

Effects of surface heating on the laminar boundary layer of a two-dimensional, Mach three wind tunnel nozzle
by Torence P Brogan

A thesis submitted in partial fulfillment of the requirements for the degree of Master of Science in Mechanical Engineering
Montana State University
© Copyright by Torence P Brogan (1995)

Abstract:
The effects of surface heating on the mean laminar boundary layer on a supersonic wind-tunnel nozzle have been investigated. Experimental measurements and theoretical predictions were compared with an emphasis on the comparison with and without surface heating. Fifty-one profiles of pitot pressure and total temperature were measured in various combinations of tunnel stagnation conditions, surface temperature, and a turbulence screen. A total of 24 profile pairs were acquired at several streamwise locations with wall-to-total temperature ratios of about 1.10 at the throat. Overall, the theory predicted the boundary layer development well both with and without surface heating. All trends predicted by the theory were present in the experimental results. Velocity profiles obtained by experiment and by theory showed good agreement when the normal coordinate was plotted as $\gamma l \delta$ where $y\gamma$ is the distance from the wall, and δ is the boundary layer thickness. In both experiment and theory, the surface heating produced a very small effect on the boundary layer. In view of the observed stabilizing effect that surface heating has on supersonic nozzles, it has been concluded that the boundary layer stability is extremely sensitive to the surface temperature. Although heat was concentrated at the throat, numerical computations showed that the surface heat flux was directed into the boundary layer for all streamwise locations along the nozzle, therefore, the observed stabilizing effects of surface heating on the wind-tunnel nozzle cannot be attributed to localized cooling downstream of the heated region. This finding is contrary to current simplified stability theories for flat plate flows which state that surface heating will destabilize the boundary layer.

EFFECTS OF SURFACE HEATING ON THE LAMINAR BOUNDARY LAYER OF A TWO-DIMENSIONAL, MACH THREE WIND TUNNEL NOZZLE

by

Torence P. Brogan

A thesis submitted in partial fulfillment of the
requirements for the degree

of

Master of Science

in

Mechanical Engineering

MONTANA STATE UNIVERSITY
Bozeman, Montana

July 1995

N378
B7865

APPROVAL

of a thesis submitted by

Torence Patrick Brogan

This thesis has been read by each member of the thesis committee and has been found to be satisfactory regarding content, English usage, format, citations, bibliographic style, and consistency, and is ready for submission to the College of Graduate Studies.

Dr. Anthony Demetriades — (Signature) — Date: 7/31/95

Approved for the Department of Mechanical Engineering

Dr. Vic Cundy — (Signature) — Date: 7/28/95

Approved for the College of Graduate Studies

Dr. Robert Brown — (Signature) — Date: 8/22/95

STATEMENT OF PERMISSION TO USE

In presenting this thesis in partial fulfillment of the requirements for a Master's degree at Montana State University, I agree that the Library shall make it available to borrowers under the rules of the Library.

If I have indicated my intention to copyright this thesis by including a copyright notice page, copying is allowable only for scholarly purposes, consistent with "fair use" as prescribed in the U.S. Copyright Law. Requests for permission for extended quotation from or reproduction of this thesis in whole or in parts may be granted only by the copyright holder.

Signature Torence P. Brogan

Date 8/16/95

TABLE OF CONTENTS

LIST OF TABLES vi

LIST OF FIGURES vii

NOMENCLATURE xi

ABSTRACT xiv

1. INTRODUCTION 1

2. BACKGROUND 6

3. OUTLINE OF THE INVESTIGATION 15

- Goals 15
- Facilities 17
- Test Matrix 19
- Experiment Design 22

4. THEORY 24

5. EXPERIMENTAL SETUP 29

- Heated Nozzle Block 29
- Heated Nozzle Instrumentation 32
- Overlay Construction 35
- Data Acquisition System 43
- Profile Measurement Instrumentation 44

6. DATA REDUCTION 49

- Profile Reduction 49
- Recovery Factor 53
- Edge Calculation 57
- Near-wall Points 59
- Offset Correction 59

7. RESULTS 64

Preliminary Investigations 64
Surface Temperature Distribution and Heat Transfer 68
Boundary Layer Profiles 77
Integral Properties 87

8. CONCLUSIONS 95

REFERENCES CITED 99

APPENDIX A: Experimental Data Reduction Program and Flow Charts 103

APPENDIX B: Complete Set of Velocity and Temperature Plots 152

LIST OF TABLES

Table — Page

1. Test matrix for heated and unheated wall comparisons to theoretical predictions — 4

LIST OF FIGURES

Figure — Page

1. Free-stream turbulence modes, boundary layer transition process, and premature transition. Adapted from reference [20]. 3
2. Effect of wall temperature on transition near the throat. Adapted from reference [6]. 11
3. Research plan to understand effect of surface heating on supersonic nozzle boundary layer transition. 13
4. Main features of the MSU-SWT. 18
5. Schematic of MSU-SWT test section. 19
6. Data range of MSU-SWT adiabatic laminar boundary layer studies by this research, Berger [24], and Demetriades [23]. 22
7. Circulating fluid heater assembly and temperature controller. 30
8. Layers of the thermocouple-overlay system 32
9. Spatial distribution of new and existing type-J thermocouples used to monitor surface temperature 34
10. Vacuum test to verify quality of each layer in the instrumented overlay on the nozzle block. 37
11. Photograph of the instrumented nozzle block initial surface quality. 42
12. Photograph of instrumented nozzle block surface quality after sufficient to eliminate waviness. 42
13. Recording circuit for profile measurements. 45
14. Drawing of the dual probe for measuring P_t' and T_t' through the boundary layer. 46
15. Steps to calculate the desired boundary layer quantities from measurements. 50

LIST OF FIGURES (continued)

Figure Page

16. Recovery factor, rf, vs. M with Re_{TTP} as a parameter. $400 < Re_{TTP} < 900$ 55

17. Recovery factor, rf, vs. Re_{TTP} over Mach number range $2.1 < M < 3.0$ 55

18. Treatment of near-wall points for calculating integral properties. 60

19. Technique to calculate y-calibration from analog to physical units. 61

20. Wall-pitot pressure measurement before applying data correction. 62

21. Wall-pitot pressure measurement after applying data correction. 62

22. Free-stream quality with instrumented nozzle block installed (unheated condition). 65

23. Edge Mach number, M_e, variation, case 1a and 1b, (P_o = 400 mmHg, T_o = 32 °C, adiabatic wall); effect of overlay. 65

24. Edge Mach number, M_e, variation, case 2a - 2d, (P_o = 400 mmHg, T_o = 66 °C, adiabatic and heated wall); effect of overlay. 66

25. Edge Mach number, M_e, variation, case 3a and 3b, (P_o = 500 mmHg, T_o = 21 °C, adiabatic and heated wall); effect of overlay. 66

26. Example of transient surface temperature measurements. (case 2c/2d shown: P_o = 400 mmHg, T_o = 66 °C, $(T_w)_{x=0}$ = 97 °C).....69

27. Adiabatic wall temperature, T_{aw}/T_o, case 1a and 1b, (P_o = 400 mmHg, T_o = 32 °C). 71

28. Adiabatic wall temperature, T_{aw}/T_o, case 2a and 2b, (P_o = 400 mmHg, T_o = 66 °C). 71

29. Adiabatic wall temperature, T_{aw}/T_o, case 3a, (P_o = 500 mmHg, T_o = 21 °C). 72

30. Temperate distribution, T_w/T_{aw}, for heated wall condition, case 2c and 2d, (P_o = 400 mmHg, T_o = 66 °C, $(T_w)_{x=0}$ = 97 °C). 74

LIST OF FIGURES (continued)

Figure Page

31. Temperature distribution, T_w/T_{aw} for heated wall condition case 3b, (P_o = 500 mmHg, T_o = 21 °C, $(T_w)_{x=0}$ = 60 °C). 74

32. Theoretical heat flux distribution, q_w'' for cases 2c/2d (P_o = 400 mmHg, T_o = 66 °C, $(T_w)_{x=0}$ = 97 °C) and 3b (P_o = 500 mmHg, T_o = 21 °C, $(T_w)_{x=0}$ = 60 °C).....75

33. Effect of screens on mean-flow: Boundary layer velocity progression for case 1a and 1b. (P_o = 400 mmHg,, T_o = 32 °C, adiabatic wall) 79

34. Effect of heat on mean-flow: Boundary layer velocity progression for case 2a - 2d. (P_o = 400 mmHg, T_o = 66 °C, heated condition is $(T_w)_{x=0}$ = 97 °C) 80

35. Thermal boundary layer development. Case 2a and 2b. (P_o = 400 mmHg, T_o = 66 °C, heated condition is $(T_w)_{x=0}$ = 97 °C) 84

36. Detail view of thermal boundary layer near wall.....86

37. Boundary layer thickness (99% u/u_e), case 1a and 1b, Demetriades [23] and Berger [24] data. (P_o = 400 mmHg, T_o = 32 °C).....88

38. Boundary layer thickness (99% u/u_e), case 2a - 2d. (P_o = 400 mmHg, T_o = 66 °C) 88

39. Boundary layer thickness (99% u/u_e), case 3a and 3b. (P_o = 500 mmHg, T_o = 21 °C).....89

40. Detail view of the velocity boundary layer edge: effect of heating on boundary layer thickness. 90

41. Displacement thickness, case 1a and 1b, Demetriades [23] and Berger [24] data. (P_o = 400 mmHg, T_o = 32 °C, adiabatic wall).....91

42. Displacement thickness, case 2a - 2d. (P_o = 400 mmHg, T_o = 66 °C).....91

43. Displacement thickness, case 3a and 3b. (P_o = 500 mmHg, T_o = 21 °C).....92

LIST OF FIGURES (continued)

Figure Page

44. Momentum thickness, case 1a and 1b, Demetriades [23] and Berger [24] data. (P_o = 400 mmHg, T_o = 32 °C, adiabatic wall) 92

45. Momentum thickness, case 2a - 2d. (P_o = 400 mmHg, T_o = 66 °C) 93

46. Momentum thickness, case 3a and 3b. (P_o = 500 mmHg, T_o = 21 °C) 93

NOMENCLATURE

Symbol	Description
C_p	rms pressure fluctuation coefficient
c_p	constant pressure specific heat
G_θ	Görtler number based on momentum thickness, θ
L	nozzle length (throat to test section entrance) = 38.5 cm.
M	Mach number
$\dot{m}$	mass flow rate
P	pressure
Pr	Prandtl number
q_w''	heat flux at the wall (+ into the boundary layer)
R	gas constant for air
r_c	local radius of curvature
Re	Reynolds number
Re'	Unit Reynolds number
Re_{TTP}	Reynolds number based on total temperature probe height
rf	recovery factor defined as $(T_t' - T_e)/(T_t - T_e)$
T	total temperature measured by a probe in the flow stream
u	velocity in the streamwise direction

NOMENCLATURE (continued)

x	streamwise coordinate (x=0.0 @ nozzle throat)
y	vertical distance from the wall
δ	boundary layer thickness
δ^*	displacement thickness
θ	momentum thickness
γ	ratio of specific heats
ρ	density
μ	dynamic viscosity
ξ	streamwise coordinate in the computational plane
η	normal coordinate in the computational plane

Subscripts

Symbol	Description
aw	local adiabatic conditions at the nozzle block surface
e	conditions at the edge of the boundary layer; represents a local free-stream quantity
o	stagnation conditions
t	local total condition

NOMENCLATURE (continued)

w	local heated conditions at the nozzle block surface
∞	conditions based on tunnel core flow Mach number

Superscripts

$( )'$	Denotes a quantity measured by a probe in the flow stream with the exception of unit Reynolds Number, Re'

ABSTRACT

The effects of surface heating on the mean laminar boundary layer on a supersonic wind-tunnel nozzle have been investigated. Experimental measurements and theoretical predictions were compared with an emphasis on the comparison with and without surface heating. Fifty-one profiles of pitot pressure and total temperature were measured in various combinations of tunnel stagnation conditions, surface temperature, and a turbulence screen. A total of 24 profile pairs were acquired at several streamwise locations with wall-to-total temperature ratios of about 1.10 at the throat. Overall, the theory predicted the boundary layer development well both with and without surface heating. All trends predicted by the theory were present in the experimental results. Velocity profiles obtained by experiment and by theory showed good agreement when the normal coordinate was plotted as y/δ where y is the distance from the wall, and δ is the boundary layer thickness. In both experiment and theory, the surface heating produced a very small effect on the boundary layer. In view of the observed stabilizing effect that surface heating has on supersonic nozzles, it has been concluded that the boundary layer stability is extremely sensitive to the surface temperature. Although heat was concentrated at the throat, numerical computations showed that the surface heat flux was directed into the boundary layer for all streamwise locations along the nozzle, therefore, the observed stabilizing effects of surface heating on the wind-tunnel nozzle cannot be attributed to localized cooling downstream of the heated region. This finding is contrary to current simplified stability theories for flat plate flows which state that surface heating will destabilize the boundary layer.

CHAPTER 1

INTRODUCTION

The design of a supersonic vehicle often requires an accurate knowledge of the boundary layer transition over the vehicle's surface. The range and location of transition, and the factors which affect transition dramatically influence the vehicle flight dynamics, aerodynamic heating and thermal protection systems. Wind-tunnels serve as a valuable tool to help engineers understand the boundary layer condition over the vehicle before and during the design process. However, in many wind-tunnel experiments, free-stream turbulence or "noise" causes boundary layer transition on a model to occur at Reynolds numbers lower than those found in a free-flight environment. The increasing need for accurate supersonic transition studies has prompted the recent development of wind-tunnels which incorporate features to reduce the levels of free-stream noise associated with conventional wind-tunnels. These "quiet" wind-tunnels are able to more closely simulate the speed and altitude environment experienced by actual supersonic flight vehicles.

However, in many instances, researchers have not been able to suppress free-stream noise to acceptable levels. The primary result today is that the quiet test core in

most quiet wind-tunnels is limited in size and/or limited in the range of operating conditions. In a general review of free-stream disturbances, Morkovin [1] reports three distinct types of unsteady free-stream disturbances, namely, entropy (temperature spottiness), vorticity (velocity), and acoustic (pressure) fluctuations. The first two types, entropy and vorticity fluctuations, are convected along streamlines from upstream disturbances. The convected disturbances, traceable to sources upstream of the sonic throat, can be practically eliminated in a supersonic or hypersonic wind-tunnel with proper settling chamber design. Of critical importance is the proper use of screens, and a large velocity ratio between the settling chamber and sonic throat [1]. The third type of unsteady disturbance (acoustic fluctuations) is radiated across streamlines. For a smooth, continuously contoured nozzle, acoustic disturbances are generated primarily from transitional and turbulent boundary layers on the tunnel walls. At Mach numbers greater than about 2.5, pressure fluctuations arising from transitional and turbulent boundary layers are large, and for Mach numbers greater than 3.0, they completely dominate the free-stream noise [2]. The transitional boundary layer can create noise levels five times greater than can the turbulent boundary layer [3].

The acoustic noise radiated from a transitional or turbulent boundary layer on the tunnel walls creates premature transition on a wind-tunnel model causing the transition data obtained in a supersonic wind tunnel to be invalid. Figure 1 illustrates this process of premature transition. The transitional or turbulent boundary layer

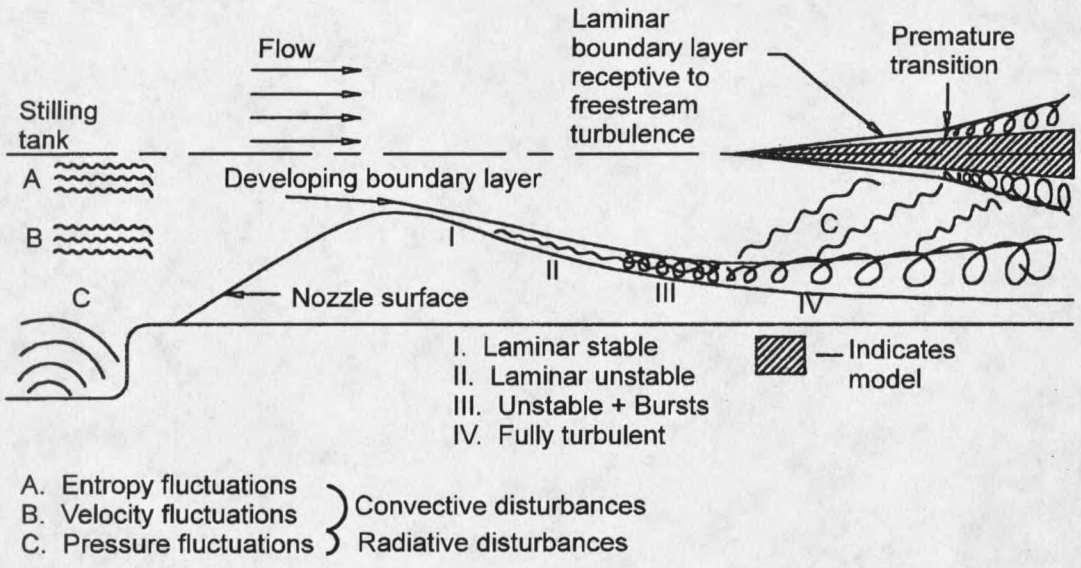


Figure 1. Free-stream turbulence modes, boundary layer transition process, and premature transition. Adapted from reference [20].

occurs on the tunnel walls, and the acoustic fluctuations generated in this transitional or turbulent layer propagate along Mach lines [1, 3] across the flow and impinge on the model boundary layer. That noise originating from the nozzle boundary layer interacts with the model boundary layer to produce premature transition on the model.

As the unit Reynolds number, Re', is increased, the location of transition on the tunnel walls advances upstream. In turn, acoustic disturbances from the transitional and turbulent nozzle boundary layer contaminate a larger streamwise portion of the free-stream. Consequently, the size of the quiet test core (that portion of the test section which has acceptably low levels of free-stream noise) becomes a function of the tunnel's operating conditions. For example, if the tunnel stagnation pressure is increased, Re' increases, the location of the nozzle boundary layer transition moves

upstream, acoustic fluctuations from this transitional nozzle boundary layer contaminate the rear portion of the quiet test core, and the size of the quiet test core is reduced.

Although there has been a great deal of research in understanding the sources of free-stream noise, the actual mechanism by which the turbulent boundary layer on the nozzle surface produces pressure fluctuations is not well understood. Beyond stilling tank design, quiet wind-tunnel technology for supersonic flows presently appears to be reduced to the problem of maintaining a laminar boundary layer on the nozzle surface.

To "laminarize" the boundary layer, one must take advantage of the factors which delay transition. It is well known that boundary-layer transition is intimately related to surface roughness [4]. The smoother the surface, the larger the transition Reynolds number that can be supported. The profile shape of the mean velocity in the boundary layer has also been inseparably associated with boundary layer stability, and correspondingly boundary layer transition. An additional phenomenon noted in several supersonic wind-tunnel experiments [5 - 7] has been the effect of wall temperature on transition.

Contrary to simplified stability theories for flat plate boundary layers, wall heating has been noted to delay transition on curved supersonic nozzles, according to the results of Harvey et. al. [5], and Demetriades [7]. Demetriades has also observed the opposite effect with nozzle wall cooling in a Mach 3.0 wind-tunnel; transition is promoted with wall cooling [6]. These findings suggest wall heating as a means of delaying transition in a supersonic wind-tunnel nozzle. Consequently, a thorough

investigation of the nozzle boundary layer stability with both experiment and theory has been implemented at Montana State University (MSU). The first phase of this investigation is the study of the mean boundary layer in the presence of surface heating. The results of a comparison between experiment and theory regarding the effect of surface heating on the nozzle laminar boundary layer are the subject of this thesis.

CHAPTER 2

BACKGROUND

The subject of boundary layer stability is complex and is a function of many parameters such as free-stream turbulence, surface roughness, heat transfer, Reynolds number, Prandtl number, Mach number, etc.. Stability theory is simply the study of how small disturbances are amplified in a boundary layer. If a disturbance can be amplified in the boundary layer at a particular location, that point in the boundary layer is called unstable. Conversely, if a disturbance within the layer is damped, the boundary layer is stable. The field of stability computations (linear or non-linear) is immense and no attempt is made here to present a mathematical formulation of stability theory except as it relates to the problem of flow over a surface with heat transfer. The interested reader is referred to references [8 - 11] for further details on linear stability theory.

Boundary layer transition is an important factor in aerodynamic design; and it is only connected to stability theories by hypothesis. This hypothesis, first developed by Reynolds [8], is that transition is a consequence of instability of the laminar boundary layer. There have been numerous experiments that agree with theoretical stability

predictions to support the Reynolds hypothesis. A link between boundary layer transition and stability has enabled researchers to partially understand mechanisms controlling transition [12, 13].

The hydrodynamic stability of a boundary layer is strongly controlled by the shape of the mean velocity profile in the boundary layer. It has been shown [12] that the more negative the velocity second derivative is at the wall, $(\partial^2 u/\partial y^2)|_{y=0}$, the "fuller" and more stable the velocity profile becomes. This result applies to both inviscid and viscous instabilities. With an understanding of how $(\partial^2 u/\partial y^2)|_{y=0}$ affects hydrodynamic stability, a class of stability modifiers can be developed from the stream-wise momentum equation,

$$\left[\rho\frac{\partial u}{\partial t}\right]_{y=0}+\left[\rho u\frac{\partial u}{\partial x}\right]_{y=0}+\left[\rho v\frac{\partial u}{\partial y}\right]_{y=0}+\frac{dP}{dx}-\frac{d\mu}{dT}\left[\frac{\partial T}{\partial y}\frac{\partial u}{\partial y}\right]_{y=0}=\left[\mu\frac{\partial^2 u}{\partial y^2}\right]_{y=0}. \quad (1)$$

Note that the product rule and chain rule have been applied to the viscous dissipation term to obtain the last term on the left hand side (LHS) and the term on the right in equation (1).

Among other modifiers (suction, controlled pressure gradient, etc.) the last term on the LHS is positive or negative depending on the direction of heat transfer. In air, the viscosity, μ, always increases with temperature, therefore, $d\mu/dT > 0$ always, and $\partial u/\partial y|_{y=0} > 0$ for non-separating flows. If heat is removed from the fluid (wall cooling), $\partial T/\partial y > 0$, and the last term of the LHS is negative. The second derivative $(\partial^2 u/\partial y^2)|_{y=0}$ becomes more negative showing a fuller and thus more stable velocity

profile. Similarly, wall heating produces $\partial T/\partial y < 0$, bringing the second derivative closer to zero, and destabilizing the boundary layer. This heating effect has been verified in a number of experiments on cones and flat plates [12]. In fact, Lees [11], and later Reshotko [14] showed the possibility that at low supersonic Mach numbers, a boundary layer can be completely stabilized with cooling below some threshold ratio of cooled wall to adiabatic wall temperature.

It must be noted that the classical heat transfer effect applies to simple flow geometries such as a cone or flat plate (which constitute the majority of subsonic and supersonic flows). In these geometries the instability mechanism is primarily Tollmien-Schlichting (TS) "waves" and more specifically, the first TS mode of instability [13]. Other modes of instability, such as higher TS modes [13], Görtler vortices [15], and Cross-Flow instabilities [16] do not respond to the heat transfer effect in the same way as discussed above, and may show an opposite heat transfer effect.

Surface roughness has also been closely associated with boundary layer stability and transition. For Reynolds numbers based on roughness height greater than about 150, vortices emanating from discrete roughness elements can interact with unsteady instability mechanisms to induce transition [4]. It is conjectured from experimental observation that roughness can still influence transition by enhancing free-stream receptivity [4], where receptivity is defined as the process by which free-stream disturbances amplify and/or create unstable disturbances. In blunt body transition experiments, the size and type of distributed roughness has been named as the

dominating factor in boundary layer transition [17]. Although it is widely accepted that roughness does influence stability and transition to some extent, there is still little known about the underlying mechanisms which promote roughness-induced transition over distributed roughness, such as that which may be found on a wind-tunnel nozzle surface.

Recently, researchers have been investigating the capabilities of stabilizing air flow by local heating. Most of the theoretical and experimental studies are for subsonic flows. Masad and Nayfeh [18] have shown theoretically that the boundary layer over a flat plate can be notably stabilized when the heating strips are properly placed ahead of unstable regions. The result is that the boundary layer is being "cooled" upon entering the region of instability. Therefore, by the classical cooling effect discussed above, the velocity profile instead becomes more stable, thus moving transition onset downstream. Masad and Nayfeh [18] studied both incompressible and compressible flows (up to $M = 0.8$). Dovgal et. al. [16] have performed experiments on a flat plate with local heating at and near the leading edge. The experimental results showed transition Reynolds numbers increased by as much as 70%.

Other low-speed experiments have been conducted on convex walls showing similar effects on the Görtler instability [15]. The results are again reconciled to classical theory by noting there is a natural cooling over the convex surface which stabilizes the Görtler instability. In 1975, Harvey et. al. [5] investigated the effect of heating the surface of a supersonic nozzle as part of a quiet wind-tunnel design. Strip

heaters were placed around the outside of the nozzle surface and subsonic approach. The results showed the transition Reynolds number increased by approximately 20%. The heating of these experiments, however, was not applied locally, but uniformly over the nozzle surface. The probable mechanism of transition delay was explained as a result of an increased boundary layer thickness with heating. The increase in boundary layer thickness in turn reduced the height of surface roughness relative to the boundary layer thickness. Roughness due to ice formation has also been held responsible for the phenomena of transition "reversal" (i.e. a change in the Reynolds number trend with continued cooling or heating) [19].

Demetriades [6] studied the effect of surface roughness and wall temperature on the boundary layer in the throat of a supersonic wind-tunnel. Liquid nitrogen was circulated through ducts located approximately 1 cm below the aluminum nozzle block surface. The ducting spanned the width of the nozzle and a covered an 18 cm streamwise region centered over the nozzle throat. The transition Reynolds number was seen to decrease with decreasing wall temperature (Figure 2). This result apparently again contradicts the simpler theory for flat plate flows. Furthermore, the Reynolds number based on roughness height was estimated at less than unity over the throat [7].

During a 2 year study, Mueller [20] mapped the progression of transition on the supersonic adiabatic nozzle (sidewalls and floor) of references [6, 21, 7] as a function of stagnation conditions and turbulence screens. The results did not reveal any

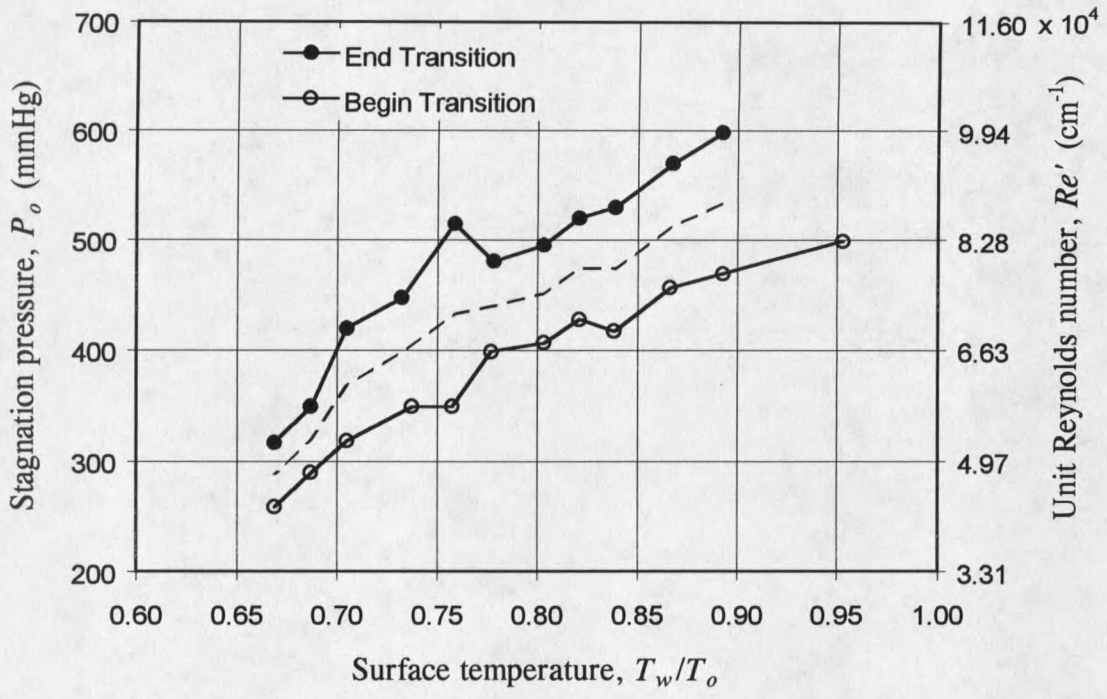


Figure 2. Effect of wall temperature on transition near the throat. Adapted from reference [6].

turbulent wakes or vortices of isolated roughness elements which may induce transition [20]. Mueller's findings and the small roughness Reynolds number of reference [6] leave the possibility of another mechanism producing the heating effect observed by Demetriades [6] and Harvey et. al. [5].

