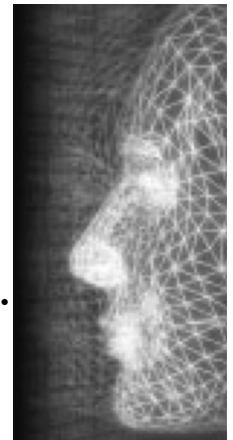**Motion Capture and Retrieval**

Animating reactive motion using momentum-based inverse kinematics

By Taku Komura*, Edmond S. L. Ho and Rynson W. H. Lau

Interactive generation of reactive motions for virtual humans as they are hit, pushed and pulled are very important to many applications, such as computer games. In this paper, we propose a new method to simulate reactive motions during arbitrary bipedal activities, such as standing, walking or running. It is based on momentum based inverse kinematics and motion blending. When generating the animation, the user first imports the primary motion to which the perturbation is to be applied to. According to the condition of the impact, the system selects a reactive motion from the database of pre-captured stepping and reactive motions. It then blends the selected motion into the primary motion using momentum-based inverse kinematics. Since the reactive motions can be edited in real-time, the criteria for motion search can be much relaxed than previous methods, and therefore, the computational cost for motion search can be reduced. Using our method, it is possible to generate reactive motions by applying external perturbations to the characters at arbitrary moment while they are performing some actions. Copyright © 2005 John Wiley & Sons, Ltd.

KEY WORDS: computer animation; inverse kinematics; real-time animation

Introduction

The synthesis of realistic human motion is a challenging research problem with broad applications in movies, cartoons, virtual environments, and games. Due to the quality and realism of the result, the use of motion captured data has become a popular and an effective means of animating human figures. However, since it is an inherently offline process, there has been great interest in developing algorithms that are suitable for interactive applications. However, designing appropriate control schemes can be difficult, and only a limited number of methods consider reactive motions due to the presence of applied external forces.[1–3]

In this paper, we propose a new method to generate reactive motions for arbitrary external perturbations such as pushing, pulling, or hitting. The method can be applied to motions with arbitrary contact states such as standing, walking, or running. It works as follows. First, a number of reactive motions of a person are captured by pushing and pulling him/her to various directions, and are stored in the motion database. During the application, some captured motion data is used as the primary motion of a human figure. When an external perturbation is applied to the body of the figure, it is transformed to an increase in the linear momentum and angular momentum around the center of mass (COM). At the same time, a counteracting motion will be searched in the database to be blended into the primary motion.

When searching for the counteracting motion, we mainly focus on the moment of inertia of the whole body and the status of the feet. The selection of the reactive motion in the database is done by a less strict (and hence more efficient) criterion, comparing to those used in previous methods[4,5] Therefore, the best matching

*Correspondence to: Taku Komura, Department of Computer Science, City University of Hong Kong, Tat Chee Avenue, Kowloon, Hong Kong SAR. E-mail: taku@ieee.org

motion can be searched in a shorter amount of time. Since it is usually not possible to pre-capture all potential reactive motions, to cover all directions and magnitude of impact forces, it is necessary to be able to edit the selected reactive motion so that it can be blended with the primary motion. To blend the postures and edit the motion, a momentum-based inverse kinematics (IK) method is used. The positions of the foot steps are adjusted to deal with impacts of various strengths. There are two advantages of using momentum-based IK to calculate the motion after the impact. First, it is possible to add linear constraints such as specifying the trajectory of certain segments during the motion. This functionality is essential when constraining the position or orientation of any segment in the Cartesian space. Second, it is possible to edit the motion by changing the profile of the momentum to generate effects such as pushing or pulling the body with different strengths and from different directions. This is an important factor when creating reactive motions, as the condition of the impact is not known in advance. As a result, wider ranges of reactive motions can be covered by a limited number of pre-captured motions.

The contributions of this paper can be briefly summarized as follows. First, we propose a framework to generate reactive motions in real-time, based on the use of the momentum-based IK and an efficient motion-search method. Second, we propose a method for editing the momentum and the footsteps of a body by changing the condition of the impact; this method is proposed through observation and analysis of actual human motion.

The rest of this paper is organized as follows. Section Related work gives an overview of related research. Section Capturing Motions explains how to generate the motion library of the reactive motions. Section Generating Reactive Motions presents our method for generating reactive motion. Section Experimental Results presents some experimental results of the new method. Finally, Section Summary and Discussion concludes the paper with some discussions.

Related Works

Much research has been conducted with the goal of creating, editing, connecting, and retargeting human motions. Due to the progress in motion capturing technology and the improved access to motion captured data, the focus of studies has been shifted to stochastic methods that utilize the motion data stored in the database.

Lee et al.[4] Kovar et al.,[5] and Arikan et al.[6] propose interactive character control through searching and connecting motions in the database. They treat human motions as nodes in a graph and search for the motion available in the database that can be used to realize the desired motion. They suggest criteria to decide if a motion can be blended to another. For example, Lee et al.[4] uses the following cost function:

$$D_{i,j} = d(\boldsymbol{q}_i, \boldsymbol{q}_j) + v d(\dot{\boldsymbol{q}}_j, \dot{\boldsymbol{q}}_j) \qquad (1)$$

where $d(\boldsymbol{q}_i, \boldsymbol{q}_j)$ and $d(\dot{\boldsymbol{q}}_i, \dot{\boldsymbol{q}}_j)$ are the difference and their derivative of the generalized coordinates between frames i and j, and v is a weighting factor. $d(\boldsymbol{q}_i, \boldsymbol{q}_j)$ is defined a

$$d(\boldsymbol{q}_i, \boldsymbol{q}_j) = ||\boldsymbol{q}_{i,0} - \boldsymbol{q}_{j,0}||^2 + \sum_{k=1}^{m} w_k ||log(\boldsymbol{q}_{j,k}^{-1} \boldsymbol{q}_{i,k})||^2$$

where $\boldsymbol{q}_{i,0}, \boldsymbol{q}_{j,0}$ are the root positions in frames i and j. $\boldsymbol{q}_{i,k}$ and $\boldsymbol{q}_{j,k}$ are the orientations of joint k in frames i and j in the quaternion form.

