Pergamon

Comput. & Graphics, Vol. 18, No. 4, pp. 443–452, 1994
Copyright © 1994 Elsevier Science Ltd
Printed in Great Britain
0097-8493/94 $6.00 + .00

0097-8493(94)E0039-Z

Modelling, Rendering and Animation Techniques

GOAL-ORIENTED DESIGN AND CORRECTION OF ARTICULATED FIGURE MOTION WITH THE TRACK SYSTEM

Ronan Boulic,[†] Zhiyong Huang,[†] Nadia Magnenat Thalmann,[‡] and
Daniel Thalmann[†]
[†]LIG-Computer Graphics Lab, Swiss Federal Institute of Technology, Lausanne, CH-1015 Switzerland
[‡]MIRALab-CUI, University of Geneva, 24 rue du Général Dufour, CH 1211 Geneva, Switzerland

Abstract—An interactive tool is proposed for the visualization, editing, and manipulation of multiple track sequences. Multiple tracks sequences can be associated with an articulated figure and may retain motion issued from different motion generators such as walking, inverse kinematics, and key framing within a unified framework. The TRACK system provides a large set of tools for track space manipulations and goal-oriented corrections. This approach allows an incremental refinement design combining information and constraints from both the track space (usually joints) and the Cartesian space. We have dedicated this system to the design and evaluation of human motions for the purpose of animation. For this reason, we also insure the real-time display of the 3D figure motion. The interface design and the interaction device integration are realized with the Fifth Dimension Toolkit.

1. INTRODUCTION

Human animation is an active research field, including various approaches from traditional rotoscopy[1], key framing[2, 3, 4, 5], functional modeling[6, 7, 8, 9], direct and inverse kinematics[10, 11, 12], to physically based methods like direct and inverse dynamics[13, 14, 15], and space-time optimization[16, 17].

Integration of different motion generators is vital for the design of complex motion where the characterization of movement can quickly change in terms of functionality, goals, and expressivity. This induces a drastic change in the motion control algorithm at multiple levels: behavioral decision-making, global criteria optimization, and actuation of joint level controllers. By now, there is no global approach that can reconfigure itself with such flexibility.

The TRACK system has two major goals, first integrating a wide range of motion generators within the unified framework of multiple track sequences, and second, providing a large set of tools for the manipulation of these entities. Our tools are suited for the manipulation of motions associated with tree-structured articulated figures, especially virtual humans. We propose the following methodology (Fig. 1):

First, the output produced by any motion generator is adequately sampled in the joint space and each value is recorded and becomes a key value within a track.

Then, a large set of tools can manipulate the resulting multiple track sequences. The first is usually a compression filter used to reduce the key value to minimum within a predefined error rate.

Among these tools is also the Coach-Trainee method, which allows the kinematic correction of joint-space based motion with respect to Cartesian constraints[11, 12]. In such a way, it is still possible to modify the key frame sequence, a low-level description of motion, for a higher level goal-oriented requirement.

In the following sections, we first recall the basic features of the 3D hierarchy and the structure of the associated multiple track sequence. The functionality of the Fifth Dimension interface toolkit is reviewed concerning interface layout and interprocess communication. Then we develop the techniques used for motion manipulation in the track space: key value editing, time-base manipulation, compression, Coach-Trainee Correction and finally concatenation and blending.

2. BACKGROUND

This section reviews the basic models and techniques used by the TRACK system.

2.1. *Hierarchical 3D data structure*

The data structure used for human modeling is the first important building block. Some data structures, such as octree, CSG, and the PHIGS hierarchy, are interesting representations of 3D models and they have had many applications. Nevertheless, they have some limitations when applied to human animation[18]. Even a more recent approach like INVENTOR[21], which manages a powerful Direct Acyclic Graph (DAG), does not meet our needs. This is mainly because the information is organized in a procedural display list fashion that requires some traversal to get global information about any component of interest. Also, the DAG structure sets the motion propagation flow and obviously this tool has not been designed to deal with easy reconfiguration of the root of motion without changing the other informational context. On the other hand this organization optimizes the memory occupation with its implicit attributes inheritance mechanism. The following major trade-off and requirements have been considered in the design of our human model for animation:

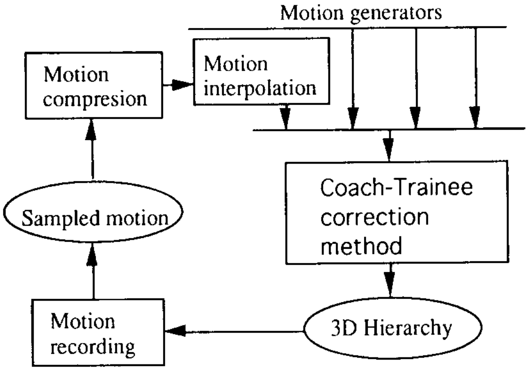


Fig. 1. The TRACK methodology.

1. Homogeneity vs. integration of information.
2. Calculus efficiency vs. memory space occupation.
3. Static vs. dynamic Topology for animation.

The hierarchy basic component, named the node_3D, is chosen to retain the geometric, topological, and display information sufficient to set a frame in a 3D space[18]. In a way similar to[22], an additional functionality can be associated to a node_3D through the binding of one typed structure. If a node_3D has no associated information it is said to be of NEUTRAL type. It can be useful as an intermediate frame to identify some functional location. In Fig. 2, each small black box represents a node_3D. The internal node_3D convey general purpose mobility information while terminal node_3Ds are more application oriented. This illustrates the design principle we have retained to deal with the first trade-off homogeneity-integration in the sense that two families of typed data have been defined. First, a small set of types are used to construct the internal and mobile components of the hierarchy (also called the skeleton of the hierarchy). The idea is to provide autonomous structures with respect to external information depending on applications. Then, all other information is considered as external to the skeleton and attached as terminal node_3D. Figure 3 shows the two node_3D families.

