September 1993
NASA MSFC Contract No. NAS8–37821
Technical Monitor: James Cannon

FINAL REPORT

A FINITE–ELEMENT–BASED PERTURBATION MODEL FOR THE ROTORDYNAMIC ANALYSIS OF SHROUDED PUMP IMPELLERS :

PART 2: USER'S GUIDE

Erian A. Baskharone
Principal Investigator
Department of Mechanical Engineering
Texas A&M University

Turbomachinery Laboratory

(NASA-CR-193917) A FINITE-ELEMENT-BASED PERTURBATION MODEL FOR THE ROTORDYNAMIC ANALYSIS OF SHROUDED PUMP IMPELLERS: PART 2: USER'S GUIDE (Texas A&M Univ.) 75 p

N94-24999

Unclas

G3/37 0206642

September 1993

NASA MSFC Contract No. NAS8–37821
Technical Monitor: James Cannon

FINAL REPORT

A FINITE–ELEMENT–BASED PERTURBATION MODEL FOR THE ROTORDYNAMIC ANALYSIS OF SHROUDED PUMP IMPELLERS :

PART 2: USER'S GUIDE

Erian A. Baskharone
Principal Investigator
Department of Mechanical Engineering
Texas A&M University

TABLE OF CONTENTS

OVERVIEW

This report describes the computational steps involved in executing a finite-element-based perturbation model (Baskharone and Hensel, 1991) for computing the rotordynamic coefficients of a shrouded pump impeller or a simple seal (Fig. 1). These arise from the fluid/rotor interaction in the clearance gap of Fig. 1. In addition to the sample cases in this Figure, the computational procedure also applies to a separate category of problems referred to as the "seal-like" category. The problem, in this case, concerns a shrouded impeller (such as that in Fig. 1), with the exception that the secondary (or leakage) passage is totally isolated from the primary-flow passage. The difference between this and the pump problem in Fig. 1 is that the former is analytically of the simple "seal-like" configuration, with two (inlet and exit) flow-permeable stations, while the latter constitutes a double-entry/double-discharge flow problem. In all cases, the problem is that of a rotor clearance gap of Fig. 1. The problem here is that of a rotor excitation in the form of a cylindrical whirl around the housing centerline as shown in Fig. 2a for a smooth annular seal. In its centered operation mode, the rotor is assumed to give rise to an axisymmetric flow field in the clearance gap. As a result, problems involving longitudinal or helical grooves, in the rotor or housing surfaces, go beyond the code capabilities.

Discarding, for the moment, the pre- and post-processing phases, the bulk of the computational procedure consists of two main steps. The first is aimed at producing the axisymmetric "zeroth-order" flow solution in the given flow domain (Fig. 1). Detailed description of this problem, including the flow-governing equations, turbulence closure, boundary conditions and the finite-element formulation, was covered by Baskharone and Hensel (1989 & 1991b) for the problem categories in Fig. 1 The second main step is where the perturbation model is implemented, with the input being the centered-rotor "zeroth-order" flow solution and a prescribed whirl frequency ratio (whirl frequency divided by the impeller speed). As shown in Fig. 2b, the computational domain, in the latter case, is treated as three dimensional, with the number of computational planes in the circumferential direction being specified a priori. The reader is reminded (Baskharone and Hensel, 1991a) that the deformations in the finite elements are all infinitesimally small (despite the exaggeration in Fig. 2), because the rotor eccentricity itself is a virtual displacement. This explains why we have generically termed the perturbation model the "virtually" deformable finite-element category. The primary outcome of implementing the perturbation model is the tangential and radial components, $F_\theta{}^*$ and $F_r{}^*$ of the fluid-exerted force on the rotor surface due to the whirling motion. Repetitive execution of the perturbation model subprogram over a sufficient range of whirl frequency ratios, and subsequent interpolation of these fluid forces, using the least-square method, finally enable the user to compute the impeller rotordynamic coefficients (Baskharone and Hensel, 1991a) of the fluid/rotor interaction. These are the direct and cross-coupled stiffness, damping and inertia effects of the fluid/rotor interaction.

The computer code described in this report is composed of five subprograms. These are, respectively, responsible for creating the centered-impeller finite element model of the type

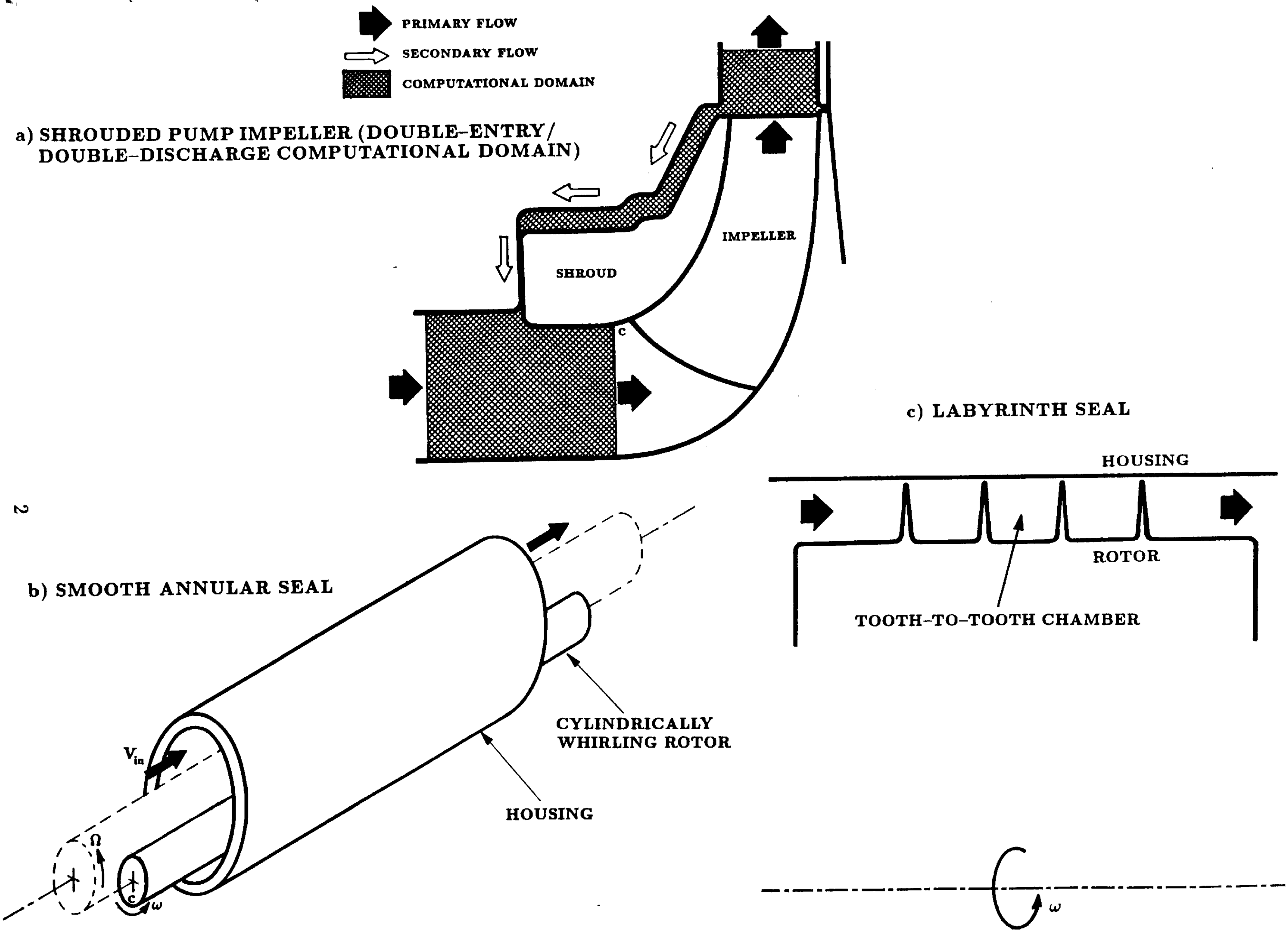
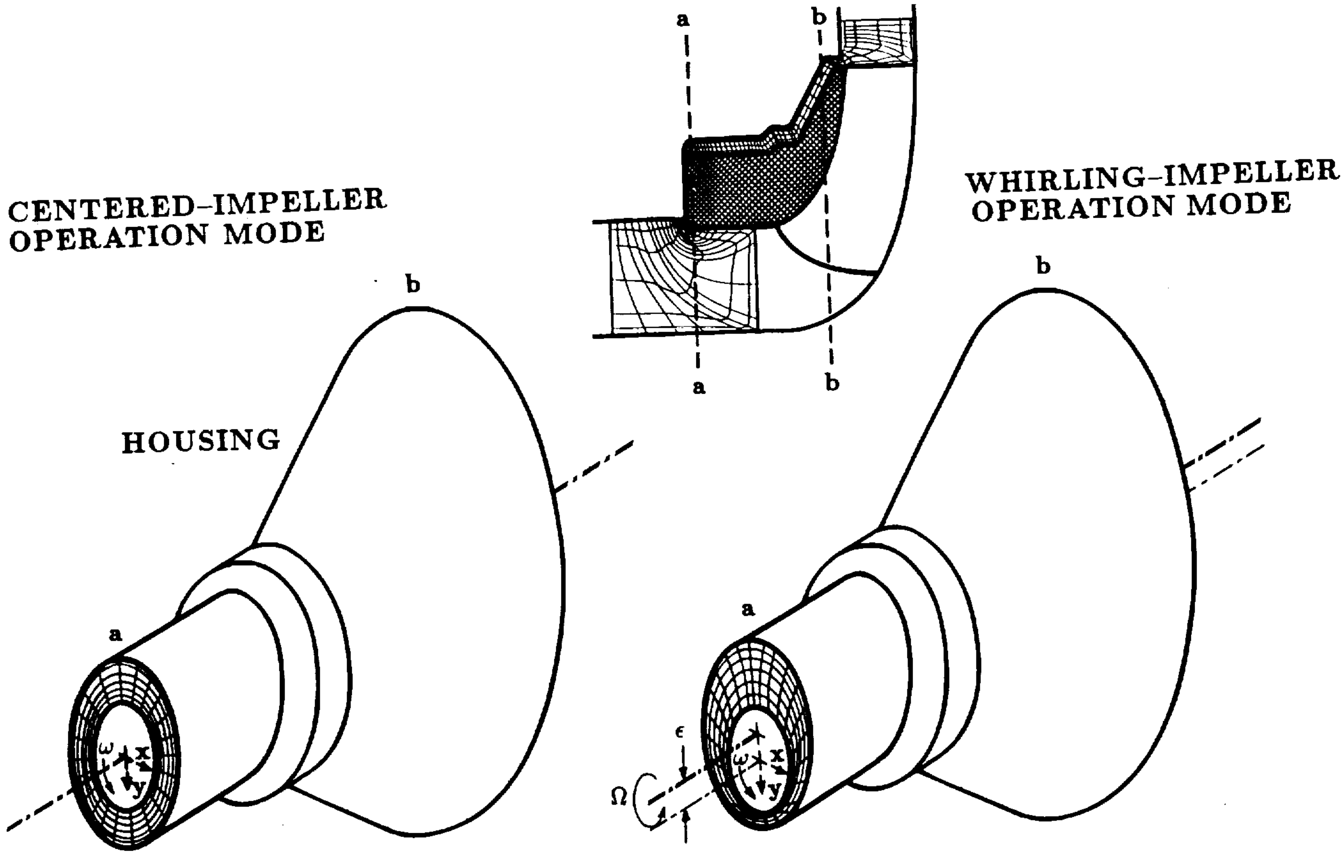
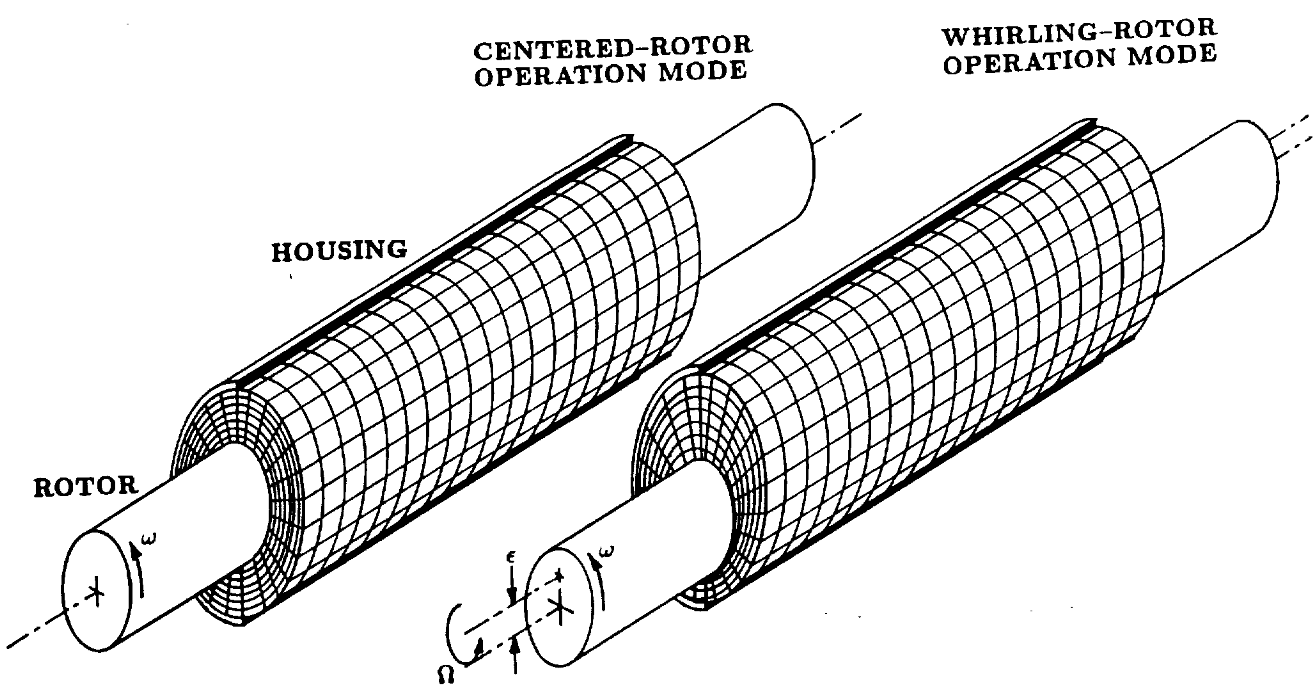


Fig. 1 Problem categories under consideration



a) SHROUDED-IMPELLER PROBLEM



b) ANNULAR SEAL PROBLEM

Fig. 2 Destruction of the flow domain axisymmetry as a result of the rotor whirl

shown in Fig. 3 (subprogram "PREPRO"), solving the centered-impeller "unperturbed" flow field (subprogram AXISYMM), utilizing two-dimensional interpolation to reduce the number of finite-element grid points, if desired, before executing the perturbation model (subprogram INTERPOLATE), implementing the perturbation analysis (subroutine PERTURB) and producing the rotordynamic coefficients (subprogram LEASTSQ). Progressing from one of these subprograms to the next is highly automated in the sense that the output of the former is precisely the input to the latter, unless the user wishes to alter some of previously- specified or built-in parameters. The sequence of subprograms (above) contains no graphical means of displaying the intermediate output files. Although we have created different post-processors for this purpose, these post-processors are based on specific commercial software packages (such as the DI-3000), to which the user is not assumed to have access. Instead, a separate section is included in the report,where a step-by-step procedure is provided with the intention of aiding the user in creating his/her own post-processor using whichever graphics software package that is capable of producing the required plot(s).

The main body of this report is substructured in such a way that each of the five subprograms (above) is separately covered, with the emphasis being on the contents of the input and output files. A sample case, involving a face-seal pump configuration which was fabricated and rotordynamically tested by Sulzer Bros. (Bolleter et al., 1989), is used as an example throughout the entire report. The input, output and some intermediate files, for this sample pump configuration, are provided in the APPENDIX for illustration.

The entire computational procedure can be executed on a workstation or a mainframe. For instance, running the Sulzer Bros.' test case was done on a MIPS workstation, which has a RAM capacity of 24 MB, and a disk space of approximately 350 MB. A grid dependency study, in which the resolution of the finite element model was progressively elevated, was executed on the IBM-3090 mainframe. Fortunately, the outcome of the study came to suggest that the optimum size of the finite-element model falls within the workstation capability range.

A pivotal section of this report addresses what may be a highly desirable option to the user in making use of the current rotordynamic analysis. This is an execution mode of the computational procedure whereby the user can override the centered-impeller flow field computations using, instead, an existing axisymmetric flow solution of the problem at hand, or an equivalent set of experimental data. Under such circumstances, the user should be prepared to write a relatively small "interface" code, with the purpose of converting the existing grid into a finite-element-like model. since the geometrical details and numbering pattern of nodal points are often vastly different from one grid to another, the report section (above) is intentionally written to be instructional in nature, with the emphasis being on the logic with which the conversion process can be achieved.

Reference to the computational procedure in this report should be by the name **FEPERT**. This name is solely for identification purposes, and is totally irrelevant as far as the source and executable versions of the computer code are concerned. As explained later in this

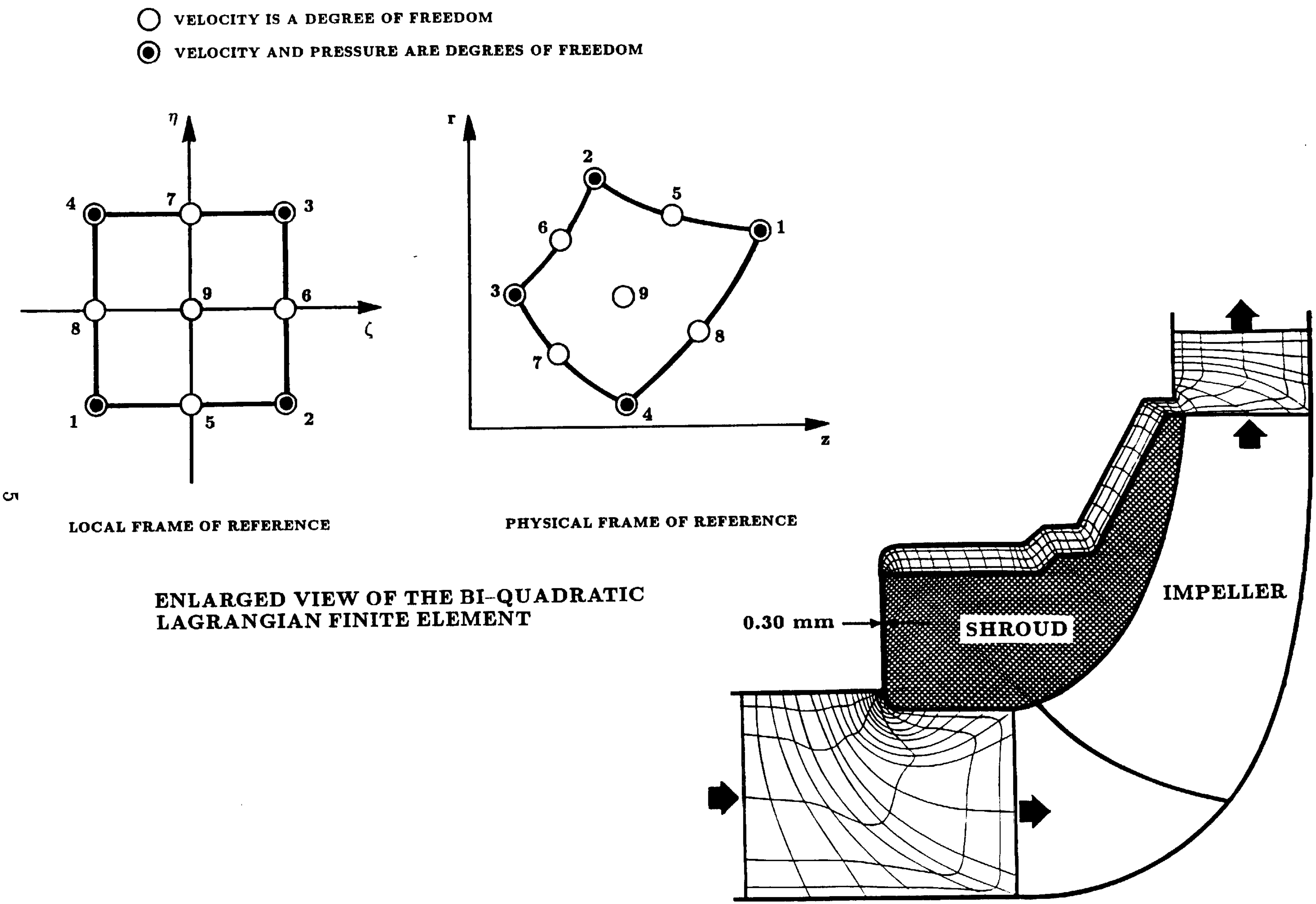


Fig. 3 Finite–element model for a shrouded–impeller pump with a face seal (SulzerBros., Switzerland)

report, the computational procedure here consists of a sequence of subprograms, each with its own name and a specific set of input and output files. These subprograms should be individually executed in the same order they are presented in this report, using the name of each subprogram exactly as indicated in the report.

