IN-07
2935
112 P

NASA Contractor Report 194460

Structural/Aerodynamic Blade Analyzer (SAB)

User's Guide-Version 1.0

M.R. Morel
Sverdrup Technology, Inc.
Lewis Research Center Group
Brook Park, Ohio

Prepared for
Lewis Research Center
Under Contract NAS3-25266

National Aeronautics and
Space Administration

Preface

This report is the User's Guide for the Structural/Aerodynamic Blade (SAB) analyzer. The system provides an automated tool for the static-deflection analysis of turbomachinery blades. This is accomplished by coupling a finite element structural code and an aerodynamic solver. Currently, SAB version 1.0 is interfaced with MSC/NASTRAN (to perform the structural analysis) and PROP3D (to perform the aerodynamic analysis). This document is meant to serve as a guide for the operation of the SAB system with specific emphasis on its use at NASA Lewis Research Center (LeRC). A detailed explanation of the structural and aerodynamic analyses are beyond the scope of this document, and may be found in the references. The SAB system has been developed by the NASA LeRC Structural Dynamics Branch.

Please note that the SAB system is being made available strictly as a research tool. Neither NASA LeRC, nor any contractors, nor grantees that have contributed to the code development, assume liability for application of the code beyond research needs.

The user's guide was designed to assist the user in utilizing the SAB system. This guide consists of six chapters: (1) an introduction which gives a summary of SAB; (2) SAB's methodology, component files, links, and interfaces; (3) input/output file structure; (4) setup and execution of the SAB files on the Cray computers; (5) hints and tips to advise the user; and (6) an example problem demonstrating the SAB process. In addition, four appendices are presented to define the different computer programs used within the SAB analyzer and describe the required input decks.

Any questions or related items concerning this computer code can be directed to the Structural Dynamics Branch at the NASA Lewis Research Center, Cleveland, OH 44135.

Acknowledgement

The following people have made contributions to this report and the development of SAB: Oral Mehmed and Dave Janetzke of NASA LeRC, Rakesh Srivastava of University of Toledo, and the Sverdrup team of Rick Shimko, Rich August, Ron Morris, and Osamu Yamamota.

TABLE OF CONTENTS

1. Introduction

1.1 Summary

This document describes the use and execution of SAB (Structural/Aerodynamic Blade analyzer) version 1.0, with specific emphasis on its use at NASA LeRC. The SAB system has been devised to meet the needs of propulsion engineers for the static analysis of advanced turbomachinery blades. The automated system couples a finite element (FE) solver and an aerodynamic (aero) solver to predict the steady-state deflections. Due to the coupling of the solvers, both rotational and aerodynamic loads can be considered during the analysis. It is the intent of this report to describe the operation and environment in which to execute the SAB system.

The SAB system was specifically tailored to account for the structural and geometric characteristics of turbomachinery blades. The FE solver, MSC/NASTRAN [1], uses a geometrically nonlinear (large displacement) analysis which considers centrifugal softening and stiffening effects [2]. For the aerodynamic analysis, a 3-dimensional hybrid scheme (PROP3D, [3]) was developed for solving the Euler equations in order to analyze the blade efficiently. The user is not restricted to MSC/NASTRAN or PROP3D for the coupled analysis but were chosen for their convenience and availability.

The SAB system is currently implemented on the Cray Y-MP and X-MP supercomputers at NASA LeRC using the UNICOS operating system. There are four types of files that constitute the SAB system:

(1) Executable codes which do the calculations, written in FORTRAN. Note, the FE solver, MSC/NASTRAN, is a commercial software package.
(2) Shell files used to link the executable files, written in Bourne-shell.
(3) Job control files which submit executable files to the computer, consisting of NQS commands.
(4) Three input files in which the user must create:
    (a) FE input deck of blade
    (b) Aero solver input deck for the blade rows
    (c) Parameter file containing miscellaneous data

2. SAB System

2.1 SAB Overview

The SAB system integrates a structural finite element analysis code (FE solver) and an aerodynamic code (aero solver) into an iterative approach to effectively model the characteristics of turbomachinery blades [3-5]; the methodology is depicted in Figure 2.1-1. In the following description of the methodology, the generic terms FE and aero solver are utilized in place of actual code names that were used. This done to demonstrate that any comparable finite element or aerodynamic code can be incorporated into the methodology.

A finite element model of the blade, *e.g.* a typical model is shown in Figure 2.1-2(a), is prepared for input. Not only should the user have an accurate finite element geometric model to represent the blade but initial input should include:

1.) the blade rotational speed (Ω)
2.) the nominal blade setting angle ($\beta_{\frac{3}{4}R}$)

in order to determine the steady state displacements (ΔX, ΔY, ΔZ) under the centrifugal loading.

After the structural analysis of the blade, step 1, the steady state geometry from the FE solver is transformed onto the blade surface grid points used by the aero solver; this is done by Interface I in step 2 and depicted in Figure 2.1-2. Since the two grids are not identical, an interpolation of the data is required. The aerodynamic mesh generation code establishes the computational mesh (step 3) around the deflected blade, shown in Figure 2.1-3. With the appropriate freestream Mach number (M) and advance ratio, the aero solver proceeds to compute the propeller flow field (step 4). Interface II (step 5) extracts the surface pressures from the aerodynamic data, and the differential pressure (ΔP) is transformed onto the centroids of the structural elements. A new deflection under the initial centrifugal loads and the newly created aerodynamic loads (transformed into pressure loads) is computed by the FE solver in step 6. This completes the first iteration.

Interface I establishes a new aerodynamic blade geometry and the new aerodynamic mesh is generated for the centrifugally deformed geometry. An updated set of steady aerodynamic loads are calculated by the aero solver. The structural-aerodynamic interaction is iterated (steps 2-7) until convergence is reached, *i.e.* until the change in deflection from the previous iteration is within a specified tolerance compared to the current iteration. This is presently accomplished by monitoring the angle of a blade section at a specified radial location, preferably near the radial location where maximum displacement occurs. If the blade angle diverges instead of converges, the blade is statically unstable.

Figure 2.1-1: Structure-Aerodynamic Integration Approach

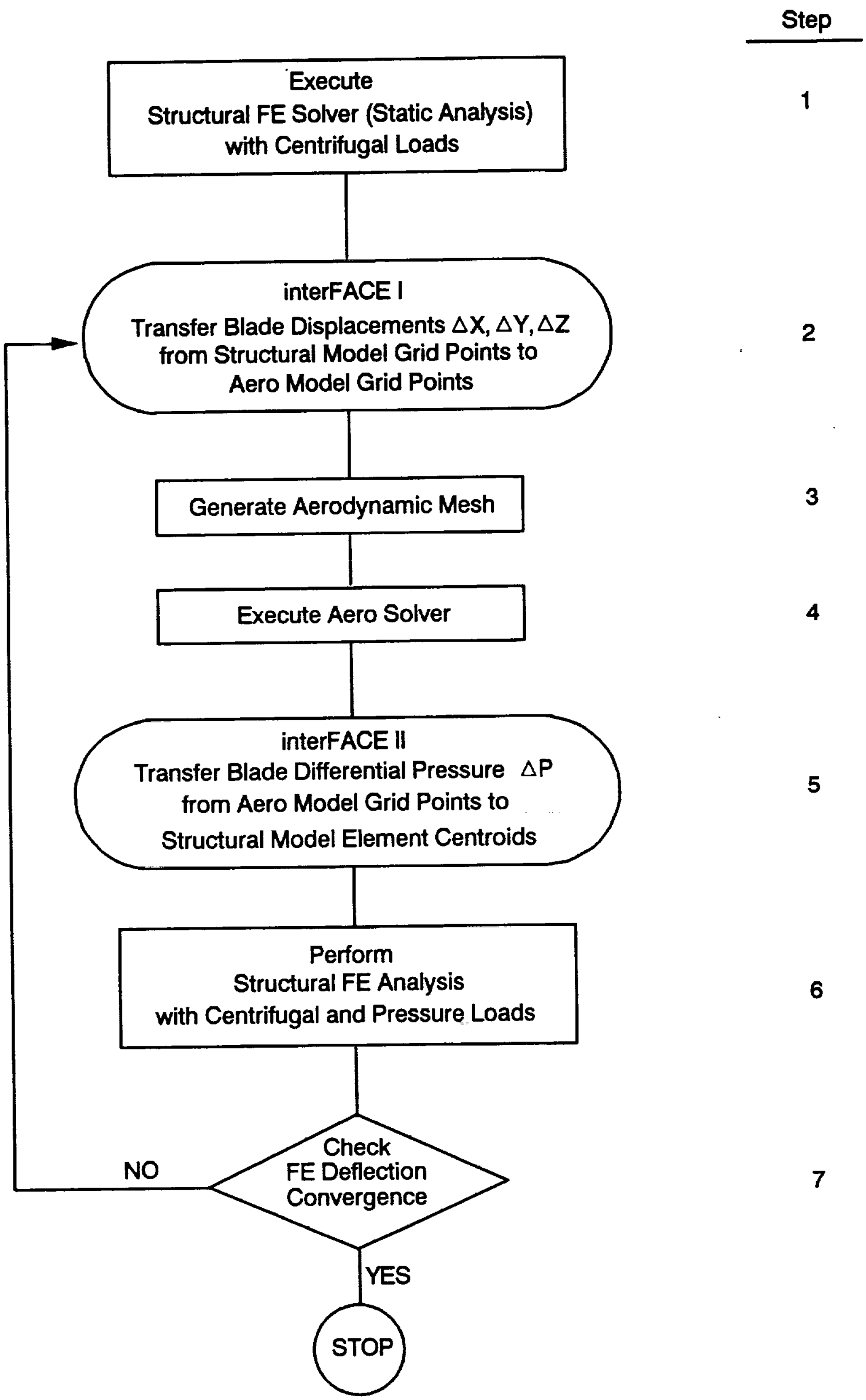


Figure 2.1-2: Example of the Transformation Between the FE Model and the Aero Plan Form View shown in the Standard Coordinate System: (a) FE Mesh of Blade; (b) Aero Plan Form View of Blade; (c) Standard Coordinate System Internally Set in SAB.

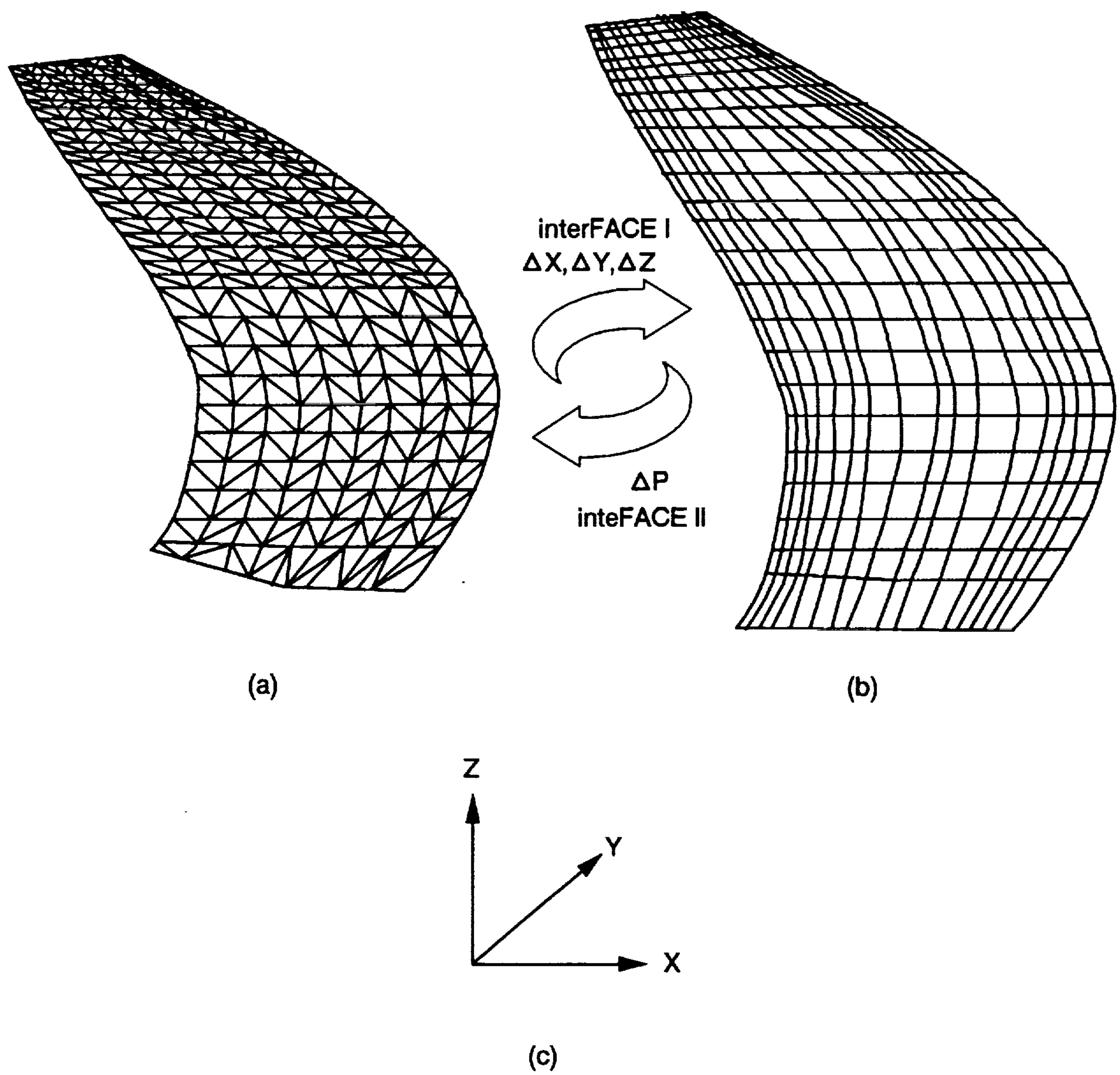
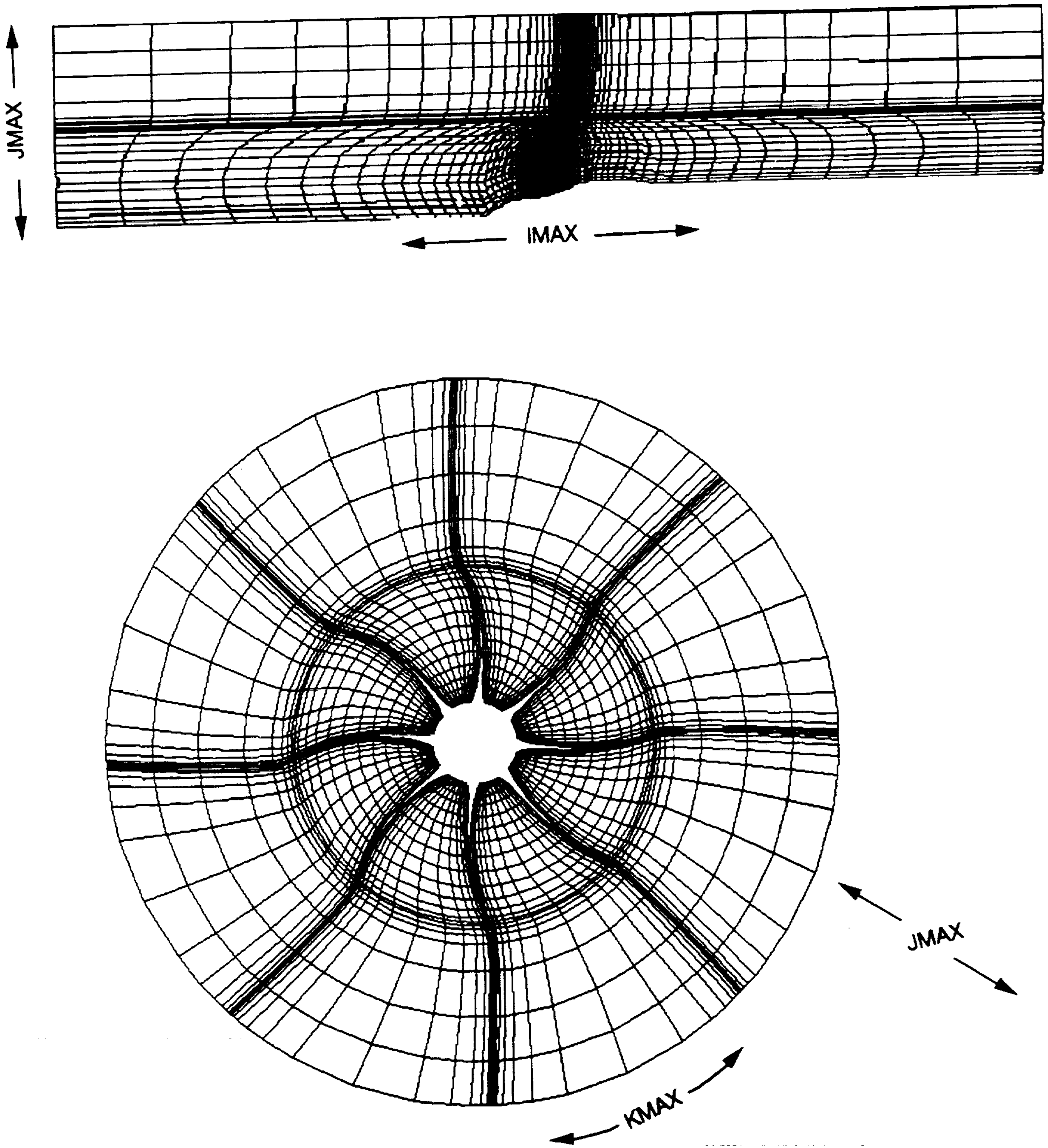


Figure 2.1-3: An Example of the Grids Used in the Aero Solver



2.2 SAB Components

In this section the individual components of the SAB package are presented and briefly described. These components make-up the SAB system presented in the previous section. Excluding the input decks, Table 2.2-1 list the files used in the SAB system for a batch mode execution. A more extensive explanation and file listings can be found in Appendix A. As previously mentioned, the three groups of files that create the SAB system are:

1.) Job control files (*.job) which submit the executables to the Cray Y-MP and X-MP computers
2.) Shell files (*.sh) that automate the execution
3.) Object files (*.o) in compiled format which perform the calculations

The * symbol represents all files with the corresponding extension.

The job control or shell modules are either NQS [6] and/or UNIX [7] commands. These two groups of files are labelled script files within this document. The object files are FORTRAN codes that have already been compiled on the Cray Y-MP and are ready to be loaded and executed. The group of FORTRAN object files are labelled source files in this document. In all the user will need a total of 14 script and source files to create the SAB batch environment.

2.2.1 Script Files

start_sab.sh
This file starts the SAB system by creating a working directory, and copying all scripts, sources, and input files into the working directory. The entire execution of the SAB process will be executed in this directory. All files (scripts, sources, inputs and outputs) will be stored in the working directory. After transferring all files, the SAB iterative process is initiated by executing two shells. The first shell is to make the structural analysis input (**mk_nas.sh**) by merging two files and the second to submit the input for execution (**run_nas.sh**). Refer to Appendix A, pg. 50, for a more thorough description and listing of this UNIX shell file.

mk_nas.sh
This shell file makes an executable MSC/NASTRAN input deck by combining the MSC/NASTRAN input deck created by the user and the job file deck **nas_cards.job**. MSC/NASTRAN [2] is a commercial finite element code used for the structural analysis of the blade. When the MSC/NASTRAN input deck is inserted into **nas_cards.sh** an output file called **nastran.job** is created. The user is referred to Appendix B for further information on MSC/NASTRAN and Appendix A, pg. 52, for a more detailed description of **mk_nas.sh**.

nas_cards.job (nastran.job)
The FE solver setup is a mixture of NQS and UNIX commands to invoke the execution of MSC/NASTRAN. The MSC/NASTRAN input deck is inserted into **nas_cards.job** by

mk_nas.sh and the end result is the file **nastran.job**. Upon the successful completion of the structural analysis, **face.job** is submitted for execution. Appendix A, pg. 57, contains the file listing and further details.

run_nas.sh
This UNIX shell starts the execution of the MSC/NASTRAN code by executing the **nas_xmp.sh** module. The description and listing of **run_nas.sh** is located in Appendix A, pg. 53.

nas_xmp.sh
The **nas_xmp.sh** UNIX shell transfers the MSC/NASTRAN input deck and related NQS commands from the CRAY Y-MP computer to the Cray X-MP computer at NASA LeRC for execution. The machine transition is done in batch mode and is transparent to the user. This must be done because MSC/NASTRAN is only available on the Cray X-MP at NASA LeRC. Upon completion, the output files are transferred back to the working directory on the Cray Y-MP. A listing and more detailed description is found in Appendix A, pg. 54.

face.job
This script file is a combination of NQS and UNIX commands. **face.job** initiates two shell modules: (1) **conv_chk.sh** which checks if the convergence criteria is met; and (2) **face1.sh** which transforms the structural deformations from the FE gird points to the aerodynamic geometry. Also, **prop3d.job** is executed in order to start the aerodynamic analysis. A listing and more detailed description is found in Appendix A, pg. 59.

conv_chk.sh
A UNIX shell to check the convergence by comparing the current blade deflections from MSC/NASTRAN with the previous iteration's deflections. The source file **converg.o** is called to perform this task. If convergence is met the SAB system ceases operation, if not met then another loop is initiated until the convergence criteria is satisfied. Refer to Appendix A, pg. 61, for the listing and description of this file.

face1.sh
This UNIX shell interpolates the nodal displacements (ΔX, ΔY, ΔZ) from the structural model onto the aerodynamic geometry definition, refer back to Figure 2.1-2, for input into the aerodynamic code. The shell calls the source **face1.o** to accomplish this assignment. Appendix A, pg. 63, contains the file listing and further description.

prop3d.job
The main function of the **prop3d.job** file is to execute the **prop3d.o** source file and the **face2.sh** UNIX shell. The **prop3d.o** is the compiled version of the PROP3D code for the aerodynamic analysis.

face2.sh
A UNIX shell to execute the interpolation of the aerodynamic results into differential pressures (ΔP) that are applied on the structural model for the updated input deck, refer back to Figure 2.1-2. Also, the differential pressures in the form of PLOAD cards [2] are appended to the MSC/NASTRAN input deck. The source file **face2.o** is called to perform this task. Further details and a file listing are located in Appendix A, pg. 67.

2.2.2 Source Object Files

converg.o
A FORTRAN code to check the convergence criteria for the current MSC/NASTRAN results, $(i+1)^{th}$, and the previous iteration's results, $(i)^{th}$, are compared. A special output (MSC/NASTRAN punch) file of the blade displacements (ΔX, ΔY, ΔZ) is used with the nodal coordinates (X, Y, Z) to calculate the blade angle of the deflected shape (ΔX+X, ΔY+Y, ΔZ+Z) at a specified radial location. In general, the blade angle is defined by the following equation:

$$\text{blade angle} = \arctan\left[\frac{\text{XLE} - \text{XTE}}{\text{YLE} - \text{YTE}}\right]$$

where (XLE,YLE) and (XTE,YLE) are the coordinates of the leading and trailing edges. The radial location should correspond to the position of maximum displacement. The current blade angle is compared to the previous blade angle to determine if the convergence criteria is met. The flow chart for this code is displayed in Figure 2.2-1.

face1.o
The structural deformations are computed at the finite element model grid points, aerodynamic forces are computed on the blade surface grid points. To make the exchange of data between the structural and aerodynamic analyses possible, the nodal displacements from the structural model are interpolated onto the grid points of the aero chord plane. This FORTRAN code interpolates the structural nodal deflections onto the aero chord plane by a cubic spline interpolation. This must be done because of differences in the number and distribution of grid points between the two models, refer to Figure 2.1-2. Finally, the flow chart for this code is located in Figure 2.2-2.

face2.o
A FORTRAN code to convert the pressures from the aerodynamic results (PROP3D) into the differential pressure (ΔP) applied to the corresponding grid surface (aero chord) and then transferred onto the elements of the FE model using a cubic spline interpolation, refer to Figure 2.1-2. Finally, for the structural analysis, pressure loads at the centroids of the elements are generated which will be appended to the structural analysis input. The flow chart for this code is displayed in Figure 2.2-3.

prop3d.o
A FORTRAN code used to perform the aerodynamic analysis of the blade. The PROP3D code was chosen to accomplish the aerodynamic part of the SAB process. The code description and input can be found in reference [3] and Appendix C, pg. 70.

