HEAVY MOVABLE STRUCTURES, INC.
SEVENTEENTH BIENNIAL SYMPOSIUM

October 22-25, 2018

Solid Modeling: the What, the How, the Why

Reid Brooks
Steward Machine

MARRIOTT'S RENAISSANCE HOTEL AT SEAWORLD
ORLANDO, FLORIDA

INTRODUCTION

Solid modeling is becoming increasingly prevalent in all aspects of engineering. With increased computing capabilities and more ubiquitous access to the software and hardware required to make use of solid models, this tool is becoming increasingly integral to the execution of engineering projects. As this tool extends into more and more aspects of the heavy movable industry, this paper strives to provide brief answers for three questions regarding solid modeling: the what, the how, and the why. Starting with a general overview of what solid modeling is, the paper then spends the bulk of its effort giving a general overview of the architecture and principals of how solid modeling is performed. The paper concludes with an overview of uses for solid models, both currently in practice and projecting into the coming years. This paper seeks to deliver useful knowledge about the field of solid modeling, and to provide an overview of its execution and uses.

THE WHAT

It is tempting to think of solid modeling as 3D drafting. This is a helpful tool for thinking about one aspect of solid modeling, drawing creation. However, this paradigm does not quite capture the entirety of the tool. While it does lend one to think of the definition and communication of an object’s geometry, it does not entirely capture solid modeling’s emphasis on fidelity to the physical properties of an object. Solid Modeling is the representation of physical objects in a virtual three dimensional space with inputs given to all aspects of that object’s definition. While this includes the object’s geometry, it extends to other physical attributes such as density, volume, weight, center of gravity, moment of inertia, and surface area. It can include properties such as an object’s coefficient of thermal expansion, its specific heat, and its thermal conductivity. It can include an object’s yield strength, its tensile and shear strength, its Poisson’s ratio, its shear modulus, and its modulus of elasticity. This even extends to an object’s appearance to include color, texture, reflectivity, and transparency. This can also include other intrinsic information to a model such as its part number, manufacturer, and cost. Solid models strive to represent objects with as much accuracy as they can to show how they will interact with the real world. So what is solid modeling? Solid modeling is the creation of three dimensional objects in a digital environment using mathematical principals and computer modeling to ensure a high degree of fidelity to physical properties.

THE HOW

It is not really possible to think of solid modeling without considering the software that is used to create solid models. While the software do not exactly define what solid modeling is, they are currently the tools used to create solid models. So when thinking of the “how” of solid modeling, there are a few software packages to keep in mind that have become the dominant market players. These are AutoDesk’s suite of software (Inventor and AutoCad), Dassault Systemes’ platforms (SolidWorks and Catia), PTC’s Creo (formerly ProEngineer), and Bentley’s Microstation. While there are many other solid modeling platforms, these make up the core of the professional engineering software. These programs all have similar architecture when it comes to creating and using solid models, and they typify the software that this paper explores.

The first prompt received when solid modeling is to create a file. There are three types of files that can be made. The first two file types are part files and assembly files. These files are what are considered to be creating solid models. The last file type is a drawing file. This file type speaks more to the pedigree of these software as engineering software, and has historically been the end goal and deliverable for solid modeling. A part file is the most basic element of solid modeling. These are the first files that need to be created when solid modeling. Part files use sketches and operations to define and create an object's geometry. Part files use material property assignments to provide an object's physical definition. Part files can be used in bulk to create assembly files. Assembly files are made up of several part and sub-assembly files. They are directories of these files, and use constraints to define each part's relationship to the other parts in that file. The constraints placed on each part or sub-assembly define their location and orientation within the assembly. Drawing files come in two varieties, although they use the same file type for each. Drawing files can be used to create either part drawings, or assembly drawings. These provide a similar product to 2D CAD and hand drawn drawings, but are permanently linked to a solid model that drives the drawing's generation.