In 1993, Cornelius and Leonardson [21], and Demetriades [6] briefly investigated the effect of heating the nozzle to see if the heating effect of references [5] and [6] still prevailed, and to what extent. A uniform heat flux was applied at x/L = 0.28 (11 cm) upstream of the throat to (26 cm) $x/L = 0.7$, where x is the distance from the nozzle throat, and L is the nozzle length. Demetriades and Cornelius were

successful in reducing a turbulent boundary layer to a laminar boundary layer with heat addition [7].

The effect of wall heating is not clearly understood and there are very few studies -- experimental or theoretical -- which specifically investigate wall heating or cooling at the conditions tested by this author. Furthermore, the transition process on a two-dimensional supersonic nozzle surface is not well understood. There are significant differences between a nozzle floor boundary layer and that of a cone or flat plate, such as curvature, a strong pressure gradient, and sidewalls. If the heating effect of Demetriades and Harvey et. al. can be optimized (e.g. heating level, location) and transferred to other wind-tunnel facilities, implications on quiet wind-tunnel technology can be far reaching. Moreover, if transition can be substantially delayed by local heating in a wind-tunnel, the noise produced by the transitional and turbulent boundary layer will be moved down-stream, and the quiet test core will be enlarged. Moving the location of transition onset down-stream will also enable current quiet wind-tunnel facilities to operate at a wider range of stagnation conditions.

In order to optimize the Demetriades-Harvey heating effect, the mechanism(s) must first be understood. Identifying the mechanism(s) will also assist in determining to what extent the Demetriades-Harvey heating effect can be utilized to advantage in other facilities. Figure 3 shows the present research plan to understand the effect of local heating on a supersonic wind-tunnel nozzle. The mean boundary layer will first be investigated for two reasons. First, the boundary layer stability is intimately related

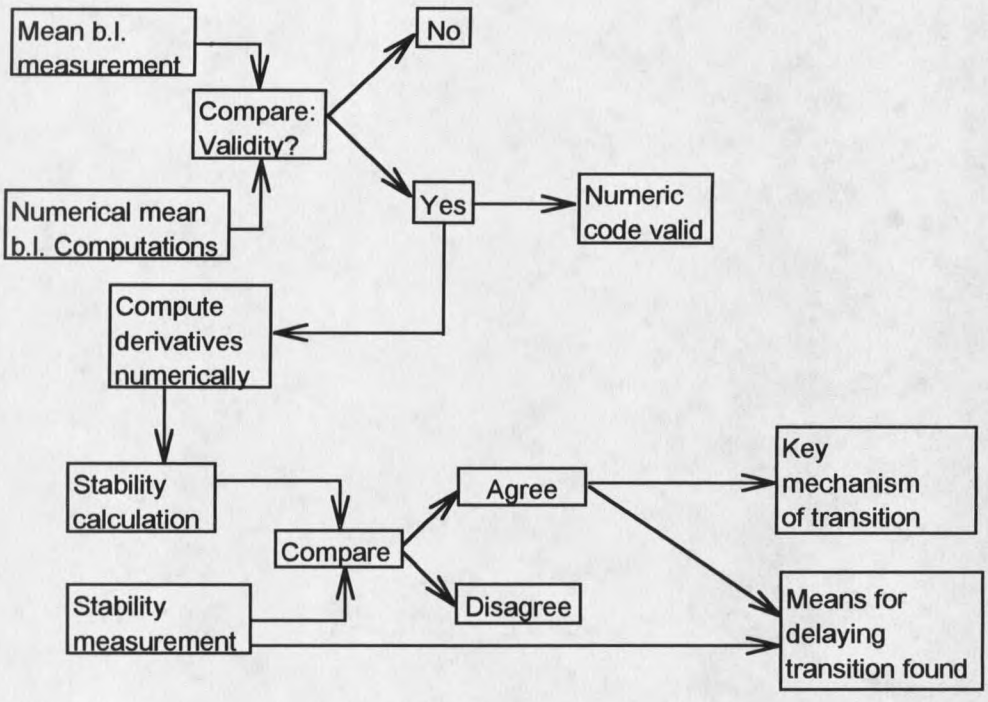


Figure 3. Research plan to understand effect of surface heating on supersonic nozzle boundary layer transition.

to the shape of the mean boundary layer as discussed above. Secondly, the derivatives necessary for stability computations cannot be obtained from experiment. Therefore, if a numerical model is confirmed, the necessary derivatives can be computed from theory and stability calculations can be performed. These stability calculations can be compared to stability measurements. If the stability measurements and calculations agree, the stability code is validated and can be used to identify the unstable mechanism(s) of transition. If stability measurements and calculations do not agree, the experimental measurements alone may be sufficient to determine a means for delaying transition with surface heating. The work presented in this thesis constitutes

the first phase of the research plan by which experimentally measured mean profiles are compared to theoretical predictions.

CHAPTER 3

OUTLINE OF THE INVESTIGATION

Goals

A favorable comparison between the measured and theoretical mean flow opens the way for stability calculations. Experimental measurements of the mean flow cannot be used for stability calculations for several reasons. First, numerical stability calculations for viscous instabilities, which involve numerical integration of the Orr-Sommerfeld equation, depend strongly on continuous and "smooth" profiles from which first and second velocity derivatives can be calculated. Even a profile numerically computed with single precision accuracy (approximately 7 significant digits) yields unusable results in the profiles of velocity second derivatives ($\partial^2 u/\partial y^2$) for linear stability calculations. The second reason is that boundary layer quantities (e.g. pressure and temperature) cannot be accurately measured near the wall due to wall-probe interference and low-Reynolds number effects.

If the stability codes are found to agree with measured stability phenomena, computer simulations can be performed to investigate scenarios which will maximize

the observed stabilizing effect of surface heating. Theoretical stability calculations can also be instrumental in determining the mechanism(s) of stability. The loop can be closed by comparing stability predictions to stability measurements in the boundary layer.

The objectives of the research presented here are to:

(a) Check, by measurement, the capability of the numeric boundary layer codes at NASA Ames to predict the boundary layer mean flow with either a cold or a hot wall.

(b) Investigate the extent to which the heating levels found to affect transition [7, 21] also change the mean profiles.

The first of these two objectives has been achieved by comparing theoretical and experimental results of the boundary layer profiles with and without surface heating. The second objective has been accomplished by a detailed survey of the boundary layer profiles with and without surface heating for both experiment and theory. Trends were extracted from the profile results by investigating the streamwise progression of the boundary layer along the nozzle surface. Emphasis was placed on differences with and without surface heating.

Facilities

All experimental data was gathered in the Montana State University Supersonic Wind-tunnel (MSU-SWT). The SWT is a two dimensional, open circuit, continuous flow facility using air as the working fluid. The nominal test section Mach number is 3.0 with a Re' range of 21,000 to 67,000 cm^{-1}. The stagnation pressure, P_o, and stagnation temperature, T_o, can be varied over a range of 200 mmHg to 620 mmHg and 16 °C to 71 °C, respectively. The stagnation pressure and temperature can be maintained at an accuracy of ±2 mmHg and ±1 °C, respectively, using continuous feedback sensors. By varying P_o and T_o, either a fully laminar or a fully turbulent boundary layer will occur naturally on the nozzle. The tunnel also has a slot upstream of the nozzle whereby the operator can insert or remove turbulence screens during tunnel operations. Figure 4 illustrates the main features of the SWT, and Figure 5 shows the tunnel nozzle and test section.

The MSU-SWT is a documented "quiet" wind tunnel [22] with rms fluctuation pressure coefficients (C_p) less than 10^{-2} when the boundary layer is laminar on the wetted perimeter. Quiet operation is obtained by a settling chamber upstream of the approach and by the absence of any moving parts upstream of the test section. The settling chamber provides a contraction ratio greater than 800:1. The DeLeval nozzle has a throat height of 1.8 cm, a length from throat to test section entrance of 38.5 cm, and test section dimensions of 8.1 cm high by 7.9 cm wide. The radius of curvature on

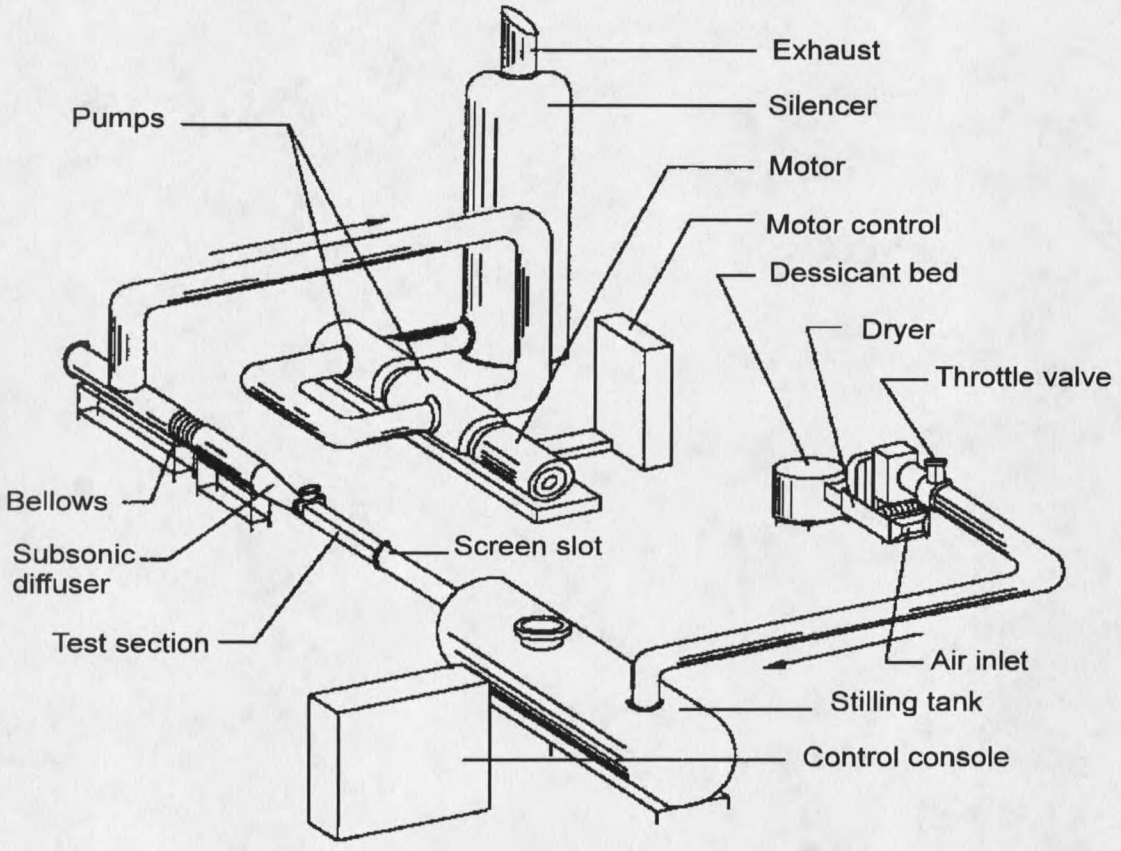


Figure 4. Main features of the MSU-SWT.

each nozzle block at the throat is 30 cm and the inflection point is located 9.78 cm downstream of the throat. Visual access from upstream of the throat to the test section exit allows the user to view the model and nozzle surfaces at all times during a particular experiment. A probe traversing mechanism also allows the user to "pilot" a probe anywhere in the nozzle or tunnel test section. This traversing mechanism can be interfaced with the MSU-SWT Automated Data Acquisition System (ADAS) to record data and control the probe motion simultaneously.

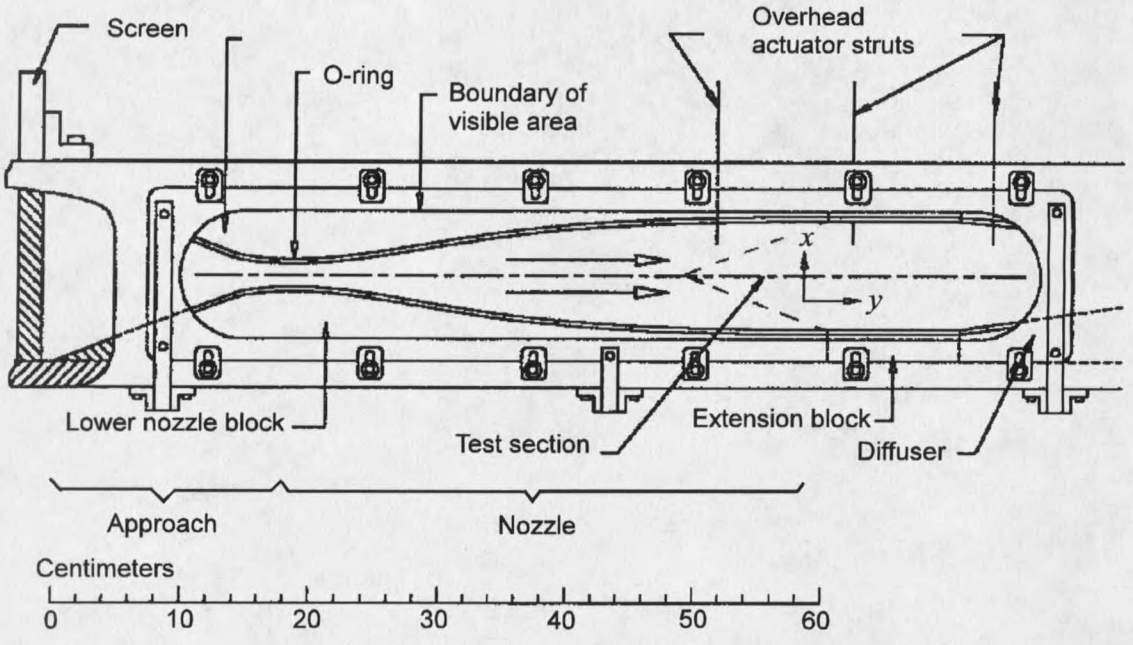


Figure 5. Schematic of MSU-SWT test section.

Test Matrix

The investigation of the laminar boundary layer was conducted as a function of P_o, T_o, T_w, x-position, and the presence of a 320 mesh screen upstream of the two dimensional contraction. Table 1 lists the combinations of P_o, T_o, T_w, and screen configuration tested. Both experimental measurements and theoretical calculations were conducted for each case listed in Table 1.

The P_o range was limited on the upper end by the occurrence of a transitional boundary layer on the tunnel walls. The lower limit was dictated by the capabilities of the tunnel. Although the MSU-SWT is capable of stagnation pressures as low as 200 mmHg, when empty, it was found that with the instrumentation necessary for these

Case	P_o (mmHg)	T_o (°C)	Screen	Steady State Floor Condition	x-range (cm) : 2.5 cm spacing
1a	400	32	No	Unheated	7.6 - 17.8
1b	400	32	Yes	Unheated	7.6 - 17.8
2a	400	66	No	Unheated	7.6 -30.5
2b	400	66	Yes	Unheated	7.6 -30.5
2c	400	66	No	Heated $(T_w)_{x=0}$= 97 °C)	7.6 - 30.5
2d	400	66	Yes	Heated $(T_w)_{x=0}$= 97 °C)	7.6 - 30.5
3a	500	21	Yes	Unheated	7.6 - 22.9
3b	500	21	Yes	Heated $(T_w)_{x=0}$= 60 °C)	7.6 - 22.9

Table 1. Test matrix for heated and unheated wall comparisons to theoretical predictions.

experiments (≈ 11% blockage with respect to the test section area), the tunnel pumps could only draw enough vacuum to consistently support a P_o of 400 mmHg. For stagnation pressures less than this the flow would often "break down." That is, the shock structure usually located in the diffuser would move upstream of the test section into the nozzle in order to satisfy equilibrium conditions. Behind the shock wave, obviously, the flow would become subsonic, and highly chaotic. Measurements upstream of this shock wave were avoided due to the induced vibrations of the probe instrumentation.

The values of T_o were chosen to generate a wider range of Re' conditions. For each adiabatic wall (unheated) case, a corresponding heated wall case with the same stagnation conditions and screen configuration was also investigated. Each heated wall case has been characterized by the surface temperature at the throat, $(T_w)_{x=0}$, since T_w

varies with streamwise position, x. The nozzle block temperature was limited to less than 100 °C due to thermal stresses in the Schlieren glass sidewalls, and other tunnel components.

The streamwise position, x, was varied from 7.6 to 30.5 cm to investigate the capabilities of the numerical boundary layer code at both near and far x locations. Furthermore, Demetriades [23] has taken boundary layer profiles of the adiabatic wall from the throat to 5.1 cm downstream of the throat, while Berger [24] has measured the adiabatic laminar boundary layer on the nozzle and test section floor at locations from 22.0 to 47.4 cm. Thus, the measurements contained in this present investigation were designed to complete the picture of the adiabatic laminar boundary layer growth along the centerline of the MSU-SWT nozzle floor.

Theoretical predictions have been previously compared [22] to the aforementioned experiments of Demetriades and Berger. All measurements were taken along the nozzle centerline. Figure 6 shows the Re_x range covered by the adiabatic data taken by Brogan, Demetriades, and Berger.

Lastly, the effect of the screen upstream of the nozzle was investigated because Mueller had shown that the screen significantly affected the locations at which transition began and ended [20]. In accordance with the goal to use the results of this present study as a basis for transition research, the effect of screens on the mean boundary layer was important.

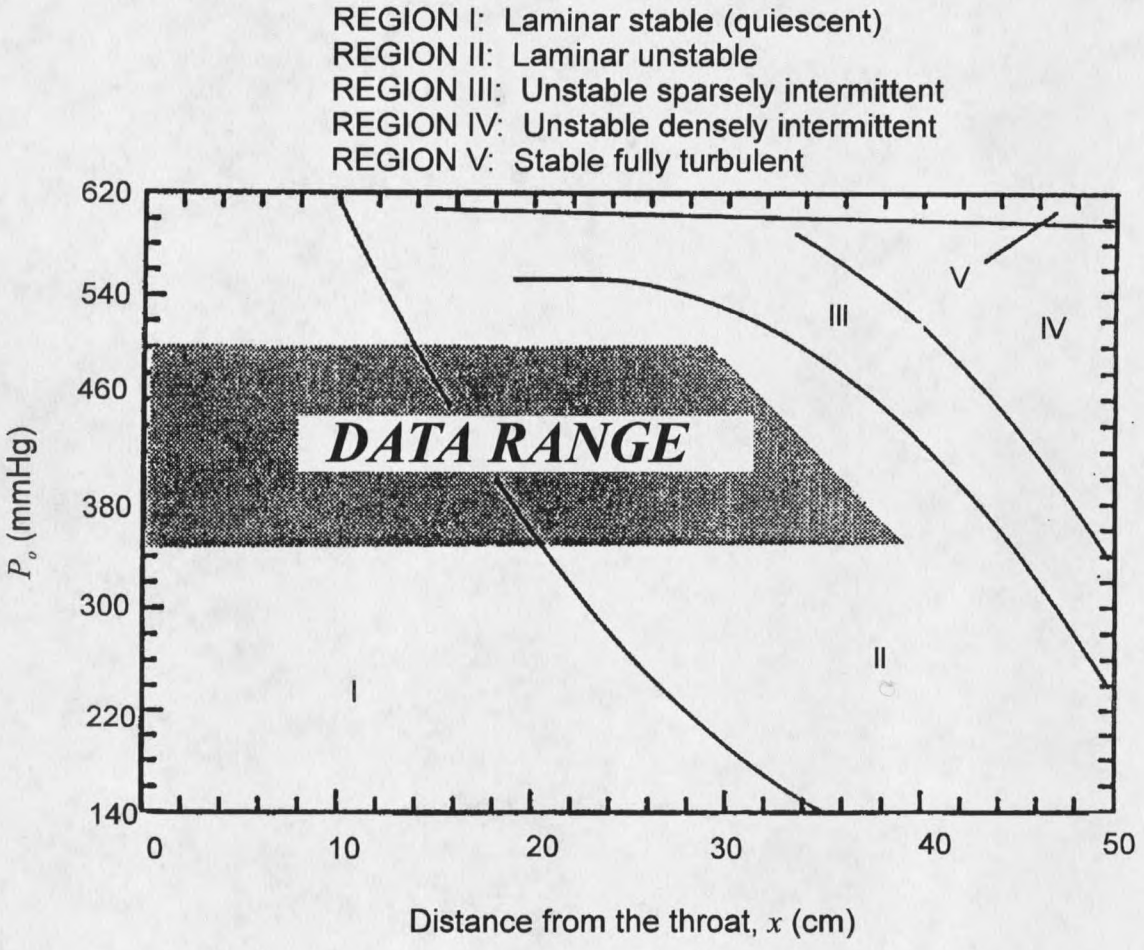


Figure 6. Data range of MSU-SWT adiabatic laminar boundary layer studies by this research, Berger [24], and Demetriades [23].

Experiment Design

For each profile, the pitot pressure, P_t', and probe total temperature, T_t', were measured at discrete points through the boundary layer. These boundary layer sweeps

started at the wall and continued for at least one boundary layer thickness into the free-stream. From profile measurements of P_t' and T_t', along with the stagnation conditions, P_o and T_o, all other profile quantities such as velocity, static temperature, density, etc., were calculated. Integral properties were then calculated from the velocity and density profiles. These calculations will be further detailed in the Data Reduction chapter.

Surface heating was applied by circulating a hot fluid beneath the surface of the nozzle throat region. Surface temperatures at 21 locations along the nozzle were continuously monitored and recorded to ensure that the profile quantities were measured at steady-state conditions. These surface temperatures were also used as input to the numerical code for the heated wall cases.

CHAPTER 4

THEORY

For each experimental case 1a through 3b theoretical calculations were performed using the boundary layer code of Harris and Blanchard [25]. King has previously used this program to investigate the mean properties and stability of the laminar boundary layer in the MSU-SWT [22]. Mean boundary layer predictions showed excellent comparison to experiments conducted by Demetriades. In light of this previous successful work, the same boundary layer code was used to further investigate the laminar boundary layer on the centerline of the nozzle floor with and without heating.

The boundary layer code of Harris and Blanchard is applicable to axi-symmetric or two-dimensional laminar, transitional, or turbulent compressible boundary layers for perfect gas flows. However, the present application is limited to calculations of the two-dimensional laminar boundary layer on the nozzle floor assuming variable properties. Governing equations of the boundary layer are solved using a coupled, iterative implicit finite-difference procedure. An efficient solution is obtained through the entire boundary layer thickness with a variable-grid scheme in the direction normal

to the surface. Therefore, grid points near the wall are closely spaced to accommodate large changes through the boundary layer, while a coarse spacing is generated near the boundary layer edge where changes through the boundary layer occur much slower. The marching solution progresses in the streamwise direction at user input arc length intervals.

Probstein-Elliot and Levy-Lees transformations [25] are used to map the system of governing equations onto the ξ-η computational plane. These transformations remove the classic boundary layer singularity at $\xi = 0$, and reduce the growth of the boundary layer as the solution proceeds downstream.

The initial profile required to start the three-point implicit marching procedure is generated with the locally similar form of the momentum and energy equations. This similar solution is calculated with a finite difference method using the variable grid for the marching procedure, thereby directly providing a realistic initial value at each grid point, η_i. Profiles downstream of $\xi = 0$ converge smoothly to the solution of the marching procedure. The solution in the neighborhood of the starting point should be viewed with skepticism since the solution may still be converging to the theoretically exact boundary layer. All solutions for the present investigations began at x = -5.5 cm. Calculations by King have shown this starting point yields identical solutions to those obtained with initial stations at x < -5.5 cm [26].

The boundary layer code requires the edge pressure distribution, P_e/P_o, as input, along with the object geometry, boundary conditions at the wall, and other pertinent

parameters. Output is in the form of station summaries and boundary layer profiles at user specified locations. In addition, the program has been modified to provide first and second derivatives of boundary layer quantities used as input to the stability code of Malik [27]. The boundary layer code lends itself well to computing the basic state (mean flow) required by the stability calculations because a dense spacing of grid points can be used with minimal computer resources. All boundary layer computations presented here used 101 points in the normal direction and a solution was obtained every 0.15 cm in the streamwise direction.

The edge pressure ratio, P_e/P_o, used as input to the boundary layer code, was theoretically calculated by King [22] with a viscous solution from a two dimensional Navier Stokes (NS) solver. The two dimensional NS code, which is a modified version of ARC2D [28], uses a generalized body-fitted coordinate system. The nozzle surface was modeled using the original design coordinates of the MSU-SWT [29], and a 5th order polynomial was used to model the contraction from the stagnation chamber to the nozzle approach section. The two dimensional NS computations were performed on a mesh with 166 points in the streamwise direction and 73 points from the nozzle wall to the centerline. The symmetry of the tunnel was utilized so that only half of the nozzle flow field needed to be computed. Approximately half of all grid points were in the nozzle wall boundary layer, and the throat was located at the 55th streamwise point.

The edge pressure ratio calculated by King [22] with stagnation conditions of P_o = 350 mmHg, and T_o = 38 °C has been used for all numerical calculations presented

herein. These stagnation conditions are different than those of the experiment, however, deviations produced in the pressure gradient by different stagnation conditions are small. The errors associated with different stagnation conditions of the NS computation and the experiment will be discussed next.

In more common terms, a fixed pressure ratio distribution, P_e/P_o, defines a fixed edge Mach number, M_e, distribution. In reality, the boundary layer displacement thickness, δ^*, produces small changes in the M_e distribution as the tunnel stagnation conditions are varied. Between the conditions of all experimental test cases (1a - 3b), δ^* at the floor of the nozzle exit changed by a maximum of 0.035 cm over the δ^* calculated at the conditions of the NS computation, P_o = 350 mmHg and T_o = 38 °C. The estimated change in the area ratio (including sidewall boundary layer) is 5%, yielding $\Delta M_\infty = 0.06$ for $M_\infty = 3.00$. A graphical comparison of M_e determined by experiment and M_e determined by theory is presented in Chapter 7 (Results). In lieu of the intensive computation requirements of the ARC2D solution (10+ hours of Cray time), these small errors were acceptable.

All unheated cases (1a, 1b, 2a, 2b, 3a) were approximated with an adiabatic wall condition. Throughout the text, the experimental investigations with no external, forced heat transfer applied to the nozzle block condition are referred to as "adiabatic." As always, this was not a truly adiabatic case. Natural convection was occurring on the outside of the tunnel's test section to or from the ambient air. Heat may also have been conducted downstream through the aluminum nozzle blocks due to streamwise

surface temperature gradients. (Gradients in the adiabatic wall temperature must be present due to a Mach number dependence in the free-stream static temperature, T_e.)

Computations for each of the heated cases (2c, 2d, 3b) used the experimentally measured steady-state temperature distribution as input to the boundary layer code. A continuous surface temperature distribution was obtained by linear interpolation between the discrete points of measurement.

CHAPTER 5

EXPERIMENTAL SETUP

Heated Nozzle Block

The solid lower nozzle block of the SWT was replaced with the internally ducted nozzle block of references [6,7,21] to facilitate the surface heating. Hot ethylene glycol was circulated through the internal ducts located approximately 1.0 cm beneath the nozzle surface. The ducting spanned the width of the nozzle and covered a streamwise range of 17.8 cm length centered over the throat. Figure 7 shows the circulating fluid heater assembly. A similar circulating fluid configuration was used in the Cornelius experiments [21], however, the circulating fluid heater assembly of Figure 7 was able to achieve and maintain much higher nozzle surface temperatures.

