

## Working With Developable Surfaces

By

Christopher D. Barry, P.E.

Developable surfaces are especially important to the home boatbuilder because they are often working with sheet materials like plywood, steel or aluminum. Developable surfaces can be formed from flat sheets without stretching, so the forces required to form sheet materials into developable surfaces are much less than for other surfaces. In some cases, particularly with plywood, the forces required to form non-developable surfaces could be so large that the material is damaged internally when it is formed. Another advantage of developable surfaces is that the development, or flattened out shape, of such a surface is exact. When other types of surfaces are expanded (note the difference in terms - "expansion" is flattening out a non-developable surface) the shape of the expansion depends on the distortion field applied to form it, so that there is no single exact expansion without detailed forming information. If you are either designing or lofting plywood or metal hulls you should understand developable surfaces and the methods for working with them.

A developable surface is mathematically defined as a surface with "zero Gaussian curvature", which means that the largest curvature times the smallest (these are the "principal curvatures") is zero. This of course requires that one of the two principal curvatures be zero. The surface must contain straight lines (which have zero curvature), but it requires more than that - the straight lines must be the smallest ("least principal") curvature on the surface.

A simple experiment helps to understand this. Take three or four rubber bands and stretch them between two pencils, spaced along the length. With the pencils horizontal, pull them apart to tension the rubber bands. The rubber bands and the pencils are all straight lines, and they form a plane. If you twist the pencils so that one point is higher than the other, the pencils and the rubber bands are still straight, but you had to stretch the rubber bands near the ends of the pencils. Despite the fact that the surface contains straight lines everywhere, it can't be formed from a plane without stretching the material.

This is because the largest curve is the curve concave up running from the high end of one pencil to the high end of the other (let's call this positive curvature). However, the smallest curvature is not one of the straight lines, but rather the concave down curve from the low end of one pencil to the low end of the other. Since this curve cups in the opposite direction to the concave up one, its sign is opposite, so it's negative and hence the smallest. When the two curvatures are multiplied, the Gaussian curvature is negative, not zero. (Figure 1)

This is an important distinction, which often causes problems for builders and designers. It is not enough for a surface to have straight lines (or rulings); the rulings must be the smallest curves. The most common problem for builders is when a hull has straight frames that are not parallel. These forms are not developable. They will be hard to plate in metal, and if covered in plywood, the shear stress induced in the plywood may cause "rolling shear failure" in the inner plies of the wood and hence early structural failure. Such a surface is a "warped surface", with negative Gaussian curvature. In general, surfaces with rulings ("ruled surfaces") between two curves in space are warped. In fact, there is at most one developable surface between any two curves and there may be none. There are, however, infinitely many warped surfaces, each warped in a different way.

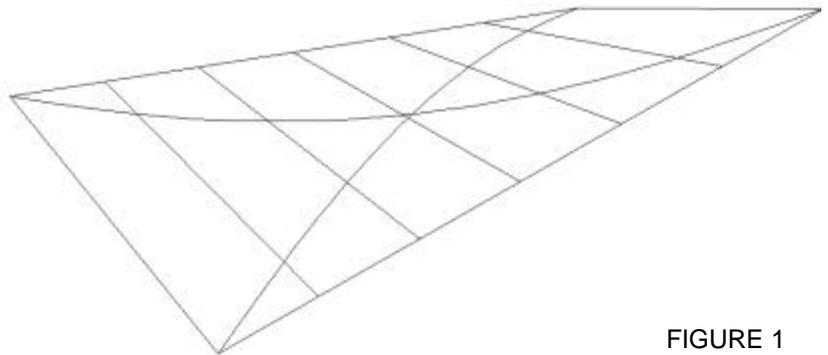


FIGURE 1

Figure 2 shows two edges, specifically co-axial identical circles.

On the left is the only developable surface between these two edges, a right circular cylinder. On the right is a nondevelopable surface on the same circles created by warped rulings. To build a developable hull we have to find the one developable surface *if it exists*. A developable surface is composed of an infinitely closely spaced series of straight lines, the rulings, with a small fold at each ruling.

The small folds form a smooth curve (Figure 3).

