The Focus

PADT

September 15, 2008 | A Publication for ANSYS Users | Issue 67

Fast and Easy 2D Geometry with DesignModeler

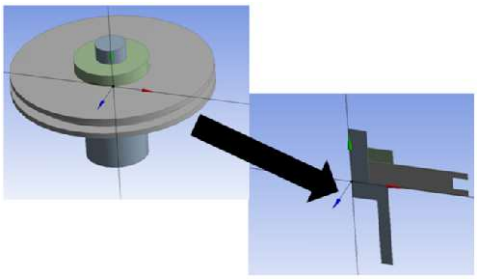
Details View

Details of Thin1	
Thin/Surface	Thin1
Selection Type	Faces to Keep
Geometry	3 Faces
Direction	Inward
FD1, Thickness (>=0)	0 in
FD2, Face Offset (>=0)	0 in

By Doug Oatis

Long has the Workbench community toiled to create 2D geometry from 3D CAD models. Maybe toiled is the wrong word, but it wasn't something that was considered "pleasant". In my opinion, it was easier to setup a 2D analysis in ANSYS PREP7 than in Simulation…until today!

The requirements for doing a 2D analysis in Simulation are identical to PREP7. Your geometry must exist on the XY plane. If you're doing an axisymmetric problem, the geometry must also be on the +X side, with your Y-Axis as the axis of revolution. As far as I knew, there were two ways to create this geometry. The first method involved being skilled in whatever CAD package you were using, creating datum lines, and manually building up each area from lines. The second method used DesignModeler to slice/dice the geometry to 'expose' the axisymmetric face. You would then create sketch planes from each face, and finally used 'Create > Surface from Sketch' to build each 2D body. This worked okay, unless you had more than ~5 bodies, at which point you needed super-ergonomic hardware to prevent permanent ligament damage.

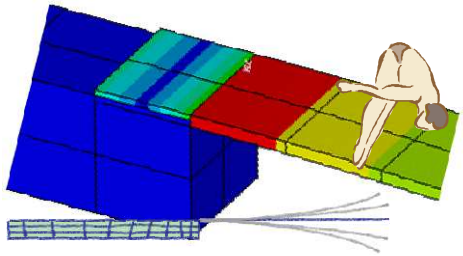


Now, just as Ash realized that by strapping a chain-saw to his arm he could create a great weapon, I have realized DM has a tool well suited to simplify and improve this procedure. That tool is Thin/Surface. Now, the typical usage of the Thin/Surface is to hollow out solid bodies. You select the faces you want to keep/remove, give a thickness, and the CAD engine chugs along until it's hollowed the part out while keeping/removing the selected surfaces. However, if you specify a thickness of 0, and a face offset of 0, it acts as face extractor. If you're a /prep7 user (don't call it ANSYS Classic!), this is the same as deleting all the volumes and areas except for the selected faces.

So now we have a face-extractor tool, all that needs to be done is to slice the geometry to create the faces. This can be done through a series of Slices and Body Operation = Delete. Or, you can use 'Tools > Symmetry' and select up to three symmetry planes. The 'Symmetry' tool will automat-

(Cont. on pg. 3)

Shell To Solid Interfaces: Using Contact



By Rod Scholl

Continuing from last month's article <link> we explore the various methods of attaching shell element to solid element regions. Last time we explored sharing nodes, especially using a "painted-on region" or overlap region to transfer the moments. This approach requires a matched mesh between the shell and solid part, a definite disadvantage.

In this article, we will explore the use of contact elements and contrast the results with the theoretical, as well as the "painted-on" approach. We defined our test case previously, and show a theoretical equivalent stress of 100 in the shells at the interface, and 4.00 at the base of the solid.

Using CONTA17X Elements

The contact pair will be created by meshing the thin area's (shell element region) interface nodes with CONTA175s or interface lines with CONTA177s, and the solid face(s) with CONTA 170s. When using the CONTA17Xs, use Keyoption 12=5, and 2=2. Also, you might specify the SHSD setting, but given that it impacts the

(Cont. on pg. 2)

Table of Contents

(Shell-Solid cont...)

generation of the virtual shell elements, this would be more meaningful for a mis-matched mesh.

Lap Joint

For comparison with the shell method the results for a lap joint are shown in Figure 2. The edge contact method is also shown because of its characteristic edge effects, in Figures 3 & 4. The results are summarized in Table 1.

First let's look at the results of the edge contact – we see easily that there is a false stress concentration at the interface in the shell elements. Thus for a lap type joint, it's hard to recommend this Edge contact approach.

However, the nodal contact approach, affords great accuracy for both the nodal solution and element solution where we've seen earlier that the overlapping shell approach (painted-on method) the nodal solution was much less accurate with this mesh.