Apart from the NEUTRAL type, members of the hierarchy skeleton family are the followings. A JOINT type describes a one degree of freedom joint either transitional or rotational. The other mechanical joints can be constructed with a chain of such JOINT node_3D. A FREE type specifies an arbitrary geometric transformation. It is used for the animation of independent objects. Some specific modes have been designed to support flexible camera viewing and tracking.

Although terminal data is application-dependent, we can at least present two types. A FIGURE type is used to integrate information about geometrical modeling of surface information. We are currently using a triangular mesh based data structure to represent the human skin surface[20]. The whole human surface model has been cut into multiple parts attached to appropriate joint node_3Ds. Finally, a CAMERA type defines a perspective or parallel projection.

Concerning the second trade-off, we have definitely favored the calculus efficiency against memory space occupation in our conception. This mainly derives from our major requirement of low dependency of a node_3D with respect to other node_3Ds. Each node_3D has a local coordinate system and maintains both the direct and inverse transformation matrices to its parent node_3D frame. In addition to the local direct-inverse transformation, each node holds global direct and inverse transformation matrices with respect to the world coordinate system (the root of the hierarchy). It follows from these conception choices that a modification of the motion root, inducing a different motion propagation flow, can be managed ef-

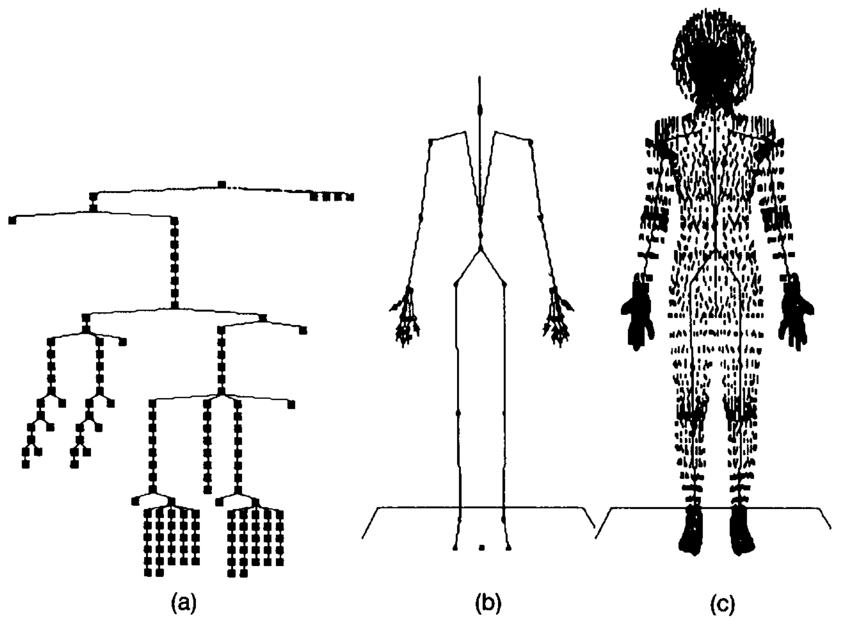


Fig. 2. (a) 2D display of a 3D hierarchy structure; (b) corresponding 3D wire frame display; (c) associated set of polygonal surfaces.

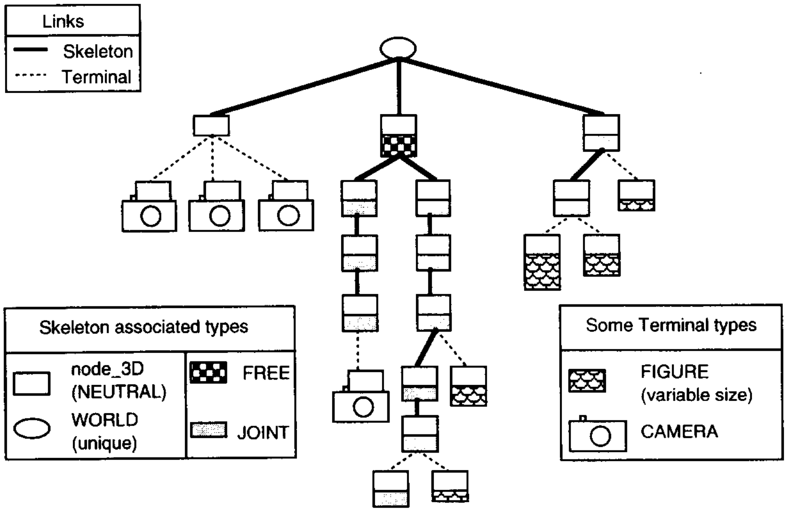


Fig. 3. A symbolic representation showing the hierarchy skeleton and some typed data.

ficiently without changing the topology of the hierarchy. So our approach to the third trade-off is to prefer a static view organization of the hierarchy with a dynamic motion propagation mechanism as advocated in[10]. The 3D hierarchy library is provided as a low-level software layer on which other animation software is built, like the motion generators and the Coach-Trainee Correction.

2.2. *Multiple track sequence*

Previous work using tracks in computer animation are MUTAN[2] and Twixt[3]. MUTAN (MUltiple Track ANimator) is an interactive system for independently animating three-dimensional graphical objects with synchronization of different tracks. More recent systems are mainly based on global posture interpolation as Compose[5], PODA[7], and PINOCCHIO[1]. Apart from key frame design[5], their motion generators can derive walking trajectories from a kinematic model[6, 7] or manage rotoscopy data[1].

