

Dynamic Modeling with Constraint-based CAD in Introductory Engineering Graphics

Eric N. Wiebe, Ted J. Branoff, and Nathan W. Hartman
NC State University, Raleigh, NC

ABSTRACT: This presentation is part of an ongoing research project by the authors looking at how constraint-based 3D modeling can be used as a vehicle for rethinking instructional approaches to engineering design graphics. A particular goal is moving from a mode of instruction based on the crafting by students and assessment by instructors of static 2D drawings and 3D models. Instead, an approach based on dynamic modeling is proposed. This approach provides the opportunities for students to learn about geometry through the embedding of geometric behaviors into models and then testing these behaviors via dynamic change of the model. An advantage to dynamic modeling is the ability to provide visual embodiment to the underlying logic of the constraint strategy, providing direct feedback to the student and a straightforward mechanism for assessment by the instructor. Also, this approach mimics the real-world activities of engineers and designers executing 'change orders' on their virtual product models. As such, this approach is better aligned with larger goals of better problem-solving abilities in the technology and engineering workforce. A previously developed taxonomy of geometric behaviors is used to guide the development of activities that provide an appropriate range of activities for the students. Example instructional activities will be presented..

I. Introduction

Over the last two years, the faculty in the Graphic Communications program at North Carolina State University has been integrating constraint-based solid modeling into their introductory engineering and technical graphics courses¹. While 3D constraint-based modeling has been in use in industry for more than ten years, it has only been within the last few years that there has been widespread adoption of these modeling tools at colleges and universities. 3D modeling tools have been identified as a key component of some areas of engineering education for a while², but it has only been recently that educators have begun to investigate how the newer constraint-based modeling tools bring a new set of possibilities to instruction³. Even more recently, researchers have looked more holistically at how these constraint-based tools can be effectively integrated into a modern engineering graphics curriculum^{1, 4, 5}. In looking at the curricular implications of these tools, it becomes important to identify what are the core concepts and abilities we intend for students to gain from an introductory course in engineering graphics.

The knowledge and abilities gained have to be relevant both in the short term for further courses they will be taking in school and in the long term as life-long learners and productive workers in industry⁶. A particular challenge is to strike a balance between narrowly defined competencies with particular software tools and larger 'big picture' conceptual knowledge about what drives this whole class of constraint-based modeling tools.^{7, 8} has previously outlined the similarities in the underlying functionality between the most popular constraint-based tools and their relatively accessible user interfaces. This trend continues to this day and has largely removed the need to focus large portions of an engineering graphics course to training on specific software packages

in order to meet industry needs. Working under this assumption, this article will instead focus on one of the key conceptual areas of modeling that should be emphasized in any modern engineering graphics curriculum: geometry. That is, a primary instructional goal should be the development of students' understanding of Euclidean and topological properties of planar and solid geometry using constraint-based 3D modeling tools.

Applied geometry has been part of engineering design graphics since its inception. Being applied, however, means that how it is integrated into the curriculum becomes dependent on what tools are being employed and what the perceived needs are for the students in their future education and employment. The introduction of constraint-based modeling into the curriculum means that, once again, there needs to be extensive discussion of how geometry should be applied in this evolving curriculum⁹. This paper will outline how applied geometry can be integrated into a modern engineering graphics curriculum. This will include the use of geometric relations within a 3D solid model representing a virtual, engineered product. Of particular interest is the definition of the size and location of a 'feature' in a solid model relative to its constituent features or a frame of reference. These goals are addressed within a framework of a student identifying key geometric features within the virtual product and how this geometry needs to be controlled in order for the model to be of maximum usefulness in the product development process.

II. Dynamic modeling

Engineering graphics has, in many ways, struggled with the same issues that technology education has over the past 40 years. That is, how to move beyond teaching 'craft' with the end all and be all goal of creating a 'product', to teaching 'process', where how you get to the finished stage is more important than the product itself¹⁰. In other words, the final product and its precursor stages become artifacts that represent the problem-solving process of an applied geometric problem. It is, however, a challenge to come up with modeling problems where a 'good solution' can be assessed by measures other than geometric accuracy¹¹.

Historically, engineering graphics artifacts turned in as part of coursework are assessed by visual inspection. This could mean reading dimensions on a multiview drawing or it could be judging the length, angle, or higher order geometric relations of model elements. Another alternative is electronic measuring of geometry within the computer model. This still does not get past an assessment of the model solely based on the final artifact.

An alternative to visual inspection of hard copy printouts or passive inspection on the computer monitor is active probing of the actual computer model. While complete inspection of how the entire model has been constructed and constrained can be very time consuming, focused probing of key features is more effective. Often this probing takes the form of changes of key dimensional values, revealing how the underlying construction of the model responds to these changes. This use of 'dynamic modeling' shows promise as a strategy when the goal of assessment goes beyond geometric accuracy of the final model¹.

