A SolveSpace Learning Guide

by Thomas Knight (421)
2018

(for the 2D/3D CAD software SolveSpace, v2.3)

My approach here is like a concise textbook. All of the concepts in the 5 SolveSpace Tutorials are covered. There's also a fair amount of information from the Reference document. But, this text imparts the ideas given here in about half the space (based on file size). The goal is to get you competent, comprehensively, as efficiently as possible.

It's organized in sections, so that you get some concepts, then practical methods, then more concepts, etc. Some content is in blocks labeled ***Intermediate*** and ***Advanced***. You can skip those the first time through, then come back to them afterward. They're scattered so that they're close to related basic material, which aids integrated learning, I think.

Unlike a tutorial, there are no step-by-step examples to follow. (Procedures are stepwise, of course.) So, try out the ideas and click around in the software while you read, even in the conceptual sections. **Undo (Ctrl-Z)** and **File > New** will clear your mistakes or start fresh if you get stuck.

The Reference remains a good source for details, such as for configuration, line styles and model export. And credit must go to the SolveSpace creators and community that generated the Tutorials and Reference, which contain many or most of the concepts presented here.

Contents

Environment
View
Construction Concepts
Doing
.....Drawing, Extrusions
More On The View
.....Groups
.....Other
More Construction Concepts
.....Workplanes
.....Constraints (Symmetry, Dimensions, The Solver)
.....Constraints (2D vs. 3D) ^
.....Extrusions (Multiple Extrusions & Assembly Consideration)
.....Lathes
More Doing
.....Finding Things
.....Drawing (Tangent Arcs, Splitting Curves, Text, Chord Tolerance)
.....Bezier Curves
.....Workplanes
.....Workplanes ^
.....Constraints
.....Extrusions
.....Selecting & Styles (Styles)
.....Exporting Drawings & Models
More Construction Concepts #2
.....Construction Geometry
.....Stepping Copies (Translating, Rotating)
.....Area & Volume Measurement
Assemblies ^
.....Stepping To Place Copied Parts
.....Checking Interference
The Assembly Mechanism ^^
Linkages ^^
.....Testing
.....Errors
.....Linkages & Solid Models

Environment

On startup, SolveSpace shows 2 windows, the main drawing window and a Property Browser. In the drawing window, there are 3 coordinate reference planes, drawn with dashed quadrant lines, with the XY plane facing the screen. The X, Y, Z axes indicators are bottom-left, in red, green, and blue. The first user workplane (called "g002-sketch-in-plane"), in which your first 2D drawing will be, is coplanar with XY and also shown. It should be listed in the Property Browser.

The toolbar at the left of the drawing window shows 4 sections:

1st (top): Sketching (drawing) Tools	3rd: Group Tools
2nd: Constraints	4th: View Controls

Many commands are available in the menubar, toolbar, and with keyboard shortcuts. The shortcuts are in parantheses after their commands in this text.

^ = Intermediate, ^^ = Advanced

View

To rotate the view (to see YZ and XZ planes) middle-click the mouse and drag. To pan the view, right-click and drag. To zoom, use the mousewheel or **+** and **-**. (This might look strange with no drawing yet--the dashed lines of the workplanes are always the same size, regardless of zoom.) To return to a centered, perpendicular view of the active workplane (the one you're working in, or default XY when starting), use **View > Align View to Workplane** (**W**). To center on a point, select it and hit **F4**.

A contour (shape) with 2 lines constrained to be horizontal, 1 line and 1 point selected, and 1 line highlighted, near the origin.

Hovering over an object highlights it in yellow (including the workplanes). Hovering over toolbar icons shows tooltip labels.

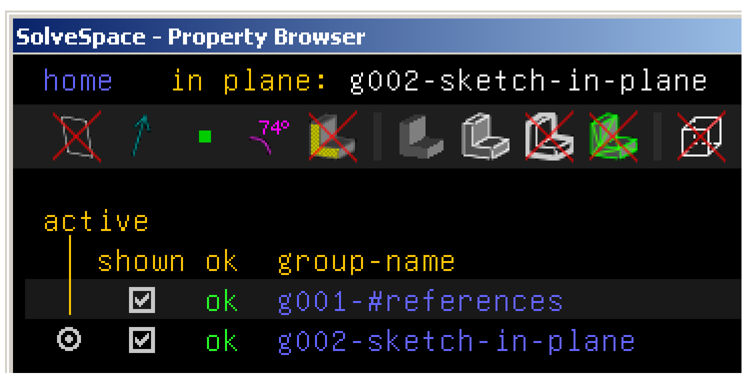
Clicking on an object (point, line, face, etc) selects it--it turns red. Clicking on many objects selects them all. Select everything in a rectangular area by left-clicking and dragging a selection box. Clicking on nothing or using **Escape** (or **Edit > Unselect All**) de-selects all.

The **Property Browser** window shows information about points, shapes and actions taken, such as the coordinates of a selected point. It's also where configuration settings are. Enable it in the **View** menu if it isn't shown.

The top of the **Property Browser** has an iconbar for toggling visibility of drawing elements, such as workplanes, normals, points, solid model parts, hidden lines, etc.

Below the iconbar in the **Property Browser**, in its home screen (reachable with the **home** link or **Escape**) is a list of groups. All of your drawing will exist in at least 1 group. More on groups later.

Construction Concepts



All geometry is represented in 3D. But objects can be restricted to 2D by having a workplane active while drawing them. Workplanes are just planes used to draw 2D objects. You can have as many as you like.

2D objects are drawn as shapes using line segments and curves. 3D objects are made (most easily) by drawing 2D objects, functioning as cross-sections, and extruding them. But you can draw in 3D, without a workplane, as much as you like.

2D shapes (i.e. contours) can be closed (enclosing a space) or open (a "not closed contour" message will show). Most objects get created with closed contours.

Points and lines in contours can be dragged with the mouse--left-click and drag. This works even on contours with constraints (limitations) on them. But they won't move such that constraints are violated.

Doing

Drawing

Make the default (g002) workplane active if it isn't already with **Sketch > In Workplane (2)**, for 2D.

Draw a line by choosing **Sketch > Line Segment (S)**, left-clicking in the draw space to start it, and right-clicking to end it. To draw multi-sided shapes and multi-segment lines:

1. Instead of ending the first line, create a series of connected lines by left-clicking at each vertex.

2. Then finish 1 of 2 ways:

- left-click on the starting point (hover until it's highlighted) to close the contour
- right-click or **Escape** at the last point to leave contour open

To make a circle, choose the circle tool (**C**), click where its center should be, drag out the circumference, and click to finish. Draw an arc with its tool (**A**) by clicking where one end should be, dragging to where the other end goes, and clicking. An arc is placed as a semicircle--change it's size/span or radius by moving the endpoints or center, and drag it with the arc-line.

Datum points (**P**) are similar to the points at the end of a line. They're useful for adding to contours for making constraints.

Regular polygons must be constructed with constraints. And Bezier curves are covered later, in the ***More Doing*** section.

Again, you can change all your shapes by dragging their points and edges, as long as you don't violate constraints. Constrain a point or contour, or 2 such objects to be related, by selecting it or them and choosing the constraint to apply.

- Example: Constrain a point to the origin by selecting both the point and the origin, and choosing **Constrain point on line/curve/plane/point (O)**. In this case, the origin can't move. So the other point gets moved to it.

Much more on constraints is below.

Extrusions

To create an extrusion from a contour you've drawn, choose **New group extruding active sketch** (**Shift-X**). The default is to extrude on 1 side of the contour's workplane. Try it and rotate your view to see it's shape. You can drag extruded points (and edges except on cylinders) to change the extrusion length. A new group has been created to contain the extrusion, and is separate from the contour on which it's based. It's details are shown in the **Property Browser**. (Again, hit **Esc** or the **home** link to see the group list.) You're placed in 3D mode, too--select a workplane (e.g. XY) and hit **2** for 2D. Extrusions are covered more later.

More On The View

Groups

Drawings are organized in groups, which you create with **New Group** menu commands or tools. The 1st one ("**g001-#references**") contains the coordinate planes and origin, and cannot be changed. Subsequent groups arrange the historically-sequenced record of your group-relevant actions. So, you'll draw in a group, make a new group, draw in that, make another group, etc.

