4
Hermite Interpolation

The curve and surface methods of the preceding chapters are based on points. Using polynomials, it is easy to construct a parametric curve segment (or surface patch) that passes through a given one-dimensional array or two-dimensional grid of points.

The downside of these methods is that they are not interactive. If the resulting curve or surface isn't the one the designer wants, the only way to modify it is to add points. Moving the points is not an option because the curve has to pass through the original data points. Adding points provides some control over the shape of the curve, but slows down the computations.

A practical, useful curve/surface design algorithm should be interactive. It should provide user-controlled parameters that modify the shape of the curve in a predictable, intuitive way. The Hermite interpolation approach, the topic of this chapter, is such a method.

Hermite interpolation is based on two points $\mathbf{P}_1$ and $\mathbf{P}_2$ and two tangent vectors $\mathbf{P}_1^t$ and $\mathbf{P}_2^t$. It computes a curve segment that starts at $\mathbf{P}_1$, going in direction $\mathbf{P}_1^t$ and ends at $\mathbf{P}_2$ moving in direction $\mathbf{P}_2^t$. Before delving into the details, the reader may find it useful to peruse Figure 4.1 where several such curves are shown, with their endpoints and extreme tangent vectors.

It is obvious that a single Hermite segment can take on many different shapes. It can even have a cusp and can develop a loop. A complete curve, however, normally requires several segments connected with C^0, C^1, or C^2 continuities, as illustrated in Section 1.4.2. Spline methods for constructing such a curve are discussed in Chapter 5.

The method is called Hermite interpolation after Charles Hermite who developed it and derived its blending functions in the 1870s, as part of his work on approximation and interpolation. He was not concerned with the computation of curves and surfaces (and was actually known to hate geometry), and developed his method as a way to interpolate any mathematical quantity from an initial value to a final value given the rates of change of the quantity at the start and at the end.

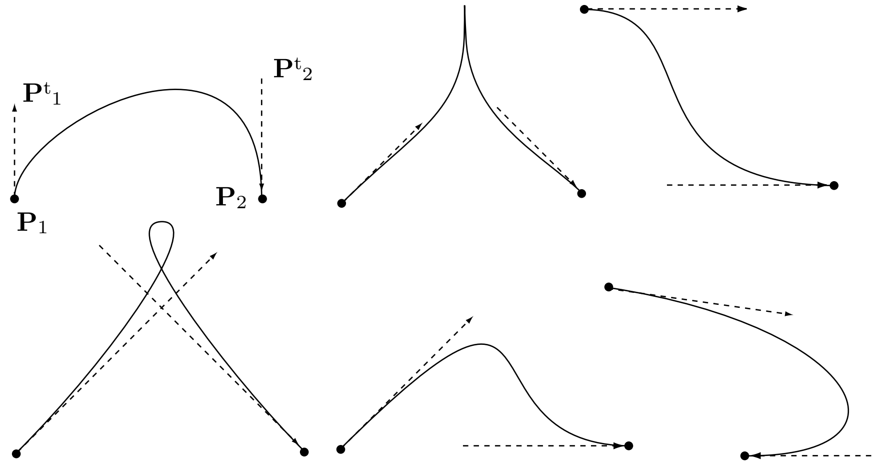

Figure 4.1: Various Hermite Curve Segments.

> [Hermite] had a kind of positive hatred of geometry and once curiously reproached me with having made a geometrical memoir.
>
> —Jacques Hadamard.

4.1 Interactive Control

Hermite interpolation has an important advantage; it is interactive. If a Hermite curve segment has a wrong shape, the user can edit it by modifying the tangent vectors.

◇ **Exercise 4.1:** In the case of a four-point PC, we can change the shape of the curve by moving the points. Why then is the four-point method considered noninteractive?

Figure 4.1 illustrates how the shape of the curve depends on the directions of the tangent vectors. Figure 4.2 shows how the curve can be edited by modifying the magnitudes of those vectors. The figure shows three curves that start in a 45° direction and end up going vertically down. The effect illustrated here is simple. As the magnitude of the start tangent increases, the curve continues longer in the original direction. This behavior implies that short tangents produce a curve that changes its direction early and starts moving straight toward the final point. Such a curve is close to a straight segment, so we conclude that a long tangent results in a loose curve and a short tangent produces a tight curve (see also exercise 4.7).

The reason the magnitudes, and not just the directions, of the tangents affect the shape of the curve is that the three-dimensional Hermite segment is a PC and calculating a PC involves four coefficients, each a triplet, for a total of 12 unknown numbers. The two endpoints supply six known quantities and the two tangents should supply the remaining six. However, if we consider only the direction of a vector and not its magnitude, then the vectors $(1, 0.5, 0.3)$, $(2, 1, 0.6)$, and $(4, 2, 1.2)$ are all equal. In such a case, only two of the three vector components are independent and two vectors supply only four independent quantities.