# Experimenting Prioritized IK for Motion Editing

Category: scientific research

Ronan Boulic[1], Benoît Le Callennec[1], Martin Herren[2], Herbert Bay[2]
[1] VRlab, [2] SMT, EPFL Lausanne CH-1015 Switzerland

**Abstract**

The presented work illustrates the potential of an IK algorithm enforcing priorities among constraints for motion editing. Motion capture is a very efficient technique to deliver believable motions but usually most of the captured motions need to be edited before they match the end-user's needs. Motion editing can be achieved by applying constraints to parts of the animated character while trying to retain most of the original motion. However all constraints are not equal: some have more importance than others for the animator. The sequences presented here were obtained with an IK solver that allows the user to associate a priority level to a constraint (without limitation in the number of priority levels). We first provide a short overview of the edited sequence prior to describe the general framework of the test application and the potential of the IK algorithm with priorities. We focus especially on the first case where three priority levels are exploited to provide a fine control over the arm animation. The two next cases focus on the specification of goals attached to mobile frames. A general discussion reviews the advantages and limitations of the recent state of the IK algorithm.

Categories and subject Descriptors : I.3.7[Three Dimensional Graphics and Realism]: Animation

## 1. Overview of the Motion Editing Framework

### 1.1 Summary of the results

We have first chosen a dance motion to experiment with the priorities as it is important to retain its original expressiveness while enforcing the constraints. Figure 1 shows a frame where both the elbow and the wrist are constrained to reach some positions in space. Changing their relative priority leads to different sequences (here the elbow is given a higher priority).

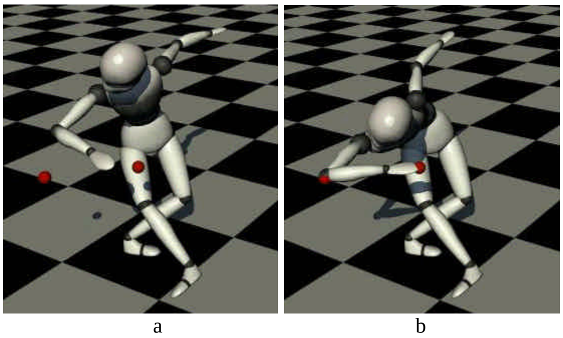

**Figure 1:** *Editing of a dance movement, postures for the original (a) and the edited (b) movements*

The next two edited movements highlight the specification of goals attached to mobile frames, like the head frame (Figure 2) or the thorax frame (Figure 3).

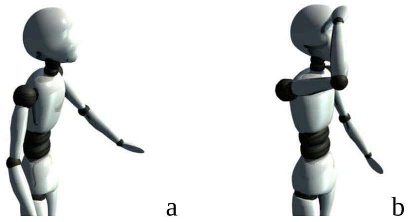

**Figure 2:** *constraint goal relative to the head*

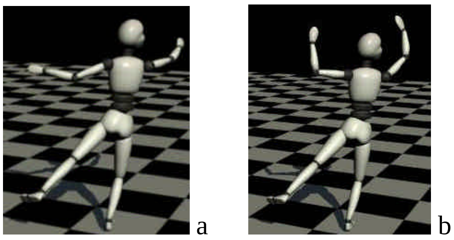

**Figure 3:** *constraint goal relative to the thorax*

### 1.2 The motion production pipeline

All the sequences have been first processed in our Motion deformation application DefMove where the user can set the precise timing and priority level of each constraints (Figure 4). The files describing the H-anim compliant animate character, the constraint set and the keyframed motions have been then imported in a plug-in for Maya 4.5 where the final rendering was made.

