

# AutoSpline: Animation Controls Only When and Where You Need Them

Mark Hessler, Jeremie Talbot  
Pixar Animation Studios

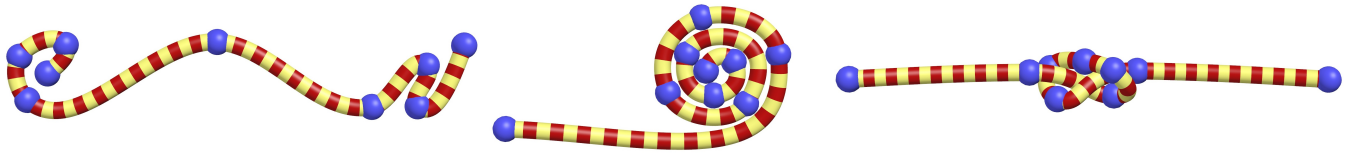


Figure 1: Various AutoSplines demonstrating smooth automatic tangent behavior and control repositioning to focus detail.

## Abstract

Unsimulated character animation of items like ropes or snakes is challenging in the amount of articulation required and the degrees of freedom needing control. At Pixar we developed a powerful new system for animating highly directible curves: AutoSpline. This system is designed to give animators enormous control, without the overhead of managing every aspect of every control when only simple shapes or motions are required.

We accomplished this by providing the ability to switch sets of controls between manual and automatic modes on a frame-by-frame basis. Controls can also be dynamically repositioned per frame so that animators can focus the available controls where highest detail is needed for any given shape or motion. The AutoSpline architecture also includes advanced constraint abilities, simplifying historically difficult interactions between the flexible object and other objects in the scene.

While these features made it feasible to animate the many-tentacled Hank in Pixar's *Finding Dory* as a primary character, AutoSpline was built as a general solution. Early versions were successfully used in Pixar's *Inside Out* and *The Good Dinosaur*. Notably, it inspired a similar tool at ILM, used on *Star Wars: The Force Awakens*.

**Keywords:** animated spline, automatic shape, time varying

**Concepts:** •Computing methodologies → Animation; Parametric curve and surface models;

## 1 AutoSpline Architecture

The AutoSpline system appears as two different in-scene primitive types: Knots and KnotSplines. Each Knot is an oriented locator in the scene and a collection of controls. Importantly, each Knot is a first-class primitive in Presto and may be constrained to other primitives in the scene, or have other primitives constrained to it. This is extremely powerful in allowing modular and hierarchical rigging as well as critical to realistic interactions between on-screen elements.

---

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s). © 2016 Copyright held by the owner/author(s). SIGGRAPH '16, July 24-28, 2016, Anaheim, CA, ISBN: 978-1-4503-4282-7/16/07 DOI: <http://dx.doi.org/10.1145/2897839.2927439>

A KnotSpline primitive reads Knots and produces as its output a piece of virtual geometry which can be input into any deformer or controller in the rig that normally accepts curves. The KnotSpline can be used both to drive the major deformation of the geometry as well as publish information used by other secondary rigs.

AutoSpline is fully interactive in real time; KnotSplines update as their Knots are edited. The generated shape of the KnotSpline is controlled entirely by the state of its Knots on the current frame; it is not history dependent.

## 2 Dynamic Controls

The generated curve at a minimum intersects the location of each Knot. As additional Knot controls are engaged, the curve can be further directed:

- Tangent of the curve may be driven by the Knot orientation
- Tangent handle lengths may be specified
- Twist around the curve may be driven by the Knot orientation
- What geometry the Knot poses may be changed

These controls are uniformly available at every Knot and are animatable. The animator may choose to switch some or all of these controls into or out of an automatic mode on any frame. Manual modes give the animator full control over that aspect of the shape at that Knot. Putting a Knot into manual tangent orientation, twist and geometry assignment, combined with a constraint on the Knot, greatly reduces or eliminates the complementary animation that is often required to get believable interactions between on-screen objects.

The mode settings on each Knot are independent of each other, and each Knot's modes are independent of other Knots. The ability to control as much or as little as necessary per Knot, per frame, is what makes AutoSpline so powerful.

## 3 Automatic Modes

The automatic modes are what lend AutoSpline its name. When a control is set to automatic, the input values are ignored and the consuming KnotSpline poses that aspect itself, based on the values and control modes of all of the Knots in the KnotSpline.

Simple heuristics are used to ensure a smooth shape with predictable behavior as Knots are animated. Only G1 continuity is generated in order to maintain locality of control. Twist around the spline is interpolated between manually controlled Knots by

following the underlying curvature. The interpolant is a rotation-minimizing bidirectional framework with good stability behavior under deformation.

Control of what geometry a Knot affects is the mechanism which allows Knots to be repositioned to affect different areas along the length of the KnotSpline. In automatic mode, the Knot is decoupled and slides through the posed geometry, affecting only the shape of the curve. Conversely, manual control allows absolute locking of posed topology to a location in space.

## **4 Mode-Switch Compensation**

A critical part of implementing the automatic controls was leveraging Presto's powerful compensation system to enable seamless transitions between automatic and manual modes. When a Knot is switched into a manual mode, Presto is able to query the as-posed information for that Knot from the KnotSpline, and author those values to the controls such that the pose is unchanged.

## **Acknowledgements**

This project was a collaboration between many artists and engineers at Pixar. Special thanks go to Kevin Singleton and Mark Piretti for integrating and deploying the new technology in rigs, Bret Parker and Jayson Price for initial design input, and all of the animators on Finding Dory for providing valuable feedback.