

AD-A236 720



2

NUMERICAL SIMULATION OF FLUID FLOW IN A SIMPLE ROTOR/STATOR PAIR

by

JOSEPH JOHN MCVEIGH, JR.

B.S., United States Military Academy
(1978)

DTIC
ELECTE
JUN 13 1991
S D D

SUBMITTED TO THE DEPARTMENT OF OCEAN ENGINEERING
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREES OF

MASTER OF SCIENCE IN NAVAL ARCHITECTURE/MARINE ENGINEERING

and

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
June, 1991

© Joseph John McVeigh, Jr., 1991. All rights reserved.

The author hereby grants to MIT and the U.S. Government permission to
reproduce and to distribute copies of this thesis document in whole or in part.

Signature of Author

Joseph J. McVeigh, Jr.

Department of Ocean Engineering
May, 1991

Certified by

Anthony T. Patera

Professor Anthony T. Patera
Thesis Supervisor

Certified by

Dr. David P. Keenan

Dr. David P. Keenan
Thesis Reader

Accepted by

A. Douglas Carmichael

Professor A. Douglas Carmichael
Chairman, Department Committee on Graduate Studies
Department of Ocean Engineering

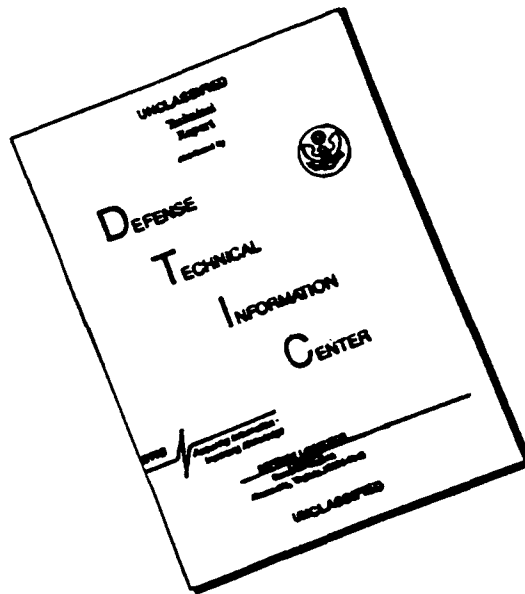
DISTRIBUTION STATEMENT A
Approved for public release
Distribution Unlimited

91-01711



91 6 10 050

DISCLAIMER NOTICE



**THIS DOCUMENT IS BEST
QUALITY AVAILABLE. THE COPY
FURNISHED TO DTIC CONTAINED
A SIGNIFICANT NUMBER OF
PAGES WHICH DO NOT
REPRODUCE LEGIBLY.**

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0198

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing the burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Project, Washington, DC 20503.

1. AGENCY USE ONLY (Leave Blank)		2. REPORT DATE JUNE 1991	3. REPORT TYPE AND DATES COVERED FINAL (MASTERS THESIS) JAN-JUNE 1991
4. TITLE AND SUBTITLE NUMERICAL SIMULATION OF FLUID FLOW IN A SIMPLE ROTOR/STATOR PAIR.			5. FUNDING NUMBERS
6. AUTHOR(S) JOSEPH J. McVEIGH, JR. MAJOR, U.S. ARMY SSN: 041-52-2437			8. PERFORMING ORGANIZATION REPORT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)			10. SPONSORING/MONITORING AGENCY REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) MILPERCEN ALEX. Va.			
11. SUPPLEMENTARY NOTES THIS DOCUMENT IS A MASTER OF SCIENCE THESIS WRITTEN AT THE MASSACHUSETTS INSTITUTE OF TECHNOLOGY. (2 COPIES)			
12a. DISTRIBUTION/AVAILABILITY STATEMENT DISTRIBUTION STATEMENT A Approved for public release Distribution Unlimited			12b. DISTRIBUTION CODE
13. ABSTRACT (Maximum 200 words) THIS THESIS DESCRIBES A SERIES OF EXPERIMENTS DEALING WITH ROTOR/STATOR INTERACTIONS IN HYDROTURBINES. THE MEANS OF ANALYSIS WAS A NONCONFORMING SLIDING SPECTRAL ELEMENT METHOD FOR THE UNSTEADY, INCOMPRESSIBLE NAVIER-STOKES EQUATIONS IN TWO-DIMENSIONAL GEOMETRY. THE VARIABLE PARAMETER IN THE SIMULATION WAS THE ROTOR ADVANCE COEFFICIENT. A COMPARISON OF LIFTING FORCES, FLOW RATE, DISSIPATION AND KINETIC ENERGY WAS CONDUCTED FOR THE VARIOUS TEST CASES. ROBUSTNESS OF THE NUMERICAL DISCRETIZATIONS WAS DEMONSTRATED BY THE CONSISTENCY OF THE COMPUTED RESULTS. THE DIVERGENCE, VORTICITY AND STREAMLINES GENERATED DURING POSTPROCESSING WERE IN STRONG AGREEMENT WITH HYDRODYNAMIC THEORY. A STEP-BY-STEP PROCEDURE IS PRESENTED FOR MANIPULATING THE WORKING ENVIRONMENT — THE DISCRETE SPECTRAL ELEMENT CONTROL VOLUME — AROUND THE ROTOR/STATOR PAIR. THE SELECTION CRITERIA FOR THE INPUT PARAMETERS AND BOUNDARY CONDITIONS IS DEVELOPED.			
14. SUBJECT TERMS TURBOMACHINERY THEORY; COMPUTATIONAL FLUID DYNAMICS; RUNNING THE SIMULATION; NAVIER-STOKES SOLUTION; DIVERGENCE; VORTICITY; STREAMLINES			15. NUMBER OF PAGES 110
17. SECURITY CLASSIFICATION OF REPORT UNCLASSIFIED			18. PRICE CODE
18. SECURITY CLASSIFICATION OF THIS PAGE UNCLASSIFIED	19. SECURITY CLASSIFICATION OF ABSTRACT UNCLASSIFIED	20. LIMITATION OF ABSTRACT	

Numerical Simulation of Fluid Flow in a Simple Rotor/Stator Pair

by

JOSEPH JOHN MCVEIGH, JR.

Submitted to the Department of Ocean Engineering on May 10, 1991, in
partial fulfillment of the requirements for the Degrees of Master of
Science in Naval Architecture and Marine Engineering and
Master of Science in Mechanical Engineering

ABSTRACT

This thesis describes a series of numerical experiments dealing with rotor/stator interactions in hydroturbines. The means of analysis was a nonconforming sliding spectral element method for the unsteady, incompressible Navier-Stokes equations in two-dimensional geometry [2]. The variable parameter in the simulation was the rotor advance coefficient. A comparison of lifting forces, flow rate, dissipation and kinetic energy was conducted for the various test cases. Robustness of the numerical discretizations was demonstrated by the consistency of the computed results. The divergence, vorticity and streamlines generated during postprocessing were in strong agreement with hydrodynamic theory.

A step-by-step procedure is presented for manipulating the working environment--the discrete spectral element control volume--around the rotor/stator pair. The selection criteria for the input parameters and boundary conditions is developed.

Thesis Supervisor: Dr. Anthony T. Patera
Title: Professor of Mechanical Engineering



Accession For	
NTIS GRA&I	J
DTIC TAB	E
Unannounced	Q
Justification	
By	
Dist to No 1	
Availability Codes	
Dist	Avail and/or Special
A-1	

Acknowledgements

I would like to thank my thesis supervisor, Professor Anthony Patera, for providing a tremendous amount of guidance during the course of this work. His positive attitude provided great hope when there did not seem to be any. For writing the algorithm and for spending many weekends at MIT giving me priceless help, I wish a very special thanks to Dr. George Anagnostou. For their assistance in helping me learn the computer systems in the fluid mechanics laboratory, I want to thank Manuel Cruz and Chris Marx.

A very special thanks goes to my thesis reader, Dr. David Keenan. Historically, thesis readers play a very small role in the overall scope of things. This was not the case with Dr. Keenan. He took a sincere interest in the development of my thesis on a daily basis, and I truly wonder if I would have made it without his guidance and optimism.

I wish to thank the US Navy and its 13A Program at MIT. As an Army officer studying how to build watercraft, my association with the Navy officers at MIT was one of the best things that could have happened to me.

Finally, and without any doubt, most importantly, I thank my family. My wife, Patty, my son, Ryan, and my daughter, Shannon, have been my inspiration over the past two years. I love them more than words can say, and I cannot thank them enough for the patience and understanding that they have provided me.

Contents

1	Introduction	6
1.1	Turbomachinery Theory	8
1.2	Axial-Flow Machines	10
2	Computational Fluid Dynamics Algorithm	12
2.1	Computational Fluid Dynamics	12
2.2	The Algorithm	13
2.3	Equations	15
3	Working Environment	20
3.1	Choosing the Number of Elements	22
3.2	Preprocessing	23
3.2.1	Beginning a Session	24
3.2.2	Setting Parameters	26
3.2.3	Building the Mesh	27
3.2.4	Setting the Boundary Conditions	37
3.3	Preparing to Run the Simulation	40
3.3.1	Modification of the Test.rea File	41
3.4	Running the Simulation	46
3.4.1	The User_fcns.c Code	46
3.4.2	Compiling the Code and Running Simula	49
3.5	Output Files	50
3.6	Postprocessing	51
3.6.1	Copies & Modifications of the Test.rea and Test.fld Files	52
3.6.2	Saving Output Graphics	53

3.7	Flow Diagram of Input/Output Files	53
4	Results	55
4.1	The Test Case Parameters	55
4.2	Analysis of the Output from the Test.iqt Files	56
4.3	Analysis of the Output from the Test.fld Files	64
4.3.1	Divergence in the Flowfield	66
4.3.2	Vorticity in the Flowfield	67
4.3.3	Streamlines	68
5	Future Work	71
Appendices		
A	Test.iqt Plots	73
B	Divergence Plots	78
C	Vorticity Plots	88
D	Streamline Plots	96
E	"Stator Only" Problem	102
F	Blade Section (FXM2 Airfoil)	108
	Bibliography	109

Chapter 1

Introduction

Great sums of money are spent annually in researching the design of aircraft, turbine engines, propellers, submarine hulls and other shapes that move through fluids [6]. During the past few years, considerable progress has been made in the numerical simulation of physical fluid flow phenomena. In particular, the accurate simulation of unsteady, two- and three-dimensional, viscous flows has been achieved in a variety of geometries. One problem of interest to both science and industry is the improvement of the efficiency of the hydroturbine. Numerical flow analysis is essential in order to properly conduct this evaluation. The hydroturbine is typically modeled as an infinite series of rotor/stator pairs. Figure 1 is an illustration of an axial-flow machine with rotor/stator pairs.

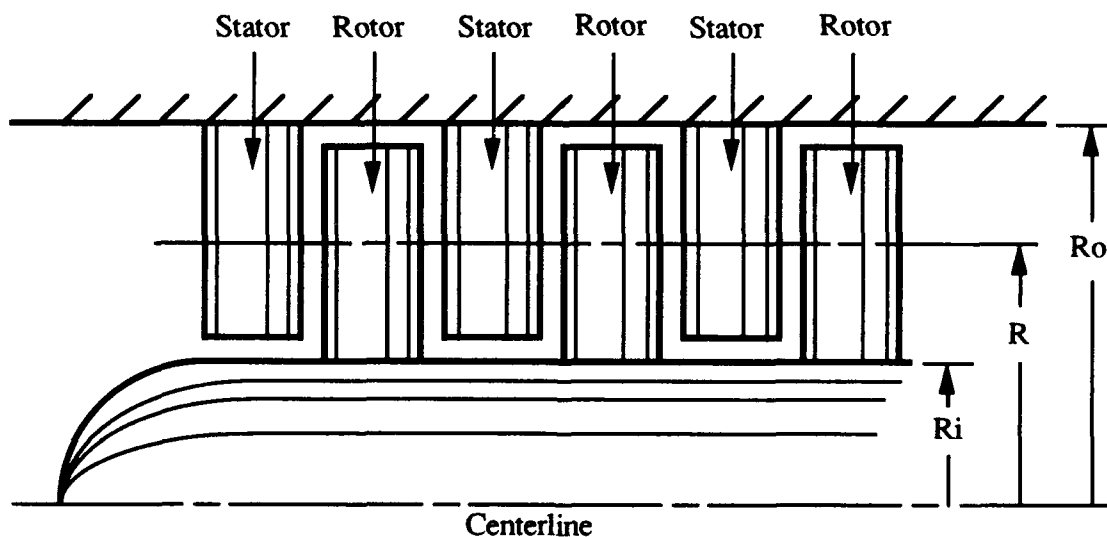


Figure 1: Diagrammatic sketch of an axial-flow machine (turbine or compressor) [19] .

Each stage of a typical axial-flow hydroturbine consists of a row of stator blades (or foils) and a row of rotor blades. The stator blade, which remains stationary as its name implies, serves to optimally direct the flow onto the rotating rotor blade in order to attain the highest possible efficiency from each stage of the hydroturbine. The flow through the successive stages of an axial flow hydroturbine is unsteady due to the interaction of the rotor/stator pairs. Difficult side effects such as wakes, vortices and unsteady surface loading on the blades inherently occur. The resulting unsteady loading, or pressure, causes the rotor blade to experience fluctuations in lift and drag. This has a direct impact on the efficiency of the turbomachine, as well as structural and noise implications [12].

The two-dimensional analysis of rotor and stator blades that are spaced far enough apart such that the interaction effects are minimal is a fairly uncomplicated task. This problem essentially becomes a simple cascade model that isolates the rotor blades, or the stator blades, as a single infinite row of airfoils. This concept, however, defeats the basic purpose of having rotor/stator pairs. The stator should serve to direct the flow onto the rotor in such a way as to optimize the efficiency of the hydroturbine. The axial spacing between the rotor/stator pair is critical to the design of such a machine. The size constraints of the machine limit the upper bound of the rotor/stator spacing while efficiency considerations limit the lower bound. As the axial gap between blade rows becomes small, the interaction effects are heightened. Blade rows that are very close together, i.e., less than 30% of chord length, tend to cause the flow to become periodically unsteady. When the rotor and stator blades are close enough together such that the interaction effects are important, the rotor/stator pair must be handled as a single system [16].

In general terms, the primary function of turbomachines, of which the turbine and the compressor are the two most widely known, is to change the energy level of the flowing medium. The compressor increases the fluid pressure and, as a result, adds

energy to the flow. The turbine, conversely, lowers the fluid pressure and removes energy from the flow. A simple way to look at them is to state that the compressor puts work into the system and the turbine takes work out. There are two basic classes of turbomachines: radial flow and axial flow. In the radial flow machine, the flow moves in a radial direction around the machine's axis of rotation. In the axial flow machine, the fluid flows in a direction that is parallel to the axis of rotation. All turbomachines have a rotor, also referred to as an impeller, which is the component that does the work on the fluid; the stator serves to guide the flow into or away from the rotor. The fluid flow through the system is continuous [19].

1.1 Turbomachinery Theory

Fluid flow in a turbomachine is highly complex. The motion of the rotor blades has a direct impact on the fluid flow. Whether the fluid does work on the rotor blades or the rotor blades perform work on the fluid, forces are generated on the blades. If we let F represent the force on the rotor blade, a corresponding torque, T , is developed. Several fundamental laws of science are necessary to explain the fluid flow in a turbomachine. Newton's 2nd law of motion states that:

$$F = ma = \frac{d}{dt} (mV) \quad (1.1)$$

In this problem, the torque developed by the force, F , is represented by the cross product of the force and the radius vector, r , at which the force acts upon the blade. The magnitude of r is measured radially from the axis of rotation in the spanwise direction of the blade. The torque, T , is calculated as follows:

$$T = r \times F \quad (1.2)$$

The above equation represents the torque due to a force acting at some distance, r , along the span of the rotor blade. Relating linear momentum, mV , to angular momentum, H , we can directly show, using Newton's 2nd law, that torque is developed due to the time rate of change of the angular momentum [19] :

$$T = \frac{dH}{dt} \quad (1.3)$$

In solving fluid flow problems in which the flow is both viscous and incompressible, as is the case for a hydroturbine, the most important equation to be addressed is the Navier-Stokes equation.

The Navier-Stokes equation is given as:

$$\frac{DV}{Dt} = -\nabla\Omega - \frac{1}{\rho}\nabla p + \nu\nabla^2 V \quad (1.4)$$

The key terms in the Navier-Stokes are defined as follows:

V is the velocity vector that provides the magnitude and direction of the fluid velocity.

$\frac{DV}{Dt}$ is the material (or substantive) acceleration. It follows the fluid particle and shows how the velocity changes over time.

Ω is the potential for gravitational force. $\Omega = gz$ and $\nabla\Omega = -gk$.

ρ is the fluid density. Density is defined as the mass per unit volume and it is a function of temperature and pressure. For an incompressible fluid,

$$\frac{\partial\rho}{\partial t} = 0 \quad (1.5)$$

∇p is the pressure gradient. It gives the change in pressure, which is the force per unit area, with respect to the fluid location.

ν is the kinematic viscosity of the fluid. It is the ratio of the absolute

viscosity, μ , to the density. Viscosity, in general, is a measure of the resistance to the sliding motion of one fluid layer over another [19].

$\nabla^2 \mathbf{V}$ is the Laplacian of the velocity vector. The Laplace operator is defined as:

$$\nabla^2 () = \left[\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \right] () \quad (1.6)$$

Thus, the Navier-Stokes equation in the x-direction becomes:

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \frac{\partial \mathbf{u}}{\partial x} + \mathbf{v} \frac{\partial \mathbf{u}}{\partial y} = - \frac{1}{\rho} \frac{\partial p}{\partial x} + \nu \left(\frac{\partial^2 \mathbf{u}}{\partial x^2} + \frac{\partial^2 \mathbf{u}}{\partial y^2} \right) + \mathbf{g} \quad (1.7)$$

The Navier-Stokes equation is the equation of motion for a viscous fluid. It is the solution of the Navier-Stokes equation that describes the velocity of a specific particle of fluid at an exact position in time. With these results in hand, the forces acting on an element of the fluid, or in this case, the forces acting on the rotor blade as a result of the fluid motion, can then be calculated.

1.2 Axial Flow Machines

Axial flow machines come in many shapes and sizes, and serve a wide variety of purposes. An elementary example of an axial flow machine is the propeller. The propeller serves as the rotor. Stator blades are not normally included and the flow is generally unconfined. The propeller is an example of a machine that does work on the fluid. The axial flow hydroturbine, conversely, removes work from the fluid. Fluid flows into the machine (or system) at a certain velocity. The fluid enters the stages of the hydroturbine in succession. Each stage consists of a rotor and a stator. The rotor/stator each have a certain

number of blades, and for water turbines this is usually anywhere from 3 to 30 [21]. As stated earlier, the stator serves to increase the velocity of the flow and to redirect the fluid into the rotor in such a way as to enhance the overall efficiency of the machine. The extracted energy is harnessed into mechanical work output by way of the machine's rotating shaft [19].

Chapter 2

Computational Fluid Dynamics Algorithm

The purpose of this thesis is to conduct an analysis of rotor/stator interactions in a hydroturbine. The means of analysis will be a computational algorithm for direct simulation of the unsteady, Navier-Stokes equations in two-dimensional geometry. Robustness of the numerical discretizations will be demonstrated by the consistency of computed results. A comparison of lifting force, flow rate, dissipation, energy balance, divergence, vorticity and streamlines will be conducted for various test cases.

2.1 Computational Fluid Dynamics

Computational fluid dynamics, referred to as CFD, is a rapidly developing science. Many extremely complex problems, that would previously have been impossible to solve before the evolution of CFD, are now relatively straightforward in nature.

Three basic principles govern the physical side of fluid flow analysis. One principle is that mass is conserved. Another is Newton's 2nd law of motion, which states that force equals mass times acceleration, or that momentum is conserved. And the third is that energy is conserved. In analyzing a fluid flow problem, these three principles are usually represented by partial differential equations. CFD is the science in which these equations are replaced by "numbers" that are advanced in space or time, or both. These advancing numbers depict a numerical solution that describes what is happening in the flowfield [3].

As a problem solving technique, CFD would never have gotten off the ground without the development of the high-speed super-computer. Typical problems, similar to the one addressed in this thesis, that contain on the order of 10^3 degrees of freedom, may require millions of calculations before a steady-state solution is obtained. Consequently, CFD has been one of the primary driving forces behind the rapid growth of the super-computer industry, and vice versa. In situations where an analytical solution would otherwise be unfeasible, or even impossible, CFD has made numerical solutions a reality.

CFD has also reduced the need for conducting expensive physical fluid flow experiments in wind and water tunnels. Aircraft, wing sections, hydrofoils, and propellers can now be modeled on computers and the desired experimental data can be obtained at a lower cost. Historically, there has been a high degree of accuracy when CFD results are compared to physical data obtained from a wind/water tunnel experiment, although many problems persist--most notably, turbulence. This confidence aspect of using CFD cannot be overemphasized [3].

The CFD model is the computer program, or algorithm, developed for a particular type of fluid flow problem. It is imperative that the CFD model contains all of the physical principles necessary to prevent any gross assumptions or violations of science that could result in false data.

2.2 The Algorithm

The computational algorithm used in this thesis was developed by Anagnostou [2] and was based on the need for such a tool in both science and industry. It provides an

original method of nonconforming discretizations for the unsteady, incompressible Navier-Stokes equations.

Modeling any fluid flow problem is a nontrivial task. The actual visualization of fluid in motion is seldom a clear cut process. The simulation of the rotor/stator interaction in axial flow hydromachinery is an excellent vehicle for demonstrating the utility of nonconforming spectral element discretizations in the analysis of fluid flow. With respect to the rotor/stator problem examined here, the computational domain, also referred to as the control volume, is a two-dimensional rectangle. It is divided into two square subdomains: one for the stator blade and one for the rotor blade. Each subdomain is further divided into an appropriate number of macro-elements. Figure 2 illustrates this concept.

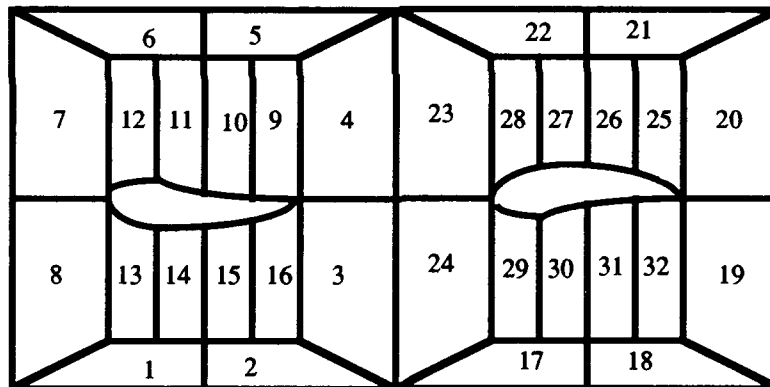


Figure 2: Domain decomposition for the rotor/stator spectral element discretization.

Theoretically, there exist no geometric restrictions on adjoining elements for coincidence of edges or corners. This property of nonconformance is at the heart of the nonconforming spectral element discretization method of computational fluid dynamics. The algorithm also allows for the decoupling of the subdomains, which permits the

modeling of sliding grids. Nonstationary geometries can thus be analyzed free of mesh distortion and great amounts of interpolation [2].

From a computational fluid dynamics perspective, Anagnostou's algorithm [2] is a first attempt at solving high order discretizations for the unsteady, incompressible Navier-Stokes equations. Dr. Man Mohan Rai, of the NASA Ames Research Center, has performed outstanding work in the analysis of compressible flows associated with rotor/stator configurations in turbomachinery using patched and overlaid grids [16, 17, 18]. An inherent limiting factor in the Anagnostou's algorithm is that the maximum allowable value for the Reynolds number is approximately 600. The algorithm is limited to low Reynolds number solutions due to both physical and financial reasons. In actual turbomachinery, the Reynolds number is often on the order of 10^5 or higher and the flow is generally in the turbulent domain. Fluid flow problems with high Reynolds numbers require much greater resolution because of the wider range of length scales. Length scales are defined as the various structures in the flow field, such as eddies, which become important at high Reynolds numbers. The greater resolution, in turn, requires much more computer memory and processing time. Considering the computers, finances and time available during the development of this algorithm, the use of higher Reynolds numbers was not a feasible option within the scope of the current research. One alternative to using low Reynolds numbers would have been to throw out the direct simulation and use a turbulence model.

2.3 Equations

In solving fluid flow problems in which the flow is both viscous and incompressible, the important equations to be addressed are the Navier-Stokes and continuity equations.

The Navier-Stokes equation:

$$\frac{DV}{Dt} = -\nabla\Omega - \frac{1}{\rho}\nabla p + \nu\nabla^2 V \quad (2.1)$$

x-direction:

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} = -\frac{1}{\rho} \frac{\partial p}{\partial x} + \nu \left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) + g \quad (2.2)$$

The continuity equation:

$$\nabla \cdot V = 0 \quad (2.3)$$

The algorithm employed here solves the Navier-Stokes equation, with a time-advancement approach, using the fractional step method. It also satisfies conservation of mass as represented by the continuity equation. Both pressure and velocity have the same approximation space. As stated previously, the Reynolds number is limited to about 600, representing a relatively slow, laminar flow.

A key portion of the algorithm is the code which computes forces, power, flow rate, dissipation and kinetic energy. The basic geometric configuration for modeling a rotor/stator machine assumes that there are an infinite number of rotor and stator combinations in sequence in the axial (x-) direction. Given that the control volume surrounding the rotor/stator has an axial length of 2L, the average pressure drop over a rotor/stator pair is represented by:

$$\frac{dp}{dx} \cdot 2L \quad (2.4)$$

This pressure head drives the flow in the axial direction. The characteristic velocity of the main flow, U_0 , was derived using equation (2.4), in combination with Bernoulli's equation. The constant "W" was set to be one periodic length in the transverse or y-direction. The characteristic velocity was determined to be:

$$U_0 = \sqrt{\frac{1}{\rho} \frac{dp}{dx} W} \quad (2.5)$$

The equation of motion for the rotor blade was developed very simply. The direction of lift was selected as the y-direction, and, given that the angular velocity of the rotor blade is equal to ωr , the equation of motion for the rotor blade is:

$$\frac{dy}{dt} = \omega r \quad (2.6)$$

Furthermore, letting the number of blade passages in a single revolution of the rotor equal N, the following relationships were derived:

$$W = \frac{2\pi r}{N} \rightarrow r = \frac{NW}{2\pi} \Rightarrow \frac{dy}{dt} = \frac{\omega NW}{2\pi} \quad (2.7)$$

Due to the "no slip" condition on the blade surface, this form of the derivative of y with respect to time is the boundary condition for the fluid velocity. By nondimensionalizing this derivative using the characteristic velocity of the main flow, U_0 , the rotor advance coefficient is designated as A:

$$A = \frac{\omega NW}{2\pi U_0} = \frac{dy^*}{dt^*} \quad (2.8)$$

Within the algorithm, the forces on the rotor blade are calculated by integrating the stress tensor around the two-dimensional foil section. Both pressure and shearing forces determine the resultant force vector. The algorithm was recently modified to enable both lift and drag to be presented as output of the code. The components of the force vector, F, are defined as [2]:

$$F_i = \left(-p\delta_{ij} + \frac{1}{Re} \cdot \frac{\partial u_i}{\partial x_j} \right) \quad (2.9)$$

The nondimensionalized form of F_i is:

$$F_i^* = \frac{F_i}{\rho U_0^2 W} = \int_r \left(-p^* \delta_{ij} + \frac{1}{Re} \cdot \frac{\partial u_i^*}{\partial x_j^*} \right) dl^* \quad (2.10)$$

The useful power out, P , due to the lifting force on the rotor blade is defined as [2]:

$$P = A \cdot \int_{\text{rotor}} (\mathbf{F} \cdot \mathbf{j}) ds \quad (2.11)$$

The nondimensionalized form of P is:

$$P^* = \frac{P}{\rho U_0^3 W \left(\frac{\Delta r}{W} \right)} = T^* \cdot \omega^* \quad (2.12)$$

Note: The torque, T , is defined as:

$$T = \int F_y dr \quad \text{and} \quad T^* = \frac{T}{\rho U_0^2 W^2 \left(\frac{\Delta r}{W} \right)} \quad (2.13)$$

The flow rate, Q , across any vertical plane is defined as [2]:

$$Q = \int_0^w u(x=0, y) dy \quad (2.14)$$

The nondimensionalized form of Q is:

$$Q^* = \frac{Q}{U_0 W} = \int u^* dy^* \quad (2.15)$$

The dissipation, Φ , which represents losses--to include non-useful work and power consumption--is defined as [2]:

$$\Phi = \frac{1}{\text{Re}} \int_0^w \int_0^{2L} \tau_{ij} \frac{\partial u_i}{\partial x_j} dx dy \quad (2.16)$$

The energy balance equation for the rotor/stator problem is given as [2]:

$$\frac{d(\text{K.E.})}{dt} = -P - \Phi + \frac{2L}{W} Q \quad (2.17)$$

The above quantities may be readily compared to classical hydroturbine parameters.