Let's experiment with this. Take a piece of paper and make three or four folds in it along lines that run sort of in the same direction, but don't cross. These are the rulings. Put the paper down on the table. Notice that the piece of the paper lies on two folds and the paper between is flat. This is the practical test for developability: any two adjacent rulings must be (nearly) coplanar. Look again at Figure 1 or at the rubber bands in our pencil experiment and you can see that no adjacent rulings in the warped cases are coplanar. (In a practical sense, two rulings may not be exactly coplanar because any two we pick aren't really exactly adjacent - there will always be other rulings between them, since there are infinitely many rulings in the surface, but the warp angle should be very small.)

We can calculate the warp angle between any two rulings with vector arithmetic, and of course this is what computer programs use. (For those interested, take the cross product of the ruling and the tangent at one edge, then the cross product of the ruling and the tangent at the other edge. The warp angle is the cross product of the two previous cross product divided by their dot product.) However, proving co-planarity doesn't require calculations. There are only two ways two lines in space can be coplanar. Either they are parallel or they intersect. The parallel case is obvious - parallel lines are coplanar by definition. The frames of classic dory form boats are parallel straight lines, so the side and bottom of the boat are portions of tilted cylinders. (Again, it is important to remember that straight stations or frames are only developable if they are also parallel.) The intersecting case is only slightly more complicated - since three points define a plane, two intersecting lines must be coplanar.

The mutual plane is defined by any point on one ruling, the intersection of the two and any other point on the other ruling.

This definition gives us the most common way of designing a developable surface, the multi-conic method. This is also known as Rabl's method, after Sam Rabl, designer of the Titmouse and author of many books on home boatbuilding and lofting. He probably did not invent the method and describes it as widely used in ship and aircraft lofting, but as he was responsible for its widespread use.

The original "conic" method uses a single intersection point for each surface. First draw the plan and side views of the two bounding curved edges of the surface but leave one view of one edge undefined. For example, for the bottom of a vee bottom hull, draw the keel line in profile and the chine in plan. (Unfortunately, "chine" is conventionally used both to mean any edge of a developable surface and also to mean the joint between the bottom and the side of a boat. I will use "chine" in the latter case and "edge" in the former.) The keel is already defined in plan view as well as profile since it is straight. Select a point well away from the hull surface. Draw a straight ruling (in both plan and side view) from this point through the keel to the chine. By projection, the profile view of the intersection of the line and the chine can be found. This is of course a point on the chine.

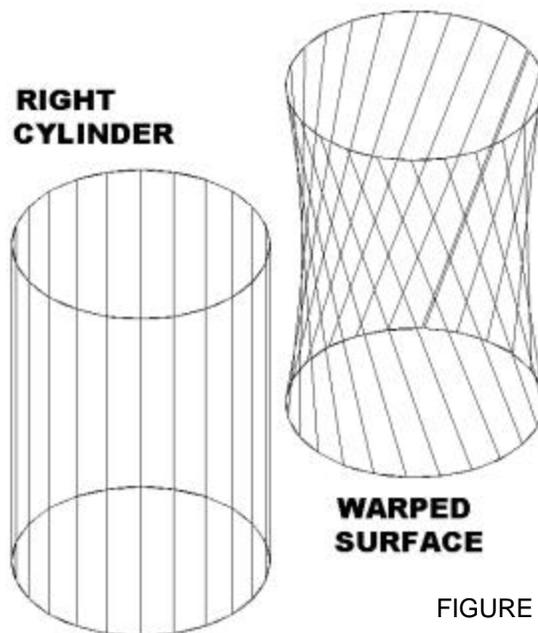


FIGURE 2

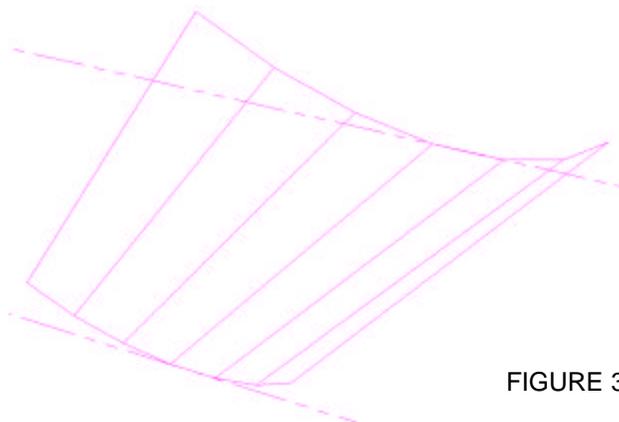


FIGURE 3

By drawing numerous straight lines radiating from the same point, the chine profile that produces a true developable surface is found (Figure 4).