SUBPROGRAM "PREPRO"

DESCRIPTION AND INPUT DATA PREPARATION:

This pre-processor accepts two input files (DIGIT.DAT & DIGIN.DAT) and creates an output file (DIGIT.OUT), which contains the finite-element discretization model (nodal coordinates and system topology) as well as data concerning the boundary conditions. This output file goes as input to the axisymmetric flow analysis code (subprogram AXISYMM), which produces the centered-impeller flow field. The user will find that the emphasis, in this and all other sections of the report, is placed on the shrouded impeller problem in Fig. 1, since this is the most complicated among the three sample problems in the Figure. The user should also know that this is one of two subprograms, in the entire computational procedure, which utilize subroutines from the Mathematics/Science library (IMSL), version 9.2. It is, therefore, assumed that the user has access to this particular software library.

In the following, the process of creating the two input files of this pre-processor is discussed. The reader should know that these are practically the only input files to the entire computational procedure in this report. One may, in this case, invest some time and effort to ensure the accuracy of these files, particularly the geometry-related input variables. The reader is reminded that full listings of these input files, along with partial listings of several output files concerning this and other computational steps, are all contained in the APPENDIX section of this report.

An accurate and sufficiently large (or enlarged) drawing of the flow domain (to any arbitrary scale) is required (Fig. 1). On this drawing, the user is expected to create a set of grid lines, which are nearly in the cross-flow direction, except near the impeller inlet and exit stations of a pump stage where the computational domain includes two primary-flow segments (making the domain one of the double-entry/double discharge type) as shown in Fig. 4. These "mildly-curved" lines are to be created using any suitable means, including free hand, but should never intersect one another. As shown in Fig. 4, each of these user-created lines starts at the housing and ends at the hub, a flow-permeable station (namely the impeller inlet and discharge stations), or on the shroud. Fig. 4 also shows that these lines should be closely spaced in regions of the computational domain where high gradients of the flow properties are anticipated (e.g. the seal upstream and downstream regions). In doing so, it is also important to avoid abrupt increments between any two successive grid lines. Note that extreme violations of these guidelines may amount to accuracy degradation of the computed centered-impeller flow field.

The next step is to create the file "DIGIT.DAT", which is one of the two pre-processor input files. In this file, each of the already created grid lines (above) is defined (in the z-r plane) by four "key" points (Fig. 4) starting, each time, at the housing line. In selecting these four points, the user should realize that each grid line will later be re-created (by the

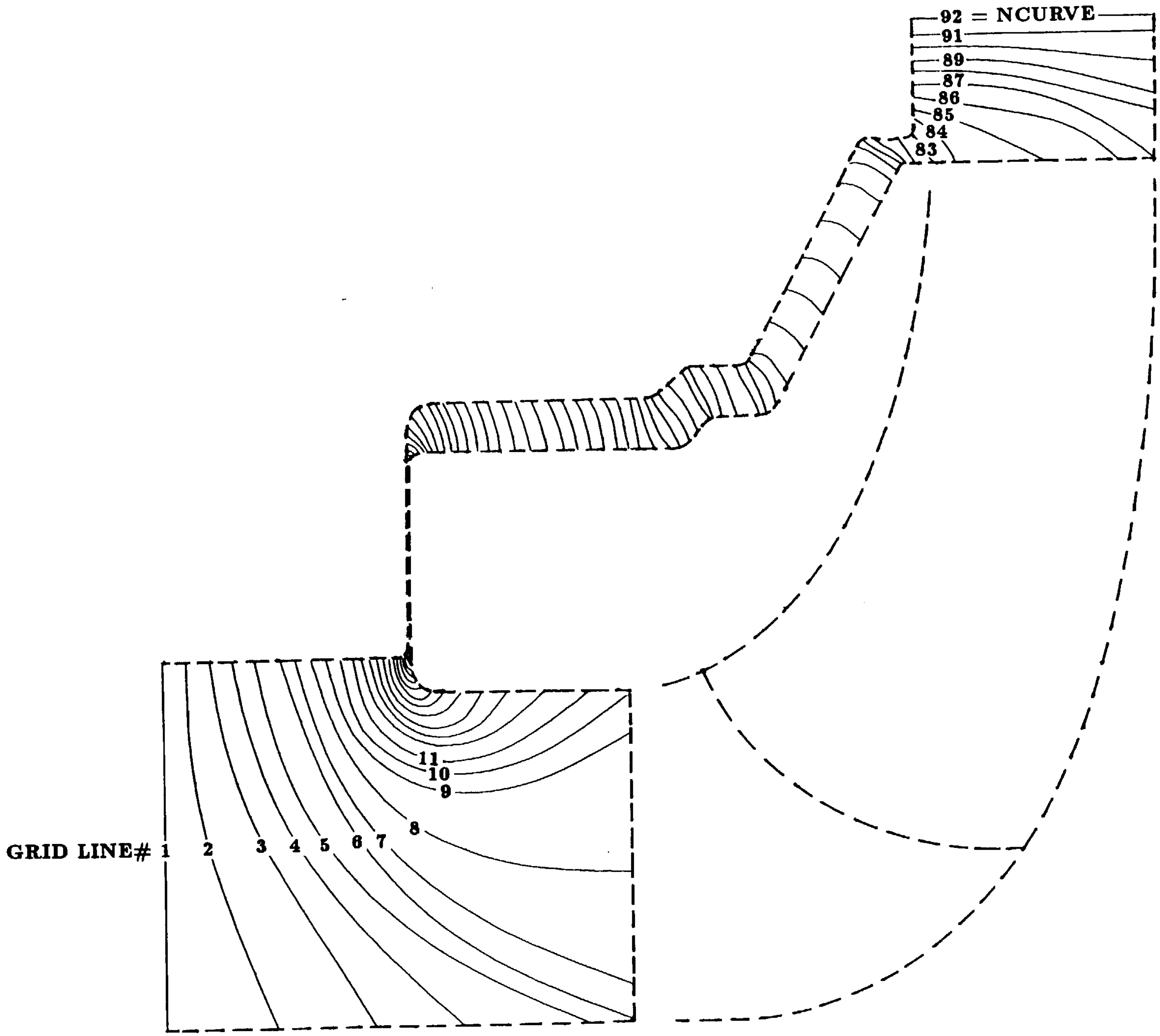


Fig. 4 Example of the user-created grid lines in the pre-processing stage

pre-processor "PREPRO") as a spline-fit, using the user-supplied four points. The re-created grid line may, therefore, look slightly different from that created by the user. The difference here will naturally depend on the user's choice of these four points. Note that each of the user-created grid lines (Fig. 4) can be defined by more than four points in the file "DIGIT.DAT". However, such choice is discouraged, as it may result in a fictitiously "wiggly" grid line whenever the line is re-created in the pre-processor "PREPRO". Also, before creating these (often free-hand) grid lines, the user should approximate sharp corners of the computational domain boundary, by slightly rounded corners. The reason is that sharp corners may cause problems in the domain-boundary interpolation process, which is one of the computational steps in "PREPRO".

The second, and only other, input file to "PREPRO" is the file "DIGIN.DAT". The variables in this relatively shorter file concern the actual dimensions of the computational domain (Fig. 4), the number of user-created grid lines [in the file "DIGIT.DAT" (above)], the desired near-wall refinement of the finite-element grid and the pump operating conditions.

CONTENTS OF THE INPUT FILES:

1. INPUT FILE "DIGIT.DAT":

This file consists of several similar sets, each defining one of the user-created grid lines (Fig. 2), as follows:

GROUP# 1 - - one line (Format I4)

NPTS is the number of points defining the user-created grid line (recommended value is 4)

GROUP# 2 - - "NPTS" lines (Free Format)

Z & R are the z- and r- coordinates of one of the NPTS points defining the grid line, where Z is measured from any arbitrary fixed location and R is measured from the axis of rotation. Note that the scale of the computational domain at this point is irrelevant, meaning that the coordinates may all be multiplied by an arbitrary (known or unknown) factor.

2. INPUT FILE "DIGIN.DAT":

Preparation of this file requires prior knowledge of the pump geometry and operating conditions. The drawing in which the user has created the grid lines shown in Fig. 4 (as an example) is also needed. The contents of this file are as follows:

LINE# 1: (Format 2I5)

IPROB is a trigger which alerts the pre-processor to whether the problem is that of the double-entry/double discharge type (meaning shrouded-impeller problem), or of the single-entry/single discharge type (which is the case of a seal or a seal-like problem), where:

IPROB = 1 for the double-entry/double discharge impeller problem
= 0 for the seal (or seal-like) problem

An example of the seal-like problem category is that of the isolated leakage passage in a shrouded-impeller pump (shaded area in Fig. 5).

JTYPE is a variable which should always be set equal to 1

LINE# 2: (Format 3I5)

NCURVE is the number of user-created grid lines (Fig. 4)

NORTH is the desired number of corner nodes (in the finite-element grid) per each of the "NCURVE" grid lines (Fig. 3). This input variable should preferably be an odd number. Note that a total of (NORTH-1) rows of finite elements will be created by "PREPRO" to occupy the entire computational domain, as shown in Fig. 3. Each of these rows will, by reference to Fig. 4, consist of (NCURVE-1) elements.

MID is the number of the user-created grid line, counted from the pump-stage inlet station (Fig. 4), where the leakage flow rate (in m^3/s) is desired. This grid line should obviously be within the shroud-to-housing leakage passage. The desired leakage flow rate will be part of the output of subprogram "AXISYMM" (to be discussed next), and is obtained by integrating the through-flow velocity component across the chosen grid line and around the circumference.

LINE# 3: (Format 2F10.0)

AFINE is a factor which determines the near-wall resolution level of the finite-element model (Fig. 6). Any value of this factor above 1.0 will cause the finite-element nodes to be increasingly close to one another as a solid wall segment is approached. Typical values of this "refinement" factor range between 1.5 and 3.5 (note the drastic change in the finite-element model (Fig. 6) as a result of varying "AFINE", while maintaining "NORTH" constant).

PERCNT is the ratio between the impeller-inlet boundary layer thickness and the impeller-inlet annulus height (usually between 0.05 and 0.2).

LINE# 4: (Format 3F10.0)

RMIN is the minimum value of the radial coordinate (measured from the axis of rotation) of any point on the boundary of the computational domain in Fig. 3. Should the inlet segment of the hub be a constant-radius line (Fig. 3), then "RMIN" is simply equal to that radius.

RMAX is the maximum radial coordinate of any point on the boundary of the computational domain. This variable, by reference to Fig. 3 is equal to the pump-stage (or far-downstream) exit radius.

ARPM is the impeller operating speed in rpm.

LINE# 5: (Format 2F10.0)

UNDRLX is the underrelaxation factor that is used in progressing from one iteration to the next in the iterative solution of the non-linear flow-governing equation, which takes place in subprogram "AXISYMM".

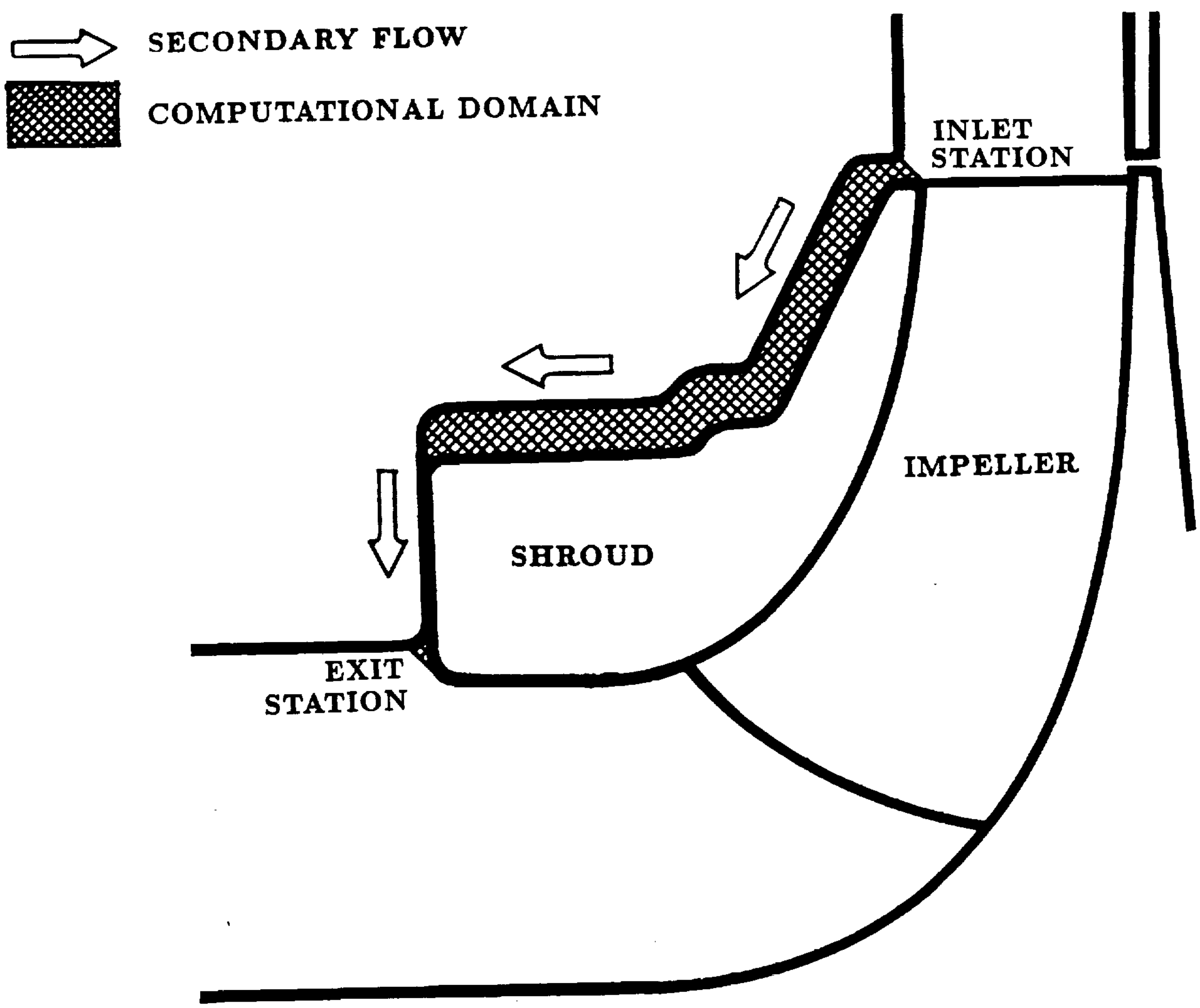
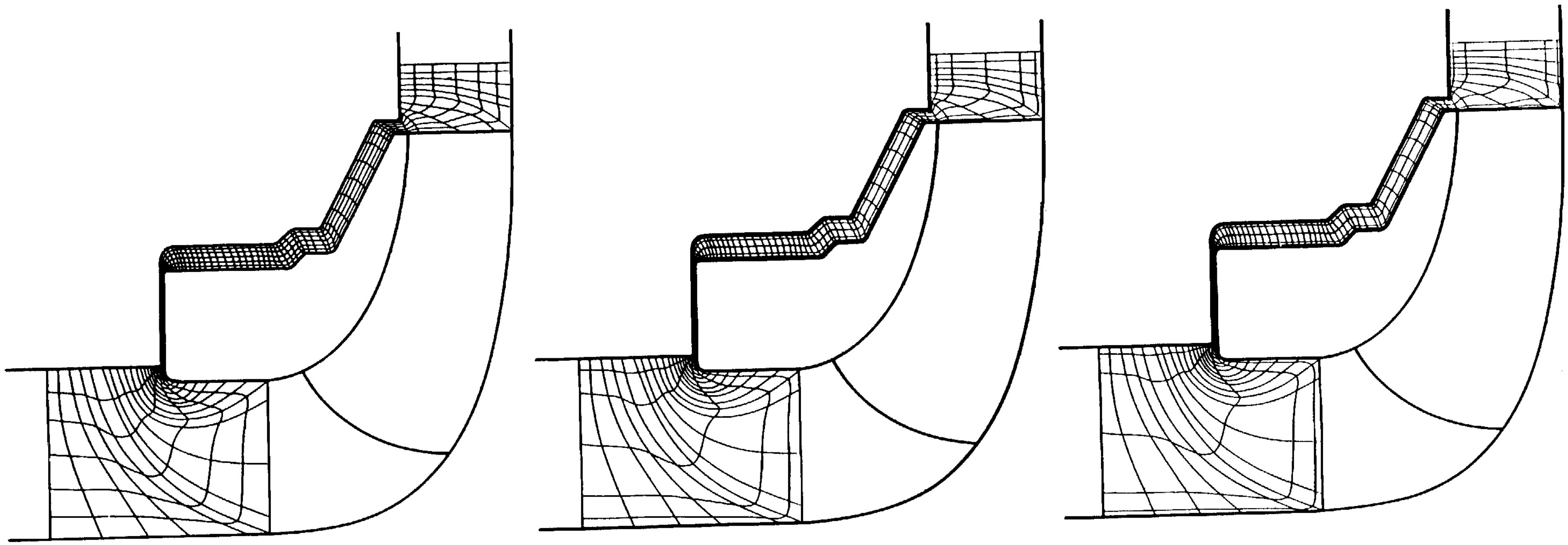


Fig. 5 Isolated leakage passage in traditional analyses



AFINE = 1.0 AFINE= 2.25 AFINE= 2.75

Fig. 6 Effect of the input parameter "AFINE" on the computational domain resolution

DELTAP is the non-dimensionalized static pressure difference across the pump stage (i.e. between the far-upstream inlet station and the stage exit stations (Fig. 3). The non-dimensionalization quantity here is $(\rho {U_t}^2)$, where ρ is the fluid density and U_t is the impeller tip speed (in m/s. Should the problem be that of a seal, then this variable should be either ignored or assigned a zero value.

LINE# 6: (Format 4F20.0)

ANU is the fluid kinematic viscosity coefficient in m^2/s.

RE is the Reynolds' number corresponding to the pump operating conditions. The Reynolds' number here is defined as $(U_t r_t)/\nu$, where U_t and r_t are the impeller-tip speed and radius, respectively, and ν is the kinematic viscosity coefficient.

SCALE is a factor which should always be set equal to 1.0.

RHO is the fluid density in Kg/m^3.

Remark:

In the event IPROB = 0 (i.e. the problem is that of the simple seal or seal-like type), then the Reynolds' number should be based on the average through-flow inlet velocity and a characteristic length that is equal to the distance between the middle points on the flow-passage inlet and exit stations. Now, should IPROB be unity, meaning a shrouded-impeller problem, then the user should skip the following file entries and go directly to Line# 7*.

LINE# 7: (Format I5)

NIN is the number of points at the seal inlet station at which the three velocity components (V_z, V_r and V_θ) are specified. In the event "NIN" is set equal to 1, all of these three velocity components will be considered uniform, with the exception of a boundary-layer profile ,the thickness of which is a function of "PERCNT"(above), at the solid walls.

The following is a group of lines, which would shrink to one should "NIN" be equal to 1.

Lines# 8,9,10,....,[8+(NIN-1)] (Format 4F10.0)

ALIN is the non-dimensionalized distance (along the seal inlet station, and measured from the minimum-radius point) of one of the points at which the profiles of the inlet velocity components are supplied. The non-dimensionalization factor here is the annulus height at the seal inlet station.

VZIN is the specified axial velocity component in m/s.

VRIN is the specified radial velocity component in m/s.

VTIN is the specified tangential velocity component in m/s.

Remark:

In the event IPROB = 0, then the file "DIGIN.DAT' is complete at this point. In the following, an astrisk will be used to signify lines (in the input

file) which exclusively pertain to the case of a shrouded pump impeller, with the computational domain being that of the double-entry/double-departure type (Fig. 4).

LINE# 7*: (Format 3I5)

MIDI1 is the number of the user-created grid line (Fig.2), counted from the pump-stage (far upstream) inlet station, which ends at the minimum-radius point of the impeller inlet station (Fig. 7).

MIDI2 is the number of the user-created grid line which ends at the maximum-radius point of the impeller inlet station (Fig. 7).

MIDI is the number of points on the impeller inlet station (Fig. 7) where the three velocity components (V_z, V_r and V_θ) are to be specified by the user. In the event "MIDI" is set equal to 1, then the profiles of the velocity components at the impeller inlet station will be treated as uniform, with the exception of a boundary-layer profile at the solid walls.

LINE# 8*: (Format 3I5)

MIDO1 is the number of the user-created grid line (counted from the pump-stage inlet station) which ends on the impeller exit-station point with the minimum axial coordinate (Fig. 7).

MIDO2 is the number of the user-created grid line, counted from the pump-stage inlet station, which ends on the impeller exit-station point with the maximum axial coordinate (Fig. 7).

MIDO is the number of points on the impeller exit station (Fig. 7) where the three velocity components (V_z, V_r and V_θ) are to be specified by the user. In the event "MIDO" is set equal to 1, then the profiles of the velocity components at the impeller exit station will be treated as uniform, with the exception of a boundary-layer profile at the solid walls.

The following are the contents of one in a group of MIDI lines. This group would obviously shrink to one line should "MIDI" be equal to 1, in which case "ALMIDI" should be set equal to 0.5.

Lines# 9*,10*,11*,....,[9 + $(MIDI - 1)$]* (Format 4F10.0)

ALMIDI is the non-dimensionalized distance (along the impeller inlet station, and measured from the minimum-radial-coordinate point on this station) of one of the points at which the profiles of the inlet velocity components are supplied. The non-dimensionalization factor here is the annulus height at the impeller inlet station.

VZMIDI is the specified value of the axial velocity component V_z in m/s.

VRMIDI is the specified value of the radial velocity component V_r in m/s.

