Animated Human Agents with Motion Planning Capability for 3D-Space Postural Goals

Moon-Ryul Jung (moon@pluto.ai.kyutech.ac.jp)
Department of Artificial Intelligence
Kyushu Institute of Technology, Iizuka 820, JAPAN.

Norman Badler (badler@central.cis.upenn.edu)
Department of Computer & Information Science,
University of Pennsylvania, Philadelphia, PA 19104-6389, USA.

Tsukasa Noma (noma@pluto.ai.kyutech.ac.jp)
Department of Artificial Intelligence
Kyushu Institute of Technology, Iizuka 820, JAPAN.

Abstract

In this paper, we present a method for an animated human agent to construct motion plans to achieve 3D-space postural goals, e.g. a goal of a hand, while avoiding collisions. We use the potential field approach by providing mechanisms to handle the problem of local minimum. Given a conjunctive goal of multiple control points on the body, the potential field approach tries to minimize the objective function typically defined to be a weighted sum of individual goals. The local minimum problem arises as the planner tries to locally minimize the weighted *sum* of individual goals, even when multiple goals of the control points do not conflict with each other in 3D space. Our approach handles this problem by trying to achieve multiple goals individually, not by means of a weighted sum. To do so, the planner uses a qualitative kinematic model, which specifies what joint motions move what body parts in which directions in 3D space. The model is used to suggest joint motions for individual goals, and to explicitly detect and remove conflicts between the suggested joint motions. The local minimum problem arises more obviously when the original goals and collision-avoidance constraints, i.e. repulsive potential fields due to obstacles conflict with each other in 3D space. Our approach avoids this conflict by finding intermediate postural goals of the endangered body parts, based on the kinematic simulation of the current plan.

1 Introduction

As well documented in [1], the animation of human motion has a wide variety of applications, e.g. entertainment, ergonomic studies, computer-aided design of human workspaces, and generation of simulated or virtual worlds. To facilitate these applications, it is desirable to use *task-level* animation [29, 30] where the user is required only to specify spatial goals of the body instead of precise descriptions of joint motions. The task-level animation consists of two steps, motion planning and motion execution. The motion planning step generates a motion plan that would achieve a given postural goal without collisions, and the motion execution step supervises the execution of the motion plan until the goal is achieved.

In the area of computer animation, the problem of motion planning is not addressed sufficiently in the existing animation literature. It is partially addressed by Ridsdale and Calvert [24] and Renault, Thalmann, and Thalmann [23]. They handled, in essence, collision-avoidance of point-bodies or walking agents, but did not consider the problem of coordinating body parts to achieve spatial goals of body parts while avoiding collisions. The problem of motion planning for articulated bodies has been extensively addressed in robot motion planning field as surveyed by Hwang and Ahuja[9]. However, the two typical approaches to motion planning, i.e. the configuration-space and potential field approaches do not work well for the present problem. The human body model used in this study is anthropometrically realistic [31, 14, 19, 20, 21], and possesses 71 bones and 70 joints. It has 88 joint degrees-of-freedom not counting fingers, and also massively redundant. Therefore, both approaches must face the degrees of freedom problem, that is, how to control the many and redundant degrees of freedom. The majority of robot motion planning methods belong to the configuration space approach [15, 16, 17, 5]. This approach assumes that the goal configuration is known in terms of joint angles, and searches for in-between joint motions in the joint space. The human body is massively redundant and so a 3D-space postural goal specifies the goal-state configuration only partially in the joint space. In the configuration-space approach, therefore, 3D-space postural goals do not provide enough information to initiate planning. Moreover, this approach searches for a collision-free joint trajectory in the joint space whose dimension is equal to the degrees of freedom. So the approach is intractably expensive for bodies with many degrees of freedom.

In the artificial potential field approach [2, 11, 18, 3], given a conjunction of multiple spatial goals of control points on the body, the planner tries to locally minimize the objective function typically defined to be a weighted sum of individual goals. This approach is more appropriate to the present problem, because it is computationally tractable and does not assume a unique goal configuration in the joint space. The potential field approach, however, suffers from a critical drawback of local minimum. The local minimum problem arises as the planner tries to locally minimize the weighted *sum* of individual goals, even when multiple goals of the body do not conflict with each other in 3D space. But a more serious local minimum arises because of the competition between goals of body parts and collision-avoidance constraints. This approach immediately generates repulsive potential fields when body parts are in danger of collisions. Then it tries to achieve the original postural goal subject to the repulsive potential fields by local minimization. When the original goal and the repulsive potential fields interfere with each other by pushing a body part to move in the opposite directions, the local minimization leads the body to reach a local minimum.

In trying to come up with an appropriate method for the present problem, the configuration-space approach is precluded because it cannot handle 3D-space postural goals in the case of a body with redundant degrees of freedom. In a certain sense, our approach belongs to the potential field approach. In the standard potential field approach, however, repulsive potential functions are generated by a fixed scheme which is very local. In our approach, repulsive potential functions are generated and updated by means of explicit reasoning, which consists of plan postulation based on a global means-ends motion model and plan evaluation via a kinematic simulation of the current motion plan. Our approach is summarized as follows. First, to handle partially known goal configurations specified by spatial conditions of body parts, we use a qualitative means-ends model that captures kinematics of the body approximately. This model is called the *qualitative kinematic model* and specifies what joint motions move what body parts in which directions in 3D space. Relative to this model, 3D-space postural goals provide enough information needed to guess at relevant joint motions. Moreover, the qualitative kinematic model helps the planner select only relevant joint motions for a given goal. Without such a means-ends model, the planner should examine every joint to check the possibility of its contribution to a given goal. The use of a means-ends model makes our approach an application of the idea and techniques of rule-based AI planning [4, 6, 25, 26] to motion planning. Second, when the current plan is postulated, the joint motions in it have unbound angle parameters. The qualitative kinematic model is not precise enough to determine the angle parameters. Therefore, the angle parameters of the joint motions are bound by simulating the future behavior of the body according to the differential kinematic equations. It means that the present approach does not ignore the quantitative aspect of the structure and motion of the body. It uses two levels of kinematic models: a qualitative kinematic model and a differential kinematic model. Third, our approach handles the interference between goals and collision-avoidance constraints by finding intermediate postural goals of the endangered body parts, so that, starting from the new situation in which they are achieved, the original goal would not interfere with the collision-avoidance constraints. To find an intermediate goal, the planner simulates the trajectory of the body due to the current plan, and thereby discover the deviations of body parts from the desired free spaces. The deviations suggest intermediate goals which the body parts should have satisfied.

