# AUTOMATED FLUID-STRUCTURE INTERACTION ANALYSIS

**Daron A. Isaac, Micheal P. Iverson**
ATK Thiokol Propulsion
Brigham City, Utah

## ABSTRACT

An automated Fluid-Structure Interaction (FSI) analysis procedure has been developed at ATK Thiokol Propulsion that couples computational fluid dynamics (CFD) and structural finite element (FE) analysis to solve FSI problems. The procedure externally couples a steady-state CFD analysis using Fluent® and a structural FE analysis using ABAQUS®. Pressure results from the CFD solution are interpolated and applied as pressure boundary conditions on the structural model. Displacements from the structural analysis are interpolated and applied to the boundary of the CFD mesh. Iteration between the CFD and the structural analysis continues until a solution is reached. The FSI procedure provides controls to monitor the solution and define termination criteria, as well as manage output. Automatic report generation of the solution is another feature of the FSI procedure. Plans and funding are in place to extend the FSI procedure to include coupling with thermal analysis as well.

The FEM Builder program provides pre- and post-processing functions for the FSI procedure, such as geometry creation, finite element mesh generation, material property definition, and boundary condition application. Several of the pre-processing functions were created exclusively for FSI solutions. The FEM Builder program provides interfaces to other finite element pre/postprocessors and a number of analysis programs. Scripted access to FEM Builder program functions is provided through the FEM Python module. The FEM Python module functions provide the basis of the FSI procedure.

The FEM Builder FSI procedure is applied to the analysis of a fictitious solid rocket motor. The problem of bore choking is examined in order to demonstrate the capabilities of the FSI procedure on a problem with potentially large structural deformations. An overview of the input required by the FSI procedure to solve this problem is discussed.

## INTRODUCTION

Interaction of physical phenomena occurs regularly in nearly every situation imaginable. The interaction between changes in temperature and the thermal expansion of an object, or the

---

This work was performed under contract no. F04611-99-C-0002 with the AFRL/PRSB, 4 Draco Dr., Edwards AFB CA 93524-7160.

deformation of an object due to an applied force are two common examples. Fortunately, these interactions are usually negligible and can be ignored for most analyses. In some cases, these interactions are significant and cannot be ignored in an analysis.

One example from the aerospace industry is the FSI that occurs in a solid rocket motor (SRM) between the internal pressure distribution and the deformation of the motor propellant. In order to accurately predict the performance of a SRM, a coupled CFD-structural analysis must be performed. Work is in process at ATK Thiokol Propulsion to develop software to facilitate and perform automated coupled analysis. In particular this paper discusses an automated analysis procedure that can be used to model fluid-structural interactions.

The automated coupled analysis described in this paper couples finite element structural analysis and computational fluid dynamics. The term "coupled solution" has several different meanings that are primarily differentiated by the level of integration. Coupled solutions may be described as: 1) *manual*–the analyst manually extracts data from one analysis for input to the next analysis, 2) *interfaced*–programmatic interfaces to analysis codes transfer data but the analyst manually directs the analysis process, 3) *external*–interfaces to analysis codes are created and the analysis process is automated, and 4) *internal* or *monolithic*–one analysis code does it all. An external approach was taken in developing the automated coupled analysis software for two main reasons. First, it was deemed desirable to use commercial, state-of-the-art analysis codes in the coupled analysis. This allowed the engineers to leverage all of the features and functions of the analysis codes that they were already familiar with. Second, the coupling was not severe enough to require internal coupling, such as is required for mass and momentum in a CFD codes. After examining these coupling approaches, an external coupling method was selected as the best method to pursue for an automated FSI analysis procedure. Since it's first release two years ago, engineers and analysts at ATK Thiokol Propulsion have used this method on several solid rocket motors, including the RSRM, as well as on some proposals and designs.

## SOFTWARE ARCHITECTURE

One of the purposes and main functions of the FEM Builder software package being developed at ATK Thiokol Propulsion is to provide necessary pre-, post-, and inter-processing functions to facilitate setting up and solving coupled analyses. These functions are available either from the graphical user interface program or from a scripted (programming language) user interface. The graphical user interface program, called FEM Builder, is a Windows®-based program. The scripted user interface, called FEM Python, is a platform-independent Python[1] extension module. Extension modules can be written to extend the native functionality of the Python programming language. Both the graphical and scripted user interfaces access the same pre-, post-, and inter-processing functions.

[1] Python is an interpreted, interactive, object-oriented programming language. Visit www.python.org for more information.

The FEM Python module enables the analyst to create automated solution sequences for anything from progressive fracture to analysis sequences utilizing the full set of analysis tools as shown in Figure 1.