Chapter 3

Working Environment

The working environment for the rotor/stator spectral element discretization is developed using the Nektonics preprocessor. Once the blade section has been chosen, the initial focus of the problem is to generate the rectangular control volume that was discussed in chapter 2. Great care must be taken in the early planning stages to insure that later work progresses in a timely fashion.

An overview of the steps involved in performing a complete simulation is given schematically in Figure 3 on the next page.

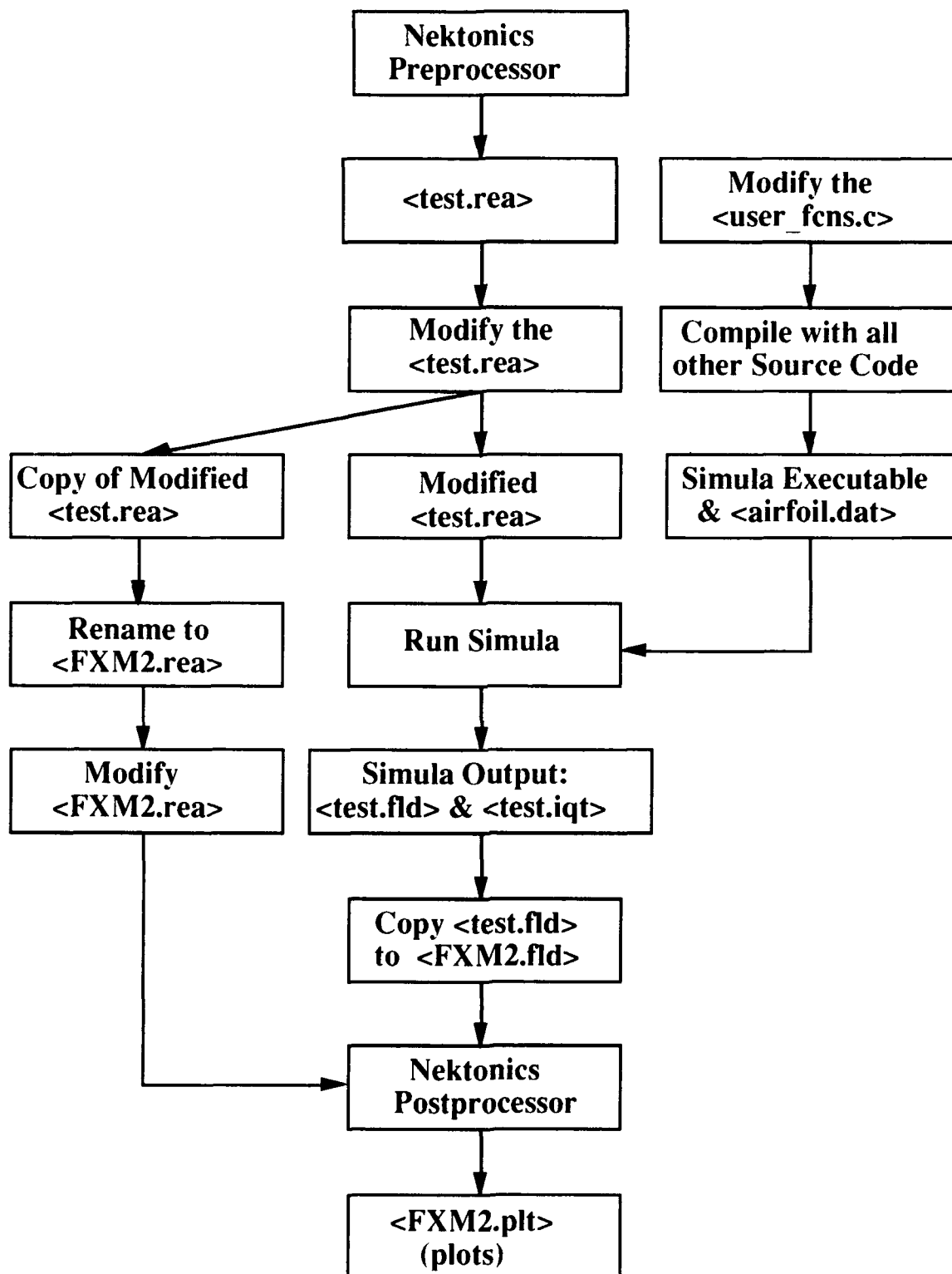


Figure 3: Schematic of a complete pre- and postprocessing simulation.

3.1 Choosing the Number of Elements

The first important consideration is the number of elements desired in the overall mesh. Several rules of thumb, with specific emphasis on fluid flow problems, are extremely valuable when developing any finite element mesh. Blade geometry versus hydrodynamics makes the selection of elements very difficult. With fluid flowing across the stator and rotor blades, the fluid boundary layers adjacent to the upper and lower surfaces of the blades are especially vulnerable to fluctuations in velocity and pressure. The requirement for good definition calls for a relatively high number of elements along these surfaces. A shortage of elements would allow for a much less accurate numerical solution of the flow.

Two additional critical locations are the leading and trailing edges of the blades. At the leading edge, the streamlines and vorticity of the flow undergo rapid changes. At the trailing edge, the possible influence of flow separation, vortex shedding and wake effects must be considered. The divergence for an incompressible flow is zero everywhere. All of these phenomena dictate the need for enhanced element definition at the leading and trailing edges of the rotor/stator blades.

Another important consideration is that all of the elements in this particular problem must be four-sided elements. This is required by the algorithm. Additionally, the ideal situation is for each element to have four corners with right angles (90°). This can be an extremely challenging requirement when working around the curved surfaces of airfoil shapes. Special care must be taken at the leading edge where the slope of the blade goes to infinity.

For the purposes of this thesis, a 32 element mesh was chosen. In reference to the rotor/stator pair, this equates to 16 elements in the square subdomain around each blade.

This is not an optimum mesh in terms of resolution; however, it satisfies the immediate requirements at this stage of the research. The mesh used for the rotor/stator problem is shown in figure 2 (see chapter 2).

3.2 Preprocessing

This section provides detailed guidance for the interface between the user and the Nektonics preprocessor. This is the stage in which the problem is defined by setting the parameters to identify the specific initial conditions and by generating the spectral element mesh. The user must have an account established with the computers, "andrei"--a Stardent 3000--and "perq"--a DECstation 3100, located in the MIT fluid mechanics laboratory. Remote connections from Project Athena color vax stations are possible. The only requirement is that the hostname of the Athena terminal is included in the .path file of the "andrei" system. The actual pre- and post-processing are done in the "perq" environment, from which the user changes directories to his "andrei" account before starting the pre- or post-processing session.

Upon logging in to a remote Athena terminal, the user types the following commands in an xterm window to enter the "andrei" system:

```
>>      xhost andrei
>>      rlogin andrei -l (username)
          password: (enter andrei password)
andrei%  setenv DISPLAY (hostname):0.0
```

From a new xterm window, the user accesses the "perq" system with the following commands:

```

>>      xhost +
>>      telnet perq
          username: (enter andrei username)
          password: (enter andrei password)
perq%    setenv DISPLAY (hostname):0.0
perq%    cd /a/(username)/(directory)
perq%    prex &

```

3.2.1 Beginning a Session

The last command--prex &--will initiate the Nektonics preprocessor. Three new windows will be generated by this command. The preprocessor will prompt the user with the following request:

ENTER SESSION NAME

The user should choose whatever name he so desires; e.g.,

test

All remaining requests appear in menu style format.

The next request is:

READ PARAMETER

TYPE IN NEW PARAMETERS

READ PREVIOUS PARAMETERS

The user is given two choices for parameter selection for a specific problem. The first choice, to type in new parameters, is used when a new problem is being developed or

when an existing problem is going to be solved again, but with different parameters. For the purposes of this tutorial, the user selects the first option: "type in new parameters".

Selections are made by clicking the left mouse button.

Upon making the above selection, a new menu appears.

EQUATION TYPE

ACCEPT CURRENT SWITCHES

(A list of options is presented here).

There are two possible settings: "Y" for yes (activate) and "N" for no (do not activate). The computational algorithm drives the selection of the following options:

(Note: the user must toggle a "Y" after each term.)

2D, UNSTEADY, FLUID FLOW, ADVECTION, SPLIT FORMULATION .

These options identify the problem as a two-dimensional, unsteady, fluid flow analysis. The advection term identifies with the Navier-Stokes (N-S) equation, while the split formulation relates to the solution of the N-S equation. All remaining parameters should be set to "N". Again, it is the computational algorithm that has specified the category of problem to be solved. The preprocessor must be formatted to meet the requirements of the algorithm. After selecting the correct options, the user should click on the menu item that reads:

ACCEPT CURRENT SWITCHES.

3.2.2 Setting Parameters

A new menu appears in the following format:

CHARACTERISTICS

ACCEPT CURRENT SWITCHES

CHARACTERISTICS

For the second time, the user should select the "select current switches" option. The preprocessor will now query the user for the parameters of the current problem. The following list of parameters appears with a default numerical value given after each item:

VISCOS:	1.000000e+00
NORDER:	5.000000e+00
TORDER:	1.000000e+00
DENSITY:	1.000000e+00
FINTIME:	0.000000e+00
NSTEPS:	10.00000e+00
DT:	0.001000e+00
IOTIME:	0.000000e+00
GRID:	0.500000e-01
TOLREL	0.100000e-01
TOLABS:	0.100000e-01
COURANT:	0.250000e+00

These parameters are extremely important for purposes of defining the fluid flow problem. The VISCOS term represents the inverse of the Reynolds number. If, for example, the user set the numerical value to be 0.0025, then the Reynolds number for the

problem would be $1/0.0025$, or 400. As stated in an earlier chapter, the "upper limit" for the Reynolds number, as applied to this algorithm and mesh, is roughly 600.

The NORDER term is the key to resolution for the spectral element mesh that is going to be built. A value of 5 equates to a 5×5 grid within each element. The algorithm would solve the Navier-Stokes equation twenty-five times per element.

The NSTEPS is the number of time steps over which the problem will be solved. The proper value is arrived by trial and error. The goal is to have sufficient time steps such that periodicity of results, or steady-state, will be reached. Also, in concert with DT, each time step should be small enough to satisfy stability and convergence conditions.

As a subset of the NSTEPS, the IOSTEP term represents the number of time steps at which the output data is "dumped" to an output file. Very simply, if NSTEPS is set at 10,000 and IOSTEP is set at 1,000, then the output data will be dumped after each 1,000 time steps for a total of 10 dumps. (The output data will be described below). Also, if NORDER is set at 5, as discussed above, then the output file will provide twenty-five solutions (rows of data) per element.

The DT term is the size of the time step. It is given in nondimensional time and is generally very small. If DT is too large, the results of the problem will inherently show a lack of resolution. A value on the order of 10^{-3} is fairly safe for this particular problem.

3.2.3 Building the Mesh

After the parameters are accepted, the next menu to appear will provide the tools to initialize the environment for physically building the mesh. The following menu appears:

CENTRAL

ALTER PARAMETERS

SHOW PARAMETERS

BUILD INTERACTIVELY

BUILD FROM FILE

IMPORT UNIVERSAL FILE

IMPORT BFC CASE FILE

ABORT (SAVING PARAMETERS)

If the user is defining a new mesh, select "build interactively". The "build from file" option is for reading-in a previously built mesh.

The next menu to be displayed is for setting the scale factors that define the interface between the user and the drawing window. This menu provides several options. The header and the key option are:

SCALE FACTOR

ACCEPT SCALE FACTORS

MOUSE

TYPE

USE NORMALIZED

Select the "mouse" option. The preprocessor will query the user to push the leftmost mouse button at 2 different x-(horizontal) locations and to choose a value for the x-length. The user has many alternatives and should use whatever works best for a particular situation. For this thesis, the two x-locations were set at 4 grid squares apart in the drawing window. An x-length of 0.5 was chosen. Thus, 4 grid squares are equivalent to a scaled length of 0.5, or 8 grid squares equal a scaled length of 1.0. The preprocessor will now repeat the queries for the y-direction. It is recommended that the same selections be made, only from the standpoint of being consistent. Nekton will now request that the user push the leftmost mouse button at a known point in the, as yet, empty grid of the

drawing window. This sets a point of reference for the preprocessor. A good choice for the rotor/stator problem is to select a point that is centered in the x-direction and that is slightly below center in the y-direction. The user must now enter the (x,y) coordinates of this known point. For simplicity purposes, it is recommended that this known point be initialized as the origin of the potential mesh with coordinates of (0,0). Once this point has been set, the user should return to the "scale factor" menu and select the "accept scale factors" option. Figure 4, on the next page, is a diagram of a preprocessing window with the scale factors and the known point set.

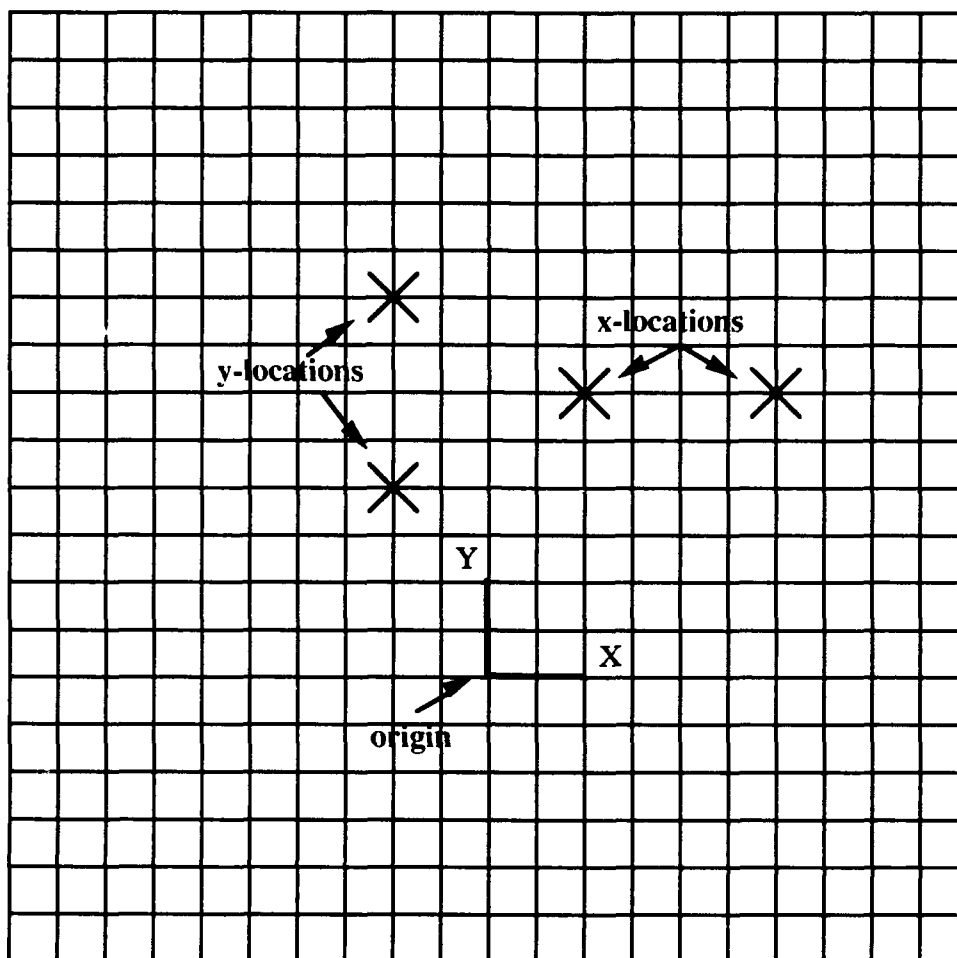


Figure 4: A version of the 20x20 grid in the Nekton preprocessor. The two scale x-and y-locations--which are arbitrary--are 4 grid squares apart. The scale length has been set such that 4 grid squares are equal to a length of 0.5. The intersection of the two bold lines represents the location of the known point, which has been set to be the origin.

The mesh generating "build" menu will come up on the screen. It reads as follows:

BUILD
 ACCEPT MESH
 ADD ELEMENT
 MODIFY ELEMENT
 GLOBAL REFINE
 CURVE SIDES
 DELETE ELEMENT

SET GRID
ZOOM
REDRAW MESH

The user is now at that stage where the control volume will be built. It is important to have carefully planned out the proposed elements on scratch paper before reaching this stage. The elements will be built around the rotor and stator blades. The blades will appear as voids (or holes) in the control volume. The user should select the "add element" option. All elements must have four sides and are entered into the drawing window one corner at a time. Within the constraints of the algorithm, the elements must be entered (by corner) in a counterclockwise orientation. The first corner of every element, except those elements that will contact the bottom surfaces of the rotor and stator blades, must be placed at the lower left corner of the element. See figure 5 for clarification.

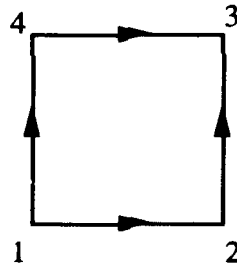


Figure 5: Corner orientation for specifying all spectral elements, except those elements adjacent to the lower surfaces of the rotor/stator blades. Note that the first corner is the lower left corner and the orientation is counterclockwise.

For those elements that contact the bottom surfaces of the blades, the first corner entered must be the upper left corner of the element. This is due to the requirement that all points along both the upper and lower blade surfaces must start at the upstream side of the flow. A sample of such an element, numbered by input sequence, follows.

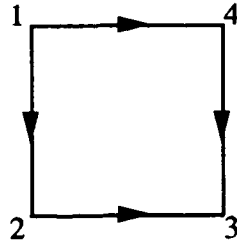


Figure 6: Corner orientation for specifying those spectral elements adjacent to the lower surfaces of the rotor/stator blades. Note that the first corner is the upper left corner and, identical to figure 4, the orientation is counterclockwise.

The corner sequence in both diagrams is counterclockwise. If this rule is violated, the results of the simulation will be wrong. The arrows indicate what the algorithm considers the positive sense along each side. There are two methods for setting the corners of the elements in the preprocessor. The corners of the elements may be positioned by clicking the mouse at the appropriate position in the drawing window. This mode will attach the element corner to the nearest grid line intersection. It is not appropriate when positioning the upper and lower blade surfaces. An alternate method is to input the exact coordinates of the corner using a numeric keypad that appears in the menu window by clicking the mouse in that window. This second approach is essential when corners do not conveniently lie on grid line intersections. Figure 7 depicts the first 8 elements that were drawn. Again, all elements were input with the first corner positioned at the lower left, in a counterclockwise direction.

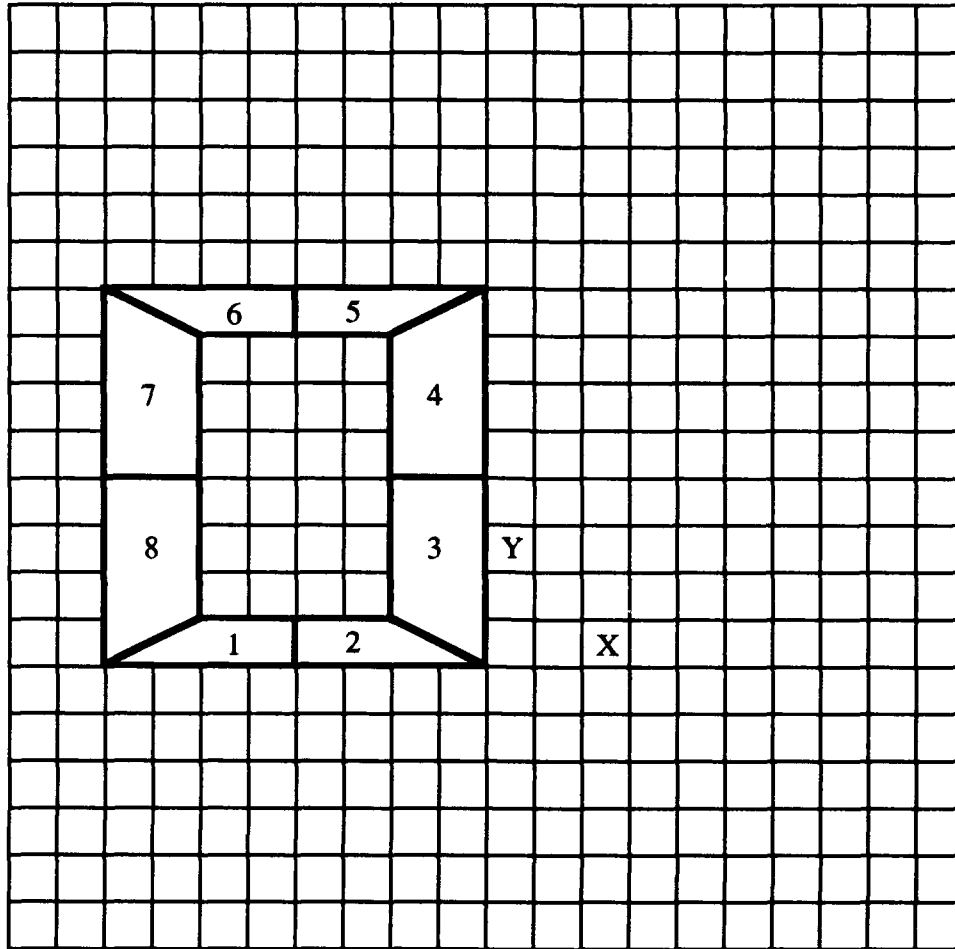


Figure 7: This is a picture of the first 8 elements of the control volume.

The user will repeatedly use the "add element" option until all 32 elements are drawn. If an error is made, the entire element can be deleted or one corner at a time may be relocated.

It is extremely helpful to treat the two square halves of the control volume, that contain the rotor and the stator blades, as separate entities when constructing them. After building the first 8 elements, elements #9 through #12 were then added to define the upper surface of the stator blade. (See Figure 8). Note that the lower corners of these four

elements are positioned at grid intersection points close to where the actual blade surface will be. Elements #13 through #16 were input to define the lower surface of the stator blade. These four elements were the only ones, out of the first 16, in which the first corner was the upper left-hand corner of the element. The reason for this was given earlier in this chapter. The elements around the rotor blade were put into the mesh in a similar fashion. The number sequence in the figures is the order in which they were built.

The square that contains the stator blade is shown in Figure 8. The coordinates that appear on the perimeter are based on the scale factors discussed earlier. Remember that four grid blocks are equivalent to a scaled length of 0.5.

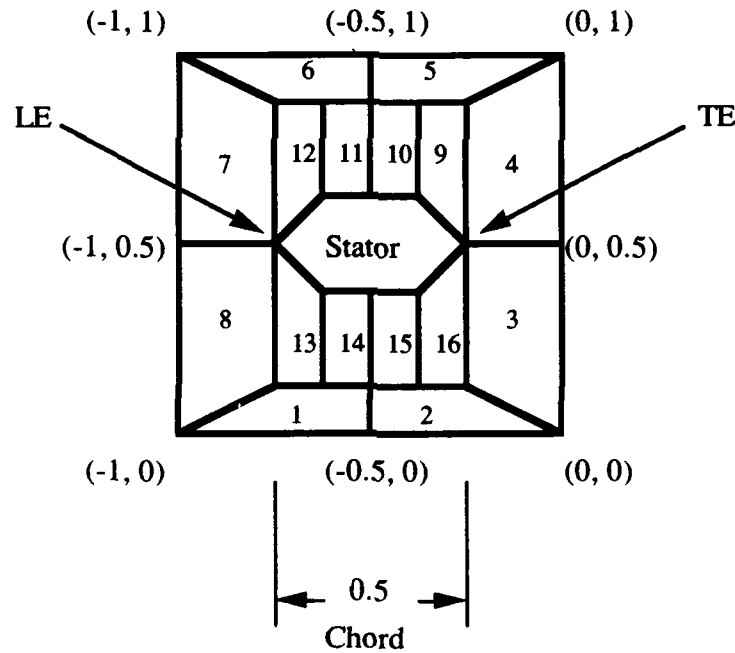


Figure 8: A diagram of 16 elements surrounding the stator blade. The coordinates are shown for reference only. The space that is not numbered is reserved for the blade section.

In the above diagram, the space for the stator blade is in the center of the square. When actually building the elements around the blade, the leading and trailing edges (LE/TE) of the foil will be at the exact positions indicated. The coordinates for the LE and TE of the stator are, respectively, $(-0.75, 0.5)$ and $(-0.25, 0.5)$. Intersections of the body surface with grid lines at the one-quarter, one-half and three-quarter chord positions for the upper and lower blade surfaces must now be adjusted from the temporary values previously assigned. The "modify element" option from the "build" menu will provide this capability. The blade surfaces are defined in the preprocessor as very "rough" sketches of what the real blade should actually look like. The actual blade section is described in **Appendix F**.

The blade surfaces, as shown in the Figure 9, are, at this stage, nothing more than four line segments that connect the LE to the TE. The end points of the line segments are located on the actual blade surface. The curved surfaces are introduced into the simulation code when the output of the preprocessing session is sent to the supercomputer for solution.

A completed preprocessing control volume is shown in Figure 9. When compared to Figure 8, note the adjustment of the element corners that touch the blade surfaces.

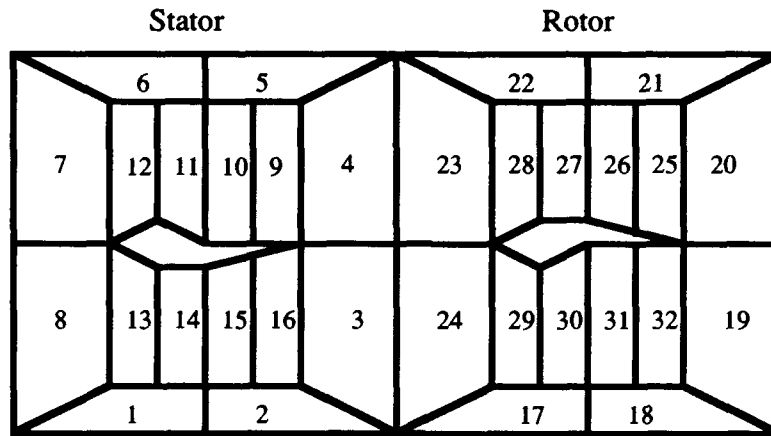


Figure 9: A completed rotor/stator control volume. At this stage, the upper and lower blade surfaces are defined by four straight line segments that intersect at actual points on the curved blade surfaces. Before the simulation is run, the "curved side data" will be added to the input file.

The nondimensional lengths, L and W , for the control volume are illustrated in Figure 10.

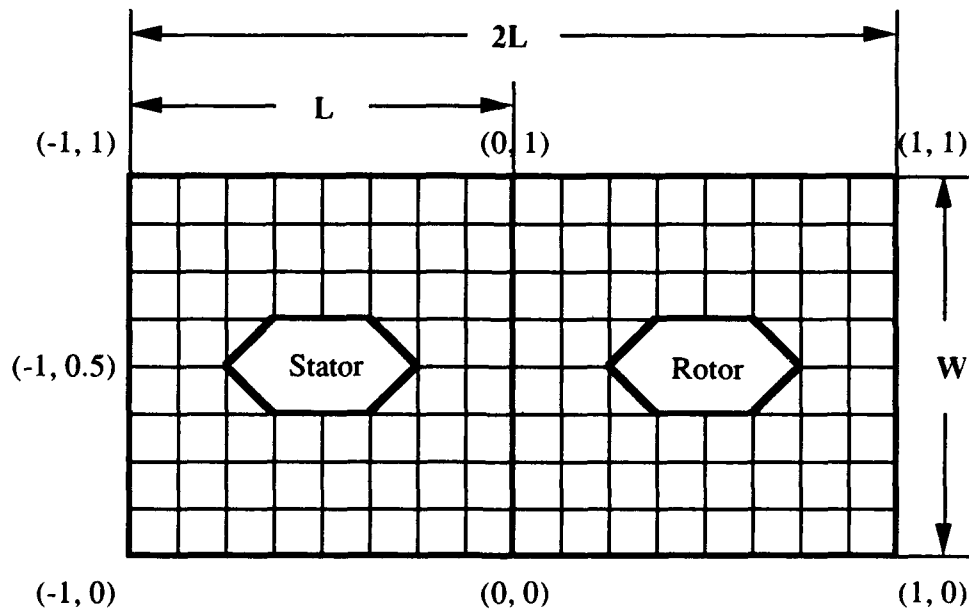


Figure 10: Nondimensional lengths for the x- and y-directions in the control volume for the rotor/stator problem.