The stochastic approach can be used to generate reactive body motion when external perturbations are applied to the body, which is the target of this study. However, directly using this method is not practical due to three reasons. First, since some external force can endanger the subject, it may not be possible to capture all reactive motions for arbitrary postures. Second, as the reactive motions differ greatly according to the timing and the part of the body to which the external perturbation is applied to, an excessive amount of data will be needed if we are to capture various reactive motions under all possible conditions. Third, even though if we may capture all the motion data, the cost of generating the motion graph that takes into account all sorts of external perturbation will be too high, due to the need to handle a large number of nodes and to compute a large number of expensive matching operations as shown in Equation (1).

Because human balance relies a lot on physical factors, physically-based techniques have been used to generate reactive motions. Ragdol physics has often been used in video games to generate the reactive motions of human figures. It has been used combinedly with motion graph to allow users to interact with the virtual character.[7] Even though such methodologies are effective to generate the motions of characters simply

214

falling down onto the ground, it is quite difficult to generate effects such as stepping out and stopping, which tend to happen more occasionally when simulating reactive motions. Zordan and Hodgins[1] propose a method to generate reactive motion using PD-control in a forward dynamics environment, for situations such as a virtual character being punched. Oshita and Makinouchi[2] apply IK to create reactive motions such as luggage suddenly attached to the back of the body. Although these methods consider the balance of the body, they do not consider motions with changing contact state such as walking or running, in which the supporting pattern changes periodically.

Komura et al.[3] generate reactive motions during biped locomotion such as walking and running. A balance controller based on robotics theory is enhanced to counteract large perturbation applied to the body. However, the stepping pattern of the feet is kept the same and no reactive motion is generated to change the original stepping pattern to counteract the impact. For example, if a human is pushed from the back while standing, it is expected to step out to stop the body. If it is pushed strongly from the back while walking, it will jog for a few steps to counteract the motion and then return to the walking motion. This kind of effect is desirable.

To create realistic reactive motions, in this research, we explore an idea to combine stochastic methods with physically-based techniques. Our method is similar to Zordan et al.[8] in that the physically generated motion is blended with the reactive motion from the database. However, they alpha blend the motion generated by PD-control with the motion from the database. The main difference between the two methods is that we explicitly control the feet by IK so that stepping motions can be generated without feet sliding. The other difference is that our method is targeted for real-time applications while their method is an offline process.

Capturing Motions

In order to generate arbitrary stepping patterns, we capture various stepping motions for keeping the balance of the body. A subject is first asked to perform the basic standing, walking, and running motions and then pushed/pulled from eight different directions, front, side, front-side, and back-side. The information of foot contact with the ground is estimated from the foot trajectories. The reactive motions are then categorized by their directions and amplitudes of the impact

relative to the frontal direction of the body. The amount of impact is calculated by subtracting the linear momentum of the body before the impact from that after the impact. This information is used as the key when searching for the reactive motion.

Generating Reactive Motions

While the virtual figure is moving by following the primary motion, reactive motion may be computed with the following steps:

1. *Adding impact to the body*: The user specifies the position, amplitude, and direction of the impact and the motion is edited by IK using the new profile of the momentum as constraints.
2. *Motion search*: The counteracting motion with new footstep pattern is searched in the motion database. The one that matches the current foot-ground contact and minimizes the criteria for motion blending is chosen.
3. *Scaling and blending the trajectories of the feet and linear/angular momentum*: The trajectories of the counteracting motion found in the database is scaled according to the strength of the perturbation and blended with those of the primary motion.
4. *Generating the final motion by momentum-based IK*: Using the trajectories of the feet and momentum as constraints, the final motion of the body is generated.
5. Repeating steps 2–4 when applying multiple impacts to the body.

Each step of the method is explained in the following subsections.

Simulating the Impact Using Momentum-Based IK

The reactive motion immediately after the external perturbation is calculated using the impulse force added to the body. The original motion is used as the basic motion and it is edited by using momentum-based IK. First, the position of action for the external perturbation is specified by the user. Based on the relationship of this point and the position of the COM, the amount of momentum added to the body is computed. Let p be the position of the impact and x_g be the position of COM. If we summarized the impact added to the body by a one-time impulse, the increased linear momentum

Copyright © 2005 John Wiley & Sons, Ltd.

215

Comp. Anim. Virtual Worlds 2005; **16**: 213–223

and angular momentum, ΔL, and angular momentum, ΔM, due to an impact F can be calculated as:

$$\Delta L = F$$

$$\Delta M = (p - x_g) \times F$$

The momentum due to the impact will be added to the original motion, and the edited generalized coordinates will be calculated by a momentum-based IK method. Let q be the generalized coordinates of the body. There is a linear relationship between the time-derivative of the generalized coordinates and the momentum as follows:

$$L = m\dot{x}_g = J_L\dot{q} \tag{2}$$

$$Iw = J_{Iw}\dot{q} \tag{3}$$

where J_L and J_{Iw} are the Jacobian matrix that correlates the time derivative of generalized coordinates with the linear momentum and angular momentum, respectively, and m is the total mass of the body.