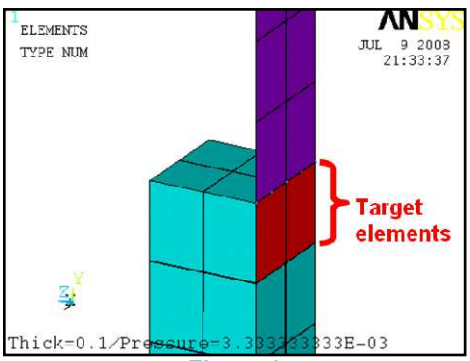


Figure 1

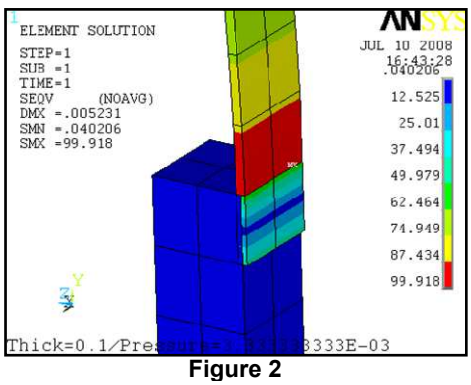


Figure 2

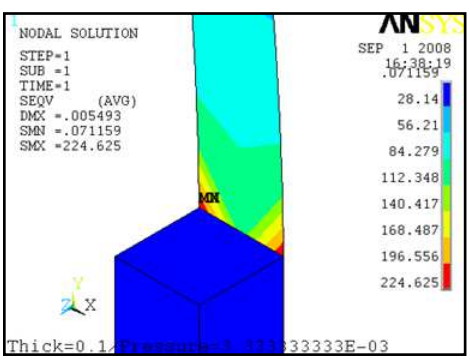


Figure 3

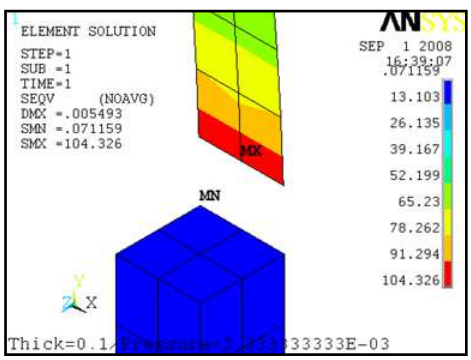


Figure 4

Table 1: Lap Joint

Configuration	Shell Nodal Stress	Shell Element Stress	Base Nodal Stress	Solid Interface
175's (Nodal Contact)	100.1	100.1	3.952	2.7
177's (Edge Contact)	224.6	238.9	3.952	2.8
Overlap Shells	93.03	99.9	3.868	7.2

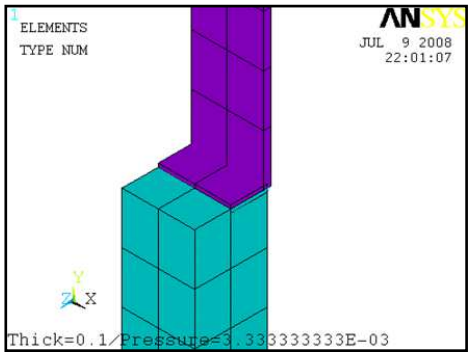


Figure 5

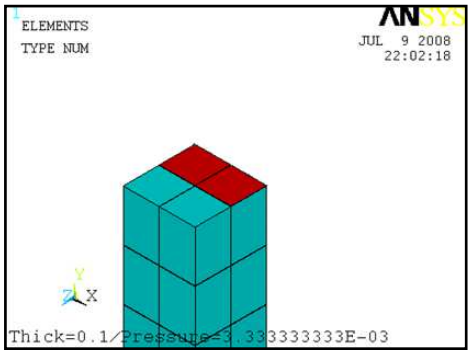


Figure 6

Table 2: Angle Steel Butt Joint

Configuration	Shell Nodal Stress	Shell Element Stress	Base Nodal Stress	Solid Interface
175's (Nodal Contact)	100.0	100.0	3.952	4.012
177's (Edge Contact)	224.7	239.0	3.952	3.822
Overlap Shells	99.91	99.92	4.006	9.594

Down near the base, for this mesh error was 3.3% using the painted shells, where the contact method reduced the error to 1.2%.

Finally, we see at the interface both contact approaches are non-conservative in the solid elements at the interface.

Take away: For lap joints, the nodal contact approach is best, remembering that the solid stresses at the interface are greatly non-conservative, and must be handled differently if they drive the design.

Angle Steel Butt Joint *(Hehe, he said Butt Joint)*

The angle steel butt joint configuration is shown in Figures 5 –6.

Here we see the same performance as above with the lap joint. Using nodal contact (175s) there is a slight improvement in accuracy over the overlapped shell approach, yet a 1% sacrifice in accuracy occurs at removed locations. Once again, the edge contact (177s) has very high stress concentrations at the interface.

Take away: For angle butt joints, the Nodal contact approach is best, remembering that the solid stresses at the interface are greatly non-conservative, and must be handled differently if they drive the design.

Butt Weld using Biased Mesh

We will again look at a biased mesh and unbiased mesh because of the effort required to create the biased mesh in a model with many shell-to-solid interfaces. Plus, we saw previously an actual advantage to leaving the mesh unbiased!

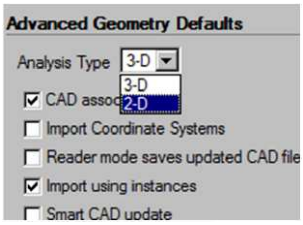
The results of the butt weld with biased mesh is shown in table 3 on the next page.

Note that for nodal contact (CONTA175) there is still the unexplained bias at the solid to shell interface on the solid side shown in figures 7 and 8.