In this study, we assume that the body is in the vicinity of target positions of body

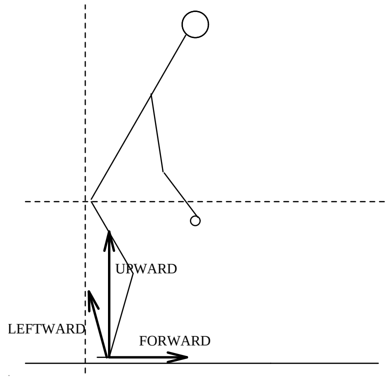

Figure 1: The Body Coordinate Frame

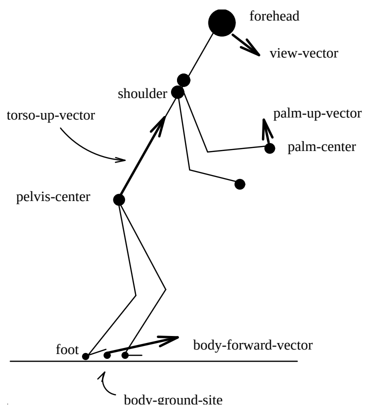

Figure 2: Control Points and Vectors: They are used to specify postural goals and motions.

parts and obstacles are convex or concave polygonal objects. The paper considers only the geometric aspect of motion, so the generated motion may not be feasible from dynamics point of view.

2 The Human Body

First of all, the structure of the human body is defined to make our discussion concrete. The human body model used in this paper is one developed as part of $Jack^{TM*}$. Geometrically, this body model consists of a set of rigid segments that are linked into a tree-like structure. The model possesses 71 segments and 70 joints (136 degrees of freedom). There are 88 joint degrees-of-freedom in our body model not counting fingers. Adjacent segments can be rotated about the joint axis connecting them. Their

* *Jack* is the trade mark of the University of Pennsylvania for its software of Human Movement Simulation, developed at the Computer Graphics Research Lab.

relative orientation is determined by the rotational transformation between the two local coordinate frames attached to the appropriate sites on the segments.

For motion planning, objects and positions in the environment are represented relative to the body coordinate frame defined at the left foot, as shown in Figure 1. The axes of the frame are aligned with the leftward, upward, and forward directions of the bounding box of the body, so that the frame changes only when the forward orientation of the body changes. They do not change even when the body bends at waist or bends at knees.

Postural goals of the body can be specified in terms of joint angles. However, such postural goals are too precise to be useful for motion planning. Above all, such precise goals may be not known at all. We, therefore, specify postural goals in terms of higher-level parameters called the *control parameters*. They consist of the *control points* and *vectors* as shown in Figure 2. The control points are defined so that relevant behaviors of the body can be described by the trajectories of the control points. We assume that there are no useful postural goals that cannot be specified by various values of the control points and vectors. An orientational goal is specified by the control vector and a given goal vector, both of which are the unit vectors. The orientational goal is achieved by aligning the control vector with the goal vector. This aligning process can be reduced to decreasing to zero the distance between the end point of the control and the end point of the goal vector. Here the goal vector is the one obtained by translating the original goal vector so that its origin is the same as that of the control vector. So, we will explicitly consider only positional goals of control points in this paper.

3 A Qualitative Kinematic Model

Here we will describe a qualitative kinematic model which allows means-ends analysis for given goals and thereby overcomes the problem of unguided and local decision-makings of the potential field approach.

3.1 Joint motions and component motions

Goals of control points are achieved by means of joint motions. So, joint motions are first defined. We use the expression

rotate(ControlVector, Joint, Axis, Ang)

to represent a rotational motion of the joint *Joint*. At the joint *Joint* a local coordinate frame is defined relative to which the motion is viewed. The expression means that the vector *ControlVector* whose origin is at the *Joint* rotates about the rotation axis *Axis* by the angle *Ang*. We assume that a joint with one degree of freedom has two rotation axes and the rotation angle about each axis is always specified in the positive direction, according to the right-hand rule. For example, the elbow joint has two rotation axes, *leftward-elbow-axis* and *rightward-elbow-axis.* Here the *leftward-elbow-axis* refers to the elbow axis that is parallel to the leftward axis of the body in the situation where the arms are fully stretched down and the palms are facing forward. The joint at the pelvis-center has three degrees of freedom, and thus six rotation axes. Three of them are *forward-pelvis-axis*, *leftward-pelvis-axis*, and *upward-pelvis-axis.*

The joint axes are fixed to the body part on which they are defined. As an example of a joint motion, *rotate(torso-up-vector, pelvis-center, leftward-pelvis-axis, A1)* means that the torso-up-vector rotates about the axis *leftward-pelvis-axis* by the angle *A1*. The joint motion *rotate(right.lower-arm, right.elbow, leftward-elbow-axis, A2)* means that the lower arm vector from the right elbow rotates about the leftward axis of the elbow by the angle *A2*. As another example, the joint motion *rotate(body-forward-vector, body-ground-site, upward, A3)* means that the body-forward-vector at the body-ground-site rotates about the upward axis of the body. Although there is no actual joint for this rotation, we use it to hide the details about orienting the body, which involves stepping motions in a complex way. There are 20 joint motions modulo angle parameters[10]. We assume that the body-ground-site is a translational joint and can be translated by itself relative to the world origin as the base frame.

The goal of the body is specified as a conjunction of individual goals of control points. Let $G_j(\mathbf{r_i})$ be a goal of a control point $\mathbf{r_i}$. It represents either a primary goal originally given to the body, or a secondary goal of collision-avoidance discovered during motion planning. Then the body is to achieve the conjunctive goal of individual goals $G_j(\mathbf{r}_i)$'s, by means of joint motions. Let a positional goal of a control point $\mathbf{r_i}$ be represented by *positioned-at(* $\mathbf{r_i}$*, Pos)*, where Pos is a goal position. Let G_{ix}, G_{iy}, and G_{iz} be the component vectors of the vector from the control point $\mathbf{r_i}$ to Pos. Then the goal *positioned-at(* $\mathbf{r_i}$*, Pos)* is decomposed as

$$move(\mathbf{r_i}, x, G_{jx}) \wedge move(\mathbf{r_i}, y, G_{jy}) \wedge move(\mathbf{r_i}, z, G_{jz}).$$

Each $move(\mathbf{r_i}, dir, G_{j,dir})$, $1 \leq i \leq m$, is called a *component motion.* It means that the control point $\mathbf{r_i}$ should be moved away from the current position by the distance $G_{j,dir}$ in the direction *dir*. Here *dir* is one of *leftward, rightward, upward, downward, forward* and *backward.* These directions are parallel to the axes of the body coordinate frame as shown in Figure 1. The decomposition makes the process of modeling kinematics of the body feasible. We need only to define the relationships between all the joint motions and the finite number of the component motions of the control points.