Creating Parts:

One of the largest challenges to solid modeling is how to break down an object's definition into simple geometries that can be fully defined. This is done by providing tools to define geometry step-by-step. There are three general types of operations that can be used to create geometry. The first step must be to create a volume. These operations will provide the base shapes that can be further defined into more complex geometry. They include extrusion and revolution operations. The second category of operations are volume modifying operations. These use existing volume to create other features such as fillets, shells, and holes. The third category are volume iterating operations such as pattern, and mirror. These use existing geometry to create iterations in a manner defined by the user.

Creating Volume using Extrusion and Revolve:

Extrusion and revolve are two of the basic volume creation tools. These are the first steps to creating a solid model. Both of these operations break a 3D object's definition down into elements that can be fully defined. For an extrusion, this involves a 2D sketch that will represent the object's cross sectional area, and an extrude feature that will define the direction and length of the extrusion. For a revolve feature, this will similarly be a 2D sketch that will represent the object's cross sectional area throughout the revolution, and a revolve feature that will define the direction and the angle of the revolution. By using a 2D sketch, the user can fully draw and define the cross section.

Modifying Volume using Chamfer/Fillet, Shell, and Hole:

Volume modifying operations require an existing geometry to change. For the most part, these operations do not require a sketch to define the operation, but rely on the existing volume's geometry and user-input data for definition. Chamfer and fillet take edges or parts, and add either a chamfer or a fillet. They only require definition of the operation (the size of the chamfer or fillet) and the selection of a location on the existing geometry on which to create the chamfer or fillet. Shell removes internal volume from a geometry, and requires only the input of how thick of a wall to create, and the option to remove surfaces to create an open volume. Hole requires a location of the hole, and the definition of the hole. Holes can be created to be counter-bored, countersunk, threaded, have sub-drill definitions, etc. The definition of the hole is entirely defined in the operation.

Iterating Volume using Pattern and Mirror:

Volume iterating operations rely on existing geometry to re-generate copies of that geometry in a specified manner. Patterns can be used to create multiple instances of a geometry in specified patterns. These commonly include linear and circular generations, but can also include table driven locations, fill operations, and more complex patterns where geometry is modified with each generation. Mirror, often considered a sub-set of pattern, creates a copy of the existing geometry across a reference plane. While no sketch is needed, and the mirror plane is often part of the existing geometry, mirror does differ from the standard pattern operations in that it does need a reference geometry to execute. Mirrored and patterned geometries are iterations of existing geometry, so one of the key elements in their definition is a user-input decision as to whether or not the iterations will have their definition tied back to the original geometry. That is, if the original geometry changes, will the iterated geometries change as well, or will they be independent of the original generation.

Most of the definition of a part file will come through defining the part's geometry. However, as previously stated, a part file contains more information than just geometry. In this file, the user can also define material properties and appearance, as well as other properties of the part. Materials can typically be selected from a catalog of pre-defined material properties. Often, these properties come with associated appearances. However, the user also has the option to change the appearance of the part. These options often are available from a catalog of standard appearances for various materials and finishes. The user will also have the option to define color, transparency, reflectivity, texture, etc.

Creating Assemblies:

Assemblies consist of multiple parts or sub-assemblies brought into one space and constrained so that their locations and orientations are fully defined. The first thing needed to create an assembly are part files or other assembly files. While there is no sub-assembly file type, an assembly is considered and treated as a sub-assembly when it is being used in another assembly. It is important to remember that the files for the parts and sub-assemblies that you use in your assembly are not native to your assembly file, but are entities of their own. The assembly file simply compiles a directory of file locations so that it can search for, find, and bring those files into the assembly file. This is a powerful feature of solid modeling software that allows the user to change definitions of a part in its native location and have those changes propagate throughout all of the instances where it is used. However, it also poses a hazard that the user must pay attention to. That is, the file location and file name are required to remain static in order for the directory to be re-created. If the location or the name of the part file in the computer's directory is changed, the assembly file will have no way of finding that file, and the software will not be able to regenerate the assembly. Ensure that this is kept in mind when creating and manipulating files and directories to avoid having to rectify file locations. Many software packages offer file management software to help mitigate this problem.