Dynamic models as a strategic instructional goal addresses a number of key issues. First, it allows for a broader assessment of the modeling problem that encompasses both 'solution' and 'accuracy'. The modeling design process that leads to the solution is embedded in the model by

the way the student chooses to construct their model. These construction decisions are revealed to the instructor through dynamic modeling activities. Second, these dynamic modeling activities address the important issue of having instructional activities reflect authentic professional practice. In this case, dynamic modeling can come in the form of 'engineering change orders'.

The worth of a model now is based not on a single geometric form, but how well it can be iterated through a number of configurations as it moves through the simulated engineering design process. Students now experience the complete cycle of model definition, comparison to design goals, and modification based on needed changes. The ease of making these geometric modifications are a primary driving force for industry to adopt these constraint-based modeling tools, however, many companies will readily admit that their engineers and designers are often ill-equipped to build robust models that truly capture the geometric design constraints¹².

Successful solutions to modeling problems now hinge on addressing three key elements. First, the solution will depend on correct initial geometry; this is usually as far as most assignments go. Second, it will depend on embedding the correct behaviors via geometric constraint definition. Finally, it will depend on correct translation of the initial design constraints and engineering change orders into geometric definitions.

This approach, however, is not without challenges. First, can a taxonomy be developed that identifies the core geometric behaviors that all students should master? Second, given the inherently open-ended nature of most design problems, can a robust enough set of activities be developed that allows for assessment within a practical amount of time and with a high degree of reliability? That is, can modeling problems be designed that have students incorporate these behaviors in a systematic and controlled fashion. The remainder of the paper will address this first challenge while indirectly pointing to solutions to the second.

III. Embedding geometric behavior

As mentioned previously, there is a high degree of uniformity of interfaces and modeling tools across the most popular constraint-based modelers used in instructional settings⁷. This uniformity allows for the definition of a generic model creation process that is applicable in a wide range of instructional settings. This modeling process is outlined in detail in¹³ and will be used as a basis for the discussion here. Also of note, that while the complete virtual product definition most always involves the creation of assemblies, instructional constraints usually limit an introductory course to focus on single part modeling. Still, all part modeling should be done in the context of knowing that these parts will eventually be part of a larger assembly.

All model behavior is ultimately based on how the model is constructed. That is, it is based both on the geometry of the model and on how this geometry is constrained relative to other geometry or to itself. Behavior reveals itself through how the model changes form when one or more of its constraints is modified. For example, if the length dimension of a bar is doubled, then how the holes in the bar change (or don't change) position is based on the behavior embedded in the model. Model behavior is central to all dynamic modeling activities and correct behavior should be explicitly defined by the modeling problem. It follows that correct (and incorrect) model behavior should be clearly identifiable by the student and the instructor assessing the modeling

problem. Which behaviors are central to an introductory engineering graphics course should be identified and integrated in modeling activities.

These behaviors appropriate to an introductory course might be called 'basic behaviors'. Most all of these basic behaviors will be based on a single constraint modification that impacts one or two dimensional constraints or geometric relations. Advance behaviors -- those that are based on multiple modifications, impact multiple constraints, or involve equation-driven constraints -- are most likely going to be confined to an advanced course.

Basic behaviors based on geometric relational constraints that students should be exposed to in an introductory course include:

- Parallelism
- perpendicularity
- horizontal and vertical (based on a global coordinate system)
- symmetry
- same size
- colinearity
- tangency
- alignment/fixing to an existing entity

Dimensional constraints that control behavior students should have experience with include:

- horizontal, vertical, angular, and aligned control of linear location
- size of linear elements
- circular/angular size of radial elements
- location of reference geometry
- internal to sketch (typically size)
- external to existing geometry (typically location)

In order to assure uniformity in behavior in the model, models should be fully constrained. That is, the geometry of the model should be fully defined via a combination of dimensional and geometric relational constraints. Allowing geometry to 'float' gives way to the potential for unpredictable behavior, impacting the ease in which the dynamic modeling assignment can be assessed. Unexpected behavior, in fact, should be a flag to the student and instructor that the student has not fully thought through how this model is to behave and how to properly constrain it so that it exhibits these behaviors under the prescribed conditions. The flip side of this is that modeling problems need to be designed so that they readily reveal common problems with embedding behavior via constraints.

IV. Examples

At NC State, the switch was made from AutoCAD to SolidWorks in the Spring 2001 semester. The faculty also removed all instrument drawing in the introductory courses. This transitional period was not easy. One of the biggest challenges was designing new exercises that were appropriate for the new courses. The previous courses were heavily focused on documentation – specifically detail drawings. Although several laboratory exercises involved solid modeling in AutoCAD, most CAD activities involved 2D drawing concepts. The exercises presented here are

designed to utilize the power of the software while introducing students to the concepts presented earlier in the paper.