It is also possible to specify the kinematic trajectories of segments such as the feet, or keep contacts between different segments of the body. Such constraints can be represented by the following form:

$$\dot{x} = J_f\dot{q} \tag{4}$$

where J_f is the Jacobian matrix responsible for the position and orientation of the segments, and $\dot{x}$ represents the linear momentum and angular velocities of the segments in Cartesian space. Using constraints in Equations (2)–(4), the velocity of the generalized coordinates are calculated by minimizing the following quadratic form:

$$(\dot{q} - \dot{q}_0)^T W(\dot{q} - \dot{q}_0) + q'^T W'q' + (\dot{q} - r)^T W''(\dot{q} - r) \tag{5}$$

where $r = -2\dot{q}_{i-1} - \dot{q}_{i-2} + \ddot{q}_0\Delta t^2$, W is a weight matrix which defines the mobility of each joint, W' is a positive diagonal matrix that decides how strictly each constraint in Equations (2)–(4) must be satisfied, $\dot{q}_i, \dot{q}_{i-1}$ are the derivatives of the generalized coordinates in the last two frames, $\ddot{q}_0$ is the acceleration of the generalized coordinates of the previous frame in the original data, Δt is the interval between the frames, and

W'' is a weight matrix to keep the acceleration of the generalized coordinates similar to those in the original data. The values of elements of W' are set high for the feet in case they are supporting the body. This is to avoid the supporting foot to slide over the ground. The third term helps to remove the high frequency vibration that often occurs when using IK. Lagrangian method is used to calculate $\dot{q}$ by minimizing Equation (5) subject to the constraint given by Equations (2)–(4). As a result, it is possible to obtain a new trajectory of the body based on the new momentum profile.

Searching for Reactive Motion in the Database

In case it is difficult to counteract the impact using the original stepping pattern, it is necessary to use another pattern to keep the balance. The most appropriate motion will be searched in the database and then blended into the current motion. Since the reactive motions are categorized by the impact of the external perturbation, this impact is used as the key for the search. The reactive motion which includes a frame that minimizes the following criteria will be selected from the database:

$$J = \sum_{i=l,r} k_i||v_{i,1} - v_{i,0}||^2 + k_2||I_0 - I_1||^2 \tag{6}$$

where $(v_{l,0}, v_{r,0}, I_0)$ are the data from the last frame of the impact phase, representing the vector between the left toes and the COM, the vector between the right toes and the COM, and the moment of inertia of the body, respectively. $(v_{l,1}, v_{r,1}, I_1)$ are the corresponding data in the initial frame in the reactive motions, and k_l, k_r, k_2 are weighting parameters. When comparing two postures, the bodies are aligned so that the vector connecting the two feet projected onto the ground become parallel.

All the frames in the reactive motions before the foot of the swing-leg lands onto the ground are scanned. The frame that minimizes Equation (6) is chosen as the best matching frame in each reactive motion. It is possible to see that the criteria by Equation (6) is much simpler than the one used by other graph-based motion stochastic methods such as Equation (1). The matching is done based on the minimum requirements to interpolate the two motions by momentum-based IK. Due to the simplicity of the criteria, more motions can be scanned in a short amount of time, which will benefit users to generate interactive animations.

Editing the Trajectories of Momentum and Feet

The trajectories of the linear momentum and angular momentum must be edited according to the impact added to the body so that it is counteracted and compensated correctly. The positions of the foot steps must also be changed according to the amplitude and direction of the impact. A method to edit the trajectories of momentum and feet based on the observation of the human motion is explained in this section.

Observation of the Momentum Profile of Humans. The trajectories of the linear momentum and angular momentum during an ordinary gait motion are shown in Figures 1(a) and 2(a), respectively. Regarding the linear momentum, the body decelerates in the initial half of the single support phase, and accelerates in the latter half. Regarding the angular momentum around the COM, at the beginning of the single support phase, the angular momentum has a constant positive value which has an effect to wake up the torso and move the swing-leg to the front. By the end of the single support phase, the value becomes negative which has an effect to turn the body to the front. At the moment the swing-leg lands onto the ground, the value quickly turns to positive.

How will these profiles change when an external perturbation is added to the body? The profiles of the linear momentum and angular momentum when the body is pushed at the back while walking are shown in Figures 1(b) and 2(b). As a result of the impact, the velocity of the COM will increase, and the angular momentum during the initial half of the single support phase will decrease. Then, the body will swing out the foot further and increase the length of the stride.
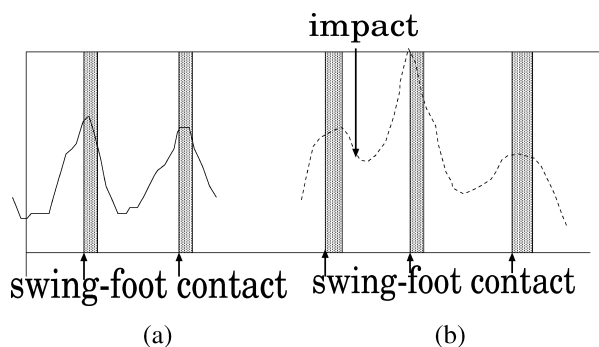

Figure 1. The linear momentum (a) during ordinary gait, and (b) when the back is pushed while walking.
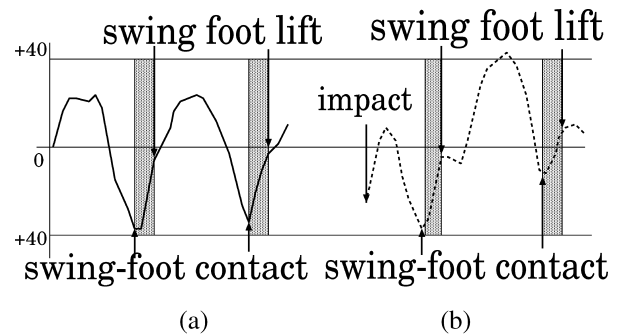

Figure 2. The angular momentum (a) during ordinary gait, and (b) when the back is pushed while walking.