3.2 Motion Dependencies

To achieve goals of control points, we need to find out means-ends relations between joint motions and motions of control points. The relations are specified between joint motions and component motions of control points along the axes of the body coordinate frame. We will call these relations *motion dependencies.* They are qualitative in that they ignore the exact values of the angles of joint motions and the distances of

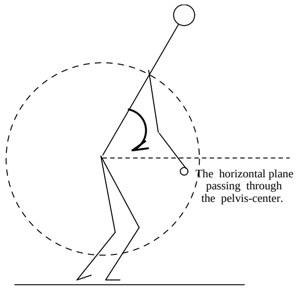

Figure 3: When the shoulder is above the horizontal plane passing through the pelvis-center, bending the torso-up-vector moves the shoulder forward.

component motions. They capture only the directional dependencies between joint motions and component motions. Figure 2 reveals the relationships between joint motions and component motions of control points. See [10] for details.

Now we show an example of motion dependencies. Consider how the component motion *move(right.shoulder, forward, D2)* is to be achieved. Given the initial situation as shown in Figure 3, we can see that the forward component motion of the right shoulder can be achieved by rotating the torso-up-vector at the pelvis-center about the *leftward-pelvis-axis*, and moving the pelvis-center forward. This relationship is described as follows:

(1)
rotate(torso-up-vector, pelvis-center, leftward-pelvis-axis, A1)
aslongas *above(right.shoulder,plane-normal-to(upward, pelvis-center))*,
move(pelvis-center, forward, D1)
$\Rightarrow$ *move(right.shoulder, forward,* M^+*(A1,D1))*.

The motions on the left hand side are called the *contributors* to the given component motion. The **aslongas** clause says that as long as its condition holds, the mentioned contributors produce the effect of moving the shoulder forward. The condition *above(right.shoulder, plane-normal-to(upward, pelvis-center))* means that the right shoulder is above the horizontal plane as shown in Figure 3 that is perpendicular to the *upward* direction and passes through the pelvis-center. If the condition holds, bending the torso-up-vector about the axis *leftward-pelvis-axis* contributes to moving the shoulder forward. Here M^+ refers to an unspecified monotonically increasing function [13]. Therefore M^+*(A1,D1)* refers to some distance that monotonically increases as *A1* and *D1* increase. To use the language of qualitative physics [12, 7, 13], the distances *A1*, *D1*, and M^+*(A1,D1)* are qualitative variables in that the rule does not specify exact quantitative relationships between them. The dependency rule (1) has captured the qualitatively distinct effect of rotating the pelvis joint, with respect to the forward component motion of the shoulder. To use the language of qualitative physics, the plane *plane-normal-to(upward, pelvis-center)* is a landmark value of the shoulder position for the rotation of the shoulder: If the shoulder is *above* the plane, the rotation moves the shoulder forward, but if the shoulder is *below* the plane, the same rotation moves the shoulder backward. There are more than 30 motion dependency rules. The qualitative kinematic model has been codified by a human designer who knows behavior of the body globally. See [10] for details.

3.3 Constraints on motion postulation

When there are multiple contributors for a given component motion as shown in the rule (1), their activation is subject to a *minimum motion* constraint. It prescribes that the base frames of joint motions, e.g. the shoulders, the pelvis-center, and the body-ground-site, are moved only when the joint motions relative to them are not sufficient to achieve given goals. Moving the base frames involves the rotations of joints below the base frames and thus activating more joint motions, which is desirable to avoid if possible. For example, suppose that the rule (1) is used. Only when the torso-up-vector rotation relative to the pelvis-center is not sufficient for the forward motion of the shoulder, the forward motion of the pelvis-center is activated. When multiple contributors are activated, their performance is subject to a *maximum concurrency* constraint. It requires that multiple contributors are to be performed in parallel when they are activated, with one exception. The exception is that relocating the body-ground-site of the body, i.e. walking, should be achieved before achieving the upper body motions if possible.

4 The Motion Planning Process

4.1 The Overall Flow

The overall control flow is depicted in Figure 4. The initial inputs to the planning are 3D space goals of control points and/or control vectors. Remember that orientational goals of control vectors can be reduced to positional goals of the end points of the control vectors. The planning process can be summarized as follows:

Planner:

1. Find a plan for the current goal, that is, contributors for the component motions for the current goal, relative to the current situation. This plan may cause collisions, but at the current moment collision-avoidance requirements are not known precisely enough. The planner first determines the extent of potential collisions by simulating the current plan, and then discovers specific collision-avoidance requirements, as described in the step 2.

2. Simulate the plan by incrementing the distance parameters of the contributors, *as long as* the contributors help the component motions make positive progresses in the desired directions. As shown in the control flow of Figure 4,