Then, lay out lines for buttocks, waterlines and stations and define these curves by plotting how they intersect the rulings. The intersection of the rulings with the plan view of the buttock lines (which are straight in plan view) projected onto the profile gives the profile view of the buttocks. The intersection of the rulings with the profile view of the waterlines (which are straight in profile view) projected into the plan view gives the plan view of the waterlines. The intersection of the station or frame lines with the rulings gives the stations and frames.

It is worth noting that a surface produced with a single intersecting point and fair edges must be fair, so you only have to fair the chines and define stations, frames, waterlines or buttocks that are actually used to make parts. There is no need to define extra waterlines, buttocks or stations just to fair the hull.

However, the process strictly dictates the missing view of the edge, so you may find that the chosen intersection point produces a weird curve. In this case, you have to guess a new point and start over.

Rabl pointed out that any two adjacent lines only have to intersect only one another, not all the other lines. This still produces an adjacent pair of coplanar rulings. Thus, the first and second ruling would intersect at some point, but the third ruling would intersect the second at some other point. Users of this method sometimes lay out both edges in both views before running rulings. Strictly speaking this is wrong, (though it may work out) because there is no sure way to know beforehand that the rulings selected are actually a true developable of those edges, or if that there even exists any true developable. If they do not subsequently verify that each adjacent pair of rulings actually intersect at the same point in both views, the surface produced may not be developable. Even worse is laying the rulings out so they merely appear to fan nicely, but don't actually intersect correctly at all. Remember the rulings in plan view and in profile view are the **same lines**, just viewed from different angles. Two adjacent rulings have to have the same intersection in space in both views.

A surface produced on multiple intersections this way is not automatically fair, so you have to check for fairness by use of buttocks, waterlines and stations in the normal way. You may also find that your assumed pattern of rulings produces weird looking edges, buttocks, waterlines or sections. Again this requires starting over with a new set of assumed rulings.

If you have a CAD system capable of 3D drawing, then either of these methods is very easy. Just draw the edges, defining points and rulings using a single model in 3D.

If we are given a pair of edges already defined, then we need to find the true rulings of the developable surface (if it exists). It is possible find the rulings mathematically, especially with computer aid, but there is also a graphical method that can be used on the drafting board or the lofting floor.

This is called "Kilgore's Method" after the late Ullman Kilgore, a naval architect with the University of Michigan who described it in 1967 (Figure 5):