Once the user is satisfied with the elements, the "accept mesh" option must be selected from the "build" menu alluded to above.

3.2.4 Setting the Boundary Conditions

A new menu entitled "material property" will now appear. It has two options, as shown:

MATERIAL PROPERTY
ACCEPT CONST PROPS
SET VARIABLE PROPS

Choose the "accept const props" option. The following menu will appear:

MIDWAY BREAK
SET BCs
ABORT

At this time, the user must set the boundary conditions for the fluid flow problem. Upon selecting "set BCs", the "boundary conditions" menu is displayed as follows:

BOUNDARY CONDITIONS
PERIODIC
WALL
VELOCITY (Fortran Function: Enter x,y components)
OUTFLOW
OUTFLOW/N
VELOCITY (LOCAL)
SYMMETRY
ZOOM
SET ENTIRE LEVEL

Each side of each element will be highlighted in a bright color in the drawing window of the preprocessor. The highlight indicates that the selected segment is waiting for the user to input the desired boundary condition. This particular rotor/stator problem uses four different boundary condition settings. The upper and lower surfaces of the stator blade are set to "wall". Each of the eight line segments that define the stator must individually be set to "wall". The upper and lower surfaces of the rotor blade are set to "velocity". Again, this represents eight individual boundary condition settings. When selecting the "velocity" option, the user is directed to a sub-menu. The "fortran function" option must be chosen. The user must input both x and y conditions using standard FORTRAN 77 syntax. For the rotor blade, with a velocity component in the y-direction only, the following equations should be typed into the preprocessor:

$$v_x=0$$

$$v_y=0.1 \text{ (Note: this value must be something other than zero. The actual value is unimportant).}$$

When the "periodic" BC is selected for a specific side of an element, the user must click the mouse on a second segment that is the complement of, or periodic with, the one that was just set. The periodic BC's in this problem are located on the exterior of the control volume. The top and bottom of the stator box are periodic, as well as the fluid flow entrance and exit. The fluid flows from the left boundary to right boundary, and there are an infinite number of rotor/stator pairs. All remaining BC's in the control volume are set to "outflow", to include the top and bottom of the rotor box. The "outflow" setting indicates that fluid flows across that particular side of the element. Figure 11 depicts some of the key boundary conditions for the rotor/stator problem.

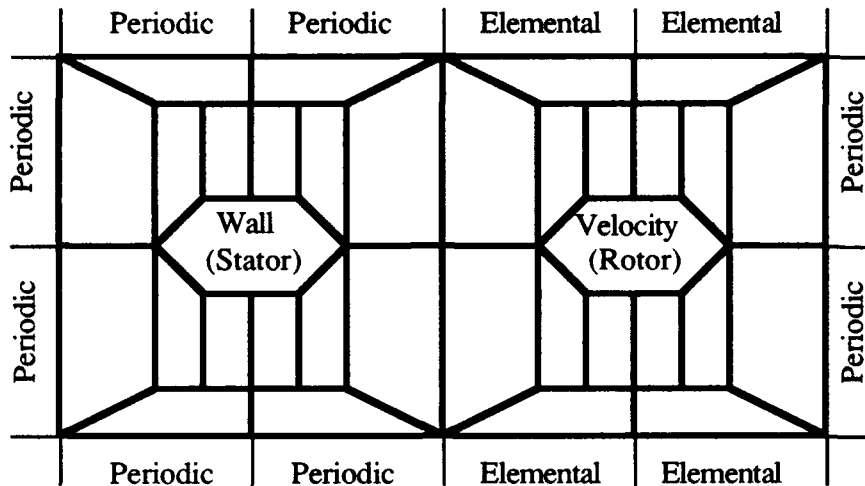


Figure 11: Boundary conditions for the rotor/stator fluid flow analysis. All internal edges not marked are set to "outflow" in the preprocessor, with two exceptions. The 8 edges that make up the stator blade are set to "wall". The 8 edges that define the rotor blade are set to "velocity". Note: all outflow BC's are changed to elemental in the test.rea file before running the simulation program.

After boundary conditions have been selected for all four sides of all 32 elements, a new menu appears as shown below:

MODIFY B.C.
 ACCEPT BOUNDARY CONDITIONS
 MODIFY B.C.
 SHOW B.C.
 ZOOM

If the user is satisfied with the boundary conditions, the "accept boundary conditions" option should be selected.

The final menu in the overall sequence of preprocessing events will now be displayed. It appears as follows:

OPTIONS

EXIT

OUTPUT

DRIVE FORCE

HISTORY

INITIAL COND

INTEGRAL QUANTITY

OBJECT

The "exit" option should be selected. The preprocessing session will end and all input data will be saved to a file with extension .rea. In this case, the file will be called test.rea by virtue of our response to the "session name" prompt.

3.3 Preparing to Run the Simulation

The computational fluid dynamics code, which runs under the command "simula", requires as input a modified version of the test.rea file that was generated by the preprocessor, Nekton. At this point, an discussion of the test.rea file is necessary. The first portion of the test.rea filename is test ; this was the name chosen at the beginning of the preprocessing session. Whatever the session name, a <__>.rea file will be generated by the preprocessor upon exiting a session. The test.rea file contains all information that was input into that specific preprocessing session. It is basically a read file that is made up of input data.

3.3.1 Modification of the Test.rea File

Before running the simulation, the test.rea file must be modified. The first modification is the addition of "curved side data" in order to correctly model the curved surfaces of the rotor and stator blades. The unmodified "curved side data" section in the test.rea file is listed as:

```
***** CURVED SIDE DATA *****  
0 Curved sides follow IEDGE, IEL, CURVE(I), I=1,5,CCURVE
```

The zero should be changed to reflect the total number of curved sides in the control volume. For this problem, there are 16 curved sides. The number 16 represents the total of the four curved sides that define the upper surface of the stator blade, the four curved sides that define the lower surface of the stator blade, the four curved sides that define the upper surface of the rotor blade, and the four curved sides that define the lower surface of the rotor blade. An example of the modified portion of the test.rea file follows:

```
***** CURVED SIDE DATA *****  
16 Curved sides follow IEDGE, IEL, CURVE(I), I=1,5,CCURVE  
1 9 0.000000 0.000000 0.000000 0.000000 0.000000 F  
1 10 0.000000 0.000000 0.000000 0.000000 0.000000 F  
1 11 0.000000 0.000000 0.000000 0.000000 0.000000 F  
1 12 0.000000 0.000000 0.000000 0.000000 0.000000 F  
4 13 0.000000 0.000000 0.000000 0.000000 0.000000 F  
4 14 0.000000 0.000000 0.000000 0.000000 0.000000 F  
4 15 0.000000 0.000000 0.000000 0.000000 0.000000 F  
4 16 0.000000 0.000000 0.000000 0.000000 0.000000 F  
1 25 0.000000 0.000000 0.000000 0.000000 0.000000 F  
1 26 0.000000 0.000000 0.000000 0.000000 0.000000 F  
1 27 0.000000 0.000000 0.000000 0.000000 0.000000 F  
1 28 0.000000 0.000000 0.000000 0.000000 0.000000 F  
4 29 0.000000 0.000000 0.000000 0.000000 0.000000 F  
4 30 0.000000 0.000000 0.000000 0.000000 0.000000 F  
4 31 0.000000 0.000000 0.000000 0.000000 0.000000 F  
4 32 0.000000 0.000000 0.000000 0.000000 0.000000 F
```

Starting with the third row of this section, the first column represents the element side and the second column represents the element number. Thus, elements 9 through 16, as shown in previous diagrams, have one side on the stator; and, elements 25 through 32 have one side on the rotor blade.

The next section of the test.rea file that requires modification is the "boundary conditions". As a result of a successful preprocessing session, a boundary condition is listed for each side of all the elements. For the rotor/stator mesh with 32 elements, that represents a total of 128 boundary conditions. The boundary conditions are given in the form of a letter. The letters and the conditions that they stand for are given below:

P	Periodic
O	Outflow
W	Wall
v	Velocity

A partial example of an *unmodified* "boundary condition" section of the test.rea file is shown on the next page.

```
*****BOUNDARY CONDITIONS*****
*****FLUID BOUNDARY CONDITIONS*****
P    1    1    0.000000    0.000000    0.000000    0.000000    0.000000
O    1    2    0.000000    0.000000    0.000000    0.000000    0.000000
O    1    3    0.000000    0.000000    0.000000    0.000000    0.000000
O    1    4    0.000000    0.000000    0.000000    0.000000    0.000000
P    2    1    0.000000    0.000000    0.000000    0.000000    0.000000
O    2    2    0.000000    0.000000    0.000000    0.000000    0.000000
O    2    3    0.000000    0.000000    0.000000    0.000000    0.000000
O    2    4    0.000000    0.000000    0.000000    0.000000    0.000000
.    .    .    .    .    .    .    .
.    .    .    .    .    .    .    .
.    .    .    .    .    .    .    .
v    32    4    0.000000    0.000000    0.000000    0.000000    0.000000
          vx=0
          vy=0.1
```

The first column is the boundary condition, the second column is the element number, and the third column is the edge of that particular element. There may be numbers other than "0.000000" displayed above. For the purpose of this research, they are not relevant. Note that for those edges with a boundary condition set to "v" for velocity, the fortran functions are listed below the row of data. There are three modifications that must be made to the "fluid boundary conditions" section of the test.rea file. The changes are:

1. For all periodic edges, denoted by a "P", delete the last five columns of data and input the periodic "strip" number for that edge. Strip numbers will be discussed in change #3 below.
2. For all outflow edges, denoted by an "O", change the "O" to an "E". The "E" represents a boundary condition setting of "elemental", and this setting is not available in the preprocessor. This change is necessary in order to make the test.rea file compatible with the simulation code.
3. Following the last row of data in the "fluid boundary conditions" section of the test.rea file, the "strip data" must be entered. Strip data refers to the four periodic strips of length "L" that were built into the control volume. As was the case for change #2 above, this addition is driven by the simulation code interface. Figure 12 displays the four periodic strips of the rotor/stator problem.

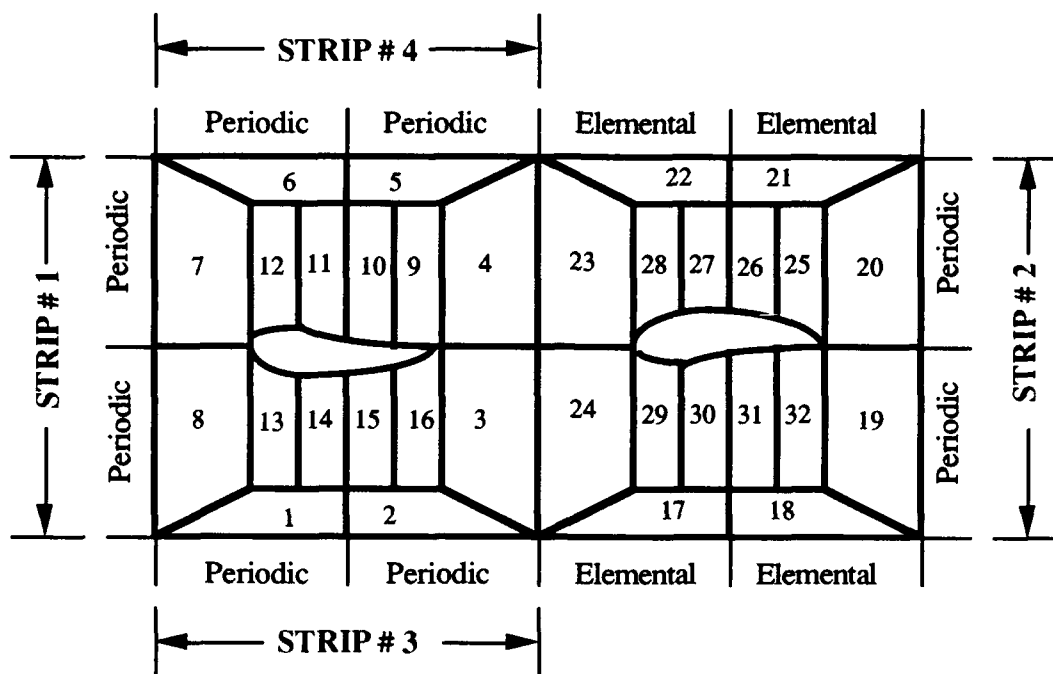


Figure 12: The four periodic strips of the rotor/stator working environment. These strips are added to the test.rea file after the preprocessing session.

A condensed version of a correctly modified test.rea file is shown below:

```
*****FLUID BOUNDARY CONDITIONS*****
P 1 1 3
E 1 2 0.000000 0.000000 0.000000 0.000000 0.000000
E 1 3 0.000000 0.000000 0.000000 0.000000 0.000000
E 1 4 0.000000 0.000000 0.000000 0.000000 0.000000
P 2 1 3
E 2 2 0.000000 0.000000 0.000000 0.000000 0.000000
E 2 3 0.000000 0.000000 0.000000 0.000000 0.000000
E 2 4 0.000000 0.000000 0.000000 0.000000 0.000000
. . . . .
. . . . .
v 32 4 0.000000 0.000000 0.000000 0.000000 0.000000
    vx=0
    vy=0.1
S -1. 0. -1. 1. 1. 1.
S 1. 0. 1. 1. 1. 1.
S -1. 0. 0. 0. 1. 2.
S -1. 1. 0. 1. 1. 2.
*****
```

From the above data, note that elements #1 and #2 both have their first (#1) edge along periodic strip #3. Additionally, the four rows of strip data at the bottom of the "fluid boundary conditions" are defined as follows: columns 2 and 3 are the x and y coordinates of the beginning of the periodic strip and columns 4 and 5 are the x and y coordinates of the end of each strip. The 6th column is the nondimensionalized length ($L=1$) of the strip; all four strips have a length of 1. The 7th column represents the overall length of that side of the control volume in which the particular strip appears.

One last item to check for in the `test.rea` file is the "coordinate" output switch located near the end of the file under the section entitled "output field specification." The switches can be set to "T" (true) or "F" (false). The "coordinate" output switch must be set to "T".

After the modifications to the `test.rea` file have been made, the simulation code can now be run. It is important that the *modified* version of the `test.rea` file is retained.

3.4 Running the Simulation

3.4.1 The User_fcns.c Code

The computational algorithm was written in "C" language. The version of the code used in this thesis is tailored to the rotor/stator problem. The one piece of code that requires some user familiarity is the `user_fcns.c` file. It is `user_fcns.c` that contains the subroutines for defining the airfoil (rotor/stator) shapes. Several different versions of this subroutine were tested. The original method for defining the airfoil shapes was the use of 4th order polynomials that are derived in reference [1]. These polynomials set forth the development of the NACA series of wing sections. The NACA wing sections are formed by combining mean lines and thickness distributions. In general, NACA airfoils are used for the higher Reynolds number category of fluid flow analyses.

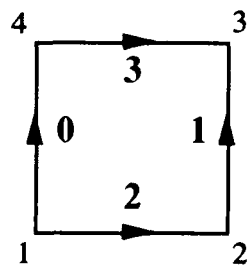
The current version of the `user_fcns.c` code reads-in a data file that contains the x and y coordinates of the upper and lower surfaces of the chosen foil shape. The data file may contain a variable number of points to define the blade surfaces. The main requirement, at this time, is that the data file, `airfoil.dat`, must be in the following example format:

FX-M2 AIRFOIL DATA		(up to 81 char. of nomenclature)
49		(# of points on upper surface)
0.00000	0.00000	(x and y coordinates of 1st pt.)
0.00107	0.00968	(x and y coordinates of 2nd pt.)
...	...	...
...	...	...
1.00000	0.00000	(x and y coordinates of last pt.)
49		(# of points on lower surface)
0.00000	0.00000	(x and y coordinates of 1st pt.)

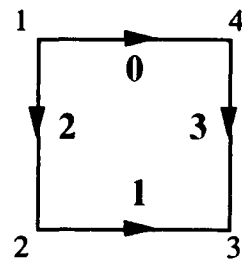
0.00107	-0.00350	(x and y coordinates of 2nd pt.)
...	...	...
...	...	...
1.00000	0.00000	(x and y coordinates of last pt.)

The section of the `user_fcns.c` code entitled **ROUTINES FOR AIRFOIL SHAPES** will automatically read-in the data file. The data file must be named `airfoil.dat`, and it must be contained in the working directory with the `test.rea` file and the algorithm.

Within the **ROUTINES FOR AIRFOIL SHAPES** section of the `user_fcns.c` code is a static array defined as `"curved_elems [32] [4]"`. The 32 rows of this array represent the 32 elements of the control volume, from #1 to #32. The 4 columns in the array are the edges of the elements. The user must insure that all values are set to "-1" except for the edges of those elements that make up the blade surfaces. There are a total of sixteen elements that have one edge on a blade surface. As the mesh is currently drawn, these sixteen elements are #9 thru #16 and #25 thru #32. (See Figure 12). To explain the edge convention, Figures 5 and 6 have been modified and are presented together in Figure 13.



Upper Surfaces



Lower Surfaces

Figure 13: Edge and corner orientation for the spectral elements along the upper and lower blade surfaces. The numbers inside the elements are the edge convention in the `user_fcns.c` code. The numbers on the corners are the order in which the corners are entered. Always enter the elements in a counterclockwise direction.

In the `user_fcns.c` code, the four edges are identified as 2, 1, 3, and 0. This is the sequence in which the edges were drawn. The numbers 2, 1, 3, and 0 correspond to the 1st, 2nd, 3rd and 4th edges, respectively. However, the four columns of the array are set up such that the 1st column is the "0" edge, the 2nd column is the "1" edge, the 3rd column is the "2" edge, and the 4th column is the "3" edge. The user should pay particular attention to this convention because there is more to come. In the `_explicit` function of the code, the blade surfaces are defined as follows:

`i = 0` upper surface of normal foil (rotor)
`i = 1` lower surface of normal foil (rotor)
`i = 2` upper surface of inverted foil (stator)
`i = 3` lower surface of inverted foil (stator)

In the static array defined as `"curved_elems [32] [4]"`, a "2" appears in the 3rd column of elements 9 thru 12. This identifies the upper surface of the stator blade, as in `i = 2`. It is in the 3rd column because that column is the number 2 edge of the element as shown in the left-hand picture of Figure 13.

The rows of the array identifying elements 13 thru 16 have a "3" in the first column. The "3" distinguishes the lower surface of the stator (`i = 3`). It is in the 1st column of the array because that column represents the number 0 edge of the element as seen in the right-hand picture in Figure 13. The remainder of the array follows this logic.

3.4.2 Compiling the Code and Running Simula

In order to dispatch the problem that was delineated in the preprocessor to the computer, the user must compile the code--which consists of 16 separate `< .c>` files--and then send in the simulation. The commands are:

```
andrei%    make
andrei%    simula test.rea 0.1 > case.log &
```

The first command, "make", compiles all of the code. If any problems exist with the creation of the object or executable files, the code will fail to compile and the user will receive appropriate error messages. Upon successful compilation, the "simula" command is given. This is the command that actually sends the job to the supercomputer. The makeup of this command line is very important.

- (1) `simula` is the command to forward the problem for solution.
- (2) `test.rea` is the file created by the preprocessor that defines the problem.
The modified version of this file is required.
- (3) `0.1` is the rotor advance coefficient, referred to earlier as the nondimensional "A" term.
- (4) `> case.log` means that in lieu of writing a status to the screen, the program will write the solution status to a file named "case.log".
This is extremely helpful information to have if a crash occurs.

This completes the steps for running a simulation. The focus of the remainder of this chapter is results and postprocessing.

3.5 Output Files

Two output files are the primary products of a successful simulation. In keeping with the earlier convention of using `test.rea` as the input file, the output files will be `test.fld` and `test.iqt`. The `test.fld` file contains the output dumps that occur at whatever frequency the `IOSTEP` parameter was set to during the preprocessing session. As stated previously, with a total number of time steps set at 10,000 and an `IOSTEP` of 1,000, the `test.fld` file will have a total of 10 output data dumps. For the rotor/stator problem in this thesis, each dump will have 807 rows of information. The first seven rows identify the dump by time and time step. The next 800 rows are the Navier-Stokes solution data. The number of rows (800) is determined by the number of elements (32) multiplied by the internal grid size (5x5) of each element. Simply stated, the Navier-Stokes equation is solved 800 times per time step. If, for example, a full period--one complete revolution of the rotor blade--were determined to be 10,000 time steps, then the computation would take place a total of 8 million times. The Navier-Stokes solution data is presented in the `test.fld` file in a five column format, as shown below:

<u>X-COORD</u>	<u>Y-COORD</u>	<u>X-VELOCITY</u>	<u>Y-VELOCITY</u>	<u>PRESSURE</u>
----------------	----------------	-------------------	-------------------	-----------------

The second output file is named `test.iqt`. It is developed using the results from the `test.fld` file. The `test.iqt` file contains one row of data per time step. The six columns represent:

<u>TIME LEVEL</u>	<u>F_x</u>	<u>F_y</u>	<u>FLOW RATE (Q)</u>	<u>DISSIPATION (Φ)</u>	<u>K.E.</u>
-------------------	----------------------	----------------------	----------------------	------------------------	-------------

The time level begins at the first time step and is incremented by `DT`. The forces are calculated in the `user_fcns.c` portion of the algorithm. The actual calculation (Eqn 2.10)

is performed by integrating the stress tensor around the rotor blade and by the using "Lee's trick" [2]. The flow rate is calculated (Eqn 2.15) by integrating the x-component of velocity across any vertical plane in the control volume. The dissipation represents losses in the system, non-useful work or energy, or power consumption. It is calculated (Eqn 2.16) by integrating the stress tensor over the whole flow domain in both the x- and y- directions. The kinetic energy is calculated (Eqn 4.7) by integrating one-half of the square of the absolute value of the velocity.

The energy balance equation from chapter 2 is:

$$\frac{d(K.E.)}{dt} = -P - \Phi + \frac{2L}{W} Q \quad (3.1)$$

The K.E., the dissipation and the flow rate are taken directly from the `test.iqt` file. The P, which represents useful power out, is just the product of A (the rotor advance coefficient) and F_y . One periodic length in the y-direction is equal to W, which is set to 1. The L represents the x-length of the rotor box or the stator box. The total control volume has an x-length of 2L. For this problem, $L=1$.

3.6 Postprocessing

A postprocessing session begins very much like a preprocessing session. Upon logging in to a remote Athena terminal, the user types the following commands in an xterm window to enter the "andrei" system:

```
>>      xhost andrei
>>      rlogin andrei -l (username)
        password: (enter andrei password)
```

```
andrei%      setenv DISPLAY (hostname):0.0
```

From a new xterm window, the user accesses the "perq" system with the following commands:

```
>>          xhost +
>>          telnet perq
              username: (enter andrei username)
              password: (enter andrei password)
perq%        setenv DISPLAY (hostname):0.0
perq%        cd /a/(username)/(directory)
perq%        postx &
```

3.6.1 Copies & Modifications of the Test.rea and Test.fld Files

The "postx" command will initiate the postprocessing session. Before this command is given, the user must make copies of the test.rea and test.fld files. These copies should have new names, e.g., FXM2.rea and FXM2.fld. Any choice would be satisfactory. The FXM2, in this case, is the airfoil nomenclature. The FXM2.rea must be modified before beginning the postprocessing session. Two deletions must be made. The "curved side data" section must be changed back to the way it was in the original test.rea file--i.e., with zero curved sides shown. Also, the strip data that was added to the "fluid boundary conditions" section must be removed. See section 3.3.1 above. These two files, the FXM2.rea and FXM2.fld, are the only files required to conduct a postprocessing session.

3.6.2 Saving Output Graphics

Giving the "postx" command from the perq window, as shown above, will start the session. In a fashion similar to the preprocessing session, three new windows will appear on the monitor. The postprocessor will prompt the user for a session name. For this problem, the user should input "FXM2", which is the filename prefix that was selected for the modified test.rea and test.fld files. Postprocessing offers many options for displaying, graphically, the Navier-Stokes solutions from the FXM2.fld file. Many examples will be discussed in the next chapter. The graphs can be saved into an automatically generated file named FXM2.plt by clicking the mouse on the "dump screen" option after each graph appears.

3.7 Flow Diagram of Input/Output Files

The flowchart on the following page depicts the pre- and postprocessing sequence. All applicable information is provided in the previous sections of this chapter. This schematic is identical to the one at the beginning of the chapter; however, section numbers have been added to show the user where appropriate references are located in this thesis.

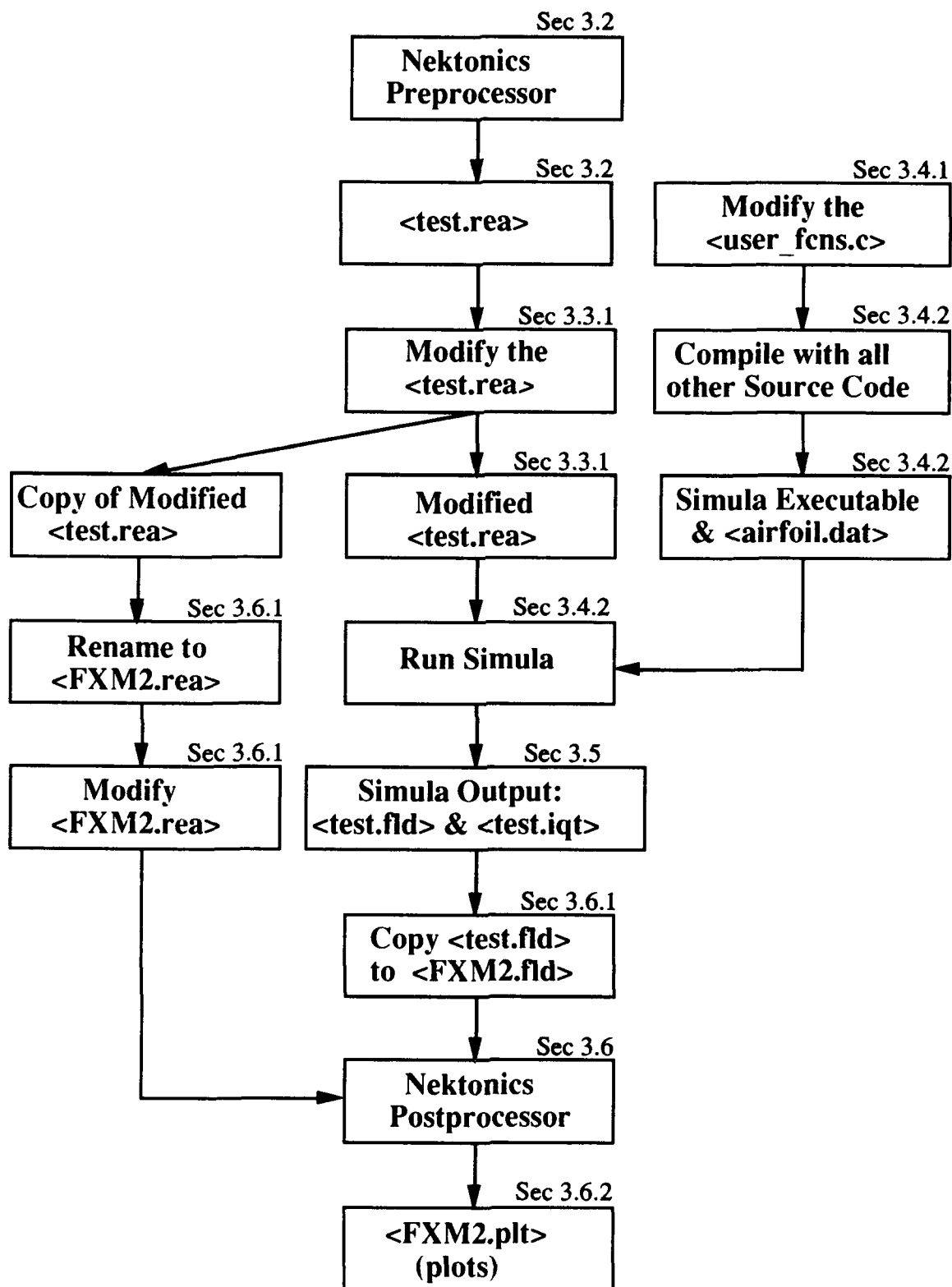


Figure 14: Schematic of a complete pre- and postprocessing simulation with appropriate references shown.