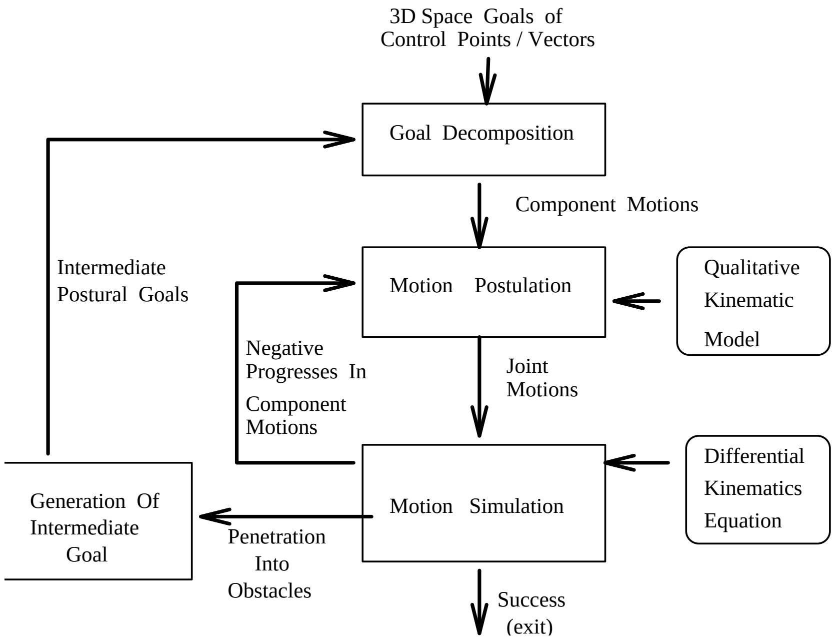

Figure 4: The Overall Control Flow of Motion Planning

when negative progresses occur in component motions, the motion postulation process is invoked again to activate a new set of contributors. See Section 4.3 for the role of the differential kinematic equation shown in the control flow. If the component motions for the current goal are achieved, accept the plan and **return**. At the end of the simulation, if there are body parts that have penetrated the surfaces of obstacles, find out their intermediate goals, that is, the desired free spaces into which they should have moved.

3. Retract to the previous situation from which the simulation has begun. Call **Planner** to find a subplan for the intermediate goals. Then call **Planner** for the original goal.

The detailed process of planning is quite complex. So we describe it by using an example goal. But we will provide principles underlying particular planning operations for the example goal. Consider a positional goal *positioned-at(right.palm-center, GoalPos)*. In the following, we will omit the qualifier "right" in the descriptions of control points. Suppose that the agent is standing just in front the table as shown in Figure 5. The side view of the initial situation is shown in Figure 6. The goal is first decomposed into a conjunction of component motions:

Plan 1:
move(palm-center, forward, F),
move(palm-center, downward, D),
move(palm-center, rightward, L).

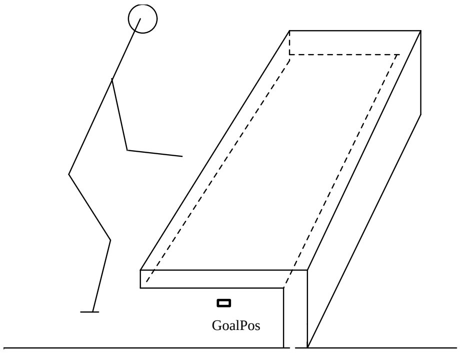

Figure 5: The Initial Situation

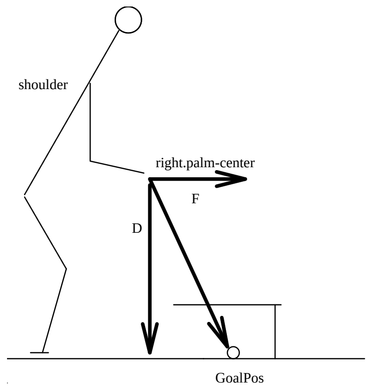

Figure 6: The Side View of The Initial Situation

Here parameters *F, D, L* are uniquely bound such that they are component vectors of the difference vector from the current position of the control point *palm-center* to the goal position *GoalPos*. To simplify the presentation, suppose that the rightward component L is zero, meaning that the position *GoalPos* is forward and downward relative to the initial position of the palm-center.

4.2 Motion Postulation

Once a set of component motions for the current goal is obtained, the motion postulation consults the qualitative kinematic model, and postulates contributors, that is, joint motions believed to achieve all the component motions. Here are some notions used to postulate contributors. Two contributors *conflict with* each other, if they are the rotations of the same joint but have opposite rotation axes. A contributor *hinders* component motions of other contributors if it has a side effect of causing the component motions to progress negatively, i.e. in the directions opposite to the desired ones. Whether or not a contributor hinders a component motion may be known during planning based on the qualitative kinematic model, or may be determined during the simulation of the current plan. A contributor *compensates for* a hindered component motion, if it helps the hindered component motion make positive progress so that the negative progress may be compensated for at least partially. Whether or not a contributor compensates for a hindered component motion is determined by consulting the qualitative kinematic model. A compensator for a hindered component motion can be activated if it does not hinder any other component motions or there are other contributors that can compensate for the hindered component motions. Using these notions, the means-ends reasoning can be described as follows:

1. For each desired component motion $move(\mathbf{r}_i,\ dir, G_{i,dir})$, $i = 1, ..., m$, find out their contributors relative to the current situation.

2. According to the minimum motion constraint, try to activate contributors starting from those further away from the root site of the body, for each component motion.

3. If two component motions have activated contributors that conflict with each other, deactivate the one whose component motion has another contributor, and activate that contributor instead.

4. If an activated contributor for a component motion hinders another component motion, see if there is a compensator for the hindered component motion. If so, maintain the activation of the hindering contributor by activating the compensator. Simply deactivating the hindering contributor is not recommended, because it has been postulated to achieve some component motion and there may be no other contributors that can replace it. If there are no compensators for the hindered motion, backtrack and see if there are contributors that can replace the hindering contributor without causing the hindrance. In general, at a given decision point if there are no ways to avoid the discovered hindrance, backtrack to the previous decision point and try another decision.

Now the means-ends reasoning is described using the goal of the right palm-center. Relative to the initial situation of Figure 6, the component motions of the palm-center in the plan (1) are expanded into their contributors as follows:

Plan 2:
(For the downward motion of the palm-center)
(a) *rotate(lower-arm, elbow, leftward-elbow-axis, D1)*
(b) *rotate(upper-arm, shoulder, leftward-shoulder-axis, D2)*,
(c) *move(shoulder, downward, D3)*
(For the forward motion of the palm-center)
(d) *rotate(lower-arm, elbow, rightward-elbow-axis, F1)*
(e) *rotate(upper-arm, shoulder, rightward-shoulder-axis, F2)*,
(f) *move(shoulder, forward, F3)*

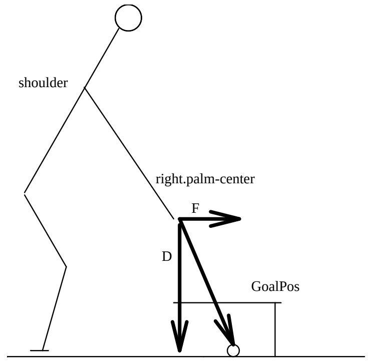

Figure 7: The Simulation of the Plan (3)