All heater components were selected to withstand temperatures up to 150 °C. Chemically pure (99.5%) ethylene glycol ($C_2H_4(OH)_2$) was chosen as the circulating fluid since it has a boiling point of 197 °C at atmospheric pressure. The chemical purity also allowed the use of accurate tabulated values for all fluid properties. An explicit expression for the mass flow rate, $\dot{m}$, as a function of fluid temperature was

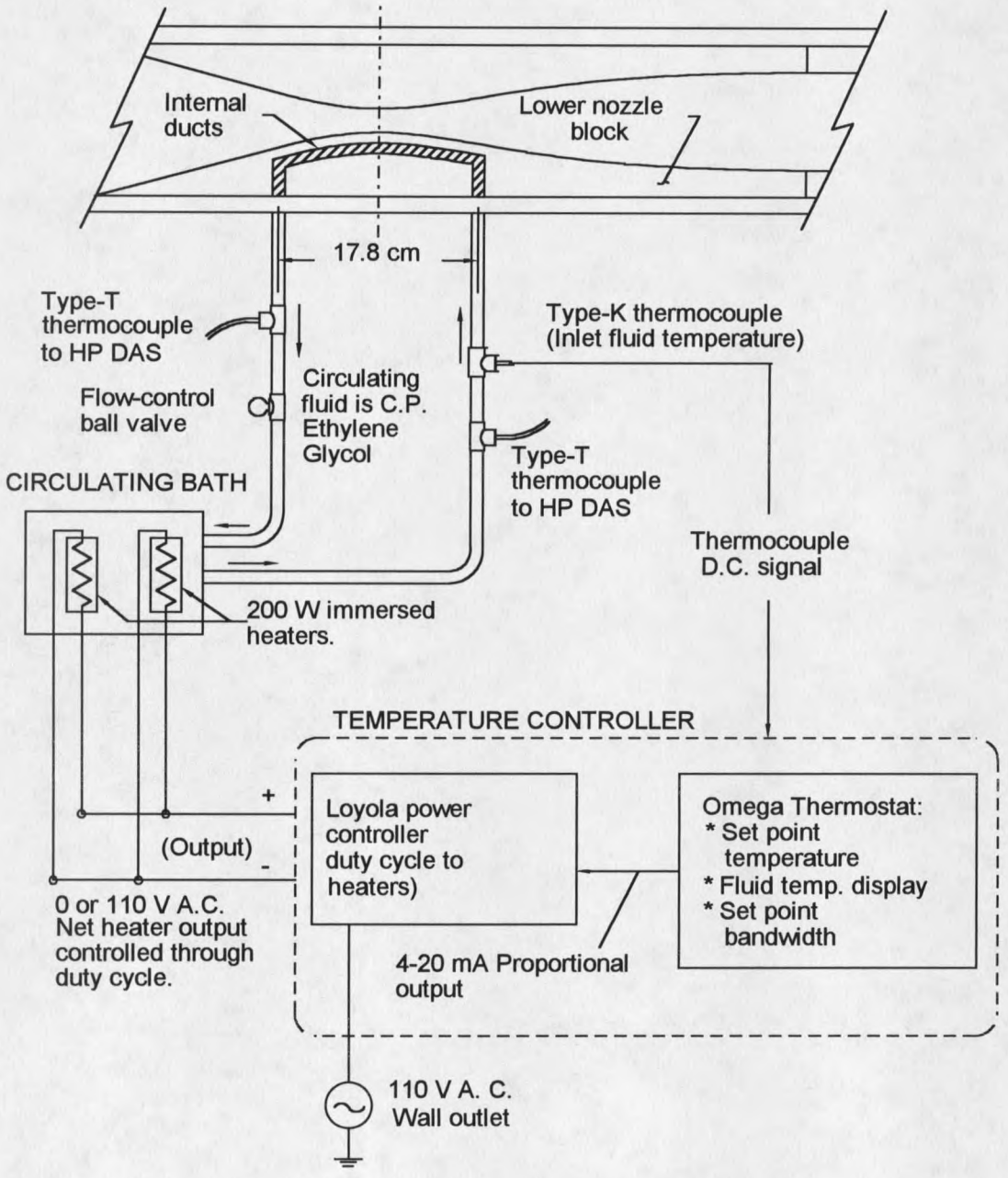


Figure 7. Circulating fluid heater assembly and temperature controller.

obtained by a polynomial curvefit of experimental data. Type-T thermocouples mounted near the nozzle's fluid ports provided a means to calculate the power dissipation into the nozzle using the product of the mass flow rate and enthalpy change($\dot{m} c_p T$) through the ducted nozzle block. Although a ball valve was installed to allow the experimenter to control the mass flow rate, $\dot{m}$, the flow rate was kept at 100% for all experiments.

A Haake FK circulating bath with two 200 W resistance heaters was used to circulate and heat the ethylene glycol. The Haake circulating bath was originally supplied with a set point temperature controller which allowed a maximum fluid temperature of 100 °C. The temperature control circuitry was completely redesigned to allow a much higher set point temperature. The second 200 W heater was also added to the bath to improve the system's ability to quickly attain and regulate the set point temperature.

Figure 7 also shows the block diagram for the temperature controller and circulating bath. In the new temperature controller, an Omega model 4001 thermostat was used to control the set point temperature. The thermostat generates a variable DC current (4 - 20 mA) proportional to the difference between the actual and set-point fluid temperatures. A SCR power control (Loyola Controls, Inc.) receives the thermostat output to control the heater duty cycles. The Omega thermostat also has a band width to control the temperature swings in the circulating bath about the set point. For all

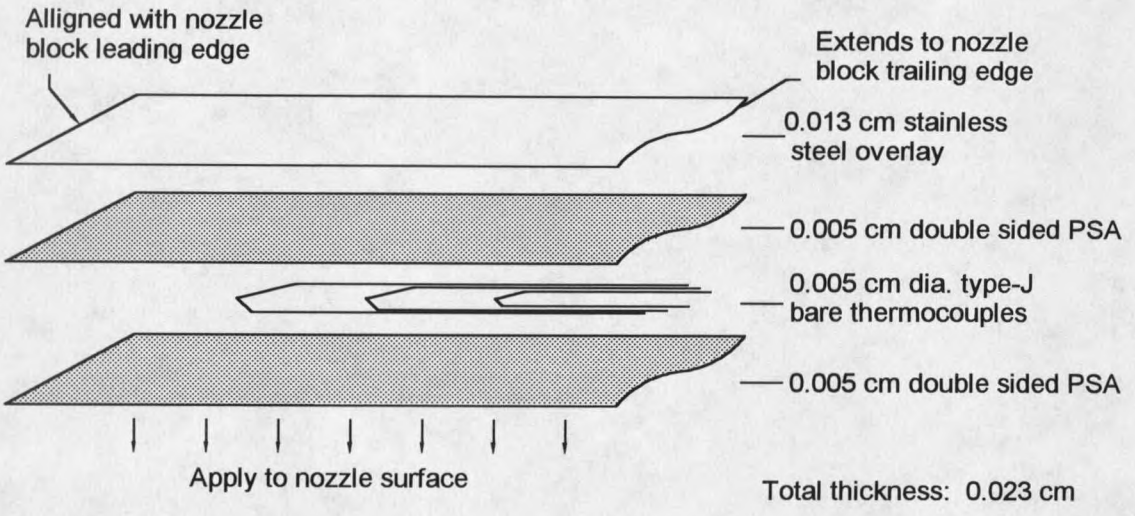


Figure 8. Layers of the thermocouple-overlay system

experiments presented here, this band width was set to maintain the fluid temperature within ±2 °C of the set point temperature.

Heated Nozzle Instrumentation

A thermocouple-overlay system was conceived to monitor surface temperature at various locations on the nozzle block surface while still maintaining the smoothness on the nozzle floor. The thermocouple-overlay system consisted of 4 layers placed on the nozzle block as shown in Figure 8. The total thickness of the thermocouple-overlay system was 0.023 cm. The thermocouples did not add to the overlay thickness since the PSA formed around the thermocouple wires. The overlay was designed as thin as possible so that the free-stream Mach number, M_∞, was not significantly altered by the presence of the overlay. Small changes in the nozzle geometry also ensured the

isentropic expansion of the original DeLeval nozzle would be maintained, and that the results of this research could be compared to previous works in the MSU-SWT. The 1.3% reduction in cross-sectional area at the throat due to the overlay thickness theoretically produced a 0.4% increase in the test core Mach number.

Bare, 0.005 cm diameter, type-J thermocouples were sandwiched between 2 layers of pressure sensitive adhesive (PSA) with a 0.013 cm polished stainless steel overlay on top. In addition to holding the overlay to the nozzle, the two layers of PSA acted to electrically insulate the thermocouple wire from the conductive overlay and aluminum nozzle block.

A double sided PSA had the advantages that it did not require high temperatures (> 150 °C) or high pressures (> 5000 mmHg, or 7 atm.) to apply. A PSA could also be removed with chemicals. The 0.005 cm thick PSA that was used (3M tape #9460) was rated for continuous use up to 150 °C, had a peel strength of 7 lbf/in, and was suitable for use under vacuum applications [30].

The thermocouple-overlay system used type-J thermocouple wire to be consistent with the ducted nozzle's 7 existing type-J thermocouples. The diameter of the new thermocouple wire needed to be small enough to allow the PSA to form around the wire without creating bumps or ridges on the nozzle surface. However, the tradeoff for a small diameter wire is high resistance which can compromise the thermocouple accuracy [31]. A 0.005 cm wire diameter was chosen because it was believed to be the largest wire that the PSA would form around.

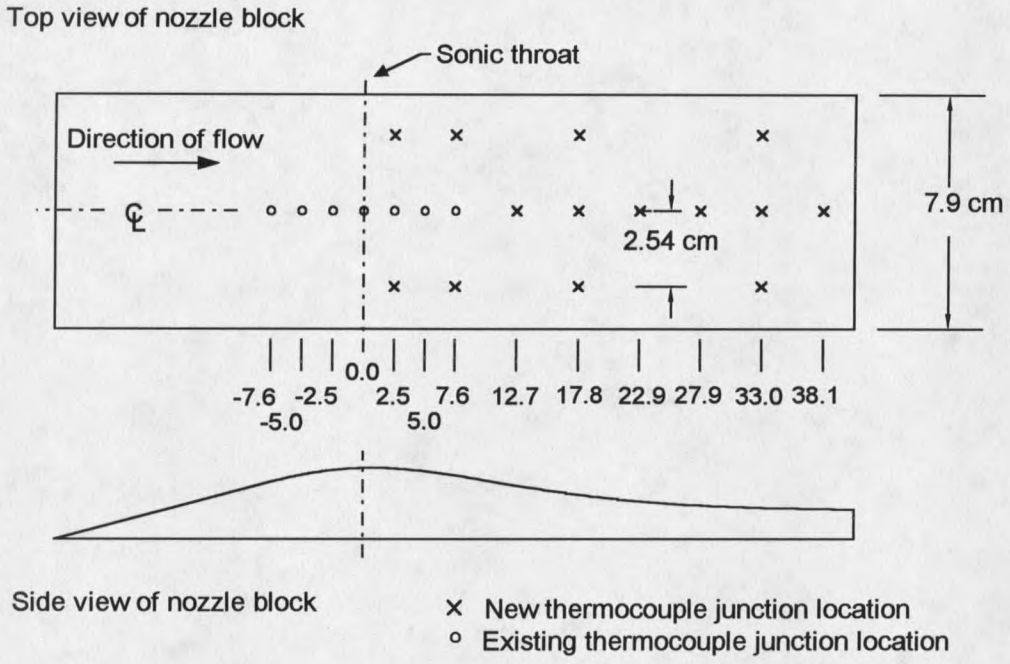


Figure 9. Spatial distribution of new and existing type-J thermocouples used to monitor surface temperature

Fourteen thermocouples were placed between the two layers of PSA in the locations shown in Figure 9. By also using the 7 existing thermocouples, the surface temperature could be monitored at a total of 21 locations.

As can be seen in Figure 9, the thermocouples provided a means to monitor both streamwise and spanwise variations in the floor surface temperature. The 0.005 cm diameter thermocouple wires were extended from each thermocouple junction to the trailing edge. At the trailing edge, each wire was soldered to its own 0.23 X 0.76 cm copper foil (electro-deposited, high-ductility 1 oz. weight) terminal strip backed with a .076 cm polyamide film. This polyamide film electrically insulated the copper foil from the aluminum nozzle block. 28 gage type-J thermocouple wire was soldered to

each terminal strip to extend the thermocouple leads out of the wind tunnel where they could be interfaced with a data acquisition system.

A smooth nozzle surface was achieved with a thin, polished stainless steel overlay placed over the top layer of PSA. With hopes to mitigate bubbling, this overlay material needed to have sufficient rigidity to prevent localized bulging. If a small pocket of air were trapped in the thermocouple-overlay system, the combination of the overlay's stiffness and the PSA bond surrounding that air pocket would prevent the air from expanding and creating a protruding bubble during tunnel flow conditions (i.e. $P < 20$ mmHg). Although the stainless steel overlay was expected to prevent bubbling by itself, during the fabrication process every effort was made to avoid trapping air between the layers of the thermocouple-overlay system.

Overlay Construction

The PSA was affixed to the nozzle following the application techniques provided by 3M -- the tape manufacturer [30]. First, the nozzle surface was cleaned with soap and water to wash off large debris and oil film incurred during handling. Then, the nozzle surface was wiped with isopropyl alcohol to clean any film left by the soap and water wash. Immediately after cleaning the nozzle surface, the PSA was applied. Since the PSA was only 10.2 cm wide, the adhesive was first oriented over the nozzle to be sure that the tape would cover the nozzle width at both the leading and trailing edges. The first several inches of tape was applied beginning at the nozzle

block's leading edge working towards the trailing edge, pressing with the thumbs, ensuring a uniform application across the width of the nozzle block. After the first several inches, the PSA was applied using a piece of 1/4" PVC plate with a machined edge. Once the PSA was applied, excess tape around the edges was trimmed.

As noted in the manufacturer's literature, a firm application pressure improves the adhesive's bond strength [30]. Pressure was applied by running the PVC tool's smooth edge along the length of the nozzle 20 to 30 times with about 44 N (10 lbf). Because this force was distributed over a very small area, it is estimated that the PVC tool allowed application pressures over 34.5 Kpa (500 psi).

A vacuum chamber was designed and constructed to simulate the low pressures reached during tunnel flow conditions. Simulating the tunnel flow conditions provided a simple means to confirm that bubbling would not occur in each layer of the thermocouple-overlay system. The chamber was designed to accommodate a MSU-SWT nozzle block as shown in Figure 10.

A Duo-Seal model 1400 vacuum pump was connected to the chamber with a flow-control gate valve, 0.95 cm copper-tubing, and compression fittings. Another port allowed the user to connect the vacuum chamber to monitor chamber pressure with the MSU-SWT pressure station, which is fully described in reference [29]. Two sealed ports in the bottom of the vacuum chamber also provided a means to use the circulating fluid heater while simulating tunnel flow conditions.

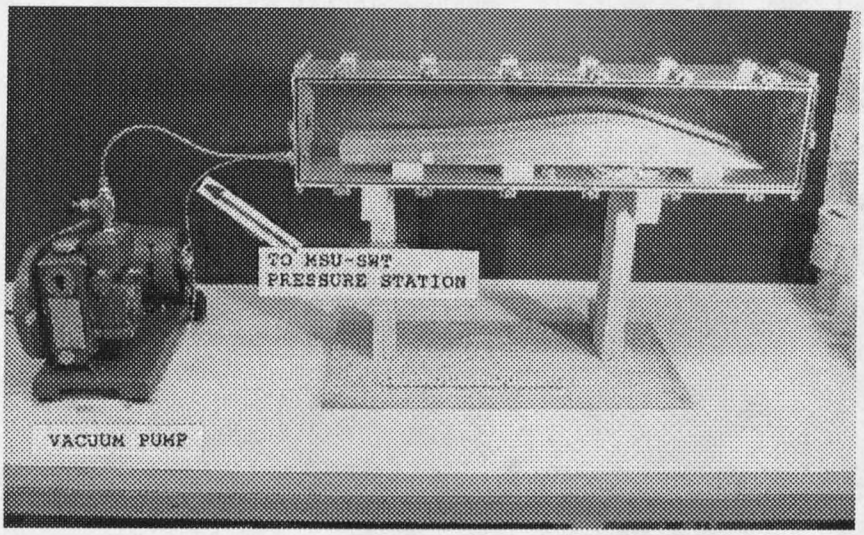


Figure 10. Vacuum test to verify quality of each layer in the instrumented overlay on the nozzle block.

After applying each layer of PSA, the nozzle block was immediately placed (adhesive not fully bonded) in the vacuum chamber to locate any bubbles. Any air trapped under the surface of the PSA formed visible bubbles and was easily located. The manufacturer's literature for the PSA indicated that 90% of the bond strength would be achieved after 24 hours and 100% bond strength after 72 hours [30].

Five bubbles approximately 0.5 cm in diameter were noted in the first 8 cm from the nozzle block's leading edge (recall that the first 8 cm were applied by hand -- not using the PVC tool). These bubbles were pierced with a razor blade and the air squeezed out. No further problems were noted with overlay bubbling in these first 8 cm. After two hours under vacuum, a small bubble grew to 1.3 cm in diameter on the nozzle's left edge at the throat. After releasing the vacuum, the bubble disappeared.

However, 3 slits were made with a razor blade in the area covered by the bubble. This area was squeezed to push any air out toward the slits. Application pressure was concentrated over the entire throat section. No further problems were noted at the throat.

The nozzle was placed in the vacuum chamber again for 30 minutes with no sign of bubbling. To avoid reducing the adhesive's bond strength, the nozzle was taken out of the vacuum chamber and put in a plastic bag, connected to the Duo-Seal vacuum pump and MSU-SWT pressure station. The plastic bag was placed under a 160 mmHg vacuum (absolute pressure) -- 25% of atmospheric -- for 24 hours. Due to difficulties in completely sealing the bag, lower pressures could not be reached.

The nozzle was put in the vacuum chamber again for 4 hours at 3 mmHg (absolute pressure). No bubbles were noted. After verifying that no bubbles were formed in the first layer, the nozzle was left to sit at room temperature for another 48 hours until the PSA reached 100% of its bond strength.

With the first layer completed, twenty-eight 0.23 cm X 0.76 cm copper foil terminal strips were placed on the adhesive at the nozzle's trailing edge. Thermocouple junctions were made using a thermocouple welder which purged the welding chamber with argon gas to provide a bead free of impurities. The thermocouple welder formed beads that were approximately 0.025 cm - 0.038 cm in diameter. Since the two adhesive layers themselves represented only 0.010 cm total thickness, the beads were flattened to less than 0.076 cm thick so the PSA could flow around the bead.

Thicknesses larger than 0.076 cm would have produced a bump in the thermocouple-overlay system and disturbed the tunnel airflow. A tool was designed and machined to precisely flatten the thermocouple beads without pinching or cold-working the thermocouple leads.

Each thermocouple was placed on the nozzle block by peeling the adhesive's protective backing only as far as necessary, placing the junction in the desired location, then laying the lead wires on the adhesive. Once a thermocouple was successfully put on the adhesive, the protective backing was replaced to keep debris and dust from accumulating on the exposed adhesive. Extreme care was taken not to touch or mar the adhesive surface. After replacing the adhesive's protective backing, the thermocouple leads were soldered to their appropriate terminal strips. This process was repeated for each of the 14 new thermocouples.

Each thermocouple was tested for proper response and lead orientation (i.e. iron on the right and constantan on the left) using a Fluke 2175A digital thermometer for type-J thermocouples both before and after placement. Once all thermocouples were placed and soldered they were tested again for proper response and orientation.

The second layer of adhesive was put on the nozzle following the same procedure used in applying the first layer. Upon testing the second layer in the vacuum chamber, several bubbles were noticed after about 2 1/2 hours. The bubbles upstream of the throat were slit through the PSA's protective backing. Bubbles downstream of the throat were pierced by first peeling the backing as far as necessary, then slitting the

area covered by the bubble. In this way, the thermocouple leads were not severed. Application pressure was concentrated on the areas covered by the bubbles. It was interesting to note that out of the vacuum chamber, the bubbles could not be seen -- even with the backing peeled away. The nozzle was placed in a bag under a 150 mmHg vacuum for 20 hours, then the nozzle was left to sit at room temperature for 48 hours.

A 0.13 cm stainless steel overlay was fabricated with tabs that fit into the O-ring grooves. Later it was determined these tabs were a major source of bubbling, and were subsequently eliminated. Upon eliminating the tabs, the bubbling problem was resolved. Once the steel was cut and formed to the nozzle, the under-side of the overlay was prepared by washing it with isopropyl alcohol. Then, beginning at the leading edge, the overlay was applied to the nozzle by slowly rolling a smooth metal tube along the length of the nozzle block. Rolling a tube (as opposed to sliding the PVC tool used in applying the PSA) prevented scratches in the overlay if dust particles were trapped underneath the PolyMask® protective covering.

Once the overlay was affixed to the nozzle, pressure was applied by repeatedly rolling the metal tube back and forth along the length of the nozzle. The metal tube allowed for estimated application pressures over 35 KPa due to the small contact area between the tube and overlay.

The stainless steel overlay was not initially tested in the vacuum chamber since there was no way to pierce the bubbles trapped between the steel and adhesive.

However, to increase the bond strength, a piece of 0.95 cm PVC plate was tightly clamped to the nozzle surface every 2.5 - 5.0 cm along the nozzle centerline. The clamped PVC plate provided continuous application pressure throughout the 72 hour curing time. The 0.95 cm PVC plate had good rigidity to evenly distribute the force of the clamps while still being flexible enough to form to the nozzle contour when clamped.

The bond strength of the PSA was further increased by heating the nozzle and overlay to approximately 65 °C with the circulating fluid heater discussed earlier, for 1 1/2 hours as recommended in the 3M literature [30]. Applying heat while the adhesive was curing allowed the adhesive to "flow" into the adjoining surfaces.

After removing the PVC, the nozzle was polished by hand using a mixture of 0.05 micron Gamma Alumina powder and water. The complete nozzle overlay showed several bubbles near the stainless steel overlay tabs when tested in the wind-tunnel; however, upon eliminating these tabs, the bubbles were eliminated. Figure 11 shows a photograph of the nozzle block surface. The Reynolds number based on roughness height at the throat was estimated at less than unity for all flow conditions tested.

Initially, there was a "wavy" character of the overlay shown in Figure 11. Although it has been assumed these undulations arose from the pliable nature of the PSA, their actual origin is unknown (e.g. pressure application method, curing procedure). Measurements taken every 2.5 cm from $x = 7.6$ cm to $x = 38.1$ cm with no heating showed no significant perturbations in the core flow. It should also be

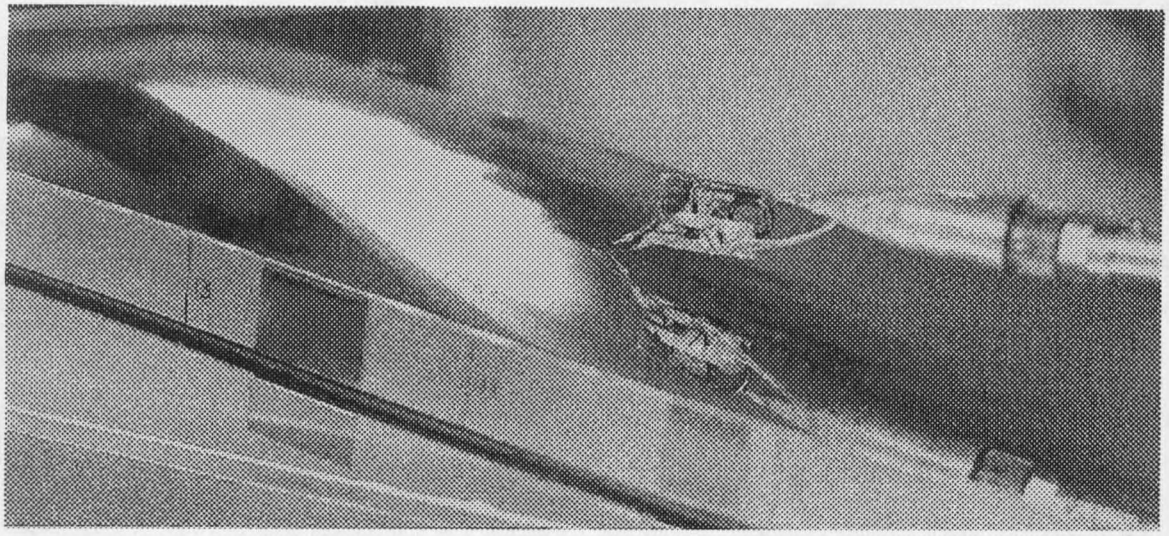


Figure 11. Photograph of the instrumented nozzle block initial surface quality.

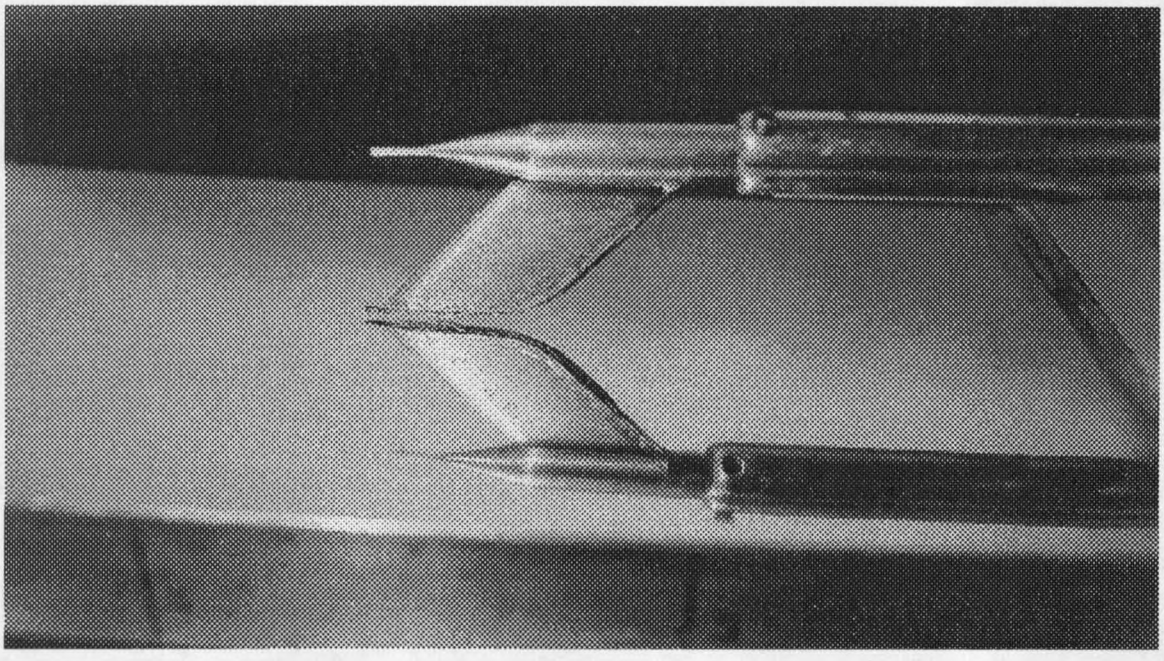


Figure 12. Photograph of instrumented nozzle block surface quality after sufficient time to eliminate waviness.

noted that the optical qualities of the polished overlay act to magnify the observed undulations. The average distance from peak to valley in these undulations was estimated at less than 0.002 cm (20 μm). This wavy character of the surface was also minimized with surface heating and diminished over time as the experiments continued. Figure 12 shows the nozzle surface quality after the undulations were eliminated.

Data Acquisition System

The SWT automated data acquisition system (ADAS) of reference [32] was used to control the overhead traversing mechanism and record various voltages during each boundary layer survey. This system consists of a Zenith, Z-100 computer with an analog-to-digital, digital-to-analog card which allowed voltages to be stored directly onto disk for data reduction at a later date. The SWT-ADAS has a positional accuracy of 0.0015 cm and moves the probe vertically in increments specified by the user. Increments used in this investigation ranged from 0.0025 cm to 0.013 cm depending on the boundary layer thickness.

Each pitot pressure and total temperature profile was also recorded in analog form onto paper plots. The analog plots were recorded directly from the amplified thermocouple and pressure transducer outputs.

The nozzle block surface and circulating fluid temperatures were continuously monitored through all experiments using the Hewlett-Packard Data Acquisition System

(HPDAS). The latter consisted of a 20 channel hardware compensated, type-J thermocouple card, a 20 channel general thermocouple card, a 3456A digital voltmeter, and a 3497A data acquisition control unit.

Profile Measurement Instrumentation

Pitot pressure, P_t', and the local total temperature, T_t', were measured at discrete points through the boundary layer to obtain a boundary layer "profile." Figure 13 shows the circuit used to record these measurements. The pitot pressure and local total temperature were recorded simultaneously using the dual probe arrangement shown in Figure 14. The SWT-ADAS described earlier was used to control the vertical probe movement while recording the profile quantities in a measure-move-pause-measure sequence of probe motion through the boundary layer.