By increasing the stepping distance, it is possible to decelerate the body more during the next single support phase. In addition to that, the angular momentum around the COM can also be reduced more by the collision of the swing leg with the ground. During the next single support phase, a rotational moment to increase the angular momentum is generated by the ground force so that the torso is brought back to the upright position. Similar patterns can be observed for impacts from different directions.

To summarize, when editing the linear momentum and angular momentum of the body to simulate momentary impact, first the effect of the impact must be added to the original curve. Next, the strides for the following steps must be changed. Finally, the profiles of the linear momentum and angular momentum must be edited so that the posture of the body becomes similar to that of the original motion at the end of the single support phase.

Editing the Linear Momentum by Changing the Position of Foot Steps. As explained in section, the impact generated by the user's interaction with the character is used as a key to search for the reactive motion. Let this impact be F_r, and the impact that induced the selected reactive motion be F_o. Usually there is a gap between F_r and F_o, and therefore the reactive motion must be adjusted according to the gap before blending the reactive motion to the primary motion.

Raibert [9] has experimentally shown that the velocity of the body is linearly dependent on the displacement of the landing foot onto the ground. This relationship is also used to control the velocity of biped robots. [10] Consider the situation as shown in Figure 3. A single support phase begins when the COM is at Point A and ends when it is at Point B. Assuming that the horizontal
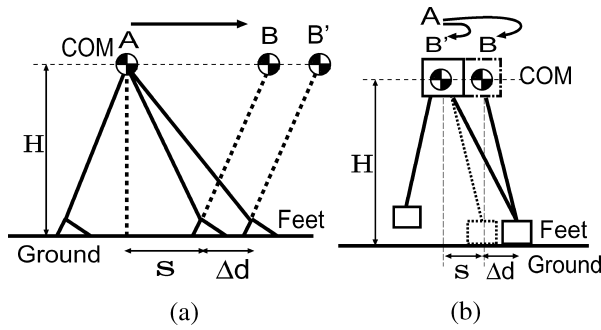
Figure 3. The relationship of the COM and the supporting foot (a) in the sagittal plane and (b) and frontal plane.

distance between Point A and the support foot in the sagittal plane is s and the average height of the COM is H, if the position of the support foot is translated by Δd, the decrement of the velocity at the beginning of the next double support phase can be approximated by

$$\Delta v = -\frac{\Delta d}{v_A}\dot{v}_A$$

where v_A and $\dot{v}_A$ are the horizontal velocity and acceleration of the COM at point A, respectively. The position of the foot landing onto the ground shall be edited according to $F_r - F_o$. The displacement of each step along the anterior axis can be calculated by

$$\Delta d = -\frac{F_{r,z} - F_{r,z}}{mn_{step}\dot{v}_A}v_A$$

where $(F_{r,z}, F_{o,z})$ are the anterior elements of (F_r, F_o) and n_{step} is the number of foot steps conducted in the reactive motion to recover the balance. It is possible to apply the same theory in the frontal plane. By stepping further to the exterior, the lateral velocity of the COM (Figure 3(b)) can be reduced. The distance of adjustment in the sagittal plane and the frontal plane can be calculated independently.

After the position of the feet landing on the ground are decided, the 3D trajectory of the foot of the swing leg is calculated by scaling its original trajectory in the horizontal plane. We use the position of the COM as the input to decide the position of the feet. Let us assume here in the original motion the relationship of the swing-foot and the COM is represented by

$$p = f(x_g)$$

where p is the position of the swing-foot relative to the position of the supporting foot, x_g is the position of the COM, and f is the function that correlates them in the captured motions. Then, in the newly edited motion, the new trajectory of the swing-foot will be obtained by

$$p' = \frac{L_2}{L_1}f(x_g)$$

where p' is the new trajectory of the swing-foot relative to the support foot, and L_1, L_2 are the length of the strides before and after changing the landing position of the swing-foot.

Editing the Angular Momentum. The angular momentum of the body induced by the user interaction rarely matches with that of the captured reactive motion. Let the difference of the latter with the former at the moment the blending starts be ΔIw.

In Figure 2(b), it can be observed that the difference of the angular momentum due to the impact is decreased to almost zero by the end of the first contact of the swing-foot with the ground. Hence, we will model the trajectory of the angular momentum during this period by the following equation:

$$Iw_I = Iw_0 + \left(1 - \frac{t - t_i}{t_l - t_i}\right)\Delta Iw \qquad (7)$$

where Iw_I is the angular momentum during this phase, Iw_0 is the angular momentum in the original reactive motion, t_i is the time of the impact, and t_l is the time the swing-foot contacting the ground.