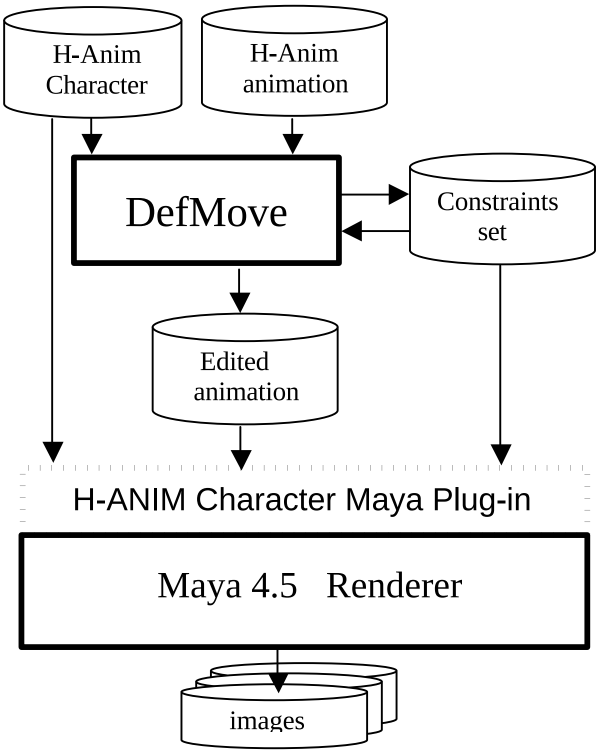

**Figure 4:** *sequence production pipeline*

In the longer term, the Maya plug-in will integrate the whole motion deformation functionality.

### 1.3 The choice of H-Anim compliant characters

The standardization of the skeleton is the initial impulse that launched the H-Anim effort [1]. It was motivated by the need of raising animation design productivity through the re-use of H-Anim compliant animations on any H-Anim compliant skeletons. In short, H-Anim goal is to provide *interoperable virtual humans* for which animations are *modelled once, run anywhere.* In the VRlab and the MIRAlab from Geneva University, we have adopted the H-Anim standard proposal for specifying the skeleton and the animation of our human-like characters. Through our current experience we see two main advantages to the use of this standard:

- A complete set of standardized joint names together with a standardized topology is shared by the animators and researchers. Any subset respecting the topology is declared valid thus allowing the definition of a wide range of characters from crudely defined to fully anatomic skeletons.
- A unique convention for joint frame orientation provides a common ground for defining animation files thus allowing to run them on any H-Anim compliant characters.

### 1.4 Our approach to motion editing

The motion of a 3D character is usually represented as a set of joint trajectories together with the root trajectory in a world reference frame. Such a representation captures very effectively the expressiveness of the motion; the joint states can be very easily mapped to characters complying with the same standard. However, the Cartesian trajectories of key points on the character body capture the interactions with the external world [2]; for example the feet trajectories may indicate a path to follow, the hand trajectories may indicate an object to grasp, other point trajectories can be important to express obstacle location, etc… We want to stress that both representation capture valuable information for the end-user. In the present approach, we exploit both motion representations to allow the end user to indicate *what* is important to preserve. Our approach to motion editing consists in:

- retaining the natural dynamics of the original movement by preserving:
    - the joint angle trajectories
    - the Cartesian trajectories of optional user-chosen points on the character.
- adding user-defined Cartesian constraints *wherever* and *whenever* needed.

The association of a *priority level* to a constraint is the key element for ensuring a high flexibility to the motion editing process. This concept of *priority* should not be confused with the concept of *importance* dynamically evaluated in [3] in an on-line context. A high priority level strictly ensures the achievement of a constraint with respect to lower priority ones. On the contrary, an importance level is equivalent to a weight, thus leading to a compromise solution similar to the approach from Badler et al. [4,5].

By default, the joint angle preservation is always requested and is assigned the lowest priority. This is a common aspect with the approach from Monzani et al. [6]; the novelty of the present architecture comes from the exploitation of an arbitrary number of higher priority level constraints together with a smoothed goal management compatible with a one-pass processing.

Our approach differs also greatly regarding the recruiting of the joints for achieving the constraints. Very often the articulated structure is divided into independent sub-structures that offer closed form solutions for user-defined constraints [12]. The drawback of this approach is the lack of global synergy in solving a set of conflicting constraints: some solutions might exist but the partitioning prevents their emergence. On the contrary, our approach allows a constraint to recruit all or part of the joints from its parent joint up to the root joint. This stage of the constraint definition is called the *joint recruiting*; by default all the joints potentially influencing a constrained effector are recruited (more details in [13]). As a consequence, multiple constraints may compete for the control of some common joints. Very often the redundancy of the joint space allows to find solutions for all constraints. In case some constraints are conflicting, their distinct priority level sorts them in terms of constraint achievability. In this way we can ensure the total achievement of the higher priority ones and the partial achievement of the lower priority ones.

