Wrestle alone : Creating tangled motions of multiple avatars from individually captured motions

Edmond S.L. Ho Taku Komura
Institute of Perception, Action, and Behavior, University of Edinburgh, U.K.

Abstract

Animations of two avatars tangled with each other often appear in battle or fighting scenes in movies or games. However, creating such scenes is difficult due to the limitations of the tracking devices and the complex interactions of the avatars during such motions. In this paper, we propose a new method to generate animations of two persons tangled with each other based on individually captured motions. We use wrestling as an example. The inputs to the system are two individually captured motions and the topological relationship of the two avatars computed using Gauss Linking Integral (GLI). Then the system edits the captured motions so that they satisfy the given topological relationship. Using our method, it is possible to create / edit close-contact motions with minimum effort by the animators. The method can be used not only for wrestling, but also for any movement that requires the body to be tangled with others, such as holding a shoulder of an elderly to walk or a soldier piggy-backing another injured soldier.

1 Introduction

3D computer animations of multiple avatars tangled with each other are the most difficult to be created. Such scenes often appear in wrestling matches or in battle fields. People will imagine tangled motions can be created either by using a motion capture system, or by keyframe animation. There can be, however, a lot of problems doing that. For example, if an optical motion capture system, which is the most commonly used tracking system today, is used to capture wrestling motions of two avatars, there will be difficulties as the actors will occlude the markers attached to their bodies. In case magnetic, mechanical or inertia sensors are used, there will be no occlusion problem, but the bulky structure of the device might endanger the actors when strong impact are added to their bodies, and the devices can also be easily damaged. Creating the movements of the avatars by keyframe animation for wrestling scenes is a headache to the animators as they have to carefully edit the postures of the avatars to avoid collisions and penetrations of the segments while keeping the segments tangled.

Although there is a big demand in the industry, there has not been much research to tackle this problem. In this paper, we propose a new method to generate animations of two persons getting tangled with each other using individually captured motions. Since the motions are captured separately, we need to specify how they should be tangled with each other. Such relationships are difficult to be expressed by positional constraints [2] as the contact area is not always the same. For example, Liu *et al.* [3] generated animations of two avatars interacting using spacetime constraints and positional constraints. However, it is difficult to generate tangled movements such as those appearing in wrestling using their method.

In this research, we propose to use Gauss Linking Integral (GLI), which is a concept that has been developed in the knot theory. Since GLI can only represent the relationship of two strands, we propose a method to apply it to express the tangled status of multibody structures such as humans. Once the relationship is specified, the motions of the characters are imported and edited automatically, so that constraints due to penetration or geometrical constraints such as keeping the support feet onto the ground are satisfied. The tangle relationships are also monitored so that the segments do not get untangled. As a result, a scene of two avatars interacting with each other can be generated. We show various examples of such scenes in the experimental section of this paper.

Our method can be used to create motions such as holds and chokes in wrestling, a helper holding a shoulder of an injured person to walk or a person piggy-backing another person. The motions created using this method can be used for applications such as computer games and 3D computer animation.

2 Overview of the Method

In this paper, we propose to capture the motions of the avatars individually, and combine them in order to create the final scene. Here is the overview of our methodology.

1. The user captures the motions of the two avatars individually using a motion capture system
2. The user specifies the topological relationship of the avatars by composing a template posture with our 3D avatar posing interface (Section 3). The postures are examined by the system and the segments composing the tangles are detected by calculating the GLI of the segments (Section 4).
3. The motion data of both avatars are edited according to the topological relationship specified in Step 2 (Section 5)

The flowchart of the algorithm is shown in Figure 1.

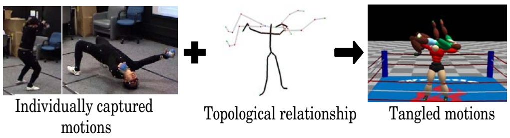


Figure 1. The outline of the proposed method

3 The Topological Relationship Template

After capturing the motions of the avatars individually, we need to specify how the two bodies are tangled with each other. We use a simple inverse kinematics (IK) interface for this purpose. We call the created postures the template postures. The advantage of our method is that we can save the template postures in the database and re-use it for creating similar tangled motions by a different set of motion data.

A snapshot of the IK interface is shown in Figure 2. The user first loads an image of a photograph that multiple persons are tangled with each other into the background. The user then edits the postures of the two skeleton models so that the position of their joints overlap with those of the persons in the background image. The user can also specify segments which positions should be constrained with other parts of the body. For example, if the hand of the attacker is supposed to be holding the hand of the opponent, a positional constraint is added between the two hands. Constraints between the supporting feet and the ground can also be added to keep their location static throughout the motion.