Chapter 4

Results

4.1 The Test Case Parameters

Using the computational fluid dynamics algorithm developed by Anagnostou [2] and the FXM2 airfoil as the rotor/stator blades in a 32 element mesh [see chapter 3], four primary simulations were conducted. From this moment forward, these primary test cases will be referred to as test1, test2, test3 and test4. The key parameters for each test case are listed below:

	<u>NSTEPS</u>	<u>DT</u>	<u>A</u>	<u>PERIODS</u>	<u>NSTEPS</u> <u>per</u> <u>PERIOD</u>
TEST1:	30,000	0.001	0.1	3.0	10,000
TEST2:	30,000	0.001	0.05	1.5	20,000
TEST3:	12,500	0.001	0.2	2.5	5,000
TEST4:	12,500	0.002	0.1	2.5	5,000

As a reminder to the reader, these parameters were discussed in chapter 3. To review them quickly, NSTEPS is the total number of time steps for which the Navier-Stokes solution will be solved. The DT term is the size of the time step. The "A" term is the rotor advance coefficient. The PERIODS column is simply the number of periods that the simulation was run for. The period is equivalent to the distance that the rotor blade moves in terms of one periodic length, W, in the y-direction. This term is calculated as follows:

$$(\text{NSTEPS}) * (\text{DT}) * (\text{A}) = \text{Distance that Rotor Moves} = \# \text{ of Periods}$$

The last column, NSTEPS per PERIOD, is the number of time steps in one full period. Stated another way, this is the number of time steps it takes for the rotor blade to travel one periodic distance W .

The Reynolds number for each test case was fixed at 1. The validation of the algorithm is in its very early stages and, realizing an upper limit on the Reynolds number of approximately 600, a value of 1 was chosen as a representative starting point and for convenience of comparison with Anagnostou's work [2]. Obviously, the fluid flow in the x -direction will be moving in a manner relative to the speed of cold molasses.

The rotor advance coefficient was selected to be the variable parameter in this experiment. Test cases 1,2 and 3 have "A" values of 0.1, 0.05 and 0.2, respectively. Test case 4, which is a derivative of test case 1, had an "A" of 0.1; the variable parameter was the DT, which was doubled to 0.002. This last case was run in order to determine if the size of the time step had any effect on the solution.

4.2 Analysis of the Output from the Test.iqt Files

The two output files--test.fld and test.iqt--discussed in section 3.5, were successfully generated for the four test cases. The test.fld file, which contains x and y coordinates, x and y velocities, and pressure, includes all of the data required for postprocessing. This file is the Navier-Stokes solution to the rotor/stator problem.

The test.iqt file contains the time step, the force (x - and y -components) on the rotor blade, the flow rate, the dissipation and the kinetic energy. These data were plotted against time to determine periodicity of results. Reaching a steady-state condition is an important objective of this experiment, and the test.iqt file was used as the standard for

verifying if and when the system went to equilibrium. In a fluid flow problem that has not reached a steady-state, drawing conclusions would be extremely presumptuous.

A graphic representation of the data in the test.iqt files for the four test cases is presented in Appendix A. An immediate observation is that the F_x , F_y , flow rate, dissipation and kinetic energy curves display periodicity after only one full period. One minor deviation to this trend occurs during the first few hundred time steps. The start-up motion of the fluid flow and the rotor blade shows effects due to the impulsive start. However, these effects are quickly damped out. They are most pronounced in the data from test3, in which the rotor advance coefficient is the highest of all the test cases.

The sudden peaks, especially in the F_x and F_y curves, are difficult to explain. Hydrodynamics and turbomachinery theory do not support their existence. Additionally, these discontinuities do not correspond to any significant event in the flowfield. They could possibly be attributed to problems at element interfaces or to problems accounting for the motion of the sixteen elements surrounding the rotor blade. These elements must wrap around in the y -direction during the rotation of the rotor blade. The sliding mesh concept may be causing inaccuracies.

The velocity and pressure terms computed during the Navier-Stokes solution are used in the force calculations. There exists a high degree of confidence in the velocity and pressure data based on postprocessing observations. Unfortunately, this confidence does not carry over into the force terms. In addition to the unexplainable spikes in the lift and drag curves, the F_y term was first thought to have a sign problem. This became apparent during an attempt to satisfy the energy balance equation. Another potential problem is that the magnitude of the dissipation appears too large in proportion to the flow rate. However, a check of the calculation of dissipation in the code shows that it is correct.

A very important assumption, with far reaching consequences, was made with respect to the output data contained in the test.iqt files. The flow rate, dissipation and kinetic energy were accepted as being correct. Both the x- and y-components of the force on the rotor blade were disregarded. Using the energy balance equation for the rotor/stator problem, the force in the y-direction on the rotor blade was solved for. Note that the quantities are in nondimensional form. The energy balance equation is [2]:

$$\frac{d(K.E.)}{dt} = -P - \Phi + \frac{2L}{W} Q \quad (4.1)$$

The P term is the useful power out of the system due to the lifting force on the rotor blade. P is represented by the following equation [2]:

$$P = A \cdot \int_{\text{rotor}} (\mathbf{F} \cdot \mathbf{j}) ds \quad (4.2)$$

or, simply stated, $P = A \cdot F_y$ (4.3)

Integrating the energy balance equation over a full period, we have:

$$\frac{1}{T} \int_t^{t+T} \frac{d(K.E.)}{dt} dt = \frac{1}{T} \int_t^{t+T} \left(-A \cdot F_y - \Phi + \frac{2L}{W} Q \right) dt \quad (4.4)$$

and, solving for an average F_y over that period:

$$\overline{F_y} = -\frac{1}{A} \cdot \left[\overline{\Phi} - \frac{2L}{W} \overline{Q} \right] \quad (4.5)$$

The derivative of K.E. with respect to time is essentially zero. The results of this calculation, in nondimensional quantities, follow:

	$\overline{Q}$	$\overline{\Phi}$	$\overline{K.E.}$	$\overline{F_y}$	A
TEST2:	0.0968	0.1970	0.0113	- 0.068	0.05
TEST1:	0.0971	0.2196	0.0141	- 0.254	0.1
TEST4:	0.1004	0.2233	0.0145	- 0.225	0.1
TEST3:	0.0978	0.3089	0.0249	- 1.133	0.2

Note that the tests are listed in order of increasing rotor speed. The graph in Figure 15 illustrates the trends in the first three output quantities with respect to an increasing "A" value.

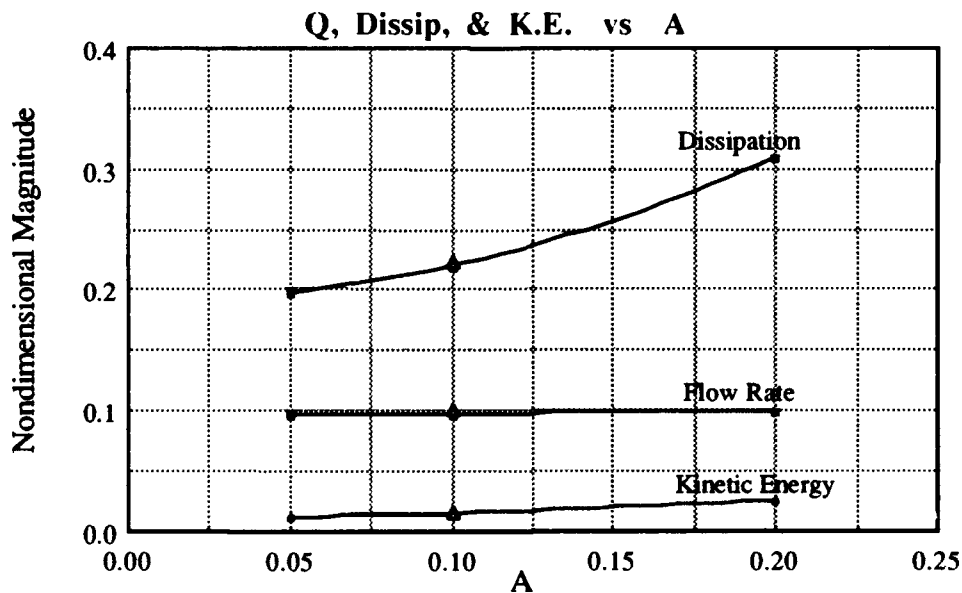


Figure 15: A graph of dissipation, flow rate and kinetic energy as a function of the rotor advance coefficient, "A". The three values for "A" in test cases 1,2 and 3 are 0.05, 0.10 and 0.20, respectively. The small triangle points at A=0.1 represent the data from test case 4, in which the DT was doubled to 0.002.

The flow rate, Q , is approximately constant because the flow is quasi-static. The flow rate was also checked in the postprocessing session. A vertical (y-direction) cut was taken through the control volume and the x-component of velocity was plotted as a profile plot. The area under the V_x curve was computed and the value was equal to the flow rate in the test.iqt file. This is in perfect agreement with the flow rate definition given in chapter 2 [2]:

$$Q = \int_0^w u(x=0, y) dy \quad (4.6)$$

The dissipation clearly increases with an increase in rotor speed. From $A=0.05$ to $A=0.1$, the average dissipation rose by 11%; from $A=1.0$ to $A=2.0$, the dissipation rose by 41%. The dissipation, which consists of losses in the form of non-useful work and power consumption, can also be described as thermodynamic losses due to the viscosity of the fluid. With the increase in rotor advance coefficient, the viscous effects of the fluid are magnified and more energy is dissipated within the fluid.

The graph in Figure 16 is a plot of natural log of the velocity versus the natural log of the dissipation. The slope of the curve, which is approximately 0.64, gives the order of the relationship between the two variables. The velocity was calculated using a relationship derived from the velocity triangle of the flow.

$$V^2 \cong \left(\frac{Q}{W}\right)^2 + A^2 \quad (4.7)$$

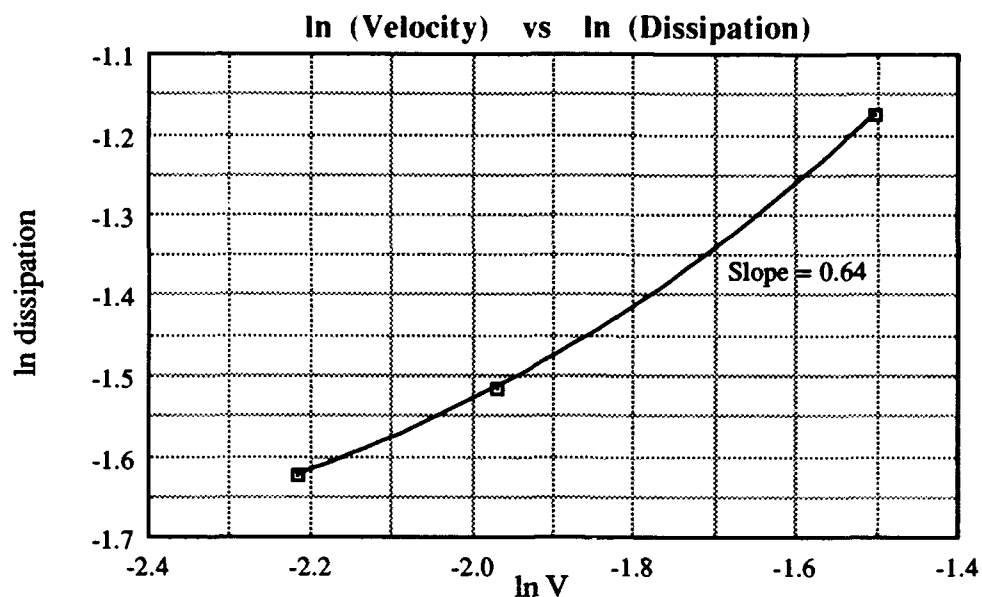


Figure 16: Plot of the natural log the of velocity versus the natural log of the dissipation.

The kinetic energy of the system, although not readily apparent from the above graph, advances in a manner similar to the dissipation. Figure 17 is another plot of A versus K.E., but with a revised scale on the ordinate.

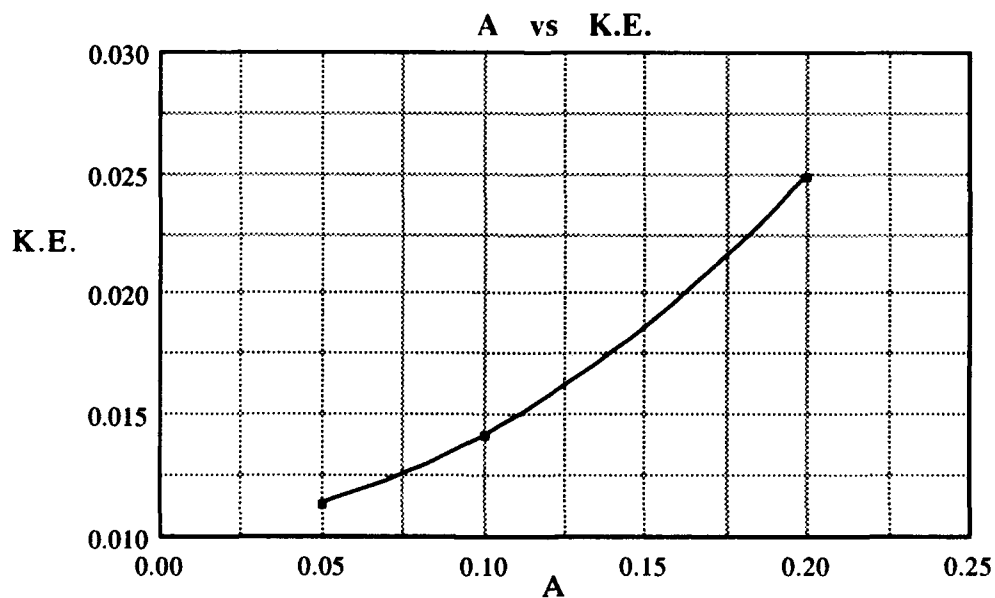


Figure 17: A graph of the the rotor advance coefficient, "A", versus the kinetic energy. The ordinate scale was magnified to show the nonlinear behavior of K.E.

From $A=0.05$ to $A=0.1$, the average kinetic energy of the fluid increased by 25%; from $A=0.1$ to $A=0.2$, the kinetic energy increased by 77%. Kinetic energy in the algorithm is defined as [2]:

$$\text{K.E.} = \int_0^L \int_0^L \frac{1}{2} \cdot |V|^2 \, dx \quad (4.8)$$

From this definition, the kinetic energy in the `test.iqt` files should behave as the $[\text{Velocity}]^2$.

The graph in Figure 18 is a plot of the natural log of the velocity versus the natural log of the kinetic energy. The slope of the curve, which is approximately 1.12, gives the order of the relationship between the two variables.

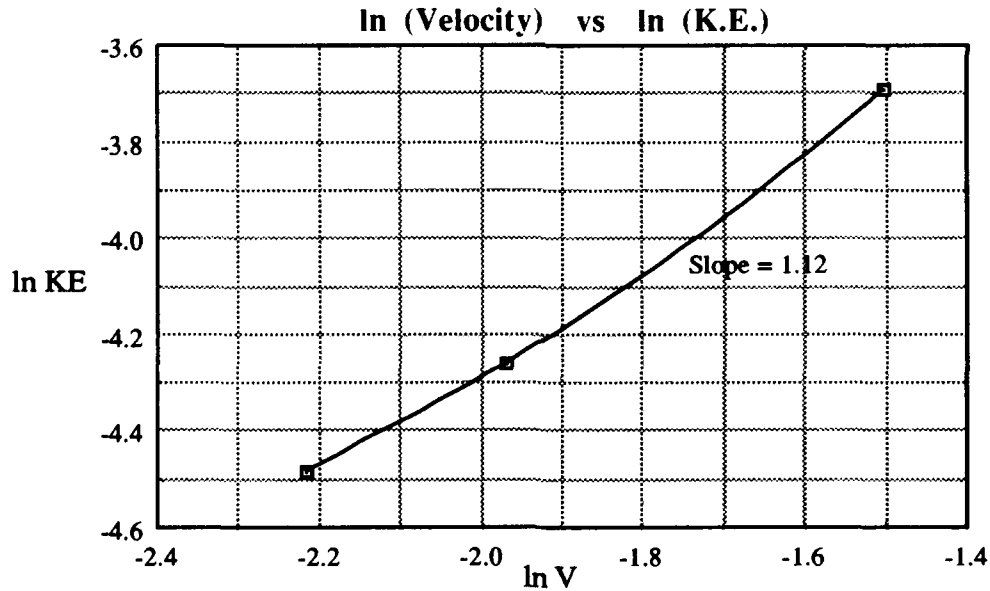


Figure 18: Plot of the natural log of the velocity versus the natural log of the kinetic energy. The velocity was calculated in the same manner as in Figure 16.

The graph in Figure 19 shows the relationship of the "calculated" force in the y-direction on the rotor blade as a function of the rotor advance coefficient.

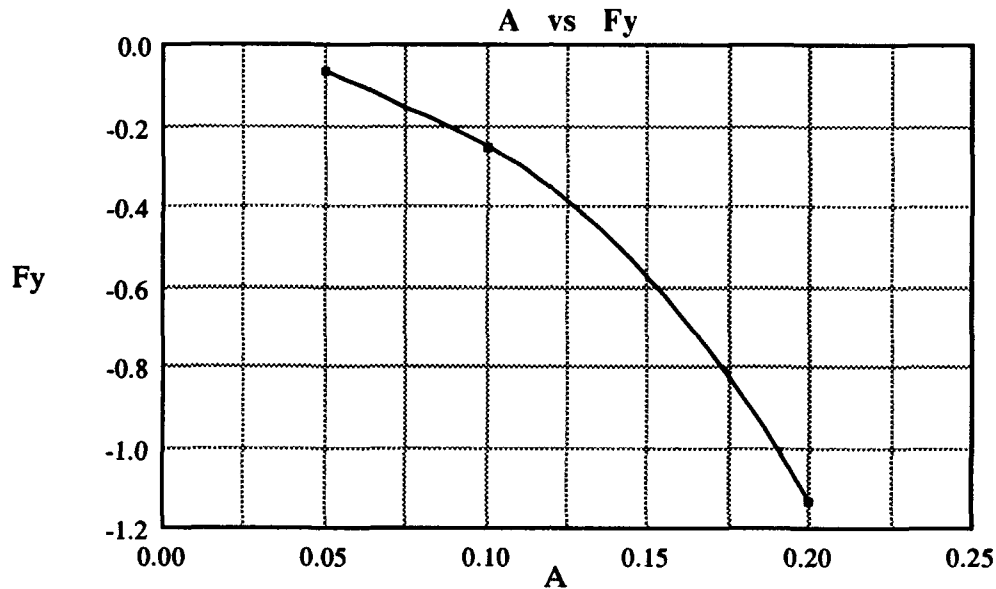


Figure 19: A graph of the lifting force on the rotor blade as a function of the rotor advance coefficient, "A".

The force on the rotor blade in the y-direction is the lifting force. The results, with respect to turbomachinery, optimally should have shown a positive lifting force. Although the average force is negative in all of the test cases, the data follows a noticeable pattern. Assuming that the energy balance equation, the flow rate, the dissipation and the kinetic energy are all correct, then this version of F_y is correct. The changes in magnitude in F_y , with respect to increasing A, are 275% and 350% of the previous value, respectively. What does this data tell us about the rotor/stator problem? Instead of the fluid pushing the rotor blade, the rotor blade is actually plowing through the fluid. The trend in F_y goes as expected.* As will be discussed in section 4.3.3 on streamlines, a rotor advance coefficient of 0.05 was plausible. Under the conditions of this experiment, an "A" of some value slightly less than 0.05 is necessary to produce positive lifting force.

* This is consistent with the fact that for a Reynolds number of 1 you would not expect any aerodynamic lift.

4.3 Analysis of the Output from the Test.fld Files

As stated earlier, the test.fld file contains x and y coordinates, x and y velocities, and pressure. This file represents the Navier-Stokes solution to the rotor/stator problem. It is organized such that the 5×5 grid within each element generates 25 solutions per element per time step. This file includes all of the output data required for postprocessing. The input data for postprocessing is contained in the test.rea file. Refer to chapter 3 for a more detailed explanation of these files, for information on file modifications and for instructions on postprocessing.

For this thesis, three primary postprocessing plot formats were utilized. The plot formats were:

1. **Divergence Plots:** Profile plots of the divergence in the control volume were obtained at the leading edge, mid-chord and trailing edge of the rotor blade. A vertical slice was taken at each of these three locations. Also, a random sampling of divergence contour plots was taken at various time steps in the problem.
2. **Vorticity Plots:** Profile plots of the vorticity in the fluid were taken along the thirteen vertical lines highlighted in Figure 20.

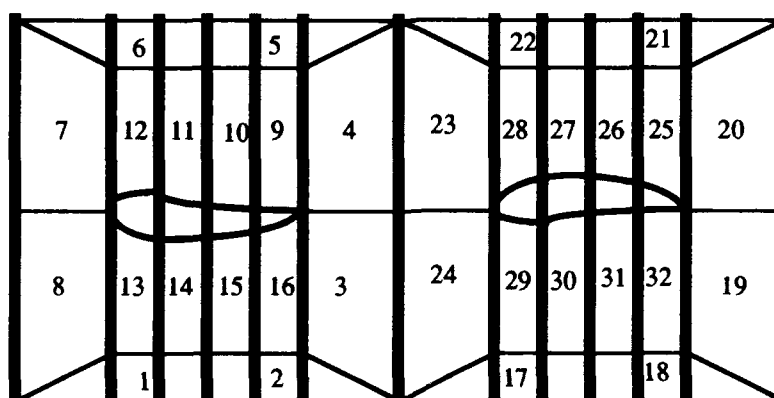


Figure 20: Rotor/stator control volume with the thirteen vertical lines in bold highlight. These lines represent the locations across which the profile plots were generated.

2. (cont.) Additionally, the vorticity was plotted as contour lines--to illustrate the overall vorticity throughout the control volume--and as fishnet grids--to show the curved blade surfaces that should appear in all postprocessing graphics.
3. Streamlines: Several streamlines were plotted at various locations in the control volume. The emphasis on plotting streamlines was directed at tracking the wake effects from the stator blade onto the rotor blade.

A variety of time steps were selected for the plots discussed above. Time steps that occurred near important events in the rotor/stator interaction were emphasized. With the knowledge that the flowfield became periodic during the first period, the following time steps were selected as being illustrative of the flow:

For test2 (with $A=0.05$ and 20,000 time steps per period):

6,000 , 8,000 , 10,000 , 16,000 , 18,000 , 20,000 , & 28,000.

For test1 (with $A=0.10$ and 10,000 time steps per period):

3,000 , 4,000 , 5,000 , 8,000 , 9,000 , 10,000 , & 19,000.

For test3 (with $A=0.20$ and 5,000 time steps per period):

1,000 , 2,000 , 3,000 , 4,000 , 5,000 , & 9,000.

Significant jumps in the test.iqt data (see Appendix A) occurred at the half and full period junctures in the rotation of the rotor blade. As a result, the time steps in the vicinity of these jumps were chosen for postprocessing analysis. Additionally, data taken at the full period position of the flow development, when the rotor blade is passing the stator blade, is always of interest. Another consideration in the selection of the above time steps was the observation of the wake effects from the stator onto the rotor.

4.3.1 Divergence in the Flowfield

The divergence of a fluid in motion is a measure of how rapidly streamlines move away from each other. For a uniform, incompressible fluid, the continuity equation states that the divergence is zero everywhere. The continuity equation, for compressible and incompressible fluids, is:

$$\frac{D\rho}{Dt} + \rho (\nabla \cdot \mathbf{v}) = 0 \quad (4.9)$$

In an incompressible fluid, the density, ρ , cannot change. As a result, the continuity equation becomes:

$$\nabla \cdot \mathbf{v} = 0 \quad (4.10)$$

From conservation of mass, fluid cannot be created or destroyed. If divergence exists, other fluid moves in to fill the void from the streamlines moving away from each other. This becomes a useful check on the numerics.

The rotor/stator configuration exists in an incompressible medium. The profile and contour plots validate the premise that the fluid is indeed incompressible. The divergence is zero virtually everywhere in the flowfield, with several minor exceptions. These exceptions exist primarily at the leading and trailing edges of the rotor and stator blades, and at interfaces between elements. The non-zero values for divergence can best be attributed to grid distortion and poor selection of element corners--angles that are too close to zero or 180°. One visible trend in the non-zero divergence values is an increase in magnitude with increasing rotor advance coefficients. The higher speeds infer greater distortion. Additionally, the divergence is exactly periodic. See Appendix B for examples of divergence plots.

4.3.2 Vorticity in the Flowfield

The vorticity is a measure of the rotation of the flow; it is calculated from the angular velocity of a fluid element about its center of mass. The generation of lift by airfoils requires rotational flow. In the rotor/stator problem, the flow is definitely rotational in nature. Vorticity, which is the curl of the velocity, is defined as:

$$\nabla \times \mathbf{V} = \boldsymbol{\omega} \quad (4.11)$$

Overall, the output plots of vorticity in the flowfield look correct. The orientation of the stator blade drives the flow at a faster rate along its lower surface. This phenomenon is caused by the Venturi effect, which results in the compression of streamlines as the flow rounds the lower surface. The same effect occurs along the upper surface of the rotor blade.

A very strong affirmation of the algorithm, with respect to the computation of vorticity, was achieved by running a simulation without a rotor blade. The results of the "stator only" problem, as provided in Appendix E, clearly show zero vorticity in the entire half of the control volume that previously contained the rotor blade. The algorithm reflects the physics of fluid flow. Without a rotor blade to impose circulation, vorticity does not exist. Note that vorticity exists only in the stator side of the control volume.

The vorticity plots support sound hydrodynamic theory. For all of the test cases, a dominant trend in the vorticity data is that the vorticity is positive along the lower surface of the blade sections and negative along the upper surface. With velocity increasing in an outward direction moving away from the both the upper and lower blade surfaces, the right-hand rule applied to the velocity vectors confirms the presence of negative and positive vorticity, respectively. Figure 21 illustrates this concept.

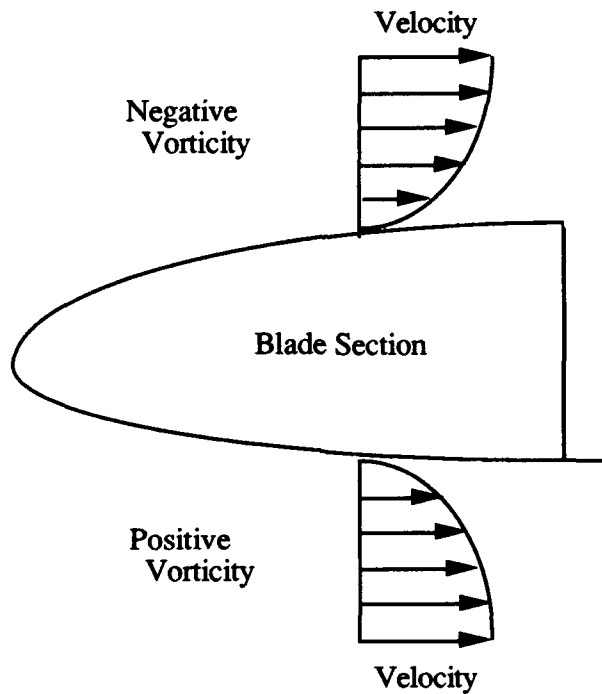


Figure 21: The velocity of the flow in the immediate vicinity of the blade surfaces. Applying the right-hand rule to the velocity vectors determines the vorticity sign convention. On the upper surface, the right-hand rule dictates that the vorticity is negative; on the lower surface, it is positive.

The vorticity plots (see Appendix C) show consistency of results whether one full period has elapsed or whether the flow is less than 20% developed. Overall, the vorticity does not display a dependence on time. This observation would not be valid in the first few time steps of the problem when the flow is being "jump-started". Additionally, the magnitude of the vorticity increases with an increase in the rotor advance coefficient.