Our approach is based on a close association between our multiple track sequence management and other motion generators resulting in the incremental goal-oriented design loop presented in Section 1. Basically, a sequence is associated with all or part of the 3D hierarchy but it can also work on a subset of special predefined variables. `JOINT node_3D` and `FREE node_3D` mobility information can be accessed to define key values within a track, or modified during interpolation. A track maintains a key value list. Each key value stores the value information and the first derivative with respect to time (also called the slope). The default mode evaluates the first derivative as the Cardinal slope, that is the chord slope built from the previous and next key value. All the key values belonging to any track point to the common time-base of the sequence. The user can always modify the value, the associated time and the slope for any key value, resulting in great freedom of motion design. Moreover, the track curve display provides a clear kinematic feeling of the motion. Finally, from the integration viewpoint, any sampled motion can be recorded, manipulated, and easily played back with standard cubic interpolation. Whatever method is used, the motion value and velocity can be recorded as a multiple track sequence and this constitutes the base of the TRACK system for complex motion design.

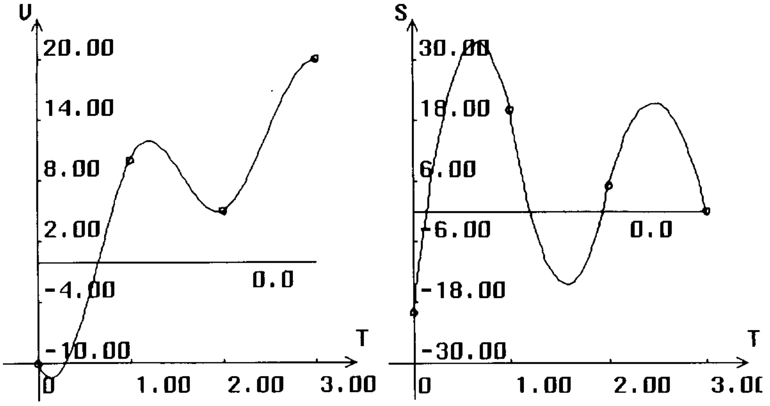
2.3. *The fifth dimension toolkit*

The high-level man-machine interactions are handled efficiently with the object oriented Fifth Dimension (5D) Toolkit[19]. We use the 2D class subset of this toolkit to design and manage the multiple window interface of the TRACK system. The 5D Toolkit also integrates recent three-dimensional input devices, such as the Spaceball and Polhemus, which are very important for human animation. Another functionality provided by the 5D Toolkit is Inter Process Communication (IPC). It is very useful for coordinating specialized software, running on different machines. For example, a motion control process can send joint level information to a physically-based surface deformation process. In the TRACK system, the 5D Toolkit plays an important role that makes the tool easy to develop as well as to use.

3. TRACK EDITING TECHNIQUES

3.1. *Key value editing*

Let us denote by V the value information of a key value, S the slope value and T the time (expressed in seconds). As an example of track, Fig. 4a presents the key values (small circles) and interpolated curve for a joint angle (in degree) while Fig. 4b shows the corresponding key slope and interpolated slope curve (in degree/sec). The user can easily modify the motion by editing the key time, key value, and key slope value as discussed next. The interpolation is based on cubic Hermite splines that insure that the curve passes through the key points with a specified key slope. For each pair of key points, this results in a system of four equations whose solution uniquely defines the cubic polynomial meeting these constraints. The animator



(a) (b)

Fig. 4. (a) A track value curve; (b) the corresponding track slope curve.

can select a current track either by picking the associated node_3D in the 2D hierarchy graph (Fig. 2a), by using a browser to select its name or by picking the joint directly on the 3D hierarchy display (Fig. 2b). After the current track selection, the insertion, removal, or modification of *V* or *S* is made with the mouse and the keyboard. The resulting track can be interpolated, interactively or with real time synchronization, with visualization of the 3D hierarchy representing motion.

3.2. *Key time editing*

There are three modes for key time editing: single, shift, squeeze and stretch. In the single mode, only the key time of the current key point is changed. This local change is constrained to stay between the previous and the next key time referred by the track. In the shift mode, the key time of the current key point is changed, as are all of its previous or next key times according to the shift side parameter. In the "squeeze and stretch" mode, in addition to the current key time T there are two limit key times TL and TR with $TL < T < TR$. If T is decreased, all the key times between TL and T are squeezed and all the key times between T and TR are stretched. This operation is enforced by maintaining the appropriate initial ratio:

for $Ti < T$: $(T - Ti)/(T - TL)$,

or for $Ti > T$: $(Ti - T)/(TR - T)$.

A local key slope value for a key value i is modified in the inverse proportion of change:

$$(T_{\text{before}} - Ti_{\text{before}})/(T_{\text{after}} - Ti_{\text{after}}).$$

For all slope modes, a squeeze increases the motion variations and a stretch decreases them.

3.3. *Compression filter*

A 100 hz sampling frequency is sufficient to retain the information dynamics of most of the motions performed by a human being[1], and, for a large range of every day motions the video sampling rate is still enough (depending on standards: around 25 Hz). As a direct consequence, the number of sampled key values for each recorded track is often penalizing for further manipulations of the sequence. For this reason, we provide a compression filter to decrease as much as possible the number of key values using the Hermite spline interpolation (see Fig. 5).

The basic principle is to discard any linear subset from the range of approximations and then apply a recursive Hermite spline approximation over the remaining nonlinear subsets. The underlying idea is that a linear subset conveys a higher level information, which structures the motion. The recursive Hermite spline approximation evaluates an error criteria over the nonlinear subset. The error criteria measures the area between the sampled curve and the cubic approximation with a Simpson integral and finally nor-

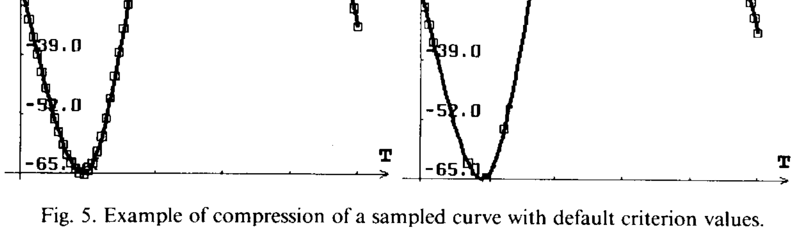


Fig. 5. Example of compression of a sampled curve with default criterion values.

malizes it by the subset duration. The nonlinear interval is validated if the error value is smaller than a defined constant. Otherwise it is divided into two parts and the algorithm is recursively applied[12].