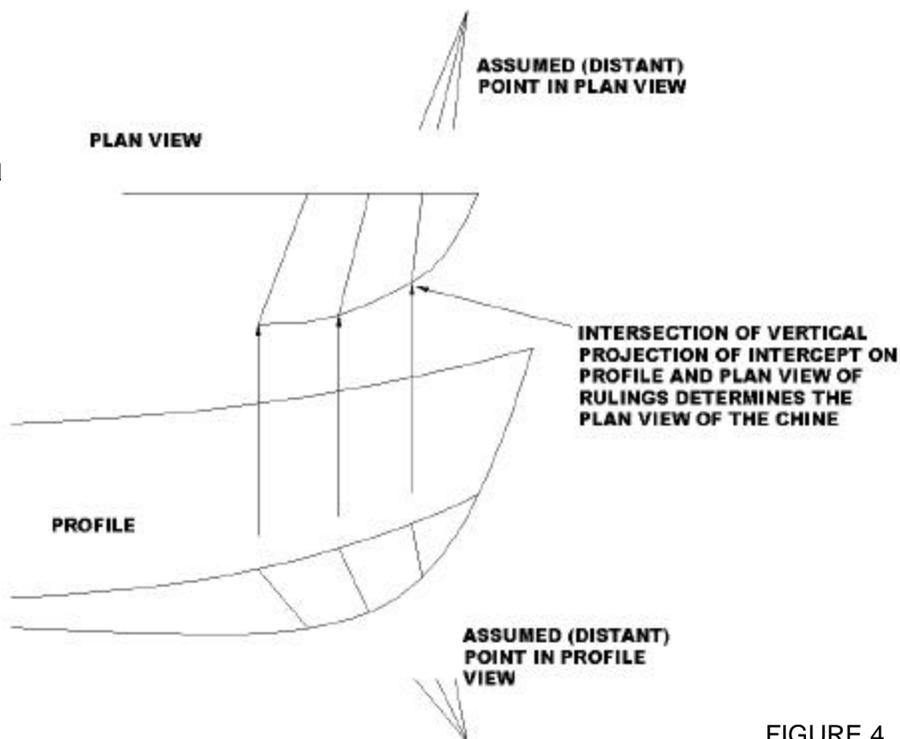


FIGURE 4

Select an originating view, in this case, the side elevation.

1. Pick a point **P** on one edge, in this case the chine.
2. Draw local tangent **a-a** to the edge at P and project it on both views.
3. Draw line **b-b** in the both views from some point along line **a-a**, and transfer the intersection of the line and the other edge (the deck edge in this case) from the other view back to the originating view.
4. Draw lines **c-c** and **d-d** parallel to **b-b** in both views.
5. Transfer the intersections of the edge in the other view to the originating view, to determine the endpoints of **c-c** and **d-d**.
6. Pass an arc through the points in the originating view.
7. The midpoint (point **Q**) between intersections of the arc and the other edge (point **E** and **F**) is tangent to **P** on the other edge so the developable ruling at this point is **P-Q**, which can be transferred up to the other view.

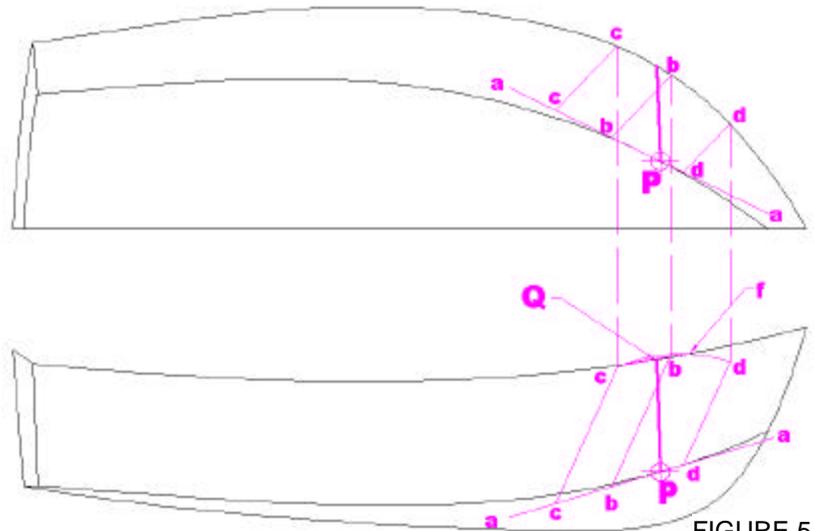


FIGURE 5

Why does this work?

- The parallel lines **b-b**, **c-c** and **d-d** form a tangent plane to the originating edge. This is the plane that the assumed ruling is in.
- The arc is where a cylinder formed by the other edge (the sheer) passes through the plane.
- **E** and **F** are where the tangent plane **b-b**, **c-c**, **d-d** cuts the cylinder.
- By the “mean value theorem” the midpoint **Q** is parallel to the tangent plane, and is on the other edge, so it is coplanar with **a-a**, and therefore **P-Q** is a developable ruling. We are basically tilting the assumed tangent plane to the lower edge at **P** (**b-b**, **c-c**, **d-d**) around **a-a** until it touches the tangent at the upper edge.

This process is then repeated several times for each surface, and the waterlines, sections and buttocks can be determined from the rulings just as in either of Rabl’s methods. It is worth extending these rulings to find their intersection points as in Rabl’s method. This is a check and helps to find new rulings to guess.