Table 2.2-1: Files Required by the SAB System for Execution in Batch Mode

Script Modules (Job Control and Shell)	Source Files (Compiled FORTRAN Object Modules)
start_sab.sh	converg.o
mk_nas.sh	face1.o
nas_cards.job	face2.o
run_nas.sh	prop3d.o
nas_xmp.sh	
face.job	
conv_chk.sh	
face1.sh	
prop3d.job	
face2.sh	

Figure 2.2-1: Flow Chart of converg.o for the Convergence Check

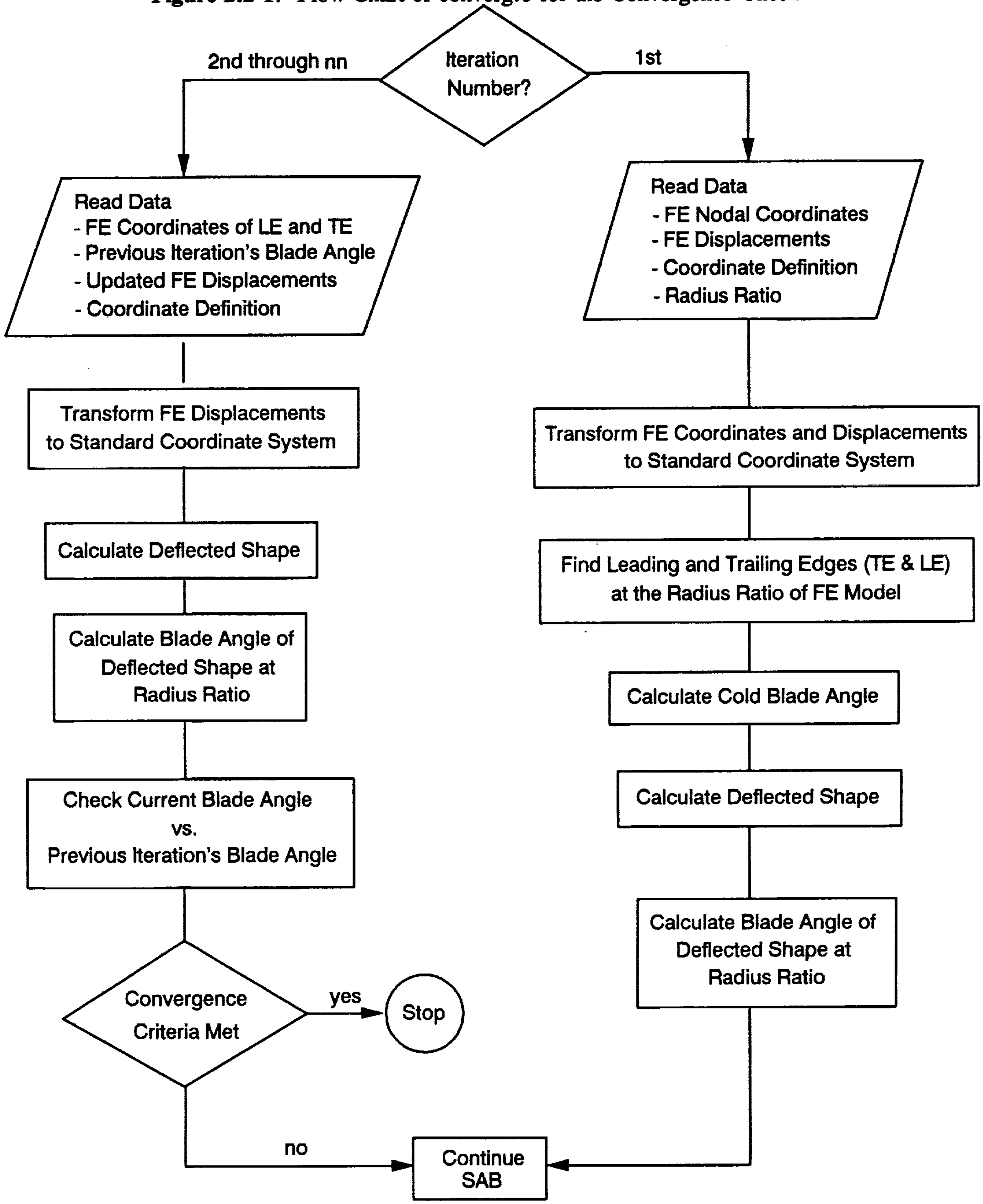


Figure 2.2-2: Flow Chart of face1.o

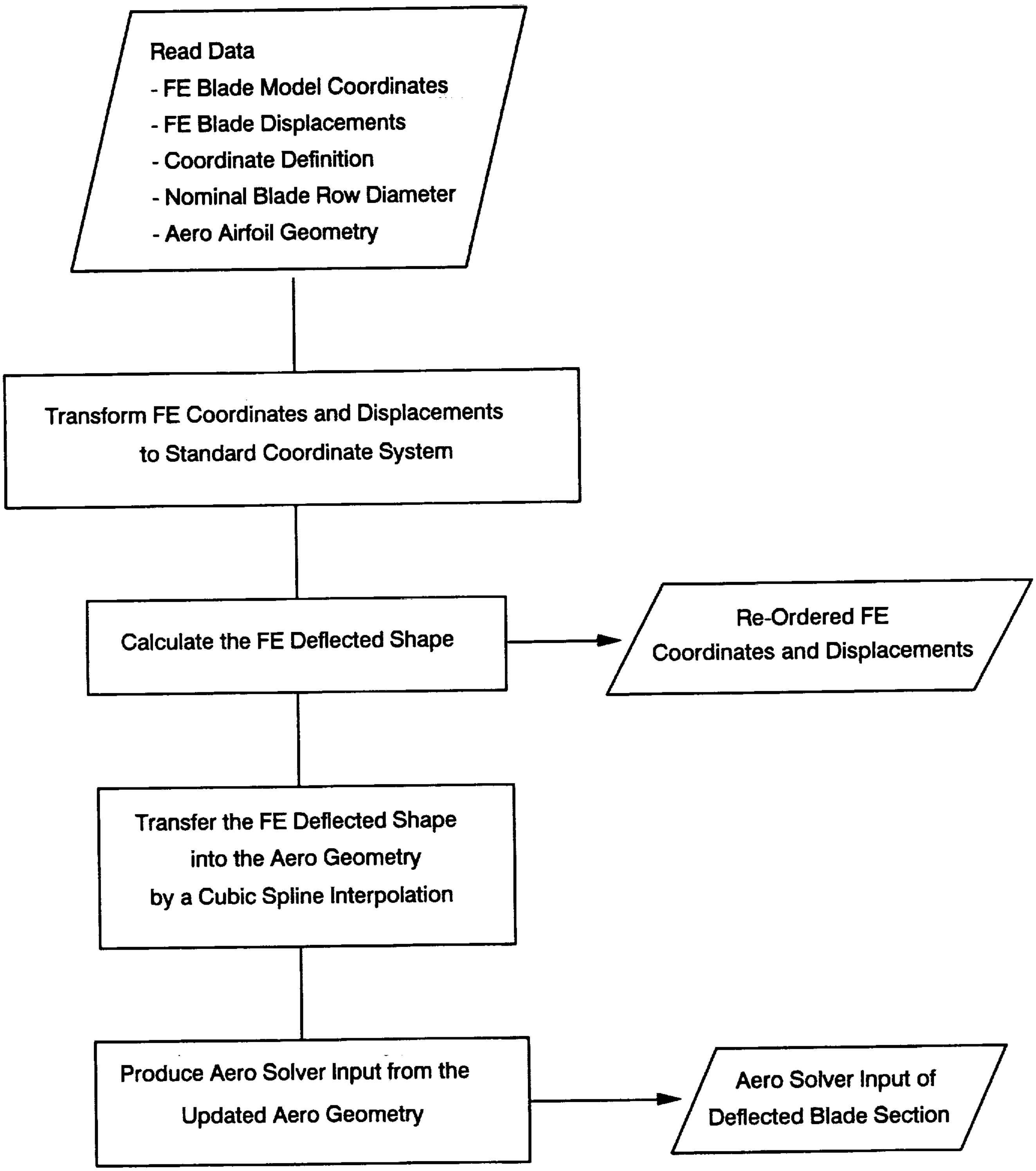
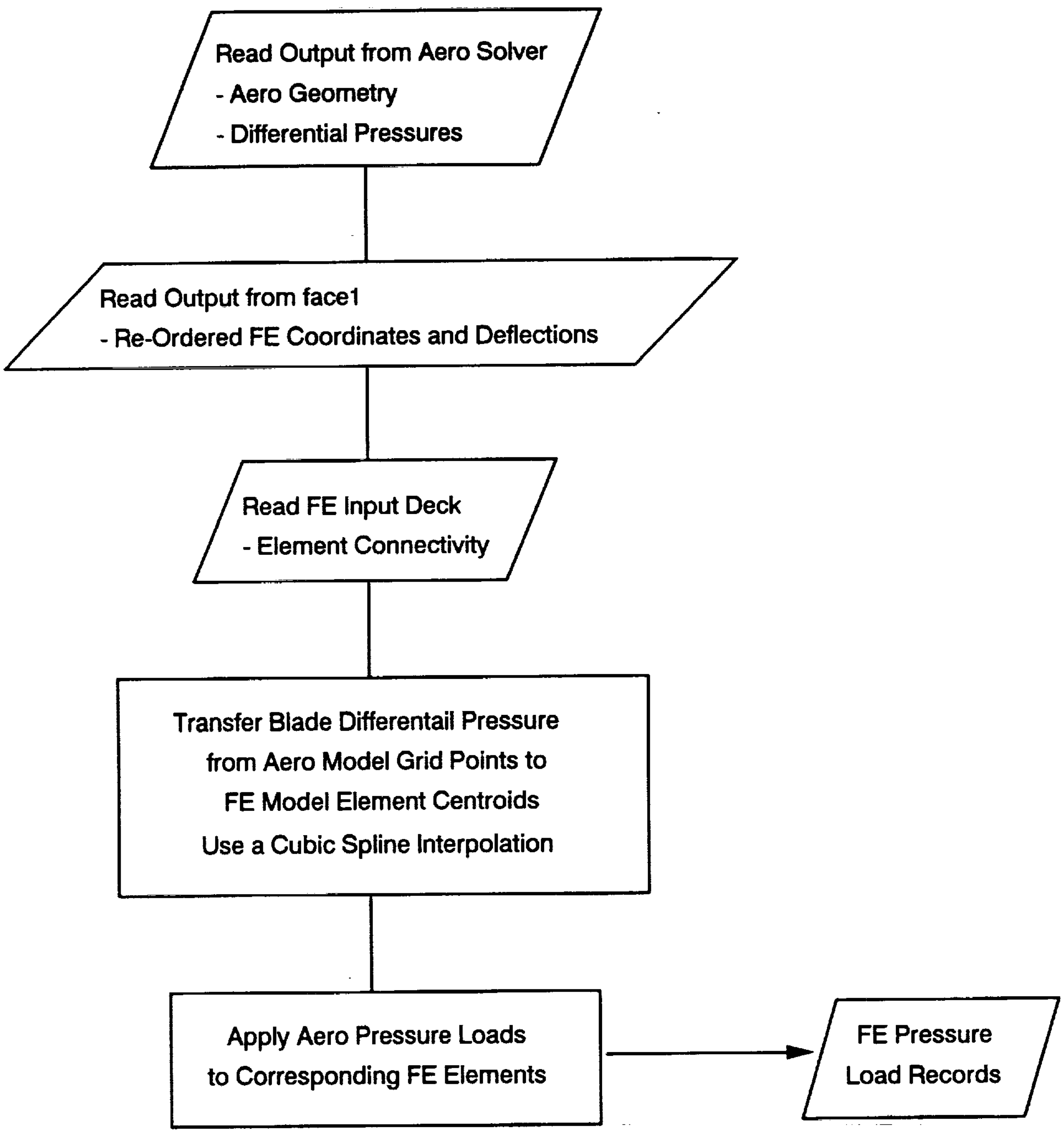


Figure 2.2-3: Flow Chart of face2.o



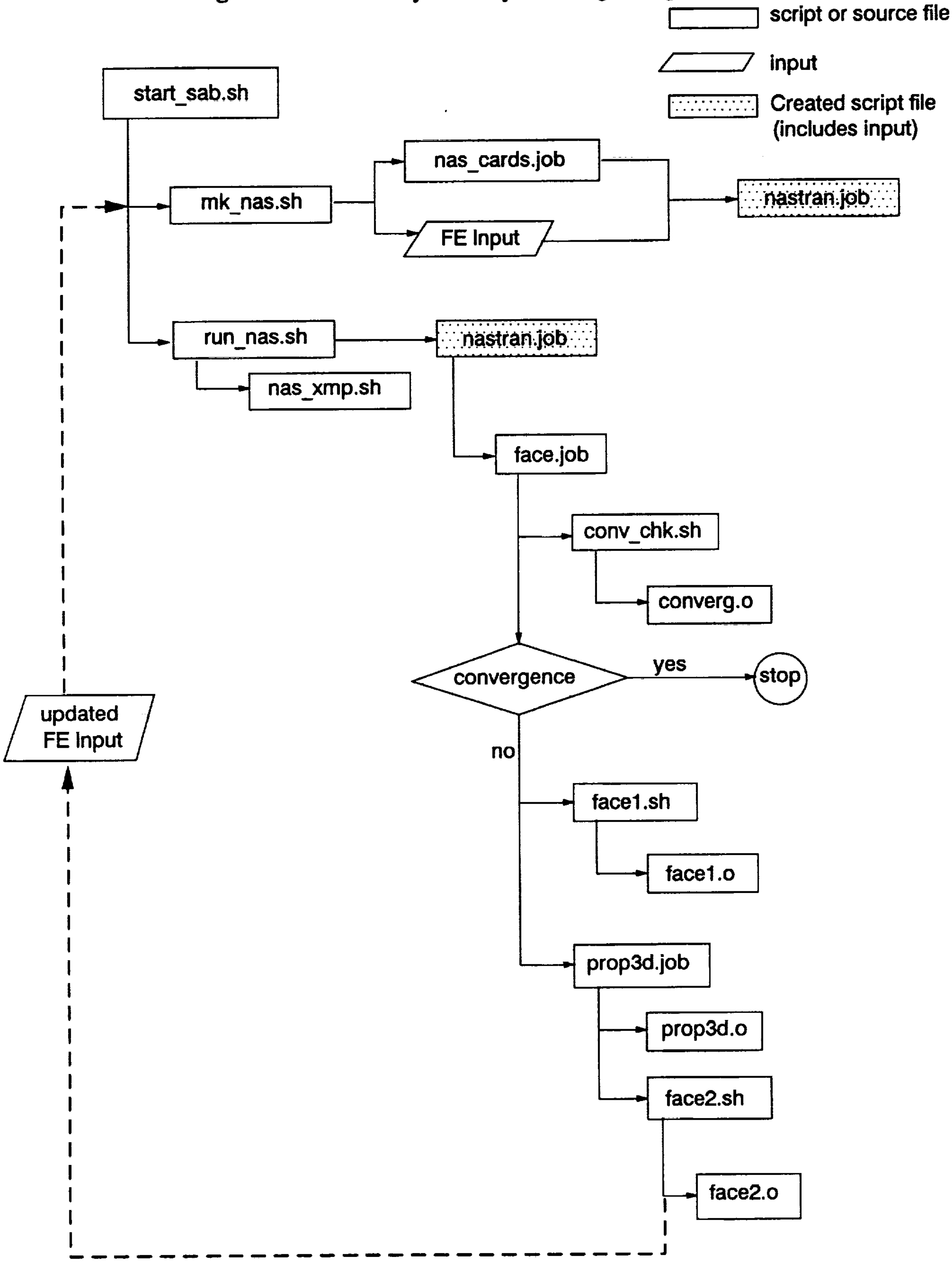
2.3 Code Links and Interfaces

Implementing the procedure described in Section 2.1 and using the files from Section 2.2, the overall layout of the SAB iterative process is shown in Figure 2.3-1. To begin the SAB process, the **start_sab.sh** file is submitted to the Cray Y-MP. This file generates a working directory, copies the required input, source, and script files to the working directory, creates the nastran.job file, and submits the file for execution. For the creation of the **nastran.job** file, the shell **mk_nas.sh** is executed to combine the FE input deck and **nas_cards.job**. Upon completion, the structural analysis input (**nastran.job**) is submitted for execution by the shell **run_nas.sh** which initiates the **nas_xmp.sh** file. The **nas_xmp.sh** file transfers the FE input deck to the Cray X-MP computer for execution, after completion, returns control back to **nastran.job** file on the Cray Y-MP.

The second part of **nastran.job** submits the file **face.job** for execution on Cray Y-MP. **face.job** has the responsibility of checking convergence, transferring the structural deflected shape onto the aerodynamic blade shape definition (aero geometry), and submitting the PROP3D code for execution. This is done by initiating the following scripts: **conv_chk.sh, face1.sh**, and **prop3d.job**. The shell **conv_chk.sh** checks for convergence, first time through the loop **conv_chk.sh** will immediately return control to **face.job**, otherwise calls the source file **converg.o**, and, if convergence is met, the process stops. If convergence is not met, control is returned back to **face.job** so that **face1.sh** can be executed next. The file **face1.sh** executes **face1.o** in order to interpolate the deflected shape onto the aero geometry of the aerodynamic model. Finally, **prop3d.job** is submitted to run the PROP3D code, *i.e.*, aerodynamic analysis.

Within the **prop3d.job** file **face2.sh** is called to be executed upon the completion of the PROP3D code. The script **face2.sh** has the responsibility of transferring the differential pressures from the aerodynamic model to grid point pressures that can be used in the structural analysis model. This is task is performed by the source file **face2.o**. The structural analysis input is updated with the new pressure loads and the script **mk_nas.sh** is recalled to make an updated nastran.job file. The **nastran.job** is submitted for re-analysis by **run_nas.sh**. The current results are compared to the previous results to determine if another iteration is warranted. This iterative process is continued until the convergence criteria is met in **conv_chk.sh**.

Figure 2.3-1: SAB System Layout Using the Specific Files



3. SAB Input/Output Files

3.1 Introduction

This chapter defines the input/output files in the SAB system and describes their content. The user is responsible for producing three external input files to begin the SAB procedure, refer to Figure 3.1-1. Fifteen different types of output files are generated throughout the process and are shown in five categories. Boxes containing two file names signify that the bottom file name is renamed to the top file name during the SAB process. The *nn* character denotes the iteration number in which the output file was generated, as a result, some output files are re-generated for each iteration. Five of the fifteen output files are used as input to the executable codes that make up the SAB system and three additional output files are entered as input to be renamed with the corresponding iteration number.

3.2 Input Files

Three input files are required to execute the SAB process:

(1) MSC/NASTRAN input deck - a structural finite element input, used to perform the structural analysis part of the SAB system. The user is referred to references [1-2] to create the MSC/NASTRAN input deck and a further description of MSC/NASTRAN is in Appendix B, pg. 69.

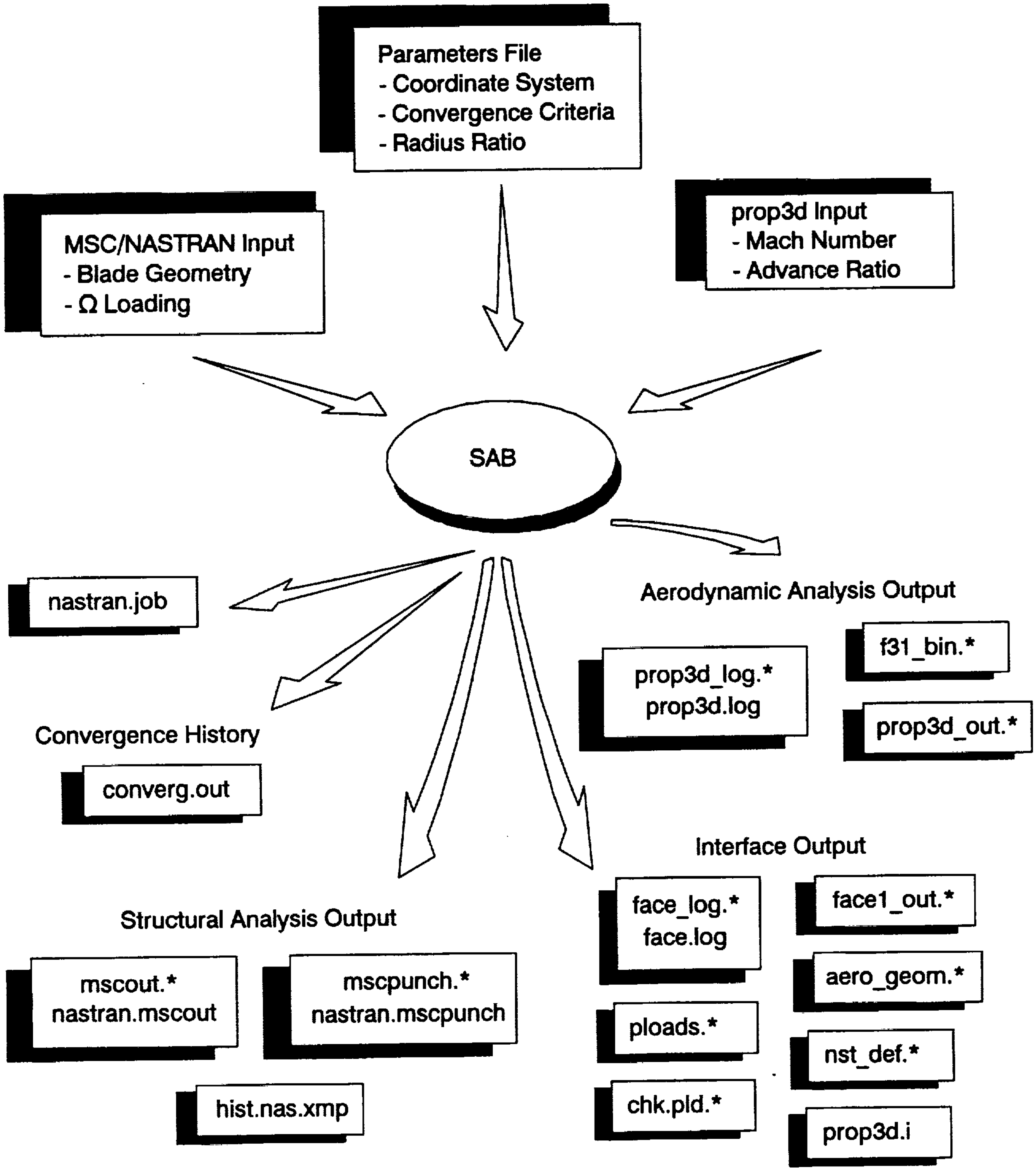
(2) PROP3D input deck - the aerodynamic code input [3]. The instructions for creating the input deck are located in Appendix C, pg. 70.

(3) parameter input file - miscellaneous information required during the SAB process. The following is a list of parameters defined in this file:

- coordinate definition: defines the spanwise and chordwise directions
- radius ratio: location along the span of the blade
- tolerance level: convergence criteria
- diameter of the rotor
- P_{dim}: normalization factor for aerodynamic pressures

For specific instructions on preparing the parameter file input refer to Appendix D, pg. 102.

Figure 3.1-1: Overall Input/Output File Structure of the SAB Process



3.3 Output Files

Due the number of computer codes and iterative process within SAB, a large number of different output files (15) are generated. In addition, five of the output files are used as input files to other codes in the iterative approach. All output files are located on the Cray Y-MP, where execution is initiated.

Listed in Table 3.3-1 are the output files created during the SAB process. The *n* character represents a numerical number given to the output file to distinguish when the file was produced. Except for the MSC/NASTRAN output, the *nn* defines the iteration number when the output file was created. The numerical numbers given to the MSC/NASTRAN output file is equivalent to one plus the iteration number. As a result, all output files with the *nn* characters are generated for each iteration with the number of MSC/NASTRAN outputs equal to one plus the number of iterations. For example, if a total of 8 iterations are needed for convergence then 8 output files would exist for the prop3d_out.*nn* file, numbered consecutively from 1 to 8. Also, Tables 3.3-2, through 3.3-7 describe the contents and format of specific output files for the user.