We also see that for edge contact (CONTA177) the nodal solution is still

(Cont. on pg. 3)

(2D DM cont...)



ically cut and remove bodies defined by the selected planes. The 3D to 2D example shown above was actually created with just two operations (not including the 'Freeze'). If your planar faces aren't on the global XY plane, which is the first plane listed in the model tree, you can use a 'Body Operation > Move' to translate the faces. Once you're ready, go to the 'Project' page and set the 'Analysis Type' to be 2D (it defaults to 3D) and then launch Simulation. Finally, set the planar behavior (axisymmetric, plane stress, plane strain, etc).

Remember to use this tool wisely…and shop smart, shop S-Mart.

Editor's note: For those of you who are trying to figure out who Ash is and if S-Mart is a typo, Doug is trying to impress you all by referencing a cult classic movie. I had to look it up. He is talking about "Army of Darkness" a cult over-the-top horror spoof film by Sam Raimi, the same guy who brought us Xena and more recently, Spiderman.

(Shell to Solid cont...)

quite off, but for this biased mesh, so is the element solution!

Take away: We are left to conclude, that if one must have a biased mesh at a butt weld location between shell and solids, the best of the above approaches is to tie into the top surface either with overlapping shells, or with nodal contact (CONTA175). The overlapping shell method requires one to rely on element results (rather than nodal). The nodal contact method requires one overlook the unexplained bias at the interface, although the peak is close in magnitude to expected stresses. Given our hesitancy to believe much about actual stresses at such an interface, this is probably palatable.

Butt Weld – Unbiased Mesh

Given the results above with the unbiased mesh, and considering that we discovered with overlapping shells the most accuracy was obtained by *not* biasing the mesh for a butt weld, let's look at what results are obtained with an unbiased mesh using contact methodology. In most cases biasing the mesh is extra work, so it's reasonable to hope that the following exploration illuminates a clear winner. The results of the butt weld with unbiased mesh is shown in table 4.

Once again, we see that the two best methods are overlapping shells on the top surface, or nodal contact (CONTA175) on the top surface. Here, however, with the unbiased mesh and contact approach, we find that the removed location of the base, has a 1% error or so… not a big deal, but notable for such a removed location. Also, the overlapping shells on top surface approach now has nodal stress results in the shell region which are quite accurate! So it seems that for both overlapping shells, and the contact approach, biasing the mesh is still not greatly advantageous.

Take Away: In a knock-down drag out fight, for the butt weld, I have to side with the overlapping shells on the top surface, rather than contact technology (CONTA17X) because of the better stress results at removed locations, and higher (more accurate) stress in the solid region near the shell to solid interface (although still quite non-conservative). With either approach, we can likely breath a sigh of relief in that there isn't a distinct advantage in biasing the mesh to match the shell thickness at the interface.

Table 3: Butt Weld with Biased Mesh

Configuration	Included Surfaces	Shell Stress (Nodal)	Shell Stress (Element)	Base Stress (Nodal)	Solid Interface Stress (Element)
175s (Nodal Contact)	Top	100.8	100.8	4.005	39.60
175s (Nodal Contact)	Side	100.0	100.1	4.005	2.540
175s (Nodal Contact)	Top & Side	100.0	100.1	4.005	2.410
177s (Edge Contact)	Top	224.0	239.6	4.005	39.81
177s (Edge Contact)	Side	224.6	238.9	4.005	3.290
177s (Edge Contact)	Top & Side	224.6	238.9	4.005	3.200
Overlap Shells	Top	86.14	106.4	4.005	33.310
Overlap Shells	Side	86.14	99.92	4.005	10.110
Overlap Shells	Top & Side	86.14	99.92	4.005	11.940

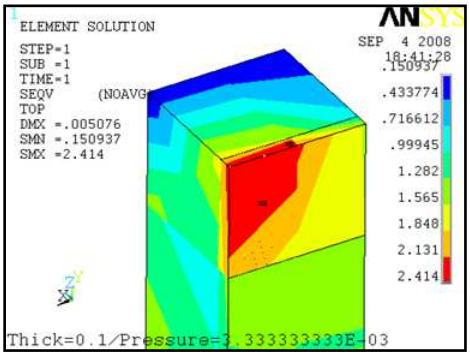


Figure 7

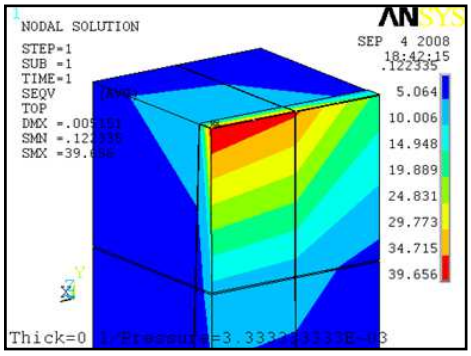


Figure 8

Table 4: Butt Weld with Un-Biased Mesh

Configuration	Included Surfaces	Shell Stress (Nodal)	Shell Stress (Element)	Base Stress (Nodal)	Solid Interface Stress (Element)
175s (Nodal Contact)	Top	100.0	100.0	3.952	4.010
175s (Nodal Contact)	Side	100.1	100.1	3.952	2.700
175s (Nodal Contact)	Top & Side	100.0	100.0	3.952	1.590
177s (Edge Contact)	Top	224.7	239.0	3.952	3.820
177s (Edge Contact)	Side	224.6	238.9	3.952	2.800
177s (Edge Contact)	Top & Side	224.6	239.0	3.952	1.690
Overlap Shells	Top	99.92	99.92	4.006	9.590
Overlap Shells	Side	93.03	99.92	3.868	7.240
Overlap Shells	Top & Side	99.92	99.92	4.006	4.130