3.4. *The coach-trainee correction method*

The Coach-Trainee correction method is based on a combination of direct and inverse kinematic control[11, 12] over an open chain (no closed loop). We just recall here the basic definitions of inverse kinematic control and the principle of our motion correction approach.

Inverse kinematic control is based on a linearization of the geometric model of a chain at the current state of the system. As a consequence, its validity is limited to the neighborhood of the current state and, as such, any desired motion has to comply with the hypothesis of small movements. The discrete form of the general solution provided by inverse kinematics is:

$$\Delta\Theta = J^+\Delta x + (I - J^+J)\Delta z \qquad (1)$$

where $\Delta\Theta$ is the unknown vector in the joint variation space, of dimension n. Δx describes the *main task* as a variation of the end effector position and orientation in Cartesian space. For example, in Figs. 6a and 6c the main task assigned to the end of the chain is to follow a line in the plane under the small movements hypothesis. The dimension of the main task, noted m, is usually less than or equal to n, the dimension of the joint space, to be of interest for inverse kinematic control.

J is the Jacobian matrix of the linear transformation, representing the differential behavior of the controlled system over the dimensions specified by the *main task*. J^+ is the unique pseudo-inverse of J providing the minimum norm solution, which realizes the *main task* (Fig. 10a). I is the identity matrix of the joint variation space ($n \times n$) $(I - J^+J)$ is a projection operator on the *null space* of the linear transformation J. Any element belonging to this joint variation subspace is mapped by J into the null vector in the Cartesian variation space. Δz describes a *secondary task* in the joint variation space. This task is partially realized via the projection on the *null space*. In other words, the second part of the equation does not modify the achievement of the main task for any value of Δz. Figure 10b has a null vector as its main task, so the displayed motion is always realized in the *null space*.

Usually Δz is calculated so as to minimize a cost function. If the main task belongs to the image space of J then the null space is $(n - m)$ dimensional in the joint variation space. This information is fundamental to evaluate the potentiality of the secondary task and clearly leads to a tradeoff between realization of main and secondary tasks.

The following concepts are at the base of the Coach-Trainee correction method (Fig. 7):

The basic idea is to consider the joint motion delivered by a motion generator as a reference model whose tracking is enforced through the secondary task while the main task insures the realization of desired Cartesian constraints for some specified end effector(s).

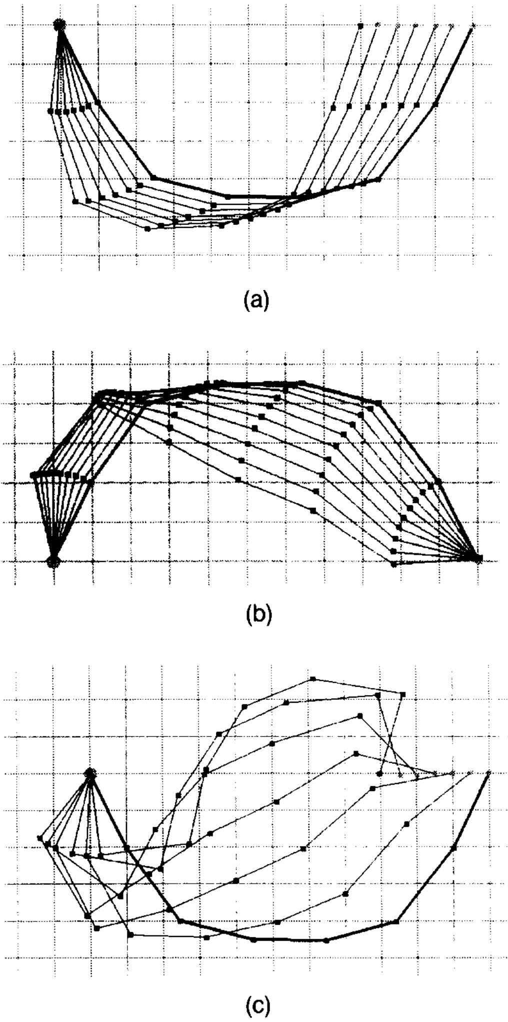


Fig. 6. (a) *Main task* only (minimum norm solution); (b) the null space is illustrated through a null vector *main task*; (c) same *main tasks* as Fig. 6a with arbitrary secondary task.

The user is more interested in local reshaping of the motion both in time and space. For this reason we prefer to use half-space Cartesian constraints (planar, cylindrical, spherical).

The controlled chain structure is duplicated to enforce the coach-trainee control metaphor. The original chain is simply controlled by the reference motion (the coach) and serves as a guide for the control of the duplicate one (the trainee). This *Coach-Trainee* metaphor is suggested by sport training as the adaptation of the coach reference movement into a trainee movement, being the closest possible with respect to a different context of body structure and/or Cartesian constraints.