### 1.5 Comparison with other approaches

Motion editing has always been a key topic of computer animation but the recent explosion of motion capture systems has stressed new user needs. Indeed, mocap is today the preferred approach to produce convincing human motions, especially those active motions involving interactions with the environment [7]. However, high production costs, low flexibility, and artefacts introduced by the capture process approximations have stimulated the proposal of numerous motion editing techniques. Some of them allow adjustments expressed in the posture space [8] or in the Cartesian space [9,10]. An extended discussion about the relative interest of preserving joint angles vs retaining Cartesian space constraints can be found in [3]. The continuity of the resulting movement being a key evaluation criteria, most of them work off-line as multiple pass editing tools. A minority of approaches target real-time retargeting for broadcast [3] or on-line applications [11]. In this latter context the continuity requirement is more difficult to enforce as only the past of the movement is known as opposed to the off-line context where all the movement information can be exploited. On the other hand on-line methods offer a great potential for the adaptive animation of autonomous characters moving in complex evolving contexts (e.g. in on-line games).

The motion editing approach presented here belongs to the per-frame family of methods as we want to exploit it for on-line adaptation of movement in the future. Presently the additional computing cost required for enforcing the prioritized constraint prevents its real-time use. It is exploited in a one-pass off-line context where the user predefines the timing and the relative priority of an arbitrary number of constraints.

## 2. Constraints definition

The motion deformation is obtained through a set of constraints $\{c_k\}$ for $k=1,N$. Each constraint $c_k$ consists at least in the definition of (Figure 5):

- a point $\underline{\mathbf{p}}_k$ attached to the character (the *Effector*)
- a set of *recruited joints* $\{ q_{kj} \}$
- a *priority level* **k** with $k=1$ being the highest level (think of it as a rank).
- A set of goals $\{\underline{\mathbf{g}}_{ki}\}$, $1 \le i \le N_g$ for the Effector

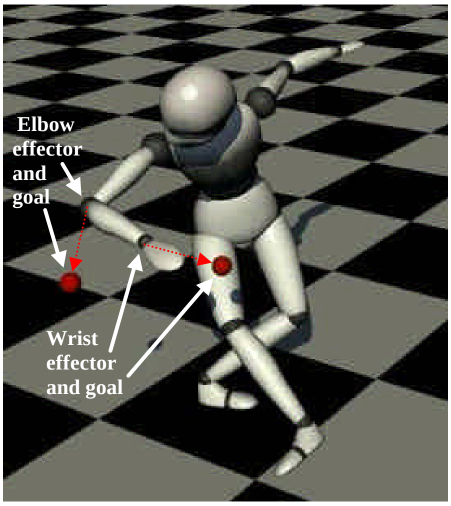

**Figure 5:** *Effectors and position goals for the elbow and wrist constraints (dotted arrows)*

Each goal has an associated *activity* timing $\{\mathbf{t_0}, \mathbf{t_1}, \mathbf{t_2}, \mathbf{t_3}\}$ freely set within the motion duration $[\mathbf{t_I}, \mathbf{t_F}]$. The goal activity is reflected in Figure 5 and the animations with colour interpolation (from *inactive* in green to *active* in red). The ease-in and ease-out phases are cubic step functions that smoothly transition the activity within its [0,1] range (Figure 6).

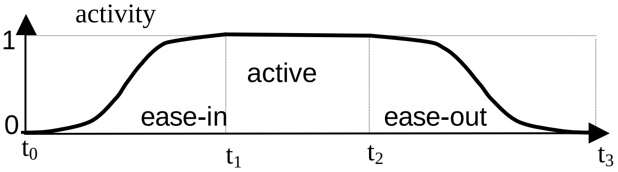

**Figure 6:** *Activity timing of one goal*

It is worth explaining how an effector is constrained depending on its number of goals and their current activity. Every effector has an *original trajectory* due to the motion to edit (Figure 7). If the user just wants to retain the original trajectory of a body point, an effector can be created at that location; its implicit goal is to reproduce its original trajectory for every frame. On the other hand, if the user wants to modify the effector location during part of the sequence, one or more explicit goals are created with an associated timing (Figure 7). Then, for each frame an instantaneous goal is computed from the knowledge of the effector's goals, of their activities and of the current position of the original trajectory [13]. In a typical motion editing session, the user first defines the effectors for which she wants to deform the motion. Then based on the resulting motion some effectors are added to retain visually important trajectories of body parts.

