SIEMENS

Goal Seeking in Solid Edge

White Paper

Goal Seeking in Solid Edge® software offers a fast built-in method for solving complex engineering problems. Whether you're drawing and dimensioning simple 2D sketches or analyzing driven and driving formulas, variables and dimensions attached to a 3D product assembly, Goal Seeking removes much of the burden of solving common engineering problems.

**www.siemens.com/plm**

Contents

Abstract

Engineers are constantly faced with solving problems – it's what they're paid to do. The faster a formidable solution or the harder the problem, the more successful that person will become. Depending on the problem or the tools, finding an answer is often easier said than done. While many formulas, examples and methods for attacking problems are of public domain, engineers often need to solve problems from the ground up as, in many cases, the question at hand is not perfectly laid out in a book.

Each engineering discipline has its own unique challenges that often fall into some common classification. For example, mechanical engineers are generally tasked with designing power transmissions, building structural frames and a host of other scenarios that are easily solved using free-body diagrams and known engineering calculations. Costly alternative solutions involve doing calculations by hand by writing specific equations and manually iterating through them, or by entering equations in spreadsheets and adjusting variables until the desired result is obtained. Extreme alternative cases require constructing and testing physical prototypes.

Difficulties arise when users iterate through those calculations as changes occur or when they try to optimize a system by varying an input variable (those on the right side of the equation).
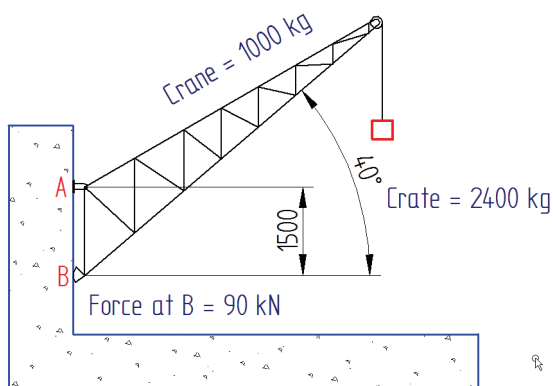
A common example is determining the location of a set of pulleys for a power transmission application. While it's not too hard to lay out the components, simply changing a pulley diameter or increasing tension is where problems begin (remembering that in most cases belts are purchased at a fixed length).

Solid Edge provides a handy tool for automating engineering calculations to achieve a specific design goal. It operates on driven and driving formulas, variables and dimensions attached to 2D and 3D geometry – from a simple sketch, all the way to a complex product assembly. Goal Seeking finds a specific value for a dependent variable (dependent by formula, for example) by adjusting the value of another variable until it returns the result you want. It shows you the effect on the geometry, and it updates the Variable Table with the new value. Results can ultimately control the size and position of 3D geometry and components, thus retaining the initial intelligence throughout the design process.

Current methods range from brute force evaluations with calculators, spreadsheets or, in some cases, licensed add-on applications that are supposed to work with the CAD application. In any case, without an elegant solution like Solid Edge Goal Seeking, arriving at a solution is prone to errors associated with first setting up the problem, forgetting to enter some variables or bad results due to mismatched units of measure. Solid Edge built-in Goal Seeking addresses these issues and lets designers focus on solving problems – as opposed to developing complex equations, solving them, and double checking the answers.