Symmetry

When modeling objects, experienced individuals typically recognize lines of symmetry immediately. Modeling a part with a datum or primary construction plane through the line of symmetry allows the user to mirror features across the plane of symmetry.

One of the first models students are asked to create in the introductory classes is the ROD GUIDE (Figure 1). Students are asked to create the initial sketch of the ROD GUIDE by constructing half of the first profile and then mirroring it about the line of symmetry (Figure 2). After adding dimensions and relations to fully constrain the sketch, students are then asked to modify the 3.000 dimension (distance between the top arcs) to check the symmetry relation.

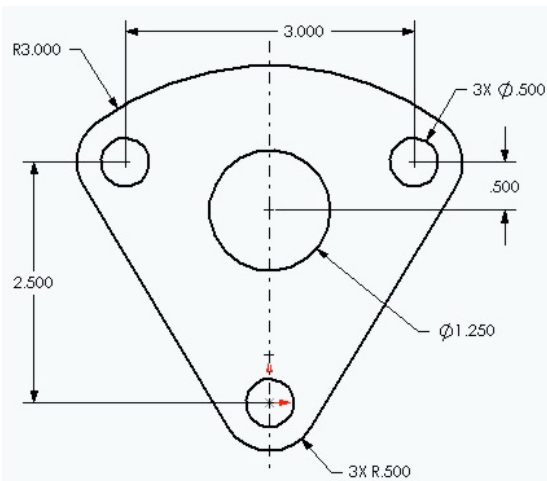


Figure 1. ROD GUIDE.

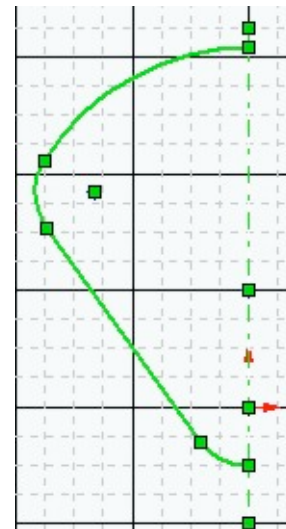


Figure 2. ROD GUIDE Line of Symmetry.

As is the case in most constraint-based CAD tools, relations (or constraints) often take the place of explicit dimensions when they are added to a Sketch. In some scenarios, this captures the intent of the designer, but manufacturers still require explicit dimensions in order to make the part. In this exercise, the students are asked to model the GUIDE part (Figure 3) in such a fashion that it is symmetric in Width as well as Depth. They are also asked to make changes to the model to test the effects of symmetric and non-symmetric conditions.

The objectives for the ROD GUIDE and GUIDE exercises are to:

- Create a solid model of the GUIDE that captures the symmetric intent of the designer.
- Revise the model by changing specific dimensions.
- Break/delete the symmetric relations in order to alter the design intent of the model.

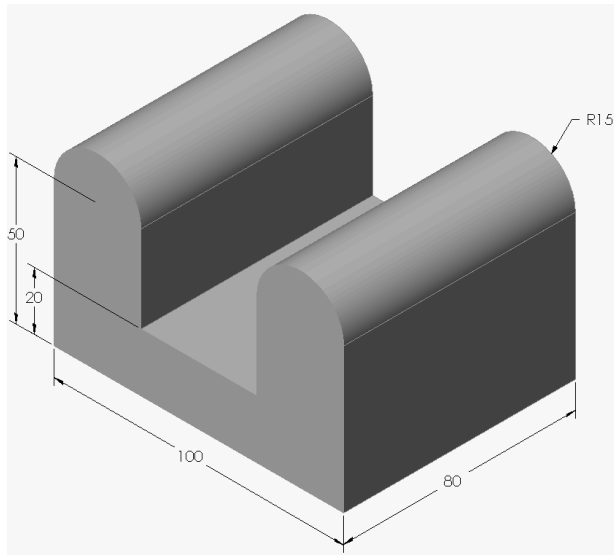


Figure 3. The GUIDE.