The motion (a) rotates the lower arm about the axis *leftward-elbow-axis* (in the positive direction) producing an extension motion of the elbow joint. The extension of the elbow joint causes the palm-center to move downward relative to the situation of Figure 6. The motion (b) rotates the upper arm about the axis *leftward-shoulder-axis* moving the upper arm downward relative to the situation of Figure 6. According to the minimum motion constraint, the motions (a) and (d) are activated. However, the motions (a) and (d) conflict with each other, having the opposite rotation directions. So, one of them should be deactivated. The motion (d) is deactivated because the motion (e) can replace it with respect to its role of moving the palm-center forward. Then we have the following plan:

Plan 3:
(For the downward motion of the palm-center)
(a) *rotate(lower-arm, elbow, leftward-elbow-axis, D1)*
(For the forward motion of the palm-center)
(e) *rotate(upper-arm, shoulder, rightward-shoulder-axis, F2).*

Note that this plan assumes that the shoulder and the control points below it do not move.

4.3 Simulating the current plan

When joint motions are postulated for the component motions of the current goal, the motion simulator simulates the behavior of the body by incrementing the angle parameters of the joint motions. The simulator should determine the joint rate, that is, how much to increment the angle of each joint at each discrete time point. Suppose that the current plan has activated n joints, $q = (q_1, ..., q_n)$, to achieve component motions $r_{i,dir}$'s. Here $r_{i,dir}$ refers to the motion of the control point r_i along the direction dir. The displacement $\Delta r_{i,dir}$ in $r_{i,dir}$ caused by the displacement Δq in the joint vector q is determined by the differential kinematic relation:

$$\Delta r_{i,dir} = (\frac{\partial r_{i,dir}}{\partial q_j})\Delta q.$$

Here the set of partial derivatives $(\frac{\partial r_{i,dir}}{\partial q_j})$ represents the rates that the displacement in each joint angle q_j contributes to the displacement in the component motion $r_{i,dir}$. These rates are dependent on the geometric configuration of the body at each discrete time point. The remaining question in plan simulation is to determine the joint rate Δq at each discrete time point, that is, how much to increment each joint angle. Lee and others[14] determine the joint rate Δq at a given time point, by means of force-related criteria. But they assumed that joint velocities and accelerations are small enough to be negligible, and considered only gravity forces. They represented the maximally exertable torque of each joint as a function of joint configuration, by using experimental data. They computed the available torque of each joint by subtracting the current torque of each joint from the maximum torque. Then Δq at each discrete time point is determined according to the available torques that each joint can exert at the time point. Greater the available torque greater the joint displacement. If the simulator uses dynamic simulation in the true sense, it should determine Δq according to force constraints that involve the joint velocities and accelerations as well. However, in this study which emphasizes on the use of a qualitative kinematic model we extremely simplify the problem of motion simulation. Without having any information for the joint rate, we use kinematic simulation in which each joint angle is uniformly incremented.

We also need to decide how long the joint motions of the current plan should be simulated. The simulation is continued until the current goal is achieved or there are component motions making negative progresses. A component motion makes negative progress when its control point moves backward in the component direction or it moves beyond its goal position in the component direction. Negative progresses in component motions do not necessarily imply that the current plan is not capable of achieving the current goals. The negative progresses detected may be only temporary and the control points may move toward the goal positions eventually. So, the planner might continue the simulation to see what happens. This strategy may be a good idea when there are no available contributors that can compensate for the negative progresses. But when such contributors are available, it seems to be safer to activate them and prevent the negative progresses. We call this policy the *local compensation policy*.

Let us see how this policy is applied to the current plan (3). When the plan (3) is simulated relative to the situation of Figure 6, the palm-center moves forward and downward for a while. But the body reaches a critical point from which a further simulation causes negative progress in the downward component, as shown in Figure 7. (The palm-center makes positive progress in the the forward component.) Following the local compensation policy, then, the planner examines the plan (2) and

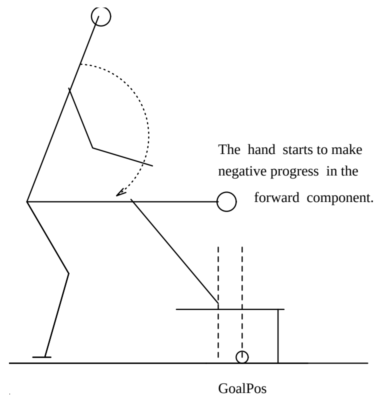

Figure 8: The Simulation of the Plan (4)

infers that the motion (c), that is moving the shoulder downward, can compensate for the negative progress of the palm-center in the downward component. According to the qualitative kinematic model, there are two potential ways to move the shoulder downward: moving the pelvis-center downward, and rotating the torso-up-vector about the leftward-pelvis-axis. The downward motion of the pelvis-center involves the activation of the knee joints. So, according to the minimum motion constraint, the planner chooses the torso-up-vector rotation, which involves less joint motions. Incidentally, the torso-up-vector rotation also contribute to the forward component of the palm-center goal. Adding the torso-up-vector rotation to the current plan (3), the following new plan is generated:

Plan 4:
(For the downward motion of the palm-center)
rotate(lower-arm, elbow, leftward-elbow-axis, D1)
rotate(torso-up-vector, pelvis-center, leftward-pelvis-axis, D2)
(For the forward motion of the palm-center)
rotate(upper-arm, shoulder, rightward-shoulder-axis, F2),
rotate(torso-up-vector, pelvis-center, leftward-pelvis-axis, D2).

Then, after retracting to the previous situation as shown in Figure 6 from which the plan (3) was simulated, the planner simulates the new plan (4). The simulation of the plan (4) reaches a critical point from which the torso-up-vector rotation would cause negative progress in the forward component of the palm-center motion, as shown in Figure 8. Even if the torso-up-vector rotation would help achieve the forward motion of the palm-center eventually, such information requires prediction too global to get under the local compensation policy. So, the planner assigns the torso-up-vector rotation only to the downward component motion of the palm-center, and activates a new motion that can compensate for the negative progress in the forward motion of the palm-center. According to the motion dependencies, the forward motion of the shoulder helps achieve the forward motion of the palm-center. Among the motions that are not yet activated, one that can contribute to the forward motion of the shoulder is the forward motion of the body-ground-site. So, the planner adds the body-ground-site motion to the plan (4), yielding the new plan (5):

Plan 5:
(For the downward motion of the palm-center)
rotate(lower-arm, elbow, leftward-elbow-axis, D1)
rotate(torso-up-vector, pelvis-center, leftward-pelvis-axis, D2)
(For the forward motion of the palm-center)
rotate(upper-arm, shoulder, rightward-shoulder-axis, F2),
move(body-ground-site, forward, F3).