Four output files have similar names in parentheses after them, these files undergo a name change during the SAB process in order to assign a numerical number to the file name. This is done to organize the numerous output files for the user. The current file or working copy of the output file is located within the parentheses and the final name (renamed version) is located above it; centered above the output files is the script file that produces the output file. Finally, Figure 3.3-1 shows the global illustration of the input/output files assigned to their respective files.

Table 3.3-1: Output Files Produced by SAB

File Name	Description	Input for Script
mk_nas.sh		
nastran.job	This file is a combination of the Cray X-MP NQS commands, current MSC/NASTRAN input file, and UNIX commands. This file also has the responsibility to execute the face.job file to the Cray Y-MP.	run_nas.sh conv_chk.sh
run_nas.sh		
hist.nas.xmp	This file contains information on the status of the MSC/NASTRAN job on the Cray X-MP, *e.g.*, if the job was successfully transferred to the Cray X-MP, the job number and related information would appear. For every job submitted to the Cray X-MP, the information is appended to this file along with the date and time.	
nas_xmp.sh (MSC/NASTRAN)		
mscnast.*nn* (nastran.mscout)	Standard output file generated by MSC/NASTRAN. The contents of this file can be controlled by the input deck, the user is referred to the MSC/NASTRAN manual [1] for additional instructions. This file undergoes a name change only.	conv_chk.sh
mscpunch.*nn* (nastran.mscpunch)	A file containing nodal displacements produced by the execution of MSC/NASTRAN. This file must be specified in the MSC/NASTRAN input deck to be generated. This file is used as an input file and undergoes a name change. Refer to Table 3.3-2 for the contents and format.	conv_chk.sh face1.sh
face.job		
face_log.*nn* (face.log)	System file to inform user of the status of the different shells (conv_chk.sh, face1.sh, prop3d.job) submitted from this job file. This file undergoes a name change only.	prop3d.job

File Name	Description	Input for Script
	conv_chk.sh (converg.o)	
converg.out	Information describing the status of convergence at each iteration and informs the user that convergence has occurred. Refer to Table 3.3-4 for the output information in this file.	conv_chk.sh
	face1.sh (face1.o)	
face1_out.*nn*	Pertinent output from the face1.o execution. Information provided in this file is used mainly to check if data was transferred properly.	
aero_geom.*nn*	Aerodynamic mesh used in the PROP3D input. Refer to Appendix C Records 24-29, pgs. 87-95, for details on the contents of this file. This file is equivalent to Records 24-29.	
prop3d.i	PROP3D input deck with the updated aerodynamic geometry replacing the previous iteration's geometry. This file is combination of the original PROP3D input deck (prop3d_i.cold) and the newly created aerodynamic geometry (aero_geom.*nn*).	prop3d.job
nst_defl.*nn*	The blade model deflected shape (given by the nodal points) calculated from the cold shape and displacements of the current loading condition. Refer to Table 3.3-3 for the format of this file.	face2.sh
	prop3d.job (prop3d.o)	
prop3d_log.*nn* (prop3d.log)	System output file to inform the user of the status of the PROP3D code and face2.sh that are executed within the prop3d.job file. This file undergoes a name change only.	conv_chk.sh
prop3d_out.*nn*	Standard PROP3D output file. Refer to Table 3.3-5 for further details.	
f31_bin.*nn*	Binary output file produced by the PROP3D code used in the calculations of the pressure loads. In Table 3.3-6 the information required by face2.sh is described. Additional information exists in this file but is not shown. This file can also be used to restart the PROP3D code.	face2.sh

File Name	Description	Input for Script
	face2.sh (face2.o)	
ploads.*nn*	PLOAD cards (*i.e.*, MSC/NASTRAN pressure loads, refer to [1]) generated from the aerodynamic pressure results that are appended to the updated MSC/NASTRAN deck (nastran.job). Pressure loads are assigned to each element of the finite element model. Refer to Table 3.3-8 for the format and contents of this file.	
chk.pload.*nn*	This file contains the data to chart the progress of the pressure transformation. Pressures produced by the aerodynamic code are printed along with the corresponding pressure loads on the structural model.	

Table 3.3-2: Contents and Format of the MSC/NASTRAN Punch File (mscpunch.*nn*)

Record 1:	TITLE	(80A1)
Records 2 to 6:	SUBTITLE	(80A1)
Records 7 to ntot+6:	NODID,TYPE,(DATA(j),j=1,3), DUMMY,(DATA(j),j=4,6)	(I10,A8,3E18.6,/, A19,3E18.6)

Parameter	*Description*
TITLE	Descriptive title taken from MSC/NASTRAN input deck TITLE card.
SUBTITLE	Information pertaining to the current analysis.
NODID	Node identification number.
TYPE	"G" indicates that the information supplied is at node points.
DATA	Result quantities of displacements ($\Delta X,\Delta Y,\Delta Z,\theta_x,\theta_y,\theta_z$).
DUMMY	Continuation characters.
ntot	Total Number FE Nodal Points * 2.

Table 3.3-3: File Containing the FE Coordinates and Displacements (nst_defl.*nn*)

Record 1:	IMAX,JMAX	(2I5)
Record 2:	TITLE	(80A1)
Records 3 to nnodes:	NODID,(DATA(j),j=1,6)	(I5,3X,6F10.5)

Parameter	*Description*
IMAX	Number of chords in spanwise direction of the FE blade model.
JMAX	Number of slices in the chordwise direction of FE blade model.
TITLE	Header descriptions of the data.
NODID	Node identification number.
DATA	Nodal coordinates and corresponding displacements ($X,Y,Z,\Delta X,\Delta Y,\Delta Z$).
nnodes	Total number of nodes (IMAX * JMAX).

Table 3.3-4: Contents of converg.out File

All information corresponds to the radius ratio supplied in the parameter input file

	Initial Data	
Record 1:	LENOD,TENOD	(A40,I8,I8)
Record 2:	(CORDLE(j), j=1,3)	(A40,3F8.4)
Record 3:	(CORDTE(j), j=1,3)	(A40,3F8.4)
Record 4:	CANGLE	(A40,F8.4)
	Records 5 to 7 are Repeated for Each Iteration	
Record 5:	ΔXLE,ΔYLE,ΔXTE,ΔYTE	(A40,4F8.4)
Record 6:	DFXLE,DFYLE,DFXTE,DFYTE	(A40,4F8.4)
Record 7:	DANGLE	(A40,F8.4)

Parameter	*Description*
LENOD	FE node identification of the leading edge point.
TENOD	FE node identification of the trailing edge point.
CORDLE	FE nodal coordinates of the leading edge (X,Y,Z).
CORDTE	FE nodal coordinates of the trailing edge (X,Y,Z).
CANGLE	Cold blade angle of the FE model.
ΔXLE,ΔYLE	FE displacements at the leading edge (ΔX,ΔY).
ΔXTE,ΔYTE	FE displacements at the trailing edge (ΔX,ΔY).
DFXLE,DFYLE	FE deflected shape at the leading edge (ΔX+X,ΔY+Y).
DFXTE,DFYTE	FE deflected shape at the trailing edge (ΔX+X,ΔY+Y).
DANGLE	Deflected blade angle of the FE model.

Table 3.3-5: Outline of prop3d.out File

Title
Atmospheric Conditions pressure, speed of sound, density
Operating Conditions rotor speed, Mach number, advance ratio, tip radius
Reiterates Key Input Variables
Reiterates Blade Geometry and Mesh Information - Front propeller - Aft propeller (if counter-rotating) - Leading and trailing edge coordinates - Chord length - Angle (α) with respect to the blade setting angle ($\beta_{3/4}$)
Information on grid shape for front (and aft if exists) - Minimum and maximum Jacobians - Indicates if grid is unsatisfactory
Maximum Density (ρ) ρ_u - X momentum ρ_v - Y momentum ρ_w - Z momentum ρ_e - total energy
Maximum residuals at each iteration and its location (i,j,k) User can determine if an adequate number of iterations are used to achieve convergence.
Pressure Profiles for front and aft propellers from hub to tip - Pressure given for the upper and lower surfaces of the airfoils - Plots of the pressure vs normalized chord length
For the front and counter rotation blade (if exists) - Advance ratio - Power coefficient - Thrust coefficient - Efficiency
CPU time for one iteration Total memory used

Table 3.3-6: Binary File (*e.g.,* f31_bin.*nn*) Containing the Flow Data from PROP3D

Not all information is shown only pertinent to the SAB process

Record 1:	(((XA(i,j,k),YA(i,j,k),ZA(i,j,k),i=1,120),j=1,33),k=1,16)
Record 2:	IL,JL,KL
Record 3:	(((Q1(i,j,k),Q2(i,j,k),Q3(i,j,k),Q4(i,j,k),Q5(i,j,k),i=1,120),j=1,33),k=1,16)

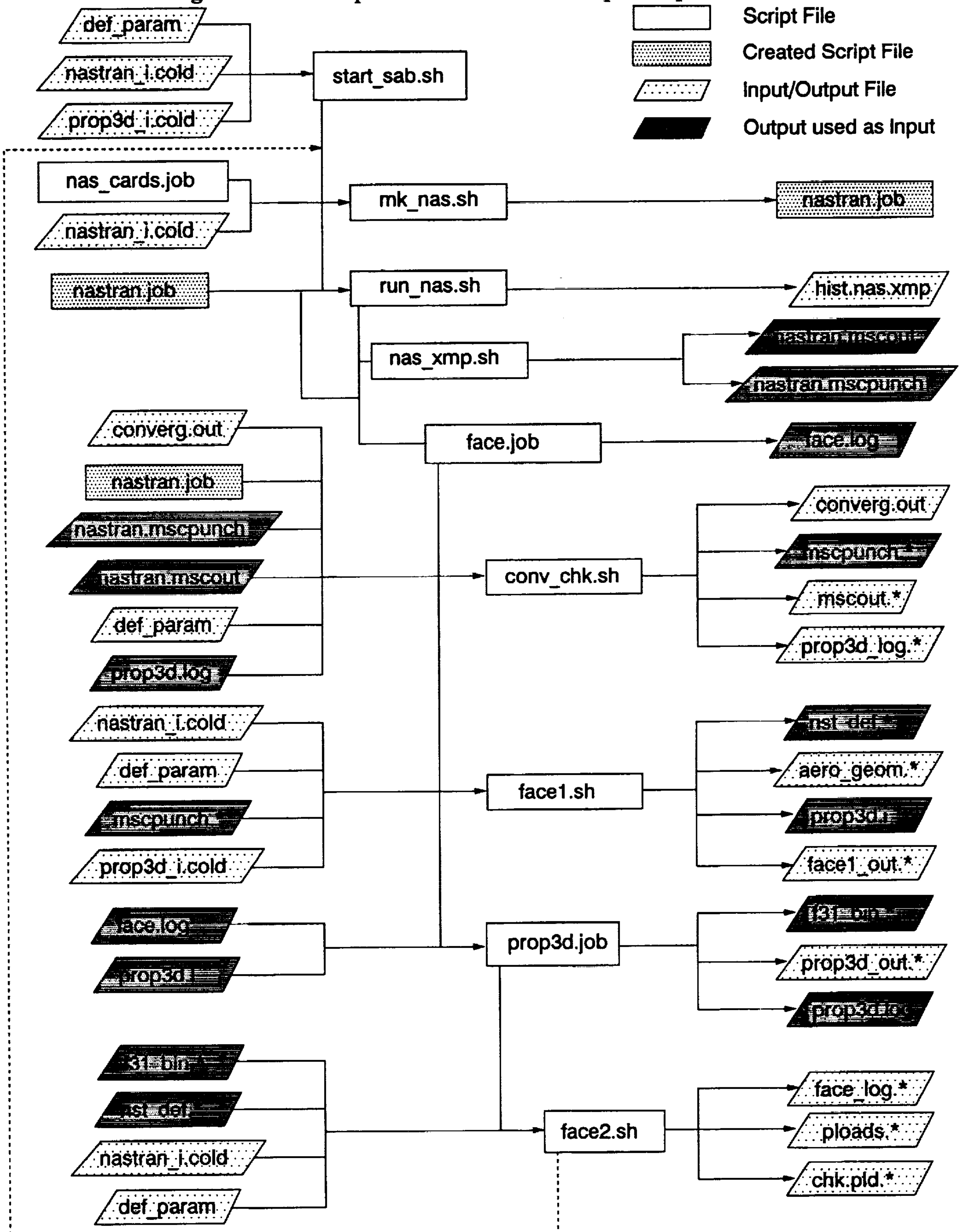
Parameter	*Description*
XA,YA,ZA	Coordinates of the aerodynamic grid shape.
IL	Total number of grids in the axial direction.
JL	Total number of grids in radial direction, hub to tip and beyond.
KL	Total number of grids in the azimuthal direction, blade to blade.
	Flow Variables
Q1	Density, ρ
Q2	Quantity corresponding to the XA direction equivalent to density times the cartesian velocity (ρ*u).
Q3	Quantity corresponding to the YA direction equivalent to density times the cartesian velocity (ρ*v).
Q4	Quantity corresponding to the ZA direction equivalent to density times the cartesian velocity (ρ*w).
Q5	Energy per unit volume.
120,33,16	Maximum size of the aerodynamic grid corresponding to the axial, radial, and azimuthal directions, respectively.

Table 3.3-7: Contents and Format of the ploads.*nn* Output File

Record 1 to nel:	PLOAD4,SID,EID,P1,P2,P3,P4	(A8,2I8,4F8.5)

Parameter	*Description*
PLOAD4	MSC/NASTRAN mnemonic to indicate pressure loads on the face of the structural element are being submitted.
SID	Load set identification number.
EID	Element identification number corresponding to the FE model.
P1,P2,P3,P4	Load per unit surface area (pressure) at the corners of the face of the element.
nel	Total number of elements in the FE model.

Figure 3.3-1: Script Files and Associated Input/Output Files



4. SAB Setup and Execution

4.1 Introduction

This section describes the necessary steps to execute the SAB system at NASA LeRC on the Cray Y-MP and X-MP computers. The user is not limited to these machines or running at NASA LeRC but changes may be necessary and recompiling of source codes will be essential for execution.

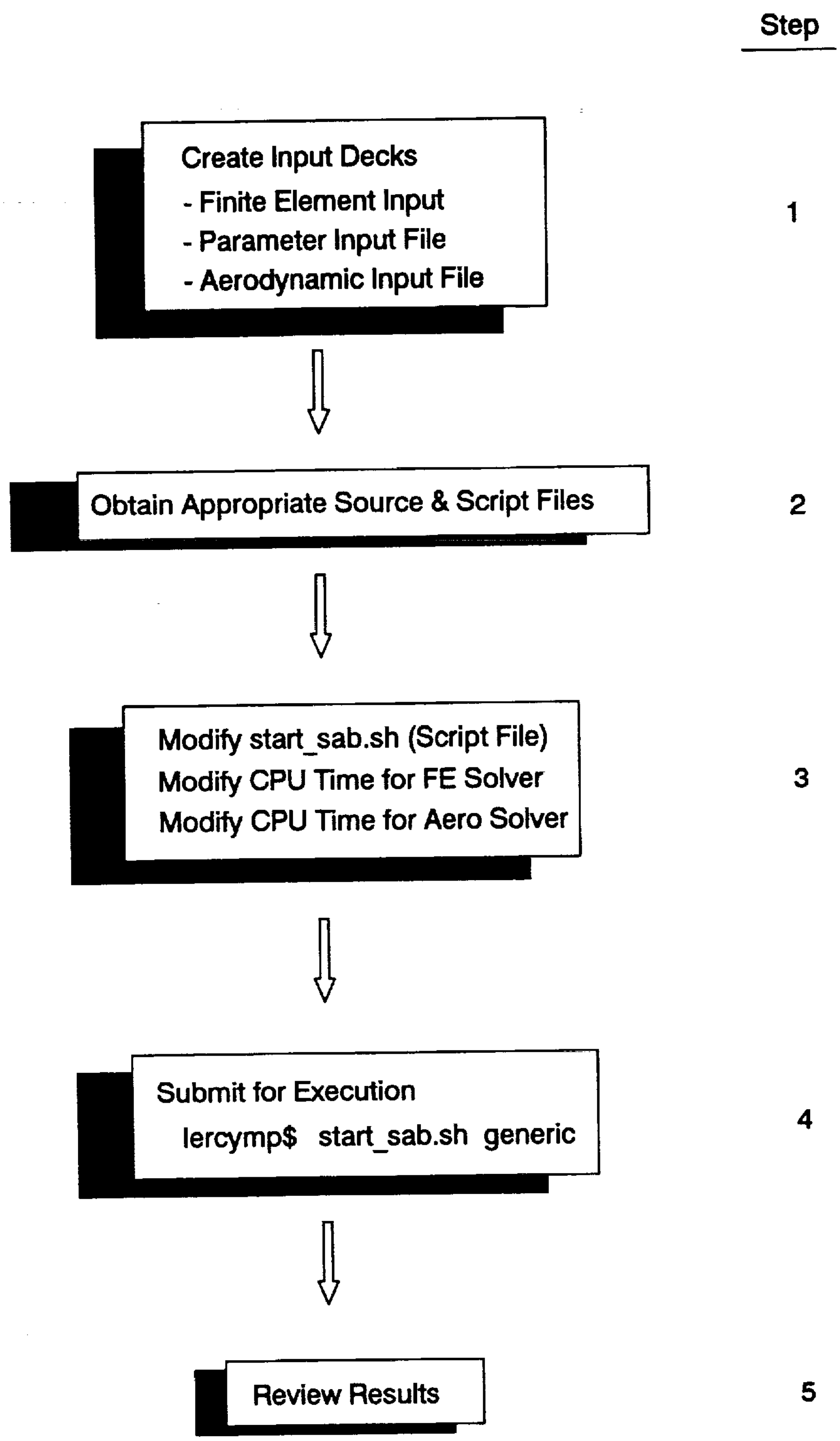
The SAB system is set-up to run in batch mode. This allows the user to have greater flexibility to change the methodology or add other capabilities, if desired. In order to execute the SAB process, the steps in Figure 4.1-1 should be followed. Before execution can begin, the user must generate the three required input files. In the next section, a detailed description of steps necessary to execute SAB on the Cray Y-MP and X-MP computers is presented. At the completion of execution, the user is responsible for reviewing the results for accuracy.

In order to communicate between the X-MP and Y-MP computers, the user must have a **.rhosts** file in their home directory. This file must exist on both systems. On the X-MP, the **.rhosts** allows information to be received from the Y-MP. In comparison, the **.rhosts** file on the Y-MP allows information to be received from the X-MP. An example of the **.rhosts** file is presented in Table 4.1-1.

Table 4.1-1: Typical .rhosts file Required to Communicate Between the Cray X-MP and Y-MP

.rhosts on the Y-MP lercxmp.lerc.nasa.gov *userid*
.rhosts on the X-MP lercymp.lerc.nasa.gov *userid*
userid = user identification on the computer system

Figure 4.1-1: Necessary Steps to Execute the SAB Process



4.2 Batch Mode Process

In order to run the SAB system, the files shown in Table 2.2-1 (pg. 9) are required. In addition, the three input files must be created. The user should store the files on their Cray Y-MP account. For ease of use and accountability of the numerous files, the organization of directories displayed in Figure 4.2-1 is suggested for executing the SAB system.

In order to use this organization, the user must create the following directories by using the **mkdir** command and store the proper files in them:

1.) **master_directory** - script file start_sab.sh exists in this directory. Execution begins in this directory and shifts to the **generic** directory.
2.) **sources** - contains all the required source files saved as compiled object files (*.o).
3.) **scripts** - contains all scripts required to perform the iterative process of SAB (*.sh and *.job).
4.) **inputs** - contains all the input files required for the SAB system.

Directory created at the time of execution:

5.) **generic** - the job is run in this directory, can be any directory name, created at run time by the script file **start_sab.sh**. All files required (inputs files, scripts, sources) and generated (output files) by SAB are stored within this directory.

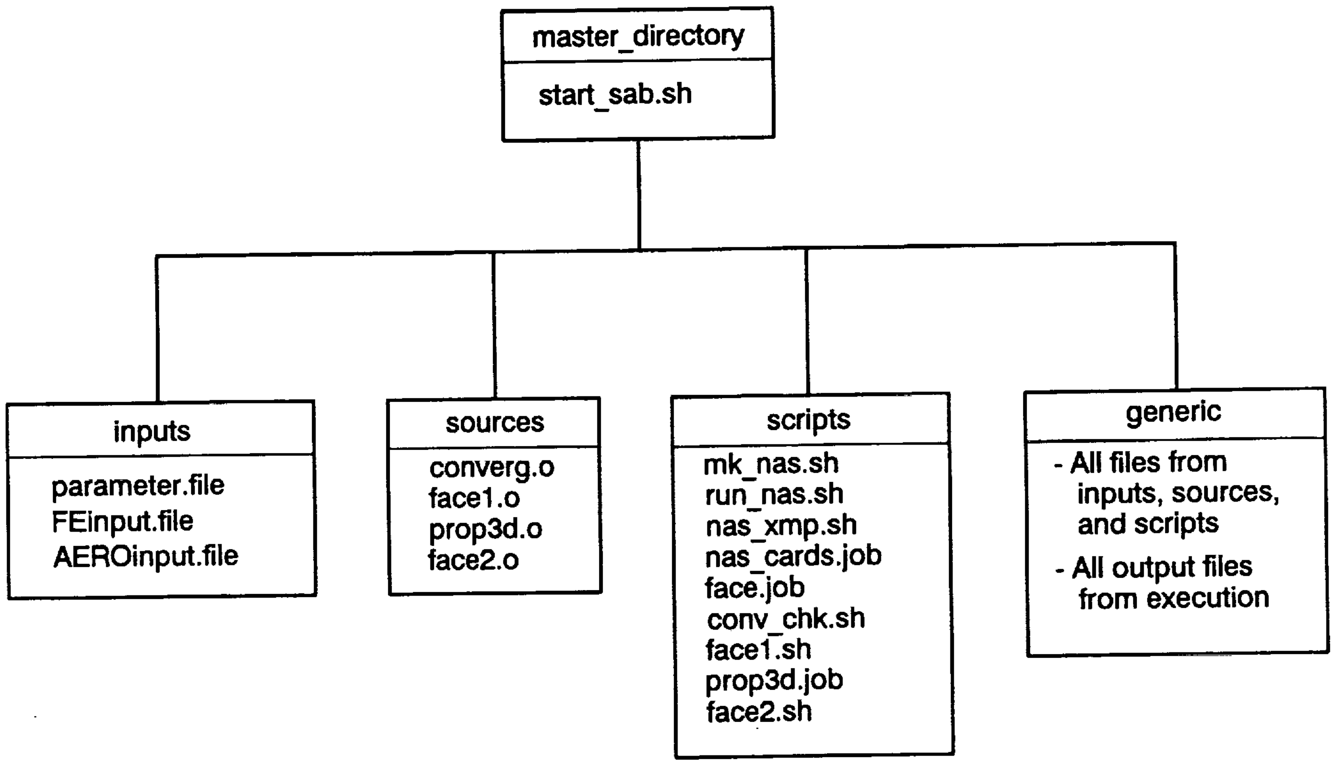
The directory names can be any valid UNIX directory name. The user is not restricted to the example directory names given above. After creating the directories the appropriate files should be stored in the proper directories.

After creating the input decks, the user must edit the **start_sab.sh** file to include the proper input decks and directories. For this task the user is referred to Appendix A, pg. 50, where the proper instructions are given. The user submits the **start_sab.sh** file and the generic directory name (**generic**) where execution is to take place.

lercymp$ **start_sab.sh generic**

The SAB system is now in progress. All outputs for this job will be found in the **generic** directory of the user's account. It is the user's responsibility to determine when the process has converged by checking the output file **converg.out**.