Even with the Coach-Trainee control, the solution presents a first order discontinuity at the interface of the half-space constraint. For this reason we confer a thickness on the boundary of the half-space constraints and we operate smooth switching on the resulting *transition zone* by means of a *transition function* f (see[11] for details). Then, the formulation of the combined "direct and inverse" kinematic control method is given by the following equation (one dimensional constraint case):

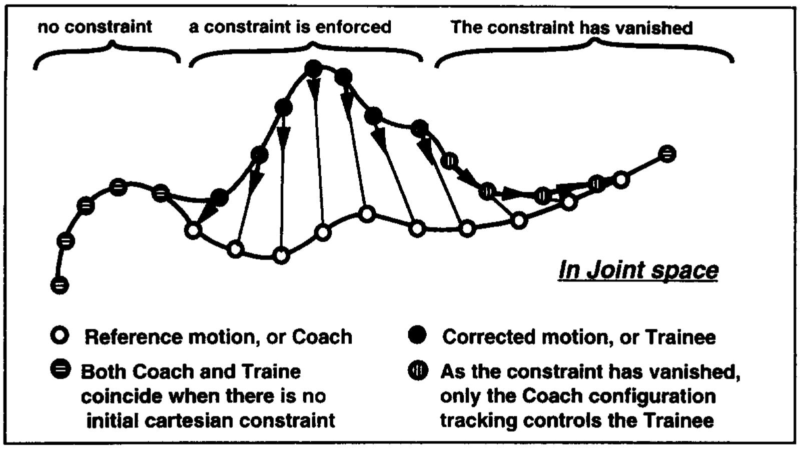


Fig. 7. The coach-trainee metaphor.

$$\Delta\Theta = J^{+}(f\Delta x + (1 - f)J\Delta z) + (I - J^{+}J)\Delta z \quad \text{with} \quad 0 \le f \le 1 \quad (2)$$

Regarding the multiple track sequence editing, we can use this motion generator to produce and record the sampled Coach motion and a sampled corrected motion of the Trainee. This motion can be compressed and further manipulated with the multiple track tools (Fig. 1). In fact, the corrected motion may not look natural due to the fact that our correction method does not take into account the dynamics characteristics of the articulated figure. The user should consider the method as a way to provide a pertinent distributed correction over a chain while trying to retain the dynamics of a reference motion. This multiple track corrected motion can be adjusted with our set of tools and checked again in Cartesian space with the Coach-Trainee method. The convergence to the desired motion is rather quick. An example is developed in Section 4 with a walking sequence.

3.5. *Multiple sequence manipulation*

This set of tools permits motion composition; we have a concatenation operator and a blending operator is currently being developed.

3.5.1. *Concatenation operator.* This operator concatenates a sequence A followed by a sequence B and puts the result into a sequence C. The concatenation mode and the delay value parameters provide a great flexibility. The sequences A, B, and C can be the same so, combined with the mode and the delay, we provide a very powerful operator. For example it is straightforward to construct a periodical motion with 2^n periods from n successive concatenation operations. The only intrinsic limitation of this operator is that a track from A cannot have its time-base overlapping the corresponding track from B. Moreover, the concatenation operator solves the continuity compromise that can occur if A and B coincide on the boundary time.

3.5.2. *Blending operator.* Blending is necessary whenever two sequences overlap in time and we need to keep their respective time-base and migrate continuously from one to the other. We can do this by defining a normalized cubic step function that varies from 0 to 1 on the overlapping zone. This function is used to weight the contribution of the sampled sequences on this zone for the evaluation of the blended key values. The resulting sequence is compressed afterwards to return to a standard Hermite spline form.

4. APPLICATION TO ARTICULATED FIGURE ANIMATION

In this section, a series of examples has been conducted with TRACK to produce some animation sequences.

4.1. *Walking generator*

In TRACK, human walking is produced by a model built from experimental data based on a wide range of normalized velocities. The user can change the current velocity and personification parameters [8] in real time.

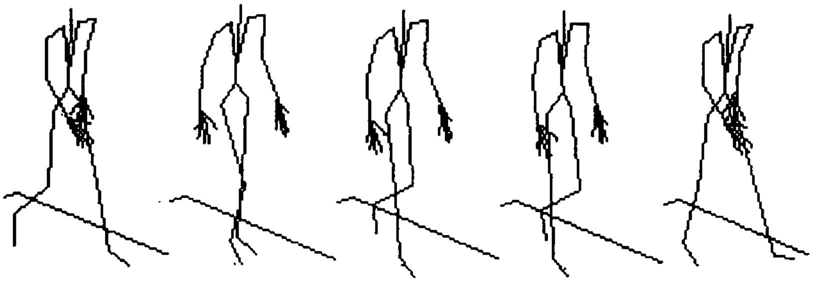


Fig. 8. Some walking frames.

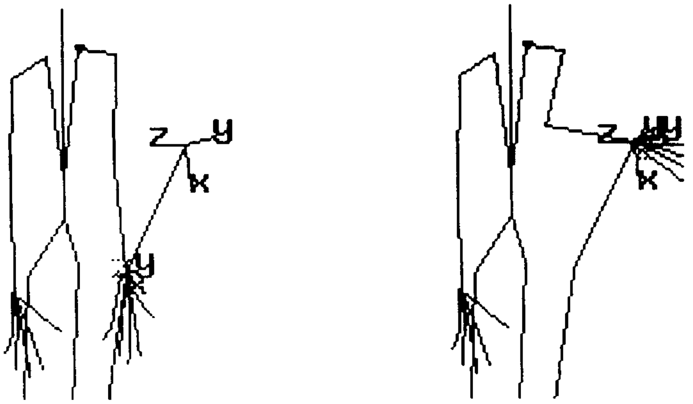


Fig. 9. Reaching a target (frame on the right) with the arm.

It is also possible to map the walking motion (Fig. 8) on a curved path[9].

4.2. *Posture design by inverse kinematics*

Motion is produced by inverse kinematics as described by Eq. (1). First, the animator picks the root and end joint, *e.g.*, shoulder and wrist. Then, the Spaceball or the Trackball can be used to define the target position and/or orientation interactively. The Spaceball is a 6 DOF interactive input device. This is essentially a "force" sensitive device that relates the forces and torques applied to the ball mounted on the top of the device. These forces and torques are sent to the computer in real time where they are interpreted and may be composed into homogeneous transformation matrices that can be applied to objects. The Trackball is a mouse simulation of the Spaceball. In Fig. 9, for example, one FREE node_3D with 6 DOF is used to represent the target that is reached incrementally by minimum norm variation of the chain posture.