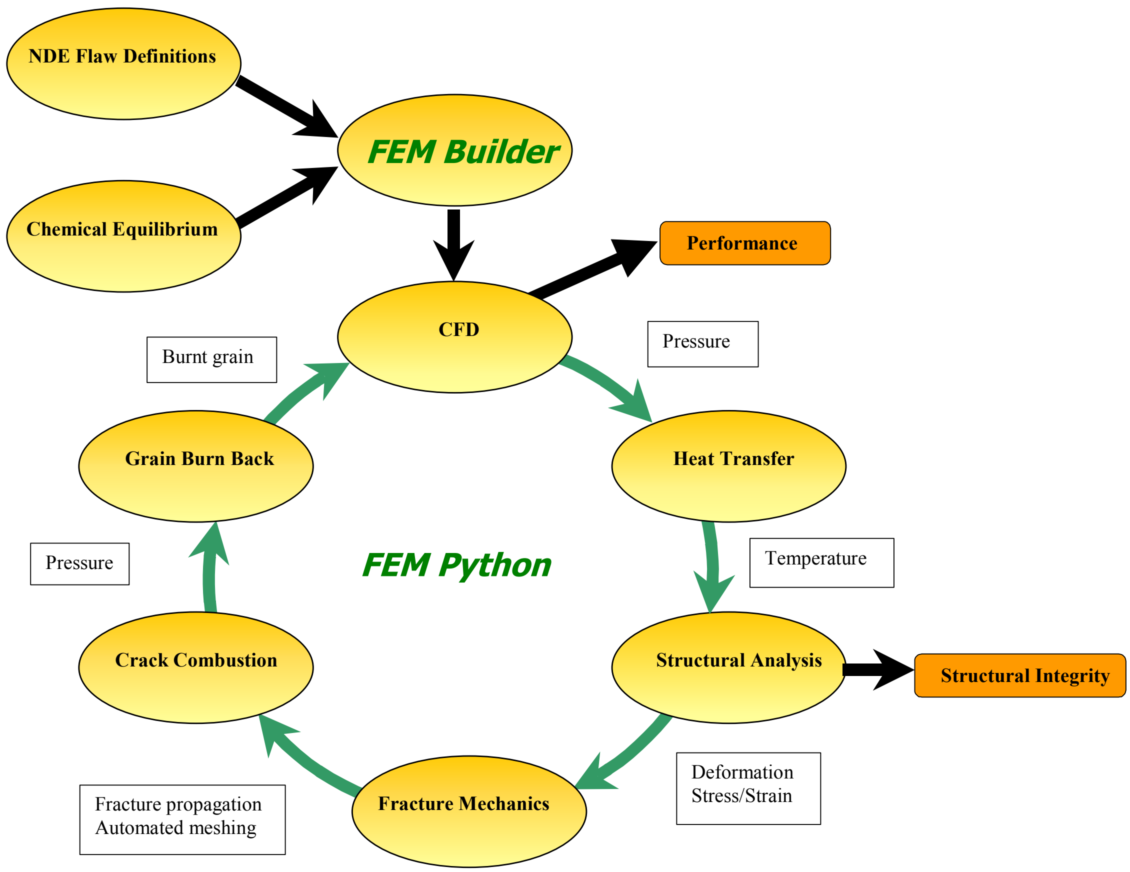


**Figure 1: Possible FEM Builder solution sequence for solid rocket motors**

FEM Builder SOLVERS

The architecture of the FEM Builder software package also includes several Python classes, called solvers that encapsulate the necessary information to solve particular analyses. For example, a solver has been created to execute ABAQUS® given a FEM Builder data file containing a structural model. The user provides information needed to write an ABAQUS® input file, which is stored as part of the solver. The solver takes the specified model, writes an ABAQUS® input file, executes ABAQUS® on any computer on the network, and reads the results from the analysis back into the structural model. A similar solver has also been created for executing CFD models using Fluent®. These two individual analysis solvers, one for structural analysis and one for CFD analysis, are driven by a coupled FSI solver that controls the solution sequence of a coupled solution of a fluid-structural interaction problem.

Additional solvers have been developed for crack nucleation and propagation. Additional solvers are currently planned to extend the analysis codes supported, as well as additional types of coupled analysis, which include fluid-fluid, fluid-thermal, and fluid-thermal-structural.

## OTHER FEM Builder FEATURES

The FEM Builder program provides most functions one might expect to find in a standard pre and post processor. Standard pre-processing functions include support for: geometry creation, 2D and 3D mesh generation, material property definition, and boundary condition application. FEM Builder will also import finite element modeling entities from I-DEAS® Master Series® and Patran®. Standard post-processing functions include display of deformed geometry, contour and vector plots, as well as XY plots. Displays may be transferred for use in documentation and presentations using the copy-to-clipboard or copy-to-bitmap options.

In addition to those standard functions, FEM Builder© provides a number of somewhat unique pre- and post- processing functions. These functions include:

- Flaw insertion for 2D models. Flaws may be zero volume cracks, elliptical flaws, or general flaws with volume; see Figure 2 - Figure 4.
- Result superposition.
- Factor of safety and margin of safety calculations for 26 different criteria as well as support for user defined criteria.
- Insertion of cracks/debonds based on continuum failure.
- J-Integral, Crack Closure Integral, and Crack Opening Displacement fracture mechanics calculations.

