A NEW TECHNIQUE FOR SYSTEM-TO-SYSTEM TRANSFER OF SURFACE DATA

Michael W. Sterling and Michael E. Lucius

CADLINC INCORPORATED
Troy, Michigan

William J. Gordon
Department of Mathematics and Computer Science
Drexel University, Philadelphia, PA

INTRODUCTION

The desirability of transferring data between computer systems has long been recognized. However, due to the diversity of today's CAD/CAM systems, few systems can handle all the currently used mathematical models, nor can new mathematical models be integrated easily into existing software. In particular, a major problem is that of transferring surface geometry information from one system (mathematical model) into another.

Although IGES, for example, allows a number of surface types to be passed, many systems have surface models that do not conform to any IGES standard. Thus, the potential user of IGES is faced with a problem: Can a new surface type be added to IGES, and if so, would all the users of IGES support it?

The purpose of this paper is to describe a recently developed technique aimed at providing a universal interface between surface types. In brief, we have developed a software package which functions as a "common denominator" of CAD/CAM surface types. This software enables one to convert from any given surface representation to any other target representation.

RATIONALE

In most manufacturing environments, surfaces are produced by causing a machine tool to visit an ordered set of points. The surface thus produced is often finished by hand. The finished product may then undergo a quality control check which is designed to insure it matches the theoretical surface to within a specified tolerance. The TILE program mimics this process by producing a set of bicubic patches that matches the original surface in some respects and approximates the surface within a user-specified tolerance.

BASIC CRITERIA

The goal is to produce a set of bicubic patches which cover the target surface in such a fashion that the following criteria are met:

* The number of patches is close to the minimum number necessary

* The patches match the target surface to within some epsilon

* The tiles maintain the same slope continuity as the target surface

The patches produced by the algorithm can be of any type. In the TILE program, we use bicubic patches since they allow us to match point, slope, and twist vectors to the target surface. Thus, slopes can be continuous or discontinuous as they are on the target surface. The patches can be of lower order if desired. For example, if only point information is available, the patches produced will be bilinear; however, the number of patches required is likely to increase correspondingly. The patches can be of higher order although many systems will not accept patches of more than order four.

ALGORITHM

The TILE program assumes the following about the target surface:

* The target surface is defined (or is definable) as a mapping from a rectangle in UV (parameter) space to the surface in XYZ space

* The target surface is divided into a rectangular grid of patches with slope discontinuities occurring only along patch boundaries

* A surface evaluator routine is provided which, given a uv point as input, will compute the surface point, the tangents in the u and v directions, and the twist (mixed partial) vector

If tangent or twist information cannot be obtained, approximations may be computed. However, our experience suggests this will greatly increase the number of patches required to maintain fidelity.

The algorithm begins by attempting to fit a tiling patch to the first patch of the target surface. The target surface evaluator is called for each of the four corners of the patch. The patches generated by TILE all have a unit square as domain. Therefore, the length of the tangent and twist vectors returned by the evaluator must be scaled appropriately. The program assumes the evaluator computes proper lengths. This is not absolutely essential, since there are methods for estimating the correct lengths; however, this may result in an excessive number of patches.

Next the tiling patch is checked for closeness of fit. This is done by comparing the point values of selected points on the edges and interior of the tiling patch with corresponding points on the target surface. If any of these points fail to be within tolerance, then the tiling patch is rejected.

Whenever a tiling patch is rejected, its domain is split into two pieces and new tiling patches are generated and checked for fidelity. How the domain splits (i.e. in the u direction or the v direction) is best determined by examining how the selected points fared in the fidelity test. The domain space is cut along the parameter of worst fit.

The final result of the program is a rectangular grid of bicubic patches. The patches fit the target surface exactly at their corners. Also, the patch corners have the same tangent and twist vectors. Adjacent patches will have slope continuity, unless a discontinuity was indicated by the target surface.