4.3. *Coach-trainee correction and key frame refinement: A walking case study*

Figures 10, 11, and 12 show the Coach-Trainee application to a sampled walking motion. We can compare in Fig. 10, the reference motion, or Coach, and the corrected motion, or Trainee for the four flexion

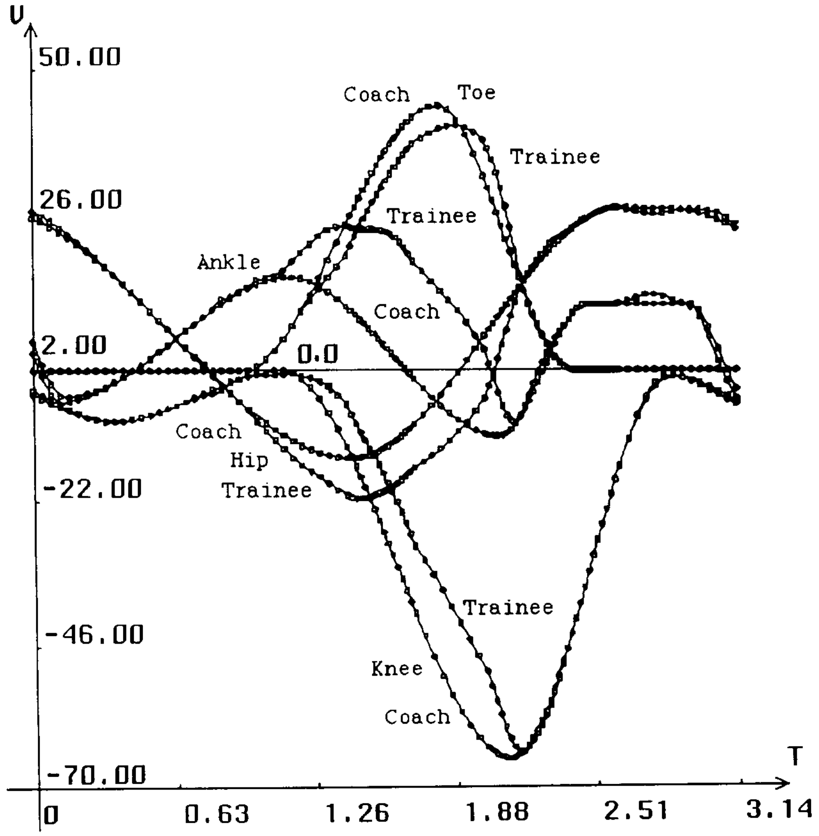


Fig. 10. Sampled coach and trainee motions for a walking leg.

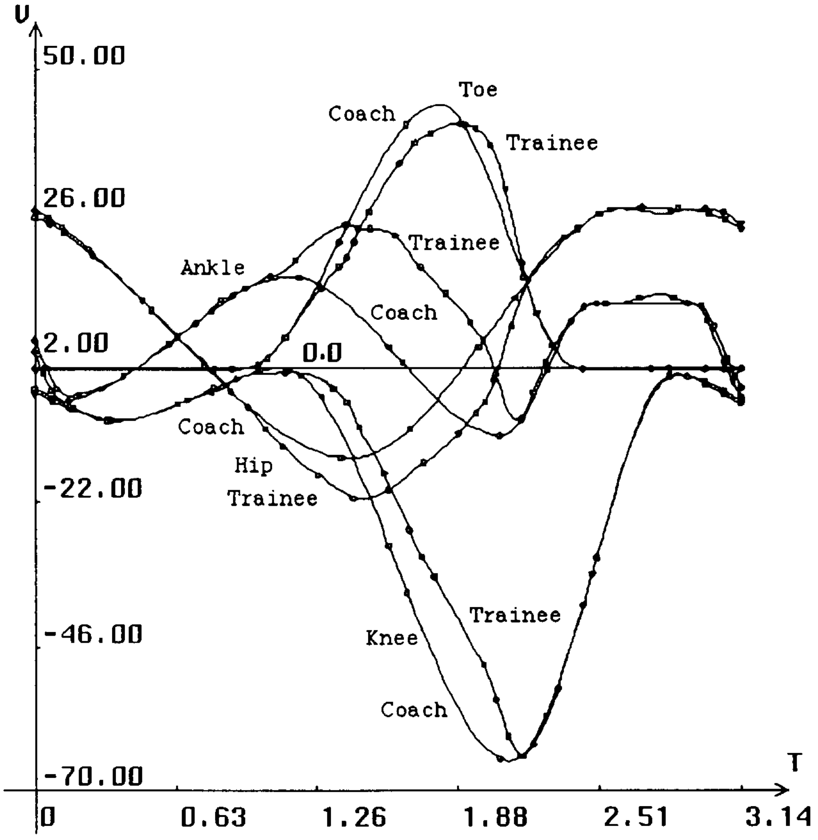


Fig. 11. Compressed coach and trainee motions for a walking leg.