While the plan (4) has the torso-up-vector rotation as a contributor to the forward motion of the palm-center, the plan (5) does not have it, because it has caused negative progress. The plan (5) needs re-arrangement. As an exception to the maximum concurrency constraint, the body-ground-site motion is placed before the other motions. So, the new contributor becomes a separate subplan:

Plan 6:
(For the forward component motion)
move(body-ground-site, forward, F3).

The subplan (6) precedes the subplan (7):

Plan 7:
(For the downward motion of the palm-center)
rotate(lower-arm, elbow, leftward-elbow-axis, D1)
rotate(torso-up-vector, pelvis-center, leftward-pelvis-axis, D2)
(For the forward motion of the palm-center)
rotate(upper-arm, shoulder, rightward-shoulder-axis, F2).

4.4 Binding distance parameters in sequential subplans

When a plan consists of two subplans and one subplan precedes the other, the situation produced by the preceding subplan becomes the initial situation of the following subplan. So the distance parameters of the preceding subplan should be determined so that the effect of the preceding subplan may help the following subplan achieve the original goals. We show how to do it by using the subplans (6) and (7). The planner wants to determine the distance parameter $F3$ of the subplan (6), so that the subplan (7) following it may achieve the forward component goal of the palm-center. The required displacement $\Delta F3$ from the current value of $F3$ is computed by the following steps:

1. Set $\Delta F3$ initially to zero.

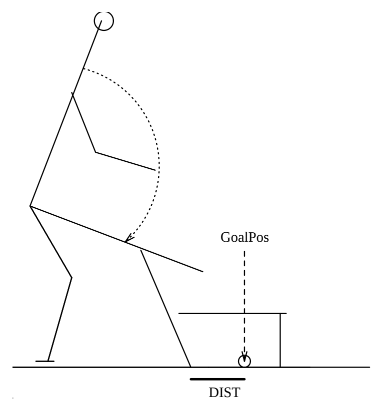

Figure 9: The simulation of the subplan (7) with $\Delta F3$ being zero in the subplan (6).

2. Simulate the subplans (6) and (7) in sequence. The subplan (7) is simulated as long as the downward motion of the palm-center makes positive progress. (Negative progress in the forward motion of the palm-center is ignored because the body-ground-site motion in the subplan (6) can compensate for it.) If negative progress occurs in the downward motion, the process of motion postulation is invoked again to suggest a new set of contributors, as shown in the control flow of Figure 4.

3. (Assume that the downward component motion of the palm-center has been achieved by the simulation of the subplan (7)) Find the difference *Dist* as shown in Figure 9 between the desired goal position of the palm-center and the actual position at the end of the simulation.

4. Assuming that the displacement *Dist* is linearly related to the displacement in the value of $F3$, find the displacement $\Delta F3$ that would nullify the displacement *Dist*.

5. After obtaining the new value of $F3$ by adding $\Delta F3$ to the old value, go to step (2).

Note that the above procedure is similar to the shooting method [22], a method for the two-point boundary value problem of differential equations. The difference is that here the simulation of motion is not governed simply by differential equations, but more complicated in that it involves activating or deactivating joint motions during simulation.

Determining the exact displacement in $F3$ to nullify the displacement *Dist* requires knowing the exact relationship between the change in that distance and the change in the forward component motion of the palm-center. This relationship is not linear in general, but the shooting method approximates it by a linear relationship J. To compute the linear approximation, the planner changes the body-ground-site

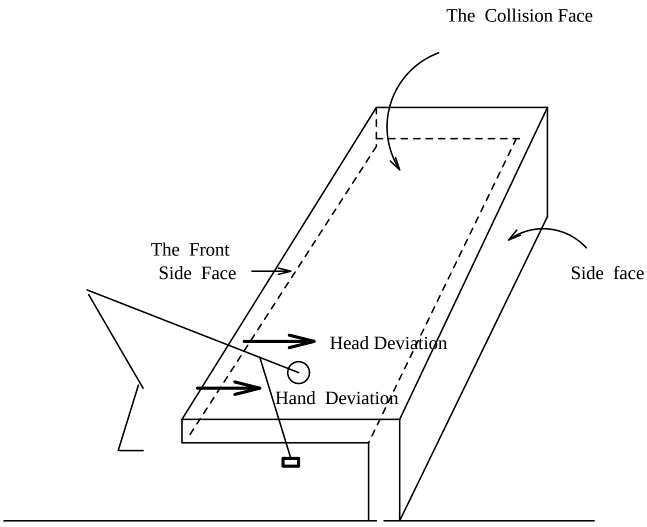

Figure 10: The simulation of the current plan has caused the head and the hand to penetrate the table. The simulated penetrations are used to find intermediate goals of the collision body parts.

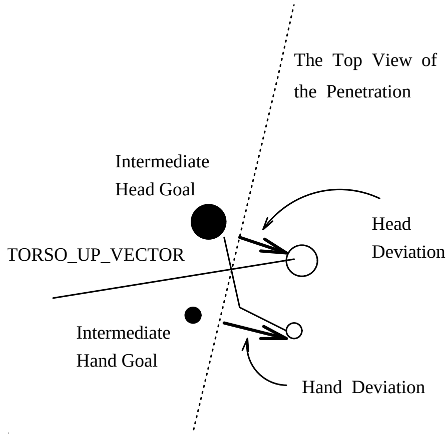

Figure 11: An intermediate goal position of the head is indicated by the dark circle. The motions to achieve it are postulated relative to the situation from which the current plan was simulated.

by a small amount and simulates the subplans relative to the new body-ground-site location. Then it finds out the difference between the old palm-center position and the new palm-center position. The ratio of this difference to the change in the body-ground-site location is used as the linear relationship J. Once the linear relationship is obtained, the displacement $\Delta F3$ is set so that it will cause the palm-center to move by the distance *Dist*, that is,

$$J * (\Delta F3) = -Dist.$$

But the value of $\Delta F3$ found this way may not enable the achievement of the goal of the palm-center. So, this process is repeated until that happens. However, it is significant to be able to suggest that the body-ground-site motion is a contributor to the forward component of the palm-center goal in the subplan (6). This is enabled by the means-ends reasoning based on the qualitative kinematic model. Without such a guidance, the shooting method must examine *every* control parameter to see if its change would contribute to the change in the forward motion of the palm-center.