VTMIDI is the specified value of the tangential velocity component V_θ in m/s.

The following are the contents of one in a group of MIDO lines. This group would obviously shrink to one line should "MIDO" be equal to 1, in which case "ALMIDO" should be set equal to 0.5.

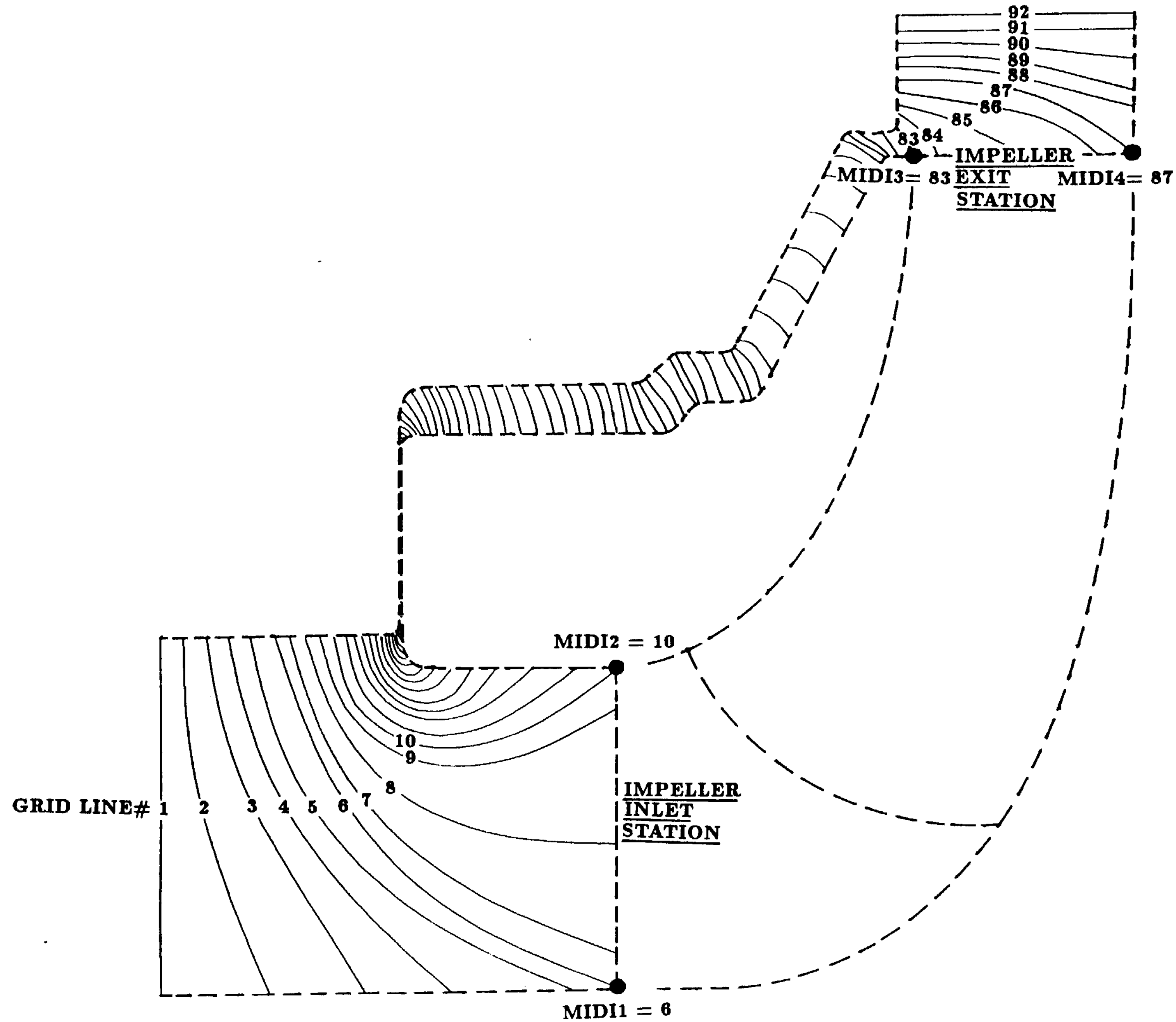


Fig. 7 Input variables in the case of a shrouded–impeller pump stage

Lines# $(9+MIDI)^*, (10+MIDI)^*, ..., [(9+MIDI)+(MIDO-1)]^*$ Format(4F10.0)

ALMIDO is the non-dimensionalized distance (along the impeller exit station, and measured from the minimum-axial-coordinate point on this station) of one of the points at which the profiles of the exit velocity components are supplied. The non-dimensionalization factor here is the axial width of the impeller exit station.

VZMIDO is the specified value of the axial velocity component V_z in m/s.

VRMIDO is the specified value of the radial velocity component V_r in m/s.

VTMIDO is the specified value of the tangential velocity component V_θ in m/s.

CONTENTS OF THE OUTPUT FILE "DIGIT.OUT":

DIGIT.OUT is the only output file that is created by the pre-processor "PREPRO". If desired, this file can be edited before it is used as input to the centered-impeller axisymmetric flow analysis code "AXISYMM" (to be discussed later in this report). Whether edited or not, the file name should be changed (from "DIGIT.OUT") to "AXISYMM.DAT" before executing "AXISYMM". The contents of this file (as produced by "PREPRO") are listed below. Variables which are often altered by the user, in the file editing process, are identified, and recommended changes are also given.

GROUP# 1 - - one line (Format 3I5,E12.5)

This group contains the variable "NITER", which is the desired number of iteration to be performed in a particular run (whether starting from scratch or restarting the code execution) in the process of iteratively solving the non-linear flow-governing equations. This variable is often adjusted depending on, for instance, the magnitude of swirl in the leakage passage (highly swirling flow in this passage usually requires a relatively large number of iterations (around 1000 or even more) for the numerical solution to converge), and the specified magnitude of the maximum tolerable pressure error "ϵ" (to be discussed later in this section). Proper specification of "NITER" is also a function of the user's intuition and/or experience. However, the user should not be overly concerned about the magnitude of this variable since the iterative flow solution process in subprogram "AXISYMM" (to be discussed in the next section) can always be continued (by simply re-executing "AXISYMM", leaving all files unaltered), should the selected value of "NITER" be insufficient. The variable "NITER" is followed, in the same line, by two integers (which should remain unaltered), and the impeller speed (in rpm).

GROUP# 2 - - two lines (Format 12I5/5I5)

The seventeen integers in this group trigger specific options in the axisymmetric flow solver "AXISYMM", such as the type of Lagrangian finite element to be utilized, the manner in which the boundary conditions are to be imposed,....etc. This group should be left unchanged.

GROUP# 3 - - one line (Format 10I5)

This group contains the variables "M1" (which is the total number of corner, midside and interior nodes in the finite-element discretization model shown in Fig. 3), the variable "M2" (which is the total number of corner nodes only), the variable "MTOT" (which is the total number of "degrees-of-freedom in the finite-element model; namely "M1" nodal values of axial velocity components, "M1" nodal values of radial velocity components, "M1" nodal values of tangential velocity components, and "M2" nodal values of static pressures (i.e. $MTOT = 3M1 + M2$). Following these are the variables "NTOT" (which is the total number of the bi-quadratic nine-noded finite elements in the computational domain shown in Fig. 3), an integer which should remain unaltered, the variable "NSPU" (which is the total number of finite-element nodes where the axial velocity component is specified), the variable "NSPV" (which is the total number of nodes where the radial velocity component is specified, the variable "NSPW" (which is the total number of nodes where the circumferential velocity component is specified), the variable "NSPP" (which is the total number of nodes where the static pressure is specified), and the variable "NEXIT" [which is the total number of finite-element sides that coincide with the pump-stage (far downstream) exit station shown in Fig. 7].

GROUP# 4 - - as many lines as the case may be (Format 8F10.6)
This group contains the z-coordinates of all corner, midside and interior nodes in the finite-element model.

GROUP# 5 - - as many lines as the case may be (Format 8F10.6)
This group contains the r-coordinates of all corner, midside and interior nodes in the finite-element model.

GROUP# 6 - - "NTOT" lines (Format 9I5)
Any line in this group contains the nine nodal numbers (in the global system) associated with a certain finite element (see the node numbering scheme in Figs. 8a, 8b, 8c and 8d).

GROUP# 7 - - "NSPU" lines (Format I5,F10.6)
Each line in this group contains a node number and the value of the axial velocity component (i.e. V_z) which is specified at this node.

GROUP# 8 - - "NSPV" lines (Format I5,F10.6)
Each line in this group contains a node number and the value of the radial velocity component (i.e. V_r) which is specified at this node.

GROUP# 9 - - "NSPW" lines (Format I5,F10.6)
Each line in this group contains a node number and the value of the tangential velocity component (i.e. V_θ) which is specified at this node.

GROUP# 10 - - "NSPP" lines (Format I5,F10.6)
Each line in this group contains a node number and the value of the static pressure (p) which is specified at this node.

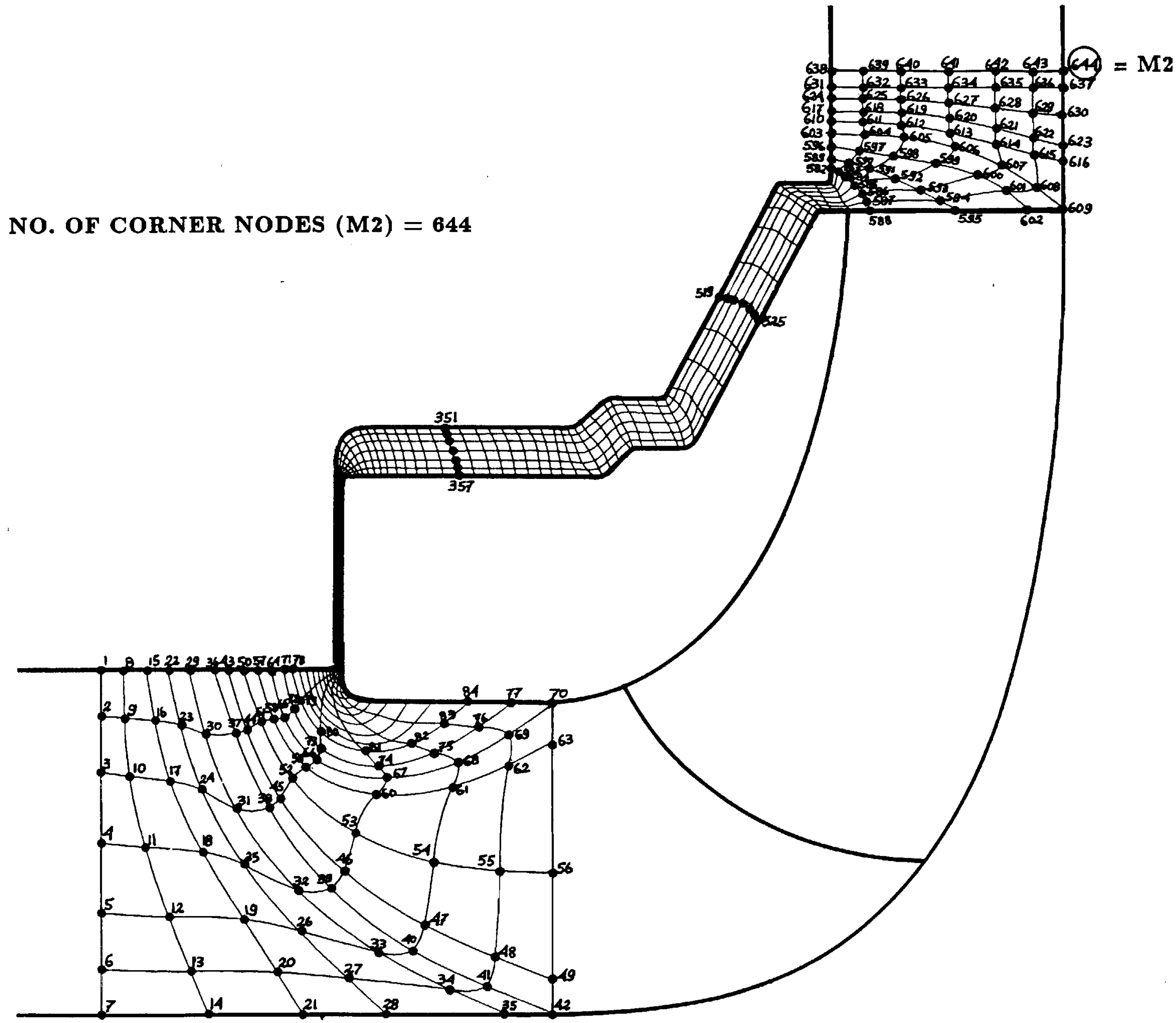


Fig. 8a Numbering of <u>corner nodes</u> in the finite–element model produced by the pre–processor "PREPRO"

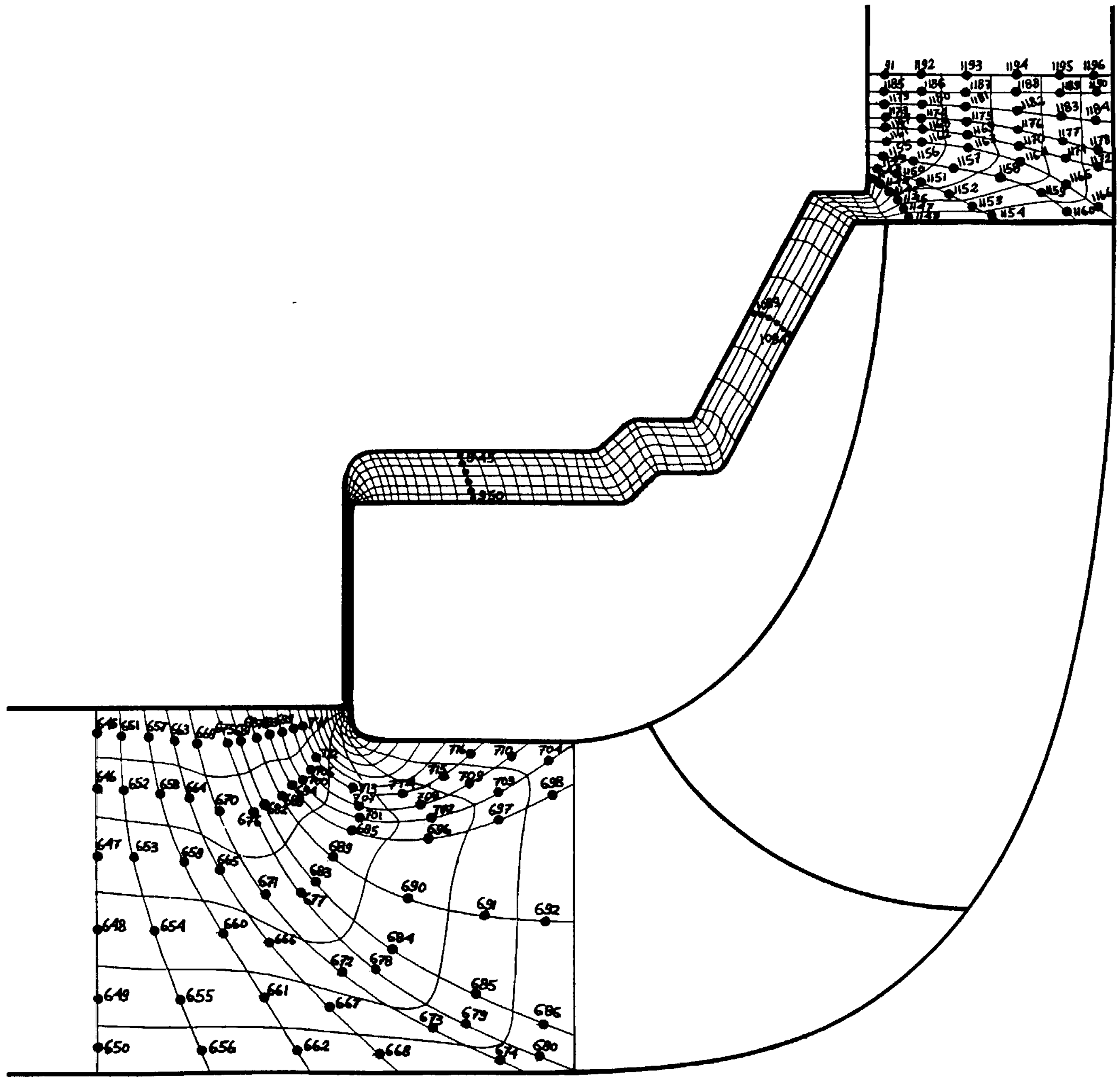


Fig. 8b Numbering of <u>first group of midside nodes</u> in the finite–element model produced by the pre–processor “PREPRO”

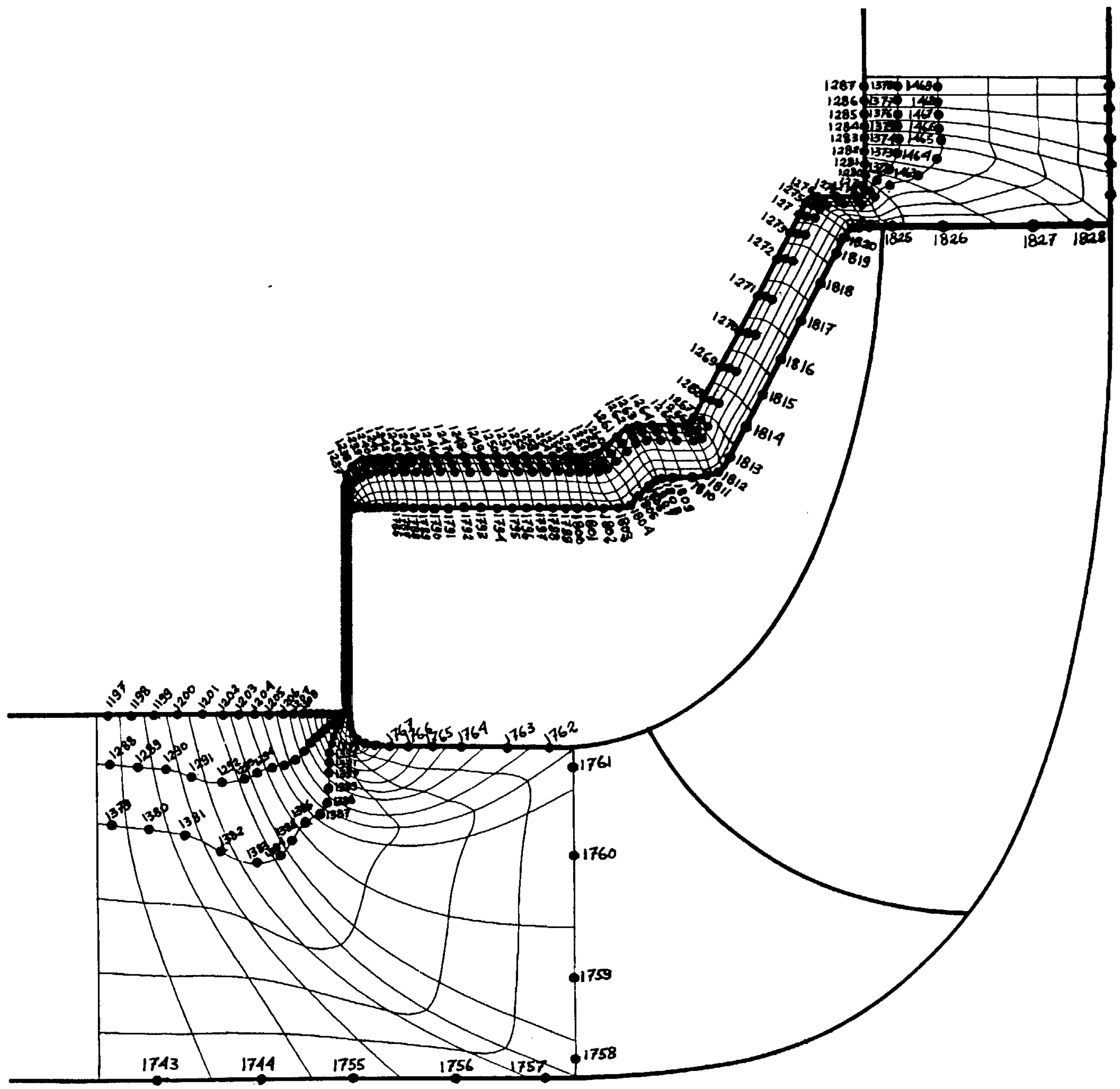


Fig. 8c Numbering of <u>second group of midside nodes</u> in the finite–element model produced by the pre–processor "PREPRO"