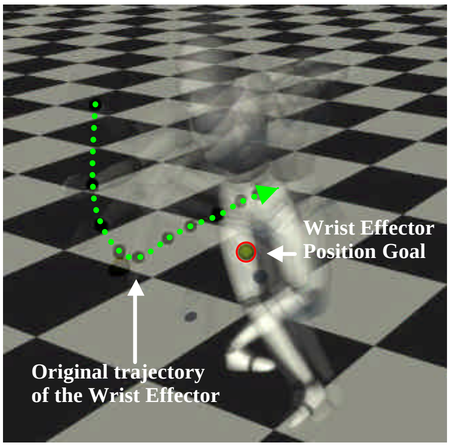

**Figure 7:** *Effector original trajectory*

Compared to other approaches our strategy is to let the animator define as many effector as she wants, even if they have conflicting constraints. The constraints can be simultaneous or partially overlapping; in all cases the associated priority determines which one must be realized as much as possible before trying to achieve those with a lower priority. Until now, motion deformation constraints were generally given higher priority levels compared to those retaining the original trajectory. In the future, it could be the reverse when we integrate the possibility to constrain the centre of mass [14] because the natural dynamics of the motion depends a lot on its original trajectory. Before en

## 3. Goal types and options

The user can specify two types of goals:

- **A position goal $G_i$** attracts the effector while the goal is active. Figure 8 top row illustrates such a case where the grey dot is the position goal and the bottom line is the original trajectory of the effector. The activity timing is the one from Figure 6 resulting in the construction of instantaneous goals (white dotted points). Combining multiple goals with overlapping timing can help minimizing the discontinuity appearing in the trajectory.
- **An offset vector $O_i$** serves to define a *goal trajectory* by offsetting the original trajectory over the activity timing. Figure 8 bottom row illustrates this context with the timing from Figure 6. this approach produces an overall smoother trajectory.

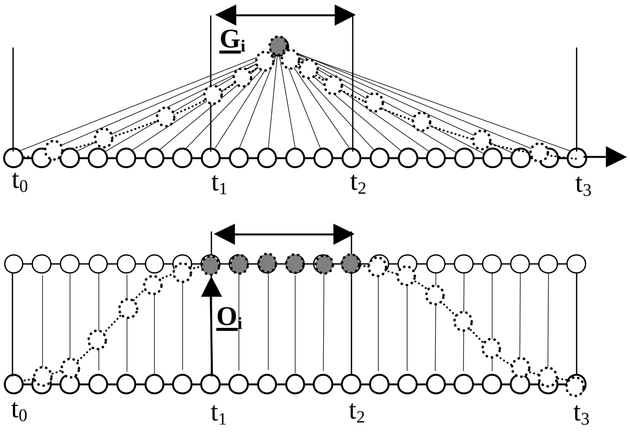

**Figure 8**: *Construction of the instantaneous goals (dotted points) for a Position goal (top row) and an Offset goal (bottom row). The original trajectory is the horizontal line with white dots.*

Two options are offered for both goal types:

- The goal can be expressed in a mobile frame.
- The constraint can be achieved along one (plane), two (line) or three (point) dimensions.

## 4. Managing priorities in the IK solver

The kernel of the prioritized IK is based on an algorithm described by Baerlocher et al. [15]. Like the classic numeric IK approach, we rely on the Jacobian matrix gathering the partial derivatives of the constraints variables with respect to the joint parameters. Building a position Jacobian like the ones we exploit is straightforward [16]. Solving for multiple constraints depending on a common set of joints require to consider them in the same linearized system. If we gather all the constraints Jacobian by piling them into a unique Jacobian matrix, we end up with a compromise solution [4,5]. We enforce here distinct priority levels by building dedicated projection operators in an efficient way [15]. At the lowest priority level we enforce a *joint angle preservation constraint* which build an error term proportional to the difference between the current joint value and the original motion value. All the examples shown here have such a low-level optimisation.