Figure 4.2-1: Batch Mode Directory Organization for the SAB System



5. SAB Analysis Hints and Tips

5.1 CPU Time Requirements

In order for efficient execution and reasonable turn-around time on analyses, the user should choose the minimum CPU time and memory requirements for the structural and aerodynamic analysis. These are the largest and most time consuming parts of the SAB procedure. If the job class is not properly entered, delays in turn-around time (*i.e.*, by remaining in the queue longer than necessary) can occur or the process could end prematurely due to insufficient CPU time. It is strongly recommended to the user that individual executions of MSC/NASTRAN and PROP3D be completed before executing the SAB process to determine if CPU time and memory are set properly. Also, the user can check the input decks for errors or problems that may cause the SAB process to end prematurely. At NASA LeRC the user should refer to the Help Desk for the guidelines on CPU and memory limits for the Cray Y-MP and Cray X-MP Network Queuing Job Batch Facility Systems.

The simplest way to obtain the proper CPU time for the MSC/NASTRAN deck is to make an initial estimate based on solution type, number of elements, complexity of elements, and other options available; this is left up to the user. Set the CPU time in the UNICOS job deck according to the queue limits of the Cray X-MP. Set the TIME command in the MSC/NASTRAN executive control deck approximately one minute less than the CPU time stipulated in the UNICOS deck. If the job does not have enough CPU time, the MSC/NASTRAN code will stop itself before the system terminates the job. At this point the user can increase the CPU time to the next job class and use the restart option available in MSC/NASTRAN. If MSC/NASTRAN executes successfully, the user should compare the actual CPU time provided at the end of the standard output file with the estimate and adjust the UNICOS CPU limit accordingly. The user can easily modify the **msc_cards.job** file in Appendix A, pg. 57 to execute the MSC/NASTRAN job independent of the SAB process. Also, the user can use the scripts run_nas.sh and nas_xmp.sh to submit the job.

For the PROP3D code, the user can put a low number of iteration steps (5) in Record 6 of the PROP3D input. The user then estimates the CPU time for the UNICOS batch job or in most instances can run the job interactively. CPU and turn-around time should be fairly quick with very few iteration steps. The prop3d.job file can be modified to execute the PROP3D code separately from the SAB process. From the PROP3D output, the user can obtain the CPU seconds per iteration. A reasonable estimate can be made from this data based on the user actual number of iterations required. Again, the user can set the CPU UNICOS time accordingly to the queue limits of the Cray Y-MP.

5.2 Interpreting SAB Results

A large number of output files are generated by SAB, refer to Figure 3.1-1 and Table 3.3-1. This section will guide the user in detecting possible errors and reviewing results. After submitting the SAB process for execution, the user can check the **hist.sab.xmp** file to determine if the MSC/NASTRAN job was received and submitted for execution on the Cray X-MP. The **mscout.*nn*** file should be reviewed for errors, warnings, and/or successful execution after returning to the Cray Y-MP from the Cray X-MP. If a problem exists and the **mscpunch.*nn*** file was not generated the system will cease operation. The user is warned that problems and inaccuracies can still exist even though the **mscpunch.*nn*** file is generated. A good practice is to thoroughly check the finite element model before using it in the SAB process.

Execution begins on the Cray Y-MP with the return of the MSC/NASTRAN output files. To monitor the progress on the Cray Y-MP, the user should review files with the **log** extension. These files should help the user to track down possible problems or premature termination of the process. The **log** files contain statements to determine the progress of the SAB process. Also, it is helpful to review is the **prop3d_out.*nn*** file, provided for each iteration of the SAB process, to determine the precision of the aerodynamic analysis. The **converg.out** file provides information on the status of the iteration and if convergence was met. A calculated converged shape is the deflected shape under the operating conditions. A calculated diverged shape indicates that the blade is statically unstable.

6. Example Analysis

6.1 Introduction

An example case is presented to guide the user in the input preparation for the SAB process. A forward swept composite propfan blade, designated F39, is used to demonstrate the input setup for the three required input files, refer to Figure 4.1-1 step 1. The input deck for the structural analysis is not provided due to its size, but a description of the model is supplied. Also, a description of the input decks is presented for the parameter file and aero solver input. A complete listing of each input deck is given. Results from the SAB analysis are not presented herein but typical results can be found in reference [8].

6.2 Geometry and Finite Element Model

The F39 propfan blade used in this guide as an example is 7.7 in. long in the radial (spanwise) direction, has a tip chord of 0.95 in., and a base chord of 2.0 in. Maximum chord length is about 2.5 in. and is located around the 2.0 in. spanwise height. At the 75% radius, the blade angle ($\beta_{3/4}$) setting is 35°. The MSC/NASTRAN finite element model of the F39 forward swept blade is shown in Figure 6.2-1. This model consists of 504 grid points (3 nodes per element) and 910 triangular plate elements (CTRIA3 element type in MSC/NASTRAN). The CTRIA3 element is an isoparametric membrane-bending element. Along the base, the middle 6 grid points are fixed in all directions and the system has 2988 degrees of freedom. These 6 grid points correspond to the titanium spar. To perform the nonlinear deflection analysis on the rotating blade, solution 64 was chosen using 14 subcases.

The model generator COBSTRAN [9] (a pre-processor to MSC/NASTRAN) was used to generate the MSC/NASTRAN model with anisotropic homogenous material properties. Table 6.2-1 presents the material properties utilized by the COBSTRAN input deck. The F39 consists of a titanium spar imbedded in graphite-epoxy plies. Material properties are defined for each element using the MAT2 material property card. The thickness varies throughout the blade with a range that varies from 0.009 at the leading and trailing edges to 0.308 inches at the center chord. Since this is a composite laminate, individual plies have varying thicknesses. Finally, a rotational speed of 5600.0 rpm is applied during the structural analysis.

Figure 6.2-1: Finite Element Model of a Forward Swept Propfan Blade

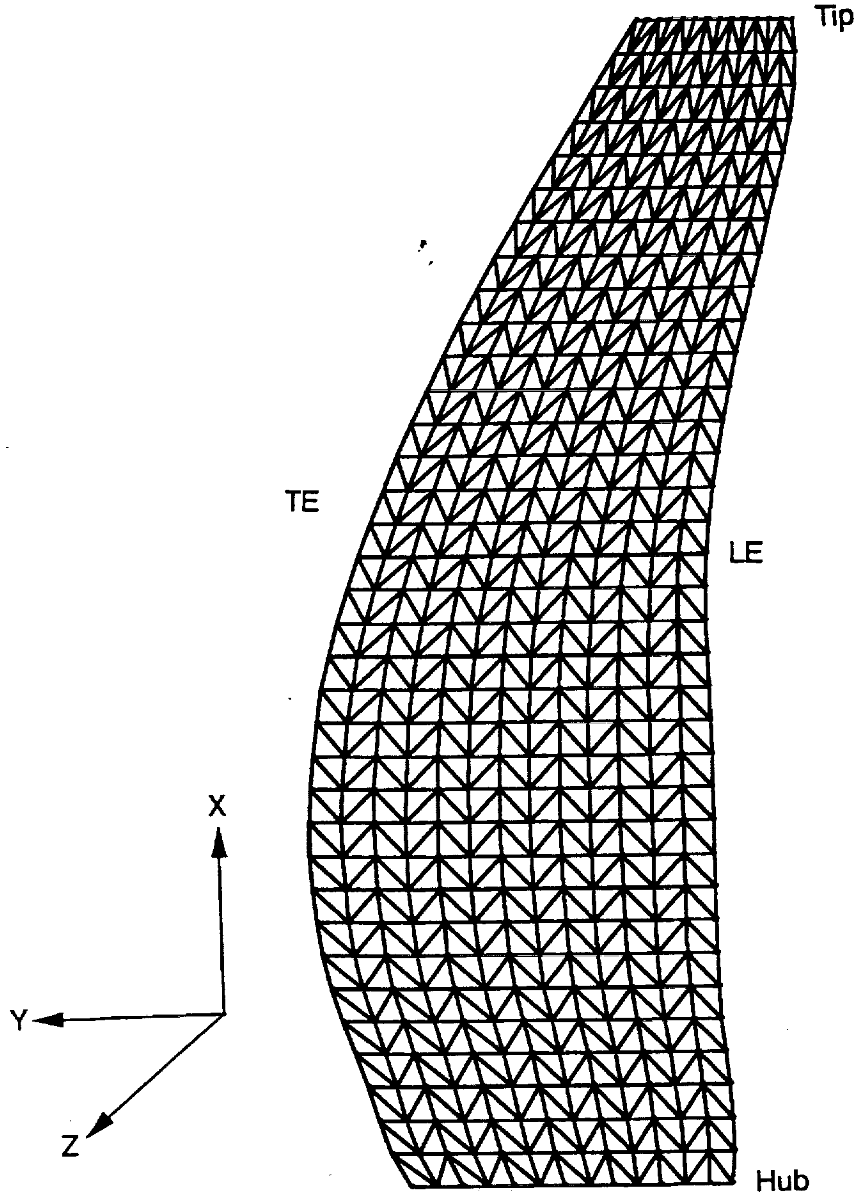


Table 6.2-1: Material Properties Utilized by COBSTRAN to Generate the Global Material Properties of the F39 Propfan Blade

Lamina Material Properties Used in Blade Design	
Longitudinal Modulus (E_{11})	21.0 x 10^6 psi
Transverse Modulus (E_{22})	1.30 x 10^6 psi
Shear Modulus (G_{12})	0.76 x 10^6 psi
Poisson's Ratio (υ_{12})	0.35
Weight Density	0.0571 lbm/in^3

Titanium Properties Used as Spar	
Elastic Modulus	16.0 x 10^6 psi
Shear Modulus	6.20 x 10^6 psi
Poisson's Ratio	0.31
Weight Density	0.160 lbm/in^3

6.3 Parameter File Input

The parameter file defines five items that are used throughout the SAB process, shown in Table 6.3-1 is the parameter input file used for this example. A complete description of the input deck is located in Appendix D. The following is a list of items that are specified in the example file.

(1) In the first line, the spanwise and chordwise directions of the finite element model are defined. The spanwise direction is in the positive X direction (`PX`), refer to Figure 6.2-1, and the chordwise direction is in the positive Y (`PY`) direction.

(2) The first value of line 2 is the reference radius where convergence is checked. In this case, the radial location was chosen at 95% (`0.95`). Referring to Figure 6.2-1(a), this is located at the 3rd chord down from the tip.

(3) The second value in line 2 is the tolerance for convergence. The current iteration's blade angle at 95% radial location is checked against the previous iteration's blade angle. In order to achieve convergence, the difference in blade angles must be within 0.01° of each other.

(4) On the third line, the full scale blade row diameter is defined in inches. For this case, the nominal blade row diameter is `26.004` inches.

(5) In the last line a dimensional factor is defined by the following equation:

$$P_{dim} = \rho * a^2$$

where ρ is the free stream density and a is the speed of sound. The value of `20.401` is used in this example.

Table 6.3-1: Parameter Input File for a Propfan Blade

```
PXPY                    coordinate definition
0.95,0.01               radius ratio, tolerance
26.004                  blade diameter
20.401                  dimensional factor
```

6.4 Aerodynamic Input File

The F39 propfan blade row has 12 blades. The steady flow in all blade passages can be assumed identical due to axisymmetric inflow and identical blades, therefore, only one blade passage is solved with the conditions of periodicity enforced. Critical factors influencing the aerodynamic results are the advance ratio (1.014), the free stream mach number (0.20), and the blade setting angle at 75% radius (35.0°). Displayed in Table 6.4-1 is the entire PROP3D input deck for the F39 propfan blade. Above each single numerical line of input, the variable names defined in Appendix D are given as the comment line. This is also done for the comment lines above the numerical sets of data. Complete details on the PROP3D input deck can be found in Appendix D.

A body fitted H-O grid is used for the aerodynamic calculations, refer to Figure 2.1-3 for a typical aerodynamic grid. The grid size is defined by the variables IMAX (80), JMAX (22), and KMAX (16). IMAX and JMAX define the H-grid in streamwise plane, whereas KMAX gives the number of grids between blades in the azimuthal plane. A total of 10 iterations (FSTP) are given for the calculation.

The first of set of data (*i.e.*, multiple lines of numerical data) defines the hub contour (XR) and the hub radii (RR). A total of 63 values are given which are stipulated by the variable FNRS. In the second set of data, the individual airfoil stations are given. The number of airfoil stations is defined by the variable FNC (10). For each individual station, the appropriate records are repeated with the proper data. Finally, the last line of input deck give the sonic velocity of the fluid (a0 = 13040.0 in/sec) and the static pressure (p0 = 14.69 psi).