ANSYS on the Lake

An Optimization of a Sail Boat Dagger Board using DesignXplorer

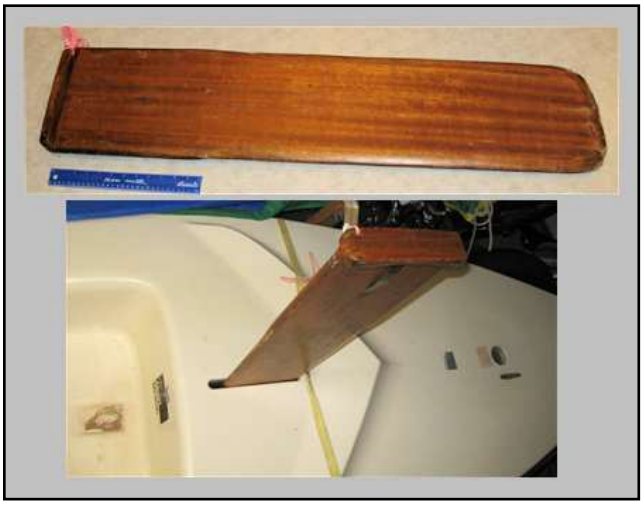
By Ted Harris

An old sailing joke says that a sailor was sailing across the bay, when a power boater zoomed in along side. The power boater yelled, "I'll race you there!" The sailor replied, "I am already there."

I am nowhere near an expert on sailing, on sail boat design, or even on optimization technology. However, because I enjoy sailing as well as learning how to do new things with ANSYS, Inc. products, I decided that I would combine those activities and try out an optimization on a small sail boat dagger board taking advantage of the link that exists between CFX and DesignXplorer at version 11.0. This endeavor resulted in a paper I presented at the recent 2008 ANSYS User Conference in Pittsburgh. The following is an article discussing the process.

First, a few definitions that are relevant to the project.

Dagger Board - a keel that slides in and out of a slot on a sail boat. It is removed in shallow water, for transport, and storage and is dropped in place when under sail. It serves as a second 'airfoil' (the sail being the first) under the water to prevent the boat from moving sideways while under sail. Figure 1 shows the original dagger board from my boat



Reaching - sailing perpendicular to the wind, which is typically the fastest way to sail in a small boat. 'Lift' created on the sail in this configuration pulls the boat in the forward direction.

CFX - one of ANSYS, Inc.'s two current CFD tools (the other being FLUENT). CFX can run as a module within ANSYS Workbench.

DesignXplorer - the ANSYS Workbench design optimization, DFSS, and robust design tool.

DesignModeler - the ANSYS Workbench geometry creation and modification tool.

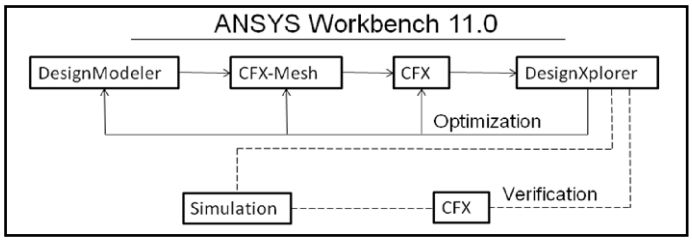
Since there was not a large amount of time or resources to complete the project, some assumptions were made to reduce the scope of the problem while still hopefully demonstrate the techniques involved. First, it was decided that the goal would be a geometric optimization of the dagger board to increase the tangential force on the board during sailing at a reach, with the goal of increasing the speed at which one can sail with a cross wind. It was thought that this would allow for greater resistance to side slip and hence faster forward speed.

Next, the effects of the sail, rigging, rudder, waves, etc. were ignored. It was assumed that the boat was moving on smooth water at 7 knots at a constant orientation. Only steady-state conditions were modeled.

Only one configuration of the board in the water was considered, 15° roll from vertical and 10° rotation from the flow direction. The DesignModeler model had those angles parameterized so other configurations could be easily considered in any future runs. Fillets on the board were removed for the optimization study, but a detailed model with the fillets and a much finer mesh was analyzed in CFX for the optimized configuration to confirm the trends.

No check of drag in the direction of travel was considered, but this could be added in future studies as well as creating a much more detailed objective function, rather than the fairly simplistic maximization of side force on the board.

The optimization process is shown in figure 2:



The optimization process started with geometry definition in DesignModeler. Since the dagger board must fit in a slot in the hull, the shape was assumed to remain as a ¾ in. thick wood plank. Therefore 2D parametric geometry was defined and extruded ¾ in. into 3D to define the board. The shape was allowed to change within the 2D plane. Seven geometric input parameters were chosen to participate in the optimization. Also, instead of sliding down from the top of the boat, it was conceded that an optimized shape

(Cont. on pg. 5

(Optimization, cont...)

would likely need to slide up from under the water into the slot in the hull, necessitating a removable attachment system.