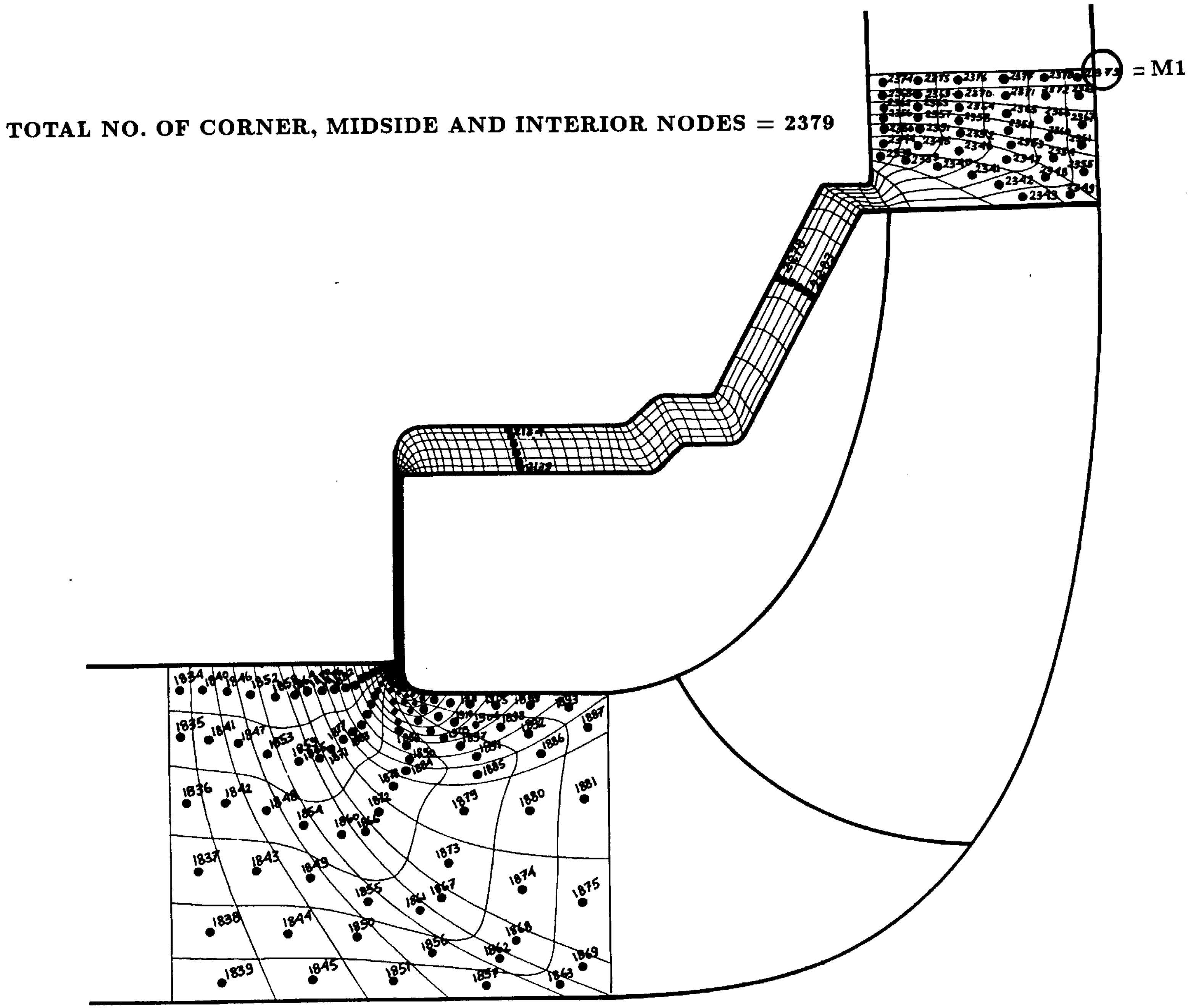


Fig. 8d Numbering of <u>interior nodes</u> in the finite–element model produced by the pre–processor "PREPRO"

Remark:
Referring to the finite–element formulation of the problem (Part I of this report), the user is reminded that the pressure is a declared degree–of–freedom (or an unknown variable) only at the corner nodes of the finite–element model in Fig. 3, while the velocity components (V_z, V_r and V_θ) are degrees–of–freedom at all nodes.

GROUP# 11 - - (NORTH-1) lines (Format 3I5)
With "NORTH" being the number of nodes produced by the pre–processor "PRE-PRO" per each user–created grid line (see the contents of the file "DIGIN.DAT"), each line in this group contains the three nodal numbers associated with a finite element side which coincides with the pump–stage exit station (Fig. 7).

GROUP# 12 - - one line (Format F10.6)
This line contains the variable "ϵ", which is the maximum allowable peak of the normalized–pressure error (at any finite–element node in the entire computational domain), for convergence of the axisymmetric flow solution to be declared. As shown in the Appendix, this variable was arbitrarily set to 2% (note that the default value in the file "DIGIT.OUT" is 1%). Editing "DIGIT.OUT" to effect this change of ϵ magnitude (from 0.01 to 0.02) was aimed at reducing the number of iterations required for convergence while still maintaining a decent, and practically acceptable, convergence criterion. In specifying the value of this variable, the user may want to conduct a reasonably limited numerical "experimentation" study, whereby a high value of, say, 5% is first imposed, then gradually reduced. Upon examination of such items as the meridional velocity fields (in each case), the pressure and swirl–velocity contours,....etc., the user may be in a position to identify the maximum tolerable error magnitude (which is the variable here) below which the axisymmetric flow field remains practically unchanged. In the end, the user's conclusions, from this "experimentation" study, will not only be of use in economically executing the problem at hand, but may also be valuable in future investigations of other characteristically–similar problems.

GROUP# 13 - - one line (Format F10.6)
This is the characteristic length (impeller tip radius for the pump problem, or total axial length for the simpler seal problem) with which all coordinates are non–dimensionalized. This length is also used in the Reynolds' number definition.

GROUP# 14 - - one line (Format 3F20.10)

This line consists of the Reynolds' number, based on the characteristic length (above) and the impeller tip speed (in the pump problem), or the seal average through–flow inlet velocity (in the seal problem). This is followed by the fluid density (in kg/m^3) and the kinematic viscosity "ν" (in m^2/s).

GROUP# 15 - - two lines (Format 3F10.6/3F10.6)
These two lines contain the Gaussian constants for integrating a general function over a square.

GROUP# 16 - - one line (Format 2F10.6)
This line contains two constants which are no longer used by the axisymmetric flow solver "AXISYMM" but should, nevertheless, remain unaltered.

GROUP# 17 - - one line (Format F10.6)
This line contains the underrelaxation factor which is used in progressing from one iteration to the next in the subprogram "AXISYMM" (where a set of non-linear flow-governing equations are solved). A large value of this factor (e.g. 0.8) may cause divergence of the numerical solution. On the other hand, a small value (e.g. 0.1) may cause numerical oscillations. The recommended range of this factor is between 0.2 and 0.6, depending on the Reynolds' number (where a low value of the underrelaxation factor is to be chosen), as well as the complexity of the anticipated axisymmetric flow field (for instance, the presence of excessive flow recirculation subdomain requires a relatively small value of the underrelaxation factor). The default value printed in this file is 0.4, and is subject to adjustment by the user depending on the problem under consideration. In general, a small value of this factor is an indication of conservatism on the part of the user.

Remark:
Group# 17 (above) is the last file segment that is read in by subprogram "AXISYMM" (to be discussed next). In reality, however, subprogram "PREPRO" will generate a few more lines (consisting of mainly integers. Although there is no harm in leaving these useless lines where they belong, the file listing (in the APPENDIX) is one where this group of lines is totally extracted.

SUBPROGRAM "AXISYMM"

DESCRIPTION

This subprogram accepts the output file of "PREPRO" (which has just been described) as the major input file. Recall that the name of this file has to be changed from "DIGIT.OUT" to "AXISYMM.DAT" prior to executing this axisymmetric flow solver. The output file of "AXISYMM" is named "AXISYMM9.DAT", and contains the solution of Navier–Stokes equations, coupled with the vorticity–based turbulence closure of Baldwin and Lomax (1978), throughout the computational domain (Fig. 1). If desired, this axisymmetric (or centered–impeller) flow solution can be displayed (for instance, as a velocity vector plot and/or a pressure contour plot) using the two (input and output) files; "AXISYMM.DAT" and "AXISYMM9.DAT". The necessity of the former file stems from the fact that it contains the geometrical data (i.e. the nodal coordinates and system topology), which is not contained in the output file "AXISYMM9.DAT".

The output file of this subprogram, namely "AXISYMM9.DAT" is usually the outcome of between 200 and 900 iteration, depending on the Reynolds' number and the complexity of the flow field. This is not the only input file to this subprogram, a complete description of all the I/O files is given below:

CONTENTS OF THE INPUT FILES:

1. INPUT FILE "AXISYMM.DAT":

As mentioned earlier, the contents of this file are precisely those of the pre–processor output file "DIGIT.OUT", which has just been discussed. Again, the user should only change this file name into "AXISYMM.DAT".

2. INPUT FILE "AXISYMM7.DAT":

As the user begins to execute "AXISYMM", this file will have to exist, and will be composed of one simple line containing zero in column# 5. However, if the user is re–executing the subprogram "AXISYMM" (because the maximum allowed number of iteration was insufficient for convergence), then the file "AXISYMM7.DAT" will contain a solution that is composed of the numerical results of the latest two iterations, with one set of results more emphasized according to the magnitude of the underrelaxation factor "UNDRLX" (which is an input variable to "PREPRO". Although the numerical results on this file constitute a yet invalid solution, the contents of this file should be left unaltered as they are used as an "initial guess" once the iterative solution process is restarted.

3. INPUT FILE "AXISYMM9.DAT":

Although this is also the output file of subprogram "AXISYMM", this file will have to be initialized, in the same manner as the file "AXISYMM7.DAT (above), if the user is running "AXISYMM" for the first time. Similar to file "AXISYMM7.DAT" (above), the user should not alter the contents of this file if the job is that of re-executing "AXISYMM".

CONTENTS OF THE OUTPUT FILE "AXISYMM9.DAT":

GROUP# 1 - - one line (Format I5,3E12.5)

This line contains the number of iterations performed, followed by the current peak value of the normalized-pressure error (over the entire computational domain), then the Reynolds' number of operation (same value supplied by the user) and, finally, the integrated volumetric flow rate through the leakage passage (in m^3/s).

GROUP# 2 - - as many lines as required (Format 6E12.6)

This group contains the nodal values of the axial velocity component "V_z" at all corner, midside and interior nodes in the finite-element model , and in the order shown in Fig. 8 (on three successive pages. Note that all of the velocity components (in this and the next two groups) are non-dimensionalized using the impeller tip speed (in the case of a shrouded-impeller pump), or the average inlet through-flow velocity (in the case of a simple seal). The number of entries in this group is "M1".

GROUP# 3 - - as many lines as required (Format 6E12.6)

Same as group# 2(above), except that the field variable here is the non-dimensionalized radial velocity component "V_r".

GROUP# 4 - - as many lines as required (Format 6E12.6)

Same as group# 2(above), except that the field variable here is the non-dimensionalized tangential velocity component "V_θ".

GROUP# 5 - - as many lines as required (Format 6E12.6)

Same as group# 2(above), except that the field variable here is the eddy kinematic viscosity coefficient "ν_t" in m^2/s.

GROUP# 6 - - as many lines as required (Format 6E12.6)

This group contains the non-dimensionalized nodal values of static pressure ($p^* = (p - \bar{p})/(\rho \bar{V}^2)$, where the reference pressure "$\bar{p}$" is the pressure at the midway node on the impeller inlet station (in the shrouded–impeller pump problem), or at the midpoint of the seal discharge station (in the simple seal and "seal–like" problems). The reference velocity "$\bar{V}$", on the other hand, is equal to impeller tip speed (for the shrouded–impeller pump problem), or the average through–flow velocity at the inlet station (for the seal and "seal–like" problems). The number of entries in this group is equal to the total number of the corner nodes in the finite–element model (i.e. "M2"), since the pressure is declared as an unknown variable only at these nodes.

<u>GROUP# 7 – – as many lines as required</u> (Format 6E12.6)

Same as group# 1 (above), except that the variable here is the absolute vorticity (in s^{-1}). This particular group is normally of little or no use to the user, but is the single most important flow variable in the procedure to compute the eddy viscosity coefficient in the subprogram "AXISYMM".

SUBPROGRAM "INTERPOLATE"

Remark:

This particular subprogram does not have to be executed, unless the user foresees that the number of nodes in the three–dimensional computational domain (Fig. 2) will require more than the available disk space. Note that this three–dimensional model is derived from its two–dimensional (axisymmetric) version shown in Figs. 3&11. For the sake of accuracy, the user is advised to first verify the disk space sufficiency by overriding this subprogram. In this case, all the user has to do is re–name the two existing files "AXISYMM.DAT" and "AXISYMM9.DAT" as "PUMPSMALL.DAT" and "PUMP9SMALL.DAT", respectively. In pursuing the computational procedure this way, the user may eventually get an error message implying insufficiency of the disk space. Should this be the case, the user has no choice but to go back and execute the current subprogram in order to reduce the size of the finite–element model. This process will require re–execution of the pre–processor "PREPRO" using two modified input files as explained later in this section.

DESCRIPTION

The purpose of this subprogram is to reduce the finite–element model size (in terms of the node and element counts) resulting in a coarser, but "manageable", grid prior to implementing the perturbation model. From a computer–resources standpoint, the aim here is to decrease the core and CPU time requirements in executing the perturbation analysis code "PERTURB" (to be discussed later in this report). The reason is that the computational domain at this computational step is the three–dimensional (non–axisymmetric) shroud–to–housing region in which a laterally–displaced impeller/shroud assembly is whirling around the housing centerline (Fig. 2). In fact, the first step in this perturbation model is to create a "geometrically" three–dimensional computational domain by repeating, around the circumference, the two–dimensional finite–element grid (Fig. 3), as well as the axisymmetric flow solution a sufficient number of times (usually between 9 and 11). This creates what is referred to as the "unperturbed" domain, and the "zeroth–order" flow solution, both being treated in a three–dimensional sense. The user, at this point, should be careful not to create an exceedingly large numerical model with which to proceed in the perturbation analysis. Since a number of planes around the conference (Fig. 3) that is less than 9 is established to cause inaccuracies in the computed rotor-dynamic coefficients, the only practical means to avoid the cost and/or computer–related consequences is to reduce the size of the two–dimensional (axisymmetric) grid (Fig. 3). This task is achieved in this subprogram, which also produces the flow solution, associated with the reduced grid, by interpolation. In doing so, The original (assumingly large) two dimensional grid (produced by the pre–processor "PREPRO") and the corresponding flow solution (produced by the subprogram "AXISYMM") are both used as input files. Also an input is a geometry file corresponding to a coarser (i.e. smaller size) finite–element model, which can also be obtained by re–using "PREPRO" only, this time, the user should feed–in a smaller of the already created grid lines (Fig. 4) and/or a reduced value of the input variable "NORTH" (refer to the contents of the input file "DIGIN.DAT" in the section entitled "SUBPROGRAM PREPRO"). The output of "PREPRO" will now reflect a coarser finite–element grid, such as that in Fig. 9.

INPUT FILE "PUMPLARGE.DAT"

This is simply the existing geometry file "DIGIT.OUT" which, at this point, has already been generated by the pre–processor "PREPRO", and used as input to the centered–impeller axisymmetric flow solver "AXISYMM". Since the file is already existing, all the user has to do is to rename it as "PUMPLARGE.DAT".

INPUT FILE "PUMP9LARGE.DAT":

This, at this point, is also an existing file, and is simply the output file "AXISYMM9.DAT" of the axisymmetric flow analysis subprogram "AXISYMM". Again, the user should simply change the file name into "PUMP9LARGE.DAT".

INPUT FILE "PUMPSMALL.DAT":

To create this file, the user will have to re–execute the pre–processor "PREPRO". The only difference this time, is that the pre–processor input files ("DIGIT.DAT" and "DIGIN.DAT") have to be edited in such a way that: a) a sufficient number of the user–created grid lines (Fig. 3) is deleted from the file "DIGIT.DAT", b) the new (smaller) number of the grid lines retained (i.e. the new value of the input variable "NCURVE") is properly entered in the file "DIGIN.DAT", c) the desired number of corner nodes per each of these grid lines (i.e. the input variable "NORTH") is sufficiently reduced, and d) the refinement factor "AFINE" in the file "DIGIN.DAT" is increased (Fig. 6).

Remark:

Although the modified input files (above) will still have to be saved under the names "DIGIT.DAT" and "DIGIN.DAT", these two files are listed in the APPENDIX under the names "DIGITSMALL.DAT" and "DIGINSMALL.DAT", respectively. The reason is that the former file names are those associated with the original (large–size) finite–element model.

Re–execution of the pre–processor with the above adjustments will result in a smaller–size finite–element model (in the sense of node and finite–element counts) or, equivalently, a coarser finite–element model. The output file of "PREPRO" (namely as "DIGIT.OUT") is then simply renamed as "PUMPSMALL.DAT"".

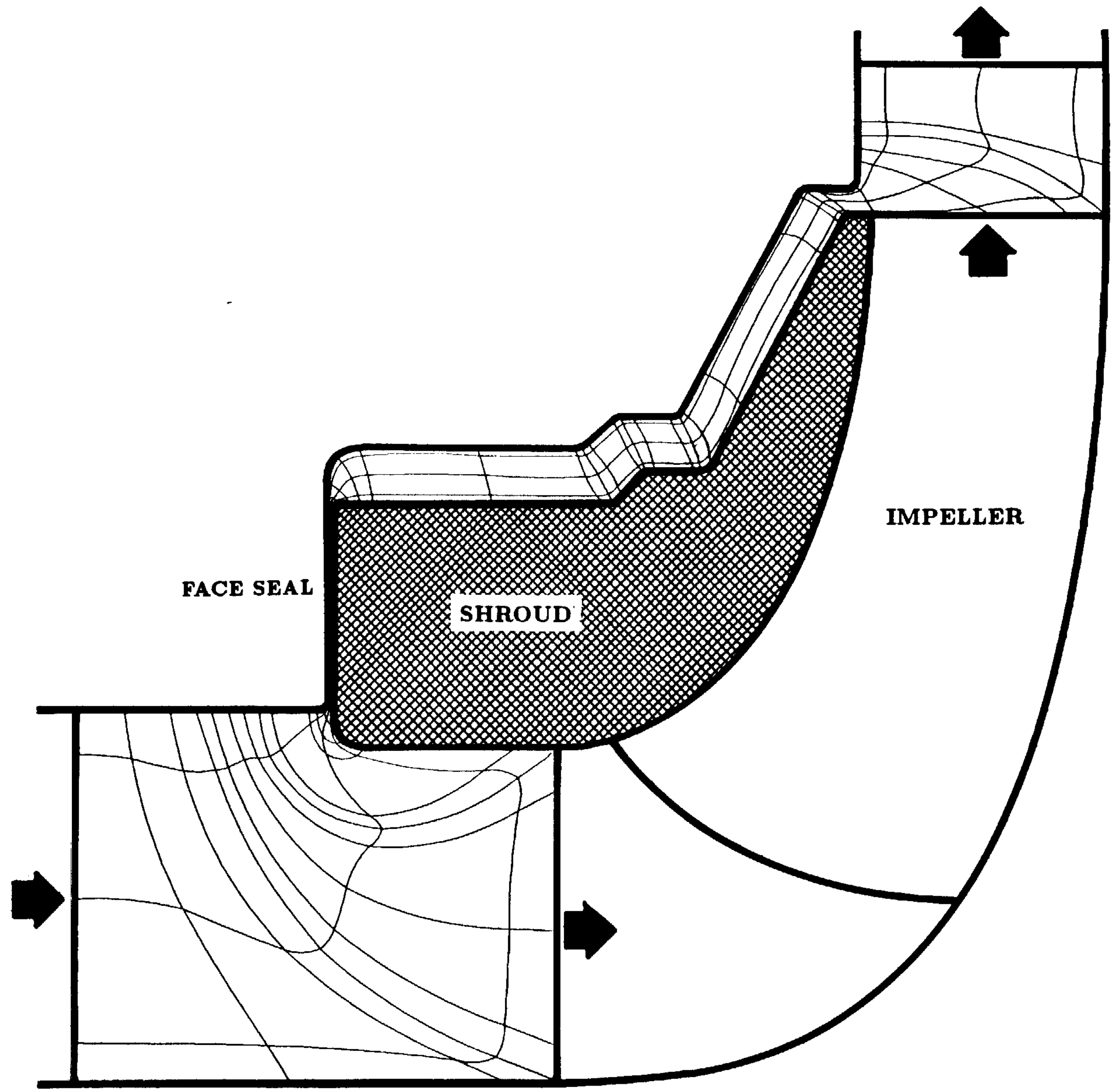


Fig. 9 A reduced–size (coarser–grid) finite–element model of the leakage passage in Sulzer Bros.' pump stage

OUTPUT FILE "PUMP9SMALL.DAT":

This file can be viewed as the smaller–finite–element–model version of the already existing file "PUMP9LARGE.DAT", which is the original axisymmetric flow–solution file (previously produced by the subprogram "AXISYMM"). The new file is the only output of the subprogram "INTERPOLATE", and is obtained by interpolating the velocity components, pressure, eddy viscosity and vorticity upon laying the smaller–size finite–element grid (in the file "PUMPSMALL.DAT") over the original grid (in the file "PUMPLARGE.DAT").

Remarks:

In executing the subprogram "INTERPOLATE", the user should expect to be prompted for the number of computational planes (N3D) around the circumference. This variable is needed in constructing the three–dimensional version of the finite–element grid (Fig. 2). The value of "N3D" should be between 9 and 13 for a realistic tangential resolution of the flow variables in the process of performing the perturbation analysis (to be discussed later in this report).

Also, from this point on, the pair of geometry and solution files, corresponding to the axisymmetric flow analysis step, will always be "PUMPSMALL.DAT" and "PUMP9SMALL.DAT", replacing their large finite–element–model counterparts which were actually used in this computational step.