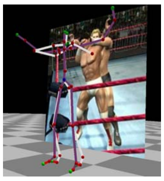
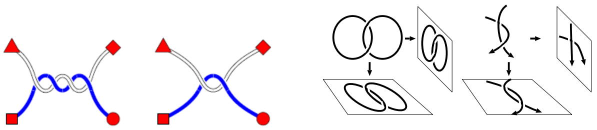


Figure 2. IK interface for creating topological relationships with an image for reference

4 Detecting the tangles made by the bodies

To represent the relationship of the two bodies, we use the concept called 2-tangles [1] in knot theory. A 2-tangle is defined as a pair of strings which end points are fixed in Euclidean 3D space.

Two examples of 2-tangles are shown in Figure 3 (a). Any tangles of the bodies can be represented by 2-tangles, or a set of 2-tangles. In knot theory, it is always assumed that the tangles or knots are projected onto a plane. As knots or links are closed curves, the projection plane does not affect invariants such as the minimum crossing numbers (Figure 3 (b), left). However, in case of tangles, the projection plane affects the number of crossings (Figure 3 (b), right) and as a result, we cannot use the crossing numbers as a feature.



(a) (b)

Figure 3. (a) Examples of 2-tangles with four crossings (left) and two crossings (right) (b) Projection planes do not affect minimum crossing numbers for links (left) but do for tangles (right)

4.1 Gauss Linking Integral

In order to detect whether the segments are tangled or not without projecting the bodies onto a plane, we compute the Gauss Linking Integral (GLI) of the segments. The GLI of two directed curves γ_1 and γ_2 can be computed by

$$GLI(\gamma_1,\gamma_2) = \frac{1}{4\pi}\int_{\gamma_1}\int_{\gamma_2}\frac{d\gamma_1 \times d\gamma_2 \cdot (\gamma_1 - \gamma_2)}{\|\gamma_1 - \gamma_2\|^3} \qquad (1)$$

where $\times$ and $\cdot$ are cross and dot product operators, respectively. This becomes the signed average number of crossings when viewing the tangle from all directions. In case the absolute value of the GLI is over 1 (Figure 4 left), the two curves are twisting around each other once. If it is over 0.5 (Figure 4 middle), the two curves are tangled, which means that they cannot be separated into two by a plane between them without cutting either of them. And if it is less than 0.5 (Figure 4 right), the two curves are untangled. Since we are only interested in knowing whether the bodies are tangled or not, we compute the GLIs for the body segments, and if their absolute values are over 0.5, we tag them as tangled and use these results when editing the motions. The details are explained in the following subsection.

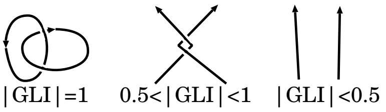
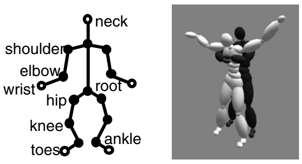


Figure 4. The GLI of directed curves when one strand is surrounding the other (left), singly tangled (middle), and untangled (right)

4.2 Tangles of the human body

We can assume the human body has a tree structure composed of rigid bodies and joints, as shown in Figure 5(a). In this research, we assume the body is composed of twenty rigid body segments and eighteen joints. The proposed method can easily be extended for a more precise model that includes the fingers and toes.



(a) (b)

Figure 5. (a)The graph structure of the body used in this paper. (b) An avatar holding the torso of another avatar from the back.

In order to apply the tangle concept which only handles two open-ended 1D strands to the tree structure of the human body, we will evaluate the tangles made by all existing paths connecting the joints / end effectors. For example, if an avatar holding the torso of another avatar from the back as shown in Figure 5(b), we can say the path between the hands of the rear avatar is making a tangle with the torso of the avatar in the front. This can be written in a mathematical form as

$$|GLI(P_a(lefthand, righthand), P_b(neck, root))| > 0.5$$

where $P_a(lefthand, righthand)$ is the path between the node *lefthand* and *righthand* of the avatar in the back, and $P_b(neck, root)$ is the path between the node *neck* and *root* of the avatar in the front. For generating animation of human avatars tangled with each other, we find out all the tangles in the template posture and try to keep up such tangles throughout the motion when combining the two singly captured motions. We tag all the tangles made by the segments of the bodies which absolute GLI is larger than 0.5.

The GLI satisfies commutativity and is linear in summation; these are convenient features when computing all the tangles made between the segments:

$$GLI(L_1, L_3) = GLI(L_3, L_1) \tag{2}$$

$$GLI(con(L_1 + L_2), L_3) = GLI(L_1, L_3) + GLI(L_2, L_3) \tag{3}$$

where $con(L_1, L_2)$ is the curve made by concatenating L_1 and L_2, which is making a tangle with another curve L_3 (see Figure 6). The pseudocode of the algorithm to find out all the tangles which absolute GLIs are larger than 0.5 is shown in Algorithm 1. What we do here is that in case the absolute GLI is larger than 0.5 and is larger than the absolute GLI of any of its subtangles, we put this tangle into the *TangleList* that we later on use when editing the motions. Sometimes the tangle has a smaller GLI than those made by its subpath. In such case, we do not include the longer path in the TangleList as we assume the sub-path tangle is more important when editing the motion. Two examples of visualizing the absolute GLI values of paths between the human avatars when the attacker is conducting a Argentine Back-Breaker and a Full Nelson Hold from the back are shown in Figure 7. In these holds, the arms of the attacker are tangled with various parts of the opponent; the system can find out such tangles by checking the segments which make a large absolute GLI values with the attacker's arm.