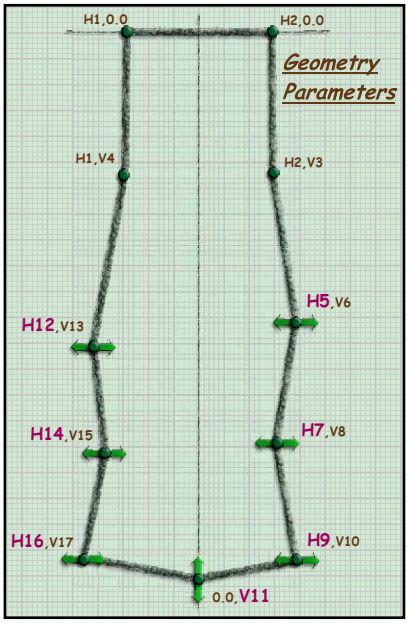


Figure 3: Board Geometry

The board 2D geometry parameterization is shown in Figure 3 with the seven parameters allowed to vary shown in red.

Named selections were created in DesignModeler to facilitate load application and post processing down stream. These named groups of entities propagated through the Workbench modules. The hull of the boat was represented by a simple cylinder and a volume of water surrounding the board was included as well. Boolean operations completed the definition.

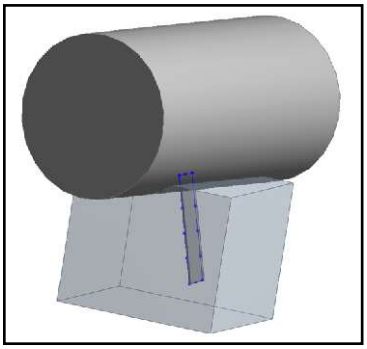


Figure 4: Solid Geometry

The full solid model showing board, simplified hull, and water volume is shown in Figure 4.

Meshing for the CFD model was performed in CFX-Mesh. A coarse mesh was used for the optimization studies, while a fine mesh was used for verification of the loads in the optimized configuration.

The Coarse CFD model for optimization Loops; with 196,000 elements, is shown in Figure 5 and the refined mesh, used for verification of the optimized design, is shown in Figure 6 and contains 1,210,000 elements.

The CFD simulation was set up in CFX with the following conditions:

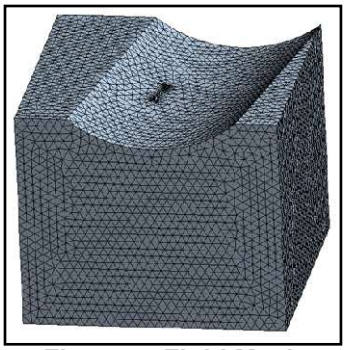


Figure 5: Fluid Mesh

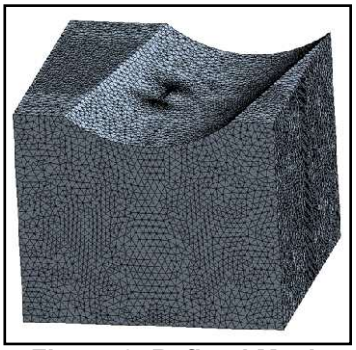


Figure 6: Refined Mesh

- *Fluid:* water, no heat transfer
- *Reference Pressure:* 1 atm
- *Turbulence Model:* k-Epsilon
- *Inlet:* Normal Speed at 7 knots (about 12 ft/sec)
  - An anecdotal max. speed for the test boat is 9 knots
- *Outlet:* 0 Pa Relative Pressure
- *Dagger Board:* Wall boundary with No Slip
- *Hull:* Wall boundary with No Slip
- *Bottom and sides of water volume:* Symmetry

Expressions were created using CEL, CFX Expression Language, to interrogate and store desired results for use in the optimization. The resulting parameters contained force in global X on each side of the board and the difference between those two force values. That force difference was passed to DesignXplorer.

A goals-driven optimization was then performed in DesignXplorer. The objective function chosen was to maximize the force on the board, perpendicular to the direction of travel. The seven geometric input parameters resulted in automatic multiple passes through the loop: DX > DM > CFX-Mesh > CFX > DX. The fore-to-aft parameters were allowed to vary by 25%, while the top-to-bottom parameter was allowed to vary by 10%.

The resulting 'best' design produced a force of 537.97 N or about 130 lb. The worst configuration produces a side force of 402.2 N. The force increase over the original design was about 20%. For verification purposes, a physical static test rig with a fish scale measured a side force on the test boat at 35 lb. in a strong breeze, which at least confirmed a rough order of magnitude correlation.

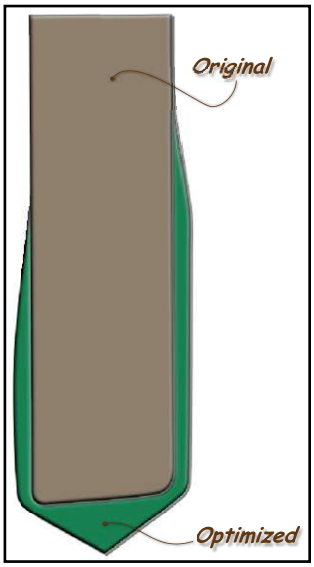


Figure 7: Original and Optimized Geometry

The resulting optimized geometry is compared to the original in Figure 7.