Finally, in the event the user desires to skip subprogram "INTERPOLATE" (for the reason that reduction of the original grid is not necessary, from a computational resources standpoint, then all the user has to do is to rename the two files "DIGIT.OUT" and "AXISYMM9.DAT" (which have already been produced by subprograms "PREPRO" and "AXISYMM") as "PUMPSMALL.DAT" and "PUMP9SMALL.DAT", respectively, before proceeding to the next subprogram.

SUBPROGRAM "ROTSUR"

DESCRIPTION

An outcome of the last subprogram of this computational procedure (namely subprogram "PERTURB", which will be described next) is the set of pressure perturbation values at the shroud–surface finite–element In order to meaningfully use this output, subprogram "ROTSUR" in effect "unwraps' the shroud surface (Fig. 10), and identifies the shroud–surface nodes, each by its number and coordinates in the 'unwrapped–shroud' plane for plotting purposes. Before executing "ROTSUR", the user should make sure that the files "PUMPSMALL.DAT" and "PUMP9SMALL.DAT" (see subprogram "INTERPOLATE") are existing, since these files go as input to subprogram "ROTSUR" Besides, "ROTSUR" will prompt the user (upon execution) for two integers; namely the two numbers (counted from the far–upstream flow–inlet station) of the user–created grid lines (Fig. 4) between which the shroud is desired to be 'unwrapped'. The output file of this subprogram (the description of which follows) goes as input to the last subprogram "PERTURB", where the file is re–created, but with the nodal values of pressure perturbations (computed in "PERTURB") properly inserted.

CONTENTS OF THE OUTPUT FILE "ROTSUR.DAT":

GROUP# 1 – – one line (Format 2I5)

NSURF is the total number of finite–element nodes on the shroud segment for which the pressure–perturbation distribution (in the 'unwrapped' shroud view, Fig. 10) is desired.

INCL is the number of computational planes around the circumference (Fig. 2) plus one (i.e. N3D+1).

GROUP# 2 – – "NSURF" lines [Format I5,2(1X,E13.6)]

ISURF is the nodal number (in the three–dimensional finite–element model, Fig. 2) of a node on the shroud surface.

XPLOT is the z–coordinate of the same finite–element node.

YPLOT is the $r\theta$– coordinate (arc length) of the same node.

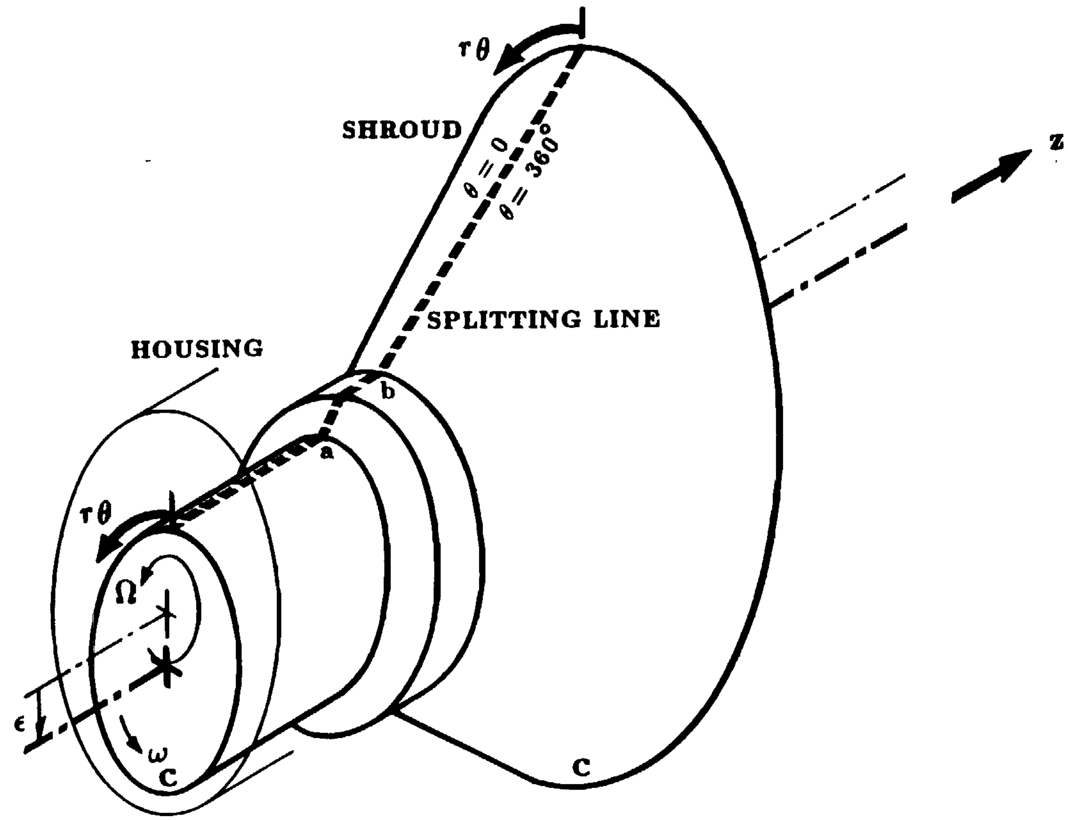


Fig. 10 The process of "unwrapping" the shroud surface

SUBPROGRAM "PERTURB"

DESCRIPTION

This is the segment of the computational procedure where the perturbation analysis (Baskharone and Hensel, 1991a&b) is performed, the fluid-exerted forces (on the shrouded impeller) computed, and the shroud-surface pressure-perturbation distribution produced. The theory on which this analysis is based is what what we have 'generically' referred to as the 'virtually" deformable finite element concept, which is fully documented in the above-referenced papers. The emphasis here is on the set of I/O files of this subprogram.

With the exception of one simple input file, all of the other input files to this subprogram; namely "PUMPSMALL.DAT", "PUMP9SMALL.DAT" and "ROTSUR.DAT", should be existing at this point. Editing of these files is neither required nor encouraged. In the following, the remaining input file, namely "WHIRLF.DAT" is described. Next, the contents of the output file, and interpretation of each output variable, are provided.

CONTENTS OF THE INPUT FILE "WHIRLF.DAT":

This file is composed of one line containing one variable (Format F10.0). This variable is the selected whirl-frequency/operating-speed ratio (or whirl frequency ratio), for which the fluid-exerted forces (over the shroud surface) are desired.

CONTENTS OF THE OUTPUT FILE "F3DX.DAT":

GROUP# 1 - -two lines (Format: mixture of character and real numbers)

This group contains the final results corresponding to the selected whirl frequency ratio. These results are composed of the tangential, radial and axial components of the fluid-exerted force on the shroud surface ($F_\theta{}^*$, $F_r{}^*$ and $F_z{}^*$, with the superscript (*) implying a partial derivative with respect to "ϵ" (the impeller eccentricity). This data (on one line) is followed (on the next line) by the fluid-induced moments "$M_x{}^*$" and "$M_y{}^*$", around the x- and y-axes [which are attached to the housing centerline (Fig. 2)], due to the impeller lateral eccentricity "ϵ". Note that these moments are created, during the impeller cylindrical whirl, as a result of the through-flow pressure gradient along the leakage passage. Units of the fluid-created forces and moments in this group are Newtons/m and Newtons, respectively.

GROUP# 2 - - three lines (Character Format)

this group contains some instructions to the user in the event the pressure perturbation contours, over the 'unwrapped' shroud surface, are desired.

GROUP# 3 - - one line (Format 2I5)

NSURF is the total number of finite-element nodes on the shroud segment for which the pressure-perturbation distribution [in the 'unwrapped' shroud view (Fig. 10)] is desired.

INCL is the number of computational planes around the circumference (Fig. 2) plus one (i.e. N3D+1).

GROUP# 4 - - "NSURF" lines [Format I5,3(1X,E13.6)]

ISURF is the nodal number [in the three-dimensional finite-element model (Fig. 2)] of a node on the shroud surface.

XPLOT is the z-coordinate of the same finite-element node.

YPLOT is the $r\theta-$ coordinate (or arc length) of the same node.

PSHROUD is the pressure perturbation at the shroud-surface node. The symbol used for this variable is p^*, where $p^* = \frac{\partial p}{\partial \epsilon}$, and the variable units are $Newtons/m^3$. The reader is reminded that integration of the forces and moments resulting from this pressure perturbation (once integrated over the entire shroud surface) is what gives rise the the fluid-induced forces and moments (GROUP# 1 above).

Remark:

At this point, the computational procedure leading, in particular, to the fluid-induced forces over the shroud surface is complete for the "selected" whirl frequency ratio. The user should now select a different ratio [in the input file "WHIRLF.DAT (above)], and re-execute subprogram "PERTURB" to compute the new fluid-exerted forces. This process should be repeated until a sufficient range of "likely-to-prevail" whirl frequency ratios (e.g. the range between -2.0 and +2.0), is fully covered. Using these forces as input to the post-processor "LEASTSQ" (described next), the force coefficients (namely the direct and cross-coupled stiffness, damping and inertia coefficients), of the shrouded impeller, are finally deduced.

POST–PROCESSOR "LEASTSQ"

DESCRIPTION:

This post–processor accepts the fluid–exerted force components (F_θ^* and F_r^*) over a range of whirl frequency ratios (as computed, for each ratio, by subprogram "PERTURB"), and yields the rotordynamic coefficients (usually referred to as the "force" coefficients). In doing so, the post–processor considers each force component (e.g. F_θ^*) separately, and establishes a parabolic least–square fit of the given force–versus–whirl frequency pairs of input data. Employing the method outlined by Baskharone and Hensel (1991a), and assuming that the force distribution (over the entire whirl frequency range) is essentially quadratic, the impeller rotordynamic coefficients are computed and printed in the output file. Note that this subprogram utilizes a subroutine from the Mathematics/Science Library (IMSL), version 9.2.

CONTENTS OF THE INPUT FILE "FEFORCES.DAT":

GROUP# 1 - - one line (Free Format)

N is the total number of whirl frequency ratios at which the fluid–exerted force components have already been computed.

GROUP# 2 - - one line (Free format)

ω is the impeller rotational speed in rad/s.

GROUP# 3 - - "N" lines (Free Format)

Ω/ω is the whirl frequency ratio, where "Ω" is the whirl frequency, and "ω" is the impeller speed.

F_θ^* is the computed tangential component of the fluid–exerted force (in Newtons/m).

F_r^* is the computed radial component of the fluid–exerted force (in Newtons/m).

CONTENTS OF THE OUTPUT FILE "COEFF.OUT":

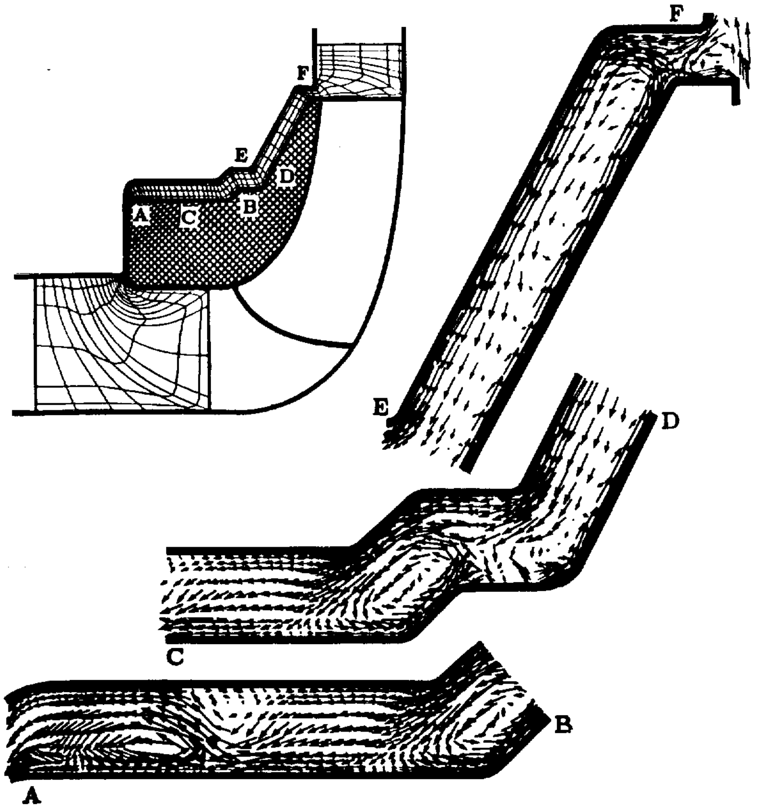
This file contains the direct and cross–coupled stiffness, damping and inertia (or added mass) coefficients. The units of these coefficients are Newtons/m (for the stiffness coefficients K & k), $\frac{N.s}{m}$ (for the damping coefficients C & c), and kg [for the inertia (or added mass) coefficients M & m].

OTHER POST–PROCESSORS

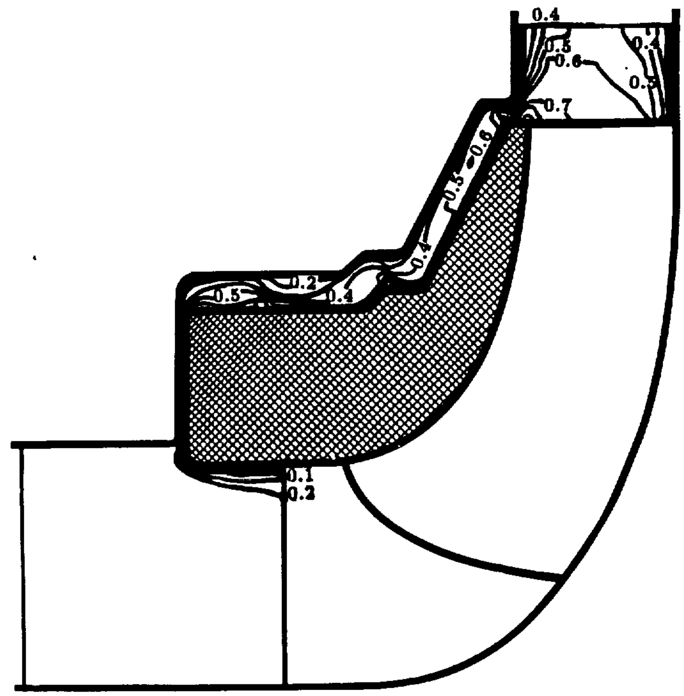
In conducting the current study, three additional post–processors were developed by the principal investigators in order to visually verify the results of the three major steps of the computational procedure. These are the steps of creating the finite–element discretization model (subroutine "PREPRO"), solving the axisymmetric 'centered–impeller' flow field (subroutine "AXISYMM"), and performing the perturbation analysis (subprogram "PERTURB"). All of these post processors utilize specific graphics software packages (such as the DI–3000) and may, therefore, be of no use to the user (note that the only external software that is assumed to be accessible to the user is the Mathematics/Science library "IMSL", version 9.2).

The emphasis in this section is placed on instructing the user on how to create the equivalent of the three post–processors (above) using the graphics packages to which he/she has access. In the end, the user should be able to create the three categories of plots shown in Fig. 11.

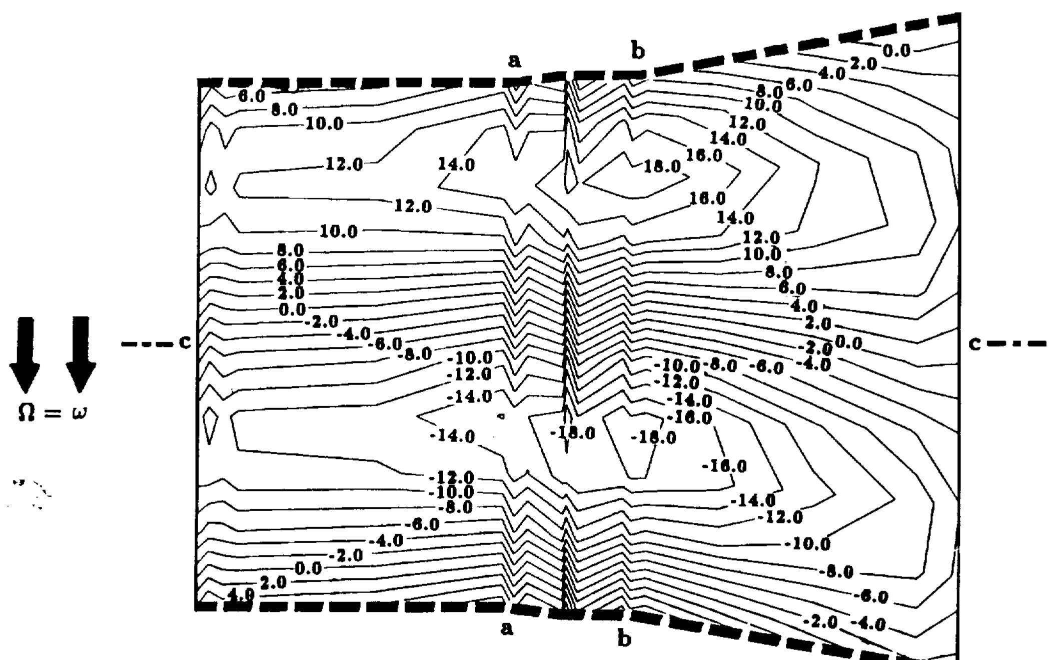
In examining the three plots in Fig. 11, the user should be aware of several approximations and choices that were made. First, the finite–element model plot (Fig. 11a) does not show any midside or interior nodes for the sake of simplicity. As for the plot of the meridional velocity associated with the centered–impeller operation mode (Fig. 11b), the user should note the following: 1) each vector in this plot corresponds to and passes through one of the finite element nodes (Fig. 3), such that the node itself lies at the middle of the vector. 2) In plotting these velocity vectors, either or both components of the vector (i.e. "V_z" and "V_r") may be substantially large (e.g. at the seal inlet and exit stations, as well as within the seal itself). As a result, the vector representing the resultant meridional velocity may fictitiously appear to cross a solid wall (depending on the scale of the vectors relative to the domain dimensions). The user in this case will have to experiment with different vector and domain scales in order to alleviate the appearance of such false flow behavior. The last plot in this group (Fig. 11c) is that of the pressure perturbation contours over the 'unwrapped' shroud surface. As indicated in the section entitled: SUBPROGRAM "ROTSUR", the user has control on which segment of the shroud surface is to be 'unwrapped' (Fig. 10). In examining the pressure perturbation distribution corresponding to this shroud segment, the user is reminded that it is this distribution which, once integrated over the entire shroud surface, yields the net fluid–exerted force for a given whirl frequency ratio. It may, therefore, be of interest to the user to examine how these pressure contours vary, in shape and magnitude, as the whirl frequency ratio is altered. This usually provides a means of understanding the changes (whether smooth or abrupt) of the fluid force components within the whirl frequency range under investigation.



a) VECTOR PLOT OF THE MERIDIONAL VELOCITY



b) CONTOUR PLOT OF THE CIRCUMFERENTIAL VELOCITY



c) CONTOUR PLOT OF THE PRESSURE PERTURBATION DISTRIBUTION OVER THE "UNWRAPPED" SHROUD SURFACE

Fig. 11 Examples of the plots produced by the post-processors

DEVELOPMENT OF BASIC POST–PROCESSORS

In order to create a post processor which generates a plot of the finite–element grid (Fig. 11a), the user needs only one readily existing output file, namely "DIGIT.OUT" of subprogram "PREPRO". as described earlier, this file contains the meridional coordinates (z and r) of each computational node, as well as the system topology (meaning the numbers of the nine nodes associated with each element). In order to display this so–called "discretization model", the post–processor (being developed) should start by reading and saving all of the nodal coordinates in two arrays, say "Z" and "R". Next the user will cycle through all elements, reading the nine nodal numbers associated with it, and determining the coordinates of these nodes from the global nodal–coordinate arrays "Z" and "R". Using an appropriate graphics software (assumingly accessible to the user) the finite element can now be plotted by drawing the parabolic element–boundary segments shown in Fig. 3, using the eight corner and midside nodes. Finally, the ninth (interior) node can be separately plotted (if desired).

A post–processor which would produce a meridional–velocity vector plot, such as that in Fig. 11b, can be developed using the same graphics software (above). Here, the input and output files of subprogram "AXISYMM" (namely "AXISYMM.DAT" and "AXISYMM9.DAT") will have to be used. The logic here is to cycle through all of the finite–element nodes, one at a time. For any of these nodes, the user should extract the coordinates of this node from the coordinate arrays "Z" and "R" (above) and plot the node. Referring to the output file "AXISYMM9.DAT", the meridional velocity components ("V_z" and "V_r") of this node can easily be found in the file "AXISYMM9.DAT". Next, the user should follow the instructions of the graphics software to draw a vector of length that is proportional to the resultant meridional velocity, and a slope that is equal to that of the resultant velocity. This vector should pass through the node under consideration, such that the node itself is at the exact middle of it.

The last basic plot which the numerical results facilitate, is that of the pressure perturbation contours over the "unwrapped' shroud surface (Fig. 11c). The output file needed in this case is that of subprogram "'PERTURB", namely "F3DX.DAT" upon "peeling off" the first two lines. The first line in the resulting 'shortened' file contains the total number of the shroud surface nodes between the two limits previously supplied by the user (see subprogram "ROTSUR"), and the number of nodes lying on one column in the 'unwrapped' shroud view. Dividing the first integer by the second yields the number of such columns of nodes.The remainder of the file "F3DX.DAT" (as mentioned earlier) is a column–by–column listing of the coordinates and pressure perturbation values corresponding to the nodes comprising each column (note that all nodes here are corner nodes, since the pressure is a declared degree–of–freedom at the corner nodes only). Stated differently, the shroud nodes are distributed over the 'unwrapped' shroud surface in the form of a number of rows (obtained by dividing the first integer in line# 1 by the second integer in the same line), with each column containing the same number of nodes. The user may, in this case, create trapezoids using these nodes as vertices, with the field variable, meaning the pressure perturbation, known at the four vertices of each trapezoid . Using a suitable graphics software, that is capable of interpolating the pressure perturbation throughout each trapezoid, the contours of the shroud pressure perturbation can be generated in a pattern similar to that in Fig. 11c. Note that a contour generator, which is based on interpolating the variable along vertical lines, can be used instead, since the columns of nodes over the shroud "unwrapped" surface are vertical lines.

