An Integrated Design Through Analysis Tool Using Rhino and Isogeometric Analysis

Matthew Getch, Alexis Moreno, Mariama Wilson

July 28, 2014

1 Abstract

Design and analysis are dependent on one another when creating complex geometries. The distinctions between the skills required for computer-aided design (CAD) and computer-aided engineering (CAE) create a gap between design engineers and analysts. The gap between CAD and CAE causes inefficiency and errors when trying to create analysis ready geometries from a design. Isogeometric analysis (IGA) spans the gap between both design and analysis on structures. This paper presents a solution for creating a NURBS based Rhino plug-in that will perform IGA on geometries without separating the use of CAD and CAE. The results from the plug-in will ensure that engineers have a way of designing and analyzing in a more precise manner without introducing geometric errors.

2 Introduction

Problem Area Engineers and analysts do not have the required tools to communicate in a beneficial way. The inconsistency of technological advancements in the fields of modeling and analysis requires an integrated design environment to enhance the outcome of the production process. Designers use computer-aided design (CAD) to create surfaces for production; these surfaces are constructed by manipulating some standard geometries. After an object has been designed it is important to perform stress analysis on the object. Traditionally, this is done through the use of finite element analysis (FEA) or computer-aided engineering (CAE) which creates a mesh of geometric shapes. To perform this analysis, the smooth curves of a design have to be approximated generating error. Once the analysis is done, if any changes are made to the design, the analysis needs to be redone. There are no options for analysis that can be done directly on the surface produced in CAD, the designer has to rely on finite element analysis.

Solution Previous studies have shown that there are two possible methods to bridge the gap between designing and analyzing, both of which include subdivision surfaces and B-Splines. Subdivision surfaces use polygonal shapes to make a mesh around an object; however, using this method causes a lower texture quality on a shape. B-Splines are similar to Bezier curves in that each point on a curve is defined by the blending of control points. Schmidt et al. study demonstrated that it is possible to integrate the design process between CAD and CAE. The approach requires an analysis-suitable generation that avoids trimming methods and utilizes NURBS or t-splines.

NURBS For this research, we will adopt the use of non-uniform rational B-splines (NURBS) into our research as introduced by Hughes et al (cite). NURBS are accurate representations of curves used to design freeform shapes. The advantages to use NURBS over subdivision surfaces, B-Splines or polygons, is that NURBS rely solely on the use of control points to edit or alter a shape. Designers are able to have complete control over the ending result of their desired object and can manipulate the components of a NURBS curve to produce the desired results. NURBS provides more precision in designing and faster way for modeling. In all, the main reason to use NURBS for this research is because it can be integrated with any software and has better transferring capabilities between designing platforms without a loss in data. Using NURBS also maintains uniformity amongst the control points enabling easier analysis.

Isogeometric Analysis To close the gap between these two fields, one solution that has been shown to significantly reduce error between CAD and CAE is the use of isogeometric analysis (IGA) on designs**(Schmidt et al)**. Isogeometric analysis relies on NURBS curves being the prime design tool. The advantage is that we can use isogeometric analysis directly on the produced curve. This reduces error significantly because it allows designers to work on the shape they produced directly.

Issues With IGA Today An issue with this potential tool is the lack of intuitive packages that address this technology; there are no packaged tools for an integrated design environment.

An Integrated Design Environment Here we attempted to integrate both CAD and CAE platforms by creating a plug-in using Rhino. This tool will allow designers to easily access IGA as a resource for NURBS surfaces they create.

3 Materials

Computer Statistics The plug-in created in Rhino which was designed on a 64-bit operating system Dell computer running Windows 7 with 12GB of RAM.

C# The research uses C# as the main programming language to design the NURBS based plug-in for Rhino. C# was chosen for its speed over the other languages supported by Rhino plug-ins. Reducing this overhead will enable faster performance for analysis and drafting. Using this programming language is beneficial because it can be compiled on a wide range of computer platform. C# also has support for strongly-typed functions in which compilers knows what a declared type is for every expression or variable. Rhino supports Visual Basic which was passed over in favor of C#.