During the following double and single support phases, an angular momentum that has an opposite effect to the impact will be added to the body. This angular momentum is needed to recover the original posture. The angular momentum during this period is calculated as follows:

$$Iw_{II} = \begin{cases} Iw_0 + \dfrac{t - t_l}{t'_i - t_l}M & \left(t_l < t < \dfrac{t_l + t'_l}{2}\right) \\[2ex] Iw_0 + \dfrac{2t - t'_l - t_l}{2(t'_l - t_l)}M & \left(\dfrac{t_l + t'_l}{2} < t < t'_l\right) \end{cases} \qquad (8)$$

where Iw_{II} is the angular momentum during this phase, and $M = -\frac{1}{2}\Delta Iw(t_l - t_i)$, and t'_l is the time the body recovers the balance. Figure 4 shows an example of editing the angular momentum profile of a walking motion. The angular momentum during Phases I and II are calculated by Equations (7) and (8), respectively.
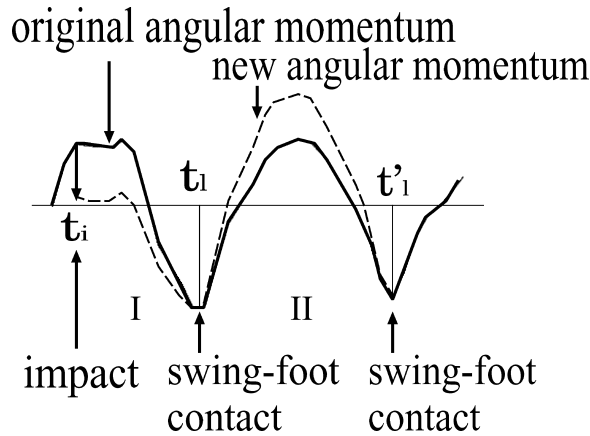
Figure 4. An example of editing the angular momentum of a bipedal gait.

Generating Reactive Motion by Motion Blending

Once the reactive motion is chosen and edited, this motion will be blended into the primary motion using momentum-based IK as explained in section. For exam-ple, assuming that the trajectories of the feet, linear momentum, and angular momentum are represented by $(x_0(t), L_0(t), Iw_o(t))$ in the primary motion and by $(x_1(t), L_1(t), Iw_1(t))$ in the reactive motion, their new trajectories during the blending process will be calculated by

$$(x_\alpha(t), L_\alpha(t), Iw_\alpha(t))^T = (1-\alpha)(x_0(t), L_0(t)Iw_0, (t))^T + \alpha(x_1(t), L_1(t), Iw_1(t))^T$$

Using these data, the trajectories of the generalized coordinates will be calculated by IK as explained in Section. Instead of using Equation (5) as the objective function, the following criteria is minimized:

$$(\dot{q} - \dot{q}_\alpha)^T W(\dot{q} - \dot{q}_\alpha) + q'^T W' q' + (\dot{q} - r_\alpha)^T W''(\dot{q} - r\alpha)$$

where

$$\dot{q}_\alpha = (1-\alpha)\dot{q}_0 + \alpha\dot{q}_1$$

$$r\alpha = -2\dot{q}_{i-1} - \dot{q}_{i-2} + \ddot{q}_\alpha \Delta t^2$$

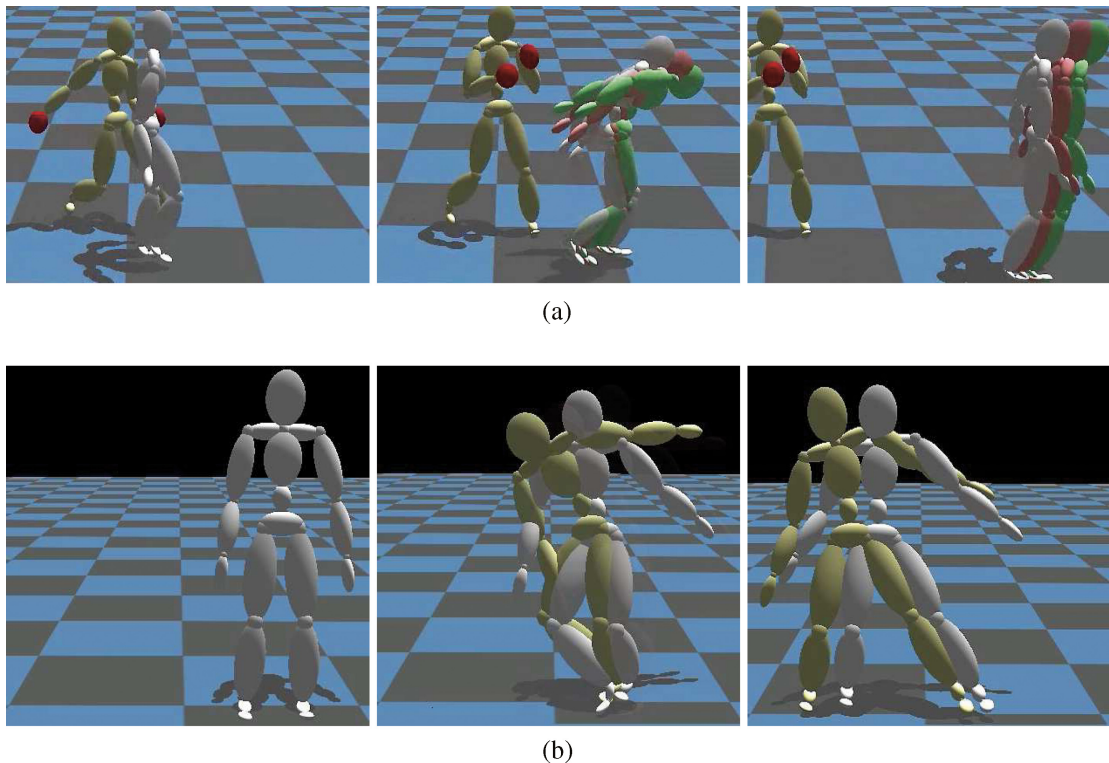and $\ddot{q}_\alpha$ is the alpha-blended acceleration of the original motions.