SPECIAL CASE: SEAL ROTORDYNAMIC COEFFICIENTS

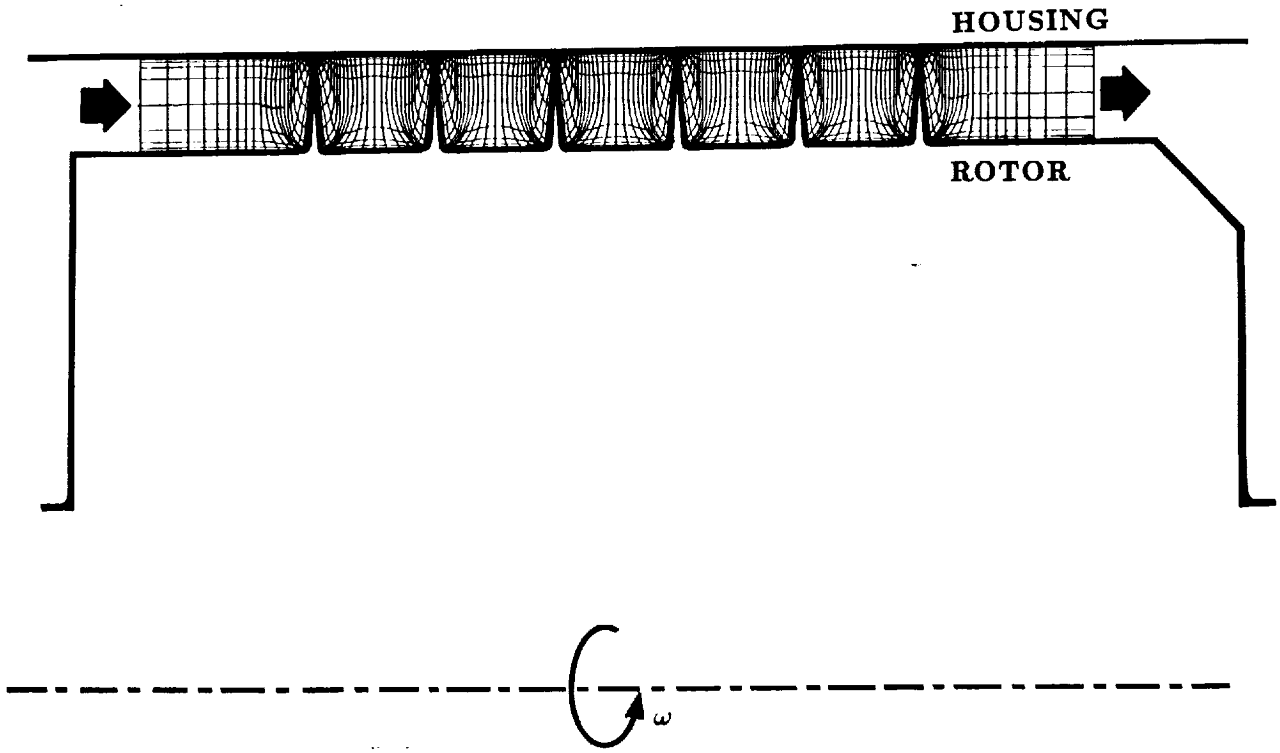
This relatively simple case was referenced earlier in describing the input files to the pre-processor "PREPRO", more specifically the file "DIGIN.DAT". The differences between the file contents in this case, relative to that of a shrouded impeller, were previously outlined (e.g. the value of the input variable "IPROB", which is to be set equal to zero in the case of a seal problem). The most significant difference, however, concerns the boundary conditions which, as would be expected, are much simpler in the seal problem. As for the other input file "DIGIT.DAT", there is conceptually no difference between the file contents in the two cases. In fact, the user-created grid lines in this file, which should ideally be in the cross-flow direction are much easier to construct in the case of a seal, as illustrated in Fig. 12 for the labyrinth and smooth annular seal configurations.

Once the files "DIGIN.DAT" and "DIGIT.DAT" are prepared, and the subprogram "PRE-PRO" executed, the rest of the computational procedure is precisely the same as that outlined in the last few sections. However, the user should be aware of the the special manner in which the field variables are non-dimensionalized in the seal problem:

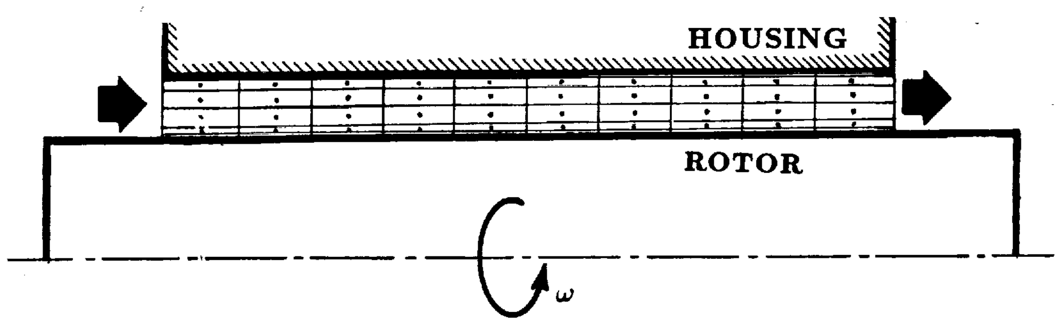
1. The velocity components (V_z, V_r and V_θ) are non-dimensionalized using the average through-flow velocity component V_{in} at the seal inlet station
2. the static pressure is non-dimensionalized using the quantity $[\rho V_{in}^2]$, where ρ is the fluid density.

Definition of the Seal-Like Problem:

The term **SEAL** in this report embraces not only the conventional categories (e.g. those in Fig. 12), but also any annular flow passage of virtually any meridional shape, provided that the passage inner surface is that of a rotor and the outer surface is the housing. An illustrative example of this definition is the isolated shroud-to-housing leakage passage in Fig. 5, with the two upstream and downstream primary-flow passage removed. Although the computational domain, in this case, contains a face seal, the domain itself is analytically treated as a generally shaped **SEAL**, the variable "IPROB" equal to zero (refer to the contents of the file "DIGIN.DAT" in the section entitled: SUBPROGRAM "PREPRO"). In fact, one can define a "seal-like" problem as one where the flow passage is any generally-shaped rotor-to-housing flow passage which is bound by only two (inlet and exit) flow-permeable stations, with the seal segment (if any) being simply a tight-clearance part of the whole passage. Fig. 13 shows two examples of problems which fall under this particular category, namely the ATD Balance Piston Cavity, and the test rig by Guinzburg (1992) in which there was no primary-flow passage to start with.

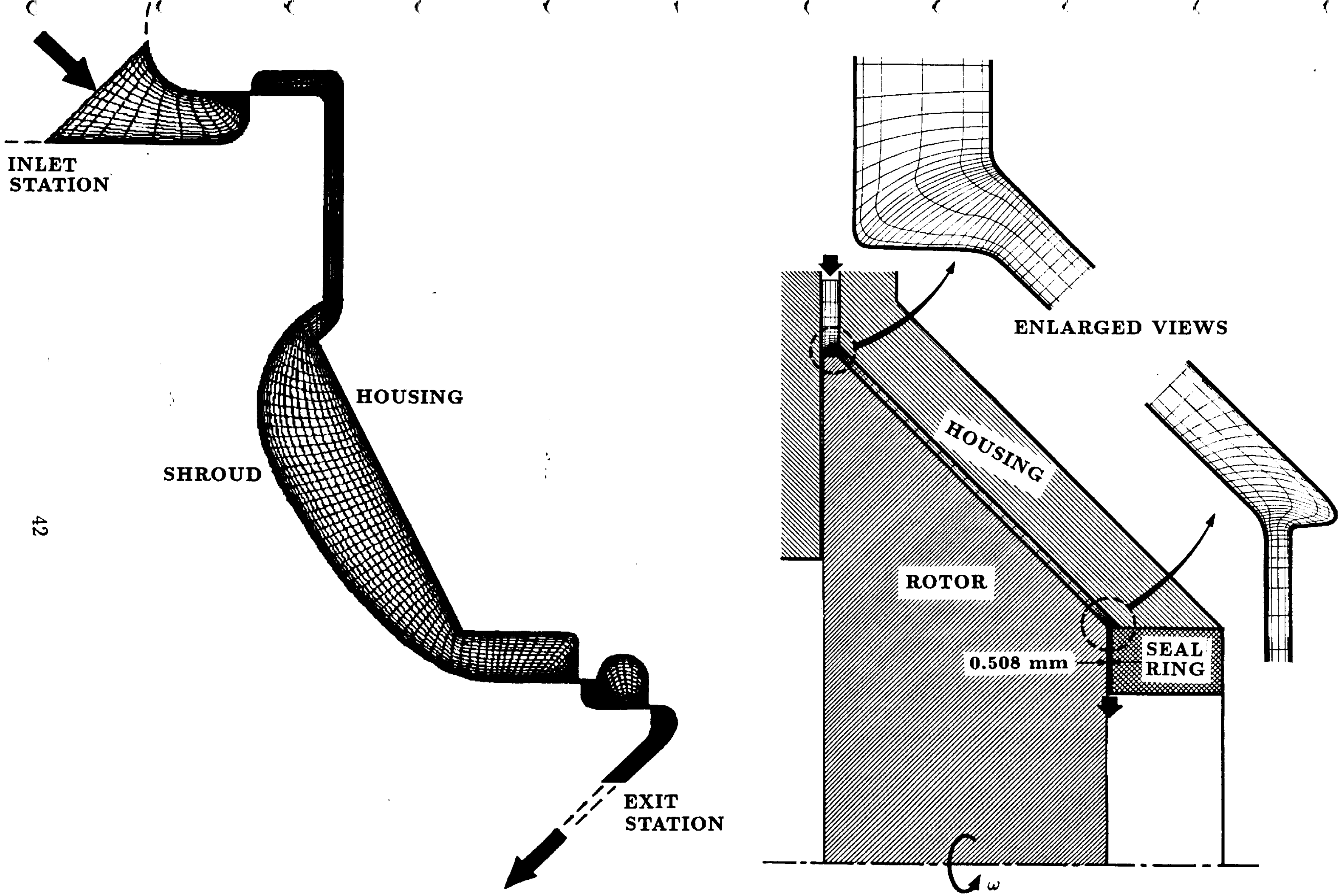


a) FIVE–CHAMBER LABYRINTH SEAL



b) SMOOTH ANNULAR SEAL

Fig. 12 Examples of the finite–element discretization models for seals



a) SSME ATD LOX BALANCE PISTON CAVITY

b) LEAKAGE–FLOW SIMULATION RIG OF GUINZBURG (1992)

Fig. 13 Eamples of "seal–like" problems to which the computational procedure is applicable

USING A PRE–EXISTING CENTERED–ROTOR FLOW FIELD TO DIRECTLY COMPUTE THE ROTORDYNAMIC COEFFICIENTS

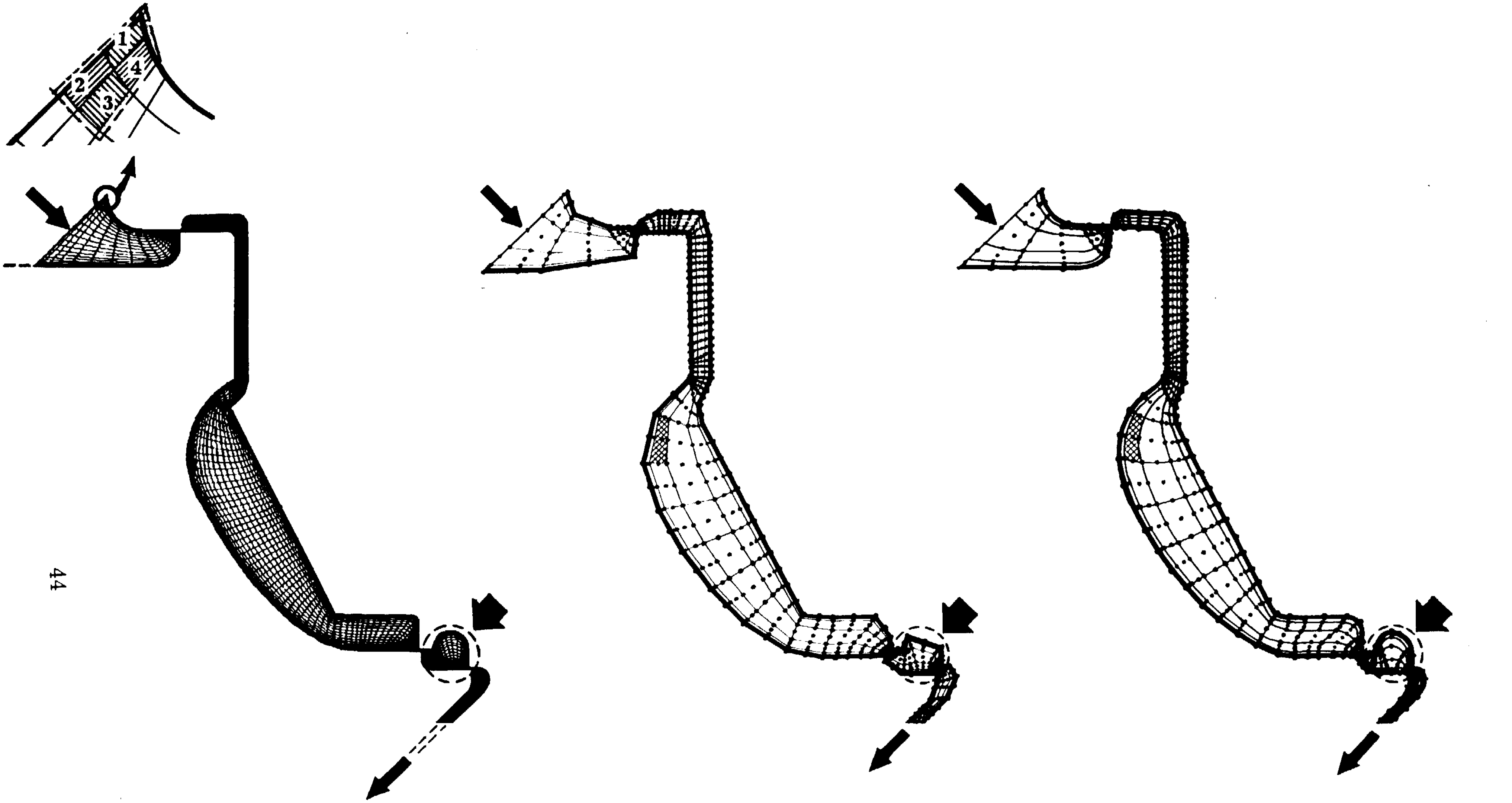
Remark:

This section contains a rather complex logic description that is intended to aid the user in developing a relatively short code, which will make it possible to use an existing axisymmetric flow solution as input to the perturbation analysis segment of the current computational procedure. In doing so, the user should understand that this "interface" code will have to create a set of files which are acceptable, in contents and structure, to subprogram "PERTURB", as explained earlier in this report. The logic described here is by no means the only path through which the user can develop this interface code. However, it may be reassuring to the user that the proposed logic was programmed by the author, and successfully tested for a special grid configuration. However, a general interface code, which would fully automate the process under such circumstances, is non–existing at this time. The task of producing a "reasonably" versatile interface code is among several extensions of the current study, and may very well be accomplished in the near future.

In some instances, a set of analytical or experimental data concerning the axisymmetric centered–rotor flow field is readily available to the user. The user's desire, in this case, is to execute only the perturbation analysis segment of the computational procedure, with the objective of computing the rotordynamic coefficients.

An example of this problem is shown in Fig. 14. The flow passage in this Figure is the ATD LOX Balance Piston Cavity which, as seen in the Figure, is isolated from the impeller primary passage. Such passage configuration makes the problem one of the "seal–like" category discussed at the end of the previous section [where it is emphasized that the variable "IPROB" (in the input file "DIGIN.DAT" of subprogram "PREPRO") should be set equal to zero]. The centered–impeller flow solution for this passage was previously determined, by the CFD division at NASA–MSFC using a finite–volume–based axisymmetric flow solver and the grid shown in Fig. 14a.

The user is warned that this section of the report would simply be meaningless unless he/she are totally familiar of the numbering pattern of the grid points in the existing (e.g. finite–volume) grid, and is reasonably capable of casting a set of given instructions in the form of an interface code. However, it is perhaps important to point out that the task of writing a code that is applicable to any irregular grid numbering pattern is (to the author) almost unthinkable. As for the instructions in this section, the phrases "existing grid" and "computational grid point" refer to the grid which was previously given to (or generated by) the user, while the phrases "finite–element model" and "node" are those associated with the new finite–element–like grid which the user will eventually create.



a) NASA–GENERATED FI:ITE–VOLUME GRID

b) EQUIVALENT FINITE–ELEMENT MODEL USING A CRUDE INTERFACE CODE (WRITTEN BY THE AUTHOR)

c) ILLUSTRATIVE PICTURE OF THE OUTCOME OF A MORE ACCURATE INTRFACE CODE UTILIZING THE ELEMENT CURVE–SIDIDNESS FEATURE (FUTURE TASK)

Fig. 14 Conversion of the ATD Balance Piston Cavity finite–volume grid into a finite– element model

Among the computational steps in this section, the process of converting the existing grid into a finite-element model is indeed the most challenging. Here, the user's objective is to produce two files which are similar, in construction and contents, to the files "DIGIT.OUT" (which is the output of subprogram "PREPRO") and "AXISYMM9.DAT" (which is the output of subprogram "AXISYMM"), using the existing grid-geometry and flow-solution files. Securing these two files is practically all that is needed to run the perturbation analysis code (subprogram PERTURB).

Among the possible ways of creating the required two files (above) is to write an interface code, particularly in the event where full automation of the computational procedure (with the axisymmetric flow solution being externally available every time) is desired. Let us begin with constructing the file that is equivalent to "DIGIT.OUT". The proposed "sequence of events" are as follows:

1) Identify, by nine of the existing-grid point numbers, nodal numbers'), and using four adjacent "cells" in the existing grid (Fig. 14), a sequence of bi-quadratic Lagrangian elements (Fig. 3), each composed of four corner nodes, four midside nodes and one interior node. In doing so, start at any of the element corner nodes, proceed around the element in an anti-clockwise direction picking up only the remaining three corner nodes, then follow the numbering scheme in Fig. 3 in determining the sequence of the midside nodes, and finally recognize the interior node (which, in this case, will be the grid points shared by all four "cells" of the existing grid). The process of creating these finite elements must begin at the flow inlet-station/housing corner of the computational domain, and progress towards the shroud. Once one "column" of finite elements is completed, you should move on to the next "column", again starting at the housing and proceeding to the shroud. In the event the number of "streamwise" grid lines (in the existing grid) is excessively large (producing, say, more than eight layers of elements between the housing and shroud lines), or in the case where you cannot construct the last (adjacent-to-shroud) element in each "column" of elements (due to the unavailability of four computational cells as the shroud line is reached), you may want to skip some of the existing grid lines prior to creating the new finite-element model. For the same two reasons (above), you may want to skip some of the cross-flow grid lines (in the existing grid) a priori. Note that the phrase 'skipping a grid line' is not intended to imply physical deletion, but is rather intended to mean 'ignoring' the the grid line, because you are supposed to always use the existing grid-point numbers which belong to the existing (e.g. finite-volume or finite-difference) grid in creating the finite elements, no matter whether you have skipped some of the grid lines or used the entire grid as is. Also, you should recall (in carrying out this step) that the larger the number of elements is, the larger the disk space and CPU time consumption will be.