## 5. Results

A simplified H-Anim character is used for all the motion editing sessions. In the first example the constraints recruit the spine in addition to the arm as the goals are too distant. For the other two examples only the arms animation are modified.

### 5.1 Prioritized constraints

In this example, an original dance motion is temporarily constrained to obtain a different arm position (Figure 1). Two constraints are acting respectively on the elbow and the wrist (Figure 5); they have position type goals (Figures 7, 8). Three scenarios of relative priorities are shown: first both constraints have the same priority, then the wrist has the highest priority and finally the elbow has the highest priority. When no priority is set among constraints none of them is completely realized; this can be a problem for the animator when tuning a complex sequence. On the other hand, the cases with prioritized constraints are consistent with the animators specifications. Although both constraints exploit some common joints, a solution for each frame. There is however one remaining continuity problem for the head motion due to the spine modified animation. This artefact can be corrected by enforcing a frame to frame continuity constraint at the IK level. We are currently exploring which criteria is best suited for that purpose.

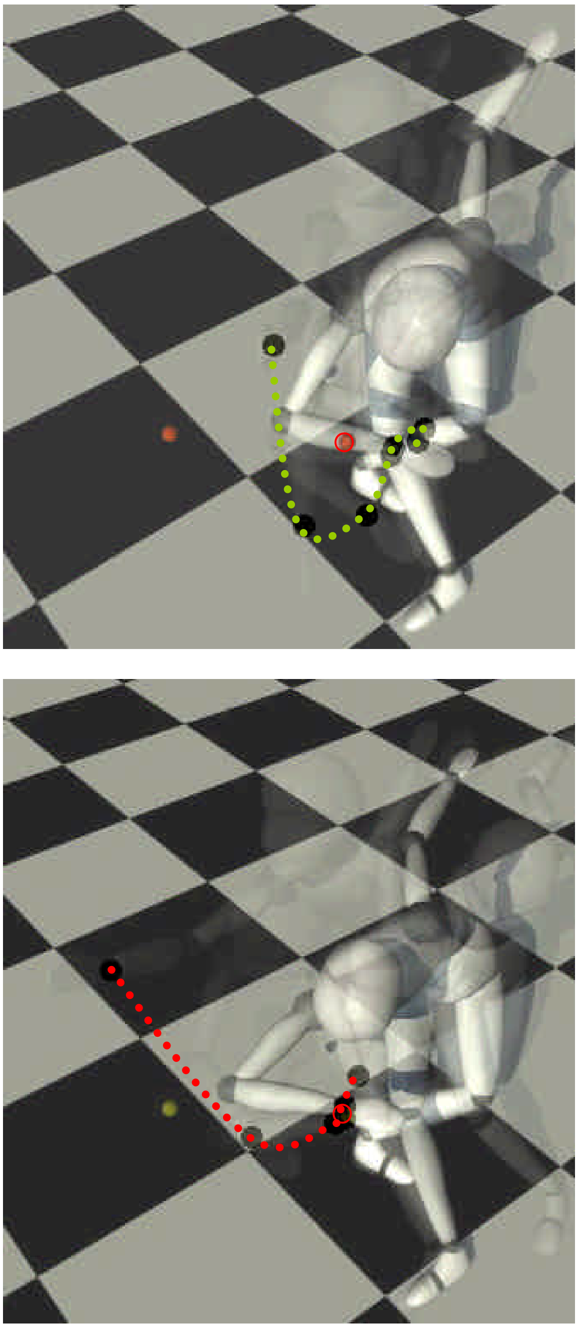

**Figure 7:** *Wrist original trajectory (top)and edited motion with priority to the wrist (bottom)*

### 5.2 Goal expressed in a mobile frame

In the second example a constraint is set on the hand so that it follows the head movement (Figure 2). Although it is not the case here, multiple point constraints can be easily combined to achieve also an orientation control.