(a)

(b)

Figure 5. Editing the reactive motion with different levels of impact: (a) hitting the head from the back and (b) hitting the torso from the side.

Experimental Results

First, an example of editing the reactive motions by changing the amplitude of the impact is presented. By hitting the character with different levels of impact, the reactive motion selected will be adjusted by changing the position of the foot steps, and changing the trajectories of the linear momentum and angular momentum. Examples of hitting the head from the back and the torso from the side are shown in Figure 5(a) and (b), respectively. As the amplitude of the impact is increased, the character steps out further and the body tilts more to counteract the strong impact.

Second, an example of applying different impacts to the body from different directions were generated. A walking motion was used as a primary motion and the impact was added to the body from the front, back, and side. The corresponding reactive motions that match the condition were found and further adjusted to be blended with the primary motion. The reactive motions are shown in Figure 6.

Third, an example of editing the reactive motions by changing the position of the impact applied to the body is presented. By hitting a character from the front, a motion to walk backward was generated. Based on the condition of the impact, the linear momentum and angular momentum profiles were recalculated. Examples of hitting the left shoulder and head are shown in Figure 7(a) and (b). When the left shoulder is hit, a momentum around the vertical axis is generated, and therefore the body rotates around the vertical axis. When the head is hit, the neck moves upward, the torso leans backward, and the arms will rotate as larger angular momentum is generated around the lateral axis. Another example of constraining the arms to the back when hitting the head is presented in Figure 7(c). Because the arms are fixed, the whole chest is bent backward to generate angular momentum that was originally generated by the arms.
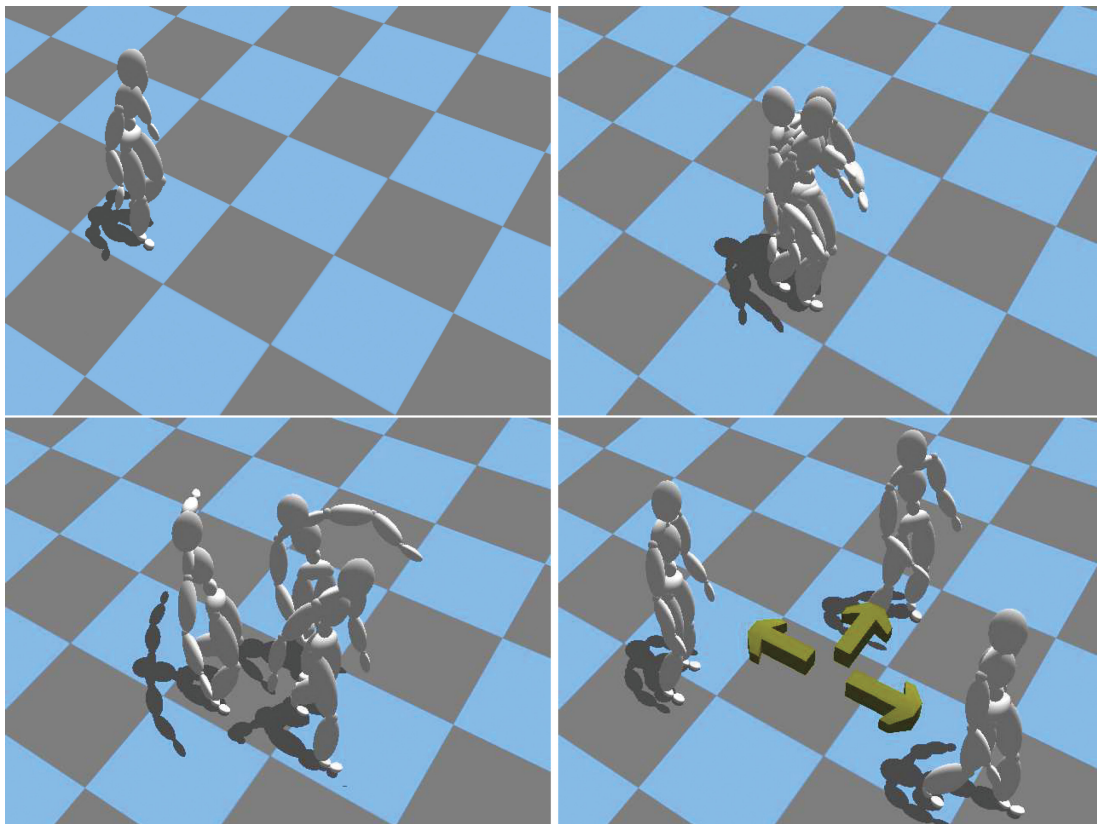