4.5 Finding intermediate goals for collision-avoidance

The two subplans to achieve the goal of the palm-center were generated while ignoring collisions. In the current example, simulating the subplans causes the head and the right hand to penetrate the table as shown in Figure 10. The discovered penetration will be used as a specific collision-avoidance constraint for the next cycle of planning.

To avoid the collisions, the body should find an intermediate configuration which is collision-free and is also believed to enable the achievement of the palm-center goal. In this study, the planning strategy to find appropriate intermediate goals is not declarative and built into the planning program.

To find intermediate postural goals whose achievement would avoid the discovered collision, specific collision-avoidance requirements should be known. For this, we assume that objects in the environment are composed of convex polyhedra (objects may have concave parts). In the current example, during the simulation of the current plan for the right hand goal, the head and the hand penetrated the table top as shown in Figure 10. From this situation, the planner should determine the desired space into which the head and the hand should have moved. The desired space is specified by a set of half-spaces. Each half-space is defined by a *reference face* and a normal vector to it. The chosen half-space lies in the direction of the normal vector to the reference face, starting from the reference face. The reference face is used as a reference base to measure how much the collision body part has deviated from the desired half-space. In this study, the desired space for a collision body part is defined by two kinds of half-spaces. The first kind is called a *goal-containing* half-space. The reference face of the goal-containing half-space is the surface of an obstacle that the body part first hit. The normal vector to the reference face points to the side opposite to the place where the body part was located before the penetration. The half-space defined this way is one toward which the collision body part was moving. So it is reasonable to believe that this half-space contains a desired goal position of the body part. In the current example as shown in Figure 10, the goal-containing half-space is one below the table top surface. The second kind is called a *collision-avoiding* half-space. Consider a face of an obstacle that a body part first hit during the simulation of the current plan. Then a collision-avoiding half-space is a half-space whose reference face is a face of the obstacle that is adjacent to that face. The intelligent choice of a collision-avoiding half-space requires landscape information such as whether the body can fit in it or it is a dead-end. In this study, however, the planner simply picks up a collision-avoiding half-space that is compatible with the goal-containing half-space. In the current example, the planner chooses the one in front of the table, relative to the body as shown in Figure 11. So, the head and the hand should be positioned below the table top face and in front of the table. These are intermediate goals of the head and the hand that they should have achieved.

4.6 Achieving intermediate goals

It is reasonable to assume that the positions of the head and the hand obtained by the simulation of the current plan are good except for the collisions. So, as candidate intermediate positions to satisfy the intermediate goals, the planner chooses ones that cause the smallest deviations from the positions of the head and the hand in the simulated world. Intermediate positions of the head and the hand obtained this way are shown in Figure 11. To achieve the intermediate goals, the body first re-tracts to the previous situation as shown in Figure 5 from which the two subplans were simulated. The body, however, does not have to achieve the suggested intermediate positions of the head and the hand blindly. It should rather use them as guides for achieving the real intermediate goals defined by the two half-spaces. The planner generates a plan for the intermediate goals of the head and the hand, in the same way as it did for the original goal of the palm-center. Consider the goal of the head first. In the situation as shown Figure 5, the distance vector from the head to the intermediate goal position as shown in Figure 11 has the downward, forward, and leftward components. So, the body needs to achieve the three component motions:

(1.) *move(head, downward, Dist1)*,
(2.) *move(head, leftward, Dist2)*,
(3.) *move(head, forward, Dist3)*.

By consulting the motion dependencies, the component motions are translated into their contributors:

1. The downward component motion *move-by(head, downward, Dist1)* can be achieved by contributors:

   (1.a.) *rotate(torso-up-vector, pelvis-center, leftward-pelvis-axis, DAng1)*
   (1.b.) *move(pelvis-center, downward, D2)*.

2. The leftward component motion *move-by(head, leftward, Dist2)* can be achieved by contributors:

   (2.a.) *rotate(body-forward-vector, body-ground-site, upward, LAng1)*
   (2.b.) *move(body-ground-site, leftward, L2)*.

3. The forward component motion *move(head, forward, Dist2)* can be achieved by contributors:

   (3.a.) *rotate(torso-up-vector, pelvis-center, leftward-pelvis-axis, BAng2)*
   (3.b.) *move(body-ground-site, forward, B3)*.

Through motion postulation and simulation as described in the case of the original goal of the palm-center, the subplans for the intermediate goal of the head are obtained in the following order:

subplan 1: For the forward component of the head goal.

move(body-ground-site, forward, B1)

subplan 2: For the leftward component of the head goal.

rotate(body-forward-vector, body-ground-site, upward, LAng1)

subplan 3: For the downward component of the head goal.

rotate(torso-up-vector, pelvis-center, leftward-pelvis-axis, DAng1)

The distance parameters *B1*, *LAng1*, and *DAng1* are determined by the simulation of the subplans. The intermediate goal of the hand can be achieved by arm motions in addition to the contributors to the head goal. After achieving the intermediate goals of the head and the hand, the body constructs and performs a plan for the original goal of the palm-center.

5 Conclusion

We have devised a task-level collision-avoidance motion planning method for massively redundant articulated bodies. A prototype motion planning and animation system that employs the strategy of the paper is implemented on top of $Jack^{TM}$ animation system.

This study was motivated by the observation that the shortcomings of the configuration space approach, in particular the assumption of the complete goal configuration, were not compatible to our task of designing animated agents capable of receiving task-level commands. We started with the potential field approach without such drawbacks. But to use this approach, we should solve the problem of local minimum. The present approach has provided two ways to overcome the local minimum problem. First, conflicts in joint motions due to multiple goals are explicitly detected during both motion postulation and simulation. This has been greatly facilitated by the qualitative kinematic means-ends model. Second, the planner discovers intermediate postural goals that would help the body avoid collisions and move toward the given goal, by means of the simulated failure of the current plan.