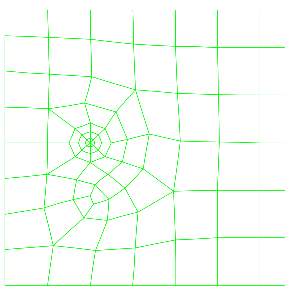

**Figure 2 Crack insertion**

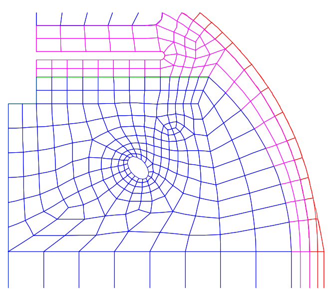

**Figure 3 Elliptical void**

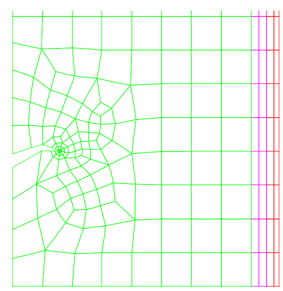

**Figure 4 General flaw**

FEM Builder also provides a number of functions created explicitly for transferring data between different finite element models. Those functions include:

- Translate, rotate, and mirror functions so that grids created in different coordinate systems can be transformed into the same coordinate space.
- Interpolation of results for use as boundary conditions, e.g. interpolation of pressure from a CFD grid to a structural grid; see Figure 5.

- Color boundary condition by value; see Figure 5. This function is especially useful in validating interpolated boundary conditions.
- Interpolation of analysis results between models, e.g. interpolation of displacements from the structural grid to the CFD grid in order to deform the CFD grid.
- Interpolation of a result for use as an initial condition, e.g. interpolation of temperature from a heat transfer grid to a structural grid.
- Ablation of a structural grid to match an ablated thermal grid boundary.

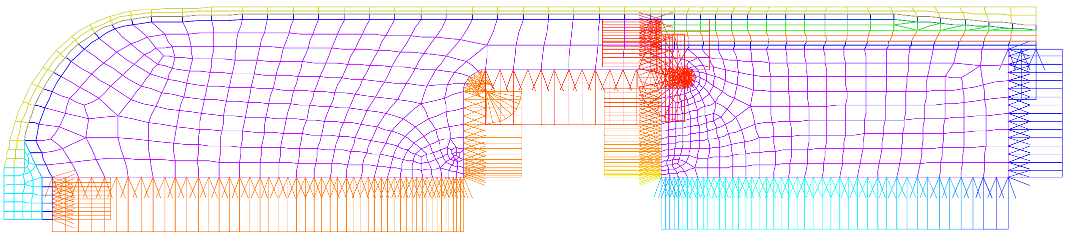

**Figure 5 Interpolated pressure boundary condition colored by value**

## AUTOMATED FSI ANALYSIS PROCEDURE

The automated analysis procedure uses external coupling to link CFD and structural analysis to solve FSI problems. The approach couples a steady-state CFD analysis and a linear elastic structural analysis. Results from each of these individual analysis codes are transferred and used to drive the other and are iteratively solved until a solution is reached. The pressures computed in the CFD analysis are used to automatically create pressure boundary conditions on the structural model. The displacements from the structural model are used to automatically deform the CFD grid. This analysis procedure has been demonstrated on axisymmetric, and 3D models for internal and external flow. The FEM Python module provides the backbone for the FSI solver.

A flowchart of the automated FSI analysis procedure is shown in Figure 6. The process starts with the analyst(s) preparing the CFD and structural models and setting up the FSI python script. The FSI python script initializes the automated FSI solver and iterates between the CFD and structural analysis. The CFD analysis is performed by automatically writing an input file, executing the CFD analysis code, and reading the output file. Once the CFD analysis is complete, the computed pressure distribution is interpolated to specified element faces of the structural model as pressure boundary conditions. The structural analysis is then performed by writing an input file, executing the structural analysis code, and reading the output file. Displacement results from the structural analysis are interpolated to specified nodes of the CFD model. The nodes of the CFD model with displacements are then moved to their deformed positions. The nodes that were not included in the displacement interpolation are relocated, either by re-meshing and/or smoothing, to obtain a good CFD mesh that matches the current deformation state of the structural model. Termination criteria control the iteration loop and determine when the analysis is complete. Once the FSI analysis has completed, the analyst evaluates the results.