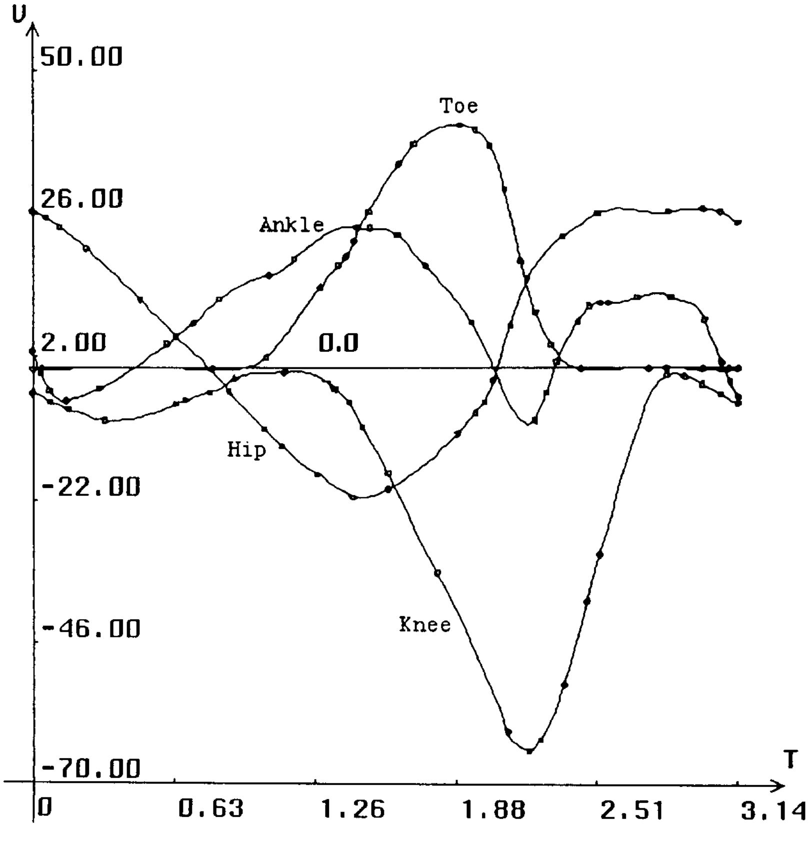


Fig. 12. Final edited motion.

angles of the leg (Hip, Knee, Ankle, Toe). Each little dot represents a sample. The purpose of the correction was to enforce an inequality floor constraint for two end effectors attached to the heel and the toe. We can see on Fig. 10 and 11 that, in the joint space, the correction deviation can be locally important but it always tracks back to the Coach motion when the Cartesian half-space constraint has vanished. Figure 11 shows the compressed motion with default criteria values. A reasonable number of remaining key points allows the designer to edit the corrected motion so as to smooth it in some regions (Fig. 12). The most visible editing work is made for the knee and the ankle around 2.2 seconds. As the resulting edited motion provides a good visual result it is kept as final motion (Fig. 13). Otherwise, a second pass into the Coach-Trainee correction method is still possible to converge to a satisfying motion with the same set of compression-editing operations. Figure 13 compares the initial and corrected posture of the foot in the Cartesian space.

5. IMPLEMENTATION

The TRACK system has been implemented in C language and Silicon Graphics GL graphic library. It integrates modules such as key frame, functional walking, direct and inverse kinematics and Coach-Trainee correction method. A library of interactive tools, based on the 5D Toolkit, has been developed for fine manipulation and display of multiple tracks sequences.

6. CONCLUSION AND FUTURE WORK

The TRACK system provides a powerful framework to combine and manipulate motion issued from different generators. Owned to the multiple track sequence and its wide set of tools, a motion designer can compose a complex motion both at the key frame level and at the goal-oriented level. In Fig. 14, illustrations are extracted from an animation produced with the TRACK system for the introducing sequence of Computer Animation 1993. The resulting sequence has been further processed with the body deformation software

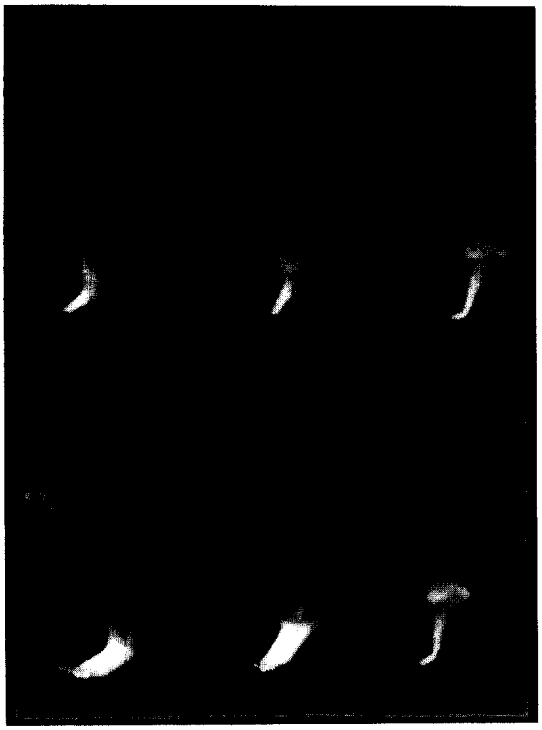
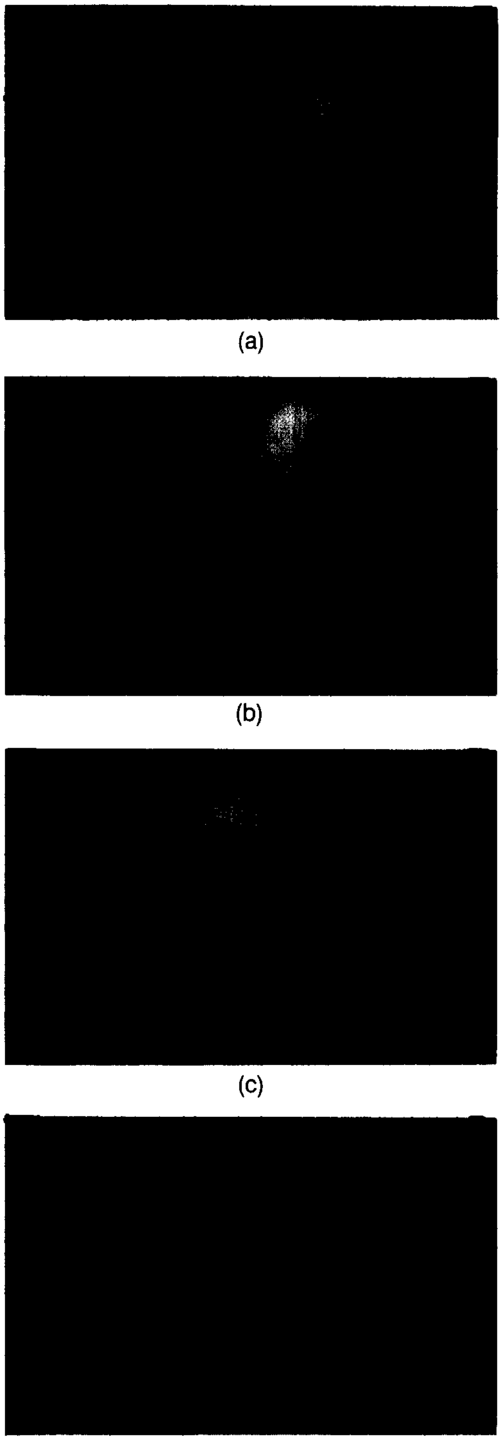


Fig. 13. Coach (top) and trainee (bottom) Cartesian foot motions for a floor half-space constraint.