Figure 8 is a DesignXplorer sensitivity chart showing that the configuration is most sensitive to geometry

(Cont. on pg. 6)

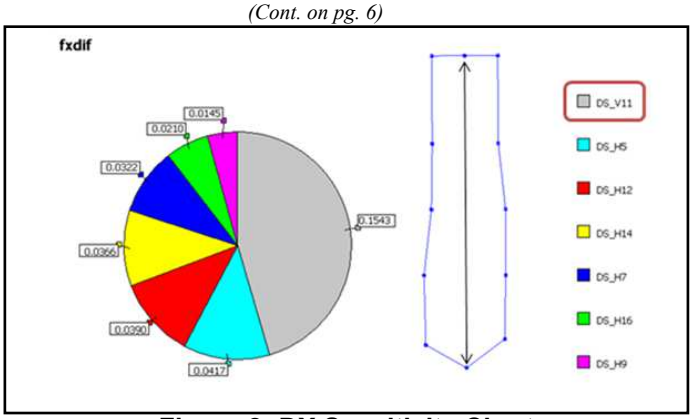


Figure 8: DX Sensitivity Chart

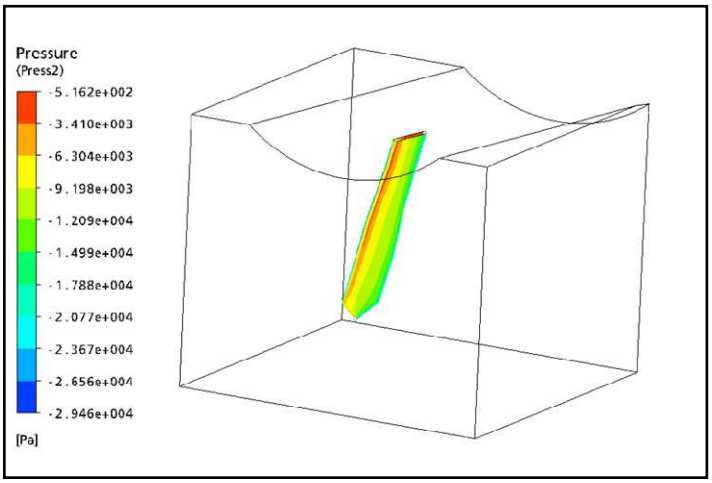


Figure 9: CFX Pressure Distribution

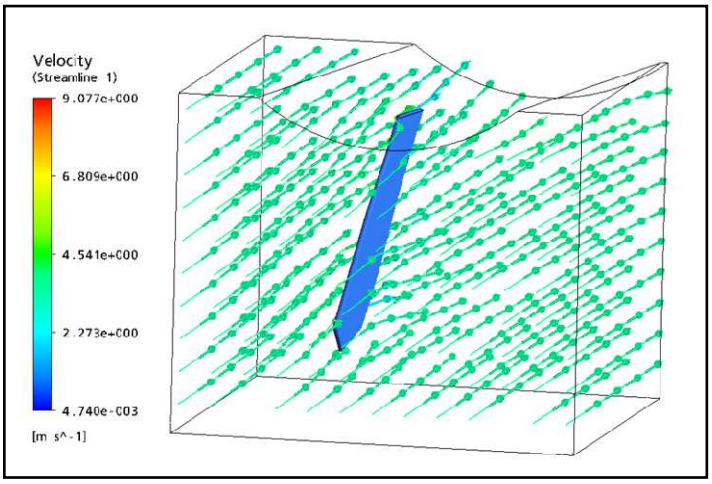


Figure 10: CFX Streamlines

(Optimization, cont...)

parameter DS_V11. This is the length (depth) of the middle of the end of the board.

Figure 9 shows the resulting CFX Pressure distribution for the optimized shape and Figure 10 shows the water streamlines for the same case.

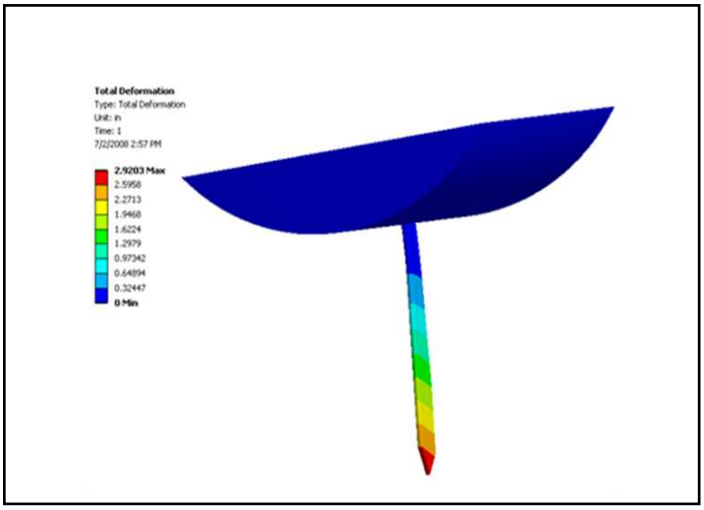


Figure 11: Structural Deflection

Structural runs were performed in Workbench Simulation to compare the deflections of the original board vs. the optimized board for their respective pressure loadings. The predicted tip deflection of the optimized board was slightly higher than the original. Figure 11 shows the deflection results from Simulation.

Figure 12: Making the Prototype

With the optimized shape obtained from the DesignXplorer/CFX process, the next step was to fabricate a board for testing purposes. The author constructed the optimized board from a plank of poplar. Poplar was chosen over oak because it was less expensive and a bit easier to work with. Figure 12 shows the Prototype

The two boards and an Alcort Minifish sail boat were then taken to a local lake to obtain experimental results. Back to back runs were made with the original dagger board and the optimized board. Time constraints limited the testing to one pass with each board. Each pass was about 0.85 mile round trip. The maximum speed for each run was stored using a hand-held GPS receiver. Although the boat 'seemed' faster with the optimized board, the max speed recorded was 7.5 mph, vs. 7.6 mph with the original board (about 6.5 knots, not too far off the assumption in the CFD runs of 7.0 knots). Since wind conditions vary constantly the results have been deemed inconclusive, although the boat did seem to handle better with the optimized board.