The general problem with problem solving

Take the pulley system in figure 1 and try to determine where the tension pulley must be placed in terms of the X and Y dimensions in order to obtain a standard belt length of 25 in. (635mm). A complex equation based on a plethora of trigonometric operations will be needed to generate a formula sufficient to solve this variable. Using a CAD system with a parametric 2D solver will still require that the user "sneak" up on the desired belt length by increasing X since 2D solvers typically treat the circumference as a driven value. There really is no easy way to set a desired belt length but to solve for X. A specific set of equations could be established to enable this, but a completely different set will be needed should a different variable need to be controlled. While a 2-pulley system is simple, it is extremely difficult to solve a system with 4 or more pulleys – do the math (closed book, no notes) to see why.
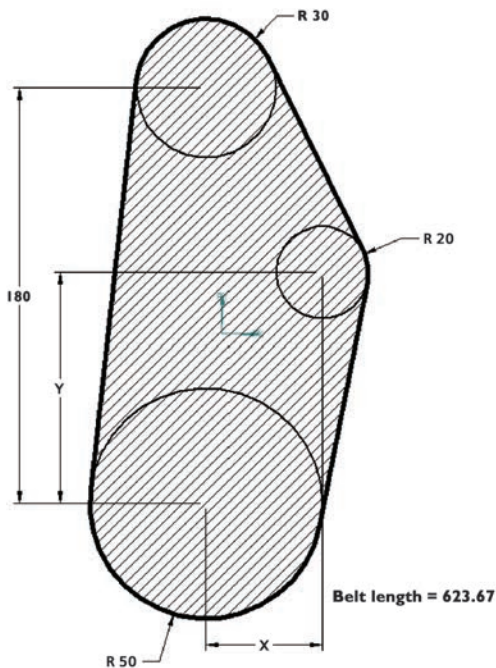
Let's back up a bit and start with something easier. The classic example used here is a lever where a force is applied to one end with the intent to lift the other. Sliding the fulcrum along the lever varies the amount of force while sacrificing the lift. A simple free-body diagram in figure 2 illustrates this concept. Applying a force of 100 pounds to the right will lift a 200-pound load – simple to compute, but where will the fulcrum need to be placed in order to lift 286 pounds? Again a simple equation, but rearranging that equation is necessary should the 120 inches need to become 102 inches.
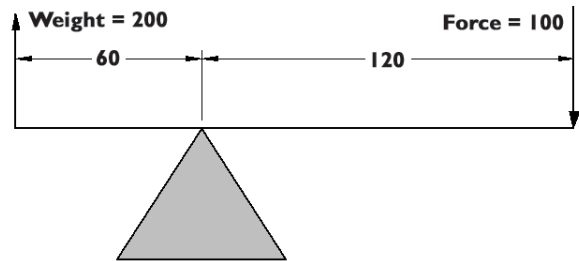
Figure 2.

The amount of work increases with every question; so does the change for error. This latter example is almost too simple to discuss, but it illustrates the need to be able to attack a problem from different angles. Other common types of problems that often require varying an input parameter to obtain a desired result are: determining beam dimensions based on loads, optimizing areas by varying some boundary parameters, finding critical angles in 4-bar linkages, finding pulley locations for constant belt lengths, computing balance point, calculating maximum and many others practical examples.

Figure 1.

The answers with Goal Seeking

Goal Seeking in Solid Edge automates engineering calculations by eliminating the need to rearrange equations and, in many cases, removes the need to even develop equations. The system is available in the 3D environments and when drawing 2D geometry on a 2D model sheet, drawing sheet, profile or sketch, Goal Seeking assumes you have dimensioned geometry containing area and or volume. Users simply dimension the geometry, establish any interrelated equations and let Goal Seek find the unknown parameter. At the heart of the system, certain measurements, such as area, perimeter and volume, are stored and used by the solver when computing the free-body diagrams. This capability greatly reduces the need for developing complex equations such as capturing areas and volumes.

Let's see this concept in a practical example. Finding the area of a triangle simply requires plugging in the base and height into the equation area = 1/2 (base x height) and out pops the answer. However if a specific area is desired, but say the height can vary, one will have to rearrange the equation and solve for height. While simple here, some of the previous examples (and many others) aren't so easy. Goal Seeking in

Solid Edge does just this. Regardless of the equation (or free-body diagram), any variable can be solved.