Groups can be useful: by changing a drawing in a group, changes will propagate to later groups, according to the constraints you create and any dependencies between the groups. For example, change a contour and the extrusion of it changes too. The active group can be chosen (click **active** in **Property Browser** for a group), and groups up to the active one can be made to show in the drawing window or not (click **shown**). Groups newer than the active one are not shown in the drawing. The active group drawing is white; earlier ones are shown in brown. Clicking a group's name in **Property Browser** shows its properties, and the entities it contains. It can be renamed or deleted, too.

Other

If you haven't noticed (via the **View** menu), there are 2 projection modes: isometric (**F3**) and perspective (**`**). And the typical orthographic view (**F2**) is isometric, oriented toward a plane. Using **Nearest Ortho View** (**F2**) or **Nearest Isometric View** (**F3**) won't center your view on the origin though--use **W** (in 2D mode) for that.

You may have noticed by now that your view can be limited or affected by what you do, and vice versa. This interplay is complicated, and will be increasingly explained as you read further.

You can disable the message regarding open contours and varied line styles in objects, if you don't need it... Clear the **check sketch for closed contour** option in **configuration** in the **Property Browser** (via the **home** screen).

More Construction Concepts

Workplanes

Workplanes are defined by an origin point and a normal. If you're in 2D, the active workplane is shown at the top of the **Property Browser**. To deactivate a workplane, activate a different one or switch to 3D with **Sketch > Anywhere In 3D (3)**. To activate a workplane:

- select it in the drawing window, and use **Sketch > In Workplane (2)**, or
- make the group it contains active in **Property Browser** and hit **W** (or **2**) (inapplicable to extrusion groups).

Workplanes can be created explicitly but this often isn't necessary. Typically, they're created incidentally by creating new groups, in workplanes parallel to or defined by existing ones, or defined by existing 3D lines.

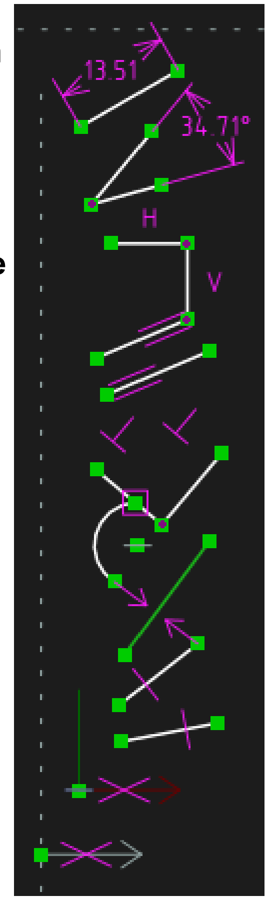
Workplanes do not have to have origins coincident with others, nor be coplanar, nor be parallel with the coordinate planes.

Constraints

Constraints are the best way to create the kinds of sketches and objects SolveSpace was designed for--parametric constructions. They constrain parts of designs to definite and precise dimensions, i.e. parameters, rather than imprecise, hand-drawn magnitudes.

They can be attributes of a single element, such as the length of a line, or relationships between points, lines, edges, and shapes that govern their behavior. They're indicated with purple symbols. For commands for multiple object types (e.g. line vs. circle), the constraint created depends on the type(s) of the selected object(s).

For most of them, you select the entity or entities to constrain, then click the constraint command. The Symmetric constraint, exceptionally, allows you to select the axis of symmetry, too.



- **Constrain distance/diameter/length (D)**: fixes the relevant dimension (length of a line, e.g.) at whatever it was drawn as. A purple dimension line with numeric value is shown. Double-click the value to change it.
- **Constrain angle (N)**: fixes the angle between 2 vectors (lines/normals). A purple angle arc with value shows in the angle, which can be moved. Double-click to change the angle. To change the setting to a supplementary angle, use **Other supplementary angle (U)**.
- **Constrain horizonal (H)** or **vertical (V)**: make a line or 2 points stay aligned horizontally (or vertically). A purple H (or V) will appear next to it. As these are simple for computation, they're recommended if possible.
- **Constrain parallel / tangent (L)**: make 2 lines/normals parallel, or 2 connected contour segments tangent. Purple line pairs will appear around each participating line, or a purple T at the connection.
- **Constrain perpendicular ([)**: make 2 lines perpendicular. Purple perp symbols will appear next to them.
- **Constrain point on line/curve/plane/point (O)**: constrain selected point to the second selected object (another point, e.g.). A purple dot on the green points shows for on-point. Purple box around point for on-line / on-curve.
- **Constrain symmetric (Y)**: centers 2 points or a line segment across a line or workplane, which is inferred if not selected (see below). Purple arrows point to each other from each point.
- **Constrain equal length / angle / radius (Q)**: The type is inferred based on types of objects selected--many combinations are possible. Purple congruent symbols appear on them.
- **Constrain > Same Orientation (X)**: makes normals of 2 objects point in the same direction (parallel or anti-parallel, whichever's closer). Also locks the rotation of the objects around their normals. The 2 normals get purple "X" labels.
- **Constrain > Length Ratio (Z)**: sets the ratio of the length of the 1st-selected line to that of the 2nd. Shows purple congruent marks plus the ratio.
- **Constrain > At Midpoint** (**M**): keeps the midpoint of a line and a point together. A purple "M" shows next to the point.

Constraints can be selected for taking action on them, for example to delete or change parameters.

Starting a line (or curve) by clicking close to an existing object will automatically create a coincident constraint between the two.

Even when drawing just 2 lines that share endoints, a constraint exists: SolveSpace automatically inserts point-coincident constraints (point-on-point) keeping the 2 points together. They're labeled with purple dots inside the green point squares. Note that this means you can delete them to break up and modify drawn shapes. (Delete them with a right-click or select them in the **Property Browser**.)

The X, Y & Z coordinate axes indicators, always at the bottom-left, are selectable, and therefore useful for creating constraints parallel to them. So are the normals of all workplanes (at their origins).

Symmetry

When creating symmetry constraints, SolveSpace will guess an axis of the workplane as the line of symmetry if one isn't selected. Whichever orientation the points are most closely aligned in (vertical/horizontal), the chosen axis will be the opposite--e.g. vertical axis for points nearly horizontally aligned.

Sometimes it may be helpful to create consruction-only lines to use for symmetry axes. Create a line, select it, and hit **G** to make one. Construction objects are covered later, in the ***More Consruction Concepts #2*** section.

Dimensions

Dimensions can be either constraints that fix the size of sketch elements as drawn (e.g. length), or references merely displaying sizes as a drawing changes. When a reference, a dimension will have "REF" in its label. Toggle which with **Toggle reference dimension (E)**. Also, in most places, fixed dimensions can be entered as arithmetic expressions.

The Solver

SolveSpace is always checking the constraints in a drawing, and whether they're inconsistent or redundant, or can't all be satisfied. If the drawing window background turns red, it means the constraint you just created doesn't pass the SolveSpace solver. You can Undo your last constraint, or see the **Property Browser** for a list of constraints to remove to try to fix the problem. Examples:

- A triangle can't have 3 line lengths of 2, 4 and 7--such a triangle can't be constructed.
- A triangle doesn't need 3 internal angle constraints of 60°, 45°, and 75°--that's redundant and thus over-constrained. Two angles suffice.

Sometimes you'll get this temporarily while you're creating a Reference dimension--toggle the constraint and the error disappears.

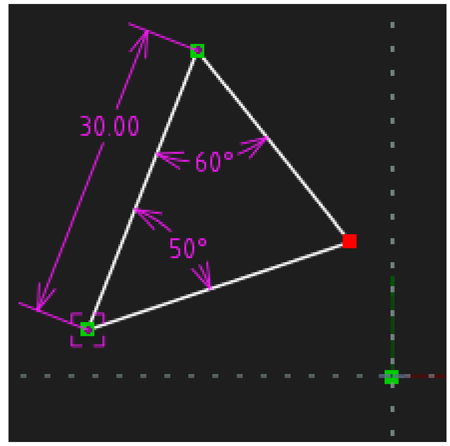
Intermediate

Constraints

The degrees of freedom (DOFs) of a drawing are intimately tied to its constraints and how it can be drawn. So understanding DOFs could help us avoid bad drawings. But it's usually very difficult to do this.