4.3.3 Streamlines

The streamline is an imaginary curve in the fluid across which there is no flow. The velocity of every fluid particle along the streamline is tangential to that streamline. Put another way, streamlines are conceptual curves in the fluid region that are everywhere

tangent to the velocity vector. Several factors drive the movement of fluid particles. In the rotor/stator problem, the fluid is defined as being viscous. Viscosity and the no-slip boundary condition along the blade surfaces give rise to a velocity gradient, as depicted in figure 15, and to shear stresses. In the algorithm, the fluid flow is a result of the forces brought about by a pressure gradient (see chapter 2). The motion of the fluid particles is controlled by inertia and by the shear stresses in the surrounding fluid [5].

The streamlines (see Appendix D) in the rotor/stator environment behave as expected. The changing slopes of the streamlines relate to physical events in the flow. One important area of focus is the fluid velocity triangle as it relates to the slope of the streamlines. As stated earlier, a separate test was conducted with only a stator blade in the control volume and no rotor blade. The streamlines for this problem have a very small positive slope. Figure 22 displays this concept.

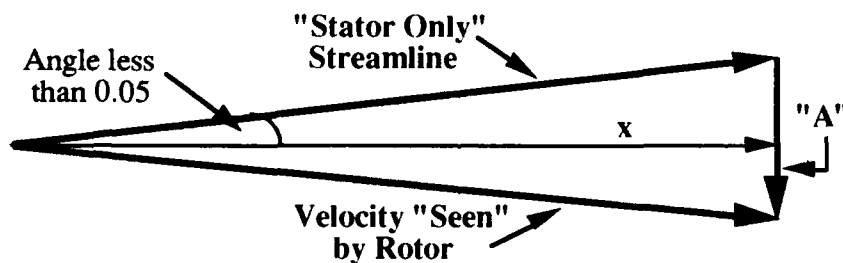


Figure 22: Velocity triangle illustrating the slope of a streamline from the "stator only" problem. Letting the x-velocity equal 1 and measuring the angle of attack as less than 5° , it is apparent that the natural rotor advance coefficient, "A", should be less than 0.05. (The angle of the streamline has been exaggerated for visualization purposes).

The above drawing of a streamline from the output data of the "stator only" problem shows that the natural rotor advance coefficient is less than 0.05. The test cases in this thesis were run at "A" values of 0.05, 0.10 and 0.20. Running the rotor/stator simulation

at "A" values higher than the natural value supports the data showing a negative F_y on the rotor blade. This is a very important justification. See Appendix E for results of the "stator only" problem.

Chapter 5

Future Work

The test cases that were used to analyze rotor/stator interactions in a hydroturbine were limited in scope. The Reynolds number of 1 represents a flowfield with a very low velocity and does not come close to representing the flow in an actual turbomachine that has Reynolds numbers on the order of 10^5 or higher. Such a low Reynolds number is not of much value experimentally. A subsequent simulation with a Reynolds number of 100 and a rotor advance coefficient of 0.1 was tested. The conjugate gradient in the solver failed to converge after completing 85% of the first period. This is not a fundamental flaw in the algorithm. It is most likely attributed to a "bug" in the code.

The integral quantities presented in section 4.2--"analysis of the output from the test.iqt files"--have discontinuities that generate a lack of trust in certain portions of the output data. (See Appendix A). It is clear that the "rolling over" of the grid on the rotor side of the control volume has a negative impact on the results. The pressure and velocity quantities from the test.fld files appear highly accurate and rate a high degree of confidence.

The geometric angle of attack for the stator and rotor blades in the simulations was zero. Ideally, the foils should have had some angle of attack to take maximum advantage of the velocity field as depicted by the streamlines exiting the trailing edge of the stator and moving into the rotor. With the current status of the code, building a mesh with angles of attack on the foils was limited by the higher number of elements required. The ideal mesh would have a movable circle built around the rotor and stator blades. The circle would allow rotation of all elements, to include the foils, on the inside of the circle. The

nonconforming spectral element discretization method used in the algorithm would permit this rotating element concept.

The elements around the blades should have much greater definition, with special emphasis on those elements in the vicinity of the leading edges which are subject to have angles approaching zero due to the curvature of the blade. Figure 23 illustrates the movable circle concept and the enhanced definition of the elements near the blade surfaces.

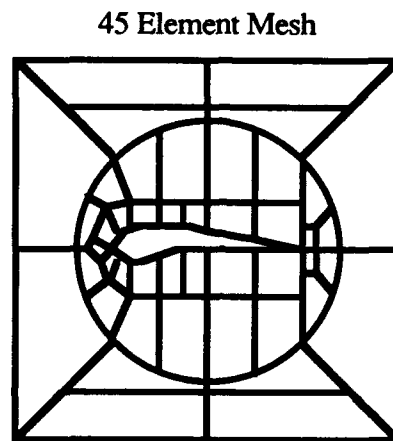


Figure 23: This is an example of a well-defined single blade control volume. Note the excellent definition of the elements around the leading and trailing edges of the foil section. Ideally, all of the elements inside of the circle would rotate to allow angles of attack to be introduced into the problem.

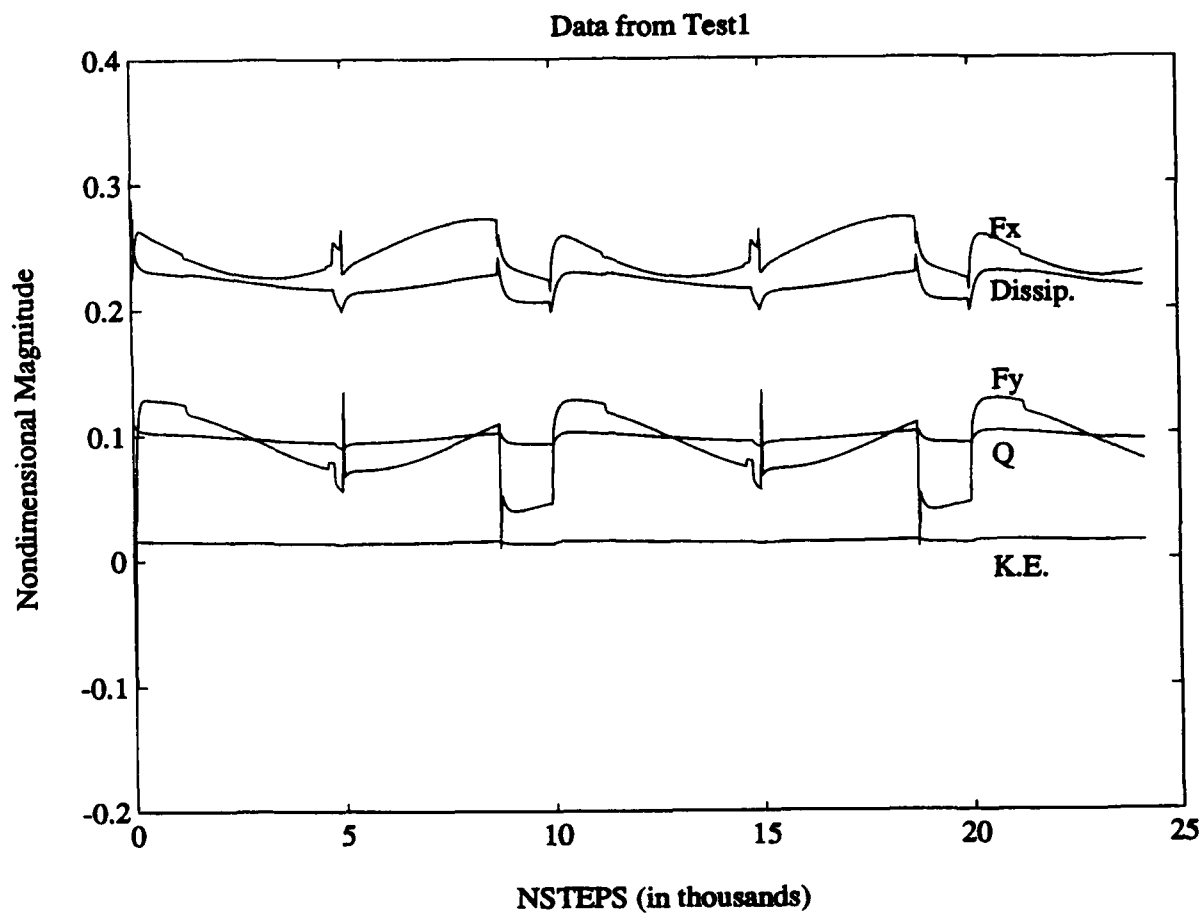
Appendix A

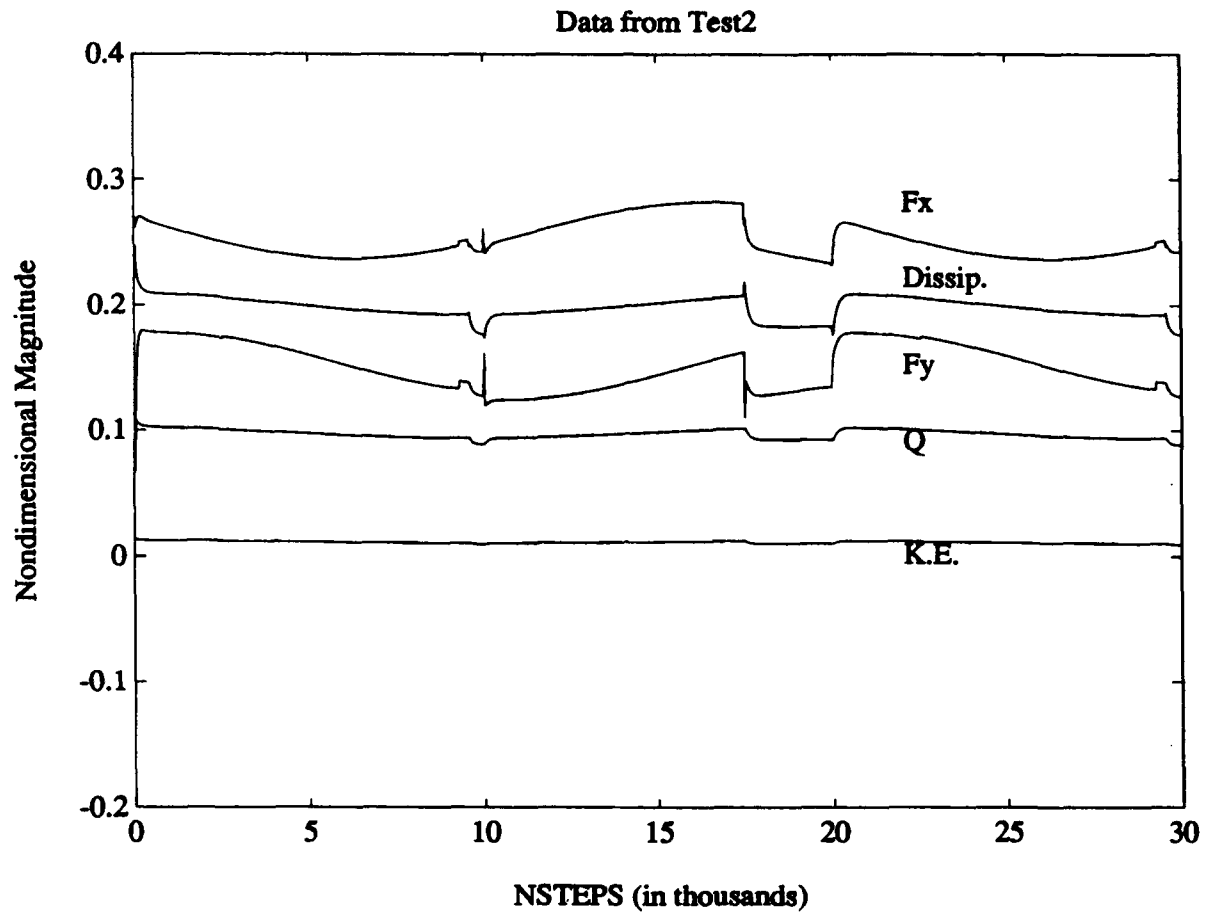
Test.iqt Plots

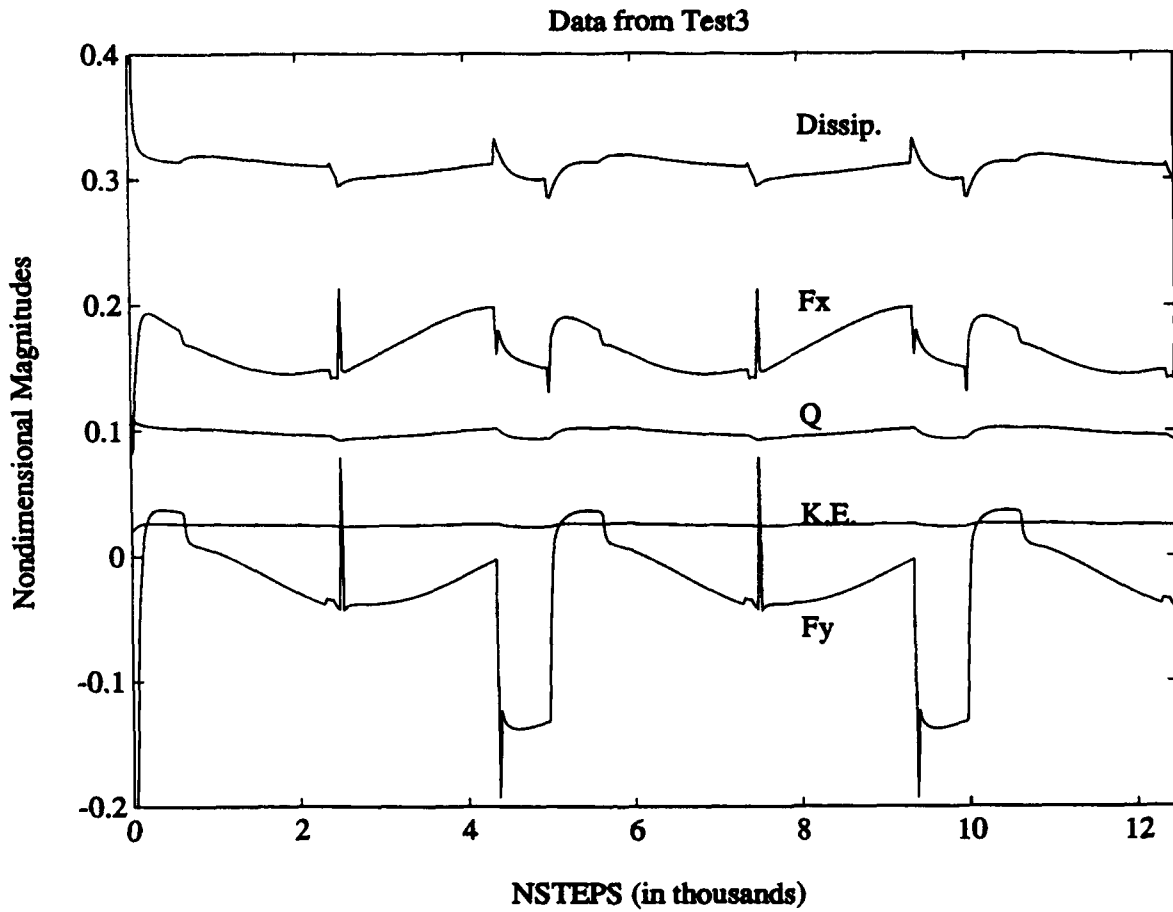
The test.iqt file is a simulation output file. It contains the time step, the force (x- and y-components) on the rotor blade, the flow rate, the dissipation and the kinetic energy. Observations and discussion of this output data are provided in detail in sections 3.5 and 4.2. As a quick reminder, the key parameters that distinguish the four test cases are:

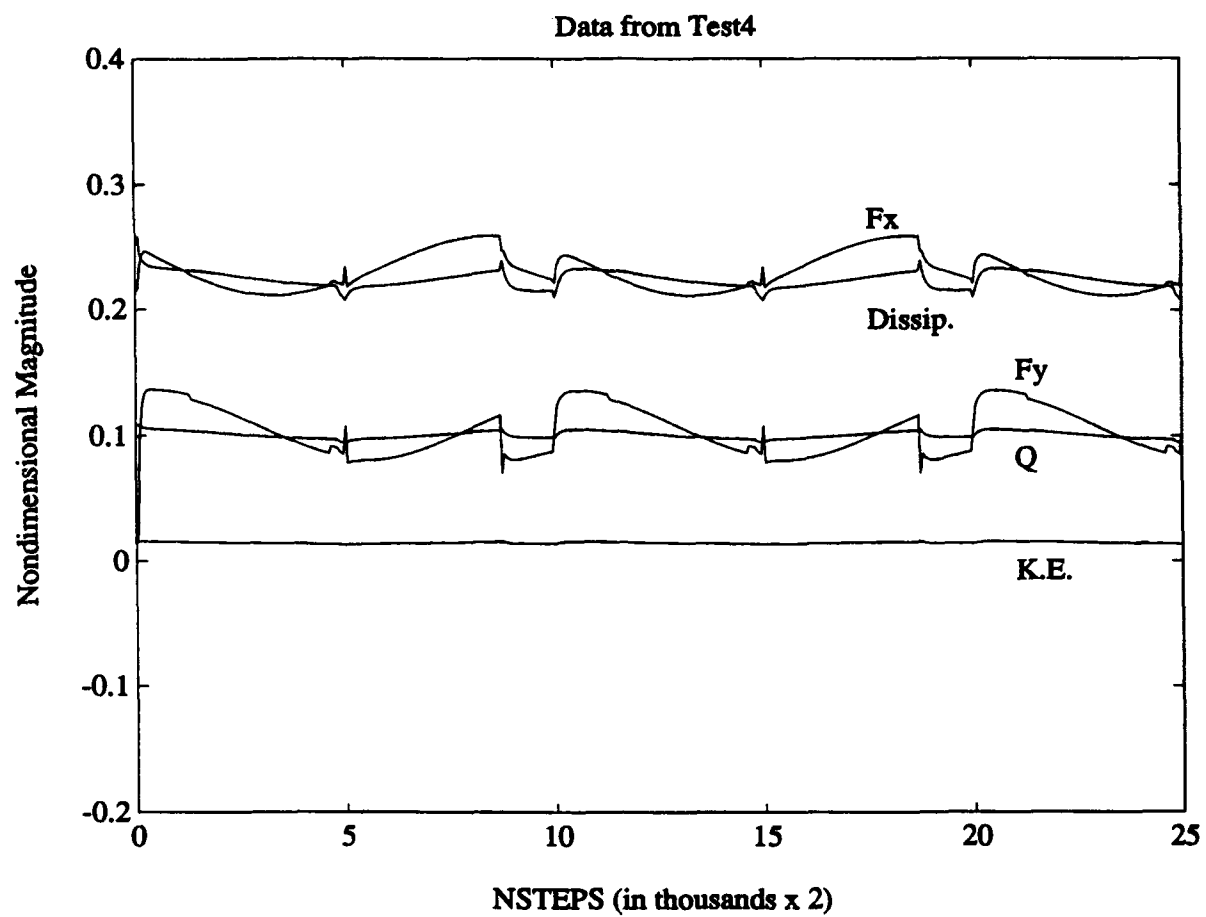
	<u>NSTEPS</u>	<u>DT</u>	<u>A</u>	<u>PERIODS</u>	<u>NSTEPS</u> <u>per</u> <u>PERIOD</u>
TEST1:	30,000	0.001	0.1	3.0	10,000
TEST2:	30,000	0.001	0.05	1.5	20,000
TEST3:	12,500	0.001	0.2	2.5	5,000
TEST4:	12,500	0.002	0.1	2.5	5,000

The output data from the test.iqt files for all four tests were plotted against time to determine periodicity of results. As the four graphs in this appendix show, steady-state was achieved after only one full period. This would not have been the case for Reynolds numbers substantially greater than 1.









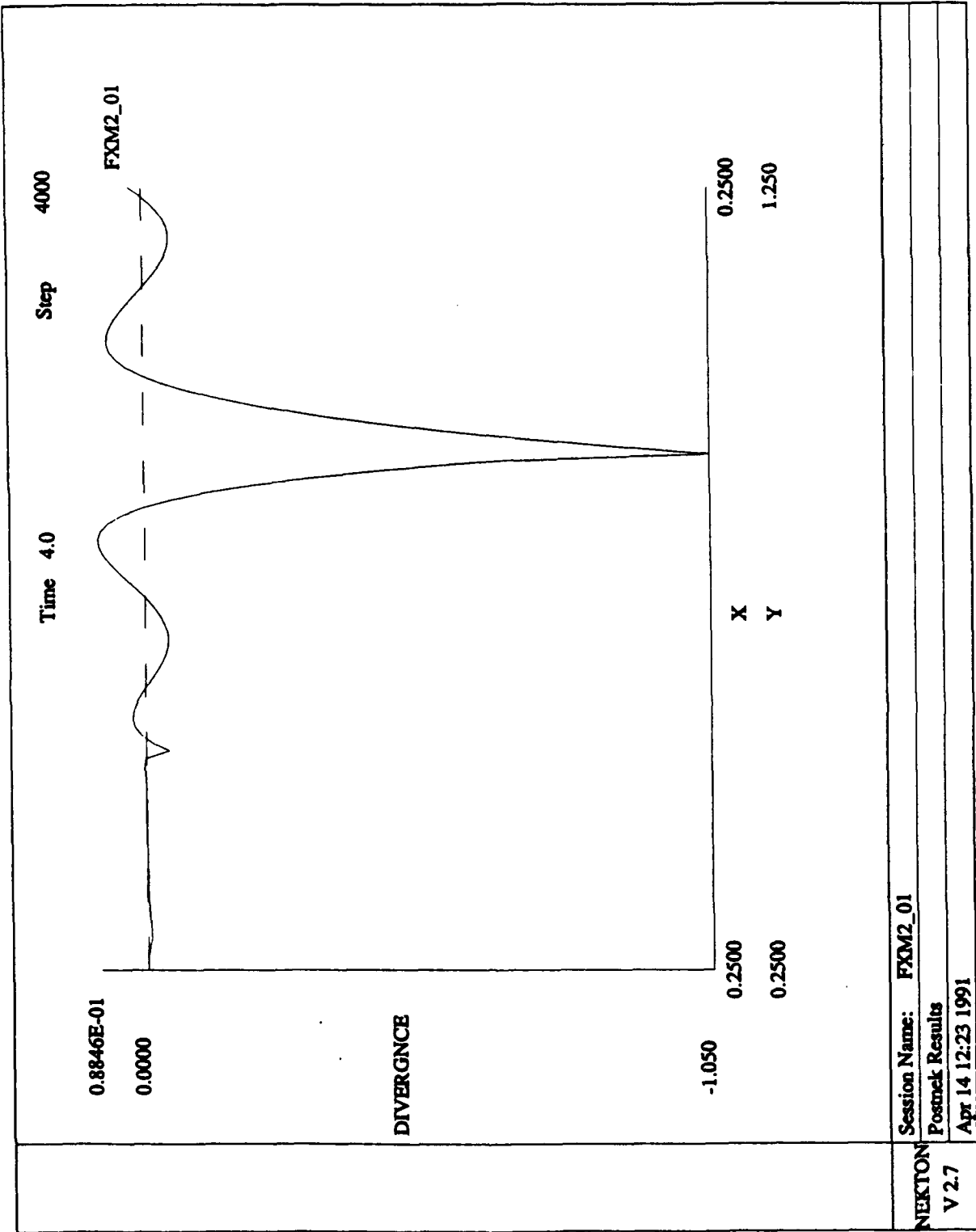
Appendix B

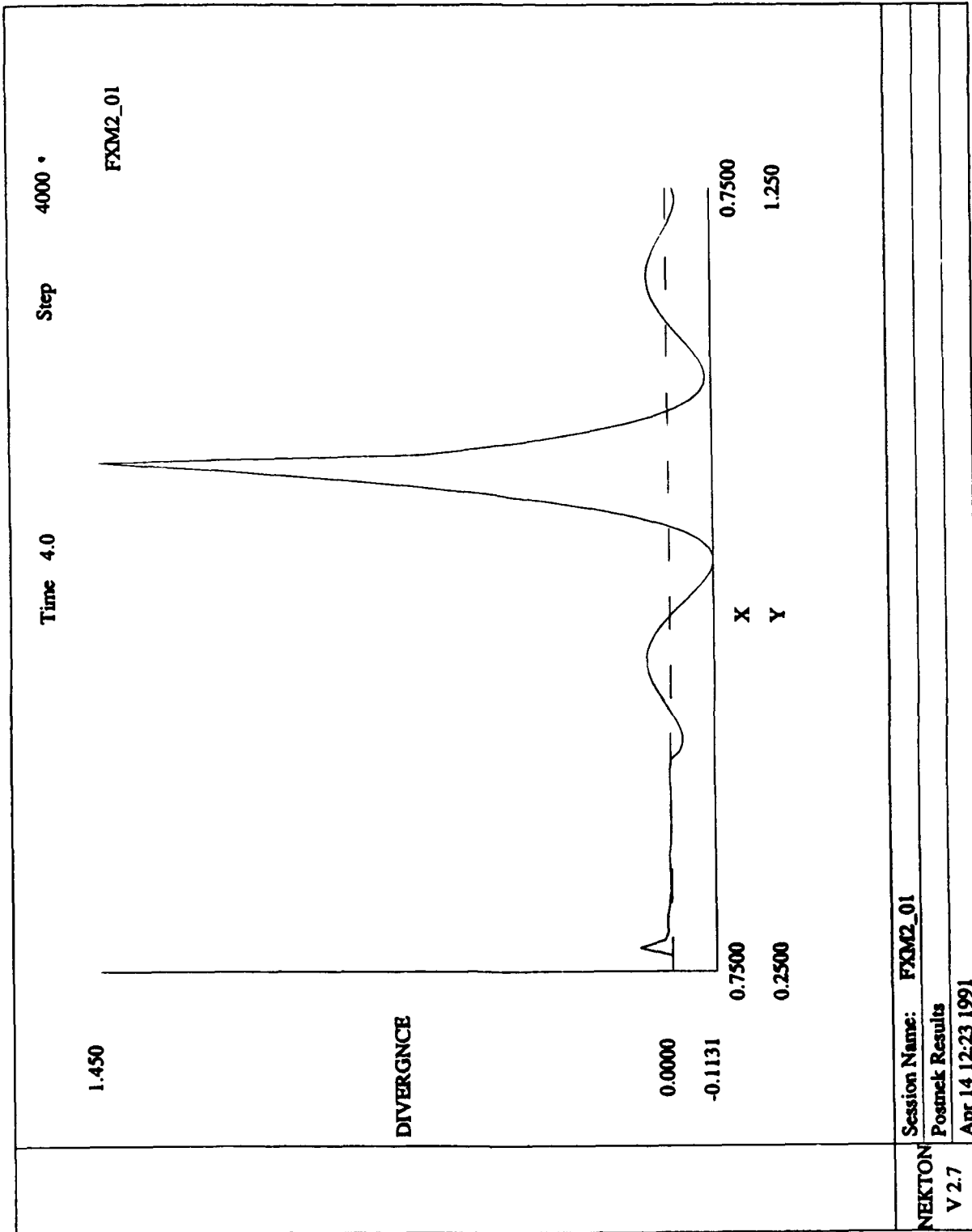
Divergence Plots

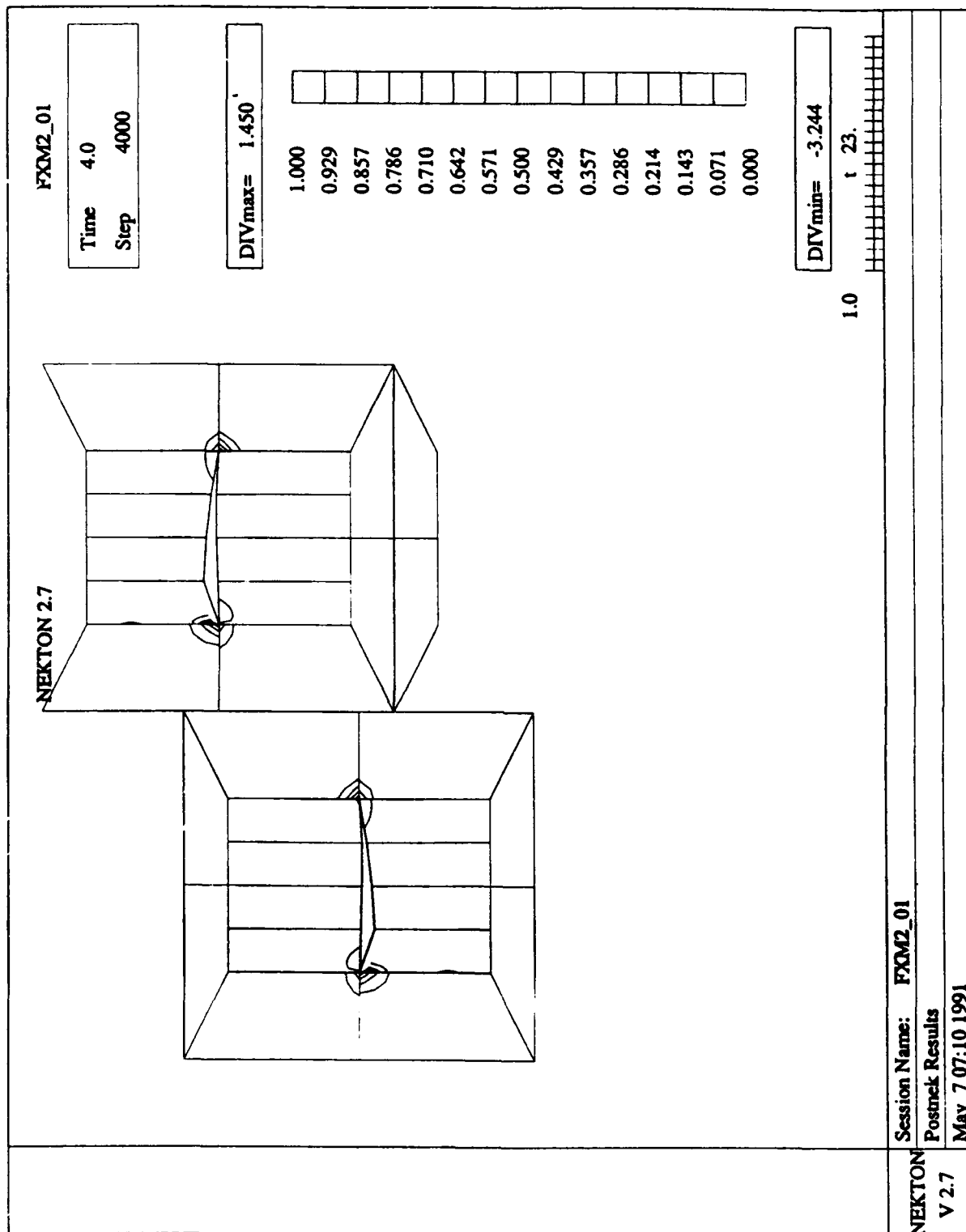
The rotor/stator configuration exists in an incompressible medium and should therefore have zero divergence everywhere. The postprocessing plots, built from the `test.rea` and `test.fld` files, reflect zero divergence virtually everywhere, with several minor exceptions. As discussed in section 4.3.1, the non-zero values for divergence can best be attributed to grid distortion and poor selection of elements at the leading and trailing edges of the blades. The graphs in this appendix show examples of the divergence in the flowfield, both as profile plots and as contour lines. The contour plots provide results for the entire control volume, while the profile plots represent data along a vertical slice of the control volume at a specified x-location. Note that each graph is annotated to indicate the particular test case and the time step at which the output data was taken from the `test.fld` file. For postprocessing purposes, the test cases (1 thru 4) are designated as:

