



Illinois Wesleyan University
Digital Commons @ IWU

John Wesley Powell Student Research
Conference

1996, 7th Annual JWP Conference

Apr 13th, 1:30 PM - 2:30 PM

A Bezier-Based System for CAGD

Patrick Crowley
Illinois Wesleyan University

Lon Shapiro, Faculty Advisor
Illinois Wesleyan University

Follow this and additional works at: <https://digitalcommons.iwu.edu/jwprc>

Crowley, Patrick and Shapiro, Faculty Advisor, Lon, "A Bezier-Based System for CAGD" (1996). *John Wesley Powell Student Research Conference*. 30.
<https://digitalcommons.iwu.edu/jwprc/1996/posters/30>

This Event is protected by copyright and/or related rights. It has been brought to you by Digital Commons @ IWU with permission from the rights-holder(s). You are free to use this material in any way that is permitted by the copyright and related rights legislation that applies to your use. For other uses you need to obtain permission from the rights-holder(s) directly, unless additional rights are indicated by a Creative Commons license in the record and/ or on the work itself. This material has been accepted for inclusion by faculty at Illinois Wesleyan University. For more information, please contact digitalcommons@iwu.edu.

©Copyright is owned by the author of this document.

Poster Presentation 32

A BEZIER-BASED SYSTEM FOR CAGD

Patrick Crowley and Lon Shapiro*, Department of Computer Science, IWU

Bezier curves are the foundation of modern computer aided geometric design. A Bezier curve is a polynomial curve expressed in terms of Bernstein polynomials. A Bernstein polynomial is defined as

$$B_i^n(t) = \binom{n}{i} t^i (1-t)^{n-i}.$$

Much like lines and polygons require only end-points and vertices, a Bezier curve requires only a control polygon to completely define the curve (see example below, control points are indicated by X's). Bezier curves are invariant under affine transformations. In particular, affine invariance allows dilations and transformations to be applied to the control points only rather than all of the points on the curve. Since Bezier curves are defined as polynomial curves of arbitrary degree, they may be used to represent lines and other non-linear, higher-order polynomial curves. To this end, a design application has been created using Bezier curves to represent the graphics primitives. Points, lines, and higher-order curves may be designed and edited interactively. The project was developed for X-windows with the object-oriented language C++. The resulting application is a 2-d design package in which all data are represented as Bezier curves. The design application will be available for demonstration and evaluation.