Motion planning addressed in this study is still primitive. The present study did not consider coordination between vision and motion. It did not spell out how to choose collision-avoidance half-spaces intelligently. This study addressed only geometric and kinematic constraints of the body. It is also an open question how to combine qualitative kinematic model with dynamics-based simulation as described in [27, 8, 28, 14]. But a qualitative kinematic model is believed to handle the difficulty of control that dynamics-based simulation must face. Dynamic motion models of a body is typically described in terms of differential equations and thus specifies the behavior of the body at an instantaneous time. So information known about the behavior of the body is extremely local, and so finding appropriate control actions needed to achieve desired goals is not easy. It would require guessing at good control actions that would achieve desired behavior, and this is a search problem, which may be arbitrarily difficult. A kinematics-based qualitative motion model can facilitate this search process. The model suggests which joint motions can achieve desired behavior. The qualitative kinematic model designed in this study is flexible enough to be compatible with dynamics constraints. It only specifies which joint motions should be moved about which axis to achieve given component motions. In the present study, kinematic simulation is used to determine how much each joint should be moved, once joint motions are postulated based on the qualitative kinematic model. Similarly, the velocities and accelerations of the joint motions can be determined by dynamic simulation, once they are postulated based on the qualitative kinematic model.

Acknowledgments

This research is partially supported by Lockheed Engineering and Management Services (NASA Johnson Space Center), MOCO Inc., NSF CISE Grant CDA88-22719, and ARO Grant DAAL03-89-C-0031 including participation by the U.S. Army Human Engineering Laboratory, Natick Laboratory, TACOM, and NASA Ames Research Center. We thank the members of the ANIMNL (Animation from Natural Language) project at the University of Pennsylvania in which the present work was motivated and was discussed.

References

[1] Norman Badler, Cary Phillips, and Bonnie Webber. *Simulating Humans.* Oxford University Press, New York, 1993.

[2] J. Barraquand, B. Langlois, and J. Latombe. Numerical potential field techniques for robot path planning. Technical Report STAN-CS-89-1285, CS Dept, Stanford University, 1989.

[3] David E. Breen. Choreographing goal-oriented motion using cost functions. In N. Magnenat-Thalmann and D. Thalmann, editors, *State-of-the-Art in Computer Animation*, pages 141–151. Springer-Verlag, New York, NY, 1989.

[4] D. Chapman. Planning for conjunctive goals. *Artificial Intelligence*, 32:333–377, 1987.

[5] B. Donald. A search algorithm for motion planning with six degrees of freedom. *Artificial Intelligence*, 31:295–353, 1987.

[6] R.E. Fikes and N.J. Nilson. Strips: A new approach to the application of theorem proving to problem solving. *Artificial Intelligence*, 2, 1971.

[7] Kenneth D. Forbus. Qualitative process theory. *Artificial Intelligence*, 24:85–168, 1984.

[8] Mark Green. Using dynamics in computer animation: Control and solution issues. In Norman I. Badler, Brian A. Barsky, and David Zeltzer, editors, *Making Them Move: Mechanics, Control, and Animation of Articulated Figures.* Morgan-Kaufmann, San Mateo, CA, 1991.

[9] Y. K. Hwang and N. Ahuja. Gross motion planning - a survey. *ACM computing surveys*, 24(3):219 – 291, 1992.

[10] Moon Jung. *Human-like Agents with Posture Planning Capability.* PhD thesis, Computer and Information Science, University of Pennslvania, Philadelphia, PA, 1992. to appear.

[11] O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. *The International Journal of Robotics Research*, 5(1):90 – 98, 1986.

[12] Johan De Kleer and John S. Brown. A qualitative physics based on confluences. *Artificial Intelligence*, 24:7–83, 1984.

[13] Benjamin Kuipers. Commonsense reasoning about causality: Deriving behavior from structure. *Artificial Intelligence*, 24:169–203, 1984.

[14] Philip Lee, Susanna Wei, Jianmin Zhao, and Norman I. Badler. Strength guided motion. *Computer Graphics*, 24(4):253–262, 1990.

[15] T. Lozano-Perez and M. A. Wesley. An algorithm for planning collision-free paths among polyhedral obstacles. *Communications of the ACM*, 22(10):560–570, Oct. 1979.

[16] Tomas Lozano-Perez. Spatial planning: A configuration space approach. *IEEE Transactions on Computers*, c-32(2):26–37, Feb 1983.

[17] Tomas Lozano-Perez. A simple motion planning algorithm for general robot manipulators. *IEEE Journal of Robotics and Automation*, RA-3(3):224–238, June 1987.

[18] A. A. Maciejewski and C. A. Klein. Obstacle avoidance for kinematically redundant manipulators in dynamically varying environments. *Int. Journal of Robotics Research*, 4(3):109–117, 1985.

[19] G. Monheit and N. Badler. A kinematic model of the human spine and torso. *IEEE Computer Graphics and Applications*, 11(2):29–38, 1991.

[20] 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.

[21] Cary B. Phillips and Norman I. Badler. Interactive behaviors for bipedal articulated figures. *Computer Graphics*, 25(4):359–362, 1991.

[22] William Press, Saul Teukolsky, William Vetterling, and Brian Flannery. *Numerical Recipes in Fortran: The Art of Scientific Computing, 2nd edition.* Cambridge University Press, 1992.

[23] Olivier Renault, Nadia Magnenat Thalmann, and Daniel Thalmann. A vision-based approach to behavioral animation. *The Journal of Visualization and Computer Animation*, 1(1):18–21, 1990.

[24] G. Ridsdale and T. Calvert. Animating microworlds from scripts and relational constraints. In *Proc. of Computer Animation 90*, 1990.

[25] E. D. Sacerdoti. *A Structure for Plans and Behavior.* Elsevier North-Holland, Inc., New York, 1977.

[26] A. Tate. Interacting goals and their use. In *IJCAI-75*. 1975.

[27] Jane Wilhelms. Dynamic experiences. In Norman I. Badler, Brian A. Barsky, and David Zeltzer, editors, *Making Them Move: Mechanics, Control, and Animation of Articulated Figures*. Morgan-Kaufmann, San Mateo, CA, 1991.

[28] A. Witkin and M. Kass. Spacetime constraints. *Computer Graphics*, 22(4):159–168, 1988.

[29] David Zeltzer. Motor control techniques for figure animation. *IEEE Computer Graphics and Applications*, 2(9):53–59, Nov. 1982.

[30] David Zeltzer. Toward and integrated view of 3-d computer animation. *The Visual Computer*, 1(4), 1985.

[31] 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.