The dual probe's pitot component consisted of a small tube connected to an encapsulated transducer. The latter was a Kulite pressure transducer (#I2-36) rated for 0-5 psia with a sensitivity of approximately 0.25 mV/mmHg. An excitation of 5.00 VDC was used to power the transducer. The probe tip had an outside opening height of 0.0064 cm and a width of 0.011 cm. The boundary layer thickness ranged from approximately 0.07 cm to 0.25 cm for a vertical resolution of at least 10:1. Care was taken to ensure that the probe opening was perpendicular to the flow. Chambre and Schaaf [33] indicate pitot measurements will be within 1% of the theoretical value for yaw angles less than 10 °. The small volume, calculated to be approximately

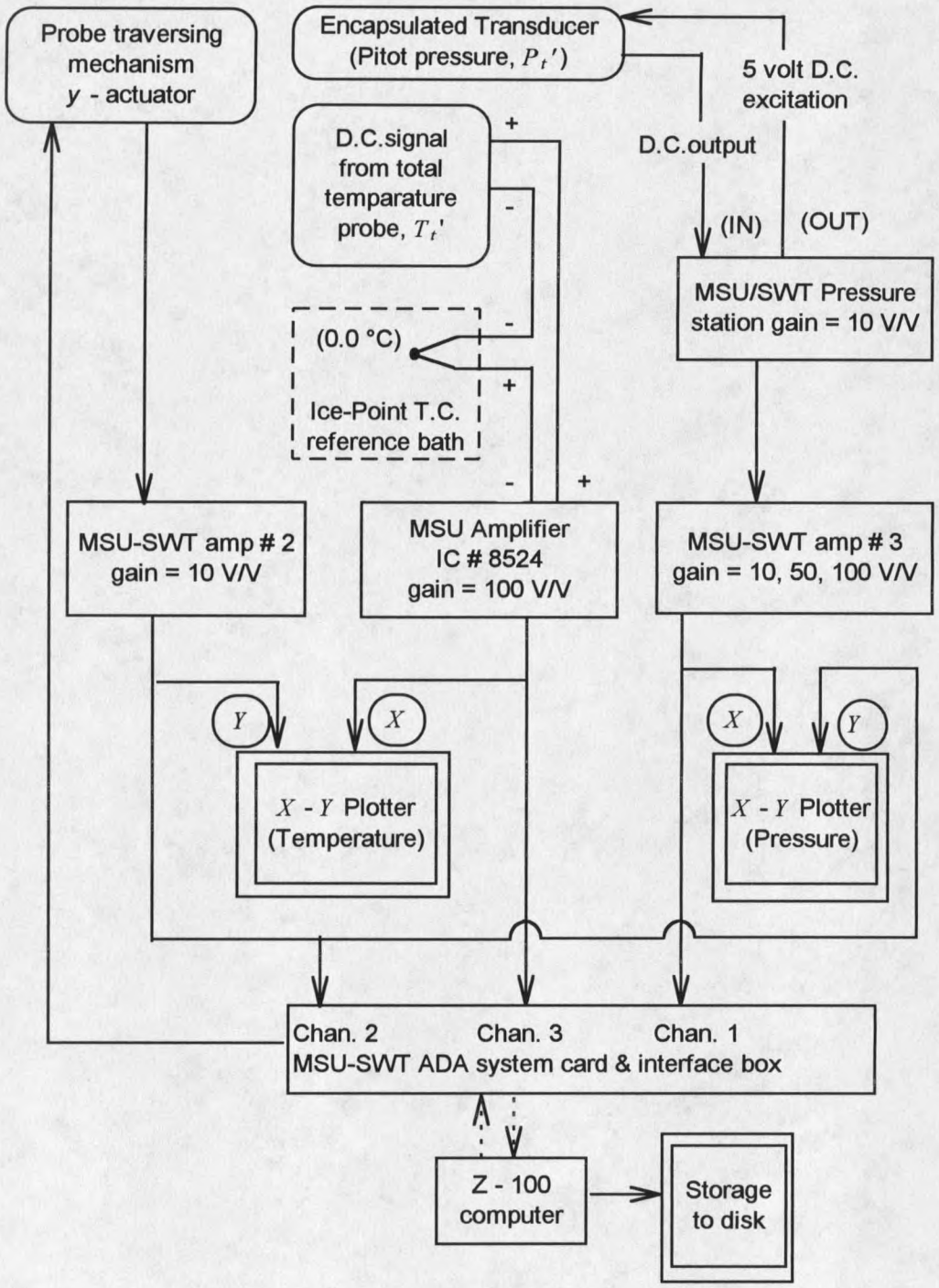


Figure 13. Recording circuit for profile measurements.

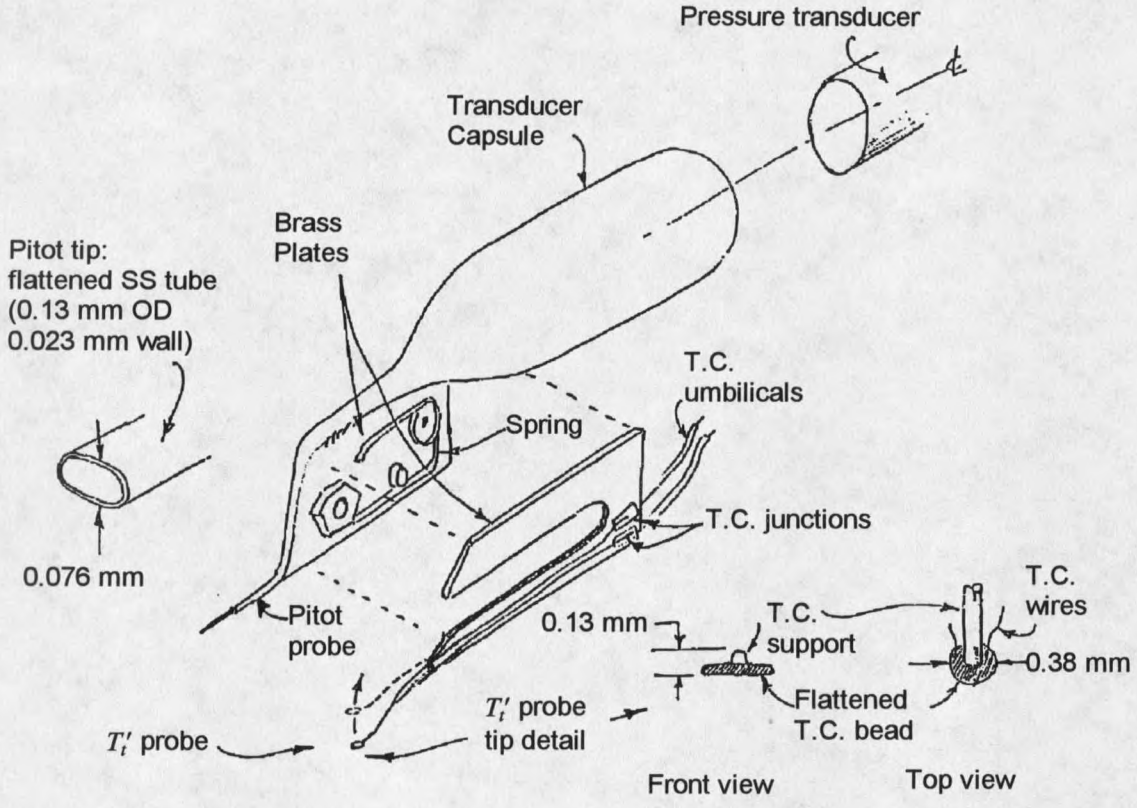


Figure 14. Drawing of the dual probe for measuring P_t' and T_t' through the boundary layer.

0.10 cm^3, allowed for quick response times (i.e. < 7 sec.) to reach a steady state condition. Transducer voltages were recorded at voltage gains of 100, 500, and 1000 such that the transducer output spanned the full voltage range allowed by the ADAS. The transducer output was calibrated at each gain setting with all circuitry connected that was used in the actual data acquisition. Thus, each calibration included the effect of all components in the recording circuit. All calibrations fit along a straight line with an r^2 better than 0.99993.

The total temperature component of the dual probe was a type-J thermocouple bead mounted to a 0.010 cm diameter steel wire. As shown in Figure 14, the thermocouple bead was flattened into a disk to reduce the frontal height, and thereby increasing the probe's vertical resolution. The modified bead provided overall frontal dimensions of 0.013 cm high by 0.038 cm wide. The small mass of the thermocouple bead also allowed quick response times (e.g. less than 1 second for a temperature change of 10 °C).

The thermocouple bead was laterally offset from the pitot probe by approximately 0.3 cm, and was placed in the same streamwise position as the pitot tube opening. The total temperature probe was not fixed rigidly to the dual probe but was instead held in place by a small spring. The spring held the thermocouple rigid enough to keep the T_t' probe at a fixed position relative to the pitot component while the dual probe traversed through the boundary layer. However, the spring supplied enough flexure that the actuator could push the disk against the nozzle surface. The same total temperature probe and pressure transducer were used for the entire experiment. Several different pitot probe tips were used; all were of the same tip geometry.

The actual thermocouple was constructed of bare 0.005 cm iron and constantan wires. The thermocouple bead was glued to a 0.010 cm steel wire which in turn was fixed to a brass plate that could be screwed to the body of the dual probe (see Figure 14). The bare thermocouple wires were offset from the steel support wire approximately 0.003 cm with a thin coating of radio cement. A PSA was used as a

dielectric and to hold the 0.005 cm wires along the lower portion of the brass plate. The larger type-J thermocouple wires (28 gage) were then used to relay the thermocouple signal to the instrumentation outside of the tunnel. An ice bath was used as an external reference junction. The thermocouple circuit output voltage was amplified with a voltage gain of 100. This voltage was converted to temperature in the data reduction software using a fifth order polynomial.

The compliant temperature probe design had two advantages. First, the experimenter did not need to know the vertical distance between the thermocouple bead and pitot probe. For all profile surveys, the dual probe was lowered to the nozzle so that the pitot probe was within 5×10^{-4} cm from the surface. This precision was obtained with the aid of a 50X magnification microscope. At this point, the total temperature probe was being pressed to the surface with the spring. The profile survey began and the data point at which the total temperature probe lifted off was marked in the digital record. Then through the data reduction software the y-position of the total temperature probe could be adjusted to reflect that lift-off point as $y = 0$ for the total temperature probe. The second advantage of the compliant temperature probe was that the force produced by the spring reduced the contact resistance between the thermocouple bead and the nozzle surface. The flattened bead also provided a greater surface area between the nozzle and the thermocouple bead.

CHAPTER 6

DATA REDUCTION

Profile Reduction

Each boundary layer survey was reduced following the steps shown in Figure 15. For each profile, the stagnation pressure, P_o, and temperature, T_o, were recorded. Then, given a boundary layer profile of measured P_t' and T_t' data, all "edge" properties were calculated (detailed below). The "edge" as used here refers to the location at which the thermodynamic conditions are those of the free-stream as seen by the boundary layer. Also note that the edge, by definition, must always be beyond the boundary layer thickness, δ.

With the boundary layer edge conditions calculated, along with the measurements of P_t' and T_t', each data point in the profile was reduced with the following procedure. Using the classical boundary layer approximation, $\partial P/\partial y = 0$, the edge pressure, P_e, can be assumed constant through the layer (i.e. $P = P_e$ at all points). If the flow is subsonic (i.e. $P/P_t' > 0.5248$), $P_t = P_t'$ and the local Mach number, M, is calculated from the isentropic relation

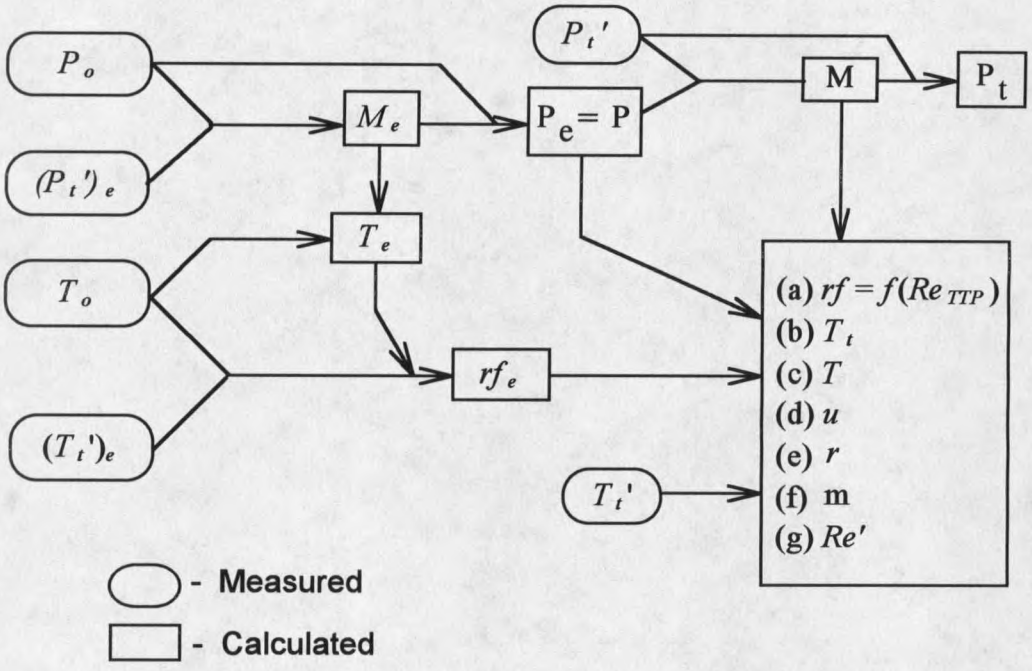


Figure 15. Steps to calculate the desired boundary layer quantities from measurements.

$$\frac{P}{P_t} = \left(1 + \frac{\gamma - 1}{2} M^2\right)^{-\gamma/(\gamma-1)} . \tag{2}$$

Otherwise, there was a shock wave in front of the pitot tube, and M is calculated from the Rayleigh supersonic pitot formula [34, p. 149],

$$\frac{P}{P_t'} = \left(\frac{2\gamma}{\gamma+1} M^2 - \frac{\gamma-1}{\gamma+1}\right)^{1/(\gamma-1)} \left(\frac{\gamma+1}{2} M^2\right)^{-\gamma/(\gamma-1)} . \tag{3}$$

Knowing M and P, the total pressure, P_t, can then be calculated from the isentropic relation, equation (2).

All temperatures and temperature dependent properties are then calculated with an iterative procedure to include the Reynolds-number-dependent recovery factor of the total temperature probe. The steps within each iteration are as follows. (a) Calculate the recovery factor based on a linear dependence of the Reynolds number based on the total temperature probe height, Re_{TTP}. (b) Calculate T_t from M and rf. (c) Calculate T from the isentropic relation,

$$\frac{T}{T_t}=\left(1+\frac{\gamma-1}{2}M^2\right)^{-1}. \tag{4}$$

(d) Calculate velocity, u, from M and T from the definition of M,

$$M=\frac{u}{\sqrt{\gamma RT}}. \tag{5}$$

(e) The density, ρ, is calculated from the ideal gas law, and (f) the viscosity, μ, is calculated from Sutherland's formula [8, 35] for air with the stagnation conditions as a reference

$$\frac{\mu}{\mu_o}=\frac{T_o+110}{T+110}\left(\frac{T}{T_o}\right)^{3/2}, \tag{6}$$

where T_o and T are in Kelvin.

The Reynolds number is then calculated directly from its definition. A new recovery factor is calculated based on this Reynolds number. If the solution has not converged, Re_{TTP} is updated, and the iteration continues. Upon convergence, the data

reduction procedure is repeated for the next data point, and so-on, until the profile has been completed.

After the boundary layer profile was reduced to obtain velocity and density profiles, the boundary layer thicknesses could be calculated. The boundary layer thickness, δ, was determined by the y-position at which $u/u_e = 0.99$. The displacement and momentum thicknesses were calculated from their definitions:

$$\theta = \int_0^\infty \frac{\rho u}{\rho_e u_e}\left(1 - \frac{u}{u_e}\right) dy \tag{7}$$

$$\delta^* = \int_0^\infty \left(1 - \frac{\rho u}{\rho_e u_e}\right) dy \tag{8}$$

Integration was carried out by applying the trapezoidal rule from the wall ($y = 0$) to the first data point of the boundary layer "edge."

A FORTRAN program was written to automate the above calculation procedure to reduce many profiles at once. Appendix A shows the flowcharts and program listing. As a note to the interested reader, the program performs the above computations with all profile quantities normalized with their respective edge values. All edge values in the program are normalized with respect to the dimensional stagnation reference conditions.

Recovery Factor

The significant total temperature variations through the boundary layer made it necessary to use a recovery factor relating the total temperature measured by the probe, T_t', to the actual local total temperature, T_t. Under adiabatic conditions, the T_t variation through the boundary layer could be reasonably approximated as a linear profile between the wall temperature, T_w, and local total temperature at the boundary layer edge $(T_t)_e$, where $(T_t)_e$ is assumed equal to the stagnation temperature, T_o. However, the heated wall profiles required a more accurate calculation of T_t. Accordingly, T_t', was measured with a type-J bare bead thermocouple as described in chapter 5 (Experimental Setup).

The recovery factor curve for the total temperature probe was determined from experimental data taken in the free-stream. The data range to calculate a recovery factor was limited to free-stream measurements in the MSU-SWT. Consequently, M ranged from 2.2 to 3.0, and Re' ranged from 31,000/cm to 77,000/cm; where Re' is calculated using properties at the local static conditions. Recovery factors are often calculated as a function of M and of Re' in the literature (e.g. references [36, 37]). Therefore, recovery factors for $(T_t')_e$ were investigated as a linear function of the Reynolds number based on the total temperature probe tip height, Re_{TTP}, and also as a linear function of M. Based on a .013 total temperature probe height, Re_{TTP} ranged from 0 to 980 for all boundary layer surveys.

Figure 16 shows the recovery factor plotted as a function of M, with Re_{TTP} as a parameter. As seen in Figure 16, the recovery factor had an increasing trend as M decreased. There also appeared to be an additional dependence upon Re_{TTP}. However, a plot of rf versus Re_{TTP} (Figure 17) revealed that the recovery factor was a linear function of only Re_{TTP} for $2.1 < M < 3.0$. Note that Re_{TTP} was based on the local static conditions at the probe tip and not the local total conditions. Apparently, the Mach number dependence was accounted for through the isentropic variations in the static conditions (ρ,T,μ). The recovery factor, rf, used here is defined as [36]

$$rf = \frac{T_t' - T}{T_t - T} \tag{9}$$

Using the isentropic relation between the static and total temperature given by equation (4), and the definition of rf from equation (9), one could obtain an expression for T_t as a function of rf, T_t', and M:

$$\frac{T_t}{T_t'} = \frac{1 + \frac{\gamma - 1}{2} M^2}{1 + rf\left(\frac{\gamma - 1}{2} M^2\right)} \tag{10}$$

The significant feature of equation (10) is that errors in rf were dampened for $M < 1$. Therefore, in the context of boundary layer surveys, as the probe approached the wall velocity calculations become extremely sensitive, errors in rf became less critical.

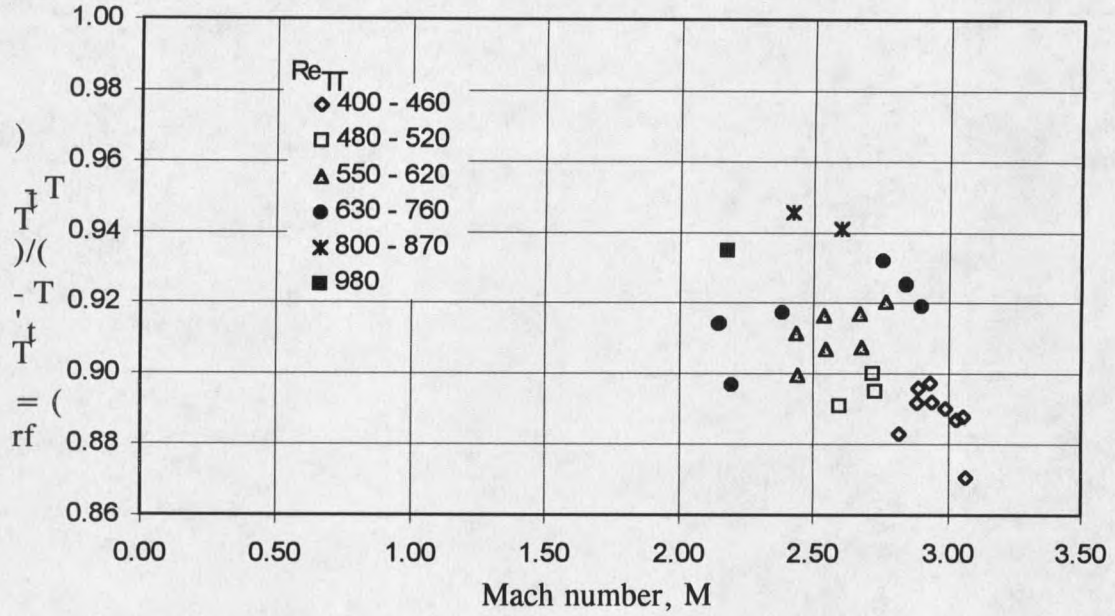


Figure 16. Recovery factor, *rf*, vs. *M* with Re_{TTP} as a parameter. $400 < Re_{TTP} < 900$

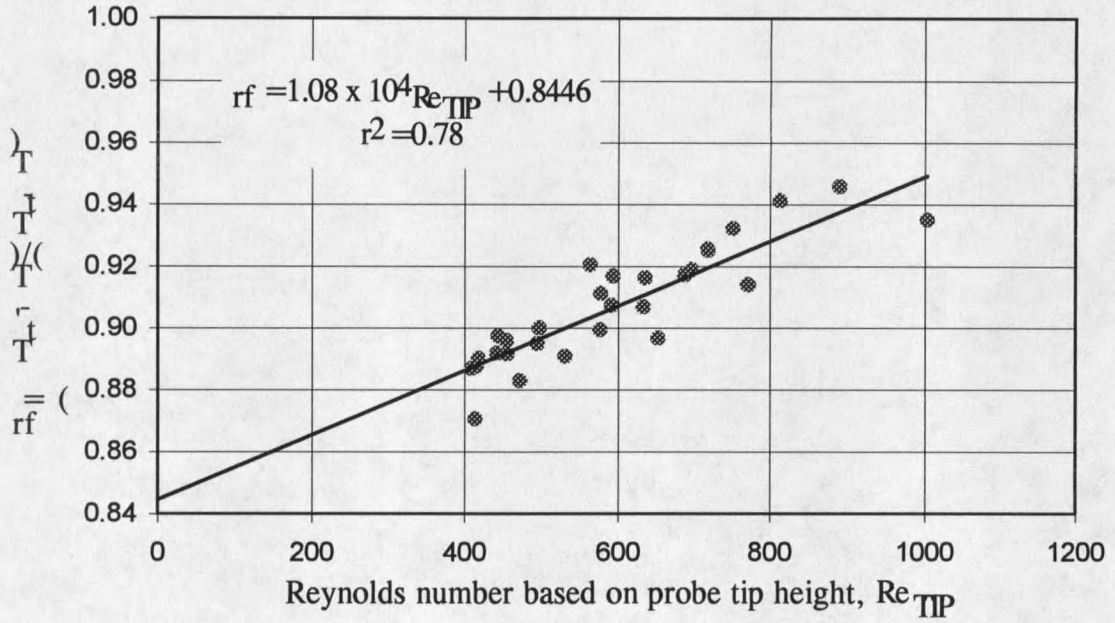


Figure 17. Recovery factor, *rf*, vs. Re_{TTP}, Mach number range is $2.1 < M < 3.0$

The probe temperature is also commonly related to the local total temperature by the ratio $T_t'/T_t = rf$[1], where rf is again determined from experimental data. The same dependencies shown in Figures 16 and 17 were noted for this second definition of a "recovery factor" with slightly less scatter in the Re_{TTP} plot. However, due to the error-dampening nature of equation (10), the classical definition of rf (equation (9)) was chosen for data reduction. Regardless of which recovery factor model is chosen, errors in T_t are strongly minimized in the velocity calculation. For example, a 1% variation in T_t=300 K, translates to variations less than 0.5% in the velocity, u.

A least squares calculation was done to obtain a linear relation for rf as a function of Re_{TTP}. The linear relation was termed the "global" curve, since a range of boundary layer edge conditions ($7.6 < x < 30.5$ cm, and varied stagnation conditions) was used in the calculation. The value of rf at $Re_{TTP} = 0$ was taken as the recovery factor infinitely near the wall, rf_w. Theoretically, rf is undefined at the wall since the static and total temperatures are the same. However, errors in rf near the wall became insignificant in the calculation of T_t [recall discussion of equation (10), above].

To preserve the physical law that $T_t = T_o$ at the boundary layer edge (assuming isentropic expansion), it was necessary to force the recovery factor curve to yield this result for each boundary layer profile. This task was achieved by (a) determining the boundary layer edge, (b) calculating rf_e directly from equation (9), assuming $T_t = T_o$,

[1] The term "recovery factor" is being used loosely in the mathematical sense. Qualitatively, the author is using it to mean a general relation between T_t' and T_t. The true definition of recovery factor is given by equation (9) [36].

then, (c) from rf_e and rf_w, calculating the slope of the rf curve. This "local" recovery factor curve was then used for all data points in that profile. Separate global curves were obtained for heated wall profiles and for adiabatic wall profiles.

Edge Calculation

The location of the boundary layer edge could be determined by inspection or by using the boundary layer reduction program. The latter method used a 5-point moving average to determine a local maximum in the pitot pressure profile. This moving average started near $y = 0$ and incremented through the boundary layer one data point at a time. The moving average was calculated from the current data point and the next n consecutive data points, where n was some user specified integer. A value of $n = 5$ was used for all profiles. The computer code checked to see if the current average was less than the previous average. If the current average was greater, the procedure was incremented to the next data point. At the first instance where the current average was less than the previous average, the procedure stopped and the boundary layer "edge" was located. The edge pitot pressure, P_t', was then calculated from that previous average (the local maximum). Using P_t', and assuming $P_t = P_o$ at the boundary layer edge, the edge Mach number, M_e, could be calculated through the normal shock relation [34, p. 148],

$$\frac{P_t}{P_t'} = \left(\frac{2\gamma}{\gamma+1}M^2 - \frac{\gamma-1}{\gamma+1}\right)^{1/(\gamma-1)} \left(\frac{1+\frac{\gamma-1}{2}M^2}{\frac{\gamma+1}{2}M^2}\right)^{\gamma/(\gamma-1)} \quad (11)$$

Given M_e and $(P_t)_e = P_o$, the static pressure, P, was calculated from the isentropic relation

$$\frac{P}{P_o} = \left(1+\frac{\gamma-1}{2}M_e^2\right)^{-\gamma/(\gamma-1)} \quad (12)$$

The $(T_t')_e$ measurement was used to calculate the edge recovery factor assuming that $(T_t)_e = T_o$ as described above. The rest of the static conditions at the edge were calculated following the same procedure used for points in the boundary layer, as detailed in the Profile Reduction section above.

In all profiles, the thermal boundary layer "edge" was assumed to coincide with the hydrodynamic boundary layer. This choice was justified because the location of the edge as determined by the moving average method was invariably at least 1.5 boundary layer thicknesses away from the wall. Therefore, even with the thermal boundary layer being slightly larger than the hydrodynamic boundary layer, a true free-stream measurement was obtained for $(T_t')_e$. Furthermore, the edge location chosen by the computer program was inspected with regard to both T_t' and P_t' profiles to ensure that the edge was in fact representative of the free-stream conditions. The free-stream was easily recognized by the region in which the T_t' and P_t' measurements no longer

changed with distance from the wall. The "moving average" method of determining the boundary layer edge worked remarkably well and was used for all profiles.

Near-wall Points

Near the surface, significant errors were produced in the pitot probe readings due to viscous effects, wall-probe interference, and the extreme sensitivity of the probe reading relative to the static pressure. These P_t' readings were obviously in error because the velocities at the wall were calculated to be greater than zero which violates the no-slip condition of boundary layer flows. For the purpose of calculating the δ^* and θ, these errors were circumvented by assuming a linear velocity profile between $u = 0$ at the surface ($y = 0$) and the nearest data point which appeared valid as shown in Figure 18. Typically, this first valid point occurred at about $y/\delta = 0.2$ and $u/u_e = 0.4$. Using the assumed velocity points, P, and T_t' as the independent variables, M, P_t, rf, T_t, T, ρ, μ, and all other velocity related quantities were recalculated.

Offset Correction

Forty of the profiles recorded were not useable in their original digital form due to the low signals being measured and unfiltered noise in the recording circuit. The digital profiles (temperature and pressure) that contained too much data scatter to be of value were restored by reconstructing the digital record from analog plots.