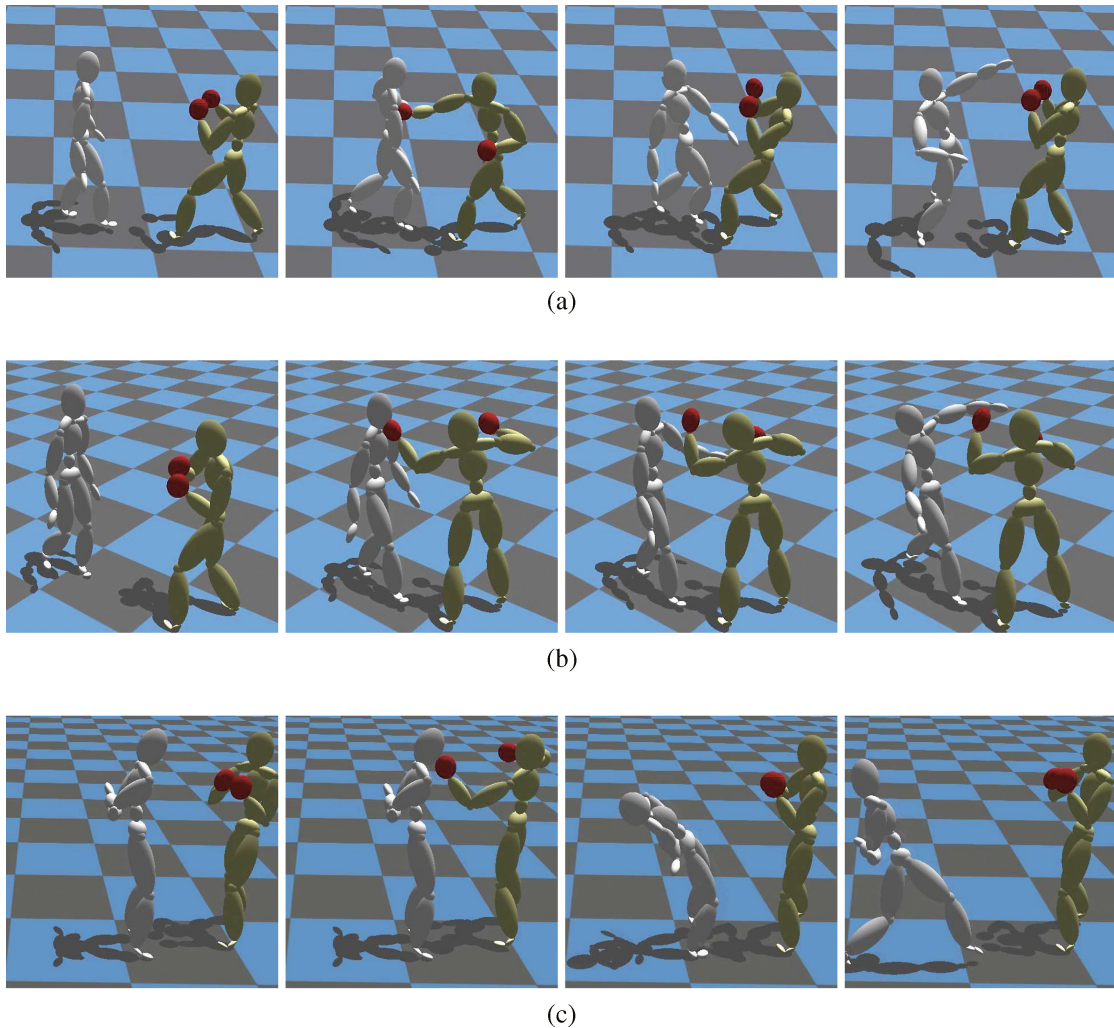*Figure 6. Perturbation added to the character from various directions while walking.*

(a)

(b)

(c)

Figure 7. The motion generated by (a) additing impact to the left shoulder, (b) to the head, and (c) hitting the head while constraining the arms to the back.

Finally a demo to repeatedly add impact was generated, as shown in Figure 8. The main character was hit by multiple other characters and according to the direction and strength of the impact, the appropriate reactive motion is searched and blended into the current motion.

In order to analyze the performance of the motion search algorithm, the computational time to search for the motions were measured. We assume 30 frames are compared per each motion, which is regularly longer than the time the swing-foot lands onto the ground after the impact is added to the body. First the performance using Equation (1) which is a method based on Lee *et al.*[4] was analyzed. The results are shown in Table 1. It is

possible to see that this criteria does not suit real-time applications due to its high computational cost. The results using the criteria written in the form of

No. of motion	Total time millisecond	% of proc. time per frame (30 fr/second)
10	33	97.24
40	129	388.97
70	226	680.70

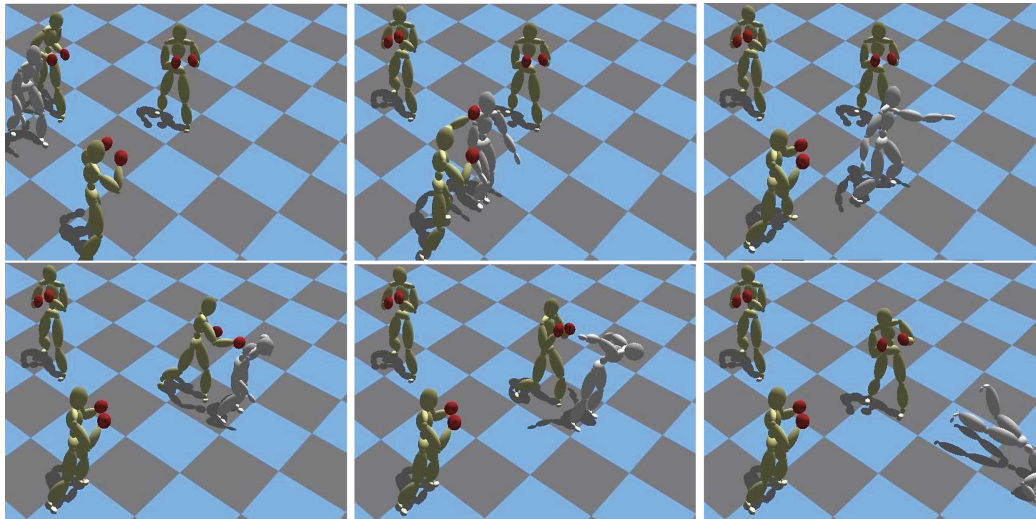**Table I. The performance of motion search using Lee *et al.*'s criteria[4]**

Figure 8. Hitting the character repeatedly from various directions until it falls down onto the ground.