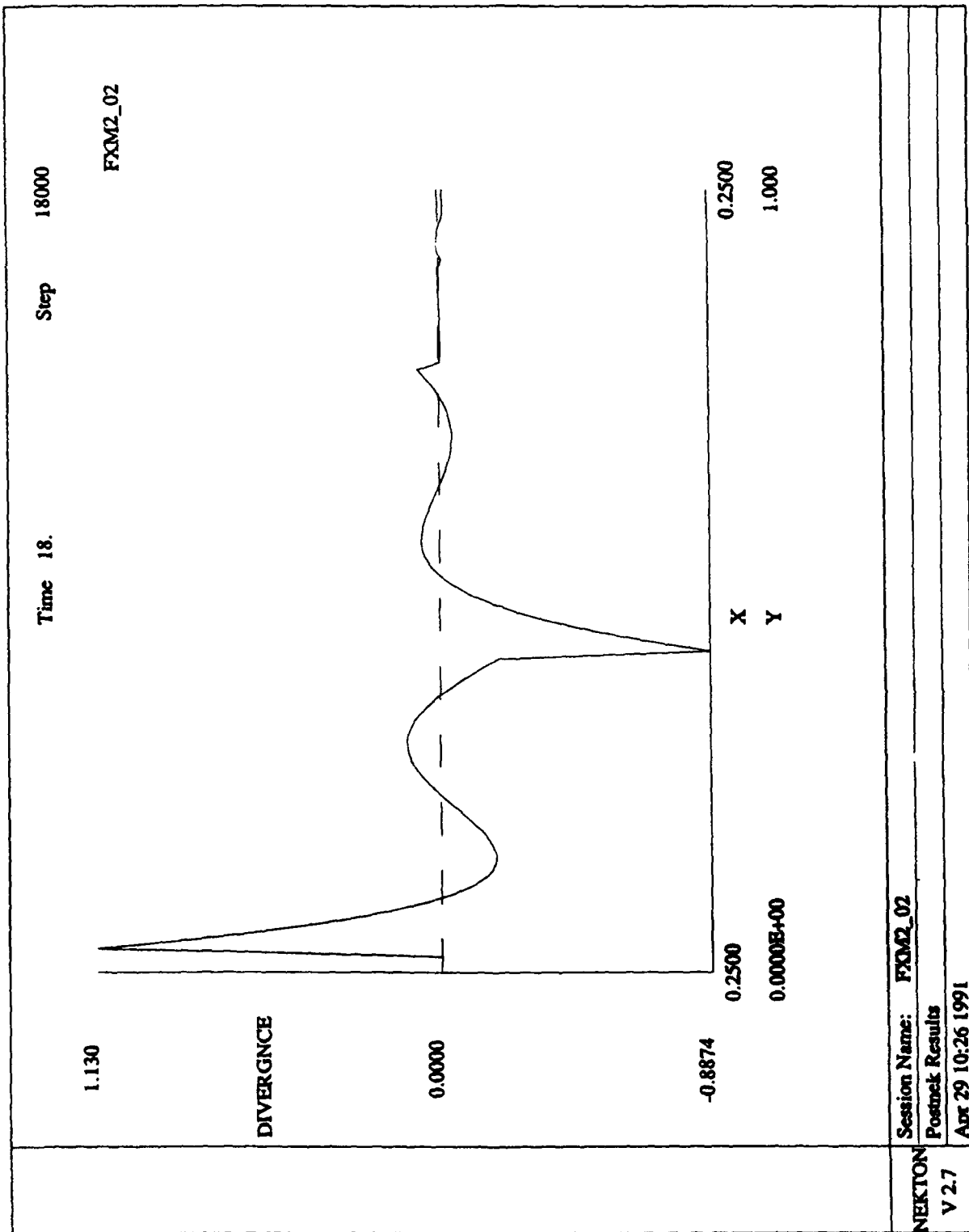
FXM2_01, FXM2_02, FXM2_03, FXM2_04 .

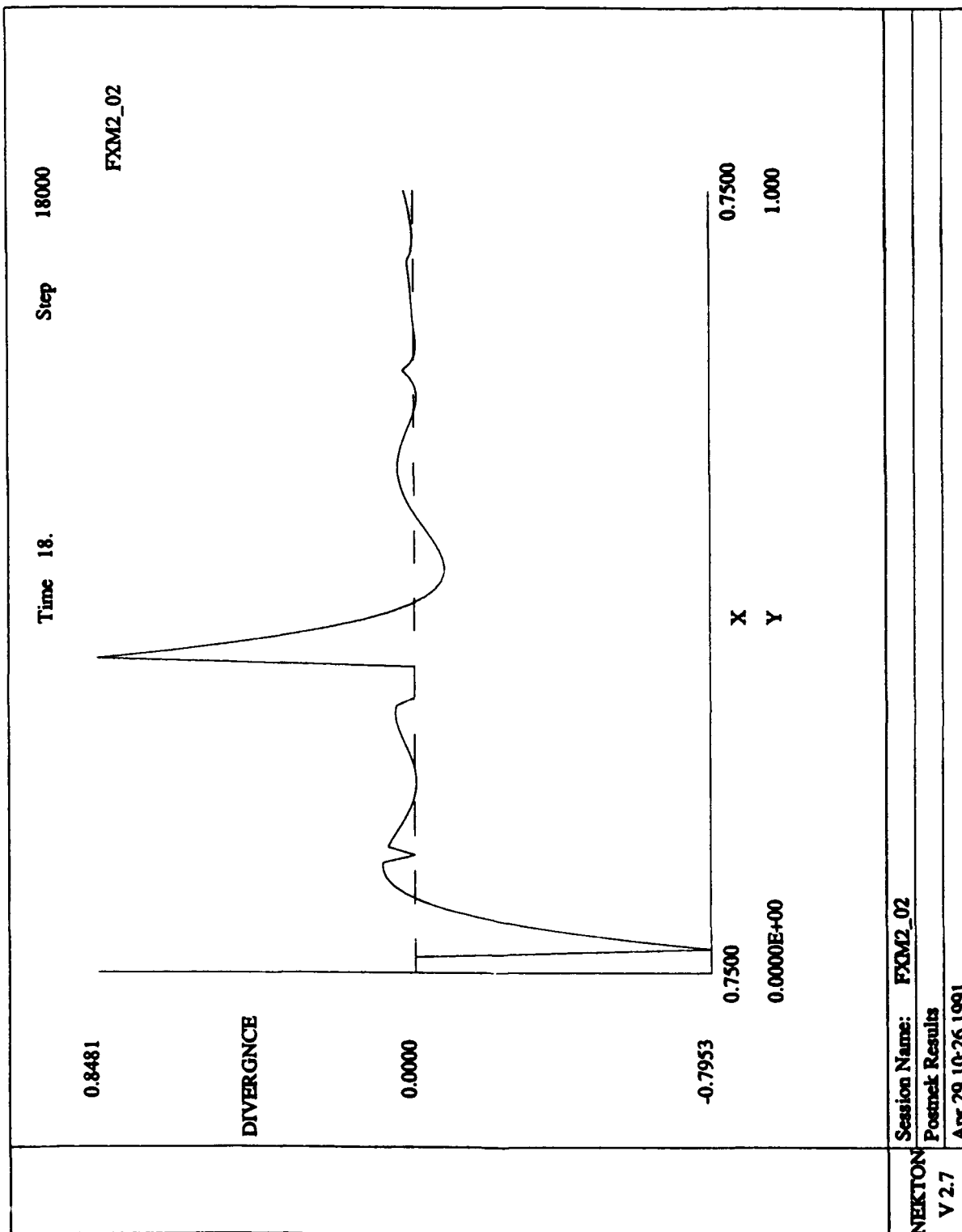
Section 4.3 explains the divergence plots, the vorticity plots and the streamlines. Additionally, it provides a discussion of the rationale for selecting the specific time steps at which plots were made.