Once the free-body diagram is drawn and dimensioned, Goal Seeking finds a specific value for a desired variable by adjusting the value of another variable. Goal Seeking shows you the effects on the free-body diagram during the process to help give a visual indication of what's happening. The solve process uses optimized algorithms to arrive at a solution fast – assuming a solution exists. If not, the system warns the user that the requested solution geometrically (and mathematically) doesn't exist. In the pulley example above, should the user set a belt length that is too short (let's exaggerate by setting it to 100mm) while attempting to find a valid X dimension, Solid Edge will warn the user that the request can't be achieved.

Let's take a close look at how Goal Seeking can be used to solve the above pulley example. After drawing the geometry using typical sketch and relationship tools, the area is measured and saved. During this process, the perimeter (belt length) is also captured. The command for gathering input
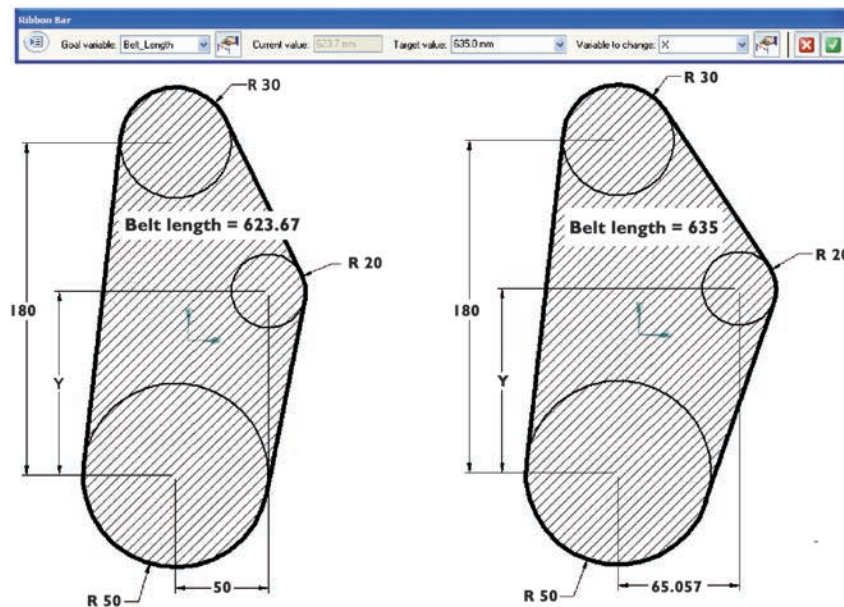

Figure 3.

guides the user through all required steps by basically asking "what's the goal?" and "what's to be changed?" Here the belt length is to be set to 635mm while adjusting X until that length is satisfied. Goal Seeking quickly snaps to the answer as seen in the change from the initial image on the left to the results on the right (see figure 3). The user interface spans the top of both. Notice the simplistic approach that Goal Seeking uses to help the user capture the inputs values.

Goal Seeking can help engineers in many other areas. The examples above are for problem solving on 2D geometry. However, it is equally effective on dimensions driving 3D models. The implications here are that initial calculations used to determine critical fit and position between parts are available throughout the design process. Should the "final" design need to accommodate a greater load, store more fluid or transmit power faster, the Goal Seeking tool can easily handle the calculations and ultimately drive the 3D geometry. Figure 4 shows an excerpt of a complete 3D assembly of an irrigation tank driven from 3D dimensions – which was sized using Goal Seeking.

The solution offered by Solid Edge is unique in the industry because it is integrated into the 2D and 3D solve process and works directly with the Variable Table, an internal table that lists all dimensions and allows equations to be established between them. Others CAD solutions often rely on add-on "calculators" that are licensed from third parties and aren't built into the solve process, hence requiring the user to force an update.