■

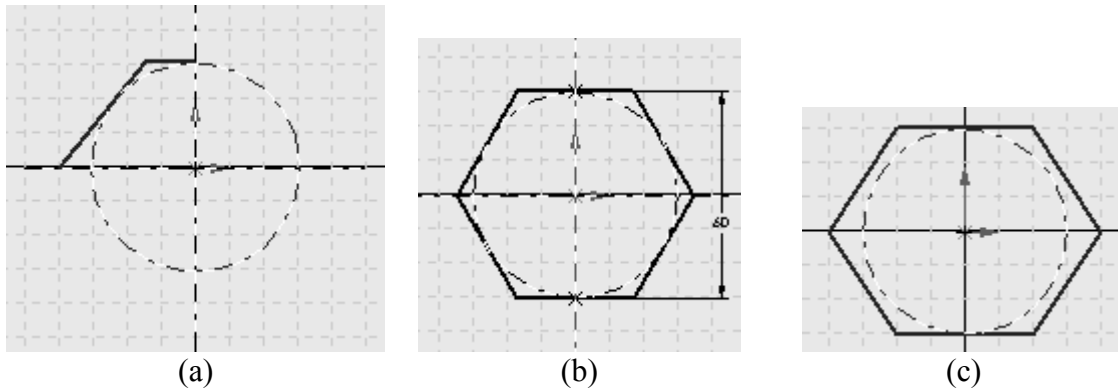


Figure 4. Regular Polygons.

Regular Polygons

Creation of a regular polygon from scratch in a profile provides an opportunity to explore symmetry, similar length, and similar angle constraints. Exercises can be designed that explore the differing solutions (e.g., that vary the number of dimensional constraints versus the number of geometric relations) to fully constrain the polygon. Also, the types and number of constraints needed to fully constrain can vary based on the number of sides and can also vary based on behaviors the polygon needs to exhibit under modification. Figure 4(a) illustrates starting with a construction circle and two centerlines. After adding tangent constraints between the circle and the two lines, the lines are mirrored about the centerlines and the necessary constraints are added to fully define the polygon as shown in example (b). Another approach, shown in example (c), is to require only a construction circle and ask the students to determine the number of constraints

required to fully define the polygon. This activity provides an introduction to construction geometry other than sketch planes and the role it plays in structuring final geometry.

The regular polygon exercise can be done with the polygon being swept out to form one feature of a multi-feature part representing a standard fastener – a BOLT. Also, an assembly could be designed requiring a hexagonal feature that was not based on a regular hexagon. In this case, a student would need to more selectively establish a combinations of equal size, equal angle, or symmetry constraints. Additional challenge can be added by requiring the hexagonal shape to be driven by dimensional constraints located on specific faces. The objectives for this exercise are to:

- create a solid model of the BOLT including constraints that would allow for specific changes in the model.
- use a circle as construction geometry to assist with constraining the hexagon feature.
- explore alternate methods for constraining the hexagon feature.
- compare their constraint method to the constraints automatically placed with the regular polygon tool.

Tangency – Breaking and redefining constraints

One of the powerful features of constraint-based programs is the ability to redefine the relations or constraints. For this exercise, students are asked to model the STOP BASE (Figure 5) and then make some modifications to the tangent and dimensional constraints on the model. The objectives for this exercise are to:

- create a solid model of the STOP BASE including constraints that would allow for specific changes in the model.
- revise the STOP BASE by changing the width of the part.
- break/delete a tangent relation in order to create a completely different part.

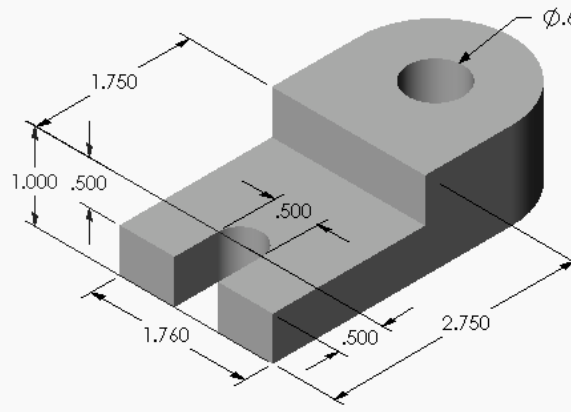


Figure 5. The STOP BASE.

Students are given a step-by-step tutorial of the modeling procedure for the STOP BASE and an additional handout. The handout includes the objectives for the exercise as well as some screen captures. For the initial solid model, students are shown the sketch for the first extrusion (Figure 6). They are told to add tangent relations between the vertical lines and the arc and to dimension

the vertical distance by selecting the bottom line and the arc. This allows them to change the properties of the 2.750 dimension later in the exercise.

The second part of the assignment is to modify the 1.760 width of the part to 2.500. The purpose of this step is to verify the tangent relations between the vertical lines and the arc in the first sketch.

The final part of this assignment is to revise the initial sketch of the STOP BASE per Figure 7. Students are asked to use the Display/Delete Relations command to delete the tangent relations between the vertical lines and the arc. They also have to change the properties of the 2.750 dimension so that the arc condition is Max instead of Center. The end result is seen in Figure 8.