The basic concept of this method can be implemented very simply in a 3D CAD system. In this case simply assume a ruling, set a viewpoint such that point P is the origin, the edge at point P is running on the x axis and the ruling in the z axis. Then just inspect at the other edge and see (by eye) if it is truly parallel to the x-axis. If it is, then the assumed ruling is correct, otherwise, it is obvious, again just by looking, where the right end of the ruling should probably be. Then assume this ruling and repeat to check. In AutoCAD, assuming a ruling has already been selected from point P to the other edge, the commands (with keystrokes shown as **BOLD** and comments in *italics*) are:

**UCS**, **3Point**, *now pick the ruling **END** at Point P as the origin, the other **END** as the point on the x axis and a **NEAR** point on the edge as the point in the plus Y plane. This sets a User Coordinate System in a plane containing the ruling and tangent to the edge at P. (At this point, if you want, you can measure the exact warp angle by taking the **DISTANCE** from the **END** of the ruling on the other edge and a **NEAR** point on it, however, this is not required.)*

*Then flip the **UCS** around the **Y** axis by **90** degrees and set the view as **PLAN** to the current **UCS** (just hit return here as current **UCS** is the default. Now you are looking right down the ruling, and you should be able to see where the other edge is tangent to it because it will be parallel to the **UCS** x-axis. The two edges should overlay each other. (It helps if they are two different colors.)*

*If the other edge isn’t parallel, draw a new **LINE** from **0,0** to the **NEAR** point on the other edge that looks parallel to the x-axis, **ERASE** the first assumed ruling and start over with the new assumed ruling. Not counting the optional warp angle measurement this is only 39 keystrokes and much less using running **OSNAPS**, aliases set in the “pigpen” and a short AutoLisp command to alias **UCS X 90**. It takes me about ten seconds.*

Figure 6 shows the appearance of a true ruling (bottom) and an assumed ruling which was not correct (top). The top view also shows the correct ruling as a dark line. Just as in Rabl's method, once we have the rulings, we can complete the lofting in the normal way.

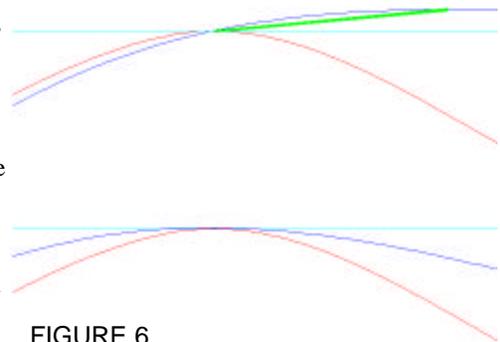


FIGURE 6

Obviously, it would really help the lofters to know the rulings intended by the designer. Unfortunately many designers remove the rulings from the drawings once they have the conventional sections, butts, and waterlines. I have seen a few drawings that the designer left the intersection points for a singly conic hull on the lines and actually dimensioned them, and I would like to encourage this. In fact, designers should not only leave on the rulings, but also tabulate them in an offset table the same way as they tabulate waterlines, butts and stations. It makes it much easier for the builder and preserves the designer's intent. If you are working without the designer's rulings though, don't despair.

You can find the intended rulings by selecting a point on an edge, running an assumed ruling from it through a point on a waterline or butt, and seeing whether it intersects the other edge. (Do this after fairing the edges, but before spending a lot of time cross fairing waterlines and butts and correcting offsets – it's easier to fair on the rulings.) If it does, it's a ruling, otherwise, guess again on a new point on the waterline or butt – the distance the ruling misses the edge, especially the second time, will help to improve your guess. Once you have a few rulings, check their intersections to make sure the surface is developable. The rulings may not intersect quite right, but this might just be due to the normal corrections required when lofting. You can then use one of Rabl's methods to correct the rulings, but by finding the designer's rulings, you will preserve his intended hull shape.

Once you have the rulings and chines, you can develop the surfaces. This is a bit tedious but basically simple. Given two adjacent rulings, a triangle is formed by one ruling, a line connecting one end of the ruling to the opposite end of the other, and a line connecting the first ruling to the same end of the other. The other ruling forms another triangle in the same way. Since the length of the three sides determines a triangle, we can find the developed shape of the surface by covering it with alternating triangles and laying out the lengths of the sides of all the triangles flat.