2) Identify, by the grid-point number and a specified field-variable value, the grid points around the entire boundary of the computational domain. In creating this 'paired' set of data, you will again use the the actual grid-point numbers in the existing grid, but you will obviously skip those boundary points which exist on grid lines that you have already skipped. As for the boundary value of any field variable (velocity component

or pressure), you are supposed to refer to the axisymmetric flow solution (which you were given or have yourself created) to find the value of this variable. This process should be straight forward for the a velocity components (i.e. V_z, V_r or V_θ), where each of these components is non-dimensionalized using the reference velocity "V_{ref}", which is equal to the impeller tip speed (for the double-entry/double discharge shrouded-impeller problem), or the average through-flow velocity at the inlet station (for the seal or "seal-like" problem). However, should the problem under consideration be that of the double-entry double-discharge type (where two primary-flow passages are included in the computational domain (Fig. 1a), then you should not specify any velocity component at the far-upstream and far-downstream stations (where the flow enters and departs the pump stage), with the obvious exception of the four grid points on the solid walls (where the no-slip boundary conditions apply). The reason is that Neumann's (rather than Dirichlit's) boundary conditions are assigned (in the current computational model) at these two stations (recall that you are supposed to ultimately produce a file that is similar to the output file "DIGIT.OUT" of the subprogram "PREPRO", and that the contents of this file, as indicated earlier, do not include Neumann-condition boundary segments). In regards to the case of a seal (or seal-like) problem, you should not specify any boundary conditions at the flow discharge station (with the exception of the two grid points on the rotor and housing lines, where the no-slip boundary conditions apply) for the reason that the boundary conditions (across this station) are of the derivative (or Neumann) type, which is also internally imposed. This concludes the set of boundary conditions concerning the velocity components for the two problem categories under consideration. As for the pressure boundary conditions, the two pressure values at the middle points on the far-upstream and far-downstream (stage inlet and exit) stations are required for the shrouded-impeller problem type (Fig. 1a), and are assumed to be part of the existing flow solution file. Note that the pressure (in the current computational model) is non-dimensionalized as follows:

$$p^* = (p - p_{ref})/\rho {V_{ref}}^2$$

where p is the local pressure, p_{ref} is the reference pressure (at an arbitrary grid point in the computational domain), ρ is the fluid density, and V_{ref} is as defined above. In the pump stage problem (Fig. 1), the reference pressure is (conveniently) selected to be that at the middle grid point of the stage inlet station, a selection which leads to a zero nondimensional-pressure value at this node. The required value of non-dimensional pressure at the stage exit station is, in this case, equal to the static pressure differential across the stage, divided by the quantity $\rho {U_t}^2$. In the event the problem is that of the seal (or seal-like) type, you should specify the non-dimensional problem at only one grid point, chosen to be the middle point on the flow discharge station. Furthermore, since the choice of the reference pressure is arbitrary, p_{ref} in this case may be that at the same chosen point (meaning the discharge-station middle point). As a result, the required value of non-dimensional pressure is simply zero. An important remark here is that you should continue (up to this point) to use the original numbers of the grid points in the existing grid even if you had previously skipped some of the streamwise or cross–flow grid lines.

3) Examine the nine–noded finite elements which you have created in item 1 (above), considering only those elements which share sides with the far–downstream station. Identify (by three grid points) those element sides starting, each time, with the corner point that is on or closer to the housing, followed by the midside point and, finally, the other corner point on the same element side.

4) Now compare the data you have generated (using the preceding instructions) with Groups# 5 through 10 of the file "DIGIT.OUT" (which is the output file of subprogram "PREPRO") and you will find out that you have essentially produced all of these groups. You will also realize that you are faced with a major problem, namely the fact that you have so far (for the lack of any better choice) used the numbering system of the existing grid, and not that of the finite–element model in Fig. 8 (on three successive pages).

To remove this inconsistency, you may start by defining an integer 'mapping' array, say [MAP(I), I=1,M] where 'M' is the total number of the original computational points in the existing grid. The argument "I" in this array is the number of any grid point, and the value of "MAP(I)" is the node number (in the finite–element model) to which this grid point corresponds. In other words, the integers "I" (in the existing grid) and "MAP(I)" (in the finite element model) both define the same physical point except, of course, for the grid points (if any) which were previously decided not to be used in the finite–element model (recall that you may have already skipped some streamwise or cross–flow grid lines from the existing grid). The only special consideration which should be given to any skipped grid point, say point number "J" in the existing grid, is that MAP(J) is simply zero, meaning that this particular point is not 'mapped' to any node in the finite–element model.

Construction of the array "MAP" (above) is by no means an easy task. The process involves careful examination of the nodal numbers in Fig. 8 (noting, for instance, that corner nodes are numbered first, then midside nodes...etc.), comparing it with the numbering scheme in the existing grid, and finally deducing (in a functional relationship form) the node number (in the finite–element model) in terms of the grid point number (in the existing grid). Such a relationship would also include such parameters as the grid–point counts on each streamwise and cross–flow grid lines, and the location (defined by the streamwise and cross-flow grid lines) of this particular grid point in the existing grid. The process would obviously be more complicated in the case where the user had previously elected to eliminate (or skip) some of the grid lines in the existing grid.

Once the array "MAP" is constructed, the user is in a position to begin developing the interface code, with the ultimate goal of creating two output files which are similar to "DIGIT.OUT" and "AXISYMM9.DAT", respectively. The first segment of this code should either recognize (by reading–in) the array "MAP" (in the "unlikely" event that the user has already prepared the array by inspection), or be the code segment which creates the array.

The contents of the file "DIGIT.OUT" have already been discussed (in the section entitled "SUBPROGRAM PREPRO"). Group# 1 in this file (one line) can be left blank, and the user can easily prepare Group# 2 (two lines). As for Group# 3, the user is assumed, at this point, to be aware of how many nodes, finite elements,...etc. there are in the newly created finite-element model, and is therefore able to prepare this Group of variables as well. The three Groups (above) are supposed to be input variables to be fed to the interface code, and subsequently recorded on the code output file.

Groups# 4 and 5 in the file "DIGIT.OUT" deserve a special attention, for they are the first two groups where the previously-defined array "MAP" is to be used. Here, the user should go back to the interface code that he/she is developing, and add a new segment which reads-in and saves the array [Z(I), I=1,M], meaning the array of z-coordinates of the 'M' computational points in the existing grid. The next step in the interface code is to generate the array [Z(MAP(J)), J=1,M1 :: MAP(J) .GT. 0], where 'M1' is the total number of corner, midside and interior nodes in the finite-element note that "M1" is less than or equal to "M", depending on whether or not some grid lines have been skipped in the original grid, with the condition [MAP(J) .GT. 0] being the reason why the upper limit in this array is 'M1' and not 'M'. Careful examination of the two arrays [MAP(J), J=1,M1 such that: MAP(J) .GT. 0] and [Z(J), J=1,M1 :: MAP(J) .GT. 0] reveals that the former is the number of the node "J" in the finite-element model, and that the latter is the z-coordinate of this particular node. The last item in this step of the interface code is to re-sequence these two arrays, by re-sequencing the former 'integer' array in such a way that the first entry is 1 and the last entry is 'M1', and interchanging the entries of the second 'real' array accordingly. At this point, the re-sequenced second array of z-coordinates is precisely Group#4 in the file "DIGIT.OUT", and should therefore be recorded, on the output of the interface code, as such. The computational step just outlined can now be repeated to arrive at the r-coordinates of the finite-element nodes, which constitute Group# 5 in the file "DIGIT.OUT", and should also be recorded on the interface-code output file.

In order to prepare Group# 6, the user should first recall that the nine numbers associated with each finite element (Fig. 3) have already been identified (refer to the front part of this section), except that these are numbers associated with computational points in the existing grid. A new segment of the interface code can now be added, whereby these 'yet unmapped' numbers are read (or created) for each finite element. Assuming that these nine numbers, for a typical finite element, can be written as the sequence [L1,L2,...,L9], then all the interface code has to produce (and record on the output file) is the sequence [MAP(L1),MAP(L2),...,MAP(L9)]. The latter sequence is in terms of finite-element nodal numbers and is consistent with Group# 6 in the file "DIGIT.OUT".

Similar computational steps can be added to the interface code to produce Groups# 7 through 10. The user is reminded that the 'unmapped' forms of these groups, meaning the point number (in the existing grid) and the boundary value of the field variable, have already been discussed and should, at this point, be known sets of data, which can be fed to (or produced by) the interface code. Once again, the correct forms of these sets of data (in the finite element model) can be achieved by the interface code by simply replacing the 'existing-grid' number assigned to the specified-variable point (say "K") by the number "MAP(K)". These newly-created set of 'paired' data (i.e. node numbers and the corresponding boundary values of field variables) are to be recorded by the interface code in the output file and will, in the end, be consistent with Groups# 7 through 10 of the file "DIGIT.OUT".

As explained earlier, each line in Group# 11 should contain the three nodal numbers (in the finite-element model) associated with each element side which coincides with the far-downstream exit station. This set of data has either been prepared by the user (in terms of 'unmapped' points in the existing grid), or should now be produced by the interface code. To convert these grid-point numbers into finite-element nodes, the interface code should simply replace the three points in each line, say 'K1', 'K2' and 'K3', by 'MAP(K1)', 'MAP(K2) and 'MAP(K3)', respectively, and record the new set of integers on the output file.

No special handling is required for the remaining groups in the interface-code output file (namely those which are consistent with groups# 12 through 17 in the file "DIGIT.OUT"). The reason is that these groups contain data concerning the pump (or the seal) operating conditions with which the user should have no difficulty. Handling these groups is equally simple whether the user's intention is to feed them to the interface code (in which case they should by reproduced in the output file) or have the interface code create and subsequently record them on the output file.

At this point, the interface code (written according to the logic described above) is capable of producing an output file which, for all practical purposes, is an acceptable version of the file "DIGIT.OUT". This, as explained earlier, is one of the two files which are required should the current mode of running the computational procedure (with the axisymmetric flow analysis segment being by-passed) be desired.

The other output file that is required is the axisymmetric 'centered-rotor' flow solution file, which is similar to the output file "AXISYMM9.DAT" in the section entitled "SUB-PROGRAM AXISYMM". It is assumed that the available solution file, associated with the existing grid, contains the arrays of the velocity components (V_z, V_r and V_θ), the eddy kinematic viscosity 'ν_t' and the static pressure 'p'. A new segment should now be added to the interface code to insure identical dimensions of these arrays to those in the file "AX-ISYMM9.DAT". First, the velocity components must be non-dimensionalized using the impeller tip speed (in the case of a shrouded pump impeller), or the average through-flow inlet velocity (in the case of a seal or a seal-like problem). Next, the eddy viscosity should be in m^2/s. Finally, the static pressure should be non-dimensionalized in the manner explained earlier in this section. The remaining problem with these flow-solution arrays is that they are sequenced in accordance with the numbering scheme in the existing grid. In order to convert each array to its counterpart in the finite-element model, the procedure used in conjunction with converting the z-coordinates (described earlier in this section) can applied here as well. The only "warning" here concerns the process of converting (or mapping) the pressure values at the computational points (in the existing grid) to be associated with finite-element nodes instead. In writing this part of the interface code, the user should know (Baskharone and Hensel, 1991a), that the pressure is a degree of freedom (or a declared field variable) at the finite-element corner nodes only. Therefore, before a pressure value at any computational point "I" is dealt with, the integer "MAP(I)" (described earlier in this section) should first be examined. The point is then ignored in the event that this integer is larger than the total number of corner nodes 'M2' in the finite-element model.

At this point, the user-created interface code should be able to produce a file that is similar to the output file "AXISYMM9.DAT" of the subprogram "AXISYMM". The remaining instructions in this section have to do with running existing (and previously described) subprograms, prior to executing the perturbation analysis subprogram "PERTURB.

Three relatively minor steps are needed before the rotordynamic analysis can be performed:

1) The two new files, created by the interface code, should be renamed "PUMPLARGE.DAT" and "PUMP9LARGE.DAT". Of these, the first file should <u>also</u> be renamed "PUMPSMALL.DAT". The subprogram "INTERPOLATE" (described earlier in this report) can now be executed. Recall that this subprogram will prompt you for the desired number of computational planes in the circumferential direction. Depending on the available size of disk space, you should enter a 'preferably' odd number between 9 and 13. Referring to the description of subprogram "INTERPOLATE", the output of this step is the file "PUMP9SMALL.DAT".

2) Execute the subprogram "ROTSUR" (discussed earlier in this report), and expect to be prompted for the cross-flow station numbers (in the finite-element model) between which you desire the shroud (in the double-entry/double-departure shrouded-impeller pump problem) or simply the rotor (in the seal or seal-like problem) to be unwrapped, with the nodal coordinates and the corresponding pressure perturbation values ultimately produced. The output of this step, by reference to the description of the subprogram "ROTSUR" is the file "ROTSUR.DAT".

3) Prepare the file "WHIRLF.DAT", which contains only one real variable, namely the desired whirl frequency/shaft speed (Ω/ω) ratio at which the fluid-exerted forces are required. As explained earlier, this subprogram's output file, namely "F3DX.DAT" contains a listing of the fluid-induced forces and moments at the desired whirl frequency ratio. The remaining contents of this output file, as mentioned earlier, can be

used (if desired) as input to practically any contour–plotting software to produce the pressure perturbation contours over the unwrapped surface of the shroud (or the rotor) corresponding to this particular whirl frequency ratio. Repetitive execution of the subprogram "PERTURB" over a sufficient range of whirl frequency ratios (usually between -2.0 and $+2.0$) will provide the input data for the post–processor "LEASTSQ" (described earlier in this report) which yields the direct and cross–coupled rotordynamic coefficients.

REFERENCES

Baldwin, B.S. and Lomax, H., 1978, "Thin Layer Approximation and Algebraic Model for Separated Turbulent Flows," AIAA Paper No. 78-257.

Baskharone, E.A. and Hensel, S.J., 1989, "A New Model for Leakage Prediction in Shrouded-Impeller Turbopumps," ASME J. of Fluids Engineering, Vol. 111, pp. 118-123.

Baskharone, E.A. and Hensel, S.J. , 1991a, "A Finite-Element Perturbation Approach to Fluid/Rotor Interaction in Turbomachinery Elements. Part 1: Theory," ASME J. of Fluids Engineering, Vol. 113, No. 3, pp. 353-361.

Baskharone, E.A. and Hensel, S.J. , 1991b, "A Finite-Element Perturbation Approach to Fluid/Rotor Interaction in Turbomachinery Elements. Part 1: Application," ASME J. of Fluids Engineering, Vol. 113, No. 3, pp. 362-367.

Bolleter, U., Leibundgut, E. and Sturchler, R., 1989, "Hydraulic Interaction and Excitation Forces of High Head Pump Impellers," presented at the Third Joint ASCE/ASME Mechanics Conference, University of California, La Jolla, California.

Guinzburg, A., 1992, "Rotordynamic Forces Generated by Discharge-to-Suction Leakage Flow in Centrifugal Pumps," Ph.D. Dissertation, California Institute of Technology.

APPENDIX

LISTINGS OF SAMPLE INPUT AND OUTPUT FILES

This section contains full and partial listings of the key input and output files of the computational procedure. All of these files concern the Sulzer Bros.' face-seal pump configuration (Bolleter et al., 1989). Two finite-element models of the shroud-to-housing leakage passage in this pump were generated at different computational steps, and are shown in Figs. 3&9. Of the two models in these Figures, the first is the outcome of the initial run of the pre-processor "PREPRO", and was used in the axisymmetric flow analysis corresponding to the centered-impeller operation mode. The finite-element model in Fig. 9, on the other hand, was created (by the same pre-processor) to be an input to subprogram "INTERPOLATE" in a successful attempt to reduce the number of nodes to the point where the available disk space was sufficient to execute the perturbation analysis segment (namely subprogram "PERTURB") of the computational procedure.

The sequence of files in this Appendix is consistent with the order at which they were generated. Moreover, no distinguishment is made between input and output files, with the exception that the latter group are, for the major part, partially listed. The reason is that most of the output files are prohibitively lengthy. Referring to the "shortened" output files in this Appendix, the reader will notice that each segment of compatible lines (usually contains the entries of a long array) is cut short, by listing only the the segment's first and last three lines. Wherever this is the case, a column of four astrisks is inserted in the middle of the page, with the intention of marking the positions (in the output file) where the unrecorded lines were extracted.

File "DIGIT.DAT"
(Full Listing)

```
    4
0.000 14.50
0.000 10.00
0.000 6.000
0.000 0.000
    4
0.950 14.50
1.200 10.00
2.500 5.000
4.500 0.000
    4
1.900 14.50
3.000 9.500
6.000 4.000
8.500 0.000
    4
2.800 14.50
4.000 10.00
7.000 5.000
12.10 0.000
    4
3.700 14.50
5.000 10.00
9.680 4.000
17.20 0.000
    4
4.750 14.50
7.500 8.000
12.00 3.400
18.80 0.200
    4
5.350 14.50
7.000 10.00
11.60 5.000
19.00 1.500
    4
6.000 14.50
8.000 10.00
13.00 6.700
19.00 6.000
    4
6.600 14.50
8.550 10.50
12.50 9.250
19.00 11.50
    4
7.200 14.50
8.500 11.200
12.50 10.00
18.80 12.95
    4
7.700 14.50
8.970 11.47
11.00 10.53
17.30 13.15
    4
8.05 14.50
9.17 12.00
11.95 11.10
15.45 13.15
    4
8.37 14.50
9.30 12.35
11.95 11.45
13.95 13.15
    4
8.68 14.50
9.40 12.60
11.75 11.85
12.93 13.15
    4
8.95 14.50
9.40 13.05
11.00 12.25
12.10 13.15
    4
9.17 14.50
```

```
9.55 13.25
10.75 12.65
11.40 13.15
   4
9.36 14.50
9.65 13.50
10.60 12.95
10.90 13.20
   4
9.50 14.50
9.77 13.60
10.25 13.33
10.48 13.38
   4
9.64 14.50
9.85 13.80
10.00 13.65
10.25 13.57
   4
9.75 14.51
9.85 14.10
10.00 13.92
10.19 13.78
   4
9.82 14.57
9.85 14.35
9.97 14.15
10.11 14.05
   4
9.88 14.60
9.95 14.47
10.03 14.35
10.10 14.27
   4
9.92 14.67
9.95 14.60
10.03 14.53
10.10 14.50
   4
9.96 14.75
10.007 14.733
10.053 14.717
10.10 14.70
   4
9.99 14.87
10.027 14.863
10.063 14.857
10.10 14.85
   4
10.00 15.00
10.033 15.00
10.067 15.00
10.10 15.00
   4
10.00 15.30
10.033 15.30
10.067 15.30
10.10 15.30
   4
10.00 15.90
10.033 15.90
10.067 15.90
10.10 15.90
   4
10.00 17.00
10.033 17.00
10.067 17.00
10.10 17.00
   4
10.00 18.50
10.033 18.50
10.067 18.50
10.10 18.50
   4
10.00 20.45
10.033 20.45
10.067 20.45
10.10 20.45
   4
10.00 21.65
```

```
10.033 21.65
10.067 21.65
10.10 21.65
   4
10.00 22.30
10.033 22.30
10.067 22.30
10.10 22.30
   4
10.00 22.48
10.033 22.48
10.067 22.48
10.10 22.48
   4
10.00 22.58
10.037 22.570
10.073 22.560
10.11 22.550
   4
10.00 22.730
10.05 22.720
10.10 22.690
10.15 22.660
   4
10.00 22.92
10.15 22.88
10.23 22.82
10.30 22.76
   4
10.00 23.17
10.19 23.10
10.35 23.00
10.51 22.79
   4
10.01 23.50
10.25 23.30
10.59 23.00
10.73 22.80
   4
10.05 23.815
10.35 23.520
10.70 23.085
10.90 22.80
   4
10.20 24.245
10.50 23.87
10.85 23.28
11.075 22.80
   4
10.50 24.55
10.85 23.99
11.125 23.375
11.305 22.80
   4
10.90 24.70
11.15 24.213
11.40 23.55
11.575 22.80
   4
11.20 24.740
11.45 24.2315
11.70 23.55
11.845 22.80
   4
11.50 24.748
11.75 24.25
12.00 23.55
12.115 22.80
   4
11.935 24.75
12.175 24.225
12.40 23.525
12.5075 22.80
   4
12.37 24.75
12.60 24.20
12.80 23.50
12.90 22.80
   4
12.835 24.75
```

```
13.05 24.1875
13.25 23.50
13.35 22.80
   4
13.30 24.75
13.50 24.175
13.70 23.50
13.80 22.80
   4
13.86 24.75
14.075 24.1875
14.2875 23.50
14.425 22.80
   4
14.42 24.75
14.65 24.20
14.875 23.50
15.05 22.80
   4
15.084 24.75
15.325 24.2025
15.5625 23.4875
15.725 22.80
   4
15.748 24.75
16.00 24.205
16.25 23.475
16.40 22.80
   4
16.311 24.75
16.575 24.2025
16.80 23.4815
16.9425 22.80
   4
16.874 24.75
17.150 24.20
17.350 23.488
17.485 22.80
   4
17.437 24.75
17.725 24.20
17.90 23.494
18.0275 22.80
   4
18.00 24.75
18.30 24.20
18.45 23.50
18.57 22.80
   4
18.54 24.75
18.765 24.10
18.905 23.43
19.01 22.80
   4
19.08 24.75
19.23 24.00
19.36 23.36
19.45 22.80
   4
19.375 24.75
19.615 24.00
19.810 23.355
20.075 22.80
   4
19.67 24.7505
20.00 24.00
20.26 23.35
20.70 22.805
   4
19.97 24.78
20.25 24.31
20.67 23.65
21.10 22.82
   4
20.25 24.9705
20.57 24.50
21.00 24.00
21.45 23.1108
   4
20.62 25.2962
```