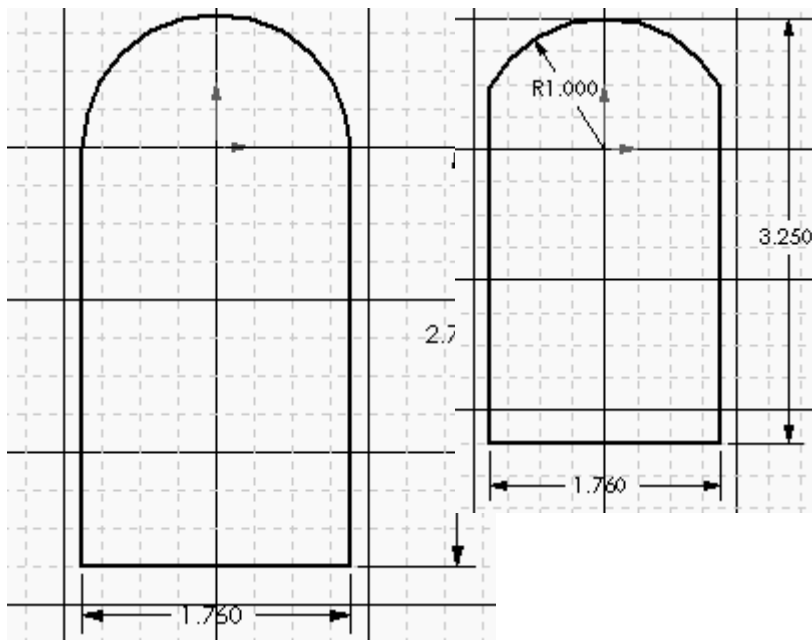


Figure 6. First Sketch for STOP BASE.

Figure 7. Part 3 of the STOP BASE.

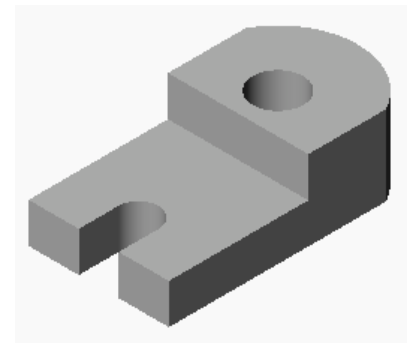


Figure 8. Completed STOP BASE.

Examining the number of constraints present

Examining the number of constraints present in a model gives some insight as to the design intent of the individual that created the model. As mentioned previously, relations (or constraints) typically take the place of explicit dimensions, and in doing so, reduce the number of driving dimensions contained in each sketch. While these may not be the dimensions shown on a drawing for manufacturing, they are the dimensions that work best with the desired geometric relations to drive the intended behavior.

Generally, there is no particular right or wrong way to constrain a sketch. However, it is important that the user understands the implications for adding certain constraint combinations and has the ability to determine when the sketch is fully constrained. By capturing specific combinations of constraints, the student can incorporate desired behaviors into the model. In these exercises, students are given sketched geometry and asked to dimension the geometry and label geometric relationships in such a fashion as to make the sketch fully constrained. The objectives for this exercise are to:

- Develop a level of familiarity with applying geometric relations.
- Recognize when a sketch is fully constrained according to design intent.
- Interpret the effects of specific combinations of dimensions and geometric relations.

Figures 9 and 10 show the sketches that the students are asked to label and dimension. The students are instructed to draw dimensions on the profiles and to list the geometric relations being applied as well as the entities to which they were applied.

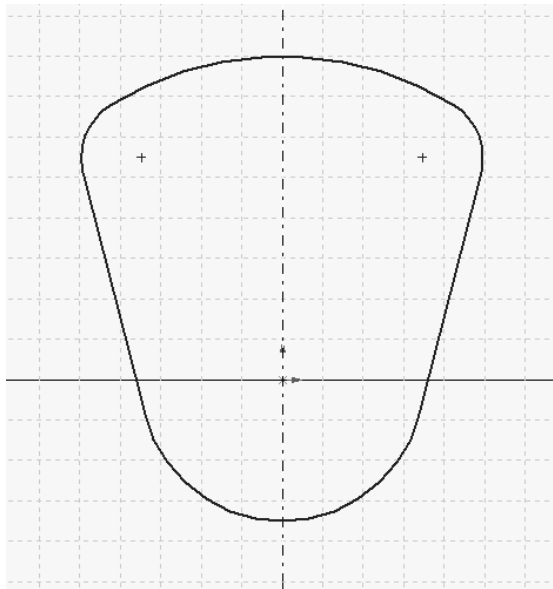


Figure 9. Sketch for Labeling Constraints.