The third example also has goals expressed in a body frame, the thorax for that example (Figure 3) so that the hands raise while the body is moving.

For both cases the angle preservation constraint is still realized at the lowest priority level (like in [6]).

## 6. Discussion

The prioritized IK definitely provides a finer means for tuning motion editing. Presently, two drawbacks limit its wider use:

- The lack of frame to frame continuity criteria may introduce some discontinuity for body part that are not explicitly controlled. We are exploring this issue.
- The computing cost is still too high (from 10ms to 300ms per frame) especially when the motion editing requires to activate the joint limits. The introduction of the frame to frame continuity constraint will certainly improve this aspect too as our convergence halting criteria tests the error variation rather than the absolute value of the error.

From a more general point of view, the possibility to express a goal in a local frame of the character body allows to handle a large family of body-centric constraints which stands in-between the joint angle preservation and the Cartesian constraints linked to the external world .

## Acknowledgements

This research has been supported by the FNRS under the grant N° 2000-061999.00. The final rendered animation sequences have been obtained with Maya through the Alias/Wavefront Research Donation programme. We also wish to thank Nicolas Elsig for the design of the humanoid character and to Mireille Clavien and Olivier Renault for assembling and compressing the animation files.

## References

1. H-ANIM, Humanoid Animation Working Group, www.hanim.org
2. BINDIGANAVALE R., BADLER N., Motion Abstraction and Mapping with Spatial Constraints, *Proc of Captech98*, LNAI 1537, pp 70-82, Springer-Verlag Berlin Heidelberg
3. SHIN H.J., LEE J., SHIN S.Y., GLEICHER M., Computer Puppetry: An Importance-Based Approach, *ACM Transaction on Graphics*, Vol 20(2), April 2001, pp 67-94
4 BADLER N.I., MANOOCHEHRI K.H., WALTERS G., Articulated Figure Positioning by Multiple Constraints, *IEEE Computer Graphics & Applications*, Vol. 7(6), pp. 28 - 38, June 1987
5. PHILIPS C.B., ZHAO J., BADLER N.I. Interactive Real-Time Articulated Figure Manipulation Using Multiple Kinematic Constraints, *Computer Graphics* 24 (2),pp 245-250, 1990
6. MONZANI J.-S., BAERLOCHER P., BOULIC R., THALMANN D., Using an Intermediate Skeleton and Inverse Kinematics for Motion Retargeting, *Proc. Eurographics 2000*, Interlaken August 2000
7. MENACHE A., Understanding Motion Capture for Computer Animation and Video Games, ISBN 0-12-490630-3, Morgan Kaufmann, 2000
8. WITKIN A., Popovic Z., Motion Warping, *Proc. of SIGGRAPH'95*, Los Angeles, 1995
9. GLEICHER M. and LITWINOWICZ P. Constraint-based Motion Adaptation, *The Journal of Vizualisation and Computer Animation*, 9, 65-94 (1998)
10. LEE J. and SHIN S. Y. A Hierarchical Approach to Interactive Motion Editing for Human-Like Figures, *Proc. of SIGGRAPH'99*, Los Angeles
11. CHOI K.-J., KO H.-S., On-line motion retargeting. *Journal of Visualization and Computer Animation* 11(5): 223-235 (2000)
12 TOLANI D., GOSWAMI A., BADLER N.I, Real-time inverse kinematics techniques for anthropomorphic arms", *Graphical Models*, Vol. 62, pp. 353 - 388, 2000
13. BOULIC R., LE CALLENNEC B., HERREN M., BAY H., Motion Editing with Prioritized Constraints. Submitted to RICHMEDIA'03.
14. BOULIC R., MAS R., THALMANN D. "A Robust Approach for the Center of Mass Position Control with Inverse Kinetics", Journal of Computers and Graphics, Vol. 20, No. 5, pp. 693 - 701, 1996
15. BAERLOCHER P., BOULIC R., Task-Priority Formulations for the Kinematic Control of Highly Redundant Articulated Structures, *Proc. of IEEE IROS 98*, Victoria, pp. 323-329, Oct. 1998
16. WATT A., WATT M., Advanced Animation and Rendering Techniques, Addison-Wesley, ACM Press, 1992.