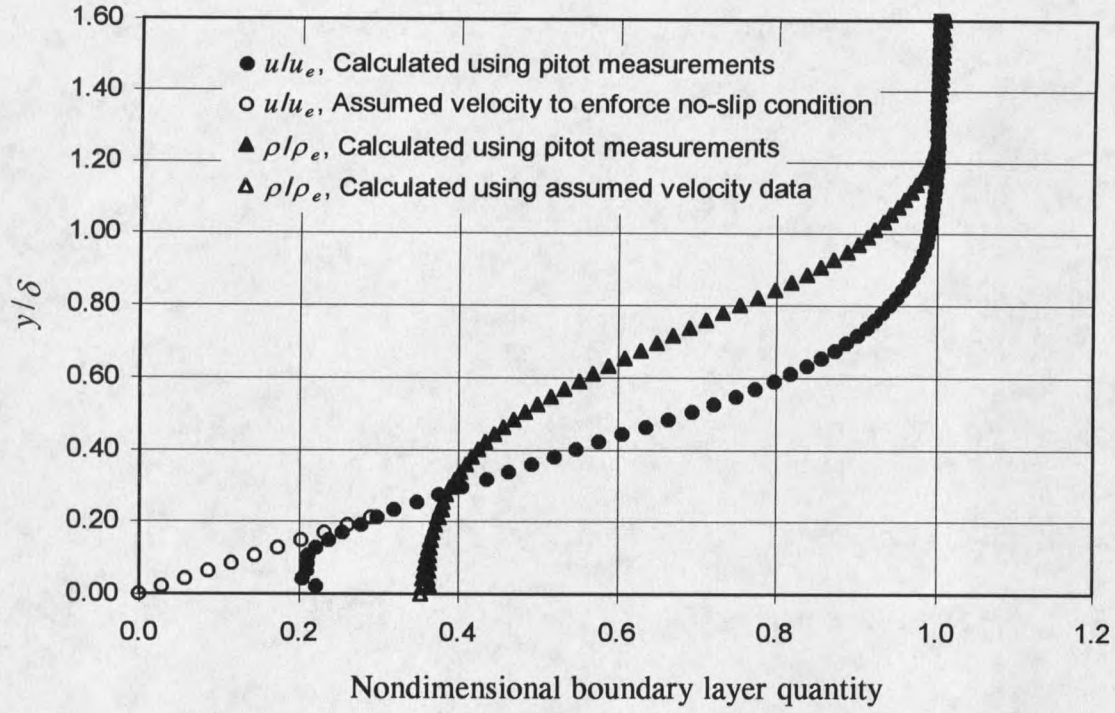


Figure 18. Treatment of near-wall points for calculating integral properties.

Analog plots were digitized using AUTOCAD (a drafting software package) and a digitizing tablet. For each profile, the tablet was calibrated to the coordinates of the plot, then each data point was "picked" with a pointing device and recorded into a file. A C-program was written to extract the x-y coordinates from the AUTOCAD file, transform the plotted units to the corresponding transducer voltage, and y-position, and write the recovered data to a file. This file could then be treated the same as the useable direct-to-digital records. The y-position recorded on the analog plots was converted to actual distance by letting the first point of the curve be y = 0.0 cm then using 2 points on the analog plot to calculate a "gain" as shown in Figure 19.

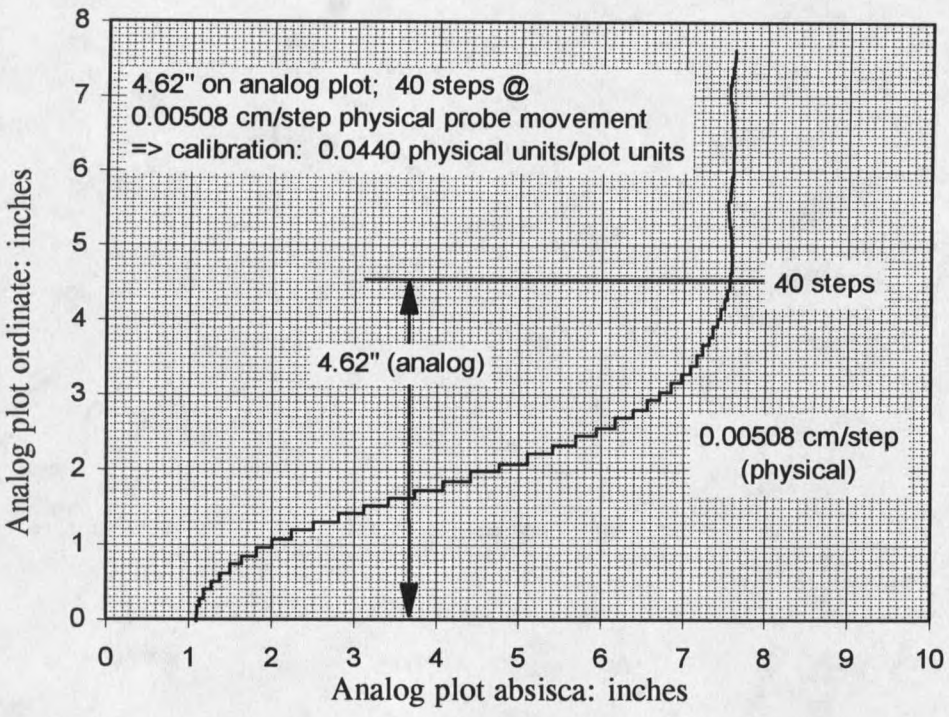


Figure 19. Technique to calculate y-calibration from analog to physical units.

The same 40 profiles also required a small adjustment to the pressure calibration. The original calibration showed the total pressure near the wall was less than the static pressure -- a physically impossible result. Furthermore, these profiles showed total pressures near the wall which were consistently 3 to 4 mmHg higher than similar measurements taken at a different transducer calibration. In the free-stream, this offset would cause an error of less than 1% in the pitot pressure reading. However, pitot pressures near the wall were less than 15 mmHg, causing at least a 20% error. Therefore, the profile was "corrected" by adjusting the transducer offset such that the difference between the pressure at the first data point and the static pressure

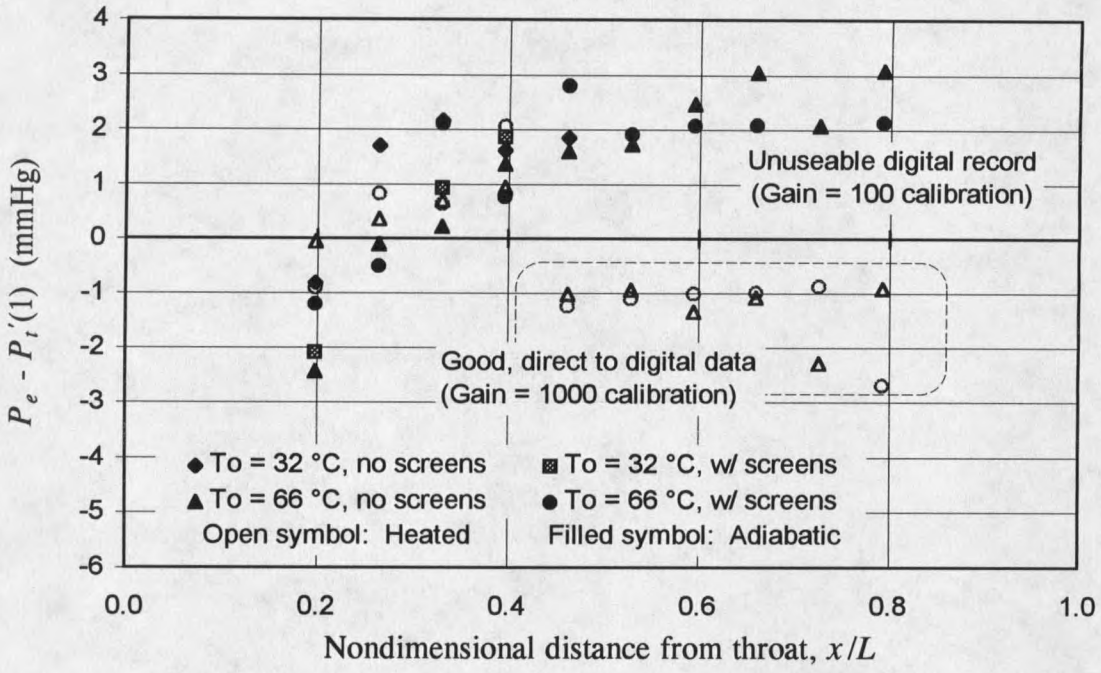


Figure 20. Wall-pitot pressure measurement before applying offset correction.

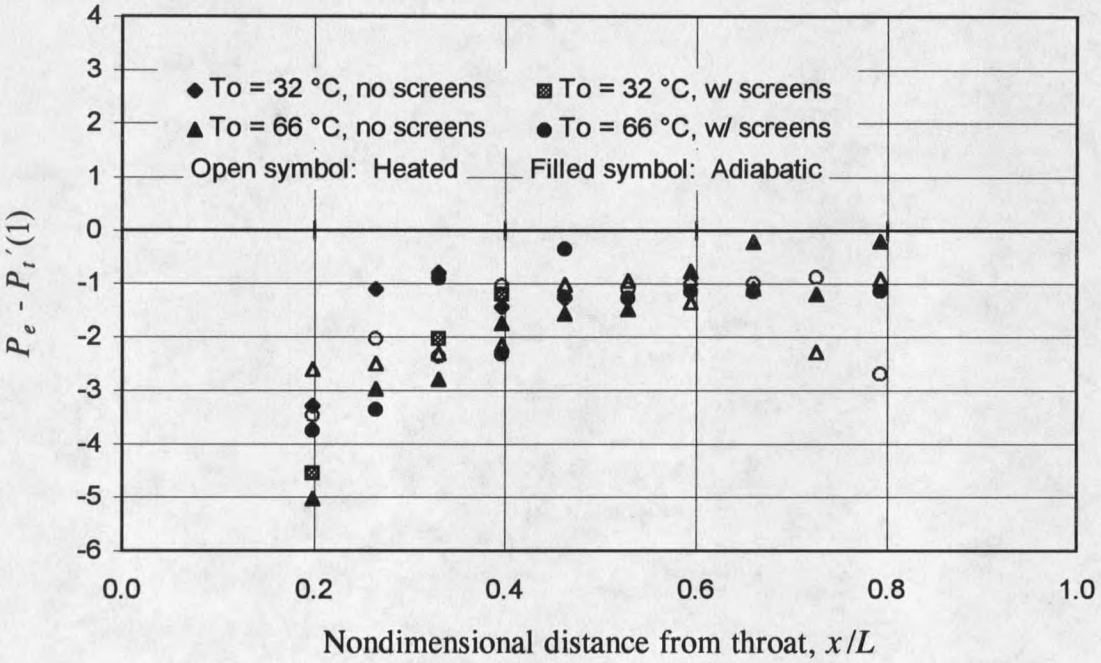


Figure 21. Wall-pitot pressure measurement after applying offset correction.

measured at the boundary layer edge was similar to the difference found in other boundary layer measurements at the same P_o and T_o. Figures 20 and 21 show this pressure difference before and after applying the correction. The ordinate is the pressure difference in mmHg, and the absisca is the streamwise location, x, normalized with the nozzle length, L. The same value for the offset correction was applied to all reconstructed profiles.

CHAPTER 7

RESULTS

Preliminary Investigations

Before acquiring profile data it was necessary to confirm that the instrumented nozzle block did not introduce any anomalies or perturbations in the tunnel flow because of the overlay and other effects. Figure 22 shows Mach number traces taken at various streamwise locations through the vertical plane of symmetry. The uniformity of M through the vertical center-plane (i.e. $\Delta M < 0.05$) reveals there were no significant shock-waves produced by the instrumented nozzle. Note that the larger variation of M seen at x=27.9 cm is partially due to the fact that the nozzle was still diverging and the test core Mach number, M_∞, had not been reached through the entire tunnel cross section.

Figures 23 - 25 show the streamwise variation of the boundary layer edge Mach number, M_e, computed numerically and measured by experiment. The theoretical curve was obtained from the two dimensional Navier-Stokes solution [22] using the nozzle design coordinates [29] and stagnation conditions, P_o = 350 mmHg, and

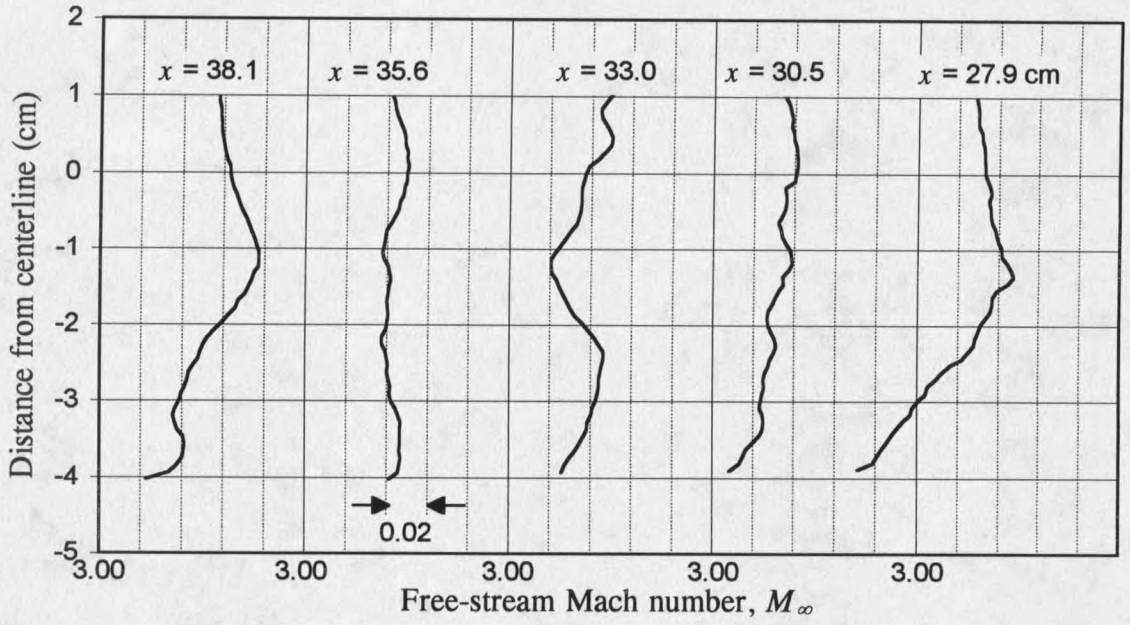


Figure 22. Free-stream quality with instrumented nozzle block installed (unheated condition).

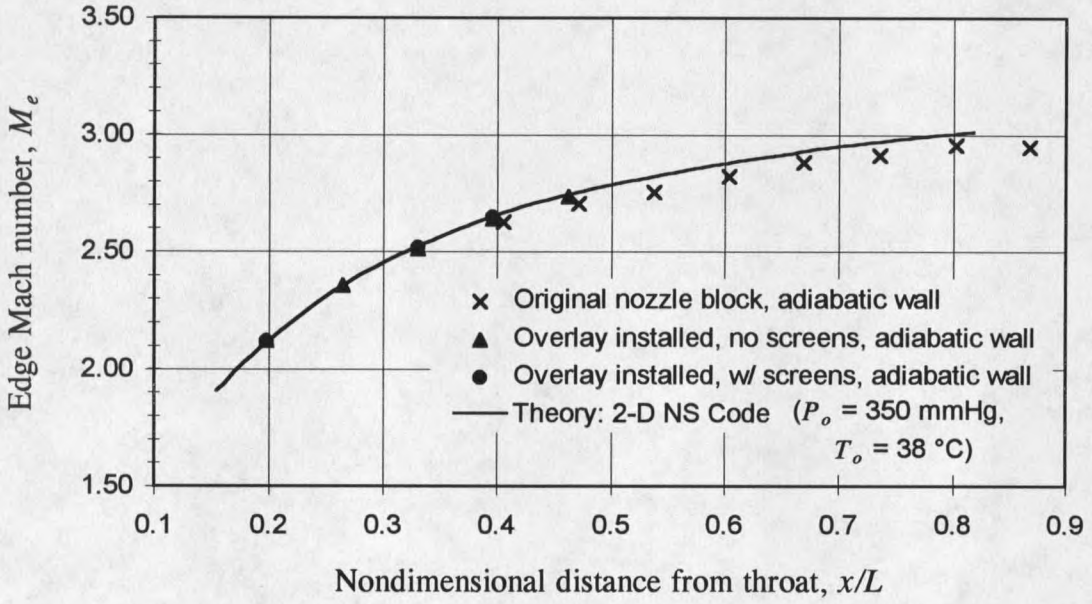


Figure 23. Edge Mach number, M_e, variation, case 1a and 1b, (P_o = 400 mmHg, T_o = 32 °C, adiabatic wall; shows effect of overlay on free-stream quality.

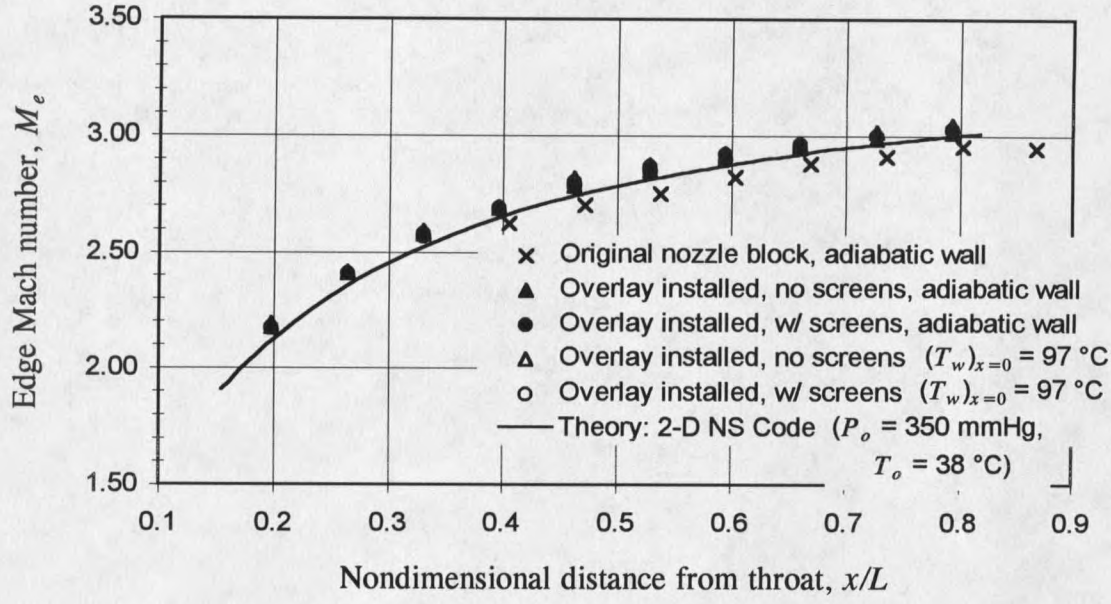


Figure 24. Edge Mach number, M_e, variation, case 2a - 2d, (P_o = 400 mmHg, T_o = 66 °C, adiabatic and heated wall); effect of overlay.

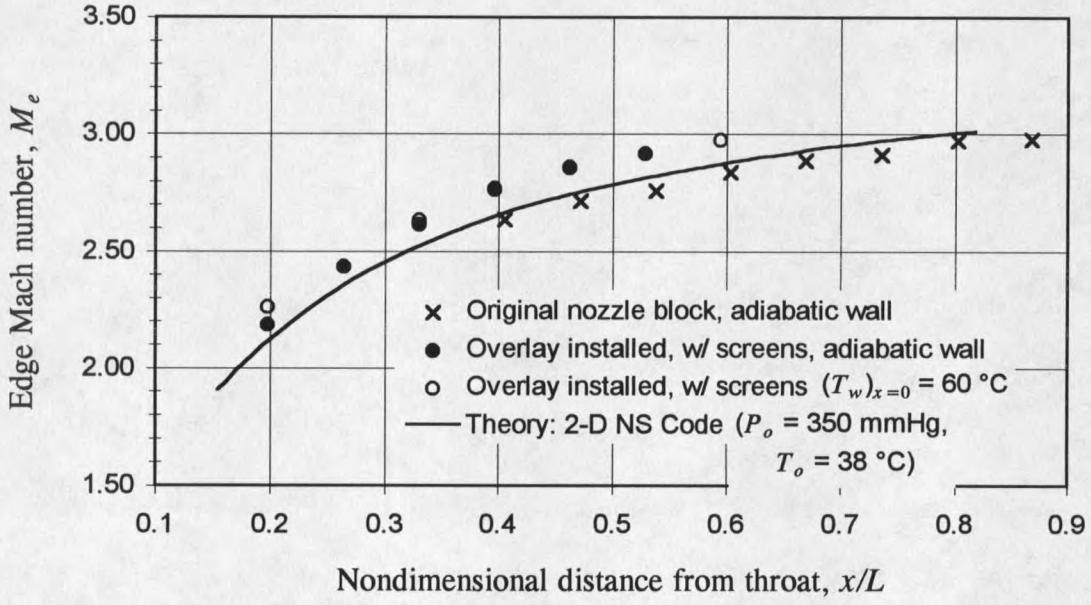


Figure 25. Edge Mach number, M_e, variation, case 3a and 3b, (P_o = 500 mmHg, T_o = 21 °C, adiabatic and heated wall); effect of overlay

T_o = 38 °C. Recall that the P_e/P_o distribution along the nozzle centerline, which is directly related to M_e by equation (12), was used as input to the boundary layer code. As discussed in Chapter 4 (Theory), M_e was approximated to be independent of P_o and T_o for the conditions of the present research.

The stagnation conditions of case 1a and 1b (P_o = 400 mmHg, T_o = 32 °C) were closest to those of the 2D Navier-Stokes numerical computations; as expected, cases 1a and 1b show the best agreement between theory and experiment. Conversely, cases 3a and 3b (P_o = 500 mmHg, T_o = 21 °C) show the worst agreement between theory and experiment. However, variations are still less than 4% for all profiles investigated. Measured M_e data for the original nozzle block [29] has also been included in Figures 23 - 25 to show the effect of the instrumented overlay on M_e. The graphs show that, as expected, M_e was increased due to the relative reduction in cross-sectional area at the throat (resulting in an increased area-ratio downstream). Recall that the thickness of the overlay reduces the cross-sectional area at the throat by a larger percentage than locations downstream of the throat. It should also be noted that the theory predicted higher values of M_e than the experimental values obtained with the original nozzle configuration. This minor difference between the theory and the original configuration was also confirmed in reference [22] for $x \leq 3$ cm ($x/L \leq 0.2$).

Both screens and heating produced a negligible effect on M_e. In fact, for a given streamwise location, most measurements for various combinations of the test matrix, were within 0.5% of each other. Although this result is partially indicative of

displacement thickness as a function of surface heating, the overall change in area ratio, and thus M_e, was minimized because the ceiling and sidewalls were not heated. The displacement thickness will be discussed further in the Integral Properties section of this chapter.

Surface Temperature Distribution and Heat Transfer

The surface temperatures were measured by the 21 embedded thermocouples in the nozzle block and by the total temperature probe discussed in Chapter 5 (Experimental Setup). All profile data were reduced using a time-averaged probe measurement for the local wall temperature, while the numerical computations utilized the temperature distribution recorded by the embedded thermocouples. This section presents and compares temperatures recorded by each method along with a discussion of accuracies and heat transfer effects.

Figure 26 shows typical transient measurements of T_w for several thermocouples along the nozzle floor centerline. As shown in Figure 26, steady conditions over the entire streamwise range of measurement occurred after about 90 minutes. All profiles were recorded at steady-state conditions which could be maintained within $\pm 2°C$, for an indefinite amount of time. Some runs were as long as 12 continuous hours. Surface temperatures (reported by embedded thermocouples) were repeatable between each tunnel run to $\pm 2°C$ at all measurement locations.

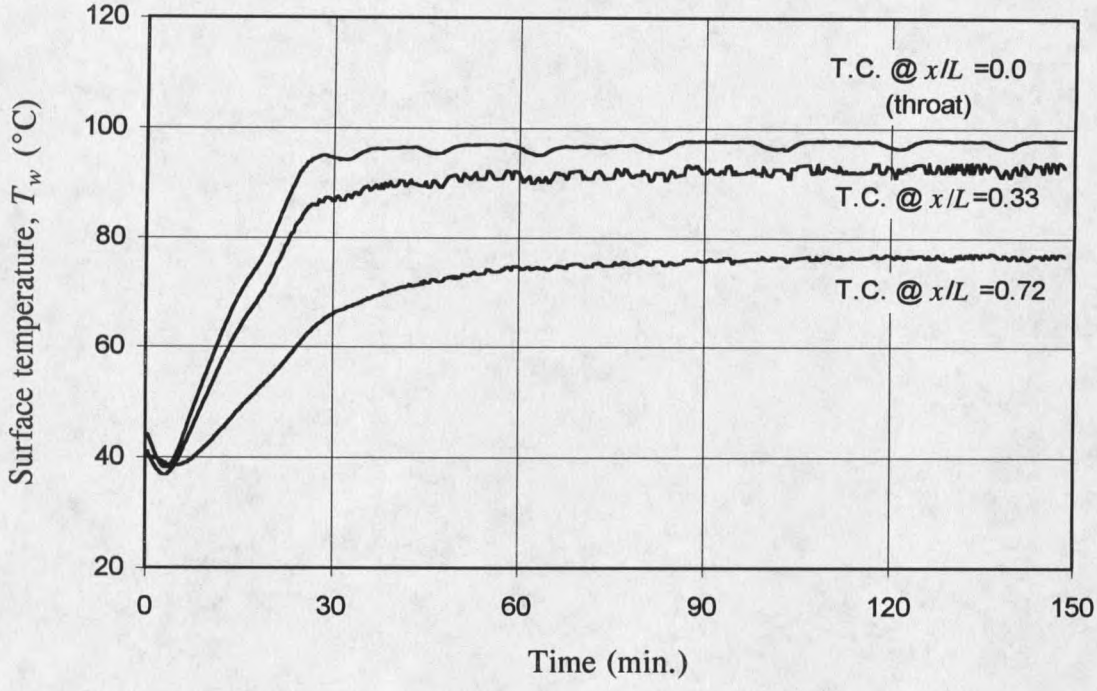


Figure 26. Example of transient surface temperature measurements. (case 2c/2d shown: P_o = 400 mmHg, T_o = 66 °C, $(T_w)_{x=0}$ = 97 °C).

Several thermocouples showed scatter in their output signals, as in the x/L=0.33 curve, and is suspected to be a result of circuit noise. No explanation can be given as to why some thermocouple signals showed more noise than others. Comparison with nearby embedded thermocouples showed that the mean values of those signals with scatter were reasonable. The periodic oscillation of $(T_w)_{x=0}$ shown in Figure 26 was characteristic of all T_w upstream of x/L = 0.2. These oscillations in surface temperature were a result of corresponding variations in the circulating fluid temperature. Steady-state conditions were still maintained in the boundary layer since each surface temperature oscillation occurred over a time span of about 5 minutes. It is

interesting to note that these surface temperature oscillations were not conducted or convected to locations downstream of $x/L = 0.33$.

Figures 27 - 29 show the time-averaged adiabatic wall temperatures for all unheated cases. The embedded thermocouple temperatures were averaged over 60 minutes to include the effect of the long wavelength oscillations in the mean surface temperature values. Theoretical values were calculated from the 2-D boundary layer code of Harris and Blanchard [25] assuming an adiabatic wall condition. The graphs show that experimental values agree well with each other, but that the experiment does not agree with the theoretical predictions.

The T_{aw} computed by the theory yielded recovery factors of 0.83 ± 0.01 for all locations and conditions computed. This result is supported by the classical theory for non-unity Pr flows which states the recovery factor is approximately $\sqrt{Pr}$ [8]. For a constant $Pr = 0.7$, the recovery factor is 0.836. Recall that all numerical computations were performed using a variable Pr.