Table 6.4-1: PROP3D Input Deck for Propfan Blade

```
F39 FORWARD SWEPT BLADE IEX=1 ISWCH=1 IWHIT=0
       ADV        GMU       ALFA        PSI         WW         DT     REYREF
    1.0140        0.0        0.0        0.0        7.0      .0030       -1.0
      IMAX       JMAX       KMAX       JTIP       ITEL        ILE        IGR
        80         22         16         15         23         23          0
      FSTP      FMINF     BETA34        DIA         DY     VIBFRE
      10.0      0.200     35.000        1.0       0.05        0.0
     IFLTR     NUMCYC      NSTDY      JMODE
        -1          0          0          2
RESTART , QUASISTEADY , INFLOW , AEROELASTIC AND COUNTER ROTATION
FALSE TRUEFALSEFALSEFALSE
      ICCW      ITURB      LTHIN       ISWF
         1          0          0          0
     FNRS
 63.010000
       XR         RR
 -2.000000  0.076145
 -1.933582  0.076345
 -1.470110  0.076678
 -1.199300  0.077645
 -1.021340  0.083649
 -0.897309  0.106003
 -0.896821  0.113438
 -0.895328  0.117709
 -0.892009  0.123474
 -0.883714  0.132558
 -0.872650  0.140937
 -0.853291  0.151785
 -0.836694  0.159174
 -0.814565  0.167446
 -0.792444  0.174518
 -0.764782  0.182170
 -0.737129  0.188832
 -0.703935  0.195803
 -0.670749  0.201862
 -0.643088  0.206318
 -0.609902  0.211038
 -0.554587  0.217522
 -0.488208  0.223248
 -0.432893  0.226381
 -0.388635  0.227836
 -0.344385  0.228331
 -0.269881  0.227354
 -0.204577  0.224487
 -0.177961  0.222762
 -0.150106  0.220601
 -0.123799  0.218138
 -0.091998  0.214441
 -0.077374  0.212342
 -0.061899  0.209595
 -0.042401  0.205585
 -0.028071  0.203196
 -0.009006  0.201586
  0.010376  0.201230
  0.035213  0.201591
  0.055307  0.201742
  0.077405  0.201150
  0.103968  0.199660
  0.134941  0.197422
  0.189412  0.192301
  0.237129  0.187710
  0.251930  0.186659
```

```
 0.271367  0.186378
 0.291097  0.187452
 0.315888  0.189380
 0.335626  0.190845
 0.356169  0.191889
 0.371396  0.192186
 0.390740  0.191811
 0.413557  0.190811
 0.440932  0.189067
 0.470969  0.186228
 0.526818  0.178315
 0.597290  0.164107
 0.680607  0.145362
 0.874329  0.110125
 1.183826  0.089781
 1.485585  0.090946
 2.000000  0.090946
 INASTRAN
        0
      FNC     FROTAT         FB      FTPRP      FTWST       FCOB        FGR
10.010000   1.010000  12.010000   1.010000   1.010000  18.010000    1.01000
   IDTYPE
 3.010000
    YW(J)         AL       ALED        FAD        CHD       FSEC      THICK
 0.191708  16.488831  -0.052793  -0.002382   0.100996   1.010000   1.000000
     ZSYM        FNU        FNL
 0.000000  65.010000  65.010000
                   X     ZUPPER     ZLOWER
            0.000000   0.000000   0.000000   0.000250   0.000296  -0.000250
            0.000500   0.000564  -0.000477   0.001000   0.001017  -0.000859
            0.002000   0.001437  -0.001211   0.003000   0.001748  -0.001436
            0.005000   0.002175  -0.001692   0.007000   0.002465  -0.001813
            0.010000   0.002789  -0.001888   0.015000   0.003250  -0.001931
            0.020000   0.003679  -0.001962   0.025000   0.004080  -0.001994
            0.030000   0.004457  -0.002025   0.035000   0.004817  -0.002048
            0.040000   0.005160  -0.002068   0.045000   0.005491  -0.002090
            0.050000   0.005807  -0.002115   0.060000   0.006383  -0.002177
            0.070000   0.006907  -0.002250   0.080000   0.007391  -0.002334
            0.090000   0.007832  -0.002433   0.100000   0.008233  -0.002542
            0.120000   0.008907  -0.002799   0.140000   0.009399  -0.003134
            0.160000   0.009760  -0.003508   0.180000   0.010026  -0.003917
            0.200000   0.010120  -0.004405   0.250000   0.009946  -0.005752
            0.300000   0.009167  -0.007340   0.350000   0.007969  -0.009011
            0.400000   0.006319  -0.010838   0.450000   0.004296  -0.012797
            0.500000   0.001952  -0.014845   0.550000  -0.000255  -0.016512
            0.600000  -0.002362  -0.017825   0.650000  -0.004132  -0.018576
            0.700000  -0.005417  -0.018656   0.750000  -0.005999  -0.017917
            0.800000  -0.005814  -0.016374   0.820000  -0.005500  -0.015524
            0.840000  -0.005069  -0.014598   0.860000  -0.004482  -0.013556
            0.880000  -0.003754  -0.012406   0.900000  -0.002896  -0.011150
            0.910000  -0.002420  -0.010480   0.920000  -0.001913  -0.009782
            0.930000  -0.001379  -0.009060   0.940000  -0.000815  -0.008310
            0.950000  -0.000215  -0.007529   0.955000   0.000099  -0.007124
            0.960000   0.000423  -0.006711   0.965000   0.000753  -0.006288
            0.970000   0.001089  -0.005860   0.975000   0.001428  -0.005434
            0.980000   0.001772  -0.005002   0.985000   0.002127  -0.004565
            0.990000   0.002494  -0.004127   0.993000   0.002721  -0.003853
            0.995000   0.002866  -0.003676   0.997000   0.002861  -0.003407
            0.998000   0.002719  -0.003150   0.999000   0.001988  -0.002497
            0.999500   0.001458  -0.001873   0.999750   0.000822  -0.001512
            1.000000   0.000000   0.000000
    YW(J)         AL       ALED        FAD        CHD       FSEC      THICK
 0.212471  15.786712  -0.053642  -0.002265   0.109999   1.010000   1.000000
     ZSYM        FNU        FNL
```

```
0.000000 65.010000 65.010000
                X      ZUPPER     ZLOWER
         0.000000   0.000000   0.000000   0.000250   0.000296  -0.000250
         0.000500   0.000564  -0.000477   0.001000   0.001017  -0.000859
         0.002000   0.001438  -0.001210   0.003000   0.001749  -0.001435
         0.005000   0.002178  -0.001690   0.007000   0.002469  -0.001810
         0.010000   0.002794  -0.001883   0.015000   0.003258  -0.001924
         0.020000   0.003689  -0.001952   0.025000   0.004093  -0.001982
         0.030000   0.004472  -0.002010   0.035000   0.004835  -0.002031
         0.040000   0.005180  -0.002048   0.045000   0.005514  -0.002068
         0.050000   0.005832  -0.002090   0.060000   0.006413  -0.002148
         0.070000   0.006942  -0.002216   0.080000   0.007430  -0.002295
         0.090000   0.007876  -0.002390   0.100000   0.008281  -0.002495
         0.120000   0.008962  -0.002744   0.140000   0.009461  -0.003073
         0.160000   0.009827  -0.003442   0.180000   0.010096  -0.003847
         0.200000   0.010192  -0.004333   0.250000   0.010018  -0.005681
         0.300000   0.009234  -0.007274   0.350000   0.008028  -0.008953
         0.400000   0.006368  -0.010789   0.450000   0.004337  -0.012756
         0.500000   0.001986  -0.014811   0.550000  -0.000228  -0.016485
         0.600000  -0.002341  -0.017804   0.650000  -0.004114  -0.018557
         0.700000  -0.005399  -0.018637   0.750000  -0.005982  -0.017899
         0.800000  -0.005800  -0.016359   0.820000  -0.005487  -0.015511
         0.840000  -0.005058  -0.014587   0.860000  -0.004473  -0.013547
         0.880000  -0.003747  -0.012399   0.900000  -0.002890  -0.011144
         0.910000  -0.002415  -0.010475   0.920000  -0.001909  -0.009778
         0.930000  -0.001375  -0.009056   0.940000  -0.000811  -0.008306
         0.950000  -0.000212  -0.007526   0.955000   0.000102  -0.007121
         0.960000   0.000426  -0.006708   0.965000   0.000756  -0.006284
         0.970000   0.001093  -0.005856   0.975000   0.001432  -0.005430
         0.980000   0.001776  -0.004998   0.985000   0.002130  -0.004562
         0.990000   0.002496  -0.004125   0.993000   0.002723  -0.003851
         0.995000   0.002867  -0.003675   0.997000   0.002862  -0.003406
         0.998000   0.002720  -0.003149   0.999000   0.001989  -0.002496
         0.999500   0.001459  -0.001873   0.999750   0.000823  -0.001511
         1.000000   0.000000   0.000000
   YW(J)       AL       ALED        FAD        CHD       FSEC      THICK
0.251260 11.012386 -0.052937  -0.001833   0.130747   1.010000   1.000000
    ZSYM      FNU        FNL
0.000000 65.010000 65.010000
                X      ZUPPER     ZLOWER
         0.000000   0.000000   0.000000   0.000250   0.000296  -0.000250
         0.000500   0.000564  -0.000477   0.001000   0.001018  -0.000859
         0.002000   0.001439  -0.001209   0.003000   0.001752  -0.001433
         0.005000   0.002181  -0.001686   0.007000   0.002474  -0.001805
         0.010000   0.002802  -0.001876   0.015000   0.003269  -0.001913
         0.020000   0.003704  -0.001938   0.025000   0.004111  -0.001964
         0.030000   0.004493  -0.001990   0.035000   0.004859  -0.002007
         0.040000   0.005208  -0.002021   0.045000   0.005544  -0.002038
         0.050000   0.005866  -0.002058   0.060000   0.006452  -0.002109
         0.070000   0.006986  -0.002172   0.080000   0.007480  -0.002247
         0.090000   0.007930  -0.002337   0.100000   0.008340  -0.002437
         0.120000   0.009031  -0.002677   0.140000   0.009540  -0.002996
         0.160000   0.009916  -0.003354   0.180000   0.010196  -0.003749
         0.200000   0.010304  -0.004224   0.250000   0.010162  -0.005540
         0.300000   0.009409  -0.007102   0.350000   0.008232  -0.008751
         0.400000   0.006599  -0.010561   0.450000   0.004590  -0.012506
         0.500000   0.002257  -0.014542   0.550000   0.000059  -0.016200
         0.600000  -0.002042  -0.017507   0.650000  -0.003813  -0.018258
         0.700000  -0.005107  -0.018347   0.750000  -0.005705  -0.017624
         0.800000  -0.005548  -0.016109   0.820000  -0.005250  -0.015274
         0.840000  -0.004837  -0.014367   0.860000  -0.004271  -0.013346
         0.880000  -0.003567  -0.012219   0.900000  -0.002735  -0.010989
         0.910000  -0.002273  -0.010333   0.920000  -0.001781  -0.009650
         0.930000  -0.001262  -0.008943   0.940000  -0.000714  -0.008209
```

```
          0.950000 -0.000131 -0.007445  0.955000  0.000175 -0.007048
          0.960000  0.000490 -0.006644  0.965000  0.000811 -0.006230
          0.970000  0.001138 -0.005811  0.975000  0.001468 -0.005394
          0.980000  0.001803 -0.004971  0.985000  0.002148 -0.004544
          0.990000  0.002507 -0.004114  0.993000  0.002730 -0.003844
          0.995000  0.002872 -0.003670  0.997000  0.002865 -0.003403
          0.998000  0.002723 -0.003146  0.999000  0.001991 -0.002493
          0.999500  0.001462 -0.001871  0.999750  0.000825 -0.001506
          1.000000  0.000000  0.000000
    YW(J)       AL      ALED       FAD       CHD      FSEC     THICK
 0.309445  5.279466 -0.056270 -0.001656  0.149360  1.010000  1.000000
     ZSYM       FNU       FNL
 0.000000 65.010000 65.010000
                 X    ZUPPER    ZLOWER
          0.000000  0.000000  0.000000  0.000250  0.000296 -0.000250
          0.000500  0.000565 -0.000476  0.001000  0.001019 -0.000857
          0.002000  0.001444 -0.001204  0.003000  0.001760 -0.001425
          0.005000  0.002195 -0.001672  0.007000  0.002494 -0.001784
          0.010000  0.002830 -0.001847  0.015000  0.003312 -0.001869
          0.020000  0.003761 -0.001880  0.025000  0.004183 -0.001892
          0.030000  0.004580 -0.001903  0.035000  0.004960 -0.001906
          0.040000  0.005322 -0.001906  0.045000  0.005673 -0.001909
          0.050000  0.006008 -0.001915  0.060000  0.006622 -0.001939
          0.070000  0.007184 -0.001974  0.080000  0.007705 -0.002021
          0.090000  0.008182 -0.002084  0.100000  0.008618 -0.002158
          0.120000  0.009361 -0.002346  0.140000  0.009920 -0.002615
          0.160000  0.010344 -0.002926  0.180000  0.010672 -0.003273
          0.200000  0.010825 -0.003703  0.250000  0.010787 -0.004914
          0.300000  0.010133 -0.006378  0.350000  0.009044 -0.007940
          0.400000  0.007478 -0.009684  0.450000  0.005513 -0.011584
          0.500000  0.003201 -0.013600  0.550000  0.001001 -0.015260
          0.600000 -0.001126 -0.016592  0.650000 -0.002946 -0.017392
          0.700000 -0.004312 -0.017551  0.750000 -0.005001 -0.016919
          0.800000 -0.004953 -0.015513  0.820000 -0.004705 -0.014728
          0.840000 -0.004344 -0.013873  0.860000 -0.003834 -0.012907
          0.880000 -0.003188 -0.011839  0.900000 -0.002416 -0.010670
          0.910000 -0.001986 -0.010045  0.920000 -0.001525 -0.009394
          0.930000 -0.001039 -0.008719  0.940000 -0.000524 -0.008018
          0.950000  0.000026 -0.007287  0.955000  0.000314 -0.006908
          0.960000  0.000613 -0.006521  0.965000  0.000917 -0.006124
          0.970000  0.001226 -0.005722  0.975000  0.001539 -0.005323
          0.980000  0.001856 -0.004918  0.985000  0.002184 -0.004508
          0.990000  0.002526 -0.004094  0.993000  0.002743 -0.003830
          0.995000  0.002882 -0.003659  0.997000  0.002871 -0.003396
          0.998000  0.002729 -0.003140  0.999000  0.001996 -0.002486
          0.999500  0.001469 -0.001866  0.999750  0.000831 -0.001497
          1.000000  0.000000  0.000000
    YW(J)       AL      ALED       FAD       CHD      FSEC     THICK
 0.387025 -1.546978 -0.068228 -0.002068  0.137341  1.010000  1.000000
     ZSYM       FNU       FNL
 0.000000 65.010000 65.010000
                 X    ZUPPER    ZLOWER
          0.000000  0.000000  0.000000  0.000250  0.000295 -0.000251
          0.000500  0.000562 -0.000478  0.001000  0.001018 -0.000857
          0.002000  0.001451 -0.001196  0.003000  0.001771 -0.001412
          0.005000  0.002217 -0.001648  0.007000  0.002526 -0.001750
          0.010000  0.002878 -0.001797  0.015000  0.003384 -0.001795
          0.020000  0.003857 -0.001781  0.025000  0.004303 -0.001769
          0.030000  0.004723 -0.001756  0.035000  0.005127 -0.001736
          0.040000  0.005513 -0.001713  0.045000  0.005886 -0.001692
          0.050000  0.006244 -0.001675  0.060000  0.006903 -0.001654
          0.070000  0.007508 -0.001646  0.080000  0.008072 -0.001651
          0.090000  0.008591 -0.001672  0.100000  0.009068 -0.001705
          0.120000  0.009889 -0.001816  0.140000  0.010521 -0.002011
```

```
         0.160000  0.011015 -0.002253  0.180000  0.011407 -0.002536
         0.200000  0.011619 -0.002906  0.250000  0.011712 -0.003989
         0.300000  0.011160 -0.005351  0.350000  0.010146 -0.006840
         0.400000  0.008634 -0.008530  0.450000  0.006708 -0.010393
         0.500000  0.004421 -0.012384  0.550000  0.002225 -0.014037
         0.600000  0.000081 -0.015386  0.650000 -0.001780 -0.016224
         0.700000 -0.003209 -0.016445  0.750000 -0.003981 -0.015894
         0.800000 -0.004048 -0.014602  0.820000 -0.003855 -0.013872
         0.840000 -0.003557 -0.013079  0.860000 -0.003115 -0.012182
         0.880000 -0.002544 -0.011190  0.900000 -0.001856 -0.010104
         0.910000 -0.001470 -0.009523  0.920000 -0.001055 -0.008918
         0.930000 -0.000617 -0.008292  0.940000 -0.000152 -0.007641
         0.950000  0.000345 -0.006962  0.955000  0.000607 -0.006610
         0.960000  0.000878 -0.006250  0.965000  0.001154 -0.005881
         0.970000  0.001435 -0.005508  0.975000  0.001719 -0.005138
         0.980000  0.002006 -0.004762  0.985000  0.002303 -0.004384
         0.990000  0.002610 -0.004005  0.993000  0.002802 -0.003767
         0.995000  0.002924 -0.003613  0.997000  0.002898 -0.003365
         0.998000  0.002756 -0.003112  0.999000  0.002017 -0.002455
         0.999500  0.001499 -0.001847  0.999750  0.000855 -0.001459
         1.000000  0.000000  0.000000
    YW(J)       AL      ALED       FAD       CHD      FSEC     THICK
 0.425815 -3.570778 -0.087072 -0.003707  0.123466  1.010000  1.000000
     ZSYM       FNU       FNL
 0.000000 65.010000 65.010000
                X    ZUPPER    ZLOWER
         0.000000  0.000000  0.000000  0.000250  0.000293 -0.000252
         0.000500  0.000560 -0.000480  0.001000  0.001017 -0.000857
         0.002000  0.001457 -0.001189  0.003000  0.001783 -0.001399
         0.005000  0.002241 -0.001622  0.007000  0.002561 -0.001713
         0.010000  0.002928 -0.001744  0.015000  0.003460 -0.001716
         0.020000  0.003958 -0.001679  0.025000  0.004426 -0.001643
         0.030000  0.004869 -0.001609  0.035000  0.005294 -0.001567
         0.040000  0.005700 -0.001524  0.045000  0.006093 -0.001484
         0.050000  0.006469 -0.001449  0.060000  0.007163 -0.001394
         0.070000  0.007799 -0.001355  0.080000  0.008391 -0.001332
         0.090000  0.008935 -0.001329  0.100000  0.009434 -0.001340
         0.120000  0.010292 -0.001414  0.140000  0.010952 -0.001582
         0.160000  0.011467 -0.001804  0.180000  0.011873 -0.002074
         0.200000  0.012093 -0.002437  0.250000  0.012185 -0.003520
         0.300000  0.011608 -0.004908  0.350000  0.010552 -0.006439
         0.400000  0.008986 -0.008183  0.450000  0.006996 -0.010110
         0.500000  0.004637 -0.012171  0.550000  0.002377 -0.013886
         0.600000  0.000178 -0.015287  0.650000 -0.001729 -0.016169
         0.700000 -0.003191 -0.016422  0.750000 -0.003987 -0.015893
         0.800000 -0.004052 -0.014597  0.820000 -0.003847 -0.013856
         0.840000 -0.003532 -0.013046  0.860000 -0.003066 -0.012125
         0.880000 -0.002466 -0.011104  0.900000 -0.001742 -0.009983
         0.910000 -0.001336 -0.009383  0.920000 -0.000901 -0.008757
         0.930000 -0.000441 -0.008109  0.940000  0.000048 -0.007434
         0.950000  0.000570 -0.006731  0.955000  0.000845 -0.006365
         0.960000  0.001129 -0.005992  0.965000  0.001418 -0.005610
         0.970000  0.001714 -0.005223  0.975000  0.001984 -0.004867
         0.980000  0.002216 -0.004548  0.985000  0.002459 -0.004223
         0.990000  0.002714 -0.003897  0.993000  0.002873 -0.003691
         0.995000  0.002975 -0.003558  0.997000  0.002931 -0.003328
         0.998000  0.002790 -0.003078  0.999000  0.002043 -0.002417
         0.999500  0.001536 -0.001824  0.999750  0.000886 -0.001413
         1.000000  0.000000  0.000000
    YW(J)       AL      ALED       FAD       CHD      FSEC     THICK
 0.445210 -4.148627 -0.098231 -0.004836  0.114139  1.010000  1.000000
     ZSYM       FNU       FNL
 0.000000 65.010000 65.010000
                X    ZUPPER    ZLOWER
```

```
            0.000000  0.000000  0.000000  0.000250  0.000292 -0.000253
            0.000500  0.000558 -0.000481  0.001000  0.001016 -0.000858
            0.002000  0.001461 -0.001184  0.003000  0.001788 -0.001390
            0.005000  0.002254 -0.001607  0.007000  0.002580 -0.001691
            0.010000  0.002957 -0.001713  0.015000  0.003504 -0.001670
            0.020000  0.004016 -0.001618  0.025000  0.004499 -0.001568
            0.030000  0.004957 -0.001519  0.035000  0.005396 -0.001463
            0.040000  0.005816 -0.001406  0.045000  0.006223 -0.001352
            0.050000  0.006613 -0.001303  0.060000  0.007339 -0.001217
            0.070000  0.008002 -0.001151  0.080000  0.008620 -0.001103
            0.090000  0.009181 -0.001081  0.100000  0.009704 -0.001069
            0.120000  0.010604 -0.001101  0.140000  0.011303 -0.001232
            0.160000  0.011859 -0.001412  0.180000  0.012280 -0.001667
            0.200000  0.012525 -0.002006  0.250000  0.012673 -0.003033
            0.300000  0.012142 -0.004375  0.350000  0.011113 -0.005879
            0.400000  0.009557 -0.007613  0.450000  0.007562 -0.009546
            0.500000  0.005189 -0.011619  0.550000  0.002912 -0.013351
            0.600000  0.000687 -0.014776  0.650000 -0.001249 -0.015686
            0.700000 -0.002735 -0.015961  0.750000 -0.003557 -0.015457
            0.800000 -0.003649 -0.014188  0.820000 -0.003456 -0.013460
            0.840000 -0.003150 -0.012660  0.860000 -0.002697 -0.011752
            0.880000 -0.002115 -0.010749  0.900000 -0.001407 -0.009643
            0.910000 -0.001009 -0.009052  0.920000 -0.000583 -0.008434
            0.930000 -0.000123 -0.007787  0.940000  0.000357 -0.007121
            0.950000  0.000870 -0.006427  0.955000  0.001139 -0.006067
            0.960000  0.001418 -0.005699  0.965000  0.001704 -0.005321
            0.970000  0.001939 -0.004997  0.975000  0.002136 -0.004714
            0.980000  0.002337 -0.004424  0.985000  0.002550 -0.004130
            0.990000  0.002774 -0.003835  0.993000  0.002915 -0.003647
            0.995000  0.003005 -0.003526  0.997000  0.002950 -0.003307
            0.998000  0.002809 -0.003059  0.999000  0.002058 -0.002396
            0.999500  0.001558 -0.001810  0.999750  0.000905 -0.001388
            1.000000  0.000000  0.000000
   YW(J)          AL      ALED       FAD       CHD      FSEC     THICK
0.484000 -5.636276 -0.119891 -0.006392  0.089797  1.010000  1.000000
    ZSYM       FNU       FNL
0.000000 65.010000 65.010000
                   X    ZUPPER    ZLOWER
            0.000000  0.000000  0.000000  0.000250  0.000288 -0.000256
            0.000500  0.000550 -0.000487  0.001000  0.001004 -0.000866
            0.002000  0.001449 -0.001193  0.003000  0.001773 -0.001403
            0.005000  0.002231 -0.001625  0.007000  0.002552 -0.001715
            0.010000  0.002919 -0.001746  0.015000  0.003448 -0.001719
            0.020000  0.003945 -0.001682  0.025000  0.004413 -0.001647
            0.030000  0.004855 -0.001612  0.035000  0.005280 -0.001570
            0.040000  0.005687 -0.001526  0.045000  0.006081 -0.001486
            0.050000  0.006458 -0.001449  0.060000  0.007155 -0.001390
            0.070000  0.007796 -0.001347  0.080000  0.008393 -0.001318
            0.090000  0.008943 -0.001309  0.100000  0.009449 -0.001313
            0.120000  0.010322 -0.001373  0.140000  0.010998 -0.001526
            0.160000  0.011528 -0.001732  0.180000  0.011947 -0.001990
            0.200000  0.012186 -0.002334  0.250000  0.012340 -0.003358
            0.300000  0.011831 -0.004680  0.350000  0.010816 -0.006171
            0.400000  0.009279 -0.007888  0.450000  0.007322 -0.009783
            0.500000  0.004998 -0.011809  0.550000  0.002773 -0.013489
            0.600000  0.000609 -0.014852  0.650000 -0.001252 -0.015684
            0.700000 -0.002678 -0.015909  0.750000 -0.005076 -0.016979
            0.800000 -0.005547 -0.016080  0.820000 -0.005488 -0.015486
            0.840000 -0.005300 -0.014803  0.860000 -0.004940 -0.013988
            0.880000 -0.004421 -0.013046  0.900000 -0.003742 -0.011968
            0.910000 -0.003337 -0.011370  0.920000 -0.002886 -0.010727
            0.930000 -0.002385 -0.010037  0.940000 -0.001819 -0.009285
            0.950000 -0.001159 -0.008440  0.955000 -0.000764 -0.007952
            0.960000 -0.000289 -0.007388  0.965000  0.000342 -0.006609
```

```
          0.970000  0.002231 -0.004650  0.975000  0.002417 -0.004436
          0.980000  0.002540 -0.004217  0.985000  0.002725 -0.003949
          0.990000  0.002900 -0.003706  0.993000  0.002984 -0.003574
          0.995000  0.003054 -0.003472  0.997000  0.002982 -0.003271
          0.998000  0.002840 -0.003026  0.999000  0.002083 -0.002360
          0.999500  0.001594 -0.001785  0.999750  0.000937 -0.001346
          1.000000  0.000000  0.000000
   YW(J)        AL      ALED       FAD       CHD      FSEC     THICK
0.498294 -6.552753 -0.122441 -0.005916  0.075275  1.010000  1.000000
    ZSYM       FNU       FNL
0.000000 65.010000 65.010000
                 X    ZUPPER    ZLOWER
          0.000000  0.000000  0.000000  0.000250  0.000286 -0.000258
          0.000500  0.000547 -0.000489  0.001000  0.000997 -0.000870
          0.002000  0.001440 -0.001201  0.003000  0.001760 -0.001414
          0.005000  0.002212 -0.001643  0.007000  0.002525 -0.001740
          0.010000  0.002881 -0.001781  0.015000  0.003393 -0.001772
          0.020000  0.003872 -0.001752  0.025000  0.004322 -0.001735
          0.030000  0.004747 -0.001717  0.035000  0.005155 -0.001692
          0.040000  0.005544 -0.001664  0.045000  0.005921 -0.001640
          0.050000  0.006282 -0.001620  0.060000  0.006947 -0.001593
          0.070000  0.007557 -0.001580  0.080000  0.008123 -0.001581
          0.090000  0.008645 -0.001600  0.100000  0.009124 -0.001631
          0.120000  0.009947 -0.001740  0.140000  0.010581 -0.001935
          0.160000  0.011074 -0.002179  0.180000  0.011465 -0.002465
          0.200000  0.011673 -0.002842  0.250000  0.011740 -0.003953
          0.300000  0.011147 -0.005361  0.350000  0.010093 -0.006893
          0.400000  0.008537 -0.008631  0.450000  0.006562 -0.010547
          0.500000  0.004244 -0.012569  0.550000  0.002051 -0.014219
          0.600000 -0.000076 -0.015547  0.650000 -0.001902 -0.016347
          0.700000 -0.003244 -0.016481  0.750000 -0.004436 -0.016344
          0.800000 -0.004561 -0.015104  0.820000 -0.004383 -0.014388
          0.840000 -0.004085 -0.013594  0.860000 -0.003616 -0.012668
          0.880000 -0.002961 -0.011589  0.900000 -0.002121 -0.010349
          0.910000 -0.001623 -0.009657  0.920000 -0.001069 -0.008910
          0.930000 -0.000455 -0.008107  0.940000  0.000230 -0.007234
          0.950000  0.001007 -0.006277  0.955000  0.001348 -0.005852
          0.960000  0.001555 -0.005560  0.965000  0.001762 -0.005259
          0.970000  0.001986 -0.004943  0.975000  0.002188 -0.004656
          0.980000  0.002388 -0.004367  0.985000  0.002595 -0.004078
          0.990000  0.002809 -0.003793  0.993000  0.002940 -0.003615
          0.995000  0.003025 -0.003501  0.997000  0.002966 -0.003287
          0.998000  0.002835 -0.003036  0.999000  0.002085 -0.002362
          0.999500  0.001604 -0.001784  0.999750  0.000948 -0.001340
          1.000000  0.000000  0.000000
   YW(J)        AL      ALED       FAD       CHD      FSEC     THICK
0.500000 -6.696277 -0.122671 -0.005776  0.073338  1.010000  1.000000
    ZSYM       FNU       FNL
0.000000 65.010000 65.010000
                 X    ZUPPER    ZLOWER
          0.000000  0.000000  0.000000  0.000250  0.000286 -0.000257
          0.000500  0.000547 -0.000489  0.001000  0.000997 -0.000870
          0.002000  0.001440 -0.001200  0.003000  0.001760 -0.001413
          0.005000  0.002212 -0.001643  0.007000  0.002525 -0.001740
          0.010000  0.002881 -0.001781  0.015000  0.003393 -0.001772
          0.020000  0.003871 -0.001753  0.025000  0.004321 -0.001735
          0.030000  0.004746 -0.001718  0.035000  0.005152 -0.001693
          0.040000  0.005542 -0.001667  0.045000  0.005918 -0.001643
          0.050000  0.006278 -0.001623  0.060000  0.006942 -0.001598
          0.070000  0.007550 -0.001586  0.080000  0.008115 -0.001589
          0.090000  0.008635 -0.001608  0.100000  0.009113 -0.001641
          0.120000  0.009936 -0.001750  0.140000  0.010570 -0.001945
          0.160000  0.011065 -0.002186  0.180000  0.011462 -0.002467
          0.200000  0.011673 -0.002841  0.250000  0.011742 -0.003950
```

```
         0.300000  0.011147 -0.005360  0.350000  0.010100 -0.006885
         0.400000  0.008563 -0.008605  0.450000  0.006599 -0.010510
         0.500000  0.004293 -0.012520  0.550000  0.002120 -0.014150
         0.600000  0.000018 -0.015454  0.650000 -0.001785 -0.016232
         0.700000 -0.003098 -0.016333  0.750000 -0.003877 -0.015784
         0.800000 -0.003849 -0.014392  0.820000 -0.003618 -0.013623
         0.840000 -0.003270 -0.012779  0.860000 -0.002769 -0.011822
         0.880000 -0.002158 -0.010788  0.900000 -0.001429 -0.009660
         0.910000 -0.001020 -0.009057  0.920000 -0.000583 -0.008428
         0.930000 -0.000118 -0.007775  0.940000  0.000378 -0.007093
         0.950000  0.000915 -0.006375  0.955000  0.001174 -0.006027
         0.960000  0.001396 -0.005717  0.965000  0.001617 -0.005403
         0.970000  0.001832 -0.005096  0.975000  0.002057 -0.004784
         0.980000  0.002283 -0.004471  0.985000  0.002515 -0.004157
         0.990000  0.002754 -0.003846  0.993000  0.002902 -0.003652
         0.995000  0.002997 -0.003527  0.997000  0.002949 -0.003302
         0.998000  0.002823 -0.003046  0.999000  0.002078 -0.002367
         0.999500  0.001600 -0.001787  0.999750  0.000945 -0.001342
         1.000000  0.000000  0.000000
     a0         p0
  13040.0      14.69
```

References

1. McCormick, C. W., ed.: MSC/NASTRAN User's Manual, Vol. I and II, MacNeal-Schwendler, 1983.

2. Lawrence, C., Aiello, R.A., Ernst, M.A., and McGee, O.G., "A NASTRAN Primer for the Analysis of Rotating Flexible Blades," *NASA TM 89861*, 1987.

3. Srivastava, R., Sankar, N.L., Reddy, T.S.R., and Huff, D.L., "Application of an Efficient Hybrid Scheme for Aeroelastic Analysis of Advanced Propellers," *AIAA Paper 90-0028, 28th Aerospace Sciences Meeting* or *NASA TM 102428*, January 8-11, 1990.

4. Yamamoto, O. and August, R., "Structural and Aerodynamic Analysis of a Large-Scale Advanced Propeller Blade," *Journal of Propulsion and Power*, Vol. 8, No. 2, pp. 367-373, March-April 1992.

5. Kaza, K.R.V., Mehmed, O., Narayanan, G.V., and Murthy, D.V., "Analytical Flutter Investigation of a Composite Propfan Model," *Journal of Aircraft*, Vol.26, No. 8, pp. 772-780, August 1989.

6. Cray Research, Inc.: UNICOS User Manual Commands Reference Manual, SR-2011 7.0, 1992.

7. Kochan, C.G. and Wood, P.H., UNIX Shell Programming, Hayden Books, 1990.

8. Srivastava, R. and Mehmed, O., "On the Static Stability of Forward Swept Propfans," *Proceedings of 34th Structures, Structural Dynamics, and Materials Conference*, La Jolla, CA., April 19-22, 1993.

9. Aiello, R. A., "Composite Blade Structural Analyzer (COBSTRAN) User's Manual," *NASA TM 101461*, April 1989.

Appendix A: Script Files for Batch Mode

This section contains a detailed description of the job control and shell files used in the batch process of SAB. In addition, the related files, *i.e.* files that are either called or called by the scripts, are listed in the appropriate tables along with input/output files used within the scripts. A listing of each script file is also presented to the user. Bold type in the script file listings signal the user that they must supply the appropriate information in the file. Following the description of the file, a section is provided to the user for instructions and revisions that are necessary in order to execute the scripts in the SAB process.

The user can change any of the file names within the scripts but make certain the corresponding files are also changed. It is recommended to the user not to change the names of the files unless the user has a full understanding of each file program and its tasks. The files are commented (the # symbol signifies a comment statement) to guide the user through the logic of the files. A comprehensive explanation of the commands is beyond the scope of this user's guide and the user is referred to references [6-7] or the man pages on the Cray computers.

For convenience, a review of the input/output files and their definitions are provided in Table A-1. The files are listed in the order as they appear in the script files. The user should note that many of the output files are used as input to other codes within the SAB process. The user is referred back to Chapter 3: SAB Input/Output File Structure for additional information. Also, the user is referred back to Figure 3.3-1 which depicts the global flow of the input/output files and their respective script files.

Table A-1: SAB Input/Output Files in their Order of Appearance

Input/Output Files	Description
parameter.file	Generic name given to user's input file that defines specific run parameters.
def_param	SAB assigned name for user's parameter.file.
FEinput.file	Generic name for user's finite element input data in its original form.
nastran_i.cold	SAB assigned name for user's finite element input data (FEinput.file).
AEROinput.file	Generic name given for user's aerodynamic analysis input in its original form.
prop3d_i.cold	SAB assigned name for user's aerodynamic analysis input (AEROinput.file).
nastran.job (nas_cards.job)	Refer to pg. 57 for definition.
hist.sab.xmp	An output file telling user the status of MSC/NASTRAN job on Cray X-MP.
nastran.mscout (mscout.*)	Standard MSC/NASTRAN output file.
nastran.mscpunch (mscpunch.*)	MSC/NASTRAN punch file of displacements.
face_log.*	System output provided from face.job and related shell calls.
converg.out	Results form the convergence check.
nst_defl.*	The finite element deflected shape.
aero_geom	Aerodynamic mesh for aero solver (PROP3D).
prop3d.i	Generated PROP3D input deck, combination of prop3d_i.cold and aero_mesh.
face1_out.*	Output results from face1.o.
f31_bin.*	Binary file produced from the PROP3D code.
prop3d_out.*	Standard results file from PROP3D code.
prop3d.log (prop3d_log.*)	System output provided from PROP3D job file and related shell calls.
ploads.*	PLOAD cards for the MSC/NASTRAN input deck.
chk.pld.*	Differential pressures that are transformed into PLOAD cards for MSC/NASTRAN input.

start_sab.sh

The start_sab.sh file begins SAB batch process by typing the filename followed by a valid directory name at the Cray Y-MP prompt. This UNIX file creates a generic directory (**generic** can be any valid directory name the user desires) were all execution and storage of all files are located for the current SAB process. Input files created by the user are copied into the generic directory along with all sources and scripts files used during execution, refer to section 4.2 Batch Mode Process, pg. 30, for further details on the linking of files. The user must place the proper input file names and the directory location of the files in order for start_sab.sh to move the user files to the SAB system file names, refer to Table A-2. Standard names are given to the three input files provided by the user for access throughout the SAB script files.