For example, a triangle in a plane with no constraints has 6 DOFs: X,Y coordinates for each vertex. If we then created constraints for 2 of its angles and the length of 1 side, which fully determines the shape of the triangle, it can then only be translated and rotated in the plane.

This possible translation and rotation are identifiable as X,Y coordinates of the location of the whole triangle and a rotation angle (relative to some reference axis). (Really, what we're saying is that given the X,Y coordinates of 1 vertex and either the X or Y coordinate of another vertex, we can determine the other X and Y vertex coordinates.) Given the constraints, there are 3 DOFs remaining.



However, the effort of tracking this prompts us to use another method. It's usually better to try to avoid inconsistencies and redundancies by using the minimum restrictions necessary, with progressive tests of the geometry as needed.

For example, with the triangle above, suppose you placed a locked-where-dragged constraint on 1 vertex, as shown in the image here. The triangle can then rotate around that point. If you wanted to stop that, you could put a similar constraint around the selected point. But that would over-constrain the triangle and create an error. To avoid the error, you could instead put a distance constraint between it and one of the axes, for example. And, instead of counting DOFs, you could recognize that simpler and permitted solution by trying to move the triangle and then thinking of the simplest constraint to stop it. Because the triangle's rotation always entails the selected point's rotational translation, which is always a change in 2 coordinates, all you have to do is fix 1 of those coordinates to block rotation.

If an error does arise, the "simplest restriction" approach should yield an answer. If not, then you finally have to try to find a different constraint (or mix of constraints) that restricts fewer DOFs. Most constraints restrict 1 DOF, some restrict 2, and some in 3D restrict 3.

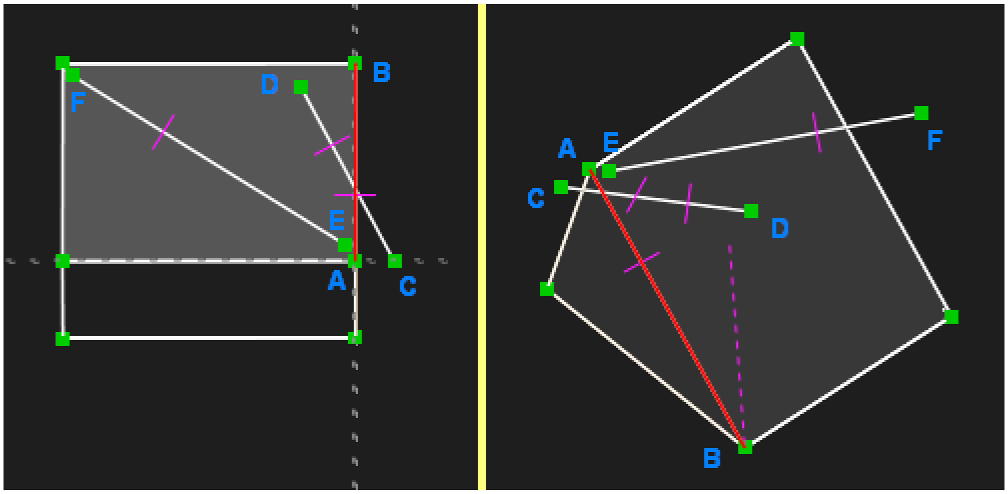
Constraints can result in multiple "solutions" for a drawing. A triangle with 2 sides vertically and horizontally constrained (surrounding a 90° angle) and the vertex between them constrained to the origin, could be in any of the 4 quadrants of a plane. SolveSpace will keep it in the quadrant in which it started. Also, a limited set of characteristics can be had by multiple shapes. So, for example, a 5-pointed shape with all sides constrained congruent and all vertices constrained to be on a circle could be a pentagon or a star. SolveSpace will keep the drawing as close as possible to its shape before constraints were applied.

2D Vs. 3D

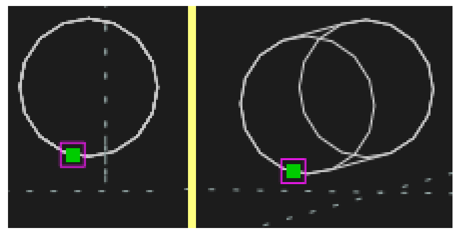
Most constraints are applicable to both 2D and 3D. But all limit only exactly what's required. So, a 3D-capable constraint can operate on 2D objects in 3D as well as on 3D objects in 2D. (Remember, a 2D object is really a 3D object with its non-workplane coordinate restricted to zero.)

If creating a constraint when a workplane is active (2D mode), then it operates on the cartesian coordinates of objects as seen in that workplane (thus the projection of any 3D objects into that 2D plane).

In the screenshot here, a 3D wedge has been drawn (gray object). Parallel to the plane of the image on the left, 2 lines, CD and EF, have been drawn. Both lines were then constrained to be the same length as line AB, an edge of the wedge (highlighted in red). But, when the view is rotated as shown on the right, we see they're different. CD was constrained with its workplane active. So the constraint operates on AB's projection into that workplane (indicated by the dashed purple line). So CD is actually shorter than AB's real, 3D length. EF was constrained while sketching in 3D, though. So it's length matches AB's true 3D length.

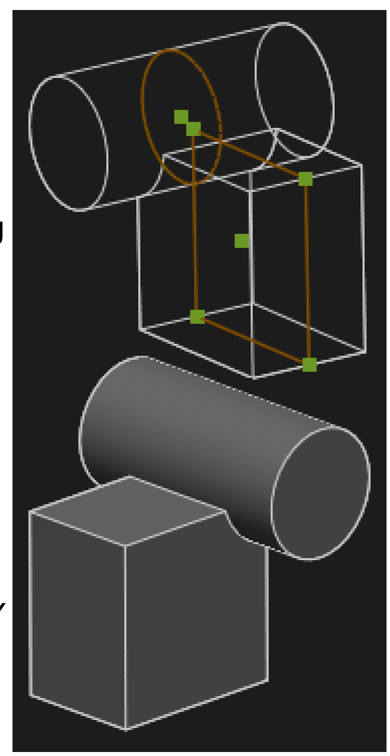


The complementary principle to the above also holds. If creating a constraint in 3D mode, then it operates on the projection of any 2D objects into 3D (with their non-workplane coordinate variable). In the example image at right, I created a point-on-curve constraint between a point in 3D and a circle in a workplane, while in 3D mode. So, the point is only limited to the circle's projection in 3D (a cylinder--of infinite length). I extruded the circle, as shown in the rotated view on the right, just to help illustrate this. The point can move anywhere in free space, as long as it stays on the surface of that cylinder.



Note, also, that constraints function across workplanes (and groups). If the point in the last example above was in a workplane (different than the circle's), it would be constrained to move along the (elliptical) intersection of the circle's cylindrical projection with that workplane. Because both objects are limited to 2D, whether the constraint was made in 2D or 3D mode wouldn't matter.

Extrusions



It's often best to create extrusions symmetrically across planes because it makes constructing models, and subsequent constraints, simpler. To do this, select an extrusion's group in the **Property Browser** and choose **extrude plane sketch : two-sided**. Dragging a point or edge (and thus sizing the shape) on one side of the symmetry plane then will move the corresponding one on the other side likewise.

When making an extrusion (to make multi-part objects, e.g.), it can combine with the rest of the model by either a boolean union or a difference. This is selectable in the **Property Browser** for the selected extrusion (**solid model as : union** or **difference**).

In the example image at right, I first extruded a square to make a rectangular solid. Then I created a new workplane coplanar with XY (this is covered in the intermediate ***Workplanes*** section below), drew a circle in it, and extruded that as a union. I set both extrusions to 2-sided because I wanted them to be aligned at their centerlines. Given that their workplanes are coplanar, their 2-sidedness makes that automatic--no constraints are needed.

The path of extrusion is always normal to the active workplane. If you have a typical contour in XY, then make YZ active for example, and extrude (along X), you'll get a flat mess filled with errors.