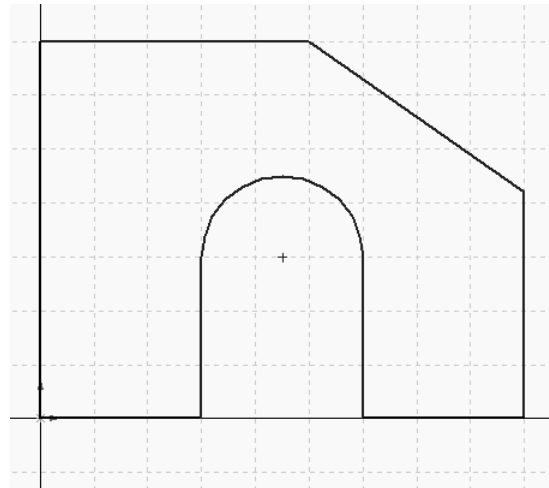


Figure 10. Sketch for Labeling Constraints.

As the students begin to apply dimensions and relations to the sketches, they have to consider how they want the geometry to behave and which dimensions they consider important for driving that behavior. Figures 11 and 12 show examples of the results of this exercise.

In addition to the dimensioning scheme shown in Figure 11, several specific geometric relations are added to the sketch:

- A Collinear relation between the two horizontal lines at the bottom of the sketch.
- An Equal relation between the two horizontal lines at the bottom of the sketch.
- Tangency relations between the arc and the two adjacent vertical lines.

Notice that there is no radius dimension given for the arc; it is controlled by the Equal relation assigned to the two horizontal lines. If the design intent calls for the radius of the arc or the

length of the second horizontal line at the bottom to be given, then the Equal relation has to be removed.

In addition to the dimensioning scheme shown in Figure 12, several specific geometric relations are added to the sketch:

- Tangency relations between the three arcs at the top of the figure.
- Tangency relations between the line segments and the adjacent arcs at each end.
- An Equal relation between the two smaller arcs.
- Symmetric relations between the center points of the two smaller arcs and between the line segments.

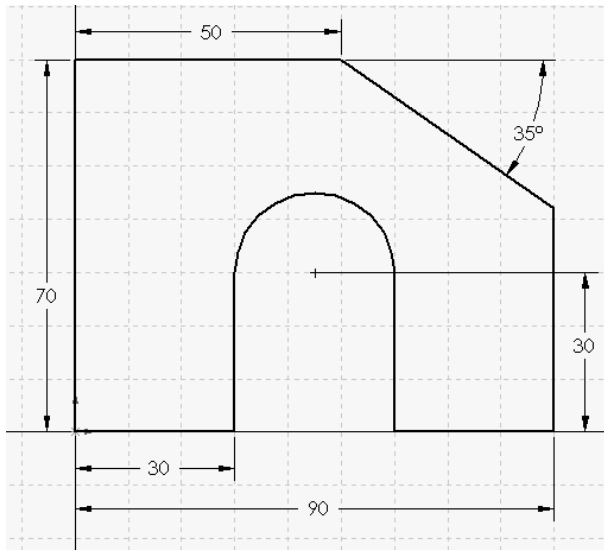


Figure 11. Possible Dimensioning Scheme for this Geometry.

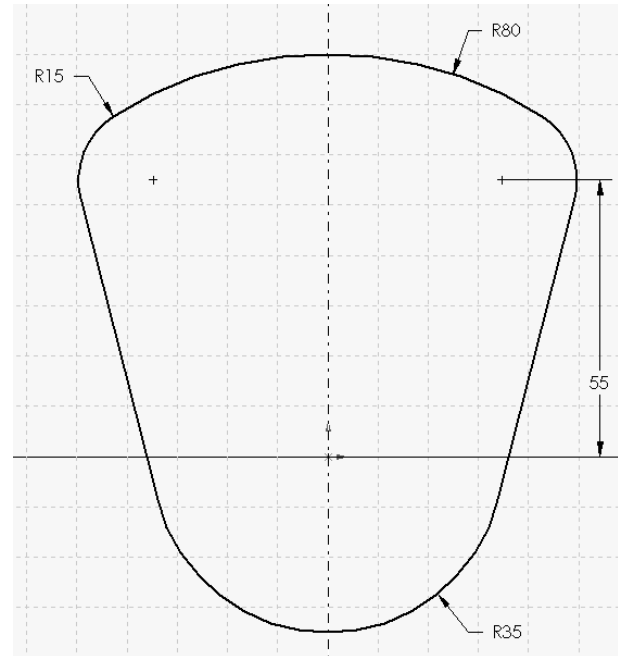
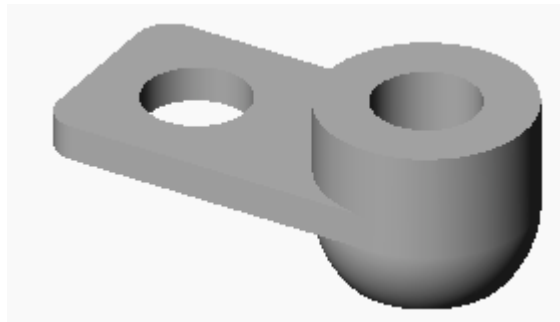
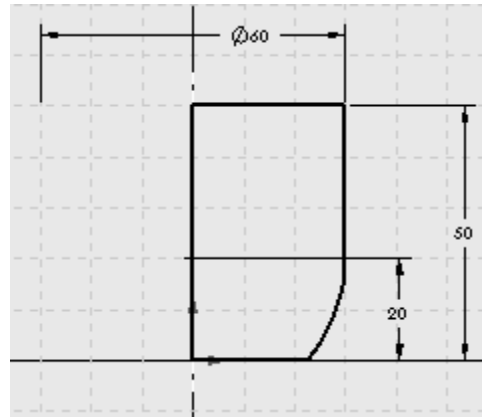