In conclusion, CFX linked with DesignXplorer can be used as a tool to find an optimal design for a well-defined problem. More details would be needed to really hone the technique for this problem, such as using a more detailed objective function, further studying mesh density effects, including fillets in each pass, utilizing other configurations of the angles of the boat and board as well as velocity, and greater potential geometry variations.

Figure 13: The Author Testing the Design on a Nice Windy Day

Awesome APDL

Looping on Numbered Tables

Let's say you want to apply a surface load or boundary force (SF/BF) to a bunch of areas. But you don't know how many tables or areas you are going to have. So you make a component/name selection for each zone you want your load on and number them sequentially (myzone_1, myzone_2, etc...) and then you make a table for each zone (mytbl_1, mytbl_2, etc...) Then you make a do loop that selects each component followed by an SF or BF to assign to assign the load:

```
*do,I,1,3
        cmsel,s,myzone_%i%
        bf,all,hgen,mytbl_%i%
*enddo
```

This is especially useful when you are writing APDL snippets a Workbench Simulation model. If you number hour named selection you can have a very general macro that goes through and semi-automatically applies the proper loads. Sounds good so far.

But, and there is always a but, if you try the above code you will find out that the BF command is applying a value of 0.7888609052E-30. This is because the parser only treats a text string as a table name if it starts and stops with a percent sign. Anything other than that and ANSYS thinks you are putting in a parameter and not a table name. If you try `%mytbl_%%i%` you get an error. And doing `atmp = mytbl_%i%` and specify `%atmp%` does not work either, because it looks for a table called `atmp`.

So, enter the old "self generating macro" approach: have your macro write the proper command to a file, then read that file. The example example given here is even more general, in that the table and component names are variables as well. It was used as a way to apply a series of tables that vary by X position under four different coordinate systems.

Near the bottom you will find the CFOPEN, VWRITE, CFCLOS, /INPUT, /DELETE needed to build and execute a command on the fly that can be used with whatever application you want to use it on.

```
*do,i,1,4
 *DEL,_FNCNAME  ! clean up variables
   *DEL,_FNCMTID
   *DEL,_FNCCSYS
   *SET,_FNCNAME,'d_test%i%' !define name of table
   *SET,_FNCCSYS,act_csys    !set csys for the table
! Define Table:
  *DIM,%_FNCNAME%,TABLE,6,12,1,,,,%_FNCCSYS%
! Build table:
  *SET,%_FNCNAME%(0,0,1), 0.0, -999
! SNIP - lines removed to conserve space
  *SET,%_FNCNAME%(0,6,1), 0.0, -2, 0, 1, 2, 17, -1
  *SET,%_FNCNAME%(0,12,1), 0.0, 99, 0, 1, -3, 0, 0
  cmsel,s,%d_root%%i% ! select named selection/group
   d_a='%'! Put percent sign in a variable
  *cfopen,d_temp,txt !open a temp file to write to
    *vwrite,d_a,_FNCNAME,d_a  ! Write the command
    bf,all,hgen,%C%C%C
  *cfclose     ! Close the file
  /input,d_temp,txt ! read in the command
  /delete,d_temp,txt ! Clean up after yourself
  act_csys=act_csys+1 !increment your csys
*enddo
```

One key thing you will notice, is that you create a character variable to hold your percent signs (d_a) because the *vwrite command treats %'s as special characters. Another way to do it would be to put double %%'s on the format line:

```
*vwrite,_FNCNAME  ! Write the command
bf,all,hgen,%%%C%%.
```

News - Links - Info

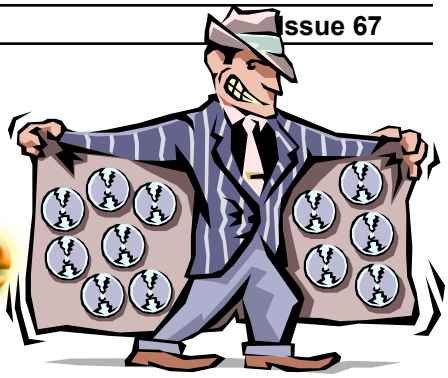
- XANSYS now has a blog for opinion, observations and some humor <xansys.blogspot.com>
- Ansoft Acquisition is Completed <link>
- The channel partner in Italy, Enginesoft has an awesome Newsletter (makes ours look amateur). Many of the articles are in english: <link>
- Get ANSYS news directly by subscribing to their RSS feed: <link>
- While looking at the Enginsoft newsletter, you should check out their optimization tool: ModeFrontier: <link>