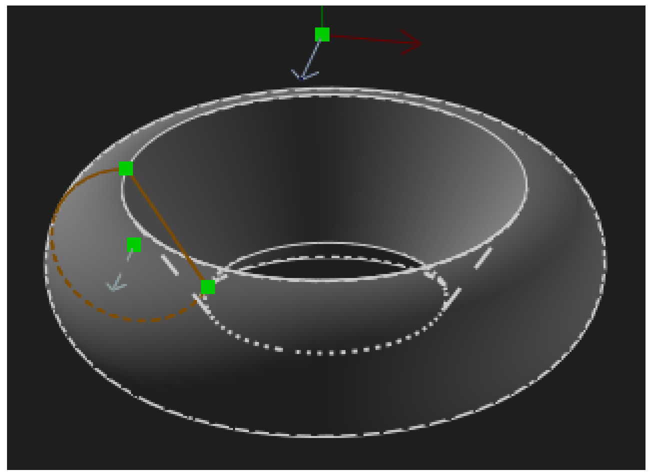
As mentioned earlier, an extrusion group's details are on its page in the **Property Browser**. You can change things there such as its color and opacity.

Multiple Extrusions & Assembly Consideration

Note that even if you draw multiple separate contours, if they are in the same group and then you perform an extrusion, they're all part of that extrusion. And if multiple contours are each in a different group before extruding, within the same file, they will be considered members of the same part when making assemblies (discussed in ***Assemblies*** section below). So if you're making parts that might get re-used independently, for example a gear and a bracket, put them in separate files.

Lathes

The Lathe tool revolves the points in contours in the active group around an axis explicitly or implicitly selected. For the axis, you can select a line segment, or a point plus a normal or line parallel to the desired axis going through that point. Contours that would produce a revolution that intersects itself will fail to lathe. Use **New group revolving active sketch** (**Shift-L**).



At right is an image of a pseudo-torus, with a conic interior, lathed from an arc connected to a line (shown in brown), around a (not shown) vertical axis in the middle.

Like extrusions, a lathe gets a new group. The page for it in the **Property Browser** has the details.

More Doing

Finding Things

Clicking on a group in **Property Browser** shows entities in it. Hovering over items in the **requests** & **constraints** lists highlights them in the drawing window. This is useful for finding things in crowded or complex drawings. Note that you'll have to activate a group to highlight its constraints--newer groups block their highlighting.

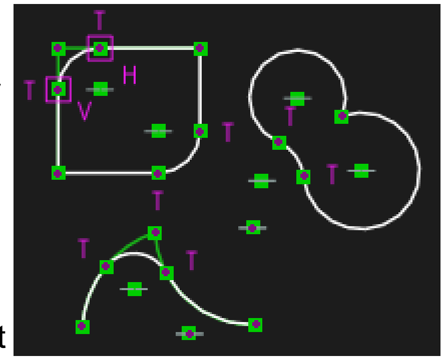
Also useful is the **Hide All** and **Show All** links in the **Property Browser home** screen. This will hide or show all lines and points and leave visible just the shaded 3D model of all groups up to and including the active one (unless you toggle off shaded view in the visibility iconbar). The **hidden lines** icon at the top right is useful for showing points and lines obscured by surfaces of solid models. Note that obscured points and lines are selectable even if **hidden lines** is off, as long as their group is set to **shown**.

Drawing

Cut and paste works on shapes in SolveSpace--in 2D only.

Points can be snapped to a grid with the **Edit** menu or **right-click**. (Toggle grid in the **View** menu.) But it's usually better to use constraints.

Tangent Arcs

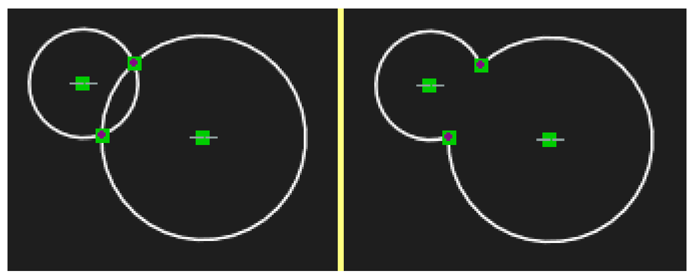


To radius a corner where 2 lines or curves join, select the corner point and use **Create tangent arc at selected point** (**Shift-A**). The corner is cut and interrupted by a new arc joined to the adjacent lines/curves. Drag its center or endpoints to change the radius. Purple "T"s show at arc ends. It can't be used on Bezier curves.

To specify parameters beforehand, invoke this command without a selection. You then can choose (in **Property Browser**) to set a radius automatically or to a fixed value, and whether to delete original lines or not, for all subsequent tangent arcs. If you don't choose to delete original entities, they'll be replaced with construction lines or curves, to which will be transferred their constraints. If you choose to delete originals, their constraints will be deleted too.

(Omitting the corner point in the original shape, drawing an arc separately, and constraining it to lie on the endpoints of adjacent lines would create the same effect. But a tangent arc is simpler.)

Splitting Curves



Multiple 2D contours that overlap can be combined to make complex shapes. Select 2 lines, or 2 curves, or 1 of each, in the contours that cross each other, and use **Split lines/curves at intersection (I)**. The crossing items will be split with a point at the intersection. Note that you have to do this for each crossover. You can then delete the newly-interior lines. In the image here, 2 splits were needed.

Like Tangent Arcs, any constraints the intersecting lines/curves had are lost. If you mark a participating line/curve as **construction** before the Split, then it gets "left behind" with any constraints, and the split gets applied to a copy of it, as non-construction geometry.

Text

Text can be added to your drawing (and constrained). There are 2 commands for this:

1. One is the **TrueType Text** sketching tool (**T**). With this (in 2D only), click where the upper- and lower-left corners of the text should go. As you move the cursor after clicking the upper-left point, SolveSpace will show you the angle and size of the text. When placed, select it to set the (OS-installed) font and text in the **Property Browser**.

2. The other is the **Comment** constraint (**;**). Choose it and click where the center of the text should be. Double click to type the text. It has the line style of a constraint by default, which you can change. If made while working in 2D, it's added to the active workplane. If in 3D, it "floats", rotating to face the screen all the time.

Chord Tolerance

SolveSpace sometimes represents curved entities as line or triangle sequences. If the calculations on these might be slowing down the program, try setting a larger chord tolerance (in **configuration** in the **Property Browser**), which is how far from the true curve a linear approximation is (in %). The **max piecewise linear segments** setting limits how many lines are used for the approximation, inversely proportional to performance. The **Reference** has more detail.

Bezier curves

To create a Bezier curve, select its tool (**B**) and click to start it. Left-click again if you want to add intermediate on-curve points (creating a multi-segment curve). Right-click to end it. Change the shape with the on-curve and off-curve control points. Deleting any point on the curve deletes the whole curve.

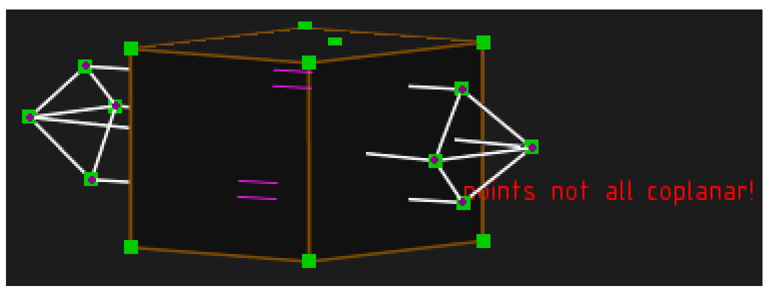
If, when you want to end the curve, you left-click on the starting point (instead of right-clicking), a periodic closed curve is created. Note that you must create 2 intermediate on-curve points (left-click) before doing so. You can constrain the start and end points of an open Bezier curve to be coincident after completion to make it a closed contour, also, but it won't be periodic.

Workplanes

When in 2D mode, some workplane must be active. Hitting **W** will align and center the view with that workplane.