After setting the proper directory and copying the appropriate files two shell files are executed, mk_nas.sh and run_nas.sh. The first shell file creates the nastran.job file and the second file submits the nastran.job file for execution. The shell file start_sab.sh is shown in Table A-3.

User Instructions/Revisions:

1. Must add the proper input file names and directory name in the start_sab.sh file for the finite element input, aerodynamic analysis input, and parameter definition file.
2. The user is advised to use the SAB system file names (user input files are copied to these files) that are located in the start_sab.sh script for ease of use.
3. Set proper directory names were scripts and sources are stored.
4. Submit the valid directory name after the start_sab.sh command. For example, at the Cray Y-MP prompt enter:

   lercymp$ **start_sab.sh generic**

Table A-2: Associated Files to start_sab.sh Shell Module

Called By	Page	Programs Executed within start_sab.sh	Page
		mk_nas.sh	52
		run_nas.sh	53

Input	Output
parameter.file/def_param	
FEinput.file/nastran_i.cold	
AEROinput.file/prop3d_i.cold	

Table A-3: Shell Module start_sab.sh Used to Initiate the SAB System

```
mkdir $1                    # shell will accept any valid directory name
# ---------------------------------------------------------------------------
#   copy user supplied input files to generic directory and rename to standard
#   input file names used by SAB system, user must supply the proper input
#   files (3) located in the inputs directory or any directory the user chooses
# ---------------------------------------------------------------------------
cp inputs/parameter.file  $1/def_param
cp inputs/FEinput.file    $1/nastran_i.cold
cp inputs/AEROinput.file  $1/prop3d_i.cold
cd $1
# ------------------------------------------------------------
#   copy all source and scripts files to generic directory
# ------------------------------------------------------------
ln -s ../sources/*.o .
ln -s ../scripts/* .
# -------------------------------------------------------------------
#   execute shells for setup and execution of structural analysis
# -------------------------------------------------------------------
mk_nas.sh   nastran_i.cold
run_nas.sh  nastran.job
```

mk_nas.sh

The mk_nas.sh file couples the users MSC/NASTRAN input deck (nastran_i.cold) and the nas_cards.job file to create the nastran.job file. This shell is executed two different times during iterative approach, refer to Table A-4. Initially the start_sab.sh file calls the mk_nas.sh for creating the introductory nastran.job for the structural analysis. Later for each iteration, face2.sh calls mk_nas.sh to generate the updated nastran.job file for execution. The nas_cards.job file is split and the nastran_i.cold file is inserted. Along with the nastran_i.cold input deck, the nastran.job file also contains the commands needed to submit the input deck to the Cray X-MP and the face.job file to the Cray Y-MP. The listing of mk_nas.sh is located in Table A-5.

Table A-4: Affiliated Files to mk_nas.sh Shell Module

Called By	Page	Programs Executed within mk_nas.sh	Page
start_sab.sh	50		
face2.sh	67		

Input	Output
nas_cards.job	nastran.job
nastran_i.cold	

Table A-5: Listing of mk_nas.sh File

```
# ---------------------------------------------------------------------------
#  split the nas_cards.job file, insert msc/nastran deck, and create the
#  nastran.job file
# ---------------------------------------------------------------------------
csplit -s -k -f ttmp nas_cards.job '/cat > nastran <<EOD/+1' '/EOD/';
cat $1 >> ttmp00;
cat ttmp02 >> ttmp00;
mv ttmp00 nastran.job;
rm ttmp0?;
```

run_nas.sh

This UNIX shell submits the shell, nas_xmp.sh, which sends the MSC/NASTRAN input deck to the Cray X-MP for execution. The "nohup" command is used because otherwise all processes are automatically terminated when the user log's off the system. Therefore, this file protects the SAB process from prematurely ending due to the user logging off. The output file, hist.nas.xmp, is a message to the user if the job has been successfully transferred and submitted for execution on the Cray X-MP.

run_nas.sh is called in two different places during one iteration of the SAB process, refer to Table A-6. First, the start_sab.sh executes run_nas.sh for the initial run of the structural analysis (MSC/NASTRAN), the second, in shell face2.sh after the structural analysis is updated from the aerodynamic results. The total number of structural analysis runs is always $nn+1$, where nn is the total number of iterations. Located in Table A-7 is the run_nas.sh file.

Table A-8: Related Files to run_nas.sh

Called By	Page	Programs Executed within run_nas.sh	Page
start_sab.sh	50	nas_xmp.sh	54
face2.sh	67		

Input	Output
nastran.job	hist.nas.xmp

Table A-9: UNIX Shell File run_nas.sh

```
date >> hist.nas.xmp ;
nohup run_xmp.sh $1 $2 >> hist.nas.xmp &
```

nas_xmp.sh

The nas_xmp.sh file has the task of sending the nastran.job (MSC/NASTRAN structural analysis job) for execution on the Cray X-MP. Currently at NASA LeRC, the commercial code MSC/NASTRAN only exists on the Cray X-MP, therefore, this file is required to send the input deck and related commands to the Cray X-MP and return execution to the Cray Y-MP computer after the MSC/NASTRAN job has been executed.

Table A-8: Related Files to nas_xmp.sh

Called By	Page	Programs Executed within nas_xmp.sh	Page
run_nas.sh	53	MSC/NASTRAN	69

Table A-9: File Listing of nas_xmp.sh

```
#! /bin/sh
usage='USAGE: run_msc script_file [ userid ]';
if [ $# -gt 2 -o $# -lt 1 ] ; then
   echo "Incorrect number of arguments";
   echo $usage;
   exit 1;
   fi;
if [ ! -f $1 ] ; then
   echo "Error: Script file $1 is non-existent, or non-ordinary";
   exit 1;
elif [ ! -r $1 ] ; then
   echo "Error: Script file $1 is non-readable";
   exit 1;
elif [ ! -s $1 ] ; then
   echo "Error: Script file $1 is empty";
   exit 1;
   fi;
lt=`grep "\-lt" $1`;
if [ -n "$lt" ] ; then :
else
   grep "\-lT" $1 > .time;
   time=`awk ' { print $NF } ' .time`;
   time=`expr $time - 5`;
   csplit -s -k -f ttmp $1 /-lT/+1;
   cat ttmp00 > $1;
   echo "# QSUB -lt $time" >> $1;
   cat ttmp01 >> $1;
   /bin/rm ttmp0? .time;
   fi;
#
#-------------------------------------------------------------------------
#     Test to see which command is used for remote shell invocation
#-------------------------------------------------------------------------
#
#man remsh > /dev/null 2>/dev/null;
#if [ $? = 0 ] ; then
#   shl=remsh;
#else
   shl=remsh;
#   fi;
#
#-------------------------------------------------------------------------
#     Set host variables for proper disposition of output files
#-------------------------------------------------------------------------
#
wkshost=`hostname`;
wksdir="`pwd`/";
if [ $2 ] ; then
   copy=`echo $2@`;
   wksid="wksid=$LOGNAME@";
   user=" -l $2 ";
   fi;
#
#-------------------------------------------------------------------------
#     Set the name of the log file for the NASTRAN run
#-------------------------------------------------------------------------
#
cprfile=`grep "# QSUB -r" $1 | cut -f4 -d" "`;
if [ $cprfile ] ; then
```

```
   cprfile=${cprfile}.log;
else
   cprfile=`grep "# QSUB-r" $1 | cut -f3 -d" "`;
   if [ $cprfile ] ; then
      cprfile=${cprfile}.log;
   else
      cprfile=`echo $1`.log;
      fi;
   fi;
#
#-------------------------------------------------------------------------
#     Copy NQS file to the CRAY X-MP for execution
#-------------------------------------------------------------------------
#
rcp $1 ${copy}lercxmp-hy:$1;
if [ $? = 0 ] ; then
   pidout=`$shl lercxmp-hy $user "sh -kc 'qsub -x -o $cprfile -eo $1 \
                 wkshost=$wkshost wksdir=$wksdir fend=wks $wksid ' " `;
   pid=`echo $pidout | cut -d" " -f2`;
   $shl lercxmp-hy $user "/bin/rm $1";
   echo $pidout;
else
   echo "Remote copy $1 to ${copy}lercxmp-hy:$1 failed";
   exit 1;
   fi;
```

nas_cards.job / nastran.job

The nas_cards.job and nastran.job files are exactly the same with the exception of the MSC/NASTRAN input deck (nastran_i.cold) that is inserted into the nastran.job file by the mk_nas.sh file. As a result, the combination of the nastran_i.cold and nas_cards.job is equivalent to nastran.job. This job file has three main functions:

(1) contains the structural analysis input and related commands to run the analysis on the Cray X-MP
(2) check if nastran.mscpunch file was returned, refer to Table A-10
(3) executes the face.job file

The file is made of UNIX and job control commands, in addition to the structural code input commands. Dependent files for this job file are located in Table A-10 and the actual nas_cards.job/nastran.job file is shown in Table A-11.

The section above the MSC/NASTRAN input are the job control commands that setup the control parameters to the Cray X-MP computer to run MSC/NASTRAN. Below the input deck are the UNIX commands to check if the nastran.mscpunch file of displacements has been returned properly, if not returned then flag the user that there is a problem with the MSC/NASTRAN run and terminate the SAB process. If nastran.mscpunch is returned then continue the execution by submitting face.job. In addition to the nastran.mscpunch, the standard MSC/NASTRAN (nastran.mscout) output file is returned.

User Instructions/Revisions:

1. To ensure the quickest "turnaround time" set the CPU requirements to the lowest possible amount. In contrast, make certain the job has an adequate amount of CPU time to avoid premature termination.

Table A-10: Related Files to nas_cards.job/nastran.job

Shell that Calls msc_cards.job	Page	Programs Executed within nas_cards.job	Page
mk_nas.sh	52	face.job	59
run_nas.sh	53		

Input	Output
	nastran.mscout
	nastran.mscpunch

Table A-11: Shell Module msc_cards.job/nastran.job

```
# QSUB -me
# QSUB -r nas_job          # job name
# QSUB -eo
# QSUB -lT 1200            # max cpu time needed by the nastran
# QSUB -lt 1195
# QSUB -lm 2.0mw           # memory limit set
set -k
cd                         # sets the user in their base directory
cat > nastran <<EOD       # give the input deck a name and copy it to the file
# -------------------------------------------------------------
#  if nastran.job:    MSC/NASTRAN input deck is entered here
#  if nas_cards.job:  NO input deck is entered here
# -------------------------------------------------------------
EOD                        # signals end of input
# --------------------------------------------------
#  execute structural analysis for the given input
# --------------------------------------------------
mscnast in=nastran;
rm nastran;                   # remove copied input deck, no longer needed
# ----------------------------------------------------------------
#  if msc/nastran punch file is not returned, then flag user that
#  there is an error in the nastran run and cease operation
# ----------------------------------------------------------------
t=0;
while true ; do
   exist=`remsh lercymp "ls $wksdir/nastran.mscpunch 2>/dev/null"`;
   if [ $exist ] ; then
      break;
   else
      t=`expr $t + 10` ;
      if [ $t -gt 120 ] ; then
          break
          fi;
      wait 10;
      fi ;
   done;
if [ $t -gt 120 ] ; then
   echo "Error occurred in NASTRAN run, check output" ;
else
# ---------------------------------------------
#  submit facel.job to continue SAB system
#  assign output the name face.log
# ---------------------------------------------
remsh lercymp "sh -kc 'qsub -x -o $wrkdir/face.log \
 -eo $wrkdir/face.job wksdir=$wksdir -s/bin/sh'";
   fi;
```

face.job

The task of face.job is threefold:

(1) check convergence of deflected shape
(2) transfer the deflected shape onto the aerodynamic geometry
(3) initiate the aerodynamic analysis

By initiating the execution of conv_chk.sh, the convergence check is performed, refer to Table A-12. Upon completion, control is returned to face.job and a flag is checked if convergence was met in conv_chk.sh. If convergence was met then terminate the process within face.job, if convergence was not met then start transferring the deflected blade onto the surface grid points of the aerodynamic model which is accomplished by executing face1.sh. Finally, submit the aerodynamic analysis job (prop3d.job) for execution.

A system output file (face.log) is provided to the user to indicate the success or failure of the above three functions. Results are provided from the print statements located in the shells (conv_chk.sh and face1.sh) and job file that face.job calls. Due to the tedious shell calls and number of files used in the SAB process, this should help the user track down problems if they exist. The program listing is shown in Table A-13.

Table A-12: Associated Files to face.job

Called By	Page	Programs Executed within face.job	Page
nastran.job	57	sh_conv	61
		sh_face1	63
		prop3d.job	65

Input	Output
	face.log

Table A-13: Module face.job Listing

```
# QSUB -r  facejob            # enter job name
# QSUB -lM 30.0Mw             # memory space required
# QSUB -lT 60                 # CPU time required
# QSUB -eo
set -k;
cd $wrksdir:
conv_chk.sh;                  # check convergence
# ----------------------------------------------------
#   check flag from conv_chk.sh is convergence met
# ----------------------------------------------------
if [ $? = 7 ] ; then
   exit;
   fi;
face1.sh;                     # transfer deflected data to aero model
ver= 'cat .version';          # current version/iteration number
# ---------------------------------
#  execute aerodynamic analysis
# ---------------------------------
qsub -x -o $wksdir/prop3d_log.$ver -eo $wksdir/prop3d.job;
```

conv_chk.sh

The shell conv_chk.sh initiates the convergence check of the data by executing the FORTRAN object file converg.o, refer to Table A-14. A flag is sent back to the calling shell, face.job, to indicate if convergence is met or another iteration is required. In order to perform the convergence check, three input files are linked. From the execution of converg.o, an output file labelled converg.out is produced to display the pertinent convergence criteria data, this file is updated for each iteration.

In addition, the MSC/NASTRAN output files, nastran.mscout and nastran.mscpunch, are assigned the current iteration number and renamed mscout.*nn*+1 and mscpunch.*nn*+1, as demonstrated in Table A-15. The *nn* characters is a wild card symbol for the current iteration number. This is also done for the prop3d.log output file, renamed prop3d_log.*nn*, only if the current iteration is greater than one. To determine if the current iteration number is greater than one, the shell checks if the output file converg.out exists. The "echo" statements in Table A-15 are printed in the face.log output file from the face.job file that initiated the execution of conv_chk.sh.

Table A-14: Related Files to conv_chk.sh

Called By	Page	Programs Executed within conv_chk.sh	Page
face.job	59	converg.o	-

Input	Output
nastran.job	converg.out
nastran.mscpunch	mscpunch.*nn*+1 (nastran.mscpunch)
def_param	
nastran.mscout	mscout.*nn*+1 (nastran.mscout)
prop3d.log	prop3d_log.*nn* (prop3d.log)

nn - indicates the iteration number when the file was generated.

Table A-15: conv_chk.sh Listing

```
ln nastran.job fort.4;          # link the current geometry to fort.4 for input
ln nastran.mscpunch fort.3;     # link the punch file to fort.3 for input
ln def_param fort.13;           # link for input
segldr -o converg converg.o;        # load the convergence code to be executed
# --------------------------------------------------------------------------
#  if output converg.out exists then perform first set of commands (up to else)
#       - add current iteration number to prop3d.log file previously generated
#       - link output file
#       - execute convergence code beta
#  if converg.out doesn't exist then perform second set of commands (after else)
#       - execute convergence code beta
#       - move output to file converg.out
# --------------------------------------------------------------------------
if [ -f converg.out ] ; then
   ver=`cat .version`;
   mv prop3d.log prop3d.log.$ver
   ver=`expr $ver + 1`;
   echo $ver > .version;
   ln converg.out fort.2;
   converg;
else
   converg;
   mv fort.2 converg.out;
   echo "1" > .version;
   ver=`cat .version`;
   fi;
rm converg fort.*;
# -------------------------------------------------------------
#   rename msc/nastran output to current iteration number
#                    mscout.nn+1
#                   mscpunch.nn+1
#       nn - indicates current iteration/version
# -------------------------------------------------------------
mv nastran.mscout mscout.$ver;
mv nastran.mscpunch mscpunch.$ver;
grep converged converg.out;
if [ $? = 0 ] ; then
   echo "Converged deflected shape";
   exit 7;
else
   echo "Not converged yet, trying another loop";
   exit 0;
   fi;
```

face1.sh

The shell face1.sh initiates the transformation of the structural deflection data (finite element results) to the blade surface grid points (aerodynamic model) by executing the FORTRAN object file face1.o, refer to Table A-14. Four input files are needed to perform this task. The nastran_i.cold and mscpunch.*nn*+1 are used in the calculation of deformed blade model, the results are stored in the nst_defl.*nn* output file. From the deformed blade model the code interpolates the deformed grid points onto the aero chord to create the aerodynamic mesh, these results are stored in the output file aero_mesh.

After the transformation, the prop3d_i.cold and aero_geom.*nn* files are coupled together to form prop3d.i. Essentially, the aerodynamic geometry in prop3d_i.cold is stripped out and replaced with the aerodynamic geometry in aero_geom.*nn* file. This new file prop3d.i is the input used in the aerodynamic analysis code PROP3D. The face1.out file contains pertinent information used in the transformation from the structural model to the aerodynamic model. Also, the "echo" statements print information in the face.log output file of face.job to post the progress of the SAB process.

Table A-16: Related Files to face1.sh

Called By	Page	Programs Executed within face1.sh	Page
face.job	59	face1.o	-

Input	Output
nastran_i.cold	nst_defl.*nn*
def_param	aero_geom.*nn*
mscpunch.*nn*+1	prop3d.i
prop3d_i.cold	face1.out

nn - indicates the iteration number when the file was generated.

Table A-17: Shell Listing for face1.sh

```
echo '... starting face1 script execution';
ver=`cat .version`;
# --------------------------------------------------------------------------
#   link the current input files to the appropriate fort.* files for input
#   to the face1.o FORTRAN object file
# --------------------------------------------------------------------------
ln nastran_i.cold fort.2;
ln mscpunch.$ver fort.3;
ln prop3d_i.cold fort.7;
ln def_param fort.13;
#
echo '... INITIATE DEFLECTED GRID TO AERO GEOMETRY INTERPOLATION';
#
segldr -o face1 face1.o;      # load face1.o file
face1 > face1.out.$ver;       # execute face1 and assign the output name
mv fort.4 nst_defl.$ver;      # move nastran deflections to nst_defl.*
mv fort.8 aero_geom.$ver;     # move aerodynamic mesh to aero_geom.*
#
echo '... COMPLETED DEFLECTED GRID TO AERO GEOMETRY INTERPOLATION';
echo ' ';
# --------------------------------------------------------------------------
#   create the prop3d.i input file for the PROP3D code from the aero_geom
#   and initial aerodynamic input file prop3d_i.cold
# --------------------------------------------------------------------------
echo '.... editing for deflected geometry';
csplit -s -k -f ttmi prop3d_i.cold '/YW(J)/' '/a0/';
cat aero_mesh.$ver >> ttmi00;
cat ttmi02 >> ttmi00;
mv ttmi00 prop3d.i;
rm face1 fort.* ttmi0?;
echo '.... completed updating prop3d.i';
echo ' ';
echo '... completed face1 script execution';
```

prop3d.job

The job file prop3d.job executes the aerodynamic analysis by using the PROP3D code, refer to Table A-18 and Appendix C, pg. 70. Before loading and executing PROP3D code, the face1.log file is renamed (face_log.*nn*) to add the current iteration number to the file name.

The prop3d.i file is used as input for the PROP3D execution, recall that prop3d.i was created in the face1.sh by combining the original input file (prop3d_i.cold) supplied by the user and the aero_geom.*nn* file generated from face1.o. Three output files are generated from the PROP3D analysis. A standard output file (prop3d_out.*nn*) from the aerodynamic analysis is provided along with a with system output file (prop3d.log). Also, a binary file of results is generated for input into the face2.o code executed by the face2.sh shell called in the prop3d.job file.

User Instructions/Revisions:

1. To ensure the quickest "turnaround time" set the CPU requirements to the lowest possible amount. In contrast, make certain the job has an adequate amount of CPU time to avoid premature termination.

Table A-18: Affiliated Files to prop3d.job

Called By	Page	Programs Executed within prop3d.job	Page
face.job	59	prop3d.o	-
		face2.sh	67

Input	Output
prop3d.i	f31_bin.*nn*
	prop3d_out.*nn*
	prop3d.log
face.log	face_log.*nn* (face.log)

nn - indicates the iteration number when the file was generated.

Table A-19: Job File Listing for prop3d.job

```
# QSUB-r  prop3d              # enter job name
# QSUB-eo
# QSUB-lT 1200                # suggested CPU time, user may want to adjust
# QSUB-lM 4.0Mw
set -k
echo cd $wksdir               # printout directory
cd $wksdir
ver=`cat .version`;
# ------------------------------------------------------------------------
#   rename the current output file face1.log to face1_log.nn to include the
#   current iteration number
#   load and execute the aerodynamic analysis
# ------------------------------------------------------------------------
mv face1.log face1_log.$ver
segldr -V -o prop3d prop3d.o
time prop3d < prop3d.i >prop3d_out.$ver
mv fort.31 f31_bin.$ver;
rm fort.* prop3d
face2.sh                      # execute face2 script
```

face2.sh

The UNIX shell face2.sh has four tasks to perform:

(1) transfer the aerodynamic differential pressure into nodal grid pressures
(2) update the MSC/NASTRAN input deck with the updated pressure loads
(3) remake the MSC/NASTRAN deck for execution
(4) submit the MSC/NASTRAN deck for execution

Item (1) is performed by the face2.o FORTRAN object file, item (3) is accomplished by calling the shell mk_nas.sh, and item (4) is achieved by executing run_nas.sh, refer to Table A-20. Whereas item (3) is performed within the shell by splitting the original MSC/NASTRAN deck (nastran_i.cold) and adding the new PLOAD cards found in the output file ploads.*nn*. This is shown in Table A-21. The "echo" statements in this shell are printed to the prop3d.log file generated in the prop3d.job file. These statements provided information on the progress of the above four tasks through this shell.

Table A-20: Associated Files to face2.sh

Called By	Page	Programs Executed within face2.sh	Page
prop3d.job	65	face2.o	-
		mk_nas.sh	52
		run_nas.sh	53

Input	Output
f31_bin.*nn*	ploads.*nn*
nst_defl.*nn*	chk.pld.*nn*
nastran_i.cold	nastran.job
def_param	

nn - indicates the iteration number when the file was generated.