Once files are brought into an assembly model space, the user can begin to use constraints to define the parts' locations and orientations in relation to one another as well as in relation to the global origin of the model space. Constraints, also called relations, use the parts' geometries and origins to establish relationships between other parts. These include: mating surfaces, lines, and points; establishing set angles between planes and lines; defining concentricity of spherical and cylindrical features; and establishing offsets between entities. Using the available constraints, the user defines an object's location and orientation until all six degrees of freedom of movement are defined.

Creating Drawings:

The biggest difference between 2D CAD packages and 3D solid modeling packages when it comes to creating drawings is that the user does not need to do any drawing of the representation of the part or assembly when using solid modeling software. With solid modeling, the drawings are based on the model of the part or assembly. The drawing uses the model to place the object lines, hidden lines, section views, detail views, etc. of the part. The user controls which views to place, where to place them, the scale, etc. The other big difference is that the user does not define any dimensions in the drawing file. The dimensions are selected and placed, but the numbers pull from the model. This is also true for other details like hole notes, chamfers, fillets, etc. Once defined in the model, the user simply has to select the feature that they wish to annotate, and place the annotation. The user has options to place other drawing features such as a revision table, bill of materials, leader balloons, etc. All of these will also be driven by the model. The drawing file keeps track of what revision the user is on, and will change all revision notes and symbols being placed to reflect the current status. The bill of materials intelligently updates with the parts on that drawing and adjusts the quantity. It can also pull other information defined in the part file such as part number, description, and material. The leader balloons intelligently follow which part representations are associated with each view, and will track to be consistent with the bill of materials. While these have to be added and placed by the user, the software tracks changes and propagates those changes throughout the drawing set. For instance, if the dimension of a part changes in the part file, all of the drawings (assembly or detail drawings) for that part will change both their visual representation of that part and the dimensions associated with the change. Styles such as drawing templates, default notations, and information in the bill of material can be set by the user, and made standard for specific drawing templates.

The Why

Currently, the main deliverable from solid modeling is a set of 2D engineering drawings. So why use 3D modeling over 2D drawing to create engineering drawings? There are both advantages for creating drawing sets as well as opportunities to use the 3D models. For the drawing set, the biggest advantage may be model fidelity. While a drawing set may have a part that is represented in several assemblies or used in many instances (and almost certainly shown in several views), with 3D modeling there will always be a single model that drives all of those drawings and representations. When making changes to that part or assembly, those modifications will propagate through the entire drawing set and change every drawing where that model is represented. This ensures that there will not be conflicting information about a part in the drawing set. This saves time by keeping the user from having to go through each file and changing the drawings, and also significantly reduces the risk of human error. This goes not only for part representations, but is also true for other drawing features such as the bill of materials. When quantities and parts of an assembly change, the bill of materials will be updated to reflect the change, reducing errors between conflicting bills of material and drawings. Another advantage to using 3D modeling is detailing time. Once one is proficient at modeling, overall detailing time can be significantly reduced. The user is not responsible for creating each view of the part, dimensions need to simply be placed, and notations are standardized and selected. Another advantage is the ease of placing detail, section, and isometric views. While non-orthographic projection views were difficult to draw using 2D CAD software and hand drawn details, the solid modeling software can easily represent any view that the user would like. This gives the user the option to add clarity to the drawings by placing non-standard views.