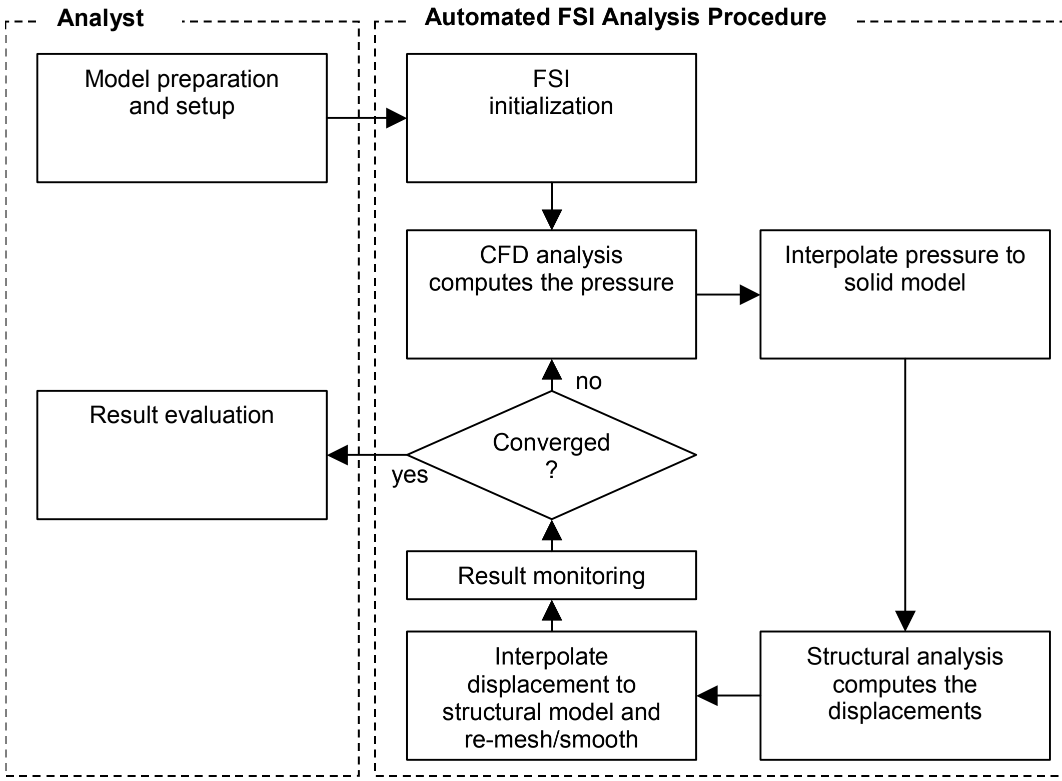


**Figure 6: Basic Flowchart showing automated FSI analysis procedure**

Other features of the automated coupled FSI analysis include:

- Result monitoring for key nodes and/or elements during the FSI solution.
- Factor of Safety calculations based on user-specified failure criteria
- Report generation (a Microsoft® Word® document) summarizing entire FSI solution
- Creation of structural deformation and fluid pressure animation files
- Local or remote execution of analysis programs

## INTERPOLATION

One of the enabling functions of the automated FSI coupled analysis is correct and proper interpolation of analysis results to another FE model. There are two interpolation methods used to interpolate pressures and displacements. The first method is used in locations where the CFD grid and the structural grid are in close proximity. For example, in pressure interpolation, the center of a structural element face is projected onto the CFD grid. This projection identifies the CFD element and the natural coordinates within that element of the projection point. The pressure value at that point is interpolated from the CFD pressure result. This pressure value is then applied as a boundary condition on the element face of the structural model. A similar operation is performed to interpolate displacements onto nodes of the CFD model in close proximity to the structural model.

The second method of interpolation is used in areas when there is a significant difference between the CFD model and structural model boundaries. Motor features considered critical to model in the structural mesh may not important in the CFD grid. Such locations may occur in solid rocket motors in the fin sections, segment joints, or stress relief flaps. This method uses an intermediate step to allow better control of the interpolation. For example, in pressure interpolation, points of a segmented line (defined by the analyst) are interpolated onto the CFD grid (points A and B in Figure 7). The corresponding pressures at these points are obtained, as described above. The center of a structural element face is then projected onto the segmented line and the pressure at that location is determined by linear interpolation. This pressure is then applied as a boundary condition on the element face of the structural model (Figure 8). Element face centroids that project past the end points of the segmented line (points A and B) are assigned the value at the nearest end point. Similar functions are used to interpolate displacements in these areas onto the CFD model.

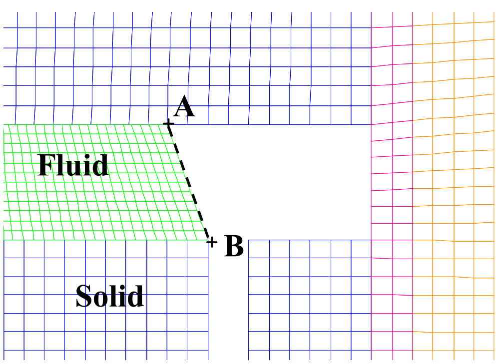


**Figure 7: Pressure interpolation in a propellant joint/stress relief flap using a segmented line as an intermediate step**