When in 3D mode, hitting **W** will align the view with the nearest coordinate plane (same as **F2**), but not centered at origin. If, when in 3D mode, a group is active that has no default workplane and doesn't contain a workplane (e.g. an extrusion), you must select a workplane to switch to 2D mode.

SolveSpace remembers the last workplane that was active in each group (if there was one), and if you were in 2D or 3D mode. Make a group active, and those settings are recalled. Hit **W**, and if in 2D, the view will be rotated to that workplane. *(I told you above it's complicated!)*



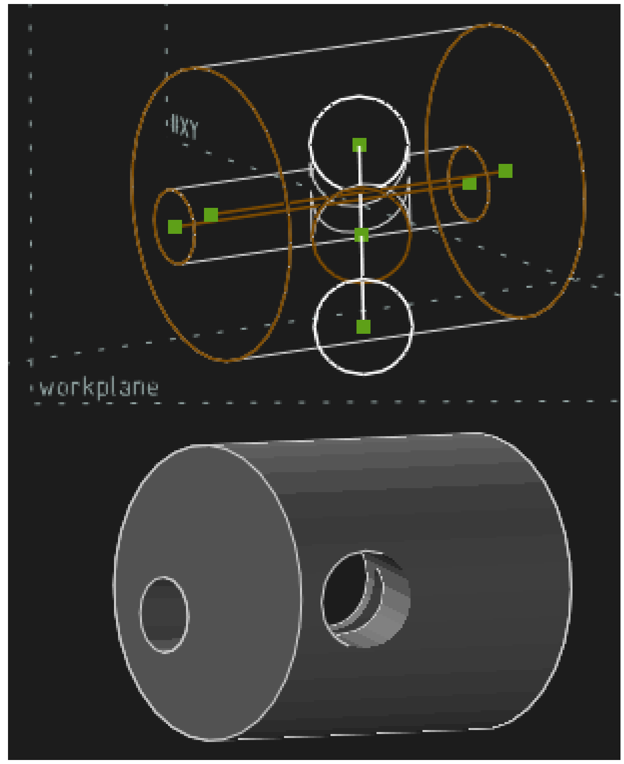
To construct complex 3D objects, it's best to create separate objects in separate workplanes, then extrude the objects to merge them (rather than drawing freehand in 3D). You can draw meshes with freehand 3D contours but it's not easy. And SolveSpace won't draw shaded surfaces for them, whether with coplanar points or not.

Intermediate

Workplanes

To create a new workplane (coincident with an XYZ plane):

1. To center it on a point other than the origin of the XYZ planes, select that point first. Otherwise, select the origin.
2. Orient your view close to orthogonal to the plane you want the new workplane to parallel
3. Click **New Group in New Workplane (Shift-W)**.
4. SolveSpace snaps the new workplane parallel to the one closeset to your view, snaps your view to it, and makes it active.



To create a new workplane not parallel to any XYZ coordinate plane (based on an existing 3D object):

1. Select an origin point.
2. Select 2 non-parallel coplanar lines that can define the plane to be created.
3. Click **New Group in New Workplane (Shift-W)**.
4. SolveSpace snaps the view to the new workplane.

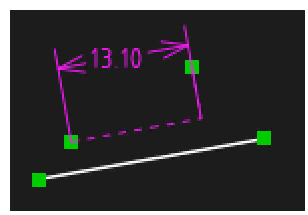
To create a new workplane not aligned to any existing drawing nor axis (oblique):

1. Click **New Group In 3D (Shift 3)**.
2. Choose **Sketch > Workplane**.
3. Orient the view so the screen is coplanar with the desired workplane.
4. Click where the origin needs to be.

- Note: if you don't do the 1st step, the new workplane, and entities you draw next (even if sketching in 2D on the new workplane), will belong to whatever group is active--a new group isn't created. This is actually usable for skewed extrusions!

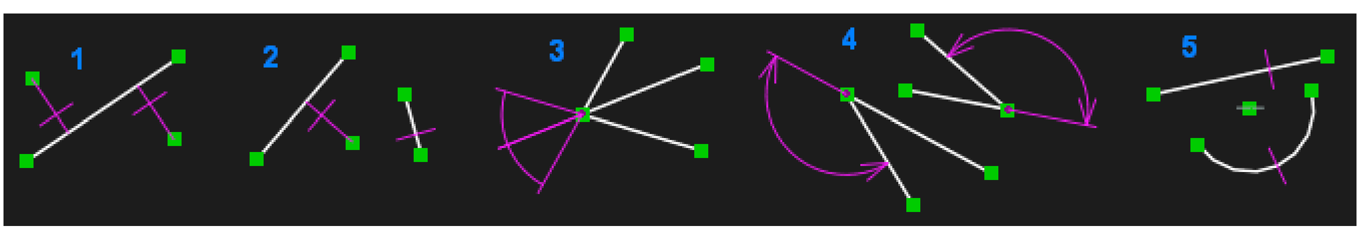
The image at right shows a difference-extrusion solid made with an oblique workplane. The 1st cylinder, with a longitudinal shaft through it, was extruded in Z from 2 concentric circles in XY. Then another cylinder was extruded from another circle in a new, oblique workplane to punch a new, slanted shaft through the first cylinder and intersect its shaft.

Constraints



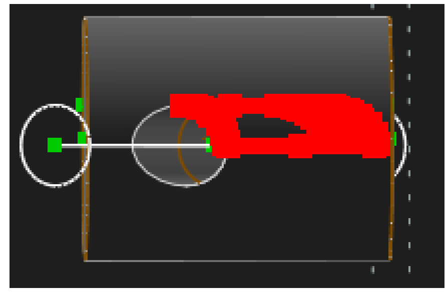
The **Distance** constraint (**D**) can operate on a projected distance. In the example image at right, the distance between 2 points is fixed, as projected onto the line. Select 2 points and a line or normal, then **D**.

The **Equal Length/Radius/Angle** constraint (**Q**) can operate on 3 or 4 entities as well as 2, and on arcs. In the image below, there is: (1) congruent point distances to a line, (2) a point-line distance congruent to a line length, (3) congruent angles among 3 lines, (4) congruent angles among 4 lines, and (5) congruent line and arc lengths.



If points of a shape are dragged too quickly, the solver may change the shape of contour while checking constraints. Try dragging slower, or moving it in increments (or using **Analyze > Step Dimension** to increment it automatically).

Extrusions

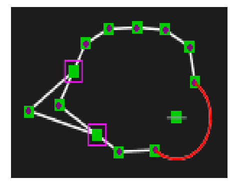


You'll notice that an extrusion always extrudes the origin into a line. The tutorials state that having them may be useful for creating constraints.

Sometimes you may get red lines in part of the intersection of multiple extrusions, like in the image here. Use **Analyze > Show Interfering Parts (Ctrl+Shift+I)**, and if the errors are false, they'll disappear and you'll be told it doesn't interfere.

If the errors keep showing up every time you activate an extrusion group, you can set its **force NURBS surfaces to triangle mesh** option in **Property Browser** to prevent them. This tells the solver to use the finite-lines approximation (mesh) of the shape instead of its exact NURBS mathematical definition (and therefore be more forgiving).

Selecting & Styles



Use **Select > Edge Chain** to select all contour segments connected, in a series, to the selected one through endpoints. This is useful for applying line styles. Doing this in the example image at right will select the entire contour, except for the 2 lines on the left which are connected through point-on-line constraints.

Styles

The attributes of lines used to draw entities (including text) are set with line styles. Use **Right-Click > Assign to Style** on a selected line/curve to apply a custom style to it (or remove one with **No Style**). You can create a new style in the process. Details are shown in the **Property Browser**. Use **Line Styles** there also to view, change or create styles. You can assign (or remove) a custom style to multiple entities by entering its page in the **Property Browser**, then clicking on the entities, then clicking **assign to style ...**. Custom styles are saved in the particular model file.

Default styles have **#def** in the name. Changes you make to them are saved and used for all files on that computer.

If the width of the line or text of a style is set in pixels, its size relative to the model changes upon zoom in the drawing window (and for export). If set in millimeters or inches, it doesn't. All default styles are specified in pixels.