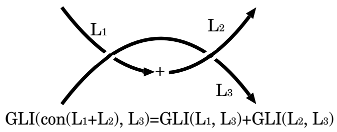


Figure 6. GLI satisfies the commutative rule

5 Simulating the tangled motion

In this section, the method to edit the captured motions so that they satisfy the topological relationship in the template postures is explained. We use physical simulation for this purpose. When running the simulation, the following external forces are added to the bodies: (1) repulsive force due

Algorithm 1 Computation of GLIs of all the local paths

```
for every path $p_1$ in Body 1 do
  for every path $p_2$ in Body 2 do
    if $|p_1| = |p_2| = 1$ then /* the number of segments composing both paths are 1 */
      Calculate $GLI(p_1, p_2)$ using Equation 1
    else
      Calculate $GLI(p_1, p_2)$ by summing the GLI of its components using Equation 3.
      if $|GLI(p_1, p_2)| > 0.5$ and $|GLI(p_1, p_2)| > |GLI(p'_1, p'_2)|$ for arbitrary subpaths $p'_1$ and $p'_2$ of $p_1$ and $p_2$ then
        Add $Tangle(p_1, p_2)$ to the $TangleList$
      end if
    end if
  end for
end for
```

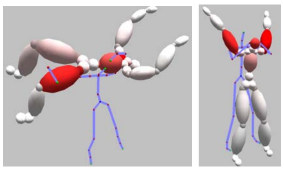


Figure 7. Visualization of the absolute GLI of the arms of the attacker and each segment of the opponent. The more red the segment is, the larger the absolute GLI value is.

to collisions ($\tau_{collision}$), (2) elastic and damping force at the joints to approach the target posture (τ_{PD}), (3) gravitational force (G), and (4) elastic force to keep the tangles (τ_{tangle}). The acceleration of the generalized coordinates of the two bodies are updated by the following equation

$$\begin{aligned} M\ddot{q} &= V(q,\dot{q}) + G + \tau_{PD} + \tau_{tangle} + \tau_{collision} \quad (4) \\ 0 &= C\ddot{q} + \dot{C}\dot{q} \quad (5) \end{aligned}$$

where q are the generalized coordinates of the body, $V(q,\dot{q})$ is the Coriolis force, G is the gravitational force, C is the Jacobian matrix to represent positional / rotational constraints due to contact of the body segments. PD control is used to guide the avatars towards the individually captured motions. Feedback torques τ_{PD} are added to the joints according to the difference of the joint angles / angular velocities with those of the original captured motions. In order to keep the topological relationship between the two bodies, we need to make sure that the tangles do not get untangled. We pull the segments composing the tangle towards each other by elastic force (τ_{tangle}) to keep up the tangle. These forces are virtual external forces which actually do not exist in real wrestling or tangled motions; they are added here to keep up the tangles in the template postures.

When creating the animation, we first set the posture of each avatar to those of the template postures, and set the desired posture to the initial frame of the captured motions, and use Equation 4 and 5 to update them until convergence. The converged posture is used as the initial posture for the animation. We then start to proceed time, and update the postures using the acceleration computed by Equation 4 and 5.

6 Experimental Results and Future Work

We have simulated a number of wrestling motions including the Argentine Back-Breaker (Figure 8 (a)), the Rear-Chokehold (Figure 8 (b)), and the Octopus Hold (Figure 8 (c)). The readers are referred to the attached video to view the individual motions.

(a) (b) (c)

Figure 8. (a) Argentine Back Breaker, (b) The Rear-Chokehold and (c) the Octopus Hold simulated using our method.

In this research, we do not allow the topological relationship to change. Therefore, we cannot create animations of two separate avatars approaching and starting to entangle the segments around each other and finally arriving to tangled postures. In order to simulate such effects, a lot of issues such as path planning have to be coped with; however, this is an interesting problem to tackle and we will leave it for future research.

Acknowledgment This project was partly supported by a CERG grant from the Research Grant Council of Hong Kong (RGC Reference No.:CityU1149/05).

References

[1] J. Conway. *An enumeration of knots and links*. Pergamon Press, Oxford, 1969.

[2] M. Gleicher and P. Litwinowicz. Constraint-based motion adaptation. *Journal of Visualization and Computer Animation*, 9:65–94, 1998.

[3] C. K. Liu, A. Hertzmann, and Z. Popovic. Composition of complex optimal multi-character motions. *ACM SIGGRAPH / Eurographics SCA* , pages 215–222, 2006.

[4] R. Smith. Open dynamics engine. www.ode.org. 2005.