Figure 12. Possible Dimensioning Scheme for this Geometry.

Notice that there is no size dimension associated with each line segment or with the second smaller arc. This is due to the symmetric relations that are being applied, and in the case of the two line segments, it is due to a combination of the symmetry and tangency relations. In addition, it is plausible that an angle dimension would be desired to control the orientation between the line segments. If this is the case, then the Symmetry relations between the smaller arcs' center points and between the line segments would have to be removed.

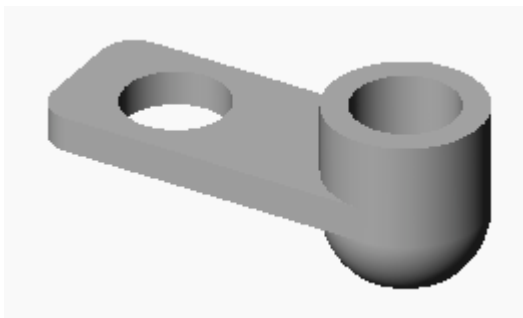


(a)

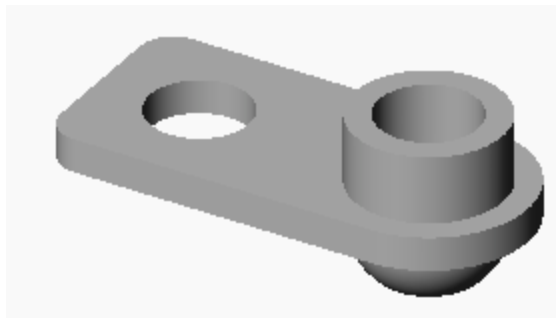


(b)

Figure 13. Driving 3D Geometry with 2D Data



(a)



(b)

Figure 14. Modifying the Model and Breaking Constraints

Driving 3D geometry with 2D sketches

Another powerful exercise for students is one where they are required to use 2D sketch data to drive 3D features. Figure 4 shows an object where the depth of the part is driven by the 60mm diameter dimension for the cylinder. Figure 13(a) illustrates what happens when the diameter is changed to 45mm if the correct tangent constraints are present between the horizontal plate and the cylinder. Figure 13(b) shows the same modification when the student is required to break the tangent constraint. As with the tangency example above, the spherical surface can be made tangent to the cylinder (Figure 14a) or not (Figure 14b), depending on how the design requirements are specified.

VI. Conclusion

A modern engineering graphics curriculum needs to move beyond the crafting of static models and drawings derived from these models. To do so begins with a clear taxonomy of geometric behaviors derived from dimensional and relational constraints applied to the model. Problem-solving oriented modeling assignments that demand dynamic manipulation of the models provides a vehicle for revealing model behaviors to the students and the instructors assessing the student work. Assessment based on dynamic modeling assignments provides the opportunity to

judge the quality of the model problem solution both on the modeling process and the end product.

Increasingly, industrial use of constraint-based modelers depends on the creation of sophisticated virtual products that represent multiple geometric configurations of current products and the embedding of geometric behaviors that represent engineering design constraints¹⁴⁻¹⁶. 'Smart' models that reflect critical design constraints allow for rapid creation of new versions of the product and saves companies considerable time and money. Increasingly, employers will demand graduating students who are able to create these sophisticated models. For students learning modeling in this dynamic, problem-based environment there will be the added benefit of being able to experience engineering design from the standpoint of geometric problem-solving. This will provide an essential counterpoint to classes engineering students already take where much of the problem-solving is numerically based.