Figure 7 shows this: Dark line a-b is a ruling as is c-d. A-b, b-c, and c-a form one triangle. The other triangle is c-d, d-b and b-c. Determine the true three-dimensional lengths of each side. This can be done by projection methods described in most lofting texts, but in this day of cheap calculators it is easier to calculate. (Especially if you are working in metric units.) The distance between two points is simply square root of the sum of the difference in height squared, plus the difference in half-breadth squared, plus the difference in length location squared, which is, of course, the Pythagorean theorem. The flat shape (or development) of the surface is found by laying out one length, such as a-b then swinging arcs for the other lengths b-c, a-c and b-d. The intersection of arcs a-c and b-c gives point c, and an arc c-d is swung to get point d. Connecting the points gives the developed shape of the panel from a-b to c-d. The process is then repeated for the next ruling.

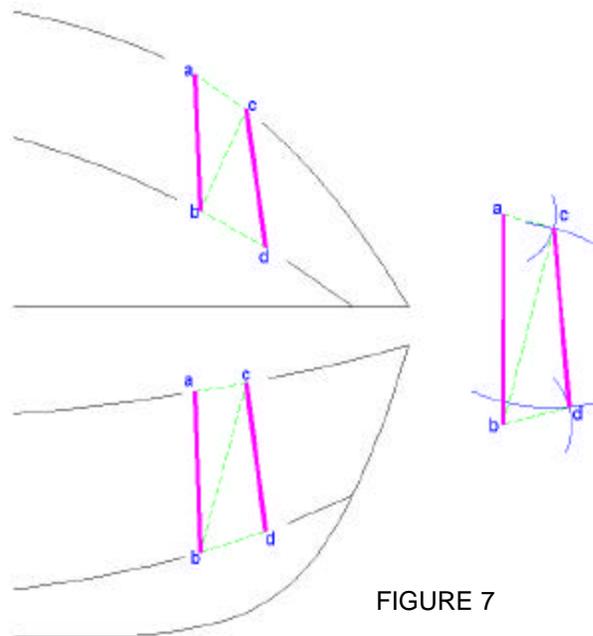


FIGURE 7

Of course, only the rulings are actually straight lines. A-c, b-d and b-c are all curved, and hence slightly longer than the straight line connecting them. It is possible to determine the approximate length along the curve by common lofting methods, but if enough rulings are used, then the curved lines are close enough to straight that the final result is close enough. How close, though, is close enough?

Figure 8 shows the increase in length along typical conic curve over the straight line connecting them as a function of the belly of the curve. This plot shows, for example, that if the straight line distance between two points was ten inches and the curve bellied one inch off the straight line (10% depth) it would be about 2.6% longer (10-1/4" total length) than the straight distance. For close rulings, the lines themselves are short, the belly is small and the additional distance is very small. However, errors could build up, since they would occur the same way at each step. You should use your judgment and experiment a bit with variant lengths to get a feel for how close the rulings have to be and how much belly you can accept – I would tend to try to stay under 3% belly.

All of this business is obviously a good candidate for a computer, and there are numerous computer programs available for developable surfaces. Unfortunately, a computer can't do the whole job alone, and you may have to use some of the older methods to help it along. In general a program uses vector arithmetic to find some set of rulings with minimum warp. This surface is often not the one you want. For example, a computer might be happy with a surface between two warped lines comprised of one group of rulings all starting at one end of one line and ending at equal spaces on the other line, and the opposite on the other line, forming two fan like patterns (Figure 9).