Rhino Rhino is 3D modeling NURBS based software for producing curves and surfaces. This program allows for scripting and supports user development via a plug-in system. This program was chosen because it is the only CAD software that supports user developed plug-ins which is essential for creating the isogeometric analysis plug-in.

RhinoScript RhinoScript is being used as the developmental platform to create the IGA plug-in. RhinoScript is a programming language built atop Visual Basic that runs directly in Rhino. This language permits users to call other plug-ins inside Rhino which allows users to work with other facets of Rhino, in particular Grasshopper.

RhinoCommon RhinoCommon is a software development kit (SDK) which is a complete set of functions and objects that allow users to create an application within a program. Because of our reliance on Rhino we were also dependent upon the tools provided through RhinoCommon.

Grasshopper Grasshopper was chosen for its ease of use and ability to rapidly create workable geometry and is a visual programming plug-in for

Rhino. Using Grasshopper allows other members of the development team to engage in the development process in a non-trivial manner producing faster prototyping.

4 Methods

The Importance of Modeling All 3D models of objects and structures must be precisely designed using modeling software and analyzed using FEA. To understand the process of modeling, our first step in creating the prototype for the plug-in included designing a wind turbine using Grasshopper.

Why a Wind Turbine? Designing a wind turbine using the S809 airfoil geometry was the main goal because this structure includes all of the desired functionality of our plug-in. To create the turbine blade, the airfoil must be scaled, rotated, and translated. To create lofted geometry these three operations have to be available to the user.

First Procedure The first version of the code was written in Grasshopper and included numerous iterations of the same series of translations 26 times for each airfoil. The iterations included various rotations, chord lengths and radial distances. The coordinates for the S809 blade profile came from the National Renewable Energy Laboratory (NREL) (cite NREL). After the 26 iterations of each airfoil, we lofted the airfoils. This method of individually adjusting each airfoil to the NREL specifications, was tedious, inaccurate, and time consuming. After performing the task, it was shown that generating a wind turbine with numerous iterations would not be efficient for engineers when creating the plug-in interface since the user would have to manually adjust each value for each airfoil. There was also a lack of functionality for users since we were still essentially working in Rhino. (insert Alexis picture).

Second Iteration This procedure was suboptimal and was replaced with a more time efficient loop function. The new function constantly read information from a series of text files giving users the advantage of being able to modify the data files much faster. The individual text files were formed in Notepad for each important function in the airfoil. Each aspect of the turbine blade was given a text file with specific data points. These text files were then read into a series of translations. In this procedure we also centered each airfoil at the origin based off of another parameter in the NREL specification list. To further ensure a measurable result we split the airfoil and base then further divided those into upper and lower sections. (Getch picture and NREL turbine file or Prof Hsu data points)

Interface Design Subsequently, it was essential to design the interface after building this Grasshopper code. Ideally, the user would be able to upload a series of text files with the data points for their cross sections and one for radial, twist and scale values, loft the cross sections to create a 3D object and then perform the analysis on the geometry. All of this should be packaged so that it can be done in just a few button clicks.

5 Results

So far it's possible to click a few buttons and loft a particular geometry. The interface currently functions, but still feels a bit clunky.

6 Future Work

After creating the first version of the NURBS-based Rhino plug-in, many steps have to be taken to guarantee maximum user experience of the interface.

GUI Development It is essential to continue GUI development on the plug-in; this includes updating the design layout and implementing interface changes based on user-feedback. We would also like to increase accessibility options for those who have impaired vision.

Analysis Options Users should be able to choose what type of loft they would like, how many control knots to use, and how much tolerance there should be. We would also like to offer t-spline functionality at some point for improved analysis.

Browser Based IGA It's possible to run a version of Grasshopper on a server without the visual interface. This would allow users to upload files for analysis via their browser allowing IGA to happen anytime.

User Testing Ultimately, after various updates to the plug-in, we will conduct student user studies to provide feedback on the functionality of the plug-in; user flexibility and overall functionality.