Equation (6) is shown in Table 2. In our method, the cost is only 22% of the total available time for one frame (if the frequency of the animation is 30 frames per second) even 70 motions are scanned. Therefore, the method is applicable for real-time applications. The system has run on a machine with a P4 2.6 GHz processor, 1 GB RAM and a nVidia GEFORCE FX 5900 display card. The frame rates of the input motions are standardized at 30 Hz. By using a human model with 63 degrees of freedom, the system can process the motion at 11.71 Hz with our IK engine.

Summary and Discussion

In this paper, we have proposed a new method to generate reactive motion when an external perturbation is added to the body. The method is based on momentum-based IK. The reactive motion with the most appropriate foot steps is selected from the databased and adjusted so that it can be naturally blended into the motions after the impact.

No. of motion	Total time millisecond	% of proc. time per frame (30 fr/second)
10	1.0	3.21
40	4.3	12.87
70	7.5	22.52

Table 2. The performance of motion search using our method

The momentum-based IK approach is most suitable to generate reactive motion such as to step out a few steps and recover the balance, as it can handle constraints such as the motion of the feet and the linear/angular momentum at the same time. The positions of the feet stepping on the ground are very important when creating the recovery motions. Foot sliding can easily happen if the primary motion and the reactive motion is simply interpolated. The momentum-based constraints help simulate effects such as hitting or pulling the body, and also keep the motion realistic even after editing the motion or adding new constraints, such as fixing the hands.

Since the criteria to select the motion in the database is much simpler than previous methods such as in Lee *et al.*,[4] the computation time for the motion search is much shorter. We do not mean that our criteria is superior in all aspects. For offline process, such as Motion Graph, there is no doubt that Equation (1) works better. As all the generalized coordinates and their derivatives are compared precisely, all the connected motions will look natural by motion blending. However, for online searching and motion blending, as required in this study, such methods are not suitable due to high computational cost.

Therefore, we believe that the approach proposed in this paper is one of the suitable approach for generating real-time animations of motions reacting to external perturbations. For the future research, we are planning to extend our method to enable generating animations of several characters densely interacting with each

other. Such methodology will be useful for generating scenes of sports such as rugby and American football.

References

1. Zordan V, Hodgins J. Motion capture-driven simulations that hit and react *Proceedings of ACM Computer Animation*, 2002.
2. Oshita M, Makinouchi A. A dynamic motion control technique for human-like articulated figures. *Computer Graphics Forum* 2001; **20**(3): 192–202.
3. Komura T, Leung H, Kuffner J. Animating reactive motion for biped locomotion. *Proceedings of ACM VRST*, 2004, pp. 32–40.
4. Lee J, Chai J, Reitsma P, Hodgins J, Pollard N. Interactive control of avatars animated with human motion data. *Proceedings of ACM SIGGRAPH* 2002; **21**(3): 491–500.
5. Kovar L, Gleicher M, Pighin F. Motion graphs. *Proceedings of ACM SIGGRAPH* 2002; **21**(3): 473–482.
6. Arikan O, Forsyth D. Motion generation from examples. *Proceedings of ACM SIGGRAPH* 2002; **21**(3): 483–490.
7. Mandel M. *Motion Capture and Physical Simulation for Character Animation*. PhD thesis, CMU, 2004.
8. Zordan V, Majkowska A, Chiu B, Fast M. Dynamic response for motion capture animation. *Proceedings of ACM SIGGRAPH* 2005; **24**(3): 697–701.
9. Raibert M. *Legged Robots that Balance*. MIT Press: Cambridge, MA, USA, 1986.
10. Kajita S, Kanehiro F, Kaneko K, Fujiwara K, Yokoi K, Hirukawa H. A real-time pattern generator for biped walking. *Proceedings of IEEE International Conference on Robotics and Automation*, 2002.

Authors' biographies:

Taku Komura is currently an assistant professor in the Department of Computer Science at City University of Hong Kong. Before joining City University of Hong Kong, he worked as a postdoctoral researcher at RIKEN, Japan to simulate various human motion using the musculoskeletal model. He received his Ph.D. (2000), M.Sc. (1997), and B.Sc. (1995) in Information Science from the University of Tokyo. His research interests include human motion analysis and synthesis, physically-based animation and real-time computer graphics. His research area covers computer graphics, robotics and biomechanics.

Edmond S.L. Ho received the B.Sc. degree (first-class honors) in Computer Science from the Hong Kong Baptist University in 2003. He is currently a research student in the Department of Computer Engineering and Information Technology, City University of Hong Kong. His research interests include computer animations.

Rynson W.H. Lau received a (top) first class honors degree in Computer Systems Engineering in 1988 from the University of Kent at Canterbury, England, and a Ph.D. degree in Computer Graphics in 1992 from the University of Cambridge, England. He is currently an associate professor at the City University of Hong Kong. Prior to joining the university in 1998, he taught at the Hong Kong Polytechnic University. From 1992 to 1993, he worked at the University of York, England, on a defense project on image processing. Rynson LauÀÎs research interests include computer graphics, virtual reality and multimedia systems.