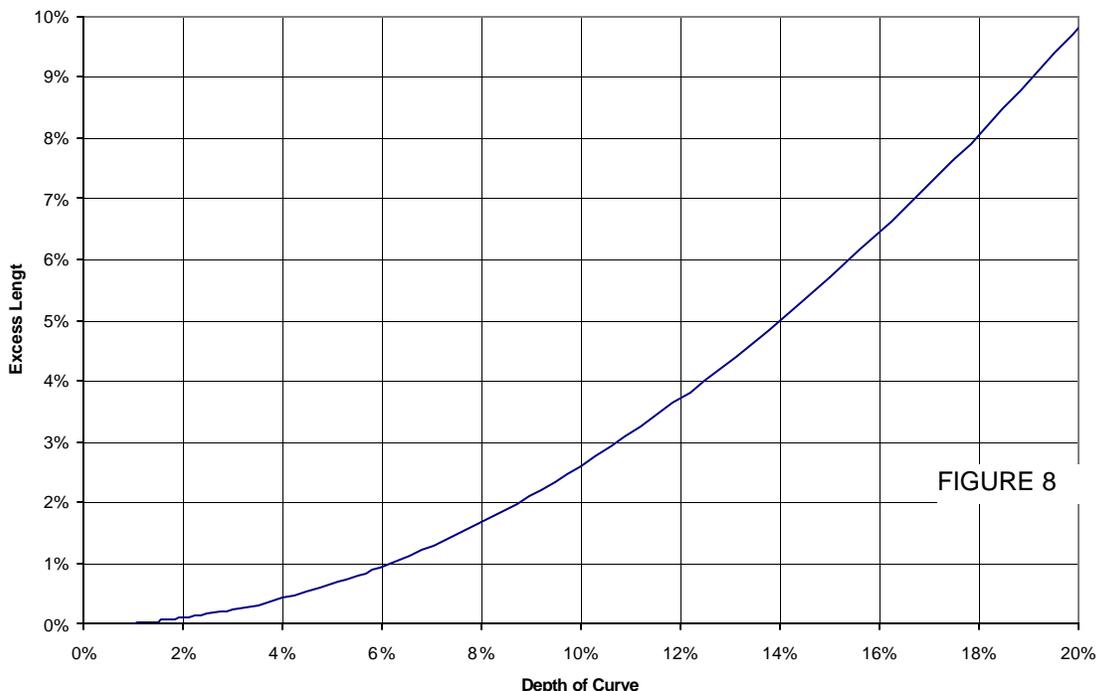


FIGURE 8

This surface is probably not what was desired, but it is developable. These surfaces aren't fair - the join between the two fans is a hard turn. It is possible to have a developable surface between two fair chines that is not itself fair - always inspect the butts and waterlines closely. It is important to note that many programs will allow highly fanned surfaces, and they frequently come up in boats with straight, non-parallel keels and chines. Therefore, it is important that any developable surface program give you tools to manipulate the surface.

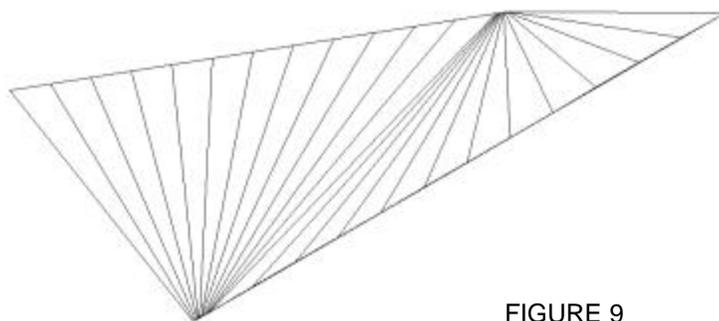


FIGURE 9

For example, the program I use, ShipConstructor, ([www.albacoreresearch.com](http://www.albacoreresearch.com)) allows the user to set the maximum allowable warp, and the maximum allowable divergence (in plane) between two adjacent rulings. This is called "fanning control".

SHIPCAM also allows the edges to be broken into segments and provides for other tricks as well. If you know the surface you want and its approximate rulings, you can use these controls or similar features in other programs to get it. Thus, it is wise to start a design out by using Rabl, or Kilgore's methods or calculation to find a few rulings you like. You will be able to verify what surface you are getting. Then use the program's controls or just input an edge as a series of connected curves joining the rulings you found - the program will be forced to at least begin and end each segment with the rulings you want and will probably get the surface you desire.

*This article originally appeared in **Boatbuilder**, Jan/Feb, 2001 (800-786-3459) and was developed from a presentation made to the Society of Boat and Yacht Design, Chesapeake Chapter, in 1999 which was subsequently excerpted for the Ballast, the SBYD newsletter. Information on SBYD can be obtained by e-mail at [sbydnews@aol.com](mailto:sbydnews@aol.com).*

*The author is a naval architect and has worked at design firms and shipyards in California, the UK and the Pacific Northwest designing offshore oil vessels, small ships, workboats, military craft and yachts. He is currently with the Engineering Logistics Center of the U. S. Coast Guard in Baltimore. The opinions expressed are those of the author and do not reflect official policy of the Coast Guard. He can be reached via email at [cdbarry@hotmail.com](mailto:cdbarry@hotmail.com) or at 8100 West End Drive, Baltimore MD 21226.*