The competitive landscape for product development means products must be designed better and faster. Having to manually calculate basic parameters to solve fit and position problems is often counterproductive and subject to error.
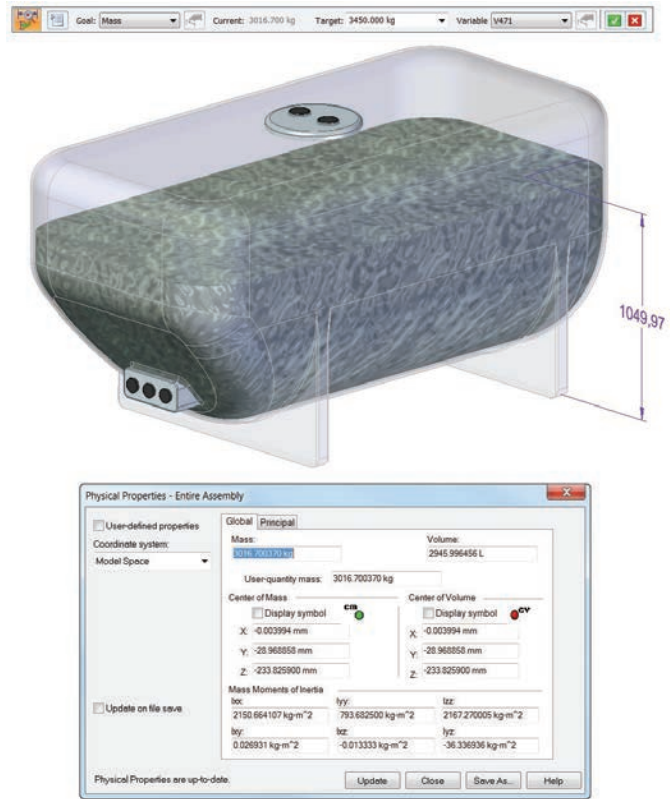

Figure 4.

Conclusion

Engineers are constantly faced with solving tough problems and the buck often stops with them. It's important that they have efficient and reliable tools to provide answers that justify their confidence. Goal Seeking in Solid Edge does just that – by providing powerful problem solver tools that are integrated with a 3D modeling system. Using driven and driving formulas, variables and 2D and 3D dimensions to define the problem, engineers with Solid Edge have a fast, easy way to solve complex problems – the perfect formula for faster product development.

Additional reading

This section lists other white papers that illustrate further advantages with Solid Edge.

When a complete finite element analysis is required, turn to Solid Edge Simulation or Femap™ software. "FEA For All Engineers" (922KB, PDF)

Learn more about how to tie 2D conceptual sketches and layouts to 3D, from data import to Zero D. "Solid Edge Hybrid 2D/3D Design" (442KB, PDF)

Create motion studies and generate high quality images and animations of products in action with the explode, render and animate environment. "Solid Edge motion simulation, explode – render – animate" (909KB, PDF)

Siemens PLM Software

Headquarters
Granite Park One
5800 Granite Parkway
Suite 600
Plano, TX 75024
USA
+1 972 987 3000

Americas
Granite Park One
5800 Granite Parkway
Suite 600
Plano, TX 75024
USA
+1 314 264 8287

Europe
Stephenson House
Sir William Siemens Square
Frimley, Camberley
Surrey, GU16 8QD
+44 (0) 1276 413200

Asia-Pacific
Suites 4301-4302, 43/F
AIA Kowloon Tower, Landmark East
100 How Ming Street
Kwun Tong, Kowloon
Hong Kong
+852 2230 3308

About Siemens PLM Software
Siemens PLM Software, a business unit of the Siemens Industry Automation Division, is a world-leading provider of product lifecycle management (PLM) software, systems and services with nine million licensed seats and 77,000 customers worldwide. Headquartered in Plano, Texas, Siemens PLM Software helps thousands of companies make great products by optimizing their lifecycle processes, from planning and development through manufacturing and support. Our HD-PLM vision is to give everyone involved in making a product the information they need, when they need it, to make the smartest decisions. For more information on Siemens PLM Software products and services, visit www.siemens.com/plm.

A white paper issued by: Siemens PLM Software.