The other advantages to solid modeling come from the creation of an interim product when making drawings, the 3D models themselves. With advances in technology, more and more uses for solid models are being made available. These can help with many aspects of engineering, project management, and design. In addition, many of these features are being built into solid modeling software and offered as standard parts of the packages. There are several different programs that provide computer based analysis for different loading conditions. Finite element analysis allows the user to input support constraints, and apply loads to a solid model. The program outputs both internal stress analysis for the part as well as deflection analysis. Computational fluid dynamics programs allow the user to set boundary conditions for a par, and analyze the resultant forces due to fluid flow over the surface of a model. Thermal analysis prompts the user to input a set of heat inputs, and analyses heat flow through the part.

There are also ways to use the 3D modeling program to gain useful information about parts. For instance, given a density or densities, the program will calculate several physical properties for the part or assembly. These include surface area, volume, weight, center of gravity, and moments of inertia around different native and user-defined axis. These properties can be very cumbersome to evaluate conventionally, especially for complex geometries, making solid models very useful tools for this application. For instance, surface area can be used to quickly calculate coating square footage. Center of gravity and weight can be used for shipping, designing picks, and for balance calculations. Weight can also be used to bulk price material for estimates. Solid modeling software also includes measurement tools that can be useful to measure and visualize complicated geometries. They also include interference analysis tools that will identify and highlight any interferences in assembled parts. These are great tools for collaboration of parts for larger projects, and can be used to determine such things as working envelopes, clearances for tools, personnel, equipment, installation clearances, and can be used to design picks and placements. They can also be used visually to get an idea of how things will fit together before getting them into the field. Solid modeling software has a few ubiquitous “shareable” file formats that make collaboration and design easier as well. Models can be shared and analyzed to ensure coordination between products. This also makes it easy to share and keep catalogs of standard products from vendors. There are vast libraries of parts on the internet that can be searched for and imported into the model space, saving modeling time and giving the user a more complete understanding of the product. On top of this, solid modeling packages often come with a library of commonly used parts such as nuts and bolts that conform to industry standards. All of these tools can help accelerate design and make collaboration a more thorough process.

The last benefits of solid modeling are in manufacturing. Advances in manufacturing technology are becoming more intertwined with solid modeling to give users tools to more directly go from solid models to manufactured parts. This comes in the form of both traditional and additive manufacturing. Currently, many shops use 2D drawings to write programs for machines that manufacture parts. Those programs are becoming more and more able to use 3D models to assist with the creation of specific codes. There are also add on programs as well as 3D software modules that allow G code to be created natively to the solid model. These programs use 3D modeling to generate the raw material, and compare it to the 3D model of the desired finished part. The software allows the users to select tools, show specific setups, and design tool paths for each operation. Similarly, 3D models can be used for additive manufacturing. Because this process does not require material with stock to manufacture the finished product, the solid model can be used on its own to designate a tool path to create the part. This can then be sent directly to a 3D printer for manufacture.

Inspection is another area of manufacture where 3D models can be used to benefit production. In the above model of manufacturing, the 2D drawing can be eliminated entirely. The 3D solid model would be submitted for approval, reviewed, and once certified, would be the standard for the finished part. In this vein, the final product would not be inspected and certified to a printed drawing, but would be digitally scanned and modeled using digital inspection tools as an “as-built” 3D model. This model would then be digitally compared to a 3D model with tolerances on all of its dimensions and shown to be acceptable, or out of conformance. Inspection reports would visually be able to show final dimensions of parts, and any geometry could be virtually inspected, measured, and analyzed.

Conclusion

The barriers to 3D solid modeling have in the past been largely hardware issues. However, as computing power becomes more available and less expensive, solid modeling will continue to become more and more ubiquitous in all aspects of engineering, design, and manufacturing. While there are several realities on the horizon that are not quite within reach, the prospects of solid modeling seem promising and the technologies that are available right now can be used very effectively. From creating 2D engineering drawings, to inspection, to design and analysis, 3D solid modeling is a beneficial technology that is accessible and both usable for current practices, and promising for future ones. In many ways, what solid modeling is is the future of design. How we apply it currently makes it a compelling tool for use right now, but its most promising uses are yet to come.