Exporting Drawings & Models

A drawing can be saved as an image, PDF or object file. There are separate chord tolerance settings, and a scale factor setting, for exporting meshes and some 2D Views and Sections. The **Reference** document has more detail.

File > Export Bitmap saves a screenshot of the current view you see (except for the tools).

File > Export 2D View creates a PDF with a line drawing of what you currently see, including constraints.

- If the shapes in the PDF are darkly filled in, turn off **Export Shaded 2D Triangles** in the configuration if you want just their outlines.

- To make constraints smaller relative to the shapes, and the lines thinner, in the PDF, zoom in on your drawing before exporting.
- Whatever elements are enabled with the visibility iconbar (in **Property Browser**) will be output.

File > Export 2D Section should create a PDF with a line drawing of a section of the model at the workplane you choose. (But it's not a true section tool--it only outputs lines from the solid in that workplane. In this author's experience, SolveSpace will generate a blank PDF when asked for a 2D Section through a solid.)

File > Export Wireframe, or **Triangle Mesh**, or **Surfaces** will export 3D models as either a wireframe or solid with surfaces. The wireframe can be STL or DXF. The solid options are STL, OBJ or Three.js for meshes or NURBS STEP file for surfaces.

As pointed out in the 2D Sketching Tutorial, a way to get take-off dimensions on a 2D export of a model with constraints that clutter the view is to add a layer for them:

1. Create a **New Group in New Workplane (Shift-W)**, parallel to the desired view workplane.
2. Enable **treat all dimensions as reference** for the new group.
3. Create distance constraints for needed dimension lines in the new group.
4. Hide constraints in the drawing groups and export.

More Construction Concepts #2

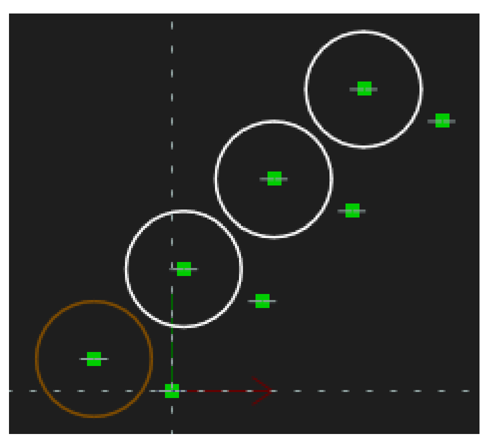
Construction Geometry

This type of geometry is in the drawing but isn't intended to be part of the final product, and isn't exported in the CAM data. They're useful as axes of symmetry or other operations, or guides to help you make actual drawing parts.

To create one, draw a regular contour, and use **Toggle Construction (G)**. Its color will change to green. These are also created automatically when you do certain operations. If you draw in a workplane not associated with a group, the shapes will be construction geometry by default.

Stepping Copies

Translating

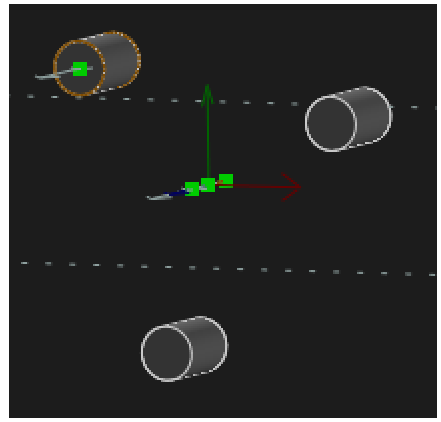


To create copies of the drawing in the active group, that are evenly spaced, use **New Group > Step Translating** (**Shift-T**). In the **Property Browser**, change the number of instances (**translate original sketch : repeat**) and whether the original is in the middle (**2-sided**) or on the end (**1-sided**). If 1 sided, you can choose if the stepping starts with the original or not. The stepped circle in the image at right is set to one-sided and repeat 3 times, starting with copy #1.

You can then move the 1st copy relative to the original, and any additional copies will move also, with that same displacement magnified proportionally to their copy number. (So the 2nd copy will move twice as far as you move the 1st, the 3rd 3 times as far, etc.--try it to see it.) You can move any copy to move all of them. And if you move or change the original, the copies will move or change with it.

If you constrained the original before stepping, the constraints won't necessarily be copied. If a workplane was active, the copies will be constrained to it. Otherwise they'll be free in space (but still along their copy line). Even if in 3D, their orientation still stays the same as the original's. And, most importantly, you can't extrude copies stepped in 3D.

Rotating



This makes copies of the active group spaced circularly around a chosen axis. That axis can be chosen by selecting a point it will pass through and a direction it will be parallel to, such as a line or normal. If a workplane is active, you can just choose a point and the axis will be through that, parallel to the workplane's normal.

The options work the same as for translating. The movement of the copies, though, is not linear, but evenly spaced along a circle on which the original lies, centered on the chosen axis of rotation. In this example image, the cylinders are a rotation around the Z axis normal, with 3 repeats starting with the original.

Notice that, with both translating and rotating, the spacing of the copies can be made exact with a constraint (distance or angle, respectively).

Area & Volume Measurement

In the **Analysis** menu are commands to measure the areas of contours and volumes of solids. They operate on the entire active group. If the drawing contains any curves, they're approximated with piecewise lines or triangles. So the measurement is approximate. The error can be reduced by setting a lower chord tolerance for curves in the configuration (in **Property Browser**).

That's the end of basic-level content!

Intermediate

Assemblies

Typically, these are aggregates in 1 file of existing parts from multiple files, to make a complex object. Any saved SolveSpace file can be linked-in, including other assemblies. Parts can be edited in their file and the edit will be reflected in the assembly. It's best to put an assembly and all its part files in one directory for easy file management.

When making individual parts for assemblies, each one that must be individually placeable in an assembly must be in its own file. (Except parts that get copied multiple times in the assembly--explained below).

Start a new file for the assembly, and link-in the 1st part with **New Group > Link / Assemble (Shift-I)**.

> LIMITATION ALERT (v2.3): All points that you want to work with in the assembly must be visible in the individual part file, by enabling **Shown** for their group(s) in the **Property Browser** (and saving).

A linked-in part can be moved thusly:

- translated around by **left-click-dragging** any of its edges or points
- tumbled around one of its points with **Shift-left-click**
- tumbled around its origins with **Shift-left-click** on one of its normals
- rotated in the screen plane around one of its points with **Ctrl-left-click** on that point
- rotated in the screen plane around its origin with **Ctrl-left-click** on one of its normals

As with regular drawings, only with constraints can objects be placed exactly. Also, it's best to use constraints so that if you want to modify parts in their individual files, those modifications will propagate correctly in the assembly. For example, you wouldn't want to constrain a linked-in bar using a point on the end of it if the bar's length might change later.

Placing the 1st part is easy: select one of its normals, and one of the coordinate axes, and use a **Same Orientation** constraint. As noted in the constraint list above, this keeps it pointed in the same direction as the coordinate axis and keeps it from rotating. Then fix its translation in space. Do this with a **Point-On-Point** constraint (**O**) between one of its points and the origin, or a **Locked Where Dragged** constraint (**]**), for example.

> Tip: Assemblies are easier if all parts are built in their files to be oriented relative to the same coordinate plane. That way, when they're all linked into a drawing, selecting the normals of each of them and 1 coordinate axis, and applying a Same Orientation constraint, will make them all orient identically.

To place remaining parts, it's often easiest to use a combination of same orientation, point coincident, and point-on-plane constraints. Point-on-plane constraints make it easy to leave a part unconstrained in 1 direction when needing to move parts for an exploded view of the assembly.

If you change the design of a linked-in part in its own file (the only place it can be edited), the assembly can then be re-calculated with **Edit > Regenerate All (Spacebar)**.

Stepping To Place Copied Parts

If you have multiple instances of the same part in an assembly, and they're not to be evenly spaced, link-in the part multiple times.

But if they're to be evenly spaced, it's easiest to link-in and place/constrain the 1st one, then use **New Group > Step Translating (Shift-T)** to copy it. They'll all have the same orientation restriction if you constrained the orientation of the 1st one before stepping. Then they just need to be translated.