(a)

(b)

(c)

(d)

Fig. 14a-d. Final images from an animation produced with the TRACK system.

and the fashion animation software[23] prior to generating the final images.

The future work will focus on dynamic control to produce physically-based motions automatically. Another direction is to integrate the TRACK system with the body deformation system into one tool. It is more efficient to design the sequence when the motion and the resulting deformation of the human body are simultaneously visible. A related issue is the collision detection and response with physical realism that will also be investigated in the future.

Acknowledgements—The authors are grateful to Hans Martin Werner for his help in revising this text. We also wish to thank Laurent Bezault, Ying Yang, Hans Martin Werner, and Natacha de Sousa for the design of the introducing animation of Computer Animation 93. The research was partly supported by the Swiss National Research Foundation and OFES and is part of the ESPRIT Project "HUMANOID I."

REFERENCES

1. R. Maiocchi, B. Pernici, Directing an animated scene with autonomous actors. *Computer Animation 90,* Springer Verlag, Tokyo Geneva (1990).
2. D. Fortin, J. F. Lamy, and D. Thalmann, A multiple track animator system for motion synchronization and perception. In *Motion Representation and Perception,* N. I. Badler and J. K. Tsotsos, (Eds.), ACM (1986).
3. J. E. Gomez, Twixt: A 3D animation system. *Eurographics '84.*
4. S. N. Steketee, and N. I. Badler, Parametric keyframe interpolation incorporating kinetic adjustment and phrasing control. *SIGGRAPH '85,* 255-262.
5. T. W. Calvert, C. Welman, S. Gaudet, T. Schiphorst, and C. Lee, Composition of multiple figure sequences for dance and animation. *Visual Comp.* 7(2/3), (1991).
6. M. Girard and A. A. Maciejewski, Computational modeling for computer generation of legged figures, SIGGRAPH '85. *Comp. Graph.* **19**(3), 263-270 (1985).
7. M. Girard, Interactive design of 3D computer-animated legged animal motion. *IEEE Comp. Graph. Appl.* 7(6), 39-51 (1987).
8. R. Boulic, N. M. Thalmann, and D. Thalmann, A global human walking model with real time kinematic personification. *Visual Comp.* **6**(6), 344-358 (1991).
9. L. Bezault, R. Boulic, N. M. Thalmann, and D. Thalmann, An interactive tool for the design of human free-walking trajectories. *Computer Animation '92,* Geneva (May 1992).
10. C. Phillips, J. Zhao, and N. I. Badler, Interactive real-time articulated figure manipulation using multiple kinematic constraints, SIGGRAPH '90. *Comp. Graph.* **24**(2) (1990).
11. R. Boulic and D. Thalmann, Combined direct and inverse kinematic control for articulated figures motion editing. *Comp. Graph. Forum* **2**(4), 189-202 (1992).
12. R. Boulic, and D. Thalmann, TRACK: A kinematic goal-oriented animation system for coordinated editing of joint-space based motions. *Eurographics '92, Third Workshop on Animation and Simulation.* Cambridge, England (5-6 September 1992).
13. P. M. Isaacs, and M. F. Cohen, Mixed methods for kinematic constraints in dynamic figure animation. *Visual Comp.* **4,** (1989).
14. W. Armstrong, and M. Green. The dynamics of articulated rigid bodies for purposes of animation. *Proc. Graph. Interface,* 407-415 (May, 1986).
15. J. Wilhelms, and B. Barsky, Using dynamic analysis to animate articulated bodies such as humans and robots. *Proc. Graph. Interface,* 97-104 (1985).
16. A. Witkin, and M. Kass, Spacetime constraints, SIGGRAPH '88. *Comp. Graph.* **20**(2), 159-168 (1988).
17. M. F. Cohen, Interactive spacetime control for animation, SIGGRAPH 92. *Comp. Graph.* **26**(2), 293-302 (1992).
18. R. Boulic, and O. Renault, 3D hierarchies for animation. *New Trends in Animation and Visualization,* Wiley Professional Computing, New York (1991).
19. R. Turner, E. Gobbetti, F. Balaguer, A. Mangili, D. Thalmann, and N. M. Thalmann, An object-oriented methodology using dynamic variables for animation and scientific visualization. In *CGI90,* Springer-Verlag, Berlin, 317-327.
20. A. LeBlanc, P. Kalra, M. N. Thalmann, and D. Thalmann, Sculpting with the ball & mouse metaphor. *Proc. Graph. Interface '91,* 152-159. Calgary, Canada (1991).
21. P. S. Strauss and R. Carey, An object-oriented 3D graphics toolkit, SIGGRAPH 92. *Comp. Graph.* **26**(2), 341-349 (1992).
22. D. Zeltzer, S. Pieper, and D. J. Sturman, An integrated graphical simulation platform, *Proc. Graph. Interface '89,* London, Ontario (June 1989).
23. M. Carignan, Y. Yang, M. N. Thalmann, D. Thalmann, Dressing Animated Actors with Complex Deformable Clothes, SIGGRAPH 92. *Comp. Graph.* **26**(2), 99-104 (1992).