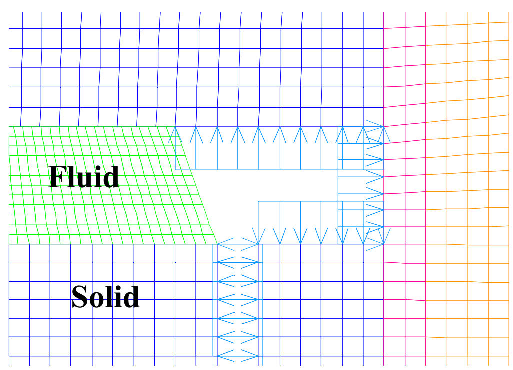


**Figure 8: Pressure boundary conditions from pressure interpolation applied to structural model**

## SAMPLE ANALYSIS – BORE CHOKING

Bore choking is a possible problem in solid rocket motor designs, and has the potential of causing motor over-pressure and catastrophic failure. Bore choking occurs when the propellant deforms radially inward and disrupts the flow field, causing a choked flow condition inside the motor. Bore choking is most likely to happen downstream of segment joints or radial slots. In this area 2D/3D CFD is necessary to accurately predict the flow field and the resulting fluid-structure interaction.

The phenomenon of bore choking in a solid rocket motor is typically caused by localized areas of low pressure. These areas of low pressure develop primarily due to flow separation downstream of a segment joint/slot and are further enhanced by the radial flow of exhaust gas from the segment joint/slot (Figure 9). On a macro scale, a solid rocket motor is primarily one-dimensional flow—the vast majority of the mass is moving in one direction. However, the locations where bore choking is likely to occur are the areas of localized 2D/3D flow, which require CFD analysis to accurately predict. The pressure difference around a downstream corner can be significant—the CFD analysis shown in Figure 9 predicts approximately a 25.0 psi (170 kPa) difference, which cannot be predicted by a one-dimensional fluid flow analysis. This pressure difference causes the downstream corner of the propellant to deform into the flow field, enhancing the problematic flow separation. This causes greater corner deflections, and thus a lower pressure. If the elastic modulus of the propellant is not stiff enough, the downstream corner will continue to constrict the flow, resulting in an unstable condition and bore choking.

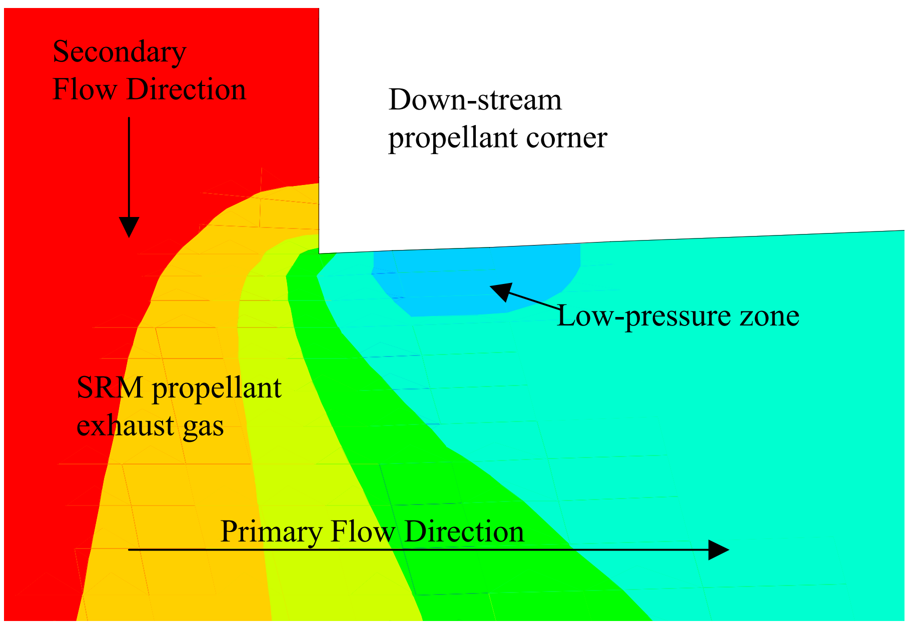


**Figure 9: Pressure contours of 5.0 psi (34 kPa) of flow around a down-stream corner of a segment joint/slot**

DESCRIPTION

The sample bore choking analysis investigates a simple, axisymmetric solid rocket motor (SRM) design with a radial slot in the propellant[2]. The SRM design is shown in Figure 10 and the basic dimensions are listed in Table 1. The FSI solution is obtained for two conditions. The first

[2] All geometry, features, and properties of this solid rocket motor and sample bore choking analysis are representative. Any resemblance to an actual solid rocket motor, real or fictitious, is purely coincidental.

condition, identified as Model A, uses a propellant elastic modulus of 200 psi (1.4 MPa); the second condition, identified as Model B, uses a propellant elastic modulus of 500 psi (3.5 MPa).