Upcoming Training Classes

Month	Start	End	#	Title	Location
Sep '08	9/18	9/20	102	Introduction to ANSYS, Part II	Tempe, AZ
	9/22	9/23	201	Basic Structural Nonlinearities	Tempe, AZ
	9/24	9/25	204	Advanced Contact and Fasteners	Tempe, AZ
Oct '08	10/2	10/3	104	Workbench Simulation – Intro	Las Veg., NV
	10/6	10/7	205	Workbench Simulation Dynamics	Tempe, AZ
	10/9	10/10	100	Engineering with FE Analysis	Tempe, AZ
	10/14	10/15	207	WB Simulation:Struct. Nonlinearities	Tempe, AZ
	10/16	10/17	302	Workbench Simulation Heat Transfer	Tempe, AZ
	10/20	10/20	107	ANSYS Workbench DesignModeler	Tempe, AZ
	10/22	10/24	902	Multiphysics Simulation for MEMS	Tempe, AZ
	10/27	10/27	411	WB Simulation Electromagnetics	Tempe, AZ
	10/28	10/28	206	WB Rigid & Flexible Dynamics	Tempe, AZ
Nov '08	11/5	11/7	101	Introduction to ANSYS, Part I	Tempe, AZ
	11/10	11/10	702	ANSYS Workbench DesignXplorer	Tempe, AZ
	11/12	11/13	301	Heat Transfer	Tempe, AZ
	11/17	11/18	102	Introduction to ANSYS, Part II	Tempe, AZ
	11/19	11/20	204	Advanced Contact and Fasteners	Tempe, AZ
	11/24	11/25	604	Introduction to CFX	Tempe, AZ

The Focus is a periodic publication of Phoenix Analysis & Design Technologies (PADT). Its goal is to educate and entertain the worldwide ANSYS user community. More information on this publication can be found at: http://www.padtinc.com/epubs/focus/about

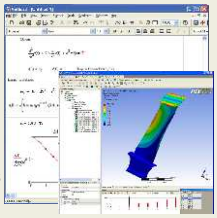
The Shameless Advertising Page

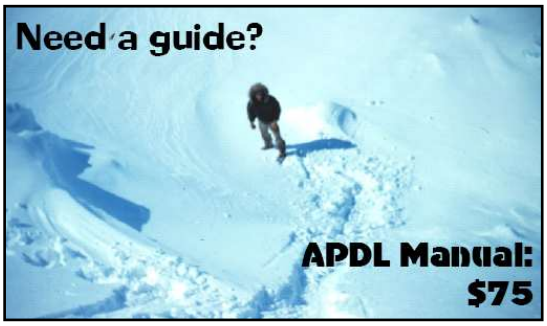


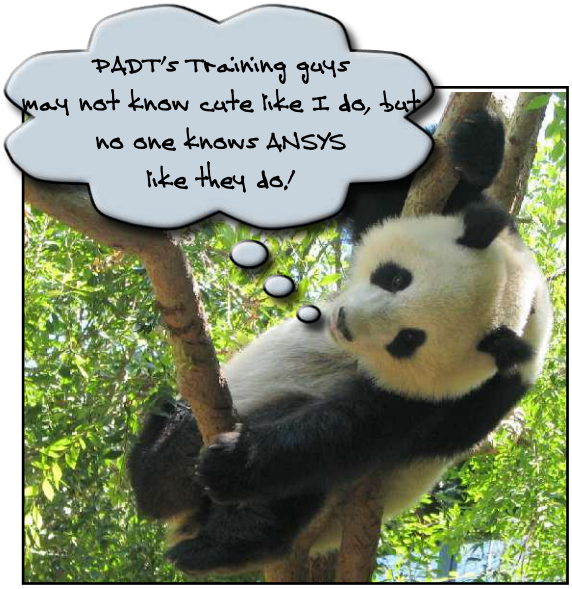
ANSYS + Mathcad

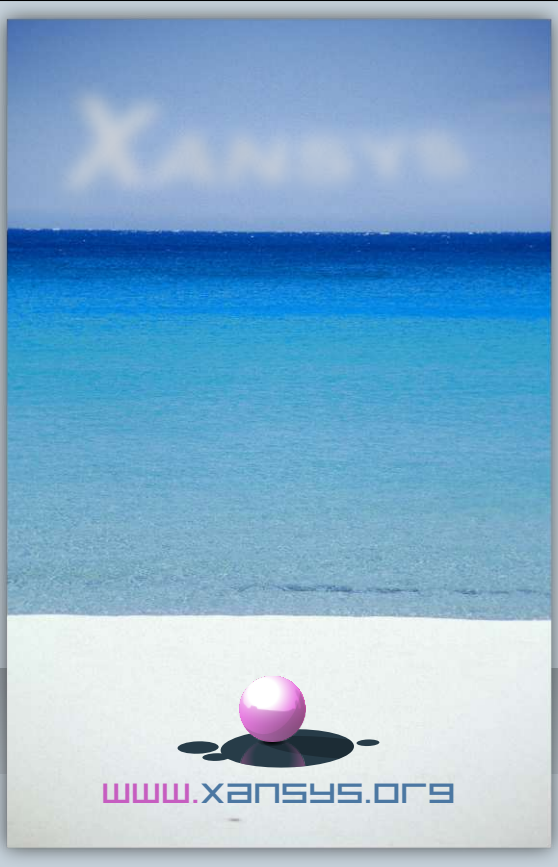
PADT is Taking the Lead

- Mathcad Sales and Support to ANSYS users
- Development of Opensource interface tools
- Training and educational materials
- Stay tuned to "The Focus" to learn more









For over a decade ANSYS users around the world have been gathering on the XANSYS mailing list to share their knowledge, experience and humor. Join the conversation with 3,500+ practitioners on the largest independent ANSYS community on the web.

XANSYS: Your Online Home

PADT

www.padtinc.com

- SLA - SLS - FDM - Rapid Injection Molding - Soft Tooling -

Delivering Speed and Price Without Sacrificing Quality