The streamwise uniformity of the measured T_{aw} suggests that the conduction of the aluminum nozzle block played a significant role. And, if conduction in the streamwise direction was a dominant factor, conduction from the ambient conditions outside the test section must also have played a role. This influence of ambient conditions is shown by the relatively high (relative to theory) surface temperatures for $T_o = 21$ °C in Figure 29, and the relatively low surface temperatures for $T_o = 66$ °C of Figure 28. The point should be re-emphasized again that the experimental

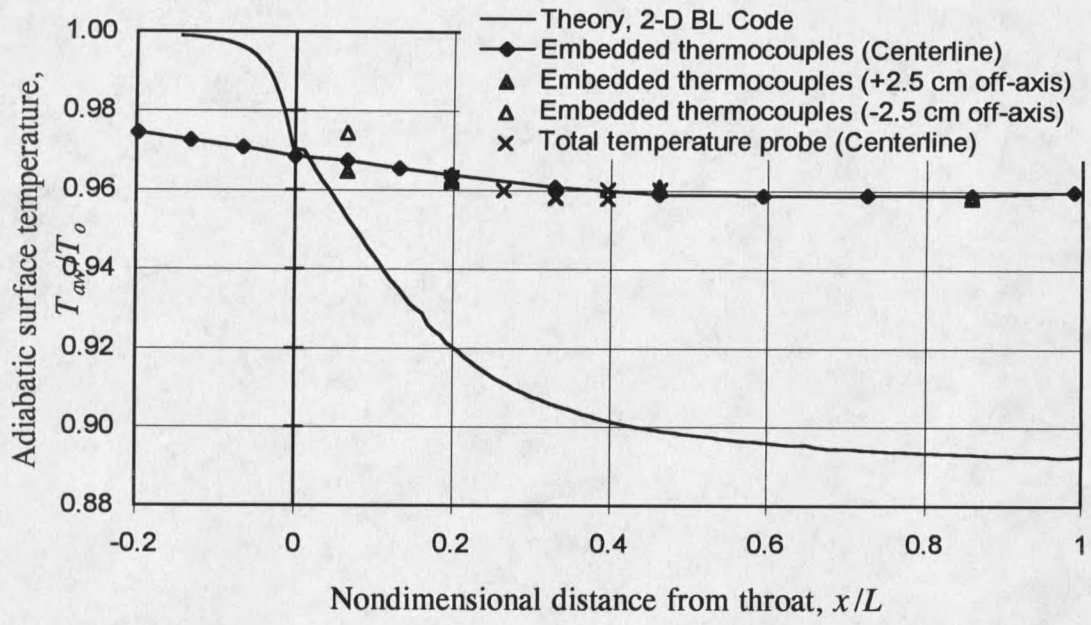


Figure 27. Adiabatic wall temperature, T_{aw}/T_o, case 1a and 1b, (P_o = 400 mmHg, T_o = 32 °C).

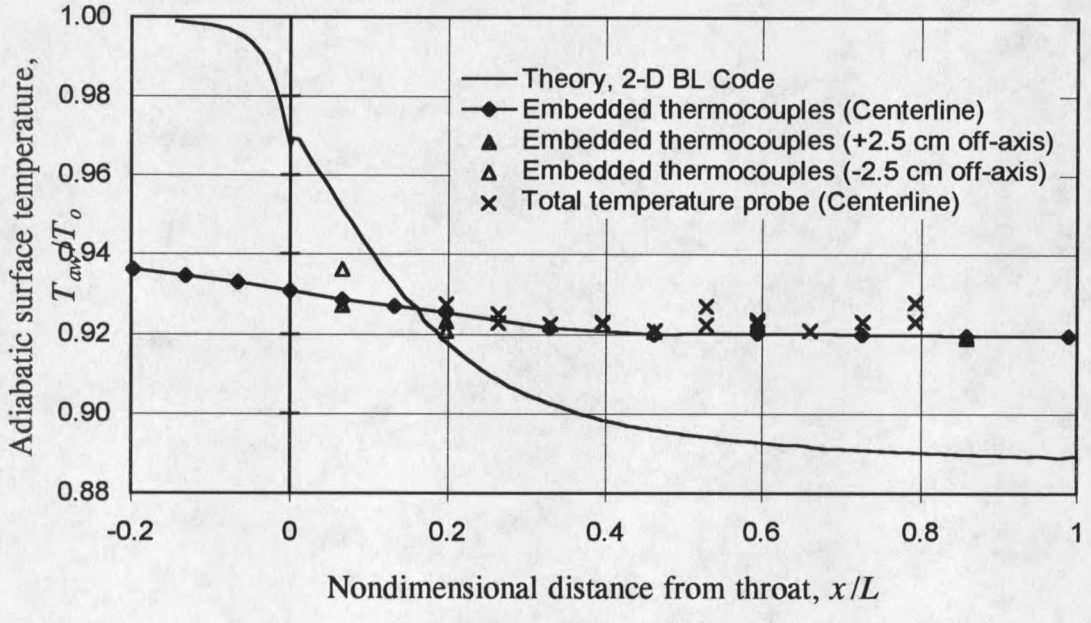


Figure 28. Adiabatic wall temperature, T_{aw}/T_o, case 2a and 2b, (P_o = 400 mmHg, T_o = 66 °C).

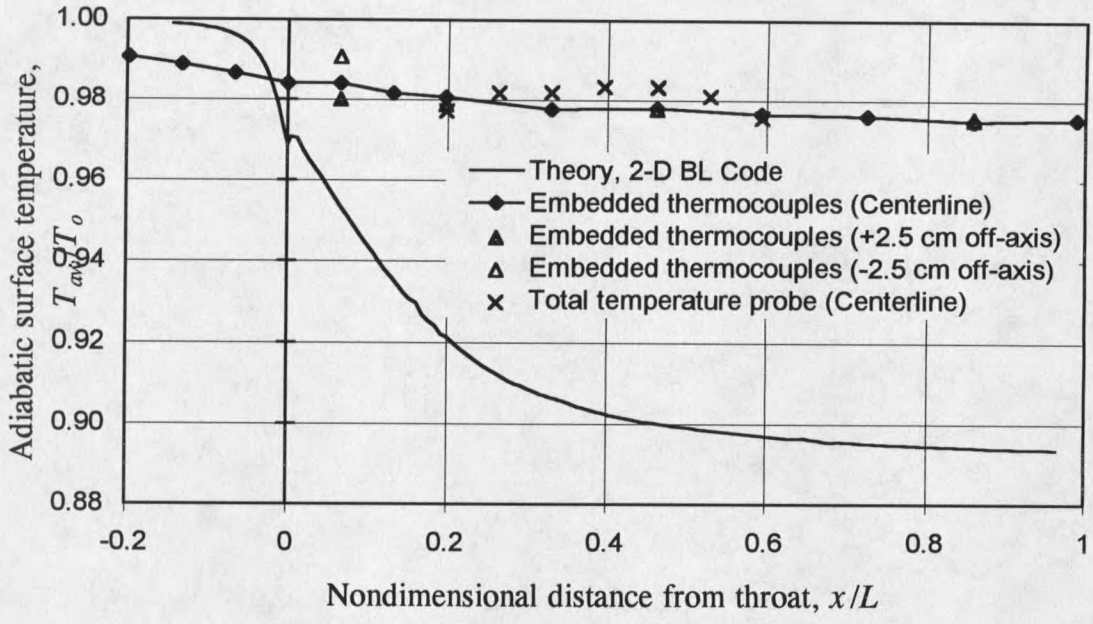


Figure 29. Adiabatic wall temperature, T_{aw}/T_o, case 3a, (P_o = 500 mmHg, T_o = 21 °C)

conditions noted as "adiabatic" were not truly adiabatic, but could only represent the cases with no heat transfer from the circulating fluid. However, it has been assumed that the case of no circulating heat is approximately adiabatic.

Within the measured T_{aw}, there was excellent lateral uniformity as noted by the off-axis embedded thermocouples. Stations x/L = 0.47 and 0.86 of Figures 27 - 29 show the off-axis temperatures were virtually the same as that of the centerline. The T_{aw} of x/L = 0.660 at +2.5 cm and -2.5 cm off axis was consistently lower and higher, respectively, than the centerline T_{aw}. There is currently no explanation for this observation.

Figures 30 and 31 show the variation of T_w/T_{aw} for the heated cases 2c, 2d and 3b, where T_{aw} is the adiabatic wall temperature from the time-averaged embedded thermocouple measurements reported in Figures 28 and 29. As with the T_{aw} measurements, T_w values from the embedded thermocouples were averaged over 60 minutes during steady-state conditions. Numerical calculations for the heated wall cases used the embedded thermocouple T_w measurements reported in Figures 30 and 31 as input to characterize the surface temperature.

The embedded thermocouples showed excellent lateral uniformity past $x/L = 0.2$. Due to a malfunction in the thermocouple, the centerline temperature at $x/L = 0.86$ of cases 2c and 2d was 1% lower than the two lateral T_w/T_{aw} measurements. The transient record of T_w at x/L = 0.86 showed an abrupt (within 10 seconds) drop of 3.5 °C at time = 30 minutes. Prior to this, the deviation between the centerline and lateral T_w values was less than ± 0.3 °C for all times. The small (1%) non-uniformities at the throat were most likely due to the path of the circulating fluid through the nozzle ducting. Apparently, these asymmetries were not convected or conducted downstream.

Values of T_w measured by the overhead total temperature probe were repeatable within 0.3%. In fact, there are actually two sets of probe data shown in Figure 30 through the range of $0.2 > x/L < 0.86$. The deviation increased between the temperatures of the embedded thermocouples and probe as one approached the throat to

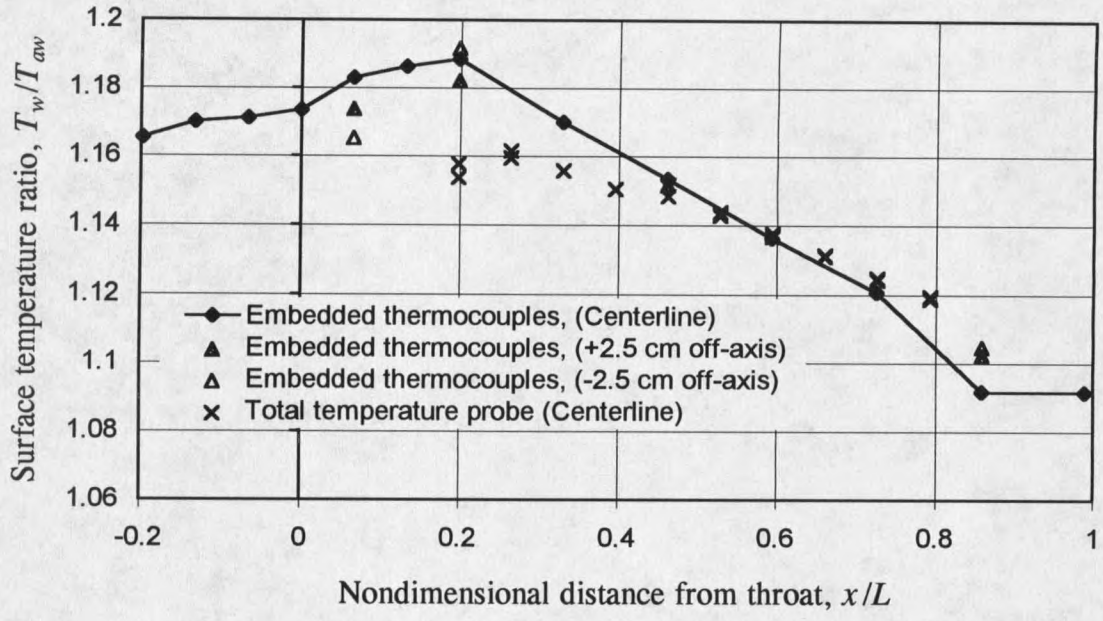


Figure 30. Temperature distribution, T_w/T_{aw}, for heated wall condition, case 2c and 2d, (P_o = 400 mmHg, T_o = 66 °C, $(T_w)_{x=0}$ = 97 °C).

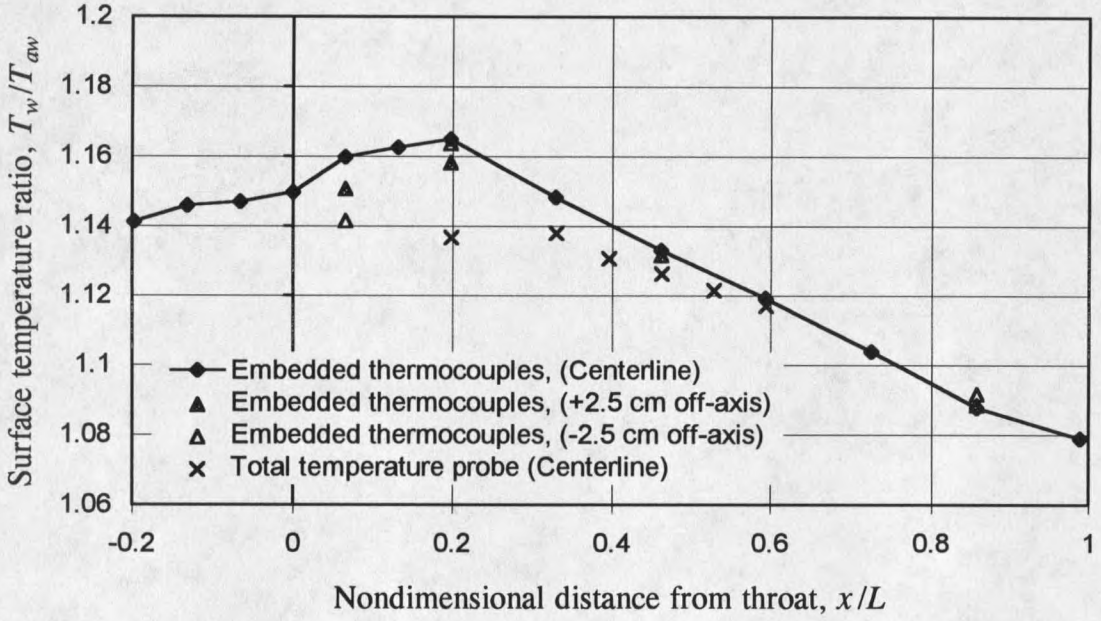


Figure 31. Temperature distribution, T_w/T_{aw} for heated wall condition case 3b, (P_o = 500 mmHg, T_o = 21 °C, $(T_w)_{x=0}$ = 60 °C).

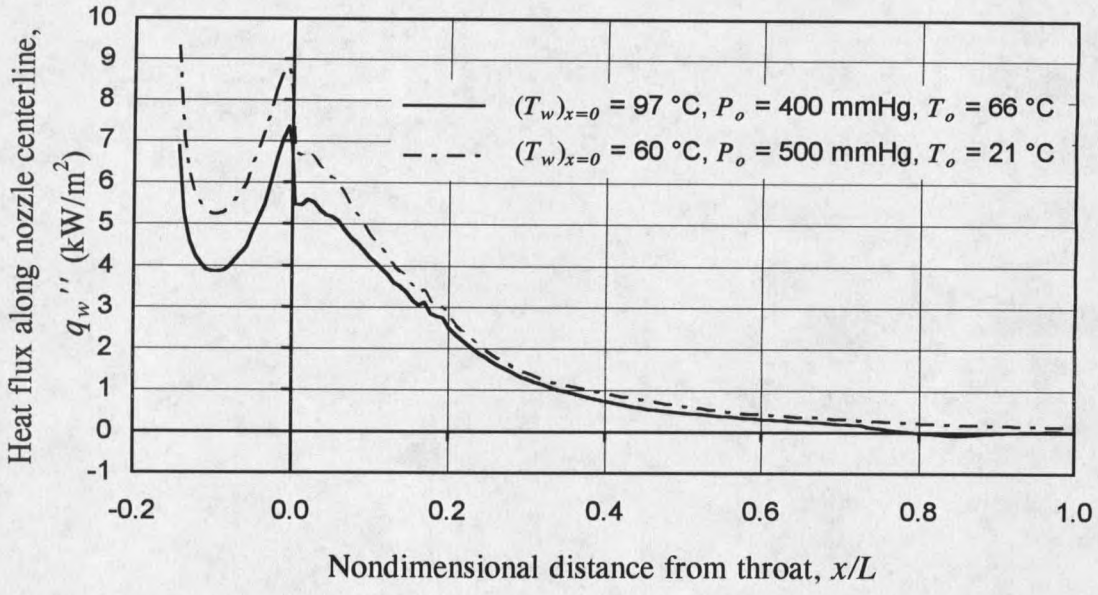


Figure 32. Theoretical heat flux distribution, q_w'' for cases 2c/2d (P_o = 400 mmHg, T_o = 66 °C, $(T_w)_{x=0}$ = 97 °C) and 3b (P_o = 500 mmHg, T_o = 21 °C, $(T_w)_{x=0}$ = 60 °C).

a maximum of 3%. This deviation was primarily due to the decreased resolution of the probe and the larger temperature gradients near the throat. Although the probe was touching the wall surface, the upper surface area of the probe was exposed to the flow. Considering the versatility of an overhead probe (provides the capability of measuring T_w at an infinite number of locations!), the 3% deviation observed between embedded thermocouple and overhead thermocouple readings was very acceptable.

Figure 32 shows the heat flux q_w'' computed by the 2-D boundary layer code as a function of streamwise position. This shows the paramount result that, according to the numerical calculations, the boundary layer was heated over the entire length of the

nozzle for the measured temperature distributions. Many previous arguments of boundary layer stabilization by surface heating have been based on a local "cooling" in the region downstream of local heating [7, 15, 16, 18]. Reliable measurements of $(\partial T/\partial y)_w$ and thus q_w'' could not be obtained from static temperature profiles due to the common problems associated with near-wall measurements. However, the profiles qualitatively suggest that heat was directed into the fluid (i.e. $(\partial T/\partial y)_w < 0$) for all streamwise locations investigated.

Measurements conducted in the MSU/SWT by this author have also shown that the boundary layer becomes more stable for case 3b (P_o = 500 mmHg, T_o = 21 °C, $(T_w)_{x=0}$ = 60 °C). Instabilities were not measured in cases 2c and 2d (P_o = 400 mmHg, T_o = 66 °C, $(T_w)_{x=0}$ = 97 °C) since the layer was laminar quiescent flow both with and without heating; that is, there were no instabilities to measure. Furthermore, calculations for cases 2c/2d and 3b showed that the boundary layer becomes more stable with respect to a Görtler instability when compared to the respective adiabatic cases 2a/2b and 3a. This instability arises from tube-like vortices in the boundary layer whose axis of rotation is aligned parallel to the flow. The streamwise vortices are generated by centrifugal forces as the fluid flows over a curved surface. Görtler vortices have been shown to be the cause of transition on the concave surface of a supersonic nozzle [38].

As with the flat plate studies, previous research has shown that Görtler vortices on concave walls can be stabilized with localized wall cooling or heating (as discussed

above, these are one in the same) for low-speed flows [15]. However, those same calculations which showed a positive heat flux over the entire nozzle length also predicted a decrease in the maximum Görtler number based on θ[2]. Furthermore, mean flow results from the boundary layer code were used as input to the e^{Malik} stability code which predicted a decrease in the *N*-factor with surface heating.

The heat flux values of Figure 32 also give an indication of the embedded thermocouple accuracy. It must be noted that the embedded thermocouples did not measure the "surface" temperature, but the temperature just below the surface. Based on the thermal conductivity of the 0.013 cm stainless steel overlay (14.9 W/m°C) and that of the 0.005 cm PSA (0.14 W/m°C) [30], the temperature drop through the instrumented overlay in the throat region was 1.5 °C (0.4% of T_w). No attempt was made to correct for the error in the embedded thermocouple reading.

Boundary Layer Profiles

The profile provides a visual comparison between experiment and theory. As discussed earlier, the shape of the hydrodynamic boundary layer strongly influences the stability and receptivity of a laminar boundary layer. The entire set of data for the present research is presented in Appendix B with normalized velocity and temperature profiles. Select curves and sets of curves have been extracted from the full set of

[2] Görtler number based on θ: $G_\theta = Re_\theta(\theta/r_c)^{1/2}$, where r_c is the local radius of curvature

profile data to illustrate main trends. Velocity and temperature profiles, as opposed to other quantities, provide a basis of common physical properties from which to compare experimental results and theoretical predictions.

Figure 33 shows the boundary layer progression for case 1a and 1b (P_o = 400 mmHg, T_o = 32 °C). This progression shows that the theory compares well with the experimental measurements for all profiles. Also, note that the presence of the screen produced little to no effect on the adiabatic mean boundary layer. This result is significant in that it confirms that transition delay due to the screen (noted by Mueller [20]) must come from a change in the free-stream quality as originally expected by Morkovin [1].

Figure 34 shows the typical progression of the boundary layer with and without surface heating. In light of the above observation that the screen did not affect the mean boundary layer, comparisons for heated and unheated profiles cross between the data obtained with the screen and the data obtained with no screen. Some profiles appear to have been less repeatable than others (screen vs. no screen data based on above conclusion), however, differences can often be traced back to slight scatter in the boundary layer thickness, δ. Because the ordinate is y/δ, differences in δ have the effect of vertically scaling the profile up or down. Thus, differences between experimental profiles at the same streamwise location and stagnation conditions must be evaluated in light of variations in the boundary layer thicknesses. In general, the repeatability for thickness measurements was good. The vertical distance from the

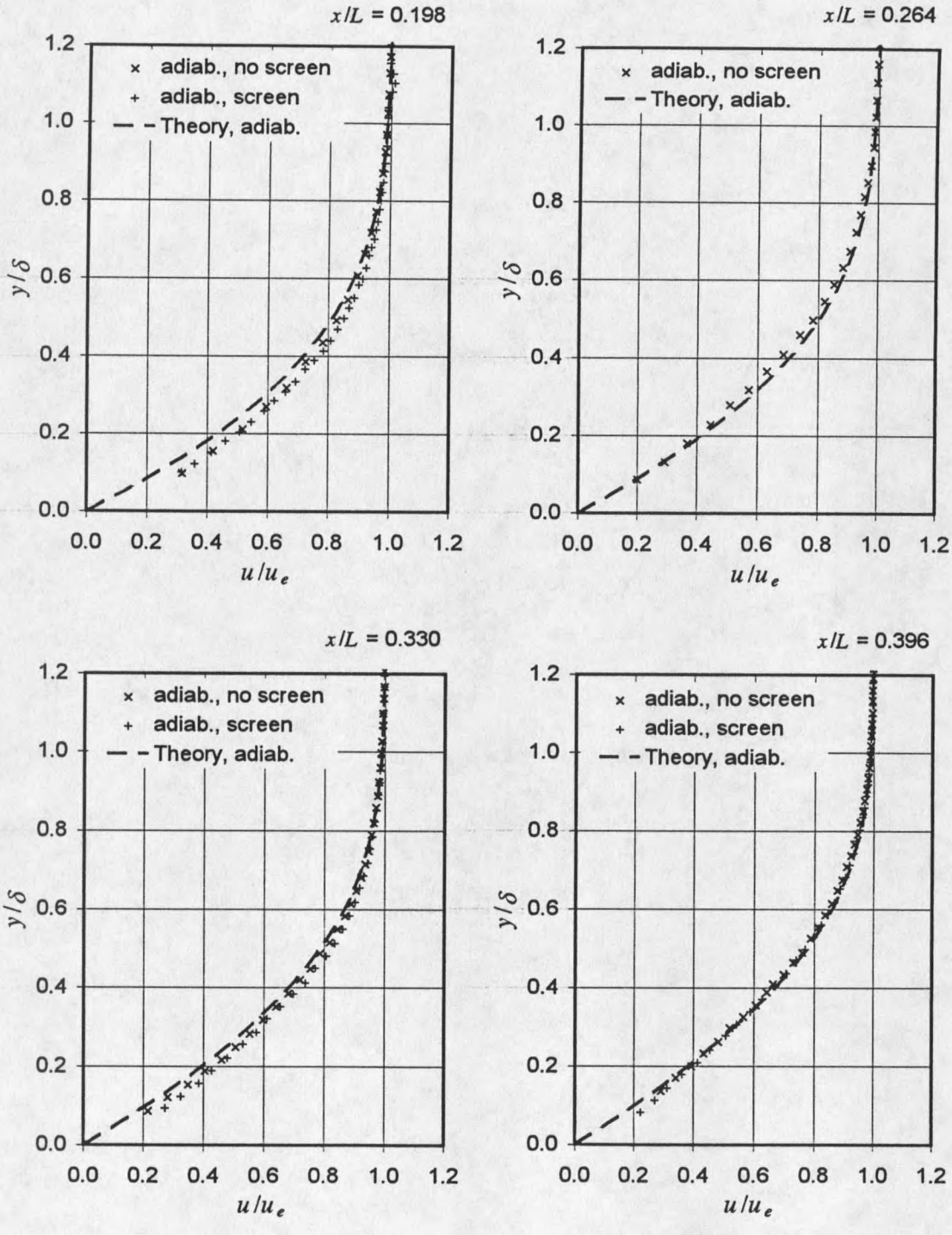


Figure 33. Effect of screens on mean-flow: Boundary layer velocity progression for case 1a and 1b. (P_o = 400 mmHg,, T_o = 32 °C, adiabatic wall)

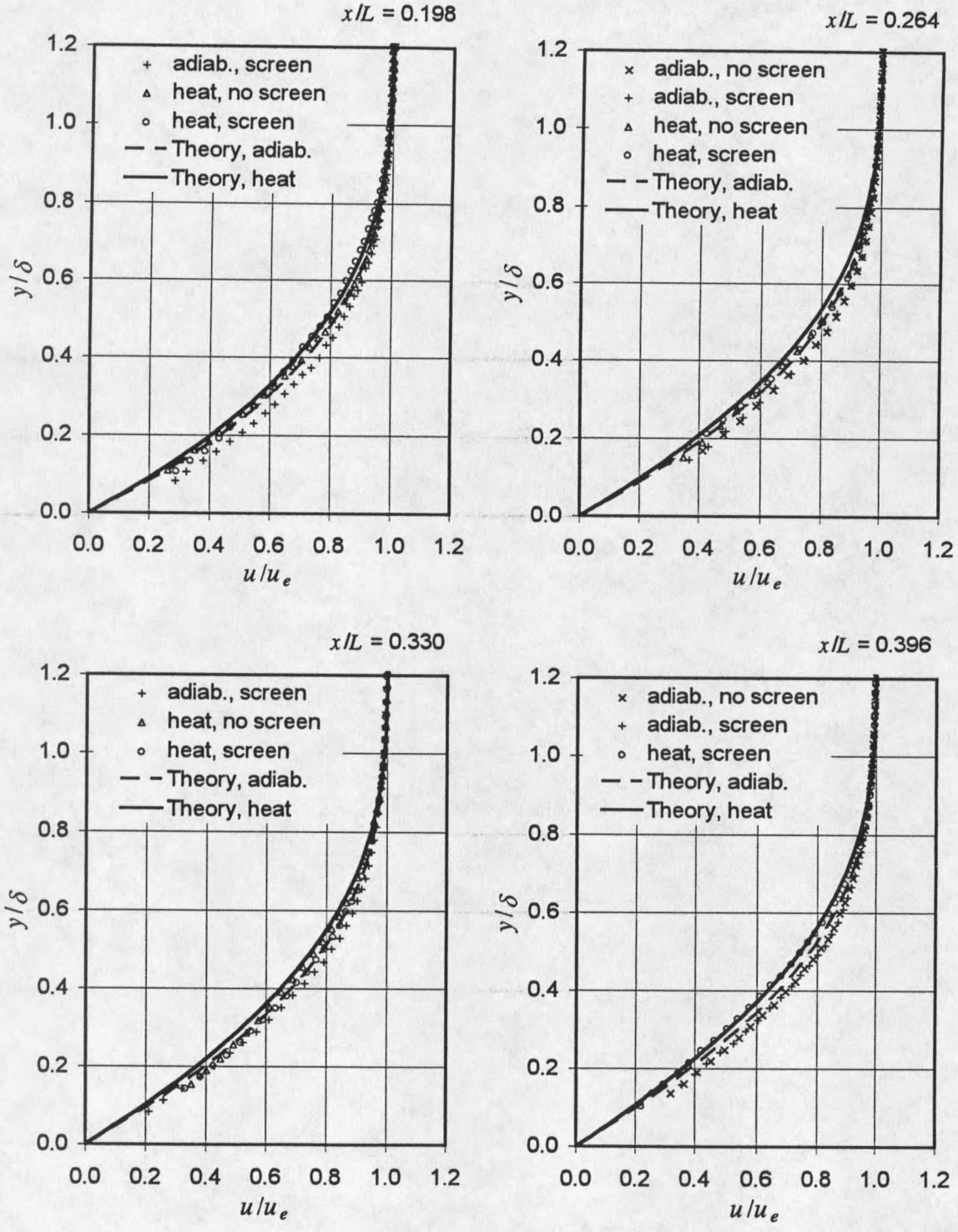


Figure 34. Effect of heat on mean-flow. Boundary layer velocity progression for case 2a - 2d. (P_o = 400 mmHg, T_o = 66 °C, heated condition is $(T_w)_{x=0}$ = 97 °C)

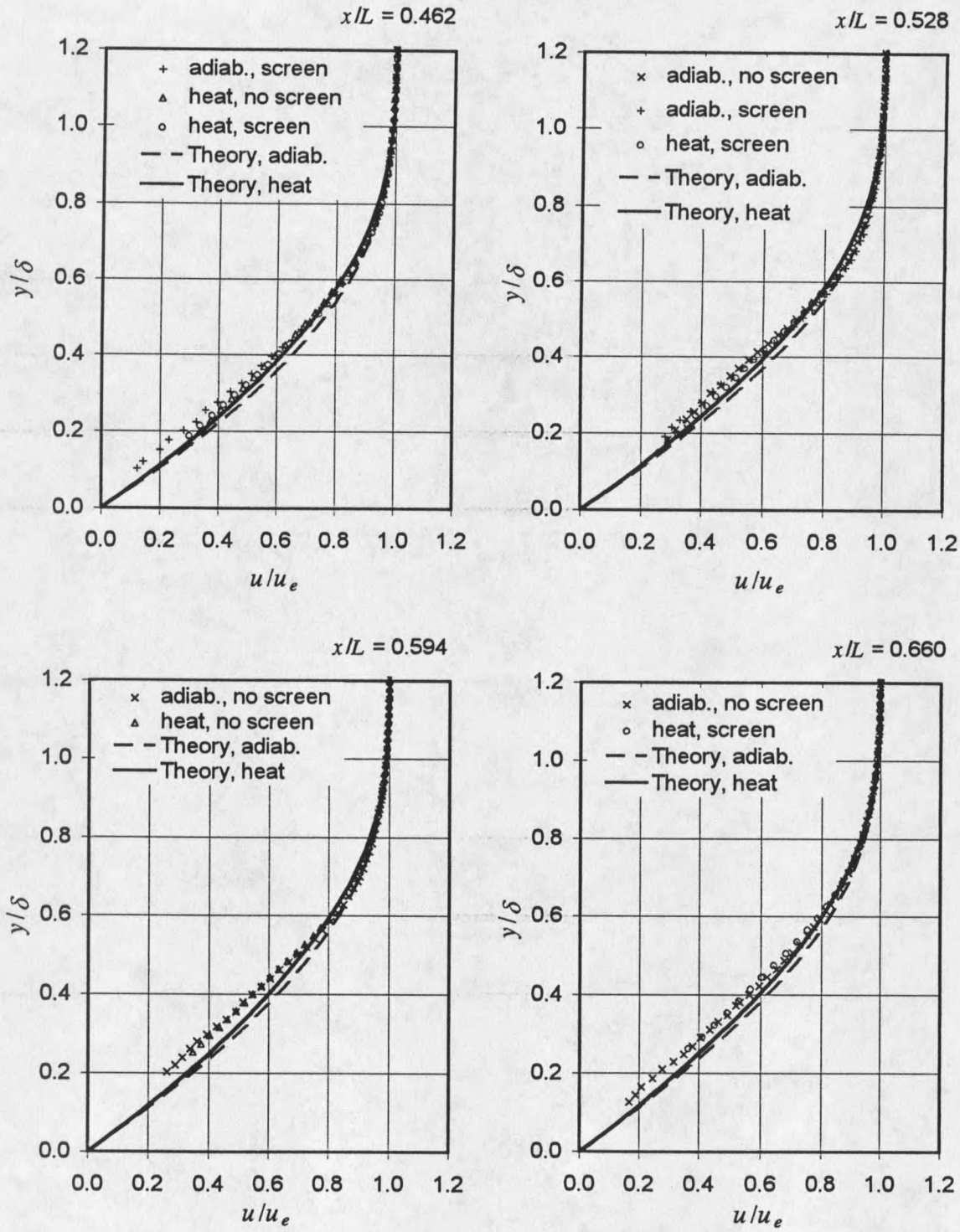


Figure 34. (continued)

wall, y, was scaled with each profile's respective δ because the theoretical and experimental velocity profiles appear to consistently differ by the scaling factor $\delta_{experiment}/\delta_{theory}$. That is, the velocity, u_1, at a given distance from the wall, say y_1, corresponds to u_1 in the velocity profile computed numerically at a distance $y_2 = y_1 \circ (\delta_{exp}/\delta_{theory})$ in the theoretical profile.

