. Barsky, and D. Zeltzer (eds.):315–321, Morgan-Kaufmann, 1990.

[2] Gary Monheit and Norman I. Badler, "A Kinematic Model of the Human Spine and Torso," *Computer Graphics and Applications*, Vol. 11, No. 2, March, 1991.

[3] Cary Phillips, Norman I. Badler, "Jack: A Toolkit for Manipulating Articulated Figures," *ACM/SIGGRAPH Symposium on User Interface Software*: 221–229, Banff, Canada, 1988.

[4] Cary Phillips, Jianmin Zhao and Norman I. Badler, "Interactive Real-time Articulated Figure Manipulation Using Multiple Kinematic Constraints," *Computer Graphics* 24(2):245–250, 1990.

[5] Craig W. Reynolds, "Flocks, Herds, and Schools: A Distributed Behavioral Model," *Computer Graphics*, 21(4):25–34, 1987.

[6] Jianmin Zhao and Norman I. Badler, "Real Time Inverse Kinematics with Joint Limits and Spatial Constraints," Technical Report MS-CIS-89-09, Department of Computer and Information Science, University of Pennsylvania, Philadelphia, PA, 1989.