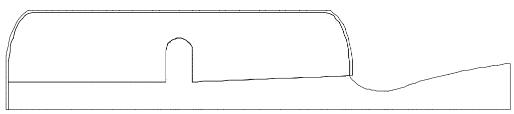

**Figure 10: Schematic of solid rocket motor used in sample bore choking analysis**

**Table 1: Dimensions for solid rocket motor used in sample bore choking analysis**

| **Solid Rocket Motor Design Parameters** | |
|---|---|
| Case Length | 13.0 in (0.33 m) |
| Case Diameter (outside) | 4.0 in (0.10 m) |
| Bore radius (slot corners) | 1.125 in (0.02858 m) |
| Nozzle Throat Radius | 0.75 in (0.019 m) |
| Nozzle Length | 6.0 in (0.15 m) |

SETUP

The input requirements for the FSI solver include a CFD model, a structural model, and an FSI python script that initializes parameters needed for the FSI analysis. Additional node groups are defined in the CFD model to indicate the nodes where displacement results are to be interpolated. An additional face groups are defined in the structural FE model to indicate the element faces where pressure loads are to be created.

The sample axisymmetric CFD model contains a grid of the flow field, properties of the exhaust gas and propellant, boundary conditions, as well as fluid flow parameters (Figure 11). Additionally, four node groups are defined on mesh region boundaries. These nodes are used in displacement interpolation and later deformed (Figure 12). This model is saved as a FEM Builder data file and is specified in the FSI python script.

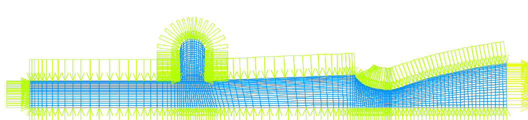

**Figure 11: CFD grid and boundary conditions**

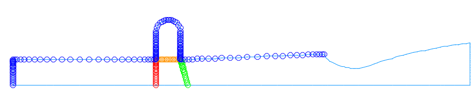

**Figure 12: Node groups for displacement interpolation**

The sample axisymmetric structural model contains a grid of the SRM, material properties, and boundary conditions (Figure 13). The nozzle was assumed not to deform, so it was not included in the structural FE model. The model is also saved as a FEM Builder data file and is specified in the FSI python script.  There is one additional group of element faces on the fluid-solid interface, which is used in pressure boundary condition interpolation and application (Figure 14).

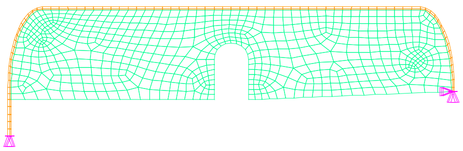

**Figure 13: Structural grid with restraints**

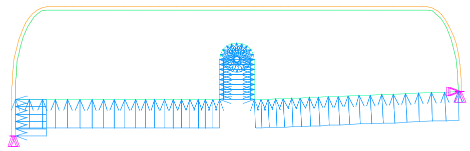

**Figure 14: Pressure boundary conditions applied to element face group**

```
# Example 11 - Coupled Fluid-Solid Analysis

#Import necessary modules
from CExecuteFluent  import *
from CExecuteAbaqus  import *
from CSolveCoupledFSISteady import *

#Assign class values
LESS  = CSolveCoupledFSISteady()
Solid = CExecuteAbaqus()
Fluid = CExecuteFluent()

LESS.SetLogFileName('Example11.log')
LESS.SetReportFileName('Example11.doc')
LESS.SetStandardPostFunctions()

#Initialize Solid member variables
Solid.SetFDBFile('Example11-solid.fdb')
Solid.SetModelName('Example11-solid')
Solid.SetModelUnits('In, F')
Solid.SetMaxTime(1.0)
Solid.SetStandardPostFunctions()
Solid.AddFoSCriteria('Energy Density')

#Initialize Fluid member variables
Fluid.SetFDBFile('Example11-fluid.fdb')
Fluid.SetModelName('Example11-fluid')
Fluid.SetModelUnits('In, F')
Fluid.SetMaxTime(1.0)

#Fluent specific
Fluid.SetSolverMethod('Segregated')
Fluid.SetLimits(LimitMinT=20.0, LimitMaxT=10000.0, LimitMinP=2.0, LimitMaxP=3000.0,LimitMaxViscRatio=10000.0)
Fluid.SetOperatingPressure(14.7)
Fluid.SetCustomSolve('Example11-fluid-custom-solve.jou')
Fluid.SetInitialResults('Y','Y','Y','Y','Y','Y','Y')

#Initialize LESS member variables
LESS.SetMaxTime(6)
LESS.SetMaxIteration(7)
LESS.SetMinIteration(3)

LESS.SetFluidSolver(Fluid)
LESS.SetSolidSolver(Solid)

LESS.SetBCLoadingType('Pressure')
LESS.SetBCLoading([0.50, 0.80,1.00])

LESS.AddConvergenceTest(Solid,'Displacements','All',5.0e-3)
LESS.AddConvergenceTest(Fluid,'Pressure','Max',5.0e-3)

# Define pressure interpolation operations
LESS.AddPressureBCInterpolation('Face', 'FluidInterface', MaxProject=.05)

#Define displacement interpolation operations
LESS.AddDisplacementInterpolation('Node', 'SolidInterface', MaxProject=0.05)
LESS.AddDisplacementInterpolation('Project', 'ProjectDisp',  Nodes=[857,1197])
LESS.AddDisplacementInterpolation('Radial', 'FwdRadialDisp', Nodes=[857])
LESS.AddDisplacementInterpolation('Radial', 'AftRadialDisp', Nodes=[1197])

LESS.AddDeformation(Fluid)
LESS.AddReMesh(Fluid,'All')

LESS.AddMonitor(Solid,'Displacements','Node','DispMonitor','Example11-MonitorDisplacement.dat')
LESS.AddMonitor(Fluid,'Pressure','Element','PressMonitor','Example11-MonitorPressure.dat')

LESS.Solve()

LESS.CloseLogFile()
LESS.CloseReportFile()
LESS.CloseAllViews()
```