These changes in the engineering graphics curriculum need to be implemented in such a way that they not create undue burden on instructors; otherwise, they simply will not be adopted. Dynamic modeling problems that allow reliable assessment through visual inspection of different iterations of a model or simple probing of the electronic model are indeed possible. All of the examples given above demonstrate behaviors that are clearly visible in pictorial views of the model, either in printouts or on the screen. The key will be to tightly define the design problem and what constitutes 'correct behavior'. Careful selection of models, geometric definitions, and required modifications will all result in a quality instructional experience. This does mean a rethinking of the traditional static model problems employed first in 2D drafting classes and currently being carried over into 3D modeling classes.

VII. Bibliography

1. Branoff TJ, Hartman NW, Wiebe EN. Constraint-based, three-dimensional solid modeling in an introductory engineering graphics course: Re-examining the curriculum. *Engineering Design Graphics Journal* 2002;66(1):5-10.
2. Barr RE, Juricic D. A new look at the engineering design graphics process based on geometric modeling. *Engineering Design Graphics Journal* 1992;56(3):18-26.
3. Ault H. Principles of parametrics - New concepts for the EDG curriculum. In: Sorby S, editor. *ASEE Engineering Design Graphics Division Mid-Year Meeting*; 1997; Madison, WI; 1997. p. 187-192.
4. Baxter DH. Expanding the use of solid modeling throughout the engineering curriculum. *Engineering Design Graphics Journal* 2001;65(1):6-13.
5. Cumberland RR, Miller CL. The foundation of a progressive engineering graphics curriculum. In: Barr R, editor. *Engineering Design Graphics Division Mid-Year Meeting*; 2001; San Antonio, TX; 2001. p. 15-21.
6. Connolly P. CAD software industry trends and directions. *Engineering Design Graphics Journal* 1999;63(1):26-33.
7. Wiebe EN. 3-D constraint-based modeling: Finding common themes. *Engineering Design Graphics Journal* 1999;63(3):15-31.

8. Wiebe EN. What's Involved with Moving from One 3-D Constraint-based CAD System to Another? In: Sadowski, editor. ASEE Engineering Design Graphics Division Mid-Year Meeting; 1999; Biloxi, MS; 1999.
9. Wiebe EN. Future applications of geometry and graphics. Engineering Design Graphics Journal 1999;63(2):13-20.
10. McCormick R, Murphy P, Harrison M. Teaching and learning technology. Wokingham, England ; Reading, Mass.: Addison-Wesley/Open University Press; 1993.
11. Clark AC, Scales AY. Assessment practices in engineering/technical graphics. Engineering Design Graphics Journal 2001;65(3):13-24.
12. Wiebe EN, Norton JJ, Summey J, Howe JE. Organizational assessment of integrating CAD and product data management tools in the furniture industry. Technical Report: Raleigh, NC: North Carolina State University, Furniture Manufacturing and Management Center; 1997 August. Report No.: 1997-3.
13. Bertoline GR, Wiebe EN. Fundamentals of graphics communication. 3rd ed. New York, NY: McGraw-Hill; 2002.
14. Courter B. Conceptual engineering: CAD tools that promote invention. White Paper: PTC, Inc.; 1999.
15. PTC. Behavioral modeling. White Paper: PTC, Inc.; 2000.
16. Versprille K. Capturing intent is essential to design reuse. Computer-aided Engineering 2001 March:56.

VIII. Biography

ERIC N. WIEBE, Ph.D.

Dr. Wiebe is an Assistant Professor in the Graphic Communications Program at NC State University. He has authored or co-authored four texts on technical graphics and has been involved in Computer-Aided Design (CAD)/3-D modeling development and use since 1986. During the past nine years, he has worked on the integration of scientific visualization concepts and techniques into both secondary and post-secondary education. Dr. Wiebe has been a member of the EDG Division of ASEE since 1989.

THEODORE J. BRANOFF, Ph.D.

Ted Branoff is an Assistant Professor in the Graphic Communications program and has been teaching there since 1986. He was previously employed with Measurements Group, Inc. as a draftsman and with Siemens, Switchgear Division as a specifications draftsman. Along with teaching courses in graphic communications, he has conducted CAD and geometric dimensioning & tolerancing workshops for both high school teachers and local industry. Dr. Branoff is a member of the Engineering Design Graphics Division of the American Society for Engineering Education, Epsilon Pi Tau, Omicron Tau Theta, Kappa Delta Pi and Phi Delta Kappa.

NATHAN W. HARTMAN

Nate Hartman is a doctoral candidate in Technology Education at North Carolina State University and a teaching assistant in the Graphic Communications program. His research interests center on the effective teaching and use of computer graphics tools. Prior to attending NC State, Nate worked for a variety of companies in using and integrating CAD into the engineering design process. He holds a Bachelor of Science in Technical Graphics and Master of Science in Technology from Purdue University.