LOCAL REFINEMENT EDITING OF B-SPLINE SURFACES

David R. Forsey and Richard H. Bartels

Computer Graphics Laboratory, Department of Computer Science
University of Waterloo, Waterloo, Ontario, Canada N2L 3G1
Tel: (519) 888-4534, E-Mail: drforsey%watcgl@waterloo.edu

ABSTRACT

Refinement is usually advocated as a means of gaining finer control over a B-spline surface during free-form surface editing. However, some care and understanding is necessary to restrict the influence of refinement to the locality at which an editing effect is desired. We present a method of localizing the effect of refinement through the use of *overlays.* We also introduce two editing techniques that are effective when using overlays: one is direct surface manipulation through the use of *edit points* and the other is *offset referencing* of control vertices.

KEYWORDS: *B-splines, free-form surface editing, refinement.*

SUMMARY

In designing a prototype B-spline surface editor intended to wrap articulated figures within a single parametric surface, two issues addressed were: the need to superimpose fine-scale surface details on larger scale features, along with a desire that large scale surface movements and distortions carry along such local, fine detail. This has led to a hierarchy of local surface refinements, which we refer to as *overlays*, to a representation of control vertices in terms of *offsets* relative to a hierarchy of local reference frames, and to the investigation of mechanisms for the direct manipulation of *edit points* on surfaces rather than manipulation indirectly through the movement of control vertices.

A uniform, bicubic, B-spline surface consisting of $(m-3)\times(n-3)$ patches is defined by an array of $m\times n$ *level* (0) control vertices $\mathbf{V}_{i,j} = V^{(0)}$ given in some reference frame. A process known as *refinement* for general B-spline surfaces is provided by the Oslo algorithm [2]. Using midpoint refinement, a special case that is particularly simple for uniform B-splines, each patch of the surface, governed by a 4×4 subarray of control vertices $V^{(0)}$, can be re-represented as four smaller patches defined by a 5×5 array of new *level* (1) control vertices $V^{(1)}$. This may be repeated for as many contiguous $V^{(0)}$-defined patches as desired in either direction to yield a corresponding system of contiguous $V^{(1)}$-defined patches [1].

Refinement of the $V^{(1)}$ patches can be carried out in turn to produce even smaller $V^{(2)}$ patches. Disjoint surface patches at any level may be refined separately and independently. We maintain each level along with their descendents at subsequent levels of refinement in a hierarchical data structure. Levels of refinement are encoded as depth in the structure, and disjoint sections of surface are encoded as breadth. The refined patches constitute a hierarchy of *surface overlays* on the originally unrefined surface.

We are interested in manipulating (sub)patches at all levels of refinement. Manipulation of surface patches at level (i) is associated with relatively large-scale changes to an area of the surface, while manipulation of patches at level $(i+1)$ is associated with finer changes of a more local area. Changes to patches at level $(i+1)$ cause them to depart from the surface defined at level (i). If these departures occur in the central region of an area of patches, and if only the highest refinement level is displayed at any point on the composite surface, the composite maintains its integrity to the viewer. If arbitrary changes are allowed at level $(i+1)$, however, the patches about the periphery of the area under manipulation may tear away from the patches that constitute the surface at level (i) of refinement. If we are careful to move only the central control vertex of a sufficiently large region of $V^{(i+1)}$ patches (e.g. the central vertex of a 7×7 (or larger) array of $V^{(i+1)}$ vertices) the $V^{(i+1)}$ portion of the surface under manipulation will maintain its continuity exactly with the underlying $V^{(i)}$ surface. As this is continued to levels of greater refinement, detail can be added incrementally onto an unfeatured surface.

If manipulation is to take place at lower, as well as higher, levels of refinement, the storage of control vertices with respect to a single origin is not suitable. When editing takes place at level (i) of surface refinement, in order to maintain continuity with the overlays, the control vertices of level $(i+1)$ and higher must be sensitive to changes brought about by manipulations of the $V^{(i)}$ control vertices. One method of achieving this is to represent the control vertices of any refinement level in "reference-plus-offset" form: $V^{(i)} = R^{(i)} + O^{(i)}$, where each control vertex may have its own reference, $R^{(i)}$, but this reference must be uniquely and dynamically derived from the next lower refinement level, $(i-1)$, of the surface. Editing may proceed at a given level of refinement by modifying the offset information. Editing at a lesser level of refinement changes the underlying surface, and thereby the reference information for the control vertices at all greater levels of refinement. Thus offset information accumulates editing changes local to an overlay, and reference information reflects editing change at all higher levels. We have designed a menu interface that facilitates "browsing" the data structure to select the refinement level used for manipulating the surface.

Rendering the composite surface for display requires descent to the leaves of the data structure so that the finest level of detail is presented. Rendering proceeds for any point on the surface using reference and offset information at the lowest level of the data structure containing that point.

The composite surface has a complicated structure of control vertices. It is far preferable to pick and manipulate points directly on the surface, rather than to confront the structure of levels of control vertices. The most fundamental relationship between surface and control vertices that can be provided relates each control vertex to the surface point over which it has maximal influence. Designated surface points, called *edit points*, can be chosen in one-to-one association with control vertices. The movement of such a surface point from its current location to a designated new position is the surface manipulation tool offered to the user. The desired edit-point move is implemented, transparently to the user, by a corresponding change in its associated control vertex. The formula for doing this is straightforward and generalizes to areas on the surface influenced by multiple control vertices.

Since only a restricted subset of the control vertices at level (i) can be moved while maintaining the integrity of the overlay with the underlying surface patches at the $(i-1)$ level of refinement, only that restricted subset is presented to the user for manipulation. This subset grows as more of the level $(i-1)$ surface is refined, and more level (i) control vertices become candidates for manipulation.

We present some examples taken from a prototype surface editor, which was written in C to run on a Silicon Graphics 2400 Turbo-IRIS workstation.

Local refinement and offset referencing provide a flexible and powerful new tool for the manipulation of B-spline surfaces during free-form surface editing. Local refinement controls the extent of any deformations and offset referencing allows localized edits to be retained over global changes to the B-spline surface. These methods have application in any modeling system that uses B-splines as the basis for object construction.

References

1. R. H. Bartels, J. C. Beatty, and B. A. Barsky, *An Introduction to Splines for Use in Computer Graphics and Geometric Modeling,* Morgan Kaufmann Publishers, Palo Alto, California (1987).

2. Elaine Cohen, Tom Lyche, and Richard Riesenfeld, Discrete B-splines and Subdivision Techniques in Computer-Aided Geometric Design and Computer Graphics, *Computer Graphics and Image Processing* **14**(2) pp. 87-111 (1980).

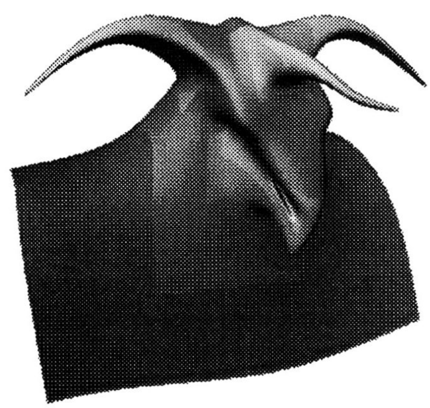

Figure 1: A B-spline patch with local refinements.