**Figure 15: FSI python script for sample bore choking analysis**

The FSI python script contains the file names of the CFD and structural models, information about writing input files, executing the CFD and structural analysis, and other data required to perform a coupled FSI analysis. Figure 15 is a sample FSI python script for this bore choking analysis. It is included only to illustrate the simple form of the input.

## EXECUTION

This sample bore choking analysis used Fluent for the CFD analysis, and ABAQUS for the structural analysis. Each FSI analysis took approximately two and a half minutes on a PC with Windows XP with a 2.8 GHz processor with 512 GB of RAM. Obviously larger and more complex models will take longer to run. The short execution time illustrates the quick analysis time made possible with the automated FSI analysis procedure. The procedure can be set up to execute the CFD and structural analysis codes anywhere on the LAN.

## RESULTS

The sample bore choking analysis for this solid rocket motor showed that bore choking occurred when the elastic propellant modulus was 200 psi, but not when the elastic modulus was 500 psi. Table 2 compares several key output results.

**Table 2: Select Results from sample bore choking analysis**

| **Result** | **Model A**<br>200 psi<br>(1.4 MPa) | **Model B**<br>500 psi<br>(3.4 MPa) |
|---|---|---|
| Number of iterations | 5 | 7 |
| Head end pressure at final step | 747.55 psi<br>(5.154 MPa) | 622.83 psi<br>(4.294 MPa) |
| Maximum radial displacement at final step | -1.185 in<br>(-3.009 cm) | -0.058 in<br>(-0.147 cm) |
| Maximum axial displacement at final step | 1.724 in<br>(4.379 cm) | 0.114 in (0.289 cm) |
| Bore choked | Yes | No |

The analysis of Model A indicates continued inward deflection of the downstream slot corner, which increases in magnitude with each subsequent FSI iteration, as shown in Figure 17. The sixth CFD solution for Model A failed because the deformations applied from the previous structural analysis caused the downstream slot corner to deform past the centerline (Figure 16), indicating a choked flow condition.

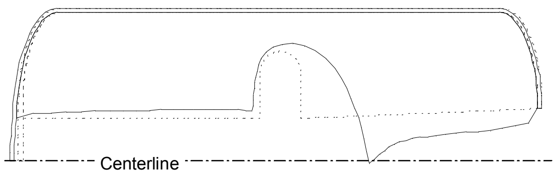


**Figure 16: Solid model with E=200 psi showing initial and final propellant deformations**

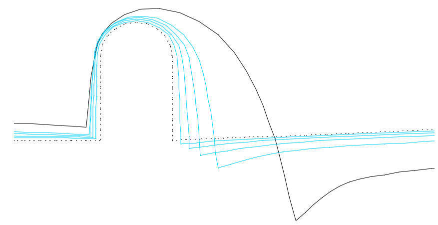

**Figure 17: Detail of slot with E=200 psi showing intermediate propellant deformations**

The analysis of Model B did not exhibit the instability of Model A; compare Figure 17 and Figure 19. The FSI solution iterated to a converged solution in seven steps. The inward deflection of the downstream slot corner did not become unstable, as it did in Model A. Both the pressure and the deformations converged to a stable solution, as shown in the plots of Figure 20 and Figure 21.

