Rigid Link Chains in Kung Fu Panda 3

Jason P. Weber*
DreamWorks Animation

Figure 1: *belt with chains*

1 Introduction

The villain of *Kung Fu Panda 3* has a penchant for chains. In addition to the links adorning his belt, he uses long chains, attached to a pair of wide bladed swords, to control or destroy the environments that he encounters. To give our artists the power to freely drive these chains, we developed a series of tools focused on interactive user control, all without dismissing the harsh rigidity of solid metal.

2 Requirements

2.1 Rigid Links

In order to capture the unyielding nature of forged iron, we impose the rule that metal does not stretch or bend. Each link that is rendered is identical in shape to its original reference model. The link models may only be transformed by rotation and translation.

2.2 Separable Stages

The pipeline needs to provide tools that can be used piecemeal. For example, the tool to apply links to an arbitrary curve can accept that curve from any custom interactive tool or even a hair simulation.

2.3 Artistic Control

While using a generic rigid body simulation is not excluded as an option, many shots are heavily art-directed. This regularly requires artists to directly guide objects, often to match hand-drawn sketches tied to specific frames. This demands that we have tools that cater to interactive, explicit authoring of curves.

3 Approach

Focusing on fully artistic authoring, chains are assembled in three stages, with the general intent is that the user is insulated from the huge numbers of links and control vertices by intuitive abstractions.

*e-mail:jason.weber@dreamworks.com

SIGGRAPH 2015 Talks, August 09 – 13, 2015, Los Angeles, CA.
ACM 978-1-4503-3636-9/15/08.
http://dx.doi.org/10.1145/2775280.2792580

3.1 Waypoints

We expect the artist to be comfortable locating and maintaining contact points at important inflections in the chain, which we refer to as waypoints. These points are far coarser than we would expect for control vertices. At some point in a shot, our villain may have a short set of waypoints such as "hilt, arch, elbow, hand". Most waypoints are keyed relative to a moving surface, such as a character's skin, but others can be in world space. In addition to simple local offsets, a radial component may be set by the artist to push curves out of implied cylinders. This allows chains to bend around the arms and legs, or even the neck or back, without a lot of mundane tweaking by the artist.

3.2 Curves

Since the artists may be carefully specifying exact locations, it is reasonable for them to expect that the chains go through the points they define. If we treat waypoints as simple control vertices and then use a b-spline (like a NURBS), then the approximating interpolation will tend to pull the result far away from the linear path, particularly around the control vertices. Additionally, loops in the input path, which are often intentional, can be washed out entirely. An alternative is to use intersecting interpolation, such as with a Cardinal spline. However, that method has it own problems, particularly with uneven spacing in the input. If two control vertices are arranged particularly close to a sharp turn (a common situation when using the waypoints), the evaluation of the spline will provoke a wild loop to form on the outside of the turn. A workable solution is to not use splines at all. We implemented bisecting matrix rational power functions that produce smooth, even curves through all the input points, even when there is seemingly haphazard spacing. Additionally, since the matrix interpolation also incorporates the normals, clean continuous twisting information falls out naturally.

3.3 Links

Starting from one end of the curve, links are added sequentially. Each link has a pitch, yaw, and twist relative to the previous link. The original reference links are automatically scanned for precise pivot points. The pivot points for pitch and yaw are necessarily in different locations, generally offset by the diameter of the link's tubular cross-section. Without end constraints, each link commonly twists 90 degrees from the previous. But since both ends can be constrained by location and by twist orientation, an iterative solver can adjust the distance and twist of each link to reach a desired goal. Additionally, a temporal mode can remember solutions from past frames to not only maintain stable results over time, but defer sudden changes, producing a more physically familiar behavior.

4 Results

Since artists only need to manipulate a few waypoints, many shots that may have previously been budgeted for a week or two can be ready to show after only a few hours. After getting feedback, an artist's job can often be as simple as bringing up a camera view and dragging a live chain to match a few draw-over sketches. Many upstream changes, like minor adjustments in animation, can often be resolved with little or no effort by the user.