Both experiment and theory show that the effect of heating on the velocity boundary layer was small, with the largest differences occurring in the range $0.3 < y/d < 0.7$. The heated versus unheated profile comparisons support the classical theory that heating causes a less-full profile. The progression of profiles also shows that the effect of heating was slightly diminished the further one travels downstream. However, it should be noted here that caution must be used in emphasizing small differences in the experimental profiles since these small differences can be distorted by experimental error.

Velocity profiles for cases 3a and 3b (P_o = 500 mmHg, T_o = 21 °C) do not show as favorable comparison between theory and experiment as that of cases 2a - 2d (P_o = 400 mmHg, T_o = 66 °C). The heated profiles obtained by experiment showed a deviation from theory below about $y/\delta = 0.5$. Previous measurements in the MSU-SWT have also shown that the boundary layer for P_o = 500 mmHg is no longer a quiescent, laminar layer [20]. And, if the boundary layer of cases 3a and 3b was in a transitional state, it cannot be adequately compared with laminar theory. Measurements for cases 3a and 3b could not be repeated, so no conclusions could be

drawn as to whether this discrepancy is actually present or whether experimental error has contaminated the results.

Profiles of the thermal boundary layer were plotted as $(T - T_e)/(T_w - T_e)$ vs. y/δ. In all heated wall cases, the wall temperature, T_w, was obtained from the embedded thermocouple immediately below the probes when the profile location coincided with a thermocouple location. T_w was determined by linear interpolation between the two thermocouples surrounding (one upstream, one downstream) the profile station. The value for T_w in the adiabatic profiles was determined by the embedded thermocouple(s) for experimental data, and by calculation assuming $q_w'' = 0$ for theoretical profiles.

Figure 35 shows the progression of the thermal boundary layer for case 2 (P_o = 400 mmHg, T_o = 66 °C). Qualitatively, both experimental and theoretical profiles show the same trend. The normalized static temperature through the boundary layer is less for the adiabatic case, as expected. Also, the normalized temperature difference between the heated and unheated profiles is quantitatively the same in theory and experiment. However, the value of $(T - T_e)/(T_w - T_e)$ differs through the boundary layer by about 0.1 (on a 0 to 1 scale) between experiment and theory. The increase of the measured over the theoretical boundary layer thickness acts to scale the experimental profile down relative to the theoretical profile. Thus, the numerical values of $(T - T_e)/(T_w - T_e)$ may show a better numerical agreement with measurements for a given physical distance from the wall, y, than indicated by Figure 35. Experimental measurements show good repeatability in the temperature profiles. As

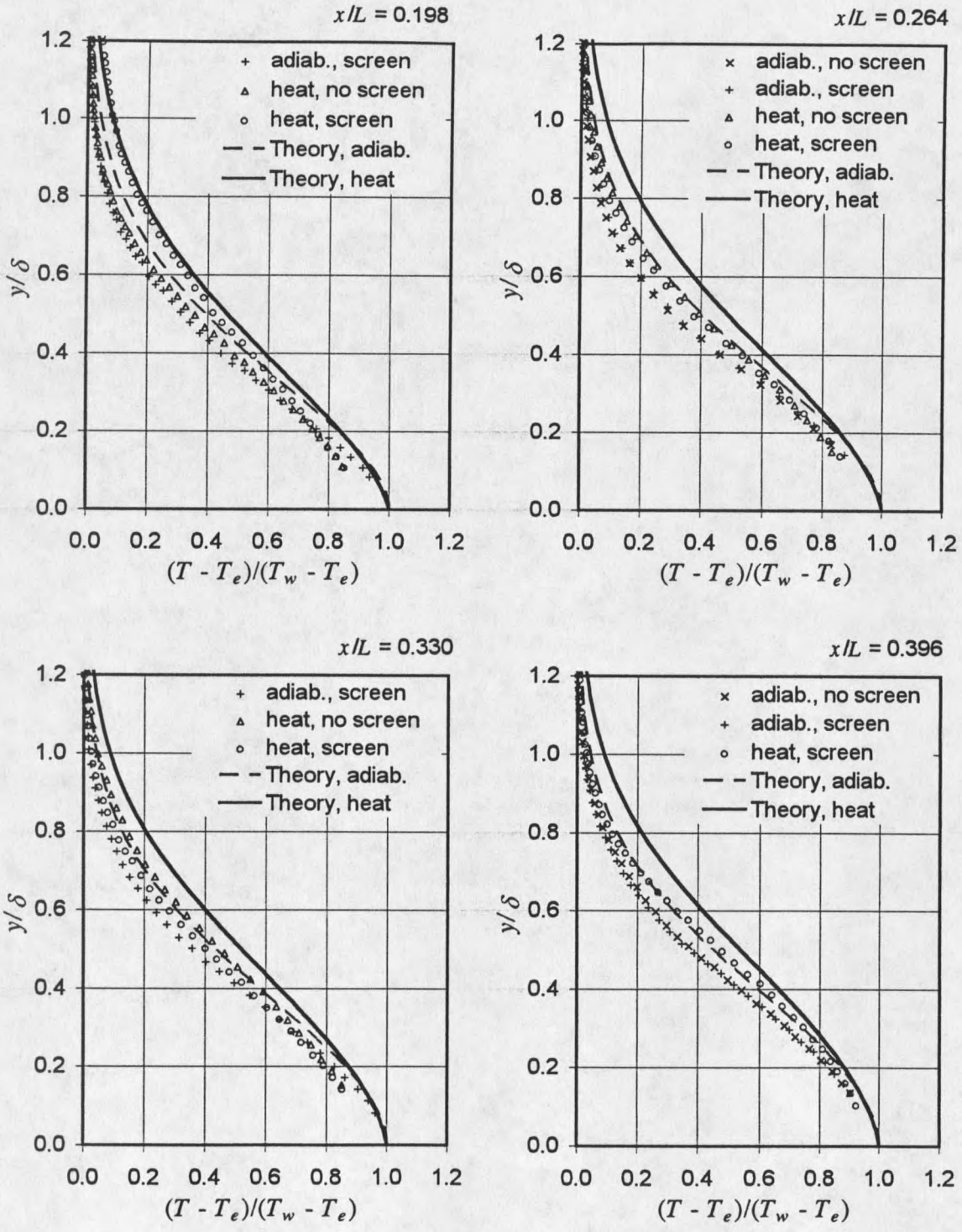


Figure 35. Thermal boundary layer development. Case 2a - 2d. (P_o = 400 mmHg, T_o = 66 °C, heated condition is $(T_w)_{x=0}$ = 97 °C)

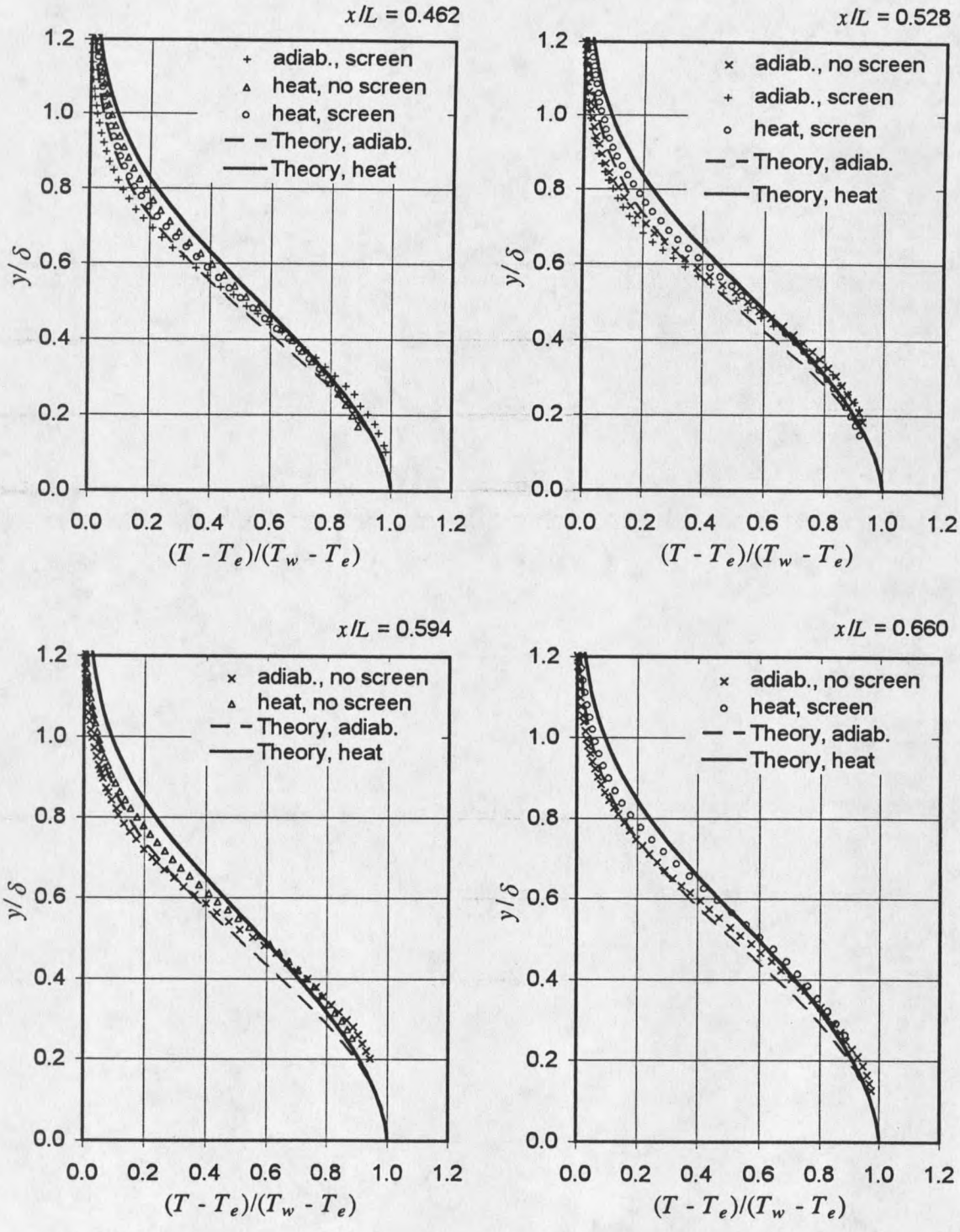


Figure 35. (continued)

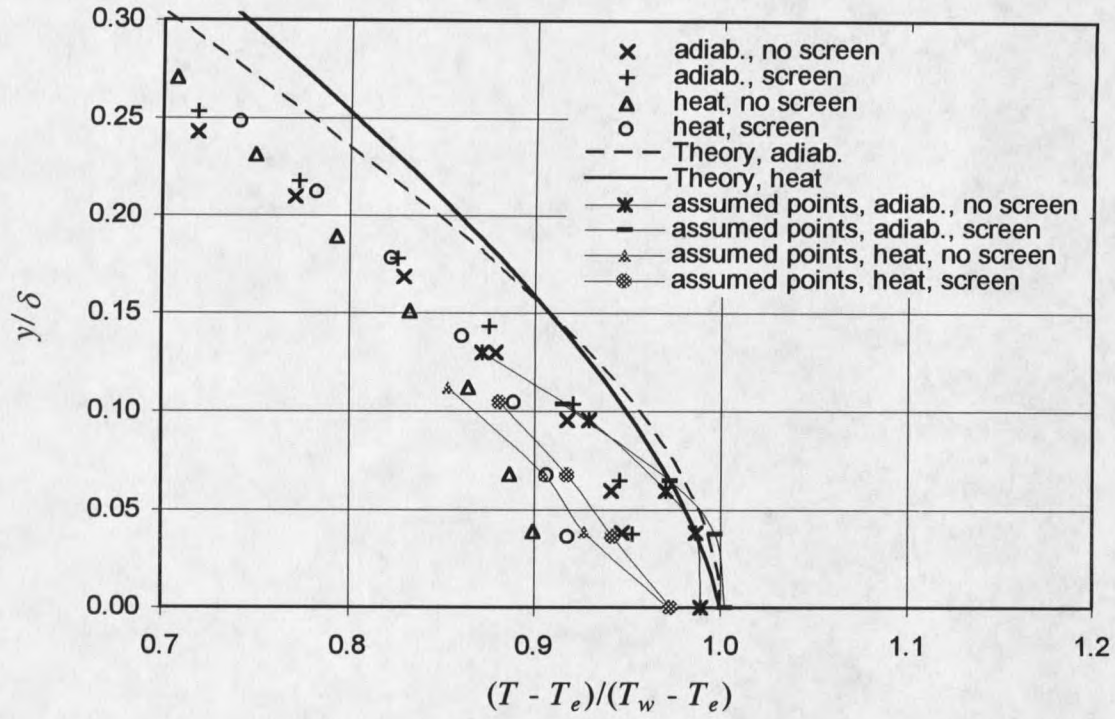


Figure 36. Detail view of boundary layer near the wall.

with the velocity profiles, in cases of apparent disagreement (e.g. $x/L = 0.198$), the discrepancy can often be traced back to a difference in δ between the measurements with and without the screen.

The theoretical profiles at first glance do not clearly show $\partial T/\partial y = 0$ as required by the adiabatic wall condition. A magnified view of the near-wall region (Figure 36) shows the zero temperature slope was in fact satisfied. The theoretical normalized temperature profiles also show very little change in slope at the wall between the unheated and heated profiles. Note that this is due to the fact that the derivative of temperature, $\partial T/\partial y$, has been scaled by the overall temperature difference,

T_w - T_e, which was as much as (case 2) 160% higher at the throat for the heated wall case than that of the adiabatic wall.

Integral Properties

Figures 37 - 39 show the boundary layer growth over the nozzle centerline for each of the cases investigated. The most significant feature is that the experiment and theory agree that heating had very little effect on the boundary layer thickness, δ, defined as the distance from the wall at which $u/u_e = 0.99$. In fact, the theoretical curves show the boundary layer actually thinned slightly with heating. Experimental values of δ also show a negligible change with surface heating; and it could be argued that the experimental results even show a small reduction in the layer thickness with surface heating. This result has a profound impact on the roughness arguments presented by previous researchers [5]. That the boundary layer thickness, δ, does not change or even thins with heat addition is contrary to intuition, and many simpler boundary layer experiments and theories; accordingly, the flow physics are worth comment.

The thinning of δ with heating has been seen before in theoretical investigations by Cohen and Reshotko [39]. Intuition suggests that due to an increase in viscosity, μ, the diffusion of momentum would occupy a larger region (i.e. boundary layer thickness increases). Or, that a decrease in density, ρ, would cause the boundary layer to

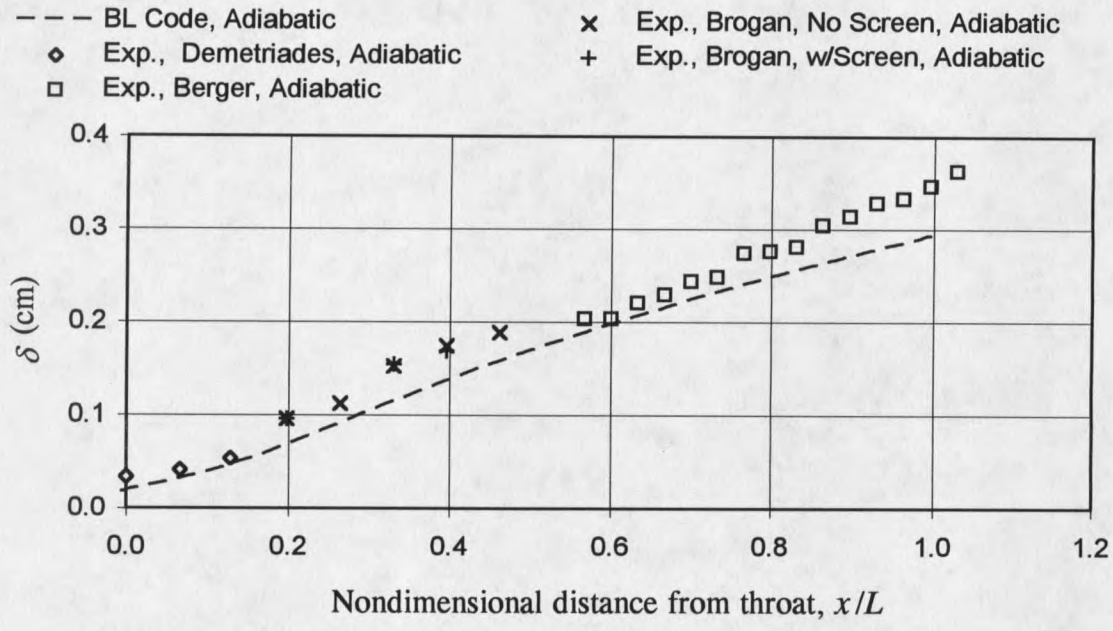


Figure 37. Boundary layer thickness (99% u/u_e), case 1a and 1b, Demetriades [23] and Berger [24] data. (P_o = 400 mmHg, T_o = 32 °C)

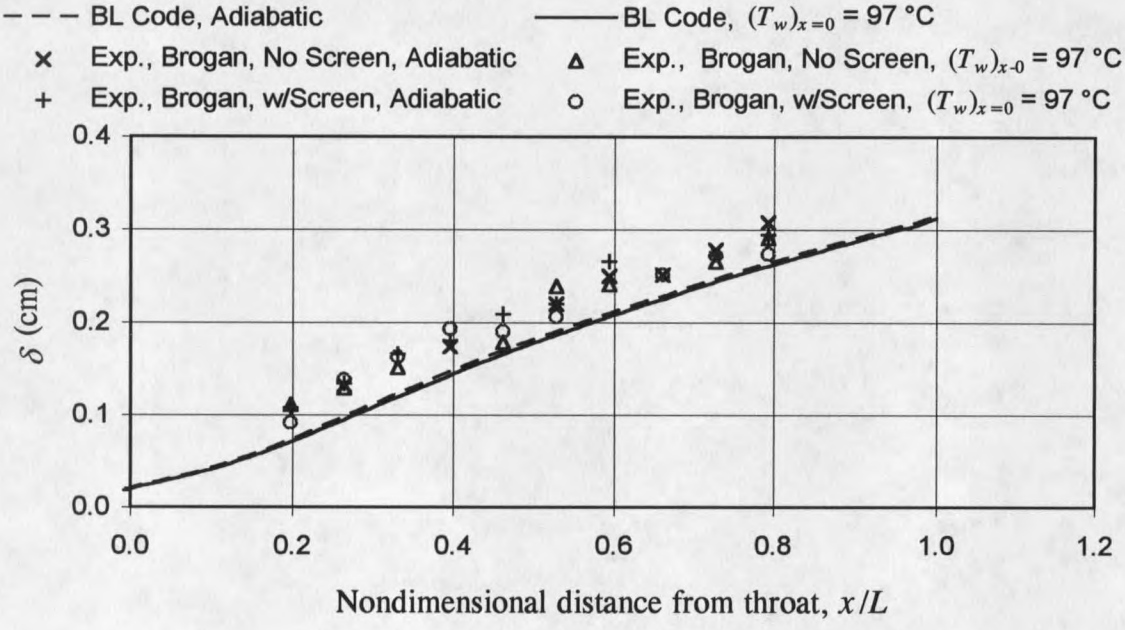


Figure 38. Boundary layer thickness (99% u/u_e), case 2a - 2d. (P_o = 400 mmHg, T_o = 66 °C)

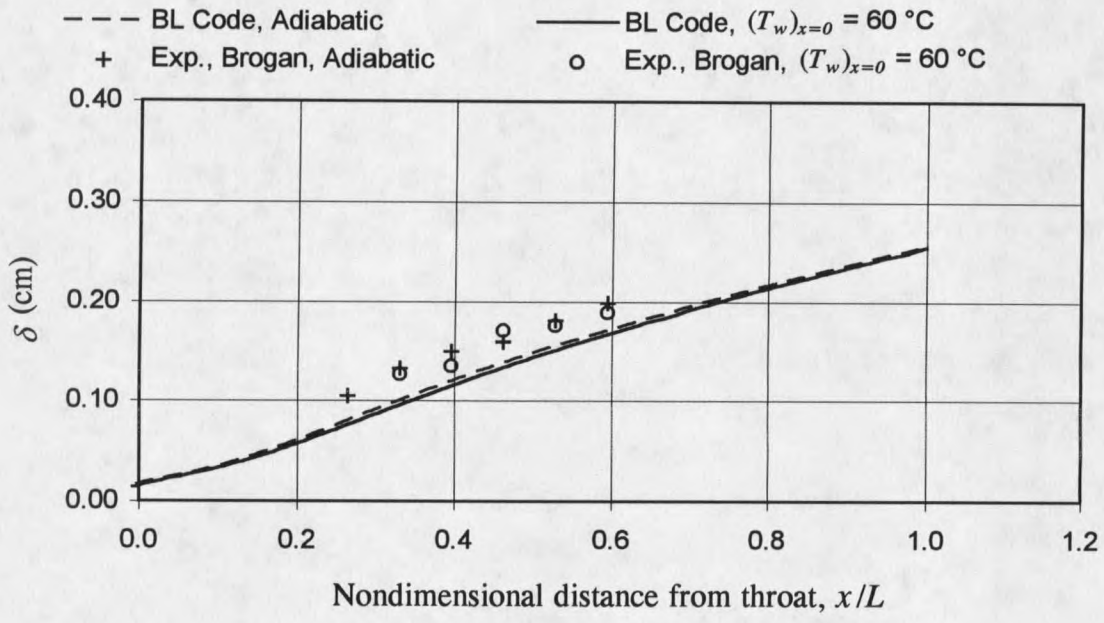


Figure 39. Boundary layer thickness (99% u/u_e), case 3a and 3b. (P_o = 500 mmHg, T_o = 21 °C)

expand. The physical explanation given by Cohen and Reshotko [39] and again by Schlichting [8] is that a favorable pressure gradient acts on a "lighter" fluid to accelerate the velocity in the outer regions of the boundary layer. In spite of the viscous retardation, the pressure gradient imposed by the free-stream produces a greater acceleration on this region of lower density fluid, causing the upper region of the boundary layer to bulge. Theoretical solutions have even shown that the velocity can exceed that of the free-stream with a sufficiently high pressure gradient and surface temperature. A closer look at the boundary layer profiles reveals this phenomenon in both experiment and theory (Figure 40).

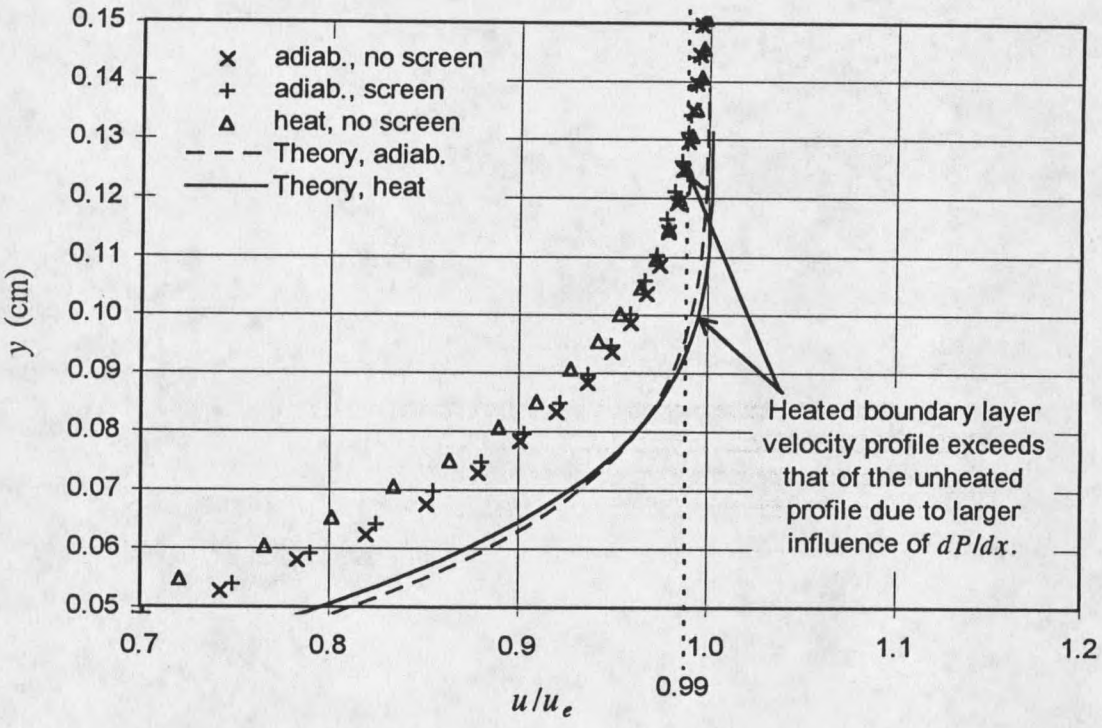


Figure 40. Detail view of the velocity boundary layer edge: effect of heating on boundary layer thickness.