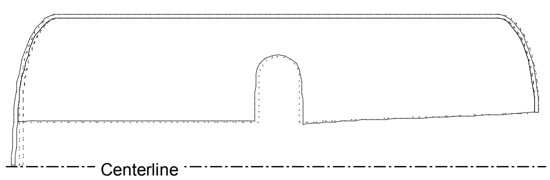


**Figure 18: Solid model with E=500 psi showing initial and final propellant deformations**

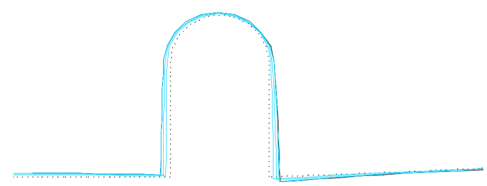

**Figure 19: Detail of slot with E=500 psi showing intermediate propellant deformations**

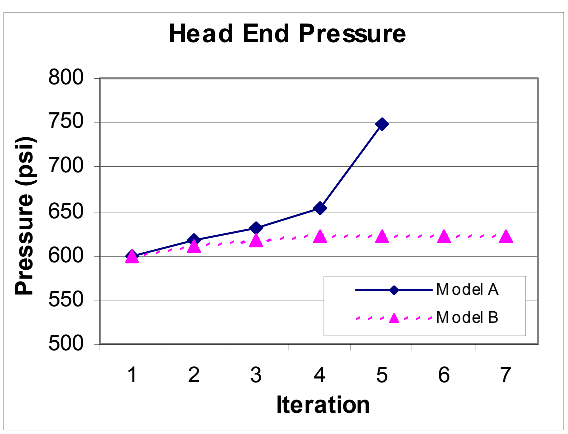


**Figure 20: Plot of head end pressure for Model A and Model B**

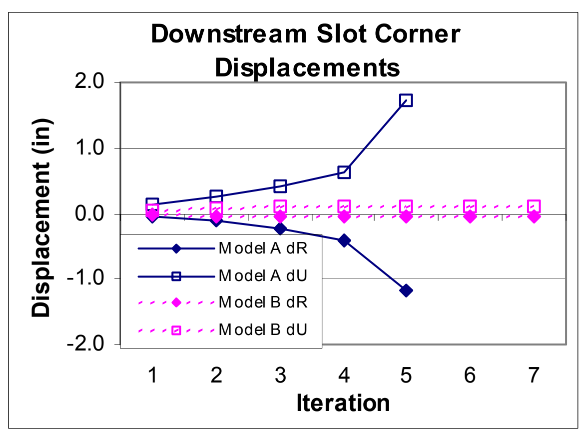


**Figure 21: Plot of downstream slot corner displacement for Model A and Model B**

Additional FSI solutions were performed for this motor design for several different values of the propellant elastic modulus between 200 psi (1.4 MPa) and 500 psi (3.4 MPa). The results of these analyses indicate that the elastic modulus of the propellant must be 350 psi or above to prevent motor failure by bore choking. The results from these analyses are summaries in Table 3 and Figure 22.

**Table 3: Motor Stability listed for values of elastic modulus and number of iterations**

| Elastic Modulus (psi) | Number of Iterations | Motor Status |
|---|---|---|
| 500 | 7 | Stable |
| 400 | 12 | Stable |
| 350 | 19 | Stable |
| 325 | 20 | Unstable |
| 300 | 11 | Unstable |
| 250 | 7 | Unstable |
| 200 | 5 | Unstable |

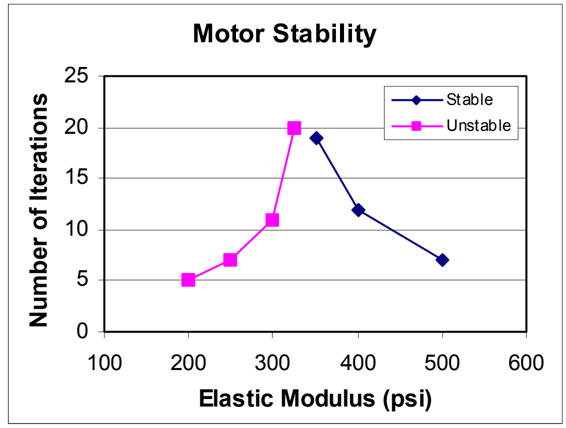


**Figure 22: Plot of number of iterations vs. elastic modulus**

## FUTURE DEVELOPMENT

The future development plans for the FEM Builder software package includes the addition of FEM Builder solver for thermal analysis to the coupled FSI solution. Besides adding a thermal solver, a 2D thermal ablation finite element code will be developed and used in this coupled analysis environment to model the thermal ablation of a SRM nozzles.

## CONCLUSIONS

The automated FSI analysis procedure developed at ATK Thiokol Propulsion externally couples CFD and structural analyses using Fluent® and ABAQUS®, respectively. The approach is straightforward and fairly simple to use. The automated feature of the FSI analysis procedure makes this method very fast and economical to use, especially when compared to doing the same analysis manually. The automated FSI procedure has application in analysis, as well as design.

## REFERENCES

1. Python programming language at www.python.org

**NOMENCLATURE, ACRONYMS, ABBREVIATIONS**

| | |
|---|---|
| CFD | Computational Fluid Dynamics |
| FE | Finite Element |
| FEM | Finite Element Model |
| FSI | Fluid Structure Interaction |
| LAN | Local Area Network |
| SRM | Solid Rocket Motor |
| RSRM | Reusable Solid Rocket Motor |