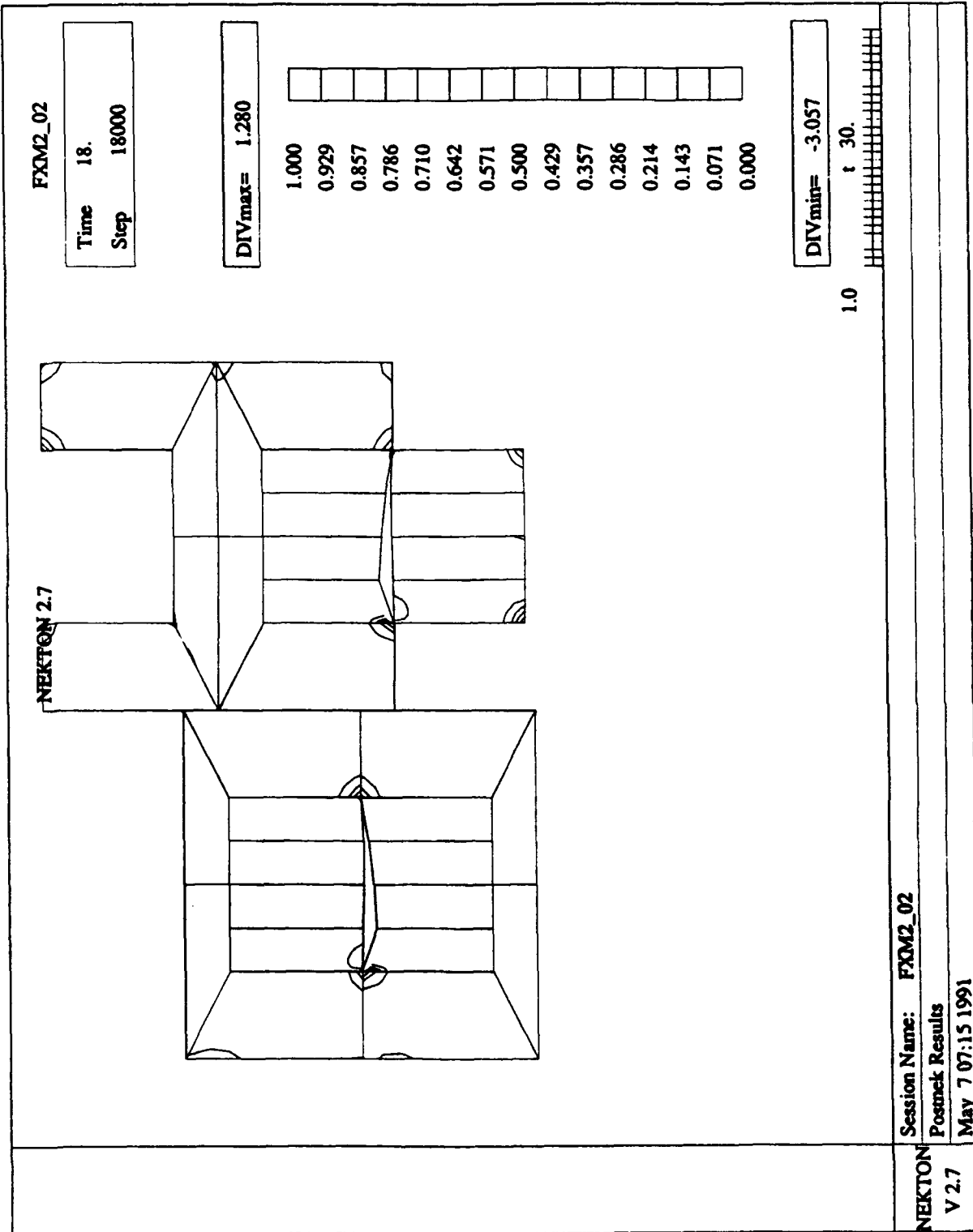


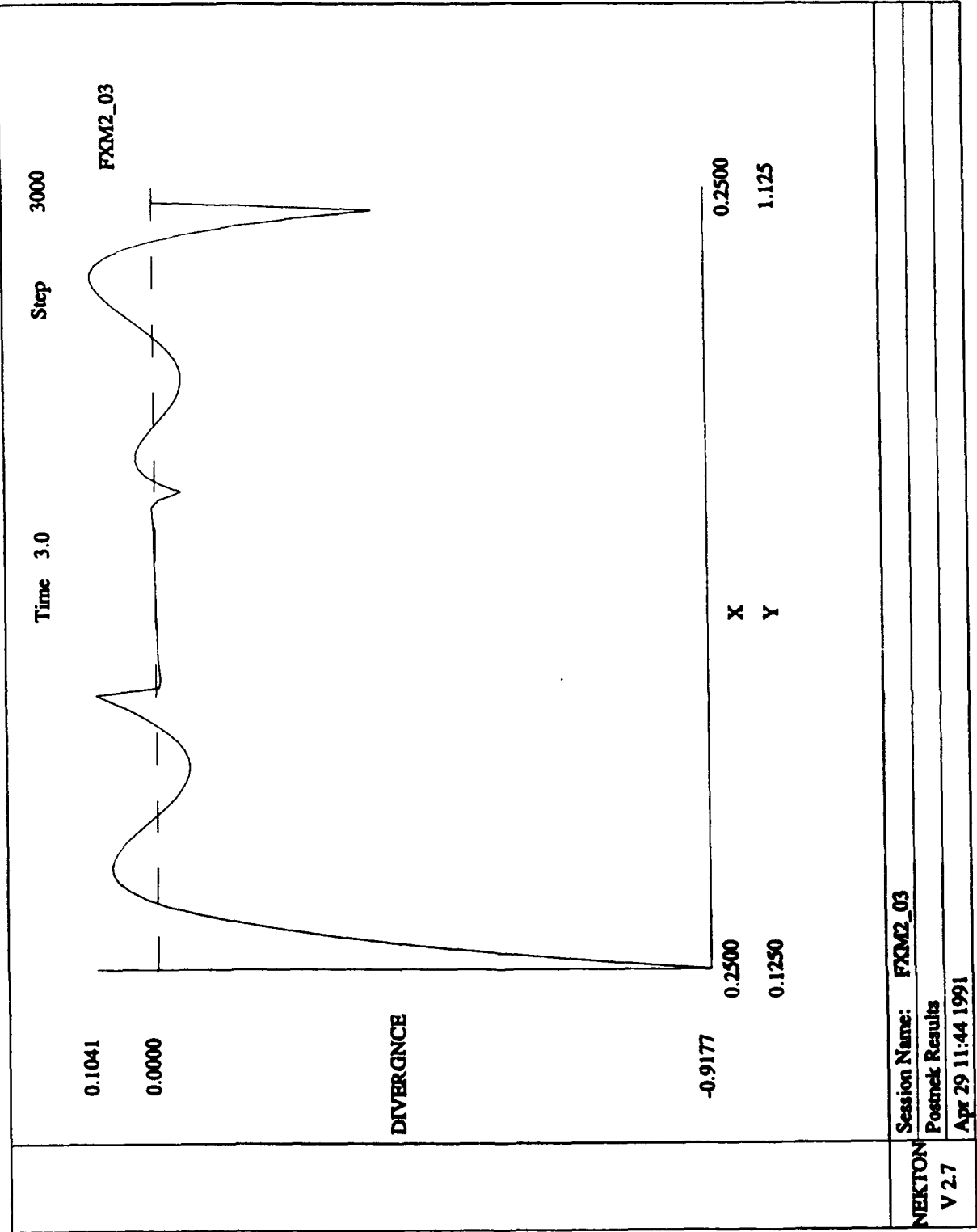


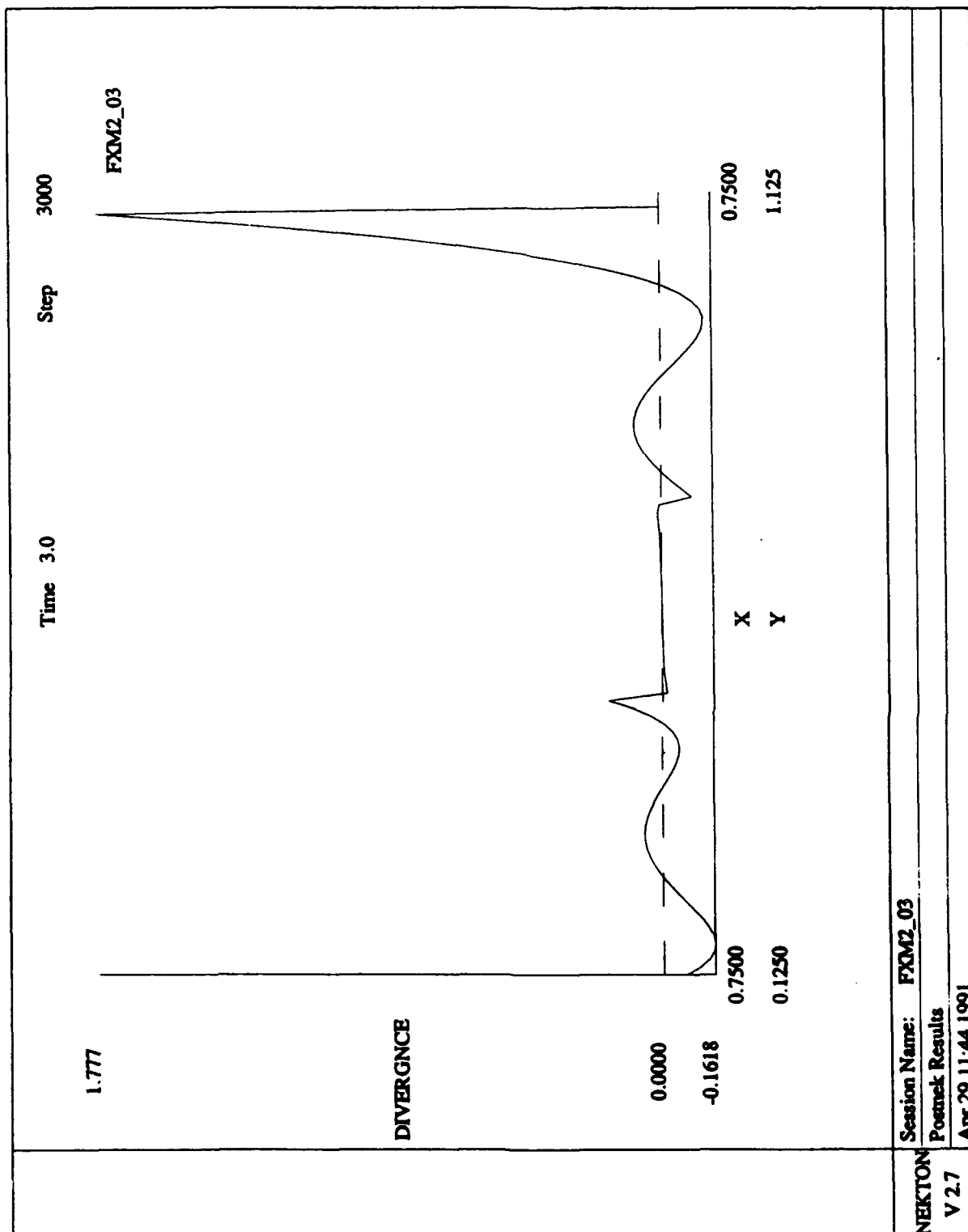








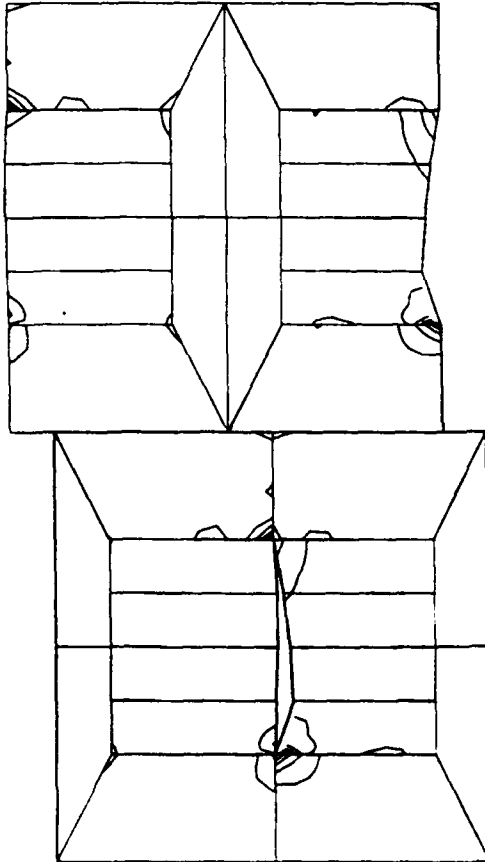




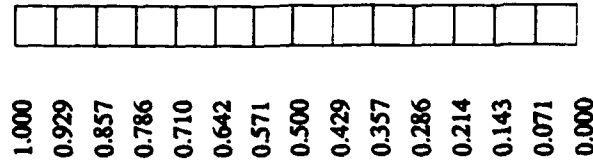
NEKTON 2.7

FXM2_03

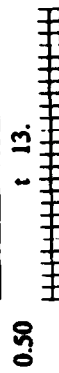
Time 3.0
Step 3000



DIVmax= 1.973



DIVmin= -3.080



Session Name: FXM2_03

Postnet Results

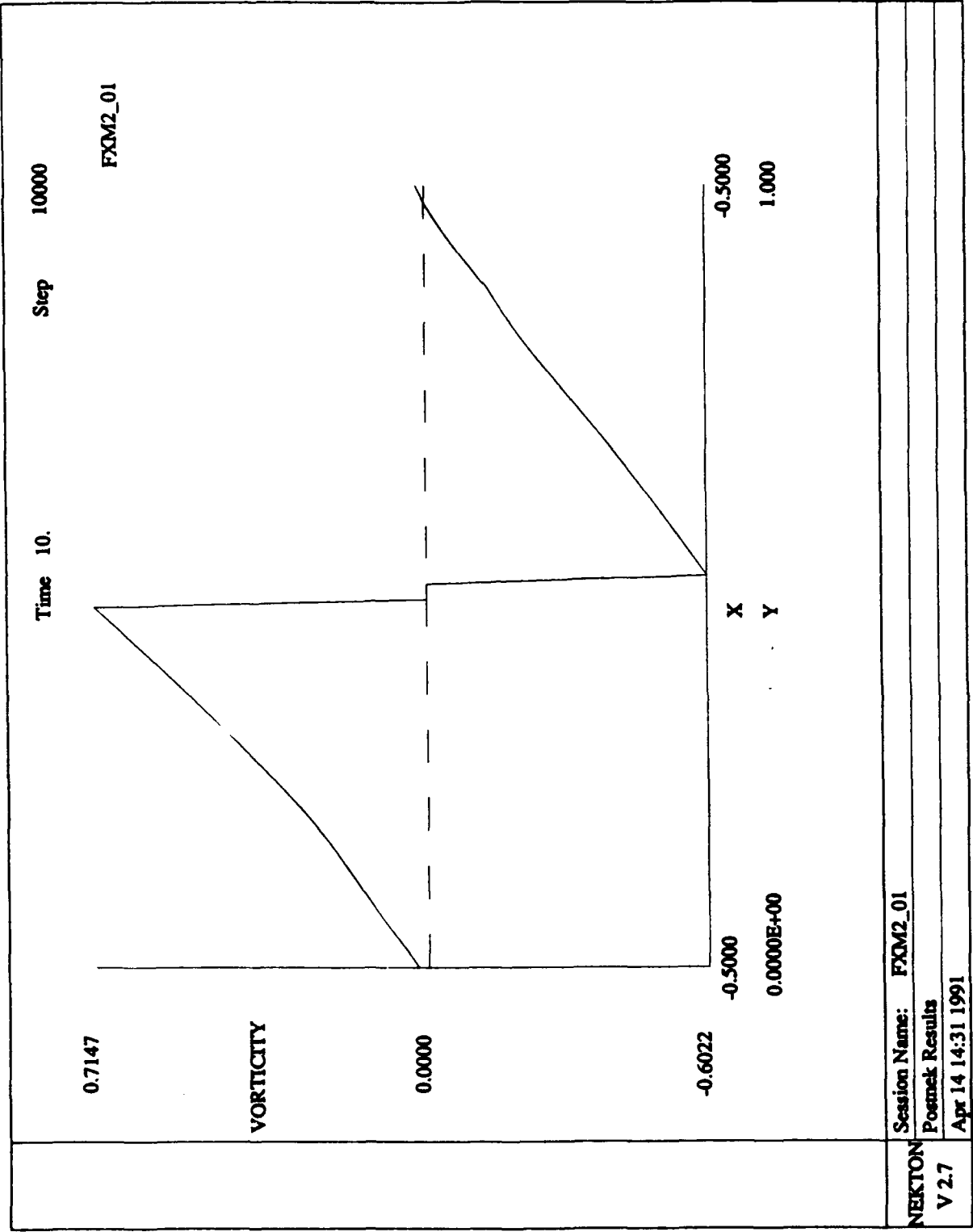
May 7 07:20 1991

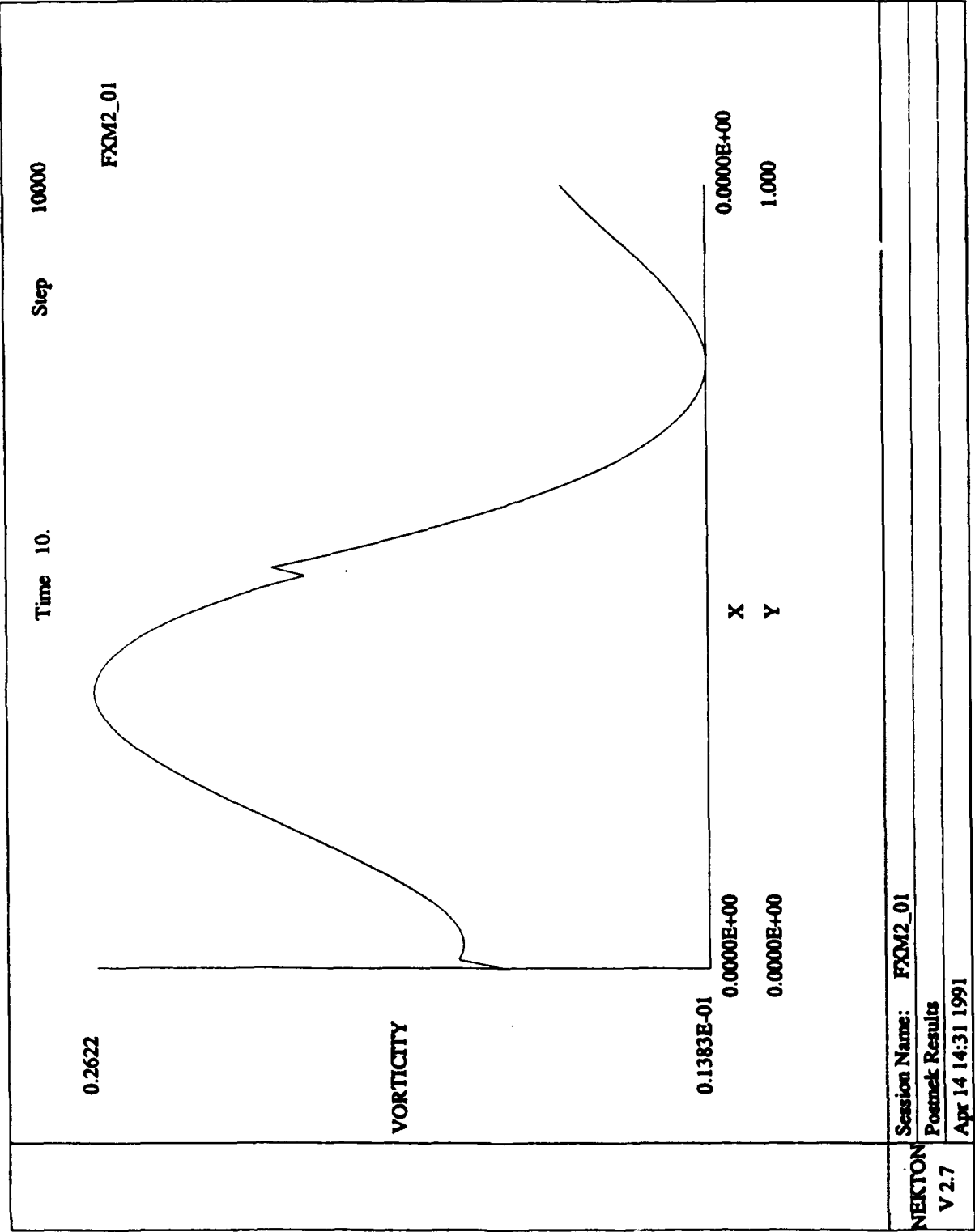
NEKTON
V 2.7

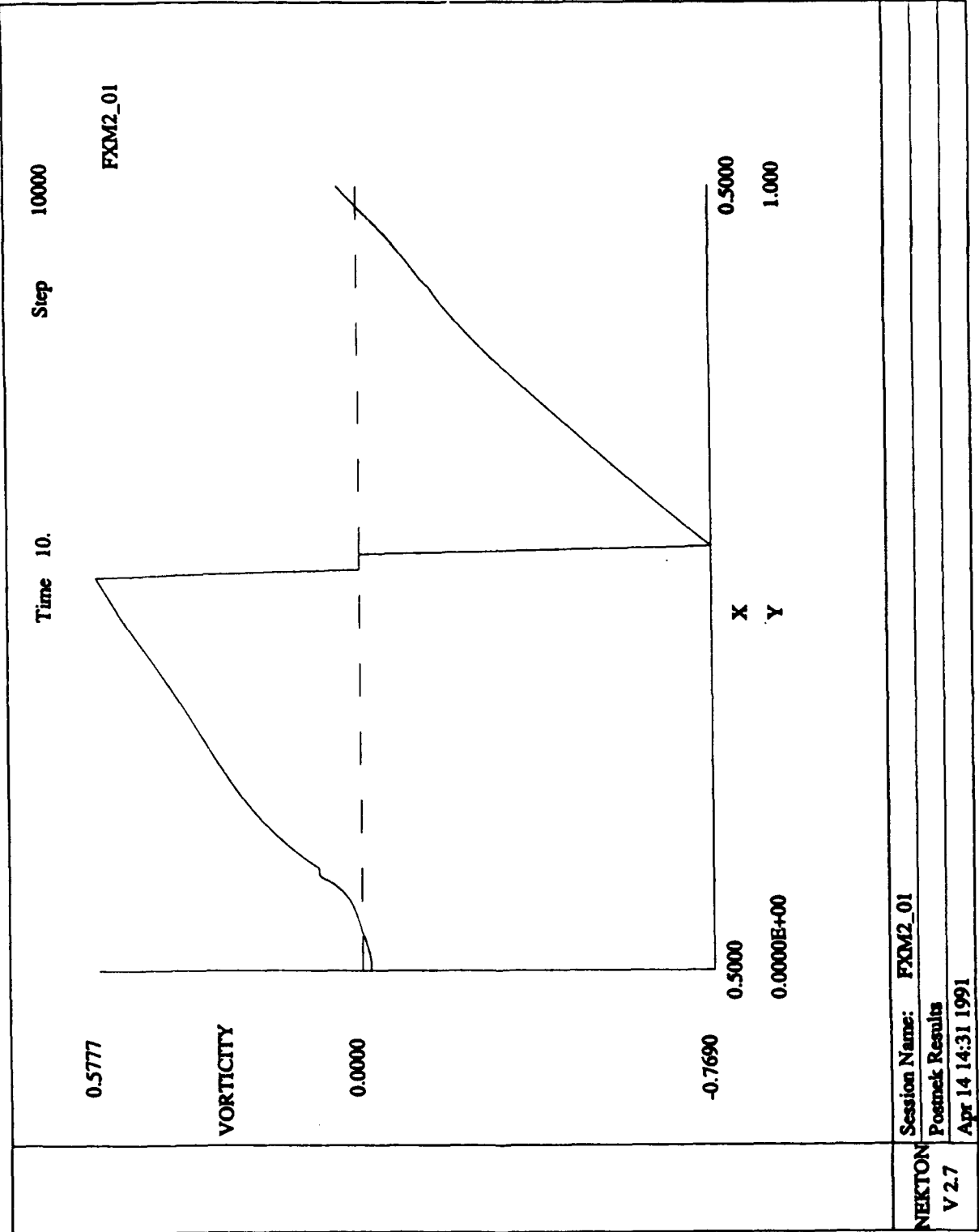
Appendix C

Vorticity Plots

The rotational nature of the flow around the rotor/stator pair is illustrated by both contour and profile plots. The graphs that follow are samples of the vorticity plots. The trends in vorticity are discussed in detail in section 4.3.2. Similar to the divergence plots, the graphs are annotated with the test case, the time step, and the upper and lower bounds of the magnitude of the vorticity at that particular time. Also included is a fishnet grid plot of the vorticity. It is provided to illustrate the curved surfaces of the blades. Due to "bug" in the postprocessor, the curved surfaces are not being plotted in the contour views as expected.



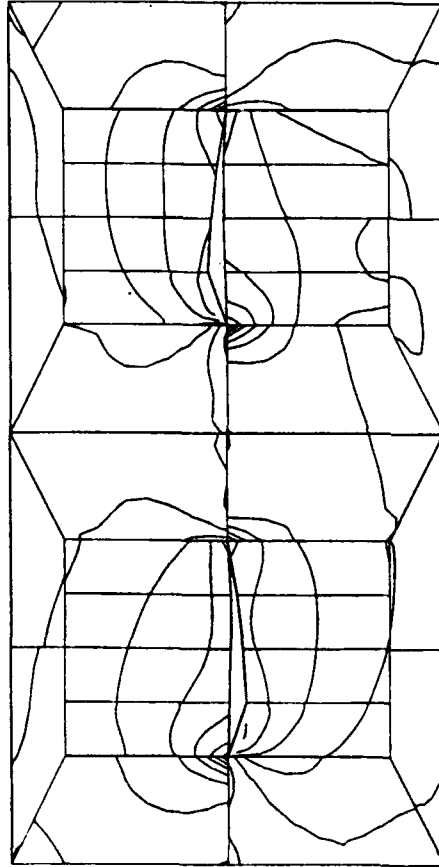




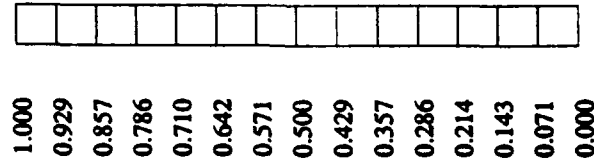
NEKTON 2.7

FXM2_01

Time 10.
Step 10000



VRmax= 1.883



VRmin= -1.766

1.0 t 23.

Session Name: FXM2_01

Postnek Results

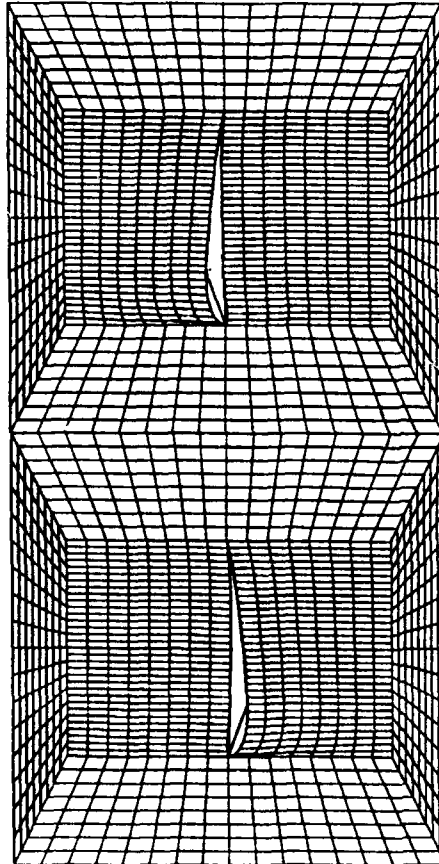
Apr 14 14:31 1991

NEKTON
V 2.7

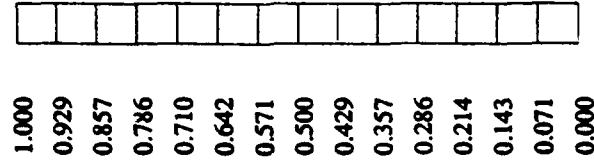
NEKTON 2.7

FXM2_01

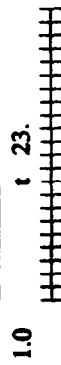
Time 10.
Step 10000



VRmax= 1.883



VRmin= -1.766

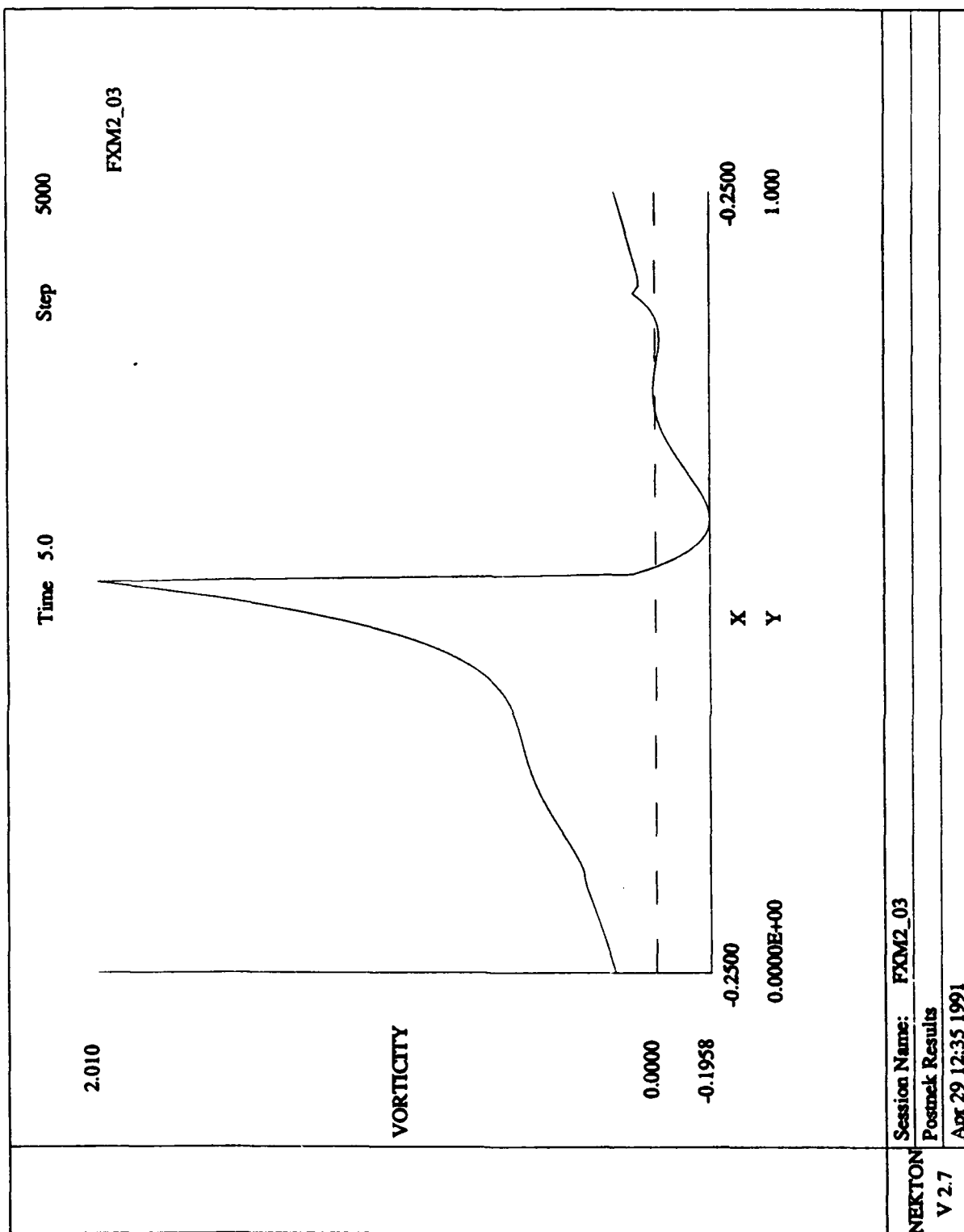


Session Name: FXM2_01

Postnek Results

Apr 14 14:31 1991

NEKTON
V 2.7

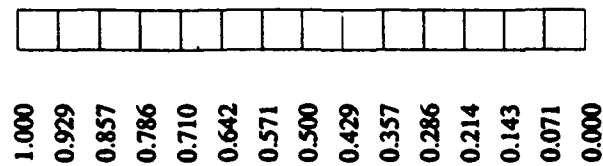
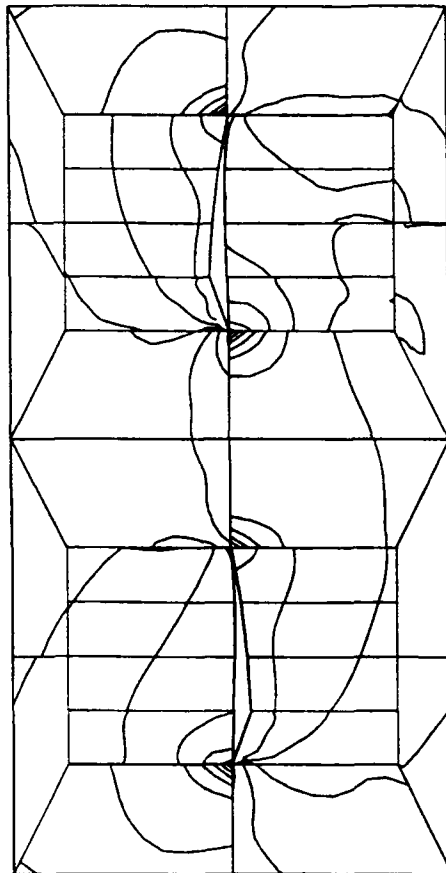


NEKTON 2.7

FXM2_03

Time 5.0
Step 5000

VRmax= 2.640



1.000
0.929
0.857
0.786
0.710
0.642
0.571
0.500
0.429
0.357
0.286
0.214
0.143
0.071
0.000

VRmin= -2.402



t 13.

0.50

Session Name: FXM2_03

Postmek Results

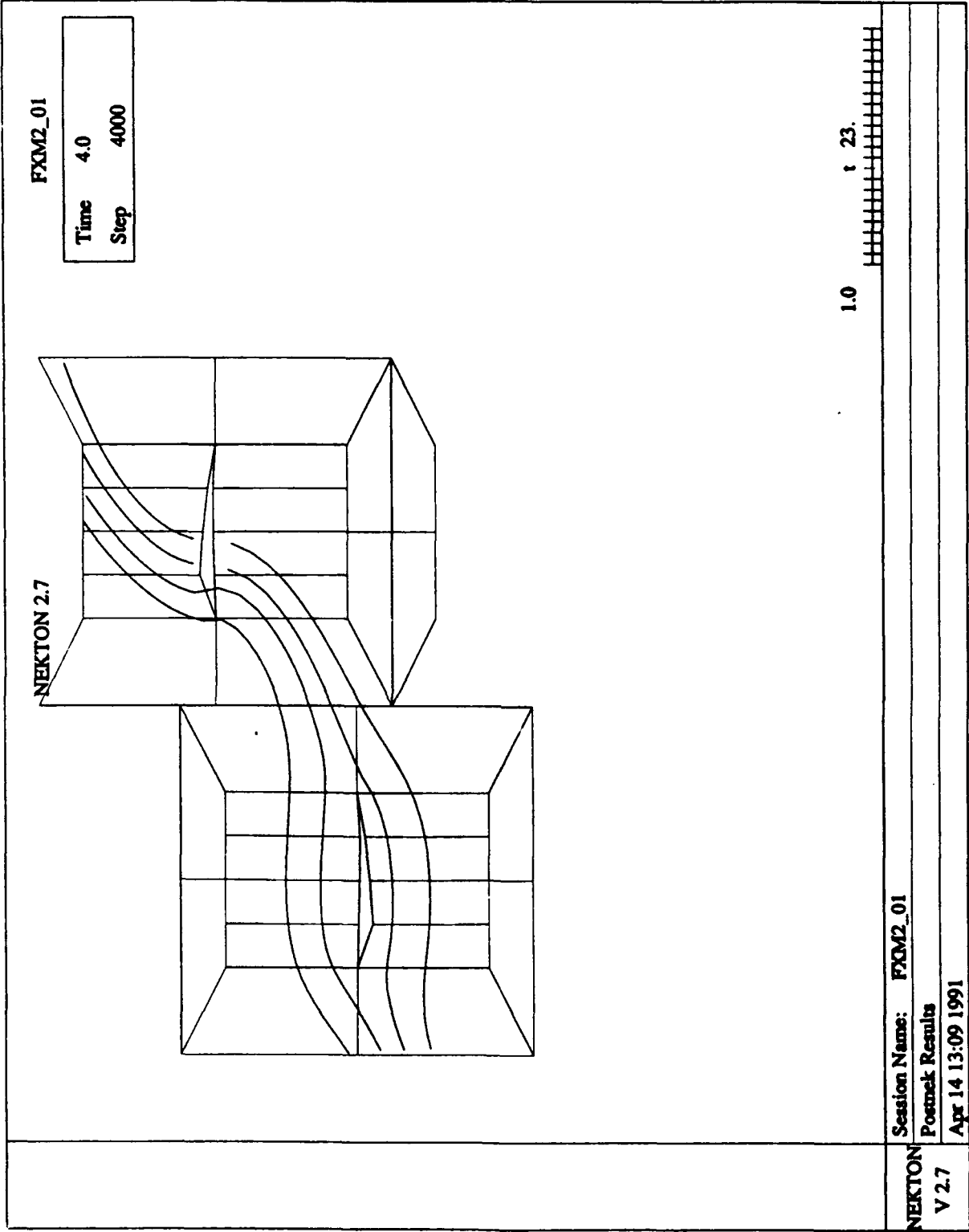
Apr 29 12:35 1991

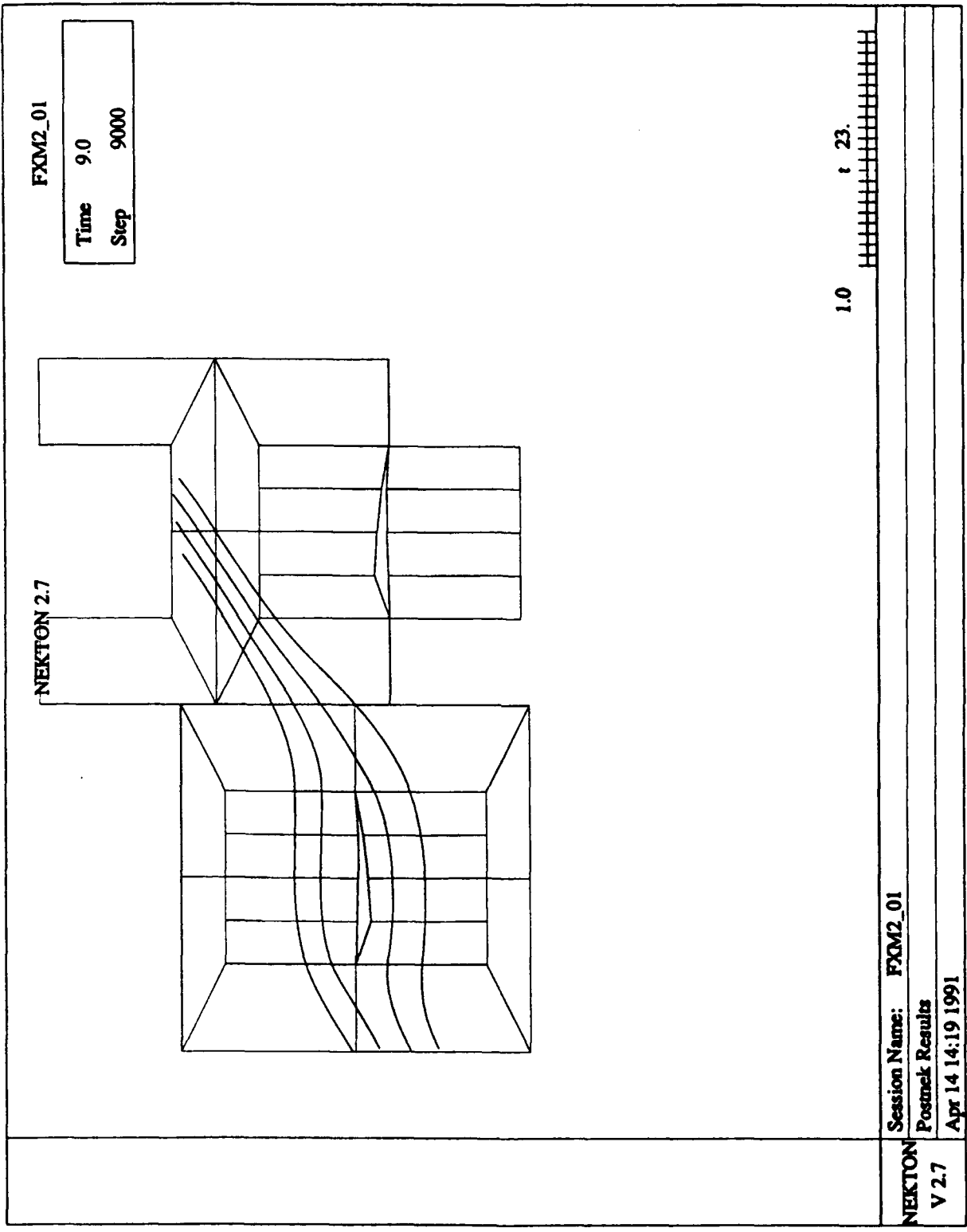
NEKTON
V 2.7

Appendix D

Streamline Plots

Streamlines were plotted at various times and locations throughout the control volume to track the wake effects from the stator onto the rotor. A full discussion of streamlines is provided in section 4.3.3.