Because each copy moves proportionally with their copy number, relative to the 1st copy's displacement from the original, you just have to place/constrain the 1st copy and the others will be placed properly.

Checking Interference

Use **Analyze > Show Interfering Parts (Ctrl-Shift-I)** to have red lines shown to indicate overlapping conflicts between parts (operates on triangle meshes of parts, and thus may show false positives for round parts).

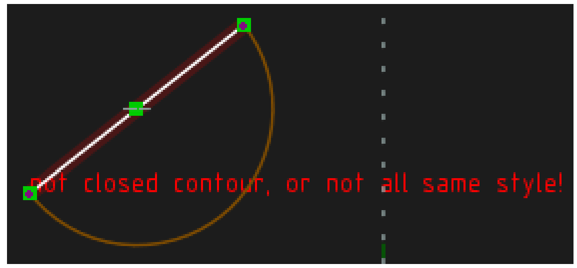
Advanced

The Assembly Mechanism

An assembly is not a special file. Parts that are linked-in for this are merely treated specially by the program. Linking in a part creates it as a new group, with an **assemble** attribute. You can see this in the **solid model as** section (next to **union** and **difference**) in the **Property Browser**. Note the distinction from **union** and **difference**, as assembled parts, unlike extrusions, are intended to not interfere with others.

Because any saved SolveSpace file can be linked in, mere 2D drawings, and even open contours, can be assembled into other files.

The Reference states that an open contour can be linked into a file, and a new line or curve drawn in the same or a different group as the contour to close it. This is partly incorrect in the author's experience. Such drawings in <u>other</u> groups do not close the contour. However, apparently because a linked-in open contour is an assemble group, an open-contour error never shows for it. And you <u>can</u> close it with new lines/curves <u>in</u> its group. SolveSpace will then fill it in with the closed-curve color.



Joining a line in a new group to a linked-in arc doesn't close the arc (nor the line).

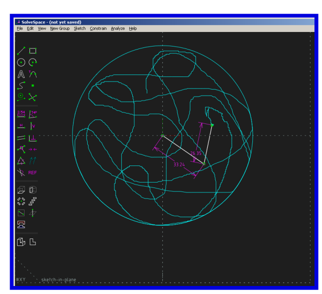
Also, if you link-in a 2D contour, you can extrude the assemble group just as if it was local to the file. If it's closed, the extrusion will show shaded surfaces. And if you close a linked-in, open 2D contour <u>within the assemble group</u>, and then extrude, it will be solid.

What all this means is that drawing re-use can be as extensive as you need it to be. You can link-in a 2D contour, extrude it and save that file, then link-in that to yet another assembly. Combine original and intermediate part files as necessary, and changes propagate throughout your design.

Advanced

Linkages

These are combinations of at least 2 parts (links) that are constrained to move relative to each other somehow. For example, 2 bars joined together on-end (with a point-on-point constraint), with 1 bar's unjointed end anchored at the origin (with another point-on-point constrain), will allow the free end of the other bar to be anywhere within a circle with radius equal to the bars's length sum (well, anywhere except the circle defined by L1-L2).

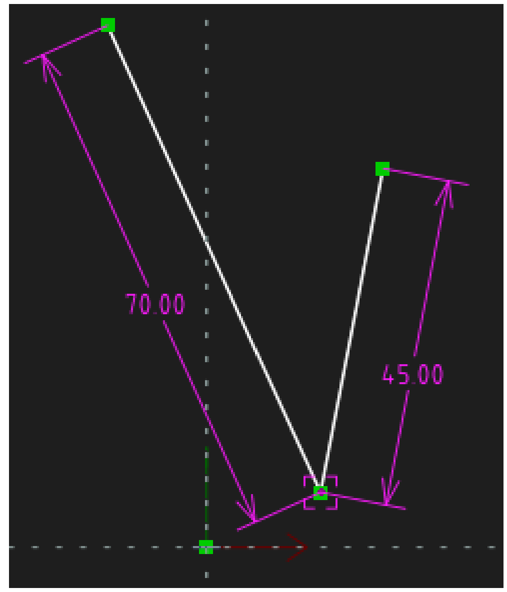


The particular usefulness of the SolveSpace linkage solver is that the parts of a 3D model (designed to move) can be checked for interference as the "skeleton" defined by their linkages is actuated through its range of motion, and have that motion recorded.

To do that, you draw the linkage as a contour sketch, with constraints defining the relations between its members, and then link in 3D parts as an assembly and connect them to the linkage with more constraints.

Linkage Terms & Constraints To Use:

- Bar link: a line with a length constraint
- Truss or Lever link: a line or 3+ sided polygon with more than 2 joints
- Pin joint: a point-coincident constraint (on 2 points of lines or shapes in a 2D workplane)
- Ball joint: a point-coincident constraint (on 2 lines or shapes in 3D)
- Sliding pin joint: a point-on-line constraint



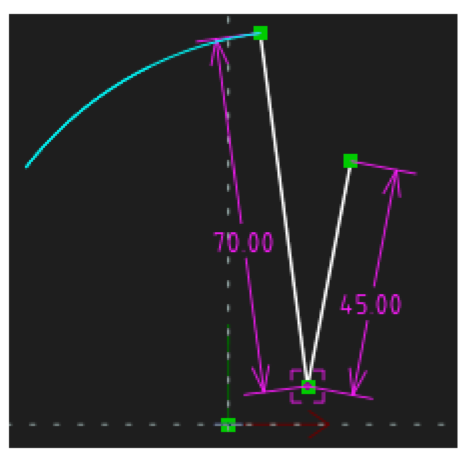
Note that, as in real life, a shape can have an extended width beyond just being a simple bar. For example, it can have 3 joints and act as a truss or bellcrank that offsets or redirects motion. Or, put 3 joints on a simple bar, with the middle one fixed in space or on a point, to represent a lever. A sliding pin joint can represent a pin in a slot or a bar or wheel in a track.

To create linkages, draw lines or polygons to represent either the parts you want, such as bars, or the force-carrying members or vectors of parts, such as the 3 (essential) legs of a bellcrank. It's helpful to fix the size of 1 part before creating the rest so the drawing is sized roughly correctly as you create it. Then, add dimension constraints to fix the size of all parts and members, as needed. Next, add other constraints to relate the parts to each other and to their position in space: e.g., point-coincident constraints for pin joints, and horizontal, vertical or distance constraints for parts fixed relative to the environment outside the mechanism. The image here shows a simple 2-bar linkage, with a locked-where-dragged constraint on the joint. A solid model for it is shown later.

> LIMITATION ALERT (v2.3): Note that you can't place a point-on-curve constraint on a Bezier curve. So a cam can't be created with just a contour. (An elliptical cam can be created with a constrained line simulating a trammel, as seen in Tutorial 4, though.)

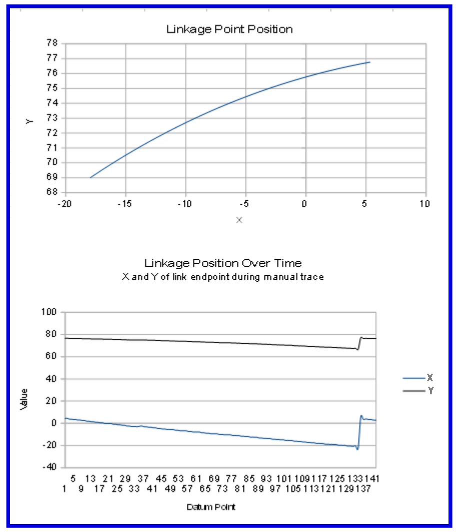
Many linkages will get reduced to 1 DOF, which represents the changeable parameter of the driving link, such as the rotation of an arm or gear. (DOFs are shown in **Property Browser** when the group you're drawing is clicked.)

Testing



You can test the linkage manually by selecting a point on the linkage that is free and moving it. To create a trace of the position of the point as it's moved, use **Analyze > Trace Point** after selecting it, then move it. The trace will show on screen in cyan. Choose **Analyze > Stop Tracing...** and a prompt for saving the position data in a CSV file will show. If saved, it can be graphed in a spreadsheet or analysis program. Note that the manual movement is irreguar (because you can't keep a perfectly constant speed as you move the point with a pointing device). In the image at right, the top point was traced while rotating its link arm back and forth.