Table A-21: Shell Module face2.sh Listing

```
echo '... starting interface2 execution';
echo ' ';
echo '... EXECUTE PRESSURE DISTRIBUTION TO GRID PRESSURE INTERPOLATION';
ver=`cat .version`;
# ---------------------------------------------------------------------
#   link input files to the corresponding fort.* file for input to pload
# ---------------------------------------------------------------------
ln f31_bin.$ver fort.3;
ln nst_defl.$ver fort.4;
ln nastran_i.cold fort.7;
ln def_param fort.13;
# ------------------------------------------------------------------------
#  load and execute the pload FORTRAN code to transfer the differential
#  pressures into nodal pressures to be used as input to the finite element
#  code
# ------------------------------------------------------------------------
segldr -o face2 face2.o;
face2;
mv fort.8 ploads.$ver;
mv fort.9 chk.pld.$ver;
echo '... COMPLETED INTERPOLATION';
echo ' ';
# ---------------------------------------------------------------------
#   update the nastran input deck with the new pressure loads from pload
#   output
# ---------------------------------------------------------------------
echo '... UPDATE NASTRAN PRESSURE LOADS';
csplit -s -k -f ftmp nastran_i.cold /PLOAD/ /ENDDATA/;
cat ploads.$ver >> ftmp00;
cat ftmp02 >> ftmp00;
# ------------------------------------------------------
#   make updated nastran.job file and submit for execution
# ------------------------------------------------------
mk_nas.sh  ftmp00 ;
echo ' ';
echo '... completed interface2 execution';
rm face2 fort.* ftmp0?;
run_nas.sh  nastran.job;
```

Appendix B: MSC/NASTRAN

The MSC/NASTRAN package is a general purpose finite element program used to perform the structural analysis required by SAB. MSC/NASTRAN was selected for its ability to accurately model the complex geometry (*i.e.*, highly twisted and swept blades), and the composite and nonisotropic material properties of advanced turbomachinery blades. The blades also experience large, nonlinear deflections due to blade flexibility, and combined centrifugal and aerodynamic loads which must be accounted for in the analysis. In conclusion, MSC/NASTRAN is well suited to handle these types of geometry, material properties, and loads encountered in the analysis of turbomachinery blades while maintaining computational efficiency.

The objective of performing the structural analysis is to determine the steady-state deflections for blade. This accomplished by using a geometric nonlinear analysis as well as the updated displacement-dependent centrifugal forces and aerodynamic loads. Solution 64 in MSC/NASTRAN [1] is used to perform this task. The solver uses a modified Newton- Raphson algorithm, along with load updating, to simulate the correct displacement versus load relationship.

The algorithm iterations are controlled through "subcases," with a minimum of two being required. The first subcase computes the initial, linear deflected shape. Subsequent subcases, or iterations. then use the preciously deflected shape to compute the differential stiffness matrix along with the new set of displacements. In addition, NASTRAN Direct Matrix Abstraction Program (DMAP) procedures are included in the analysis to account for the centrifugal softening due to the centrifugal forces acting in the same direction as the displacement. A detailed explanation of the above procedure is given in reference [2]. Pressure loads are applied to the face of the elements to simulate the aerodynamic loads, calculated from an external aerodynamic code.

Verification of the finite element model is left up to the user. Further information on the execution and setup of the MSC/NASTRAN is beyond the scope of this manual. At NASA LeRC the user is suggested to contact the User Help Desk for additional details on MSC/NASTRAN and the current version that is available to them.

Appendix C: PROP3D Input Preparation

The aerodynamic analysis used in SAB is performed by the **PROP3D** code developed by Srivastava, et. al., [3,8]. The code uses an efficient three-dimensional hybrid scheme for solving the Euler equations, in a fully conservative form. In the hybrid scheme two directions are treated implicitly and the spanwise direction is treated semi-implicitly. This leads to a reduction in CPU time and memory requirements. The method has been successfully applied to analyzing turbomachinery blade configurations. The user is again referred to reference [3] for further details on the methodology, assumptions, and capabilities of the **PROP3D** aerodynamic analysis code.

The remainder of this appendix describes the input deck for the **PROP3D** code and presents an example input deck. A total of 31 mandatory records and 2 optional records make up the input deck with the breakdown of the assortment of records located in Table C-1. In addition, Table C-2 presents an outline of the input required for the **PROP3D** input. Variable names used in the description of each record are displayed in the table to represent the actual numerical input.

A description, format, and an example are given for the appropriate records shown in Table C-2. The comment lines may be used to identify the inputs (demonstrated as the example in each comment record), however there are no constraints on the contents of these lines. All records presented, with the exception of Records 13A and 13B, are mandatory in the input deck and must be in the order of presentation.

After all data records are introduced, an example input deck is presented in Table C-3. An axisymmetric flow field is assumed around the 12 bladed propfan, therefore, all blade passages can be assumed to be identical and only one blade is solved enforcing the conditions of symmetry. The advance ratio is 1.1270 and has been designed for a free stream Mach number of 0.20. The forward propeller has a blade setting angle of 35°, in contrast, the aft propeller has a blade setting angle of 38°. Also, the input deck is not shown in its entirety because of its size. Only one of the sixteen stations of the forward and aft propellers are shown.

Table C-1: Assortment of Records in the PROP3D Input Deck

Number of Records	Description
1	title record
14	comment lines
1	optional comment line
12	single line inputs of data (real or integer)
1	optional single line input of data
1	logical line of data
2	sets of input data, multiple string of data
1	null character input line (INASTRAN)

Table C-2: Outline of PROP3D Input File

Record	Fields						
	1	2	3	4	5	6	7
1	Title						
2	Comment						
3	ADV	GMU	ALFAI	PSI	WW	DT	REYREF
4	Comment						
5	IMAX	JMAX	KMAX	JTIP	ITELF	ILEF	IGR
6	Comment						
7	FSTP	FMINF	BETAF	DIAF	DY	VIBFRE	
8	Comment						
9	IFLTR	NUMCYC	NSTDY	JMODE	NBLOKS		
10	Comment						
11	RSTRT	QSTEADY	INFLOW	AELAST	COUNT	ADBAS	
12	Comment						
13	ICCW	ITURB	LTHIN	ISWF			
	If COUNT *in Record 11 is TRUE then Records 13A and 13B are required, if False disregard*						

Record	Fields 1	2	3	4	5	6	7
13A	Comment						
13B	XLOC	DIAA	BETAA	ITELA	ILEA	INOSE	
14	Comment						
15	FNRS						
16	Comment						
17_1	XR_1	RR_1					
17_i	•	•					
17_{i+1}	•	•					
17_{FNRS}	XR_{FNRS}	RR_{FNRS}					
	If COUNT *in Record 11 is TRUE then Records 18 through 31 are required for the aft propeller*						
18	INASTRAN						
19	VALUE						
20	Comment						
21	FNC	FROTAT	FB	FTPRP	FTWST	FCOB	FGR
22	Comment						
23	IDTYPE						
	Records 24 through 29 are repeated FNC *(Record 21) times*						
24	Comment						
25	YW(J)	AL	ALED	FAD	CHD	FSEC	THICK
26	Comment						
27	ZSYM	FNU	FNL				
28	Comment						
	Record 29 Input Format Based on IDTYPE in Record 23						
29_1	X_1	ZU_1	ZL_1				
29_i	•	•	•				
29_{i+1}	•	•	•				
29_{FNL}	X_{FNL}	ZU_{FNL}	ZL_{FNL}				
30	Comment						
31	a0	p0					

Record 1: Title (A64)

Record 1: Title (A64)

Record 2: Comment Line

Example:

ADV	GMU	ALFAI	PSI	WW	DT	REREF

Record 3:

Format (7F10.4) and Example:

ADV	GMU	ALFAI	PSI	WW	DT	REREF
1.0140	0.0	0.0	0.0	7.0	0.0030	-1.0

Field Description:

Field	Name	Description
1	ADV	Advance ratio
2	GMU	Reserved for future use, enter 0.0
3	ALFA	Angle of attack for hub or angle of center body with respect to free stream
4	PSI	Reserved for future use, enter 0.0
5	WW	Artificial dissipation factor
6	DT	Time step, suggested values 0.0001 < DT < 0.025
7	REYREF	Reynolds number, must specify as + or - - Euler equations used + Navier Stokes equations used

Note:

1. An example for the comment line in record 2 could be the above format line from record 3 which would define the input values as they appear.

Record 4: Comment Line

Example:

IMAX	JMAX	KMAX	JTIP	ITELF	ILEF	IGR

Record 5:

Format (7I10) and Example:

IMAX	JMAX	KMAX	JTIP	ITELF	ILEF	IGR
80	22	16	15	23	23	0

Field Description:

Field	Name	Description
1	IMAX	Total number of grids in axial direction
2	JMAX	Total number of grids in radial direction, hub to tip and beyond
3	KMAX	Total number of grids in azimuthal direction, blade to blade
4	JTIP	Number of grids up to the blade-fluid interface along radial direction
5	ITELF	Number of points between boundary and trailing edge (for forward propeller)†
6	ILEF	Number of points between boundary and leading edge (for forward propeller)†
7	IGR	Generate computational grid = 0 internal grid generated = 1 external grid used as input

† in the case of counter rotating propellers

Record 6: Comment Line

Example:

FSTP	FMINF	BETAF	DIAF	DY	VIBFRE

Record 7:

Format (6F10.4) and Example:

FSTP	FMINF	BETAF	DIAF	DY	VIBFRE
5.0	0.200	35.00	1.0	0.05	0.0

Field Description:

Field	Name	Description
1	FSTP	Number of iteration steps
2	FMINF	Free stream Mach number
3	BETAF	Blade angle at 75% radius of forward propeller
4	DIAF	Diameter of propeller (if equal 1.0 then input distances must be normalized with propeller diameter)
5	DY	Reserved for future use
6	VIBFRE	Vibration frequency used in flutter analysis option

Record 8: Comment Line

Example:

IFLTR	NUMCYC	NSTDY	JMODE	NBLOKS

Record 9:

Format (5I10) and Example:

IFLTR	NUMCYC	NSTDY	JMODE	NBLOKS
-1	0	0	2	1

Field Description:

Field	Name	Description
1	IFLTR	Flutter command
2	NUMCYC	Number of cycles (used only in flutter analysis)
3	NSTDY	Used in flutter analysis
4	JMODE	Used in flutter analysis
5	NBLOKS	Equal to 1 or the number of blades depending on analysis - if 1 then axisymmetric inflow - if number of blades then asymmetric inflow

Record 10: Comment Line

Example:

RSTRT	QSTEADY	INFLOW	AELAST	COUNT	RESABD

Record 11:

Format (5L5) and Example:

RSTRT	QSTEADY	INFLOW	AELAST	COUNT	RESABD

```
FALSE TRUEFALSEFALSEFALSEFALSE
```

Field Description:

Field	Name	Description
1	RSTRT	Restart option TRUE - restarts, will read grid and flowfield information FALSE - start new case
2	QSTEADY	Quasi steady option TRUE - quasi steady (axis symmetric) case, NBLOKS = 1 FALSE - unsteady case, NBLOKS = number of blades
3	INFLOW	Inflow option (false for present analysis) TRUE - distribution of angle of attack along span of blade FALSE - null, no wake information
4	AELAST	Aeroelastic option TRUE - flutter analysis FALSE - purely aerodynamic calculation
5	COUNT	Counter rotation option TRUE - counter rotation geometry FALSE - single rotation geometry
6	RESABD	Restart for deflection calculations TRUE - will read previous flow solution but generate grid FALSE - RSTRT values take precedence

Notes:

1. If COUNT = TRUE, *i.e.*, counter rotation geometry exists and a counter rotation propeller is analyzed, then Records 13A and 13B must exist in the input deck. Also, Records 18 through 31 are repeated for the aft propeller.
2. If COUNT = FALSE, then Records 13A and 13B can be ignored and Records 18 through 31 do not have to be repeated for the aft propeller.
3. The aero structure interaction analysis is carried out only on the front propeller if counter rotation propeller is analyzed. The aft propeller is assumed rigid.

Record 12: Comment Line

Example:

ICCW	ITURB	LTHIN	ISWF

Record 13:

Format (4I10) and Example:

ICCW	ITURB	LTHIN	ISWF
1	0	0	0

Field Description:

Field	Name	Description
1	ICCW	Rotation of front propeller = 0 rotates clockwise = 1 rotates counter-clock-wise
2	ITURB	Viscous runs = 0 null = 1 turbulent boundary layer = 2 laminar boundary layer
3	LTHIN	= 0 full Navier-Stokes analysis = 1 thin layer Navier-Stokes analysis
4	ISWF	Output of computational grid = 0 no print out = 1 print grid (aerodynamic mesh) to fort.7 output file

If COUNT in Record 11 is TRUE then Record 13A and 13B must exist, if FALSE ignore.

Record 13A: Comment Line

Example:

XLOC	DIAA	BETAA	ITELA	ILEA	INOSE

Record 13B:

Format (6F10.4) and Example:

XLOC	DIAA	BETAA	ITELA	ILEA	INOSE
0.2776	1.0	38.0	46.0	14.0	12.0

Field Description:

Field	Name	Description
1	XLOC	Distance between the pitch change axes of front and aft rotors
2	DIAA	Aft propeller diameter
3	BETAA	Aft propeller blade angle
4	ITELA	Number of grid points between trailing edge of aft propeller blades and downstream boundary
5	ILEA	Number of grid points between leading edge of aft propeller and half distance between the propellers
6	INOSE	Number of grid points between the nose of the hub and upstream boundary

Record 14: Comment Line

Example:

FNRS

Record 15:

Format (F10.6) and Example:

FNRS
63.010

Field Description:

Field	Name	Description
1	FNRS	Number of data points to define center body or hub or number of radial stations on body

Notes:

1. Entered in as a real number, this value is converted into a integer number within the **PROP3D** code.

Record 16: Comment Line

Example:

```
    XR          RR
```

Record Set 17:

Format (2F10.6) and Example:

XR_1	RR_1
...	...
...	...
...	...
XR_{FNRS}	RR_{FNRS}

-2.00000	0.076145
-1.93358	0.076345
...	...
...	...
...	...
0.680607	0.145362
0.874329	0.110125
1.183826	0.089781
1.485585	0.090946
2.000000	0.090946

Field Description:

Field	Name	Description
1	XR	Axial stations
2	RR	Hub radius at axial station XR

Notes:

1. The number of record sets is equivalent to FNRS defined in Record 15.
2. Rows of dots are the continuation of data.
3. Only one comment statement (Record 16) is required for the set of data in Record 17.

If COUNT in Record 11 is TRUE then Records 18 through 31 are repeated for the aft propeller.

Record 18:

Format and Example:

INASTRAN
INASTRAN

Field Description:

Field	Name	Description
1	INASTRAN	Reserved for future use, enter characters

Notes:

1. Although not used must appear in the input deck.

Record 19:

Format and Example:

VALUE
0

Field Description:

Field	Name	Description
1	VALUE	Reserved for future use, enter 0

Notes:

1. Although not used must appear in the input deck.

Record 20: Comment Line

Example:

FNC	FROTAT	FB	FTPRP	FTWST	FCOB	FGR

Record 21:

Format (F10.6) and Example:

FNC	FROTAT	FB	FTPRP	FTWST	FCOB	FGR
16.000	1.01000	12.0000	1.01000	1.01000	18.0000	1.01000

Field Description:

Field	Name	Description
1	FNC	Number of stations at which blade is defined at input
2	FROTAT	Reserved, enter 1.01
3	FB	Number of blades on rotor
4	FTPRP	Reserved, enter 1.01
5	FTWST	Reserved, enter 1.01
6	FCOB	Number of root chord lengths the upstream and downstream boundaries are located at from the pitch change axis
7	FGR	Reserved, enter 1.01

Record 22: Comment Line

Example:

```
IDTYPE
```

Record 23:

Format (F10.6) and Example:

```
IDTYPE
```

```
3.01000
```

Field Description:

Field	Name	Description
1	IDTYPE	Type of analysis format used for input = 1 then Record Set 29 format is 3F10.6 = 2 then Record Set 29 format is 6F10.6 = 3 then Record Set 29 format is 10x,6F10.6 = 6 then Record Set 29 format is 10x,4F10.6

Notes:

1. Entered in as a real number, this value is converted into a integer number within the code.
2. IDTYPE determines the input format of Record Set 29.

Records 24 through 29 are repeated FNC (refer to Record 21) times

Record 24: Comment Line

Example:

YW(J)	AL	ALED	FAD	CHD	FSEC	THICK

Record 25:

Format (F10.6) and Example:

YW(J)	AL	ALED	FAD	CHD	FSEC	THICK
0.191708	16.488883	-0.05279	-0.002382	0.100996	1.010000	1.00000

Field Description:

Field	Name	Description
1	YW(J)	Span station or radius location
2	AL	Angle with respect to blade setting angle ($\beta_{3/4}$)
3	ALED	Leading edge alignment
4	FAD	Face alignment
5	CHD	Chord
6	FSEC	Symmetry parameter = 0 section airfoil cross-sections are identical = 1 section airfoil cross-sections are different
7	THICK	Reserved, enter 1.01

Record 26: Comment Line

Example:

ZSYM	FNU	FNL

Record 27:

Format and Example:

ZSYM	FNU	FNL
0.00000	65.010000	65.010000

Field Description:

Field	Name	Description
1	ZSYM	Symmetry parameter = 0 non symmetric airfoil cross-section = 1 symmetric airfoil cross-section
2	FNU	Number of points on the upper surface of airfoil
3	FNL	Number of points on the lower surface of airfoil

If IDTYPE in Record 23 = 1 then

Record 28: Comment Line

X	ZU	ZL

Record Set 29:

Format (3F10.6) and Example:

X_1	ZU_1	ZL_1
X_2	ZU_2	ZL_2
...	...	...
X_{FNL-1}	ZU_{FNL-1}	ZL_{FNL-1}
X_{FNL}	ZU_{FNL}	ZL_{FNL}

0.000000	0.000000	-0.000000
0.000250	0.000296	-0.000250
....	....	....
....	....	....
0.999750	0.000821	-0.001514
1.000000	0.000000	-0.000000

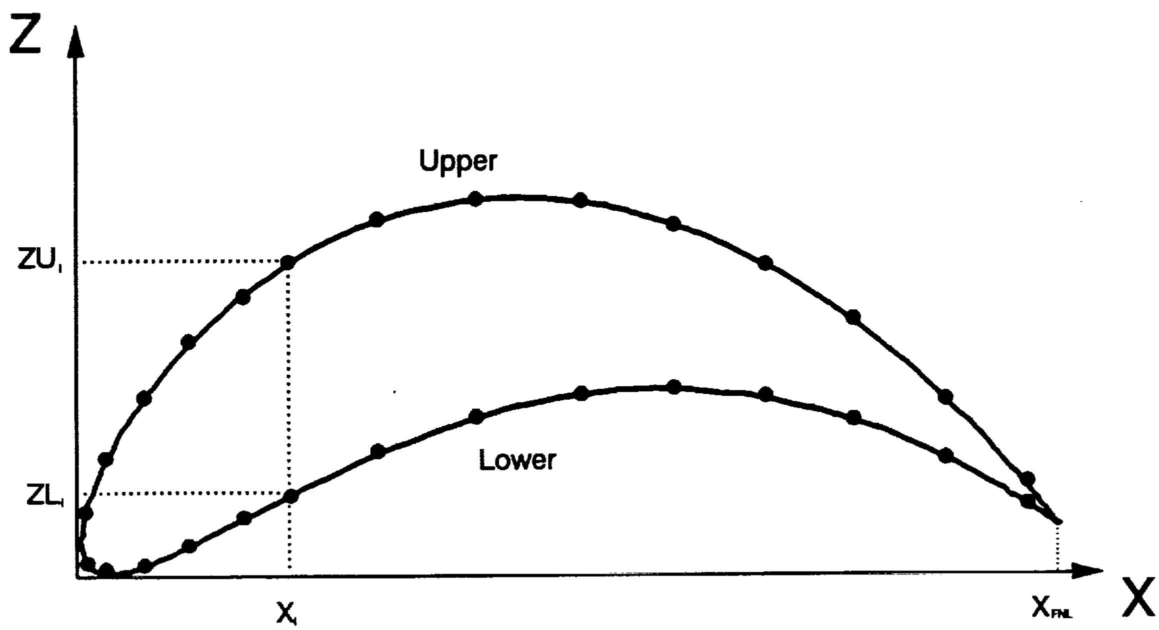
Field Description:

Field	Name	Description
1	X	Chordwise distance from leading edge
2	ZU	Upper surface coordinate
3	ZL	Lower surface coordinate

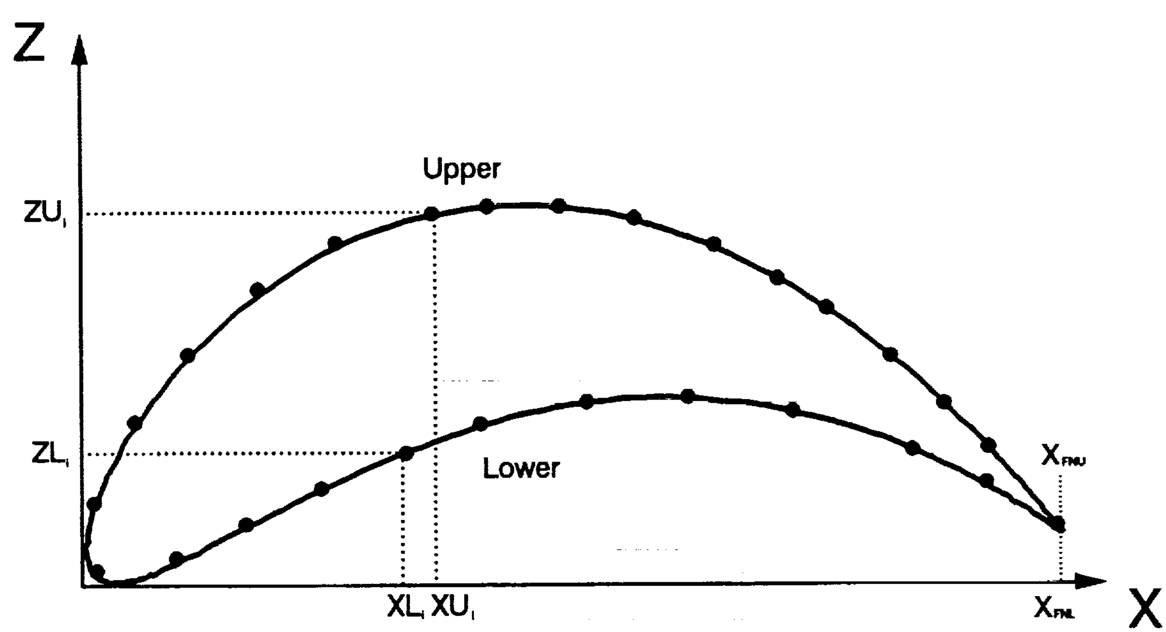
Note:

1. A total of FNL (defined by Record 26) records sets are required.
2. X, ZU, and ZL are normalized with respect to the local chord.
3. Rows of dots are the continuation of data.
4. Refer to Figure C-1(a) for an example of X, ZU, and ZL input.

Figure C-1: Typical Coordinate Definition for Record 29: (a) IDTYPE = 1,2,3; (b) IDTYPE = 6



(a) IDTYPE = 1, 2, 3



(b) IDTYPE = 6

If IDTYPE in Record 23 = 2 then

Record 28: Comment Line

X	ZU	ZL	X	ZU	ZL

Record Set 29:

Format (6F10.6) and Example:

X_1	ZU_1	ZL_1	X_2	ZU_2	ZL_2
...	...	...	...	...	...
...	...	...	...	...	...
X_{FNL-1}	ZU_{FNL-1}	ZL_{FNL-1}	X_{FNL}	ZU_{FNL}	ZL_{FNL}

0.000000	0.000000	-0.000000	0.000250	0.000296	-0.000250
....	....	....	....	....	....
....	....	....	....	....	....
....	....	....	....	....	....
....	....	....	....	....	....
0.999750	0.000821	-0.001514	1.000000	0.000000	-0.000000

Field Description:

Field	Name	Description
1	X_i	Chordwise distance from leading edge
2	ZU_i	Upper surface coordinate
3	ZL_i	Lower surface coordinate
4	X_{i+1}	Chordwise distance from leading edge
5	ZU_{i+1}	Upper surface coordinate
6	ZL_{i+1}	Lower surface coordinate

Note:

1. A total of FNL (defined by Record 26) records sets are required.
2. X, ZU, and ZL are normalized with respect to the local chord.
3. Rows of dots are the continuation of data.
4. Refer to Figure C-1(a) for an example of X, ZU, and ZL input.