21.00 24.90
21.50 24.11
21.80 23.4497
4
21.20 25.8068
21.57 25.10
21.87 24.50
22.30 23.9340
4
21.45 25.98
21.90 25.10
22.19 24.56
22.41 24.00
4
21.92 26.01
22.25 25.50
22.57 24.80
22.80 24.01
4
22.60 26.00
22.95 25.32
23.14 24.80
23.42 24.02
4
23.38 26.01
23.60 25.50
24.00 24.80
24.33 24.03
4
23.80 26.02
24.00 25.60
24.30 25.00
24.62 24.04
4
23.93 26.2058
24.28 25.90
24.77 25.20
25.00 24.3365
4
24.08 26.4865
24.58 26.40
25.00 25.92
25.47 25.2134
4
24.75 27.7401
25.25 27.50
25.75 27.10
26.18 26.5380
4
25.425 29.0031
25.96 28.85
26.47 28.50
26.94 27.95595
4
26.10 30.2661
26.67 30.20
27.19 29.90
27.70 29.3739
4
26.89 31.74425
27.63 31.60
27.995 31.36
28.525 30.9131
4
27.68 33.2224
28.59 33.00
28.80 32.82
29.35 32.4523
4
28.11 34.0270
28.65 34.00
29.40 33.60
29.80 33.2919
4
28.46 34.6819
29.00 34.45
29.50 34.10
30.02 33.7023
4
28.65 34.98

```
29.15 34.60
29.70 34.30
30.20 33.98
  4
29.15 34.99
29.50 34.71
30.00 34.40
30.37 34.00
  4
30.10 35.00
30.33 34.63
30.55 34.30
30.75 34.00
  4
30.78 35.02
31.00 34.78
31.25 34.30
31.53 34.00
  4
30.80 35.75
31.40 35.30
32.00 34.70
32.40 34.00
  4
30.80 36.10
33.00 35.45
34.00 35.00
36.00 34.00
  4
30.80 36.60
34.50 36.00
37.75 35.00
39.00 34.00
  4
30.80 37.10
34.00 36.97
38.00 35.75
40.48 34.02
  4
30.80 37.65
34.00 37.50
38.00 36.60
40.50 35.95
  4
30.80 38.05
35.00 37.91
38.00 37.28
40.50 36.60
  4
30.80 38.60
35.00 38.50
38.00 38.17
40.50 37.90
  4
30.80 39.10
34.00 39.10
38.00 39.10
40.50 39.10
  4
30.80 39.80
34.00 39.80
38.00 39.80
40.50 39.80
```

File "DIGIN.DAT"
(Full Listing)

```
     1      1
    92      7    26
  2.25       0.200
  0.070000  0.1933500
  0.40        0.44950
  0.000000799          8020000.000          1.00                    997.4
     6     10      1
    83     87      1
 0.5         7.3800     0.00        0.0000
 0.5         0.0        4.2200      20.742
```

File "DIGIT.OUT"

(Partial Listing)

```
1000     0     1 0.00000E+00
   1     0     0     0     1     1    92     7    10    83    26     0
   0     6    10    83    87
2379   644  5402   546     0   366   366   366     2     6
 0.000000  0.000000  0.000000  0.000000  0.000000  0.000000  0.000000  0.002944
 0.002988  0.003233  0.005705  0.010441  0.012854  0.013947  0.005889  0.006357
 0.007615  0.013144  0.021200  0.024767  0.026344  0.008678  0.009368  0.011188
                                       *
                                       *
                                       *
                                       *
 0.124616  0.096375  0.099356  0.106042  0.115205  0.121750  0.124638  0.096361
 0.099301  0.105910  0.115056  0.121667  0.124612  0.096361  0.099299  0.105911
 0.115065  0.121677  0.124615
 0.114939  0.112236  0.106154  0.092470  0.078785  0.072703  0.070000  0.114939
 0.112122  0.105802  0.091844  0.078416  0.072583  0.070000  0.114939  0.111944
 0.105278  0.091262  0.078269  0.072544  0.070000  0.114939  0.111709  0.104534
                                       *
                                       *
                                       *
                                       *
 0.182689  0.188786  0.188801  0.188597  0.187409  0.186152  0.185600  0.190407
 0.190407  0.190322  0.189908  0.189524  0.189371  0.192265  0.192265  0.192266
 0.192265  0.192265  0.192265
   8     1     2     9  1197   645  1288   651  1834
   9     2     3    10  1288   646  1379   652  1835
  10     3     4    11  1379   647  1470   653  1836
                                       *
                                       *
                                       *
                                       *
 641   634   635   642  1560  1188  1651  1194  2377
 642   635   636   643  1651  1189  1742  1195  2378
 643   636   637   644  1742  1190  1833  1196  2379
   1  0.000000
   8  0.000000
  15  0.000000
                                       *
                                       *
                                       *
                                       *
1831  0.000000
1832  0.000000
1833  0.000000
   1  0.000000
   8  0.000000
  15  0.000000
                                       *
                                       *
                                       *
                                       *
1831  0.000000
1832  0.000000
1833  0.000000
   1  0.000000
   8  0.000000
  15  0.000000
                                       *
                                       *
                                       *
                                       *
1831  0.000000
1832  0.000000
1833  0.000000
   4  0.000000
 640  0.449500
 638  1191   639
 639  1192   640
 640  1193   641
 641  1194   642
 642  1195   643
 643  1196   644
 0.010000
 0.175374
      8020000.000000       0.000000799000      997.400024414063
-0.774597  0.000000  0.774597
 0.555555  0.888889  0.555555
 0.000000  1.000000
 0.400000  0.400000
```

File "AXISYMM9.DAT"

(Partial Listing)

```
  855 0.18503E-01 0.80200E+07 LEAKAGE FLOW RATE= 0.42240E-02 CMS CONVERGED SOLUTION
0.000000E+000.839453E-010.131825E+000.201891E+000.218784E+000.159184E+00
0.000000E+000.000000E+000.854230E-010.133843E+000.205428E+000.216876E+00
0.155726E+000.000000E+000.000000E+000.876558E-010.136696E+000.210056E+00
                                       *
                                       *
                                       *
                                       *
0.157232E-01-.424327E-02-.133251E-02-.152011E-020.396249E-020.747748E-01
0.240923E-010.121801E-020.117162E-020.130604E-02-.354809E-030.792483E-01
0.292757E-010.240765E-020.955427E-03
0.000000E+00-.135336E-03-.436121E-03-.959303E-040.589525E-04-.240593E-03
0.000000E+000.000000E+00-.222938E-03-.568490E-03-.173540E-030.891402E-04
0.457606E-030.000000E+000.000000E+00-.344833E-03-.113557E-02-.455175E-03
                                       *
                                       *
                                       *
0.157232E-01-.424327E-02-.133251E-02-.152011E-020.396249E-020.747748E-01
0.172156E+000.844219E-010.383396E-02-.590163E-01-.411550E-010.141707E+00
0.182177E+000.853544E-010.857574E-03-.555612E-01-.510621E-010.121912E+00
0.192023E+000.849982E-010.777247E-02
0.000000E+00-.940668E-09-.126832E-08-.166704E-08-.208887E-08-.221048E-08
0.000000E+000.000000E+00-.951530E-09-.127976E-08-.168072E-08-.209534E-08
-.194911E-080.000000E+000.000000E+00-.964857E-09-.129145E-08-.177507E-08
                                       *
                                       *
                                       *
                                       *
0.607300E+000.415312E+000.208195E+000.190773E+000.351164E+000.578908E+00
0.592934E+000.412206E+000.224705E+000.186409E+000.345775E+000.532566E+00
0.581242E+000.404028E+000.212431E+00
0.000000E+000.425822E-030.228517E-020.279130E-020.144927E-020.578414E-03
0.000000E+000.000000E+000.440839E-030.239278E-020.252006E-020.144613E-02
0.561475E-030.000000E+000.000000E+000.472528E-030.254649E-020.228003E-02
                                       *
                                       *
                                       *
                                       *
0.432402E-020.402019E-020.727553E-030.690350E-030.215704E-020.177799E-02
0.433899E-020.427017E-020.697630E-030.692652E-030.311081E-020.309429E-02
0.145214E-020.150909E-020.671888E-03
-.989694E-05-.111249E-04-.256632E-070.000000E+000.619400E-040.210523E-04
0.197523E-040.742905E-040.656670E-040.342418E-04-.958612E-04-.488661E-03
-.424959E-03-.440035E-030.134685E-030.126881E-030.128978E-03-.597115E-03
                                       *
                                       *
                                       *
                                       *
0.445718E+000.445813E+000.445923E+000.439782E+000.434732E+000.435093E+00
0.434964E+000.447590E+000.447047E+000.449500E+000.438559E+000.436699E+00
0.436443E+000.436550E+00
0.928034E+010.178573E+010.888229E+000.991191E+000.563319E+000.238259E+01
0.189298E+020.911259E+010.169684E+010.859712E+000.946378E+000.612467E+00
0.238340E+010.196068E+020.883228E+010.160111E+010.818437E+000.902883E+00
0.183011E+010.113925E+020.288126E+020.269117E+020.847080E+010.416891E+01
0.217063E+010.106300E+020.269754E+020.270676E+020.696316E+010.507591E+01
0.240658E+010.107573E+020.263111E+02
```

File "DIGITSMALL.DAT"
(Full Listing)

```
   4
0.000 14.50
0.000 10.00
0.000 6.000
0.000 0.000
   4
1.900 14.50
3.000 9.500
6.000 4.000
8.500 0.000
   4
3.700 14.50
5.000 10.00
9.680 4.000
17.20 0.000
   4
4.750 14.50
7.500 8.000
12.00 3.400
18.80 0.200
   4
5.350 14.50
7.000 10.00
11.60 5.000
19.00 1.500
   4
6.000 14.50
8.000 10.00
13.00 6.700
19.00 6.000
   4
6.600 14.50
8.550 10.50
12.50 9.250
19.00 11.50
   4
7.200 14.50
8.500 11.200
12.50 10.00
18.80 12.95
   4
7.700 14.50
8.970 11.47
11.00 10.53
17.30 13.15
   4
9.17 14.50
9.55 13.25
10.75 12.65
11.40 13.15
   4
9.50 14.50
9.77 13.60
10.25 13.33
10.48 13.38
   4
9.82 14.57
9.85 14.35
9.97 14.15
10.11 14.05
   4
9.92 14.67
9.95 14.60
10.03 14.53
10.10 14.50
   4
10.00 15.00
10.033 15.00
10.067 15.00
10.10 15.00
   4
10.00 15.90
10.033 15.90
10.067 15.90
10.10 15.90
   4
10.00 18.50
```

```
10.033 18.50
10.067 18.50
10.10 18.50
   4
10.00 21.65
10.033 21.65
10.067 21.65
10.10 21.65
   4
10.00 22.30
10.033 22.30
10.067 22.30
10.10 22.30
   4
10.00 22.58
10.037 22.570
10.073 22.560
10.11 22.550
   4
10.00 22.730
10.05 22.720
10.10 22.690
10.15 22.660
   4
10.00 22.92
10.15 22.88
10.23 22.82
10.30 22.76
   4
10.01 23.50
10.25 23.30
10.59 23.00
10.73 22.80
   4
10.35 24.42
10.70 24.00
11.00 23.37
11.20 22.80
   4
11.20 24.73
11.45 24.00
11.58 23.50
11.71 22.80
   4
15.85 24.75
16.00 24.30
16.22 23.50
16.40 22.80
   4
19.67 24.75
20.00 24.00
20.26 23.35
20.70 22.80
   4
19.97 24.78
20.25 24.31
20.67 23.65
21.10 22.82
   4
20.25 24.9705
20.57 24.50
21.00 24.00
21.45 23.1108
   4
21.20 25.8068
21.57 25.10
21.87 24.50
22.30 23.9340
   4
21.45 25.98
21.90 25.10
22.19 24.56
22.41 24.00
   4
21.92 26.01
22.25 25.50
22.57 24.80
22.80 24.01
   4
23.38 26.01
```

23.60	25.50
24.00	24.80
24.33	24.03
4	
23.80	26.02
24.00	25.60
24.30	25.00
24.62	24.04
4	
23.93	26.2058
24.28	25.90
24.77	25.20
25.00	24.3365
4	
24.75	27.7401
25.25	27.50
25.75	27.10
26.18	26.5380
4	
26.10	30.2661
26.67	30.20
27.19	29.90
27.70	29.3739
4	
27.68	33.2224
28.59	33.00
28.80	32.82
29.35	32.4523
4	
28.46	34.6819
29.00	34.45
29.50	34.10
30.02	33.7023
4	
28.65	34.98
29.15	34.60
29.70	34.30
30.20	33.98
4	
29.15	34.99
29.50	34.71
30.00	34.40
30.37	34.00
4	
30.78	35.02
31.00	34.78
31.25	34.30
31.53	34.00
4	
30.80	36.10
33.00	35.45
34.00	35.00
36.00	34.00
4	
30.80	36.60
34.50	36.00
37.75	35.00
39.00	34.00
4	
30.80	37.10
34.00	36.97
38.00	35.75
40.48	34.02
4	
30.80	37.65
34.00	37.50
38.00	36.60
40.50	35.95
4	
30.80	39.80
34.00	39.80
38.00	39.80
40.50	39.80

File "DIGINSMALL.DAT"

(Full Listing)

```
   1     1
  46     5    16
 3.50      0.200
 0.070000  0.1933500
 0.40        0.44950
 0.000000799            8020000.000           1.00                 997.4
   4     8     1
  41    44     1
0.5        7.3800      0.00         0.0000
0.5        0.0         4.2200       20.742
```

File "PUMPSMALL.DAT"

(Partial Listing)

```
1000       0      1  0.00000E+00
   1       0      0      0      1      1     46      5      8     41     16      0
   0       4      8     41     44
 819     230   1868    180      0    182    182    182      2      4
0.000000  0.000000  0.000000  0.000000  0.000000  0.005889  0.006784  0.013144
0.023429  0.026344  0.011467  0.013151  0.025579  0.046671  0.053307  0.014721
0.017073  0.029988  0.051475  0.058266  0.016581  0.018590  0.031729  0.052491
                                  *
                                  *
                                  *
                                  *
0.098444  0.101881  0.104172  0.096668  0.102033  0.110249  0.114998  0.097152
0.104778  0.116029  0.122232  0.097226  0.105166  0.117166  0.124470  0.097061
0.104250  0.115726  0.123572
0.114939  0.109946  0.092470  0.074993  0.070000  0.114939  0.109417  0.091262
0.074699  0.070000  0.114939  0.107946  0.086312  0.072863  0.070000  0.114939
0.108028  0.086457  0.073253  0.070620  0.114939  0.108388  0.088751  0.077100
                                  *
                                  *
                                  *
                                  *
0.178864  0.176575  0.175479  0.182439  0.181238  0.178931  0.176249  0.184068
0.183302  0.180161  0.176165  0.185829  0.185389  0.182222  0.178706  0.190011
0.189720  0.188738  0.188021
    6      1      2      7    415    231    460    235    640
    7      2      3      8    460    232    505    236    641
    8      3      4      9    505    233    550    237    642
                                  *
                                  *
                                  *
                                  *
  227    222    223    228    504    408    549    412    817
  228    223    224    229    549    409    594    413    818
  229    224    225    230    594    410    639    414    819
    1  0.000000
    6  0.000000
   11  0.000000
                                  *
                                  *
                                  *
                                  *
  230  0.000000
  638  0.000000
  639  0.000000
    1  0.000000
    6  0.000000
   11  0.000000
                                  *
                                  *
                                  *
                                  *
  230  0.000000
  638  0.000000
  639  0.000000
    1  0.000000
    6  0.000000
   11  0.000000
                                  *
                                  *
                                  *
                                  *
  230  0.000000
  638  0.000000
  639  0.000000
    3  0.000000
  227  0.449500
  226    411    227
  227    412    228
  228    413    229
  229    414    230
 0.010000
 0.175374
     8020000.000000     0.000000799000     997.400024414063
-0.774597  0.000000  0.774597
 0.555555  0.888889  0.555555
 0.000000  1.000000
 0.400000  0.400000
```

File "PUMP9SMALL.DAT"
(Partial Listing)

```
855 0.18503E-01 0.80200E+07    11
.000000E+000.105837E+000.201891E+000.190275E+000.000000E+000.000000E+00
.110038E+000.210056E+000.184281E+000.000000E+000.000000E+000.120903E+00
.237432E+000.140609E+000.000000E+000.000000E+000.119926E+000.238818E+00
                              *
                              *
                              *
                              *
.243190E-010.724204E-02-.106759E-020.632157E-040.380830E-01-.452956E-02
.628825E-020.682700E-030.491527E-01-.364174E-02-.524416E-02-.215942E-02
.552642E-010.186889E-01-.452975E-03
.000000E+00-.199338E-03-.959303E-04-.972435E-040.000000E+000.000000E+00
.543501E-03-.455175E-030.177269E-020.000000E+000.000000E+00-.162918E-02
.177828E-020.516294E-020.000000E+000.000000E+00-.313642E-020.975376E-03
                              *
                              *
                              *
                              *
.105322E+000.135345E+000.137795E+00-.658474E-010.124442E+000.141494E+00
.100937E+00-.633179E-010.128601E+000.140937E+000.381635E-01-.628926E-01
.910495E-010.171935E+000.349465E-01
.000000E+00-.110150E-08-.166704E-08-.212777E-080.000000E+000.000000E+00
.113093E-08-.177507E-080.521654E-080.000000E+000.000000E+00-.109460E-08
.917256E-08-.291071E-060.000000E+000.000000E+00-.365500E-08-.463073E-07
                              *
                              *
                              *
                              *
.675934E+000.635038E+000.651988E+000.278828E+000.646611E+000.623632E+00
.514191E+000.268441E+000.628385E+000.595286E+000.312574E+000.248578E+00
.532197E+000.585609E+000.318769E+00
.000000E+000.124769E-020.279130E-020.157224E-020.000000E+000.000000E+00
.141032E-020.228003E-020.439844E-040.000000E+000.000000E+000.192137E-02
.120349E-020.921184E-030.000000E+000.000000E+000.180613E-020.112732E-07
                              *
                              *
                              *
                              *
.791503E-030.627407E-030.295648E-020.145138E-020.656280E-030.228921E-02
.390449E-020.145884E-020.134703E-020.179019E-020.120335E-020.869218E-03
.210425E-020.467711E-020.138619E-02
.989694E-05-.694580E-050.000000E+000.364474E-040.197523E-040.134685E-03
.126574E-03-.597115E-03-.161894E-02-.204972E-020.211446E-030.149878E-03
.204547E-02-.217919E-02-.127254E-020.200484E-030.144975E-03-.246206E-02
                              *
                              *
                              *
                              *
.439543E+000.429804E+000.421230E+000.418593E+000.441510E+000.441564E+00
.431732E+000.426585E+000.425663E+000.447590E+000.447971E+000.438559E+00
.436539E+000.436550E+00
```

File "ROTSUR.DAT"
(Partial Listing)

```
 348    12
  65   0.000000E+00   0.912045E+00
 295   0.000000E+00   0.846392E+00
 525   0.000000E+00   0.780739E+00
 755   0.000000E+00   0.715086E+00
 985   0.000000E+00   0.649433E+00
1215   0.000000E+00   0.583780E+00
1445   0.000000E+00   0.518127E+00
1675   0.000000E+00   0.452474E+00
1905   0.000000E+00   0.386821E+00
2135   0.000000E+00   0.321168E+00
2365   0.000000E+00   0.255515E+00
  65   0.000000E+00   0.189862E+00
                                     *
                                     *
                                     *
                                     *
 205   0.111254E+00   0.110191E+01
 435   0.111254E+00   0.100173E+01
 665   0.111254E+00   0.901560E+00
 895   0.111254E+00   0.801387E+00
1125   0.111254E+00   0.701214E+00
1355   0.111254E+00   0.601040E+00
1585   0.111254E+00   0.500867E+00
1815   0.111254E+00   0.400694E+00
2045   0.111254E+00   0.300520E+00
2275   0.111254E+00   0.200347E+00
2505   0.111254E+00   0.100174E+00
 205   0.111254E+00   0.000000E+00
```

File "WHIRLF.DAT"
(Full Listing)

1.2

File "F3DX.DAT"
(Partial Listing)

```
    FT=-0.548605E+06     FR=-0.247399E+06     FZ= 0.438388E+03
    MX=-0.403483E+05     MY=-0.818087E+05
IF YOU WANT TO VIEW THE PRESSURE PERTURBATION CONTOURS OVER THE
UNWRAPPED ROTOR SURFACE, YOU SHOULD FIRST PREPARE THIS FILE FOR THE
POST-PROCESSOR BY DELETING THE ABOVE SEGMENT (INCLUDING THIS LINE)
 348    12
 0.00000E+00  0.91205E+00  0.97382E+06
 0.00000E+00  0.84639E+00 -0.34440E+08
 0.00000E+00  0.78074E+00 -0.58786E+08
 0.00000E+00  0.71509E+00 -0.64446E+08
 0.00000E+00  0.64943E+00 -0.50102E+08
 0.00000E+00  0.58378E+00 -0.19482E+08
 0.00000E+00  0.51813E+00  0.17337E+08
 0.00000E+00  0.45247E+00  0.48781E+08
 0.00000E+00  0.38682E+00  0.64263E+08
 0.00000E+00  0.32117E+00  0.59574E+08
 0.00000E+00  0.25551E+00  0.36218E+08
 0.00000E+00  0.18986E+00  0.97382E+06
                                      *
                                      *
                                      *
                                      *
 0.11125E+00  0.11019E+01 -0.28296E+07
 0.11125E+00  0.10017E+01 -0.74977E+06
 0.11125E+00  0.90156E+00  0.15656E+07
 0.11125E+00  0.80139E+00  0.33854E+07
 0.11125E+00  0.70121E+00  0.41244E+07
 0.11125E+00  0.60104E+00  0.35553E+07
 0.11125E+00  0.50087E+00  0.18522E+07
 0.11125E+00  0.40069E+00 -0.43188E+06
 0.11125E+00  0.30052E+00 -0.25879E+07
 0.11125E+00  0.20035E+00 -0.39223E+07
 0.11125E+00  0.10017E+00 -0.40111E+07
 0.11125E+00  0.00000E+00 -0.28296E+07
```