To have SolveSpace test the linkage automatically with a regular movement (constant linear or angular velocity), place a constraint on the point or part that has or gives rise to the 1 DOF. This would be an angle constraint for a rotating bar or maybe a distance constraint for a sliding point. Then choose **Analyze > Step Dimension**. In the **Property Browser**, set the ending constraint value and the number of steps to be used in the movement (more steps = slower speed & more data points). Then click **step dimension now** to see it move. (If you didn't see anything, it was too fast. Increase the steps and try again.)



Plots of a trace on the linkage above are shown at right. The top graph is the link endpoint position during an automatically-stepped trace. The bottom graph is separate plots of X and Y values over time as the endpoint was moved in a manual trace--the peak at the end is where I happened to bump the mouse. If it was a stepped trace, the lines would be smooth. Also, in a stepped trace, because the step intervals are all equal, the plots would show position values as a function of the independent value of the stepped dimension--that would be angle of the link arm in this example.

To save the position tracing of a point with regular intervals as it's stepped automatically:

1. select the point
2. choose **Analyze > Trace Point**
3. select the constraint to be stepped through a range
4. choose **Analyze > Step Dimension**
5. set the stepping parameters in the **Property Browser** (many steps to get many data points)
6. click **step dimension now** in the **Property Browser**
7. choose **Analyze > Stop Tracing...** and save the output to a file

Errors

As you move a linkage manually, the solver is constantly checking it. Sometimes it will detect an error and turn the drawing background red. This may be due to a calculation error in the solver or just because you moved the linkage too fast--try moving it slower. If the problem persists it inidcates a linkage problem, such as binding (where a real-world linkage would get stuck). Use **Undo** and redesign.

Linkages & Solid Models

When the linkage is finished, a solid model of the actual mechanism it represents can be built around and connected to it. With the linkage file open, use **Link / Assemble (Shift+I)** to import each of the parts. The key action is making the connection, which comprises constraints such that the linkage and model move together exactly.

To do this for a planar linkage, for example, you must, for each part:

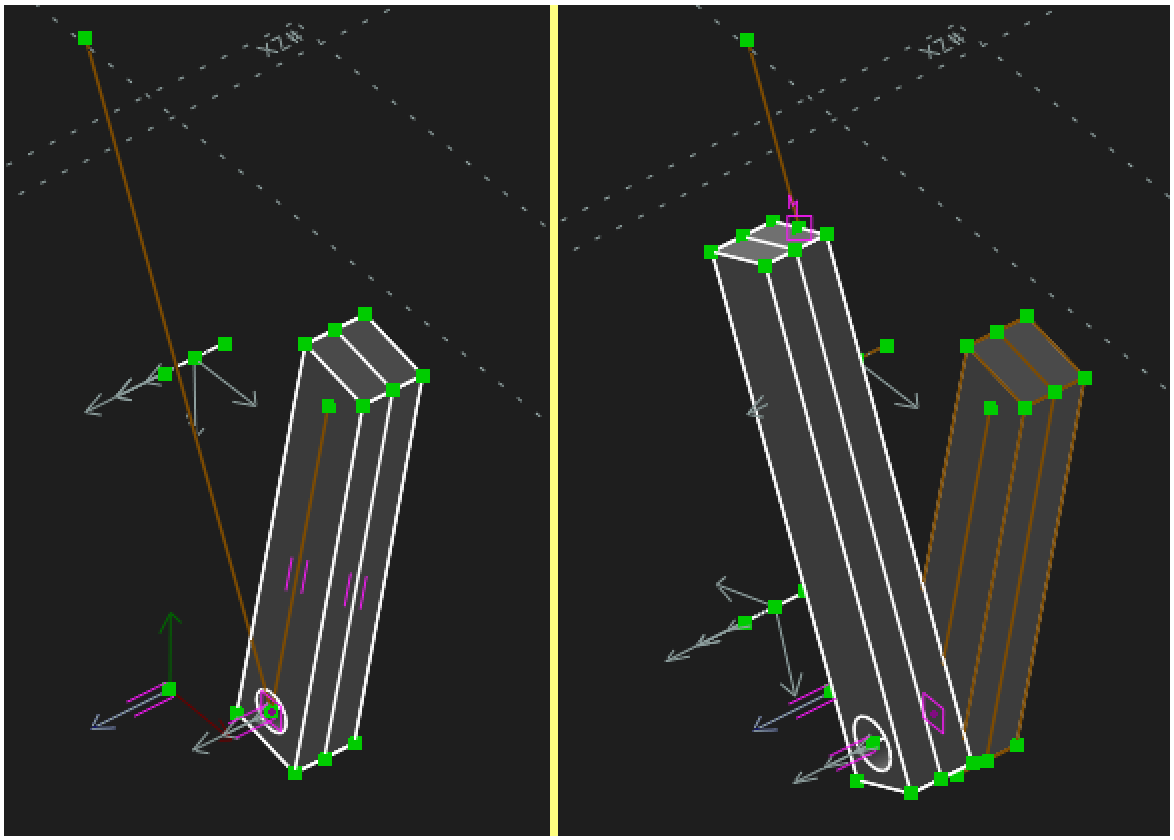
1. fix its orientation in space, with a **Parallel** constraint (**L**) between its normal and a linkage workplane normal or coordinate axis parallel with the linkage normal

2. fix its translation relative to the workplane in step 1, or to a workplane of another model part, with **Point-on-plane (O)**

3. fix its translation in a workplane relative to a linkage, with, e.g., a **Point-On-Point (O)** constraint between a part point and a linkage joint point (First, select the linkage workplane or one parallel to it, and **Sketch > In Workplane (2)**.)

4. restrict the part to move with the linkage, with a **Parallel** constraint between lines on the model and linkage, or with another **Point-On-Point** constraint between another pair of points on the part and linkage

The image below shows an example. On the left, a rectangular bar was linked in to the 2-arm linkage in the examples above. It's normal was constrained parallel to the Z axis. Then the center of its hole (on the near face) was constrained to the linkage workplane. That same point was then constrained coincident with the joint between the 2 linkage lines (which are brown). Finally, one of its edges was constrained parallel to a linkage line to make it move with that line.



Next, shown on the right, a 2nd bar was linked in. The same first 3 constraints were created on it (but using the hole center on its back face in this image). To connect its movement with the other linkage line, a point-on-line constraint was made between a midpoint on its top-back edge and the line (indicated with a purple "M" and square). The linkage group could then be activated and moved to test both it and the model.

Remember, to make it easier to see needed points and edges in complicated models when creating constraints, various groups (parts) can be un-shown in the **Property Browser** when not needed. You'll still be able to see the parts, though, if one or more of the model options is enabled in the iconbar in the **Property Browser**, such as **Show Shaded View** or **Show Edges**.

Also, if your model (and its view) makes it hard to see the points you use to move the linkage & model, you can add a line or shape to extend the point's driven link. You can give the extension a custom line style, and set that line style to not export so it won't show in exported files or drawings.

To analyze a linkage with the solid model connected to it (and showing), with a step dimension, you have to have (or create) a constraint in the linkage to step (as you did in the ***Testing*** section above). But because the last part group must be active for all part groups to be shown (and the linkage is likely in group g002), you can't select that constraint in the drawing window in order to step it. So, click on the group for the linkage in **Property Browser**, and select it there. Then use **Analyze > Step Dimension** to actuate the mechanism stepwise.

To see a point trace while stepping a dimension with the model showing (a part group active):

1. Set the linkage group to shown (if it isn't already), activate it, select the point to trace, choose **Analyze > Trace Point.**

2. Set the last part group to active. Set the linkage group and any parts to not show as desired.

3. Proceed with the **Step Dimension** procedure in the previous paragraph. You'll see the model move and the cyan point trace drawn.

4. Use **Analyze > Stop Tracing** to stop the trace and save the data if desired.

   LIMITATION ALERT (v2.3): The author has noticed that the Dimension Stepper doesn't seem to like to go beyond 180° for (at least some) angle constraints.

NB: Any split infinitives are deliberate.