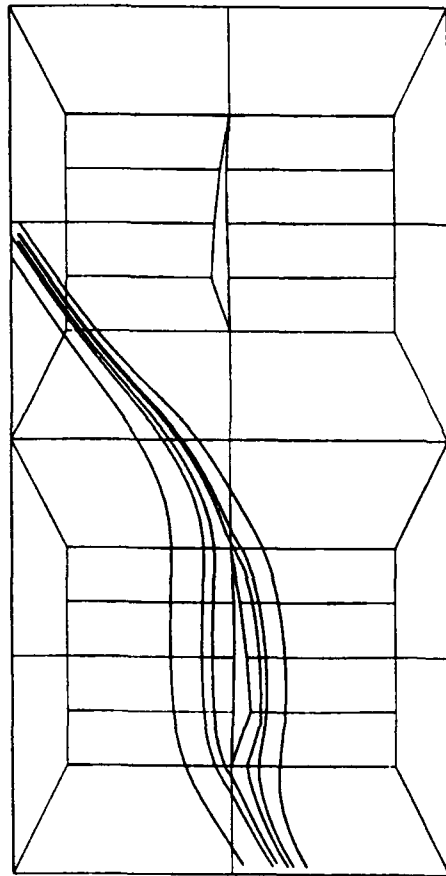




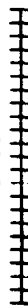
NEKTON 2.7

FXM2_01

Time	10.
Step	10000

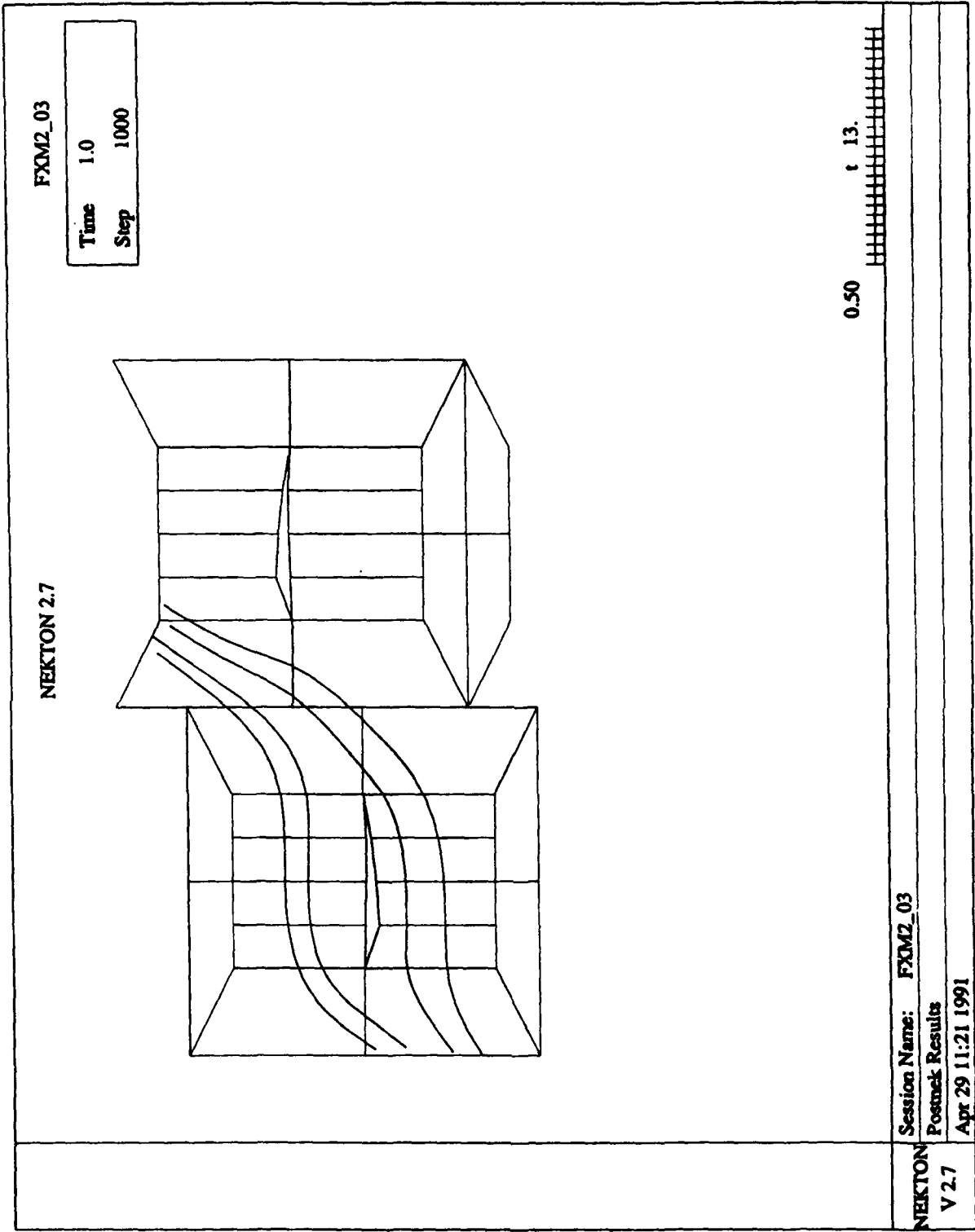


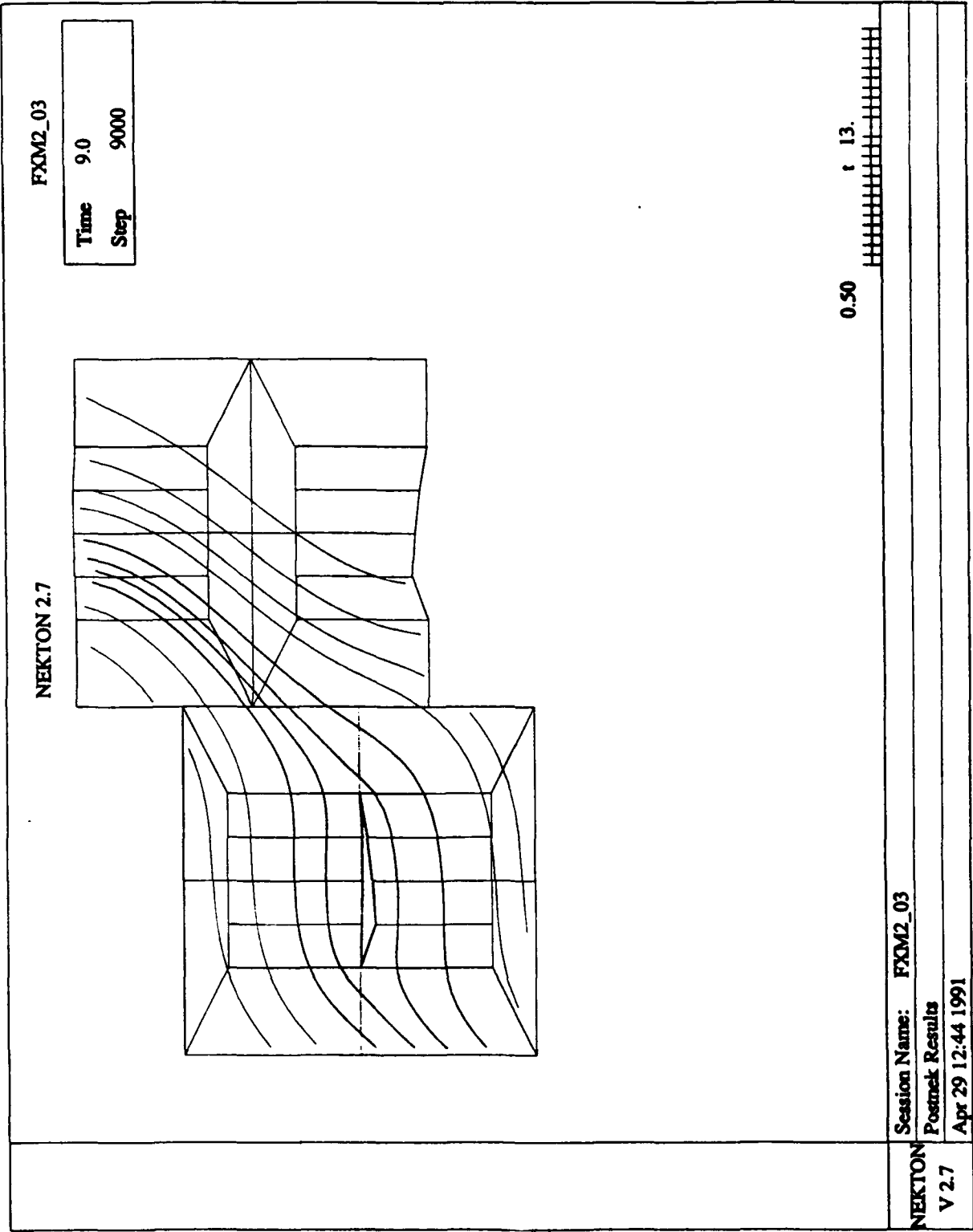
1.0 t 23.



Session Name: FXM2_01
Postnek Results
Apr 14 14:31 1991

NEKTON
V 2.7



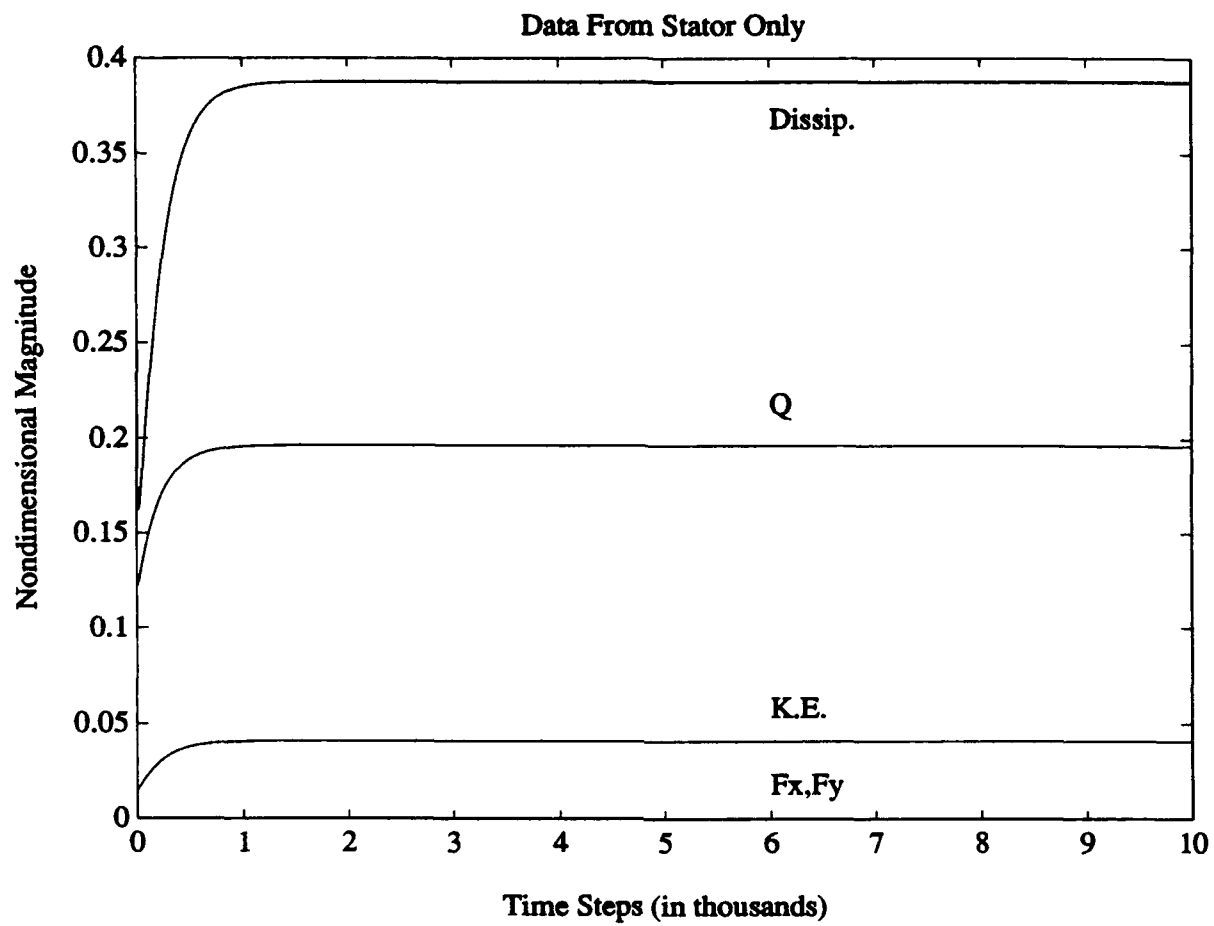


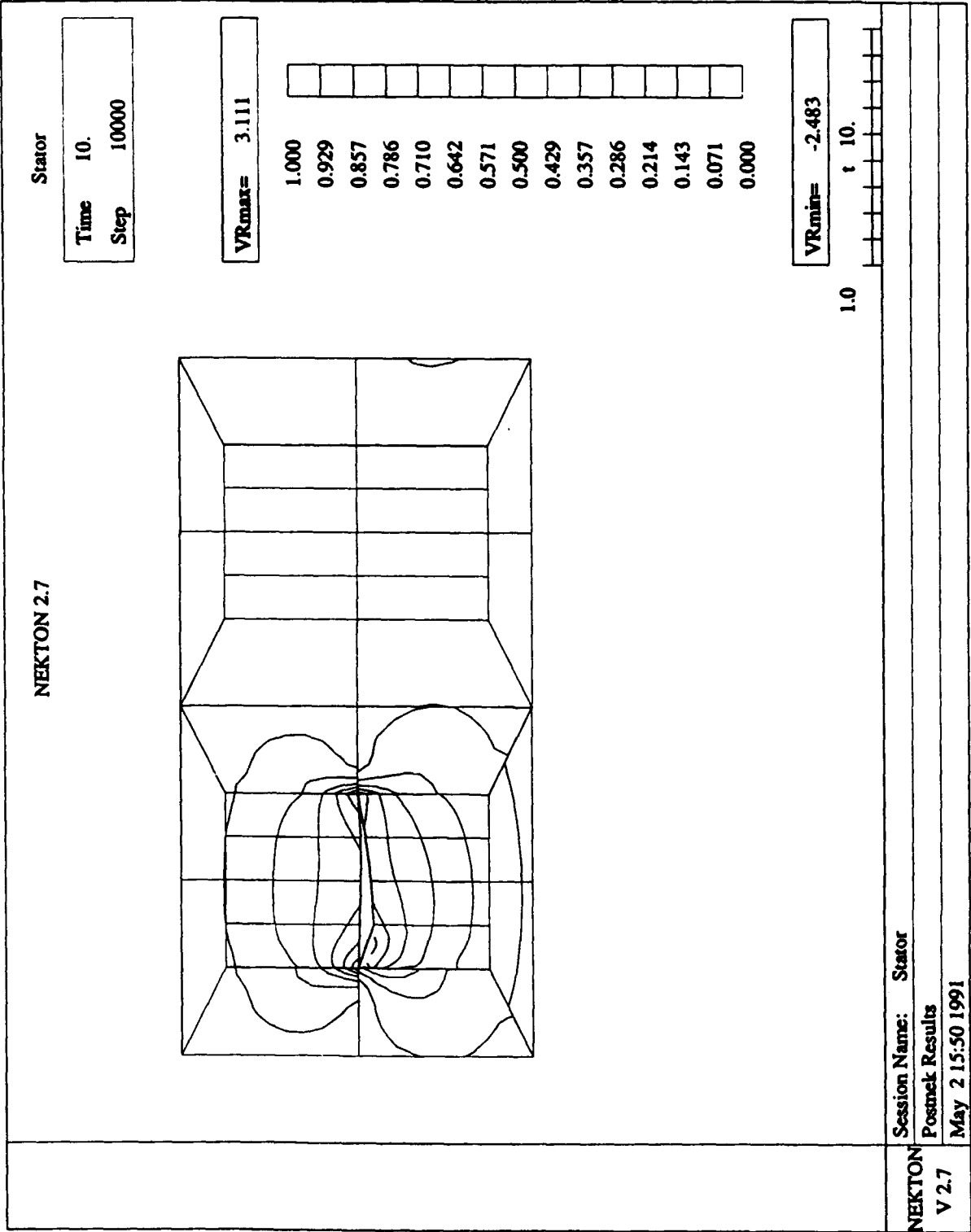
Appendix E

"Stator Only" Problem

The "stator only" problem was simulated to conduct a check of the vorticity in the other test cases. It was run in a manner similar to the first 4 test cases, except that the rotor blade was removed. The results were very positive, with zero vorticity in that half of the control volume that contained no rotor blade. As an additional bonus, the results of the "stator only" problem proved to be extremely beneficial when validating the negative lifting force on the rotor blade in the rotor/stator problem. The streamlines from the "stator only" problem show that the natural rotor advance coefficient should be less than 0.05 in order to generate positive lift for the given conditions. See section 4.3.3 for a complete discussion of this topic. The graphs that follow in this appendix, in order, are:

- test.iqt data
- contour plot of vorticity
- two sets of streamlines
- and, a contour plot of the pressure.

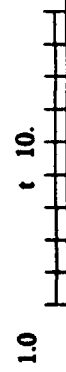
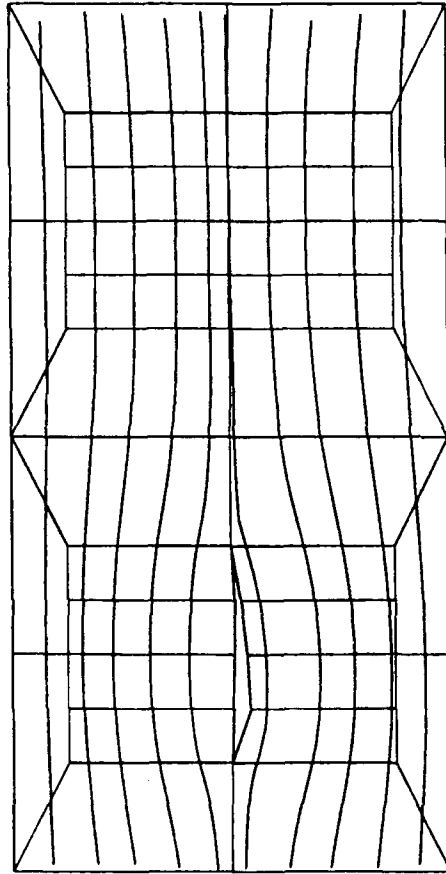




NEKTON 2.7

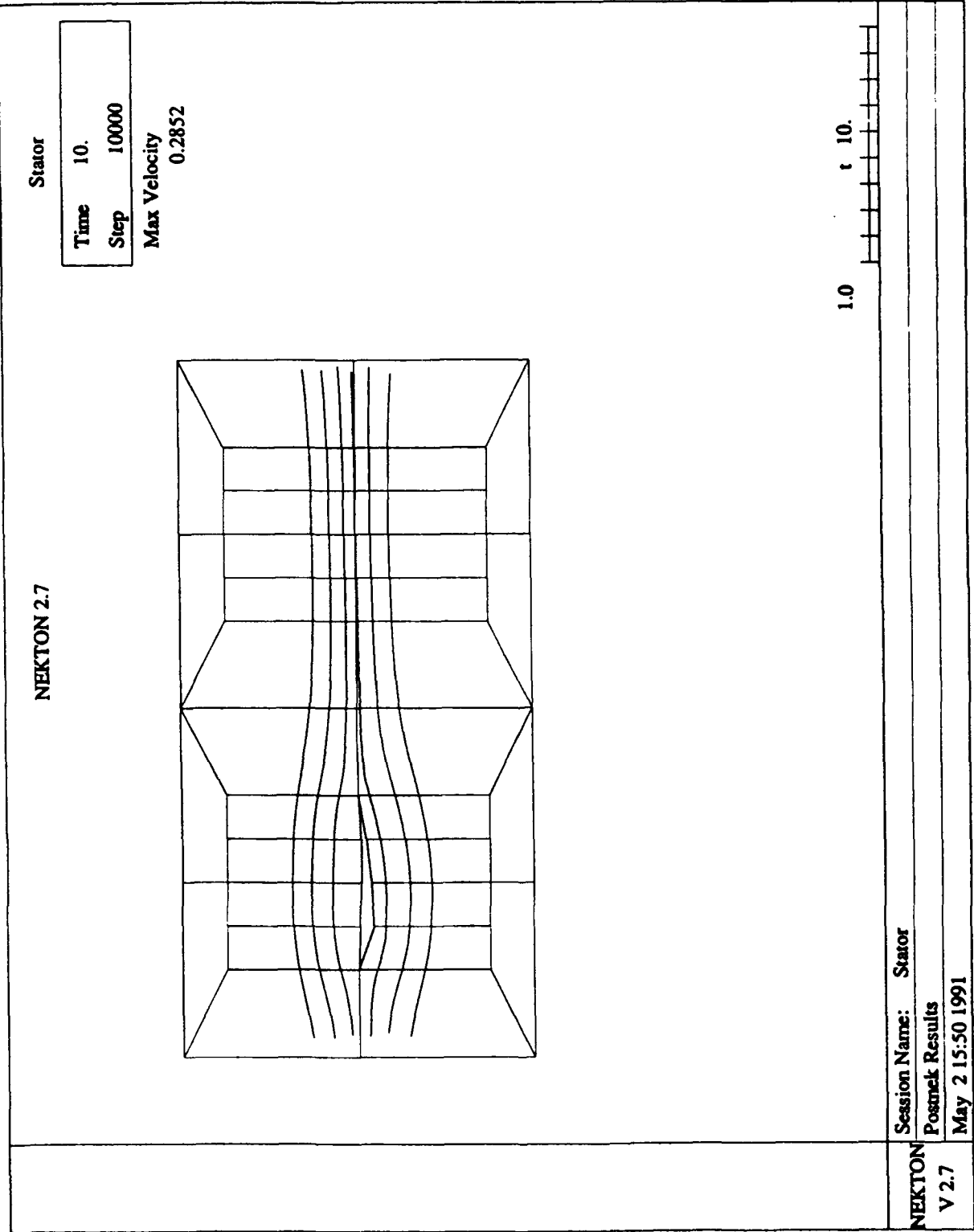
Stator

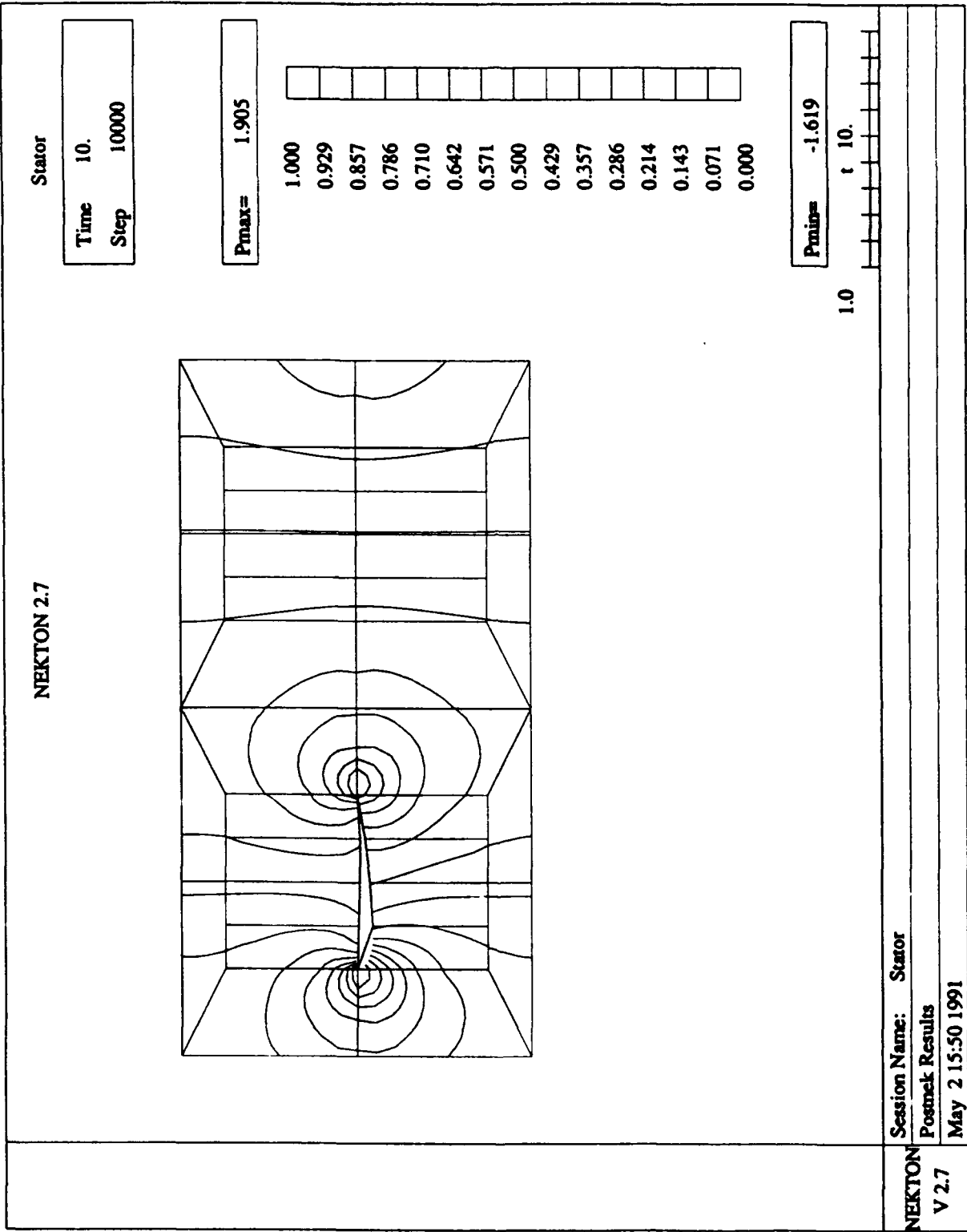
Time	1.0
Step	1000



Session Name: Stator
Postnet Results
May 2 15:50 1991

NEKTON
V 2.7



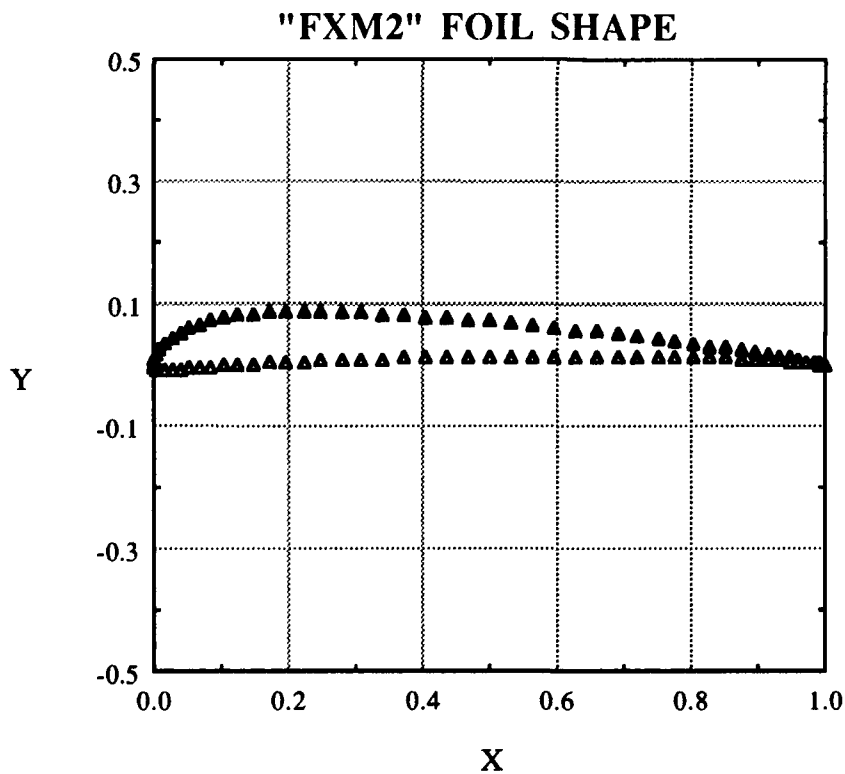


Appendix F

Blade Section

(FXM2 Airfoil)

The FXM2 airfoil shape was selected from reference [15], *A Catalog of Low Reynolds Number Airfoil Data for Wind Turbine Applications*. The low Reynolds numbers referred to in this reference are on the order of 10^4 . For comparative purposes, the Reynolds number of 1 used in the test cases is far below any real world applications. The FXM2 airfoil shape is plotted below.



Bibliography

- [1] I.H. Abbott & A.E. von Doenhoff (1959). *Theory of Wing Sections*, Dover Publications, Inc., New York, NY.
- [2] G. Anagnostou (1991). Nonconforming Sliding Spectral Element Methods for the Unsteady Incompressible Navier-Stokes Equations. *Ph.D. Thesis*, Massachusetts Institute of Technology, Cambridge, MA.
- [3] J.D. Anderson (1988). Introduction to Computational Fluid Dynamics. *von Karmen Institute for Fluid Dynamics, Lecture Series 1988-02*, pp. 1-267.
- [4] Y.S. Chen (1988). 3-D Stator-Rotor Interaction of the SSME. *AIAA Paper 88-3095*.
- [5] J.F. Douglas, J.M. Gasiolek, J.A. Swaffield (1985). *Fluid Mechanics*, 2nd Edition, Longman Scientific & Technical, Essex, England.
- [6] J. Gleick (1987). *Chaos (Making a New Science)*, Penguin Books, New York, NY.
- [7] E. Gode & R. Cuenod (1990). Numerical Simulation of Flow in a Hydraulic Turbine. *Chemical Engineering Progress*, Vol. 86/No. 8, August 1990, pp. 35-41.
- [8] J.P. Gostelow (1984). *Cascade Aerodynamics*, Pergamon Press Inc., Elmsford, NY.
- [9] K.L. Gundy-Burlet, M.M. Rai, R.P. Dring (1989). Two-Dimensional Computations of Multi-Stage Compressor Flows Using a Zonal Approach. *AIAA Paper 89-2452*.
- [10] L.W. Ho (1979). Manufacture and Evaluation of a Five-Kilowatt Axial-Flow Water Turbine. *Masters Thesis*, Massachusetts Institute of Technology, Cambridge, MA.
- [11] J.H. Horlock (1958). *Axial Flow Compressors (Fluid Mechanics and Thermodynamics)*, Butterworths Scientific Publications, London, England.

- [12] N.K. Madavan, P. Kelaita, S. Gavali (1990). Supercomputer Applications in Gas Turbine Flowfield Simulation. *The International Journal of Supercomputer Applications*, Volume 4, No. 2, Summer 1990, pp. 81-95.
- [13] J.A. Marquis (1983). Design and Evaluation of Small Water Turbines. Final Report Prepared for: U.S. Department of Energy, Under Contract No. DE-FG44-80R410243.
- [14] R. Peyret, T.D. Taylor (1983). *Computational Methods for Fluid Flow*, Springer-Verlag New York Inc., New York, NY.
- [15] S.J. Miley (1982). *A Catalog of Low Reynolds Number Airfoil Data for Wind Turbine Applications*. Texas A&M University, College Station, Texas.
- [16] M.M. Rai (1985). Navier-Stokes Simulations of Rotor-Stator Interaction Using Patched and Overlaid Grids. *AIAA Paper* 85-1519.
- [17] M.M. Rai (1987). Unsteady Three-Dimensional Navier-Stokes Simulations of Turbine Rotor-Stator Interactions. *AIAA Paper* 87-2058.
- [18] M.M. Rai & N.K. Madavan (1988). Multi-Airfoil Navier-Stokes Simulations of Turbine Rotor-Stator Interaction. *AIAA Paper* 88-0361.
- [19] R.H. Sabersky, A.J. Acosta, E.G. Hauptmann (1989). *Fluid Flow, A First Course in Fluid Mechanics*, 3rd Edition, Macmillan Publishing Company, New York, NY.
- [20] O.P. Sharma, E. Renaud, T.L. Butler, K. Milsaps, R.P. Dring, H.D. Joslyn (1988). Rotor-Stator Interaction in Multi-Stage Axial-Flow Turbines. *AIAA Paper* 88-3013.
- [21] D.G. Wilson (1984). *The Design of High-Efficiency Turbomachinery and Gas Turbines*, The MIT Press, Cambridge, MA.