If IDTYPE in Record 23 = 3 then

Record 28: Comment Line

	X	ZU	ZL	X	ZU	ZL

Record Set 29:

Format (10x,6F10.6) and Example:

	X_1	ZU_1	ZL_1	X_2	ZU_2	ZL_2
	...	...	...	...	...	...
	...	...	...	...	...	...
	X_{FNL-1}	ZU_{FNL-1}	ZL_{FNL-1}	X_{FNL}	ZU_{FNL}	ZL_{FNL}

	0.000000	0.000000	-0.000000	0.000250	0.000296	-0.000250
	....	....	....	....	....	....
	....	....	....	....	....	....
	....	....	....	....	....	....
	....	....	....	....	....	....
	0.999750	0.000821	-0.001514	1.000000	0.000000	-0.000000

Field Description:

Field	Name	Description
1		Blank
2	X_i	Chordwise distance from leading edge
3	ZU_i	Upper surface coordinate
4	ZL_i	Lower surface coordinate
5	X_{i+1}	Chordwise distance from leading edge
6	ZU_{i+1}	Upper surface coordinate
7	ZL_{i+1}	Lower surface coordinate

Note:

1. A total of FNL (defined by Record 26) records sets are required.
2. X, ZU, and ZL are normalized with respect to the local chord.
3. Rows of dots are the continuation of data.
4. Refer to Figure C-1(a) for an example of X, ZU, and ZL input.

If IDTYPE in Record 23 = 6 then

Record 28: Comment Line

	X	ZU	ZL	X	ZU	ZL

Record Set 29:

Format (10x,4F10.6) and Example:

XU_1	ZU_1	XL_1	ZL_1
...	...	...	...
...	...	...	...
XU_{FNL}	ZU_{FNL}	XL_{FNL}	ZL_{FNL}

0.000000	0.000000	0.000205	-0.000000
....	....	....	....
....	....	....	....
....	....	....	....
....	....	....	....
1.000000	0.000000	0.999914	-0.000000

Field Description:

Field	Name	Description
1		Blank
2	XU	Chordwise distance from leading edge on upper surface
3	ZU	Upper surface coordinate
4	XL	Chordwise distance from leading edge on lower surface
5	ZL	Lower surface coordinate

Note:

1. A total of FNL (defined by Record 26) records sets are required.
2. XU, ZU, XL, and ZL are normalized with respect to the local chord.
3. Rows of dots are the continuation of data.
4. Refer to Figure C-1(b) for an example of XU, ZU, XL, and ZL input.

Record 30: Comment Line

Example:

a0	p0

Record 31:

Format (F10.6) and Example:

a0	p0
13040.0	14.69

Field Description:

Field	Name	Description
1	a0	Sonic velocity of the fluid (inch/sec)
2	p0	Static pressure (psi)

Table C-3: Sample Input Deck for PROP3D Aerodynamic Code

Record1........2.........3.........4.........5.........6.........7.........8

1	F39 counter rotating blade, refer to Figure 2.1-3 for example grid						
2	ADV	GMU	ALFA	PSI	WW	DT	REYREF
3	1.1270	0.0	0.0	0.0	7.0	.0030	-1.0
4	IMAX	JMAX	KMAX	JTIP	ITEL	ILE	IGR
5	100	33	16	20	14	46	0
6	FSTP	FMINF	BETA34	DIA	DY	VIBFRE	
7	10.0	0.200	35.000	1.0	0.05	0.0	
8	IFLTR	NUMCYC	NSTDY	JMODE			
9	-1	0	0	2			
10	RESTART, QUASISTEADY, INFLOW, AEROELASTIC, COUNTER ROTATION, AND RESABD						
11	FALSE TRUEFALSEFALSE TRUEFALSE						
12	ICCW	ITURB	LTHIN	ISWF			
13	1	0	0	0			
13A	XLOC	DIAA	BETAA	ITELA	ILEA	INOSE	
13B	0.2776	1.0	38.0	46	14	12	
14	FNRS						
15	61.01000						
16	XR	RR					
17_{1}	-1.250000	0.010000					
17_{2}	-1.199300	0.077645					
17_{3}	-1.021340	0.083649					
17_{4}	-0.897309	0.106003					
17_{5}	-0.896821	0.113438					
17_{6}	-0.895328	0.117709					
17_{7}	-0.892009	0.123474					
17_{8}	-0.883714	0.132558					
17_{9}	-0.872650	0.140937					
17_{10}	-0.853291	0.151785					
17_{11}	-0.836694	0.159174					
17_{12}	-0.814565	0.167446					
17_{13}	-0.792444	0.174518					
17_{14}	-0.764782	0.182170					
17_{15}	-0.737129	0.188832					
17_{16}	-0.703935	0.195803					
17_{17}	-0.670749	0.201862					
17_{18}	-0.643088	0.206318					
17_{19}	-0.609902	0.211038					
17_{10}	-0.554587	0.217522					
17_{21}	-0.488208	0.223248					
17_{22}	-0.432893	0.226381					
17_{23}	-0.388635	0.227836					
17_{24}	-0.344385	0.228331					
17_{25}	-0.269881	0.227354					
17_{26}	-0.204577	0.224487					
17_{27}	-0.177961	0.222762					
17_{28}	-0.150106	0.220601					
17_{29}	-0.123799	0.218138					
17_{30}	-0.091998	0.214441					
17_{31}	-0.077374	0.212342					
17_{32}	-0.061899	0.209595					
17_{33}	-0.042401	0.205585					
17_{34}	-0.028071	0.203196					
17_{35}	-0.009006	0.201586					
17_{36}	0.010376	0.201230					
17_{37}	0.035213	0.201591					
17_{38}	0.055307	0.201742					
17_{39}	0.077405	0.201150					
17_{40}	0.103968	0.199660					

17_{41}	0.134941	0.197422					
17_{42}	0.189412	0.192301					
17_{43}	0.237129	0.187710					
17_{44}	0.251930	0.186659					
17_{45}	0.271367	0.186378					
17_{46}	0.291097	0.187452					
17_{47}	0.315888	0.189380					
17_{48}	0.335626	0.190845					
17_{49}	0.356169	0.191889					
17_{50}	0.371396	0.192186					
17_{51}	0.390740	0.191811					
17_{52}	0.413557	0.190811					
17_{53}	0.440932	0.189067					
17_{54}	0.470969	0.186228					
17_{55}	0.526818	0.178315					
17_{56}	0.597290	0.164107					
17_{57}	0.680607	0.145362					
17_{58}	0.874329	0.110125					
17_{59}	1.183826	0.089781					
17_{60}	1.485585	0.090946					
17_{61}	2.000000	0.090946					
18	INASTRAN						
19	0						
20	FNC	FROTAT	FB	FTPRP	FTWST	FCOB	FGR
21	16.010000	1.010000	12.010000	1.010000	1.010000	18.000000	1.010000
22	IDTYPE						
23	3.010000						

Records 24 through 29 are repeated with the appropriate input 16 (FNC) *times, only one set of Records 24 through 29 are shown below*

24	YW(J)	AL	ALED	FAD	CHD	FSEC	THICK
25	0.191708	16.488831	-0.052793	-0.002382	0.100996	1.010000	1.000000
26	ZSYM	FNU	FNL				
27	0.000000	65.010000	65.010000				
28		X	ZUPPER	ZLOWER	X	ZUPPER	ZLOWER
29_{1}		0.000000	0.000000	0.000000	0.000250	0.000296	-0.000250
29_{2}		0.000500	0.000564	-0.000477	0.001000	0.001017	-0.000859
29_{3}		0.002000	0.001437	-0.001211	0.003000	0.001748	-0.001436
29_{4}		0.005000	0.002175	-0.001692	0.007000	0.002465	-0.001813
29_{5}		0.010000	0.002789	-0.001888	0.015000	0.003250	-0.001931
29_{6}		0.020000	0.003679	-0.001962	0.025000	0.004080	-0.001994
29_{7}		0.030000	0.004457	-0.002025	0.035000	0.004817	-0.002048
29_{8}		0.040000	0.005160	-0.002068	0.045000	0.005491	-0.002090
29_{9}		0.050000	0.005807	-0.002115	0.060000	0.006383	-0.002177
29_{10}		0.070000	0.006907	-0.002250	0.080000	0.007391	-0.002334
29_{11}		0.090000	0.007832	-0.002433	0.100000	0.008233	-0.002542
29_{12}		0.120000	0.008907	-0.002799	0.140000	0.009399	-0.003134
29_{13}		0.160000	0.009760	-0.003508	0.180000	0.010026	-0.003917
29_{14}		0.200000	0.010120	-0.004405	0.250000	0.009946	-0.005752
29_{15}		0.300000	0.009167	-0.007340	0.350000	0.007969	-0.009011
29_{16}		0.400000	0.006319	-0.010838	0.450000	0.004296	-0.012797
29_{17}		0.500000	0.001952	-0.014845	0.550000	-0.000255	-0.016512
29_{18}		0.600000	-0.002362	-0.017825	0.650000	-0.004132	-0.018576
29_{19}		0.700000	-0.005417	-0.018656	0.750000	-0.005999	-0.017917
29_{20}		0.800000	-0.005814	-0.016374	0.820000	-0.005500	-0.015524
29_{21}		0.840000	-0.005069	-0.014598	0.860000	-0.004482	-0.013556
29_{22}		0.880000	-0.003754	-0.012406	0.900000	-0.002896	-0.011150
29_{23}		0.910000	-0.002420	-0.010480	0.920000	-0.001913	-0.009782
29_{24}		0.930000	-0.001379	-0.009060	0.940000	-0.000815	-0.008310
29_{25}		0.950000	-0.000215	-0.007529	0.955000	0.000099	-0.007124
29_{26}		0.960000	0.000423	-0.006711	0.965000	0.000753	-0.006288
29_{27}		0.970000	0.001089	-0.005860	0.975000	0.001428	-0.005434

29_{28}		0.980000	0.001772	-0.005002	0.985000	0.002127	-0.004565
29_{29}		0.990000	0.002494	-0.004127	0.993000	0.002721	-0.003853
29_{30}		0.995000	0.002866	-0.003676	0.997000	0.002861	-0.003407
29_{31}		0.998000	0.002719	-0.003150	0.999000	0.001988	-0.002497
29_{32}		0.999500	0.001458	-0.001873	0.999750	0.000822	-0.001512
29_{33}		1.000000	0.000000	0.000000			
33	a0	p0					
31	13040.0	14.69					

Records 18 through 31 are reproduced for the aft propeller

18_a	INASTRAN						
19_a	0						
20_a	FNC	FROTAT	FB	FTPRP	FTWST	FCOB	FGR
21_a	16.000000	1.010000	12.000000	1.010000	1.010000	18.010000	1.010000
22_a	IDTYP						
23_a	6.05						

Records 24 through 29 are repeated with the appropriate input 20 (FNC) *times, only one set of Records 24 through 29 are shown below for the aft propeller*

24_a	YW(J)	AL	ALED	FAD	CHD	FSEC	THICK
25_a	0.18363	5.91201	-0.04719	0.00483	0.10670	1.0100	1.00
26_a	ZSYM	FNU	FNL				
27_a	0.	57.01	57.01				
28_a		XUP	ZUP	XLOWER	ZLOWER		
29_{1a}		0.000000	0.000000	0.000000	0.000000		
29_{2a}		0.000059	0.000194	0.000027	-0.000206		
29_{3a}		0.000218	0.000560	0.000127	-0.000604		
29_{4a}		0.000707	0.001187	0.000523	-0.001317		
29_{5a}		0.002059	0.002050	0.001752	-0.002413		
29_{6a}		0.004420	0.002928	0.004001	-0.003688		
29_{7a}		0.008840	0.003952	0.008315	-0.005447		
29_{8a}		0.014820	0.004895	0.014221	-0.007378		
29_{9a}		0.022818	0.005980	0.022154	-0.009770		
29_{10a}		0.032820	0.007315	0.032090	-0.012694		
29_{11a}		0.045822	0.009049	0.045027	-0.016428		
29_{12a}		0.060824	0.011047	0.059981	-0.020642		
29_{13a}		0.076826	0.013188	0.075955	-0.025053		
29_{14a}		0.094822	0.015632	0.093941	-0.029962		
29_{15a}		0.114811	0.018407	0.113931	-0.035396		
29_{16a}		0.136796	0.021474	0.135925	-0.041356		
29_{17a}		0.159791	0.024606	0.158911	-0.047614		
29_{18a}		0.183793	0.027819	0.182844	-0.054332		
29_{19a}		0.208826	0.030920	0.207708	-0.061561		
29_{20a}		0.234900	0.033814	0.233501	-0.069303		
29_{21a}		0.262010	0.036503	0.260186	-0.077670		
29_{22a}		0.290148	0.039036	0.287717	-0.086783		
29_{23a}		0.319305	0.041516	0.316119	-0.096564		
29_{24a}		0.349496	0.043796	0.345639	-0.106297		
29_{25a}		0.379732	0.044656	0.375878	-0.113224		
29_{26a}		0.409888	0.042124	0.406716	-0.117024		
29_{27a}		0.439835	0.037731	0.437736	-0.118909		
29_{28a}		0.469640	0.032453	0.468790	-0.118766		
29_{29a}		0.499183	0.025859	0.499806	-0.116827		
29_{30a}		0.528884	0.020030	0.530780	-0.114344		
29_{31a}		0.558822	0.015569	0.561711	-0.111387		
29_{32a}		0.588901	0.012198	0.592566	-0.107717		
29_{33a}		0.619090	0.010009	0.623353	-0.103513		
29_{34a}		0.649331	0.008683	0.654055	-0.098734		
29_{35a}		0.679584	0.007663	0.684606	-0.093071		
29_{36a}		0.708829	0.006699	0.713950	-0.086661		
29_{37a}		0.737065	0.005752	0.742116	-0.079751		
29_{38a}		0.764291	0.004792	0.769126	-0.072508		
29_{39a}		0.790508	0.003839	0.795011	-0.065080		

29_{40a}		0.815716	0.002914	0.819804	-0.057609
29_{41a}		0.839918	0.002091	0.843536	-0.050211
29_{42a}		0.863121	0.001688	0.866324	-0.043268
29_{43a}		0.884310	0.001706	0.887193	-0.037138
29_{44a}		0.903479	0.001966	0.906113	-0.031720
29_{45a}		0.920628	0.002319	0.923051	-0.026910
29_{46a}		0.935759	0.002697	0.938000	-0.022678
29_{47a}		0.948871	0.003064	0.950954	-0.019006
29_{48a}		0.959965	0.003393	0.961913	-0.015890
29_{49a}		0.969445	0.003691	0.971276	-0.013218
29_{50a}		0.977411	0.003956	0.979143	-0.010970
29_{51a}		0.983965	0.004194	0.985617	-0.009124
29_{52a}		0.989207	0.004394	0.990795	-0.007646
29_{53a}		0.993240	0.004553	0.994779	-0.006510
29_{54a}		0.996240	0.004350	0.997664	-0.005403
29_{55a}		0.998390	0.003075	0.999393	-0.003507
29_{56a}		0.999557	0.001442	1.000027	-0.001548
29_{57a}		1.000000	0.000000	1.000000	0.000000
37_a	a0	p0			
31_a	13040.0	14.69			

Appendix D: Parameter File Input Preparation

The parameter file is an assortment of data used within the different codes of the SAB process. The following codes read the parameter file input deck:

(1) converg.o - uses the coordinate definition, radius ratio, and tolerance
(2) face1.o - uses the blade row diameter
(3) face2.o - uses the blade row diameter and P_{dim}

Input into this file is described in this appendix along with the format, example, and field description. A total of five different items are defined in 4 data records. Comments cards can be included in the parameter file as shown in Table D-1 which represents an example of the file.

Table D-1: Example Input File

```
PXPY                          coordinate definition
0.95,0.01                     radius ratio (%), tolerance (degree)
26.004                        full scale blade row diameter (inches)
20.401                        dimensional factor (psi)
```

Record 1: Coordinate Definition

Description:

Defines the structural analysis blade geometry in its global coordinate system. The user must supply two coordinate directions, the spanwise and chordwise directions of the structural model, *e.g.* see Figure D-1. The third axis is then defined internally to form a right-hand coordinate system.

Format and Example:

spanchord
PZPX

Field Description:

Field	Name	Type	Description
1	span	character*2	Spanwise axis
2	chord	character*2	Chordwise axis

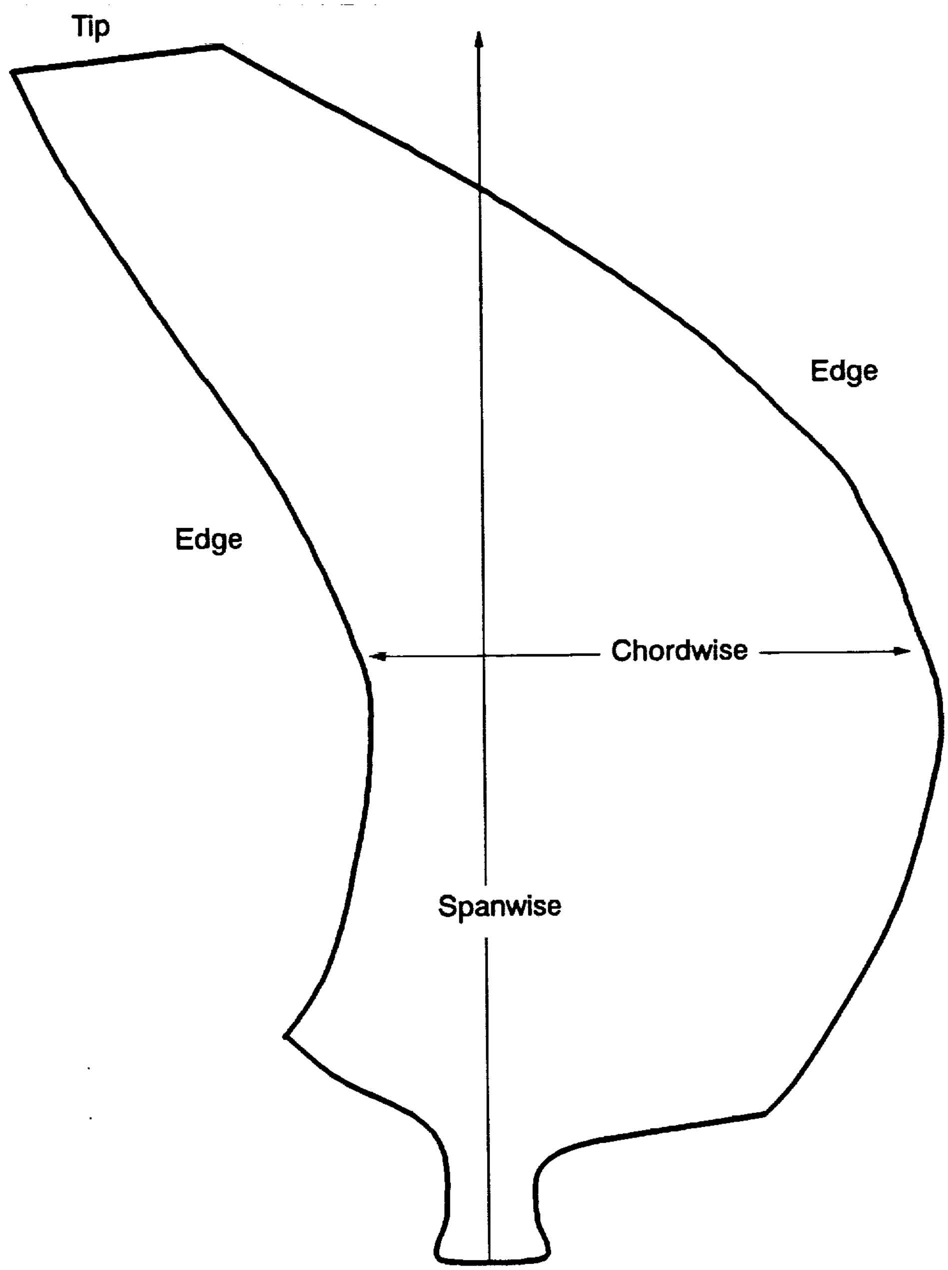
Note:

1. Possible entries for span and chord are:

   PX - positive X
   PY - positive Y
   PZ - positive Z
   MX - negative X
   MY - negative Y
   MZ - negative Z

   where the spanwise positive direction is considered hub to tip and the chordwise positive direction is considered from leading edge to the trailing edge.
2. In the above example, PZ signifies that the blade spanwise direction (hub to tip) extends in the positive Z direction. Likewise, the chordwise direction extends in along the positive X axis.

Figure D-1: Coordinate System Definition for a Typical Blade



Record 2: Radius Ratio and Tolerance

Description:

The user must define the radial location (*i.e.*, along the span) where the convergence check will be made, preferably near the location of maximum angular chord displacement. Also, the tolerance criteria used for determining if convergence is met for the iterative process is defined.

Format and Example:

RR,TOL

0.95,0.01

Field Description:

Field	Name	Type	Description
1	RR	real	Radius ratio (%)
2	TOL	real	Tolerance (degree)

Note:

1. Valid range for the radius ratio is:

$$0.0 \leq RR\ (\%) \leq 1.00$$

where 0.0 is at the hub and 1.00 is the tip.

2. Suggested values to use for the tolerance:

$$0.0001 \leq TOL \leq 0.10$$

3. For the above example, maximum displacement occurs close to the 95% radial location, therefore, 0.95 was used to define the radius ratio (RR). After a comma, the tolerance is defined as 0.001 and convergence is met under the following criteria:

$$(i+1)^{th} - (i)^{th} \leq TOL$$

Record 3: Diameter of the Blade Row

Description:

The full scale diameter of the blade row defined in inches.

Format and Example:

```
DIAM
26.004
```

Field Description:

Field	Name	Type	Description
1	DIAM	real	Blade row diameter (inches)

Record 4 Dimensional Factor

Description:

This factor is used as a dimensional factor in the manipulation of the aerodynamic analysis results, specifically the pressures.

Format and Example:

P_{dim}
20.401

Field Description:

Field	Name	Type	Description
1	P_{dim}	real	Dimensional factor (psi)

Note:

1. The following equation defines the dimensional factor:

$$P_{dim} = \rho * a^2$$

where ρ is the free stream density and a is the speed of sound.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
March 1994

3. REPORT TYPE AND DATES COVERED
Final Contractor Report

4. TITLE AND SUBTITLE
Structural/Aerodynamic Blade Analyzer (SAB)
User's Guide-Version 1.0

5. FUNDING NUMBERS
WU–505–63–5B
WU–535–03–10
WU–505–62–10
C–NAS3–25266

6. AUTHOR(S)
M.R. Morel

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Sverdrup Technology, Inc.
Lewis Research Center Group
2001 Aerospace Parkway
Brook Park, Ohio 44142

8. PERFORMING ORGANIZATION REPORT NUMBER
E–8490

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
National Aeronautics and Space Administration
Lewis Research Center
Cleveland, Ohio 44135–3191

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
NASA CR–194460

11. SUPPLEMENTARY NOTES
M.R. Morel presently at NYMA, Inc., Engineering Services Division, 2001 Aerospace Parkway, Brook Park, Ohio 44142. Project Manager, Oral Mehmed, Structures Division, organization code 5230, NASA Lewis Research Center, (216) 433–6036.

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Unclassified - Unlimited
Subject Category 07

12b. DISTRIBUTION CODE

13. ABSTRACT ***(Maximum 200 words)***

This report is the User's Guide for the Structural/Aerodynamic Blade (SAB) analyzer. The system provides an automated tool for the static-deflection analysis of turbomachinery blades with aerodynamic and rotational loads. A structural code calculates a deflected blade shape using aerodynamic loads input. An aerodynamic solver computes aerodynamic loads using deflected blade shape input. The two programs are iterated automatically until deflections converge. Currently, SAB version 1.0 is interfaced with MSC/NASTRAN (to perform the structural analysis) and PROP3D (to perform the aerodynamic analysis). This document is meant to serve as a guide for the operation of the SAB system with specific emphasis on its use at NASA Lewis Research Center (LeRC). This guide consists of six chapters: (1) an introduction which give a summary of SAB; (2) SAB's methodology, component files, links, and interfaces; (3) input/output file structure; (4) setup and execution of the SAB files on the Cray computers; (5) hints and tips to advise the user; and (6) an example problem demonstrating the SAB process. In addition, four appendices are presented to define the different computer programs used within the SAB analyzer and describe the required input decks.

14. SUBJECT TERMS
Structural analyzer; Blade defection; Aerodynamics; Computational mechanics

15. NUMBER OF PAGES
115

16. PRICE CODE
A06

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	