It must also be noted that the total temperature measurement, T_t', was necessary to correctly determine the boundary layer thickness in the presence of surface heating. One could calculate δ directly from a P_t' profile, assuming $T_t = T_o$ at the boundary layer edge, and incorrectly show that the boundary layer substantially thickens (increase of 10% for the current investigations) with surface heating. Since the Prandtl number for air is less than unity, the thermal boundary layer is thicker than the viscous layer, therefore, the assumption of $T_t = T_o$ would not be justified.

Figures 41 - 46 show the displacement and momentum thickness with and without heating. As discussed in Chapter 6 (Data Reduction), data points near the wall

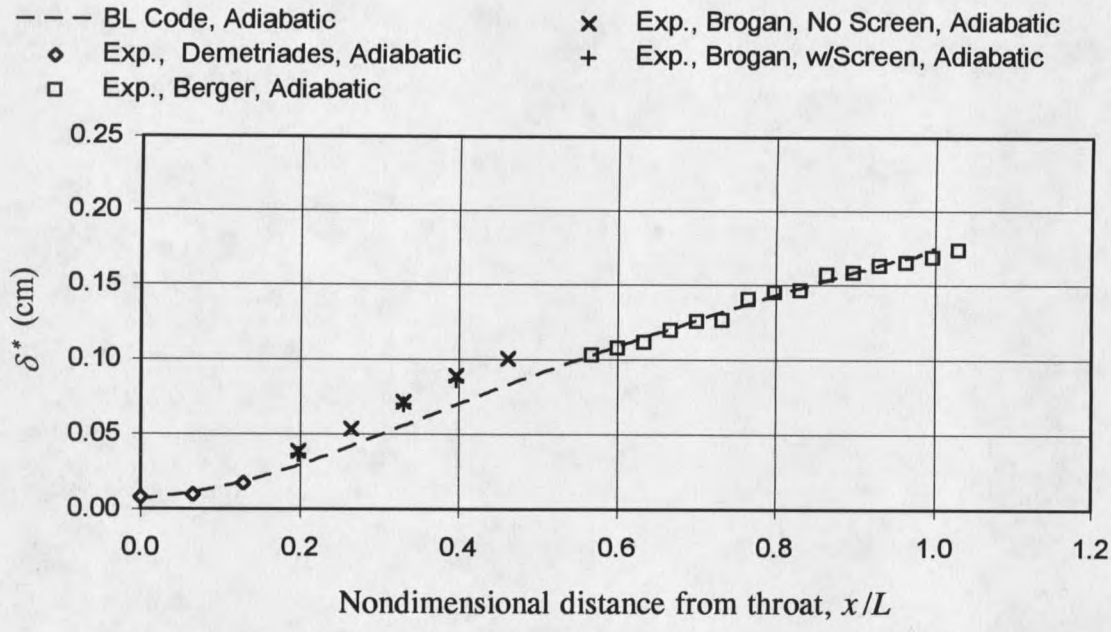


Figure 41. Displacement thickness, case 1a and 1b, Demetriades [23] and Berger [24] data. (P_o = 400 mmHg, T_o = 32 °C, adiabatic wall)

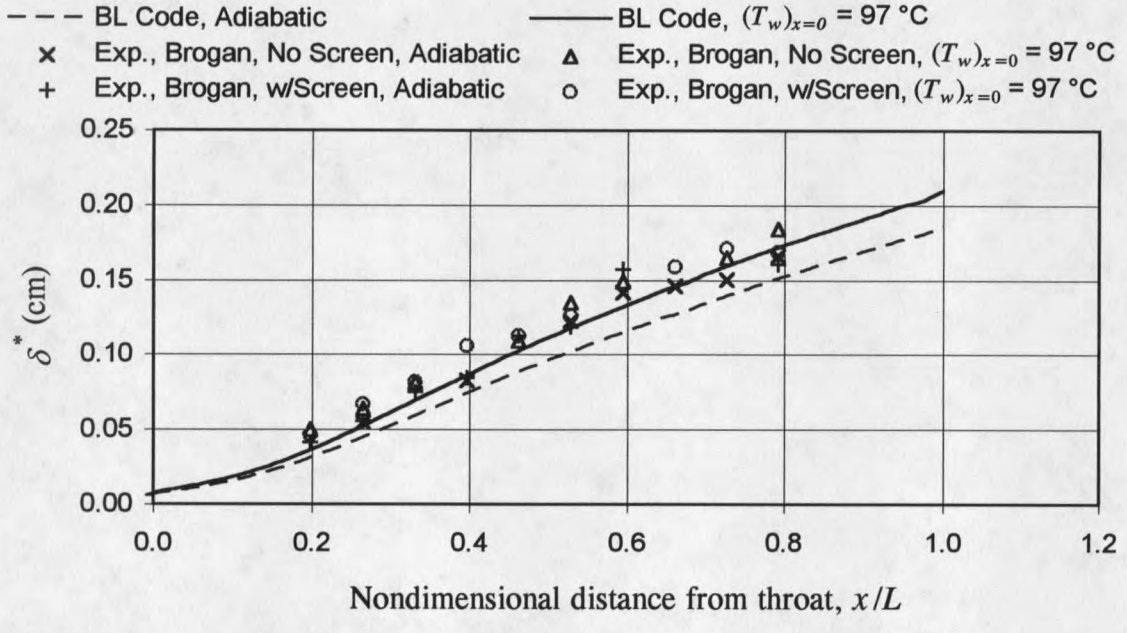


Figure 42. Displacement thickness, case 2a - 2d. (P_o = 400 mmHg, T_o = 66 °C)

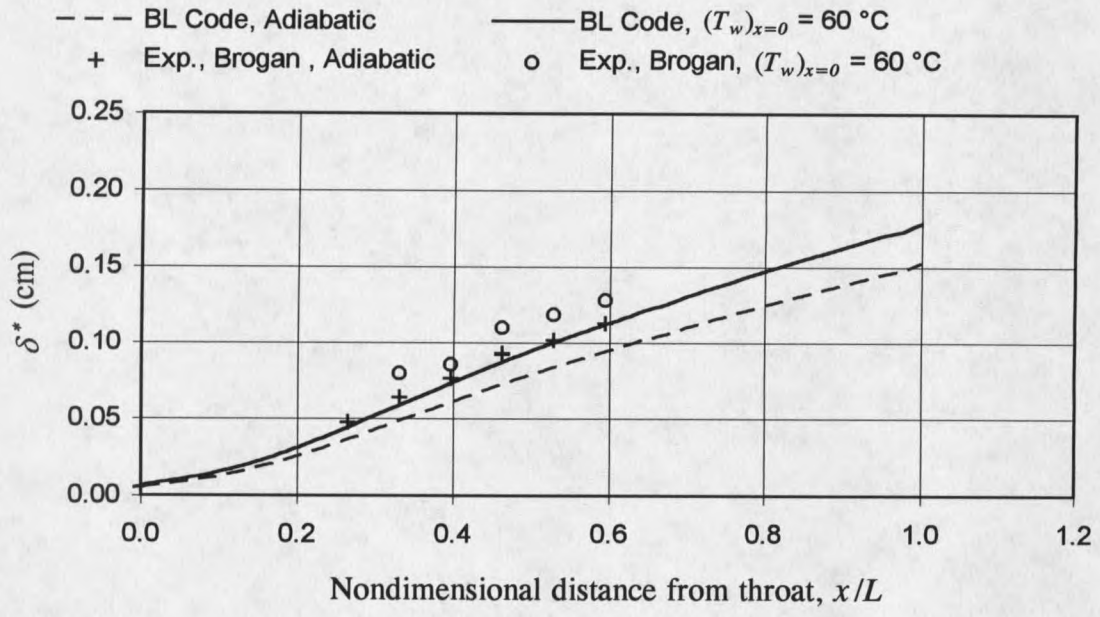


Figure 43. Displacement thickness, case 3a and 3b. (P_o = 500 mmHg, T_o = 21 °C)

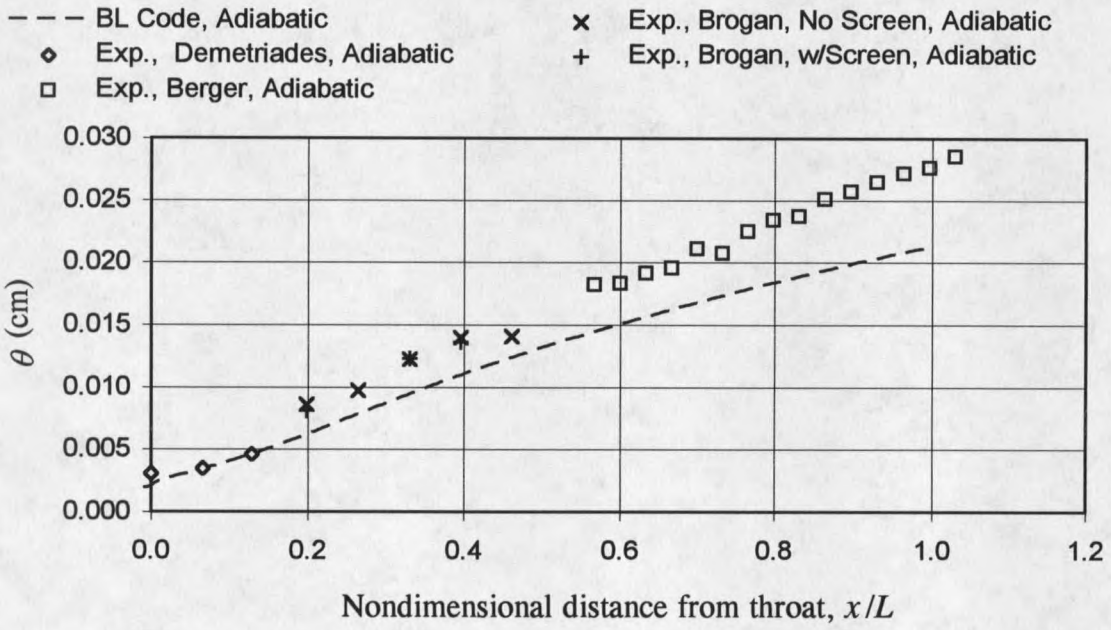


Figure 44. Momentum thickness, case 1a and 1b, Demetriades [23] and Berger [24] data. (P_o = 400 mmHg, T_o = 32 °C, adiabatic wall)

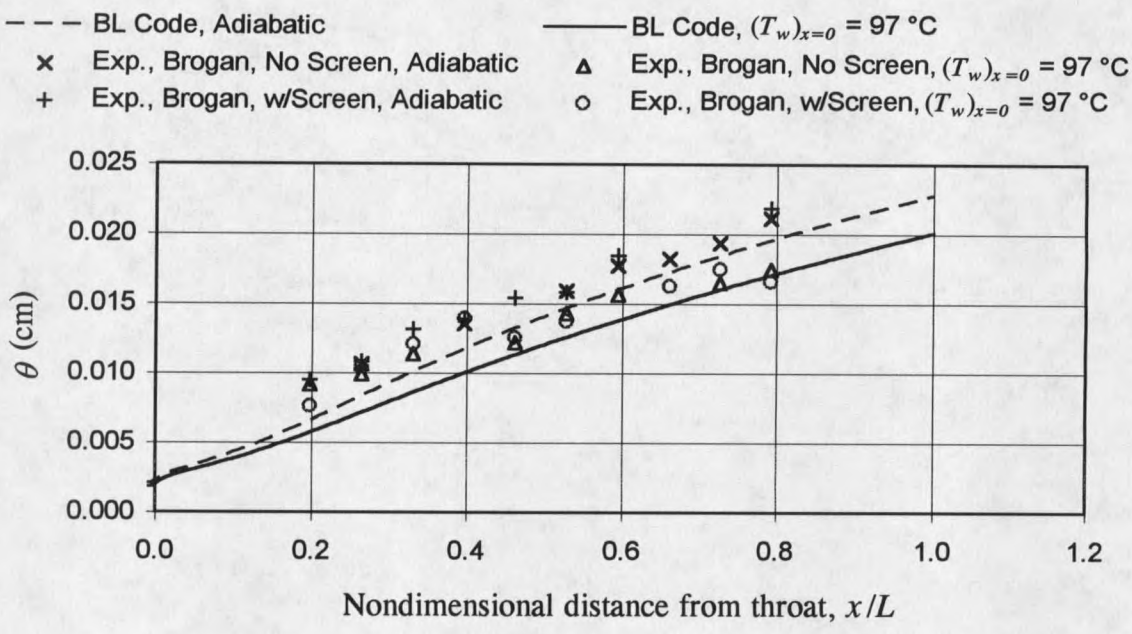


Figure 45. Momentum thickness, case 2a - 2d. (P_o = 400 mmHg, T_o = 66 °C)

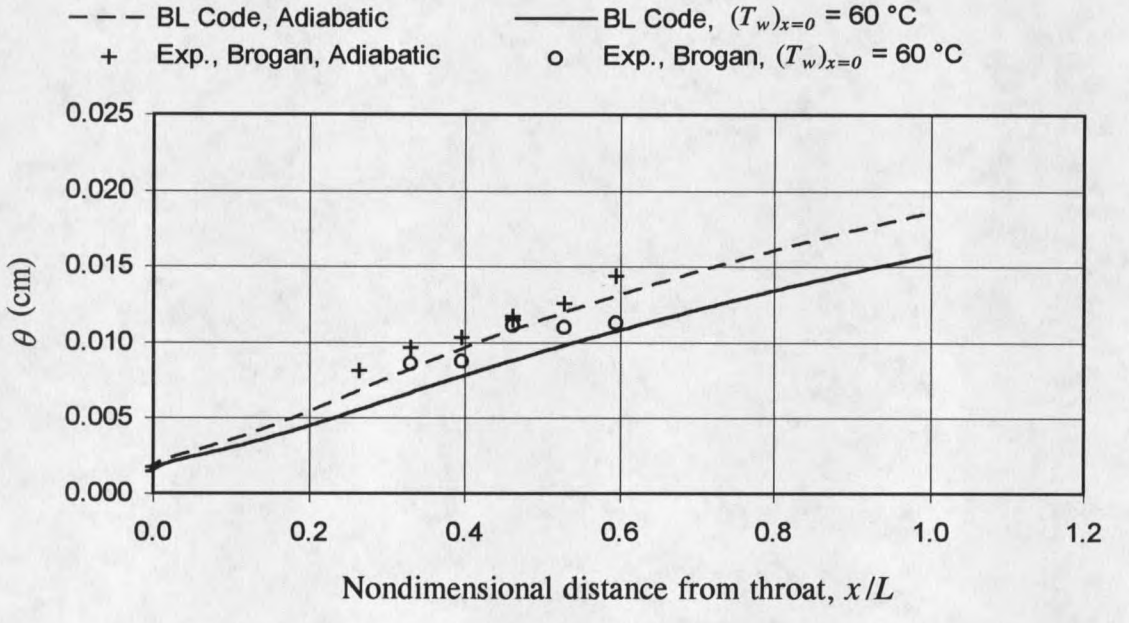


Figure 46. Momentum thickness, case 3a and 3b. (P_o = 500 mmHg, T_o = 21 °C)

were disregarded due to probe-wall interference, and low Reynolds number effects. For the purpose of calculating the integral properties, δ^* and θ, data points near the wall were assumed to follow along a straight line formed by the points $u|_{y=0} = 0$ and $u|_{y(n)} = u(n)$ where n is the first valid data point.

The displacement thickness increased with heating as shown by both experiment and theory. Roughness-related transition arguments of some authors have been related to δ^* instead of δ [4], in which case the relative roughness height could be argued to decrease with heating. However, to this author's knowledge, no conclusive evidence has shown δ^* to be the controlling factor in roughness-related transition.

The momentum thickness, θ, thinned with heating as a result of the decreased fluid density near the wall. Again, theory and experiment agree very well in regards to the change in θ experienced with surface heating.

The integral properties of cases 3a and 3b (P_o = 500 mmHg, T_o = 21 °C) agree better with the comparison trends of the adiabatic boundary layer previously investigated by Demetriades [23] and Berger [24]. The cause for the small differences in layer thickness trends between cases 2 and 3 and the data of Demetriades [23] and Berger [24] is not known.

CHAPTER 8

CONCLUSIONS

The two dimensional boundary layer code gives results that are in general agreement with experimental measurements with and without surface heating. However, some discrepancies were noted. The theory predicted the similar trends as observed by experimental measurement with and without surface heating. Several key effects of surface heating have been identified with the numerical and experimental results. Specifically, the investigation of the effects of surface heating on the supersonic nozzle laminar boundary layer has shown the following.

1. The presence of a 320 mesh screen upstream of the two-dimensional contraction produced no effect on the mean boundary layer. This result is consistent with the extensive data and results of previous research which shows the effect of screens is to reduce convected disturbances in the free-stream, and thereby delay transition on a supersonic wind-tunnel nozzle.
2. The measured surface temperatures did not compare well with the calculated values from the 2-D boundary layer code. Apparently, the conductivity of the aluminum

caused local wall temperatures to be influenced by downstream conditions in addition to upstream conditions. Ambient conditions also influenced adiabatic wall temperatures. The measured wall temperatures for the unheated condition were as much as 25 °C (8%) different from their respective calculated values at the nozzle exit. Measured and calculated "adiabatic" wall temperatures showed closer agreement near the throat area. Off-axis measurements showed that T_w was laterally uniform within 2% at all streamwise locations both with and without heating.

3. Calculations from the two dimensional boundary layer code showed that the heat flux was directed *into* the boundary layer over the entire length of the nozzle. The largest heat transfer occurred at the throat and decreased asymptotically to near zero at the nozzle trailing edge. For the first time in stability theory, the boundary layer has shown a stabilizing trend in the presence of surface heating. Previous results for theoretical flat plate studies have relied on upstream heating to create a local "cooling" immediately downstream of the heated region.
4. Measured mean boundary layer velocity profiles with and without heating compared well with theory when scaled by the ratio of computed to measured boundary layer thickness ($\delta_{theory}/\delta_{exp}$). Both theory and experiment showed a fuller profile with no surface heating. This result is consistent with classical predictions of the boundary layer equations with negligible streamwise curvature [12].

5. Measured temperature profiles showed static temperatures in the boundary layer consistently lower than those obtained numerically. The origin of this difference is not known and it cannot be stated categorically that the computer code did not predict correct static temperatures. The total temperature probe and calibration curve have significant room for error. The fact that temperature profiles disagree slightly and the velocity profiles agree well is an indication that the velocity depends only mildly on the total temperature measurement. Both experimental and theoretical profiles showed the same trends with and without surface heating; and, both showed the same profile shapes. As a result, the critical temperature derivatives were well predicted through the boundary layer.
6. Measured boundary layer thicknesses and integral properties were consistently higher than the theoretical values for cases 1 and 2 of the present study. These measurements are in contradiction with previous adiabatic measurements made by Demetriades [23] and Berger [24]. However, case 3 showed similar results to those of Demetriades and Berger. Surface heating was shown by both experiment and theory to have a negligible effect on the boundary layer thickness in all cases. The mechanism is expected to be that found in the similarity solutions of Cohen and Reshotko [39]: The fluid density decreases with heating and the pressure gradient creates a stronger effect which dominates the viscous retardation [39]. The negligible change in the boundary layer thickness sheds doubt on the previous argument [5] of transition delay related to the relative surface roughness height.

Theory and experiment both showed that the displacement thickness increased with heating and the momentum thickness decreased. These two trends are due to the decrease in fluid density near the nozzle's heated surface.

7. In view of reported decrease in fluctuations within the boundary layer with surface heating and in view of the above conclusions that, for the T_w/T_{aw} range used ($\approx$1.16 at the throat), the changes of the mean boundary layer are so small, it is concluded that the boundary layer stabilization with heating is extremely sensitive to the ratio T_w/T_{aw}.

REFERENCES CITED

1. Morkovin, M. V., "On Supersonic Wind Tunnels With Low Free-Stream Disturbances", *Journal of Applied Mechanics*, Tran. ASME, Vol 26, Series E, June 1959, pp. 319-324.

2. Laderman, A. J., "Review of Wind Tunnel Free-stream Pressure Fluctuations", *AIAA Journal*, Vol. 15, No. 5, April 1977, pp. 605-608.

3. Beckwith, I. E., "Free Stream Noise and Transition Measurements on a Cone in a Mach 3.5 Pilot Low-Disturbance Tunnel", *NASA TP-2180*, Sept. 1983.

4. Morkovin, M. V., "Roughness Panel Summary; On Receptivity to Environmental Disturbances; on Roughness-Induced Transition: Facts, Views and Speculations", *Instability and Transition, Vol. I*, Springer-Verlag, 1990, pp. 265-295.

5. Harvey, W. D., Stainback, P. C., Anders, J. B., and Cary, A. M., "Nozzle-Wall Boundary Layer Transition and Free Stream Disturbances at Mach 5", *AIAA Journal*, Vol 13, no. 3, March 1975, pp. 307-314.

6. Demetriades, A., "Boundary Layer Transition in a Supersonic Nozzle Throat in the Presence of Cooling and Surface Roughness", *MSU/SWT TR 82-01*, August 1981.

7. Demetriades, A. "Stabilization of Nozzle Boundary Layer by Surface Heating", *AIAA 94-250*, June 1994.

8. Schlichting, H., *Boundary Layer Theory*, 7th ed., McGraw Hill Publishing Co., published 1979.

9. White, F. M., *Viscous Fluid Flow*, 2nd ed., McGraw-Hill Inc., published 1991.

10. Mack, L. M., "Boundary Layer Stability Theory," Rept. 900-277 Rev. A, Jet Propulsion Lab., Pasadena, CA, Nov. 1969.

11. Lees, L., "The Stability of the Laminar Boundary Layer in a Compressible Fluid", Rept. 876, NACA, Sept. 1946.

12. Gad-el-Hak, M., "Transition Control", *Instability and Transition, Vol. I*, Springer-Verlag, 1990, pp. 319-354.

13. Mack, L. M., "Linear Stability and the Problem of Supersonic Boundary-Layer Transition", *AIAA Journal*, Vol 13, No. 3, March 1975, pp. 278-289.

14. Reshotko, E., "Transition Reversal and Tollmien-Schlichting Instability," *The Physics of Fluids*, Vol. 6, No. 3, March 1963, pp. 335-342.

15. Maestrello, L., "Transition Delay and Relaminarization of Turbulent Flow", *Instability and Transition, Vol. I*, Springer-Verlag, 1990, pp. 153-159.

16. Dovgal A.V., Levchenko V.Y., and Timofeev V.A., "Boundary -Layer Control by a Local Heating of the Wall," *Laminar-Turbulent Transition* (D. Arnal and R. Michel, eds.), Springer-Verlag, Berlin, 1990, pp. 113-121.

17. Batt, R.G., and Legner, H.H., "A Review of Roughness-Induced Nosetip Transition", *AIAA Paper No. 81-1223*, June 1981.

18. Massad, J. A., Nayfeh, A. H., "Laminar Flow Control of Subsonic Boundary Layers by Suction and Heat-Transfer Strips", *Physics of Fluids*, Vol. 4, No. 6, June 1992, pp. 1259-1270.

19. Lysenko, V.I., and Maslov, A.A., "Transition Reversal and One of Its Causes," AIAA Journal, Vol. 19, No. 6, June 1981, pp. 705-708.

20. Mueller, R.L., "Boundary Layer Transition on the Nozzle of a Two-Dimensional Mach 3.0 Wind Tunnel", *M.S. Thesis*, Dept. of Mechanical Engineering, Montana State Univ., December 1993.

21. Cornelius, D. and Leonardson, D., "Effect of Heating MSU-SWT Nozzle Throat on Boundary Layer Transition", ME470 rept., Dept. of Mechanical Engineering, Montana State Univ., May 1993.

22. King, L.S. and Demetriades, A., "Comparisons of Predictions with Measurements for a Quiet Supersonic Tunnel," *AIAA Paper 93-0344*, January 1993.

23. Demetriades, A., "Roughness-Induced Transition on a Blunt Body Simulated by a DeLeval Nozzle Throat," Ford Aerospace & Communications Corp., *Report No. U-6496*, November 1978.

24. Berger, S.E., "Experiments on the Relationship between Shape and Effectiveness for Three Dimensional Boundary-Layer Trips at Supersonic Speeds," *M.S. Thesis*, Dept. of Mechanical Engineering, Montana State University, May 1988.

25. Harris, J.E. and Blanchard, D.K., "Computer Program for Solving Laminar, Transitional or Turbulent Compressible Boundary-Layer Equations for Two-Dimensional and Axisymmetric Flow," *NASA TM 83207*, February 1982.

26. King, L., Private Communication, July, 1994.

27. Malik, M.R., "e^{Malik}: A New Spatial Stability Analysis Code for Transition Prediction Using the e^N Method," High Technology Corporation, Report No. HTC-8902, March 1989.

28. Pulliam, T.H. and Steger, J.L., "Recent Improvements in Efficiency, Accuracy, and Convergence for Implicit Approximate Factorization Algorithms." *AIAA Journal*, Vol. 16, No. 7, July 1978.

29. Drummond, D., Rogers, B., and Demetriades, A., "Design and Operating Characteristics of the Supersonic Wind-Tunnel," SWT-TR-81-01.

30. 3M, "Product Information: Scotch VHB Joining Systems, Double Coated Acrylic Foam Tapes and Adhesive Transfer Tapes," 3M, Minnesota, 1991.

31. OMEGA, *OMEGA Temperature Measurement Handbook & Encyclopedia®*, Vol. 28, Omega Engineering, Inc, USA, 1992, pp Z-12 - Z-24.

32. Berger, S.E., "Supersonic Wind Tunnel Automated Data Acquisition System Operation Manual," MSU/SWT Report TR 88-01, Dept. of Mechanical Engineering, Montana State University, Bozeman, MT, 1988.

33. Chambre, P.L., and Schaaf, S.A., "The Impact Tube," *Physical Measurements in Gas Dynamics and Combustion*, Princeton University Press, 1954, pp. 111-123.

34. Liepmann, H.W. and Roshko, A., Elements of Gas Dynamics, John Wiley & Sons, 1957.

35. Ames Research Staff, "Equations, Tables and Charts for Compressible Flow," *NACA TR 1135*, 1953.

36. Eber, G.R., "Shielded Thermocouples," *Physical Measurements in Gas Dynamics and Combustion*, Princeton University Press, 1954, pp. 186-197.

37. Demetriades, A., "Recent Developments in Immersible Sensors for High-Speed Flow Turbulence," Fluid Measurement and Instrumentation Forum, ASME, FED-Vol. 108., 1991.

38. Beckwith, I.E., and Holley, B.B., "Görtler Vortices and Transition in Wall Boundary Layers of Two Mach 5 Nozzles," *NASA TP 1869*, August 1981.

39. Cohen, C.B. and Reshotko, E., "Similar Solutions for the Compressible Laminar Boundary Layer with Heat Transfer and Pressure Gradient," *NACA TR 1293*, 1956

APPENDIX A: Experimental Data Reduction Program & Flow Charts

Program BLREDUC.EXE is written in FORTRAN 77 with several MS-FORTRAN compiler options. The code was compiled with the Microsoft FORTRAN Compiler (v. 5.10) under DOS 6.22, for execution on a PC with a math co-processor.

NAMELIST INPUT

The following is a description of the namelist input used in the control file CNTRL.DAT. The following serve as input and option controls for reducing boundary layer profiles with position, pressure, and temperature data. Data is assumed to be in an ASCII text file in tabular form.

Variable name	Variable description
NAMEIN	Name of the input file containing raw data. This file must contain at least y position of the pitot probe and pitot pressure voltages/data. Total temperature data is optional; the presence of which is noted by the value of NORDER.
NAMETIN	Name of the input file containing only total temperature data. If NAMETIN = '*NONE*', a separate file for T_t' data does not exist. Default = '*NONE*'
NAMEOUT	Name of the output file
TITLE	An 80 character string reprinted to the output file. Can be used for description of data. Default = ' '.
PO	Stagnation Pressure (mmHg)
TO	Stagnation Temperature (°K)
XPOS	Stream-wise location at which profile data was taken. Used to calculate Re based on x. (m)
ZPOS	Span-wise location, reprinted in output file for future reference. Default = 0.0 (m)
FILL	0.0 - a data fill is not used and a linear calibration equation described by MYPOS and BYPOS is used to convert input data to physical units. $\neq$ 0.0, A data fill is used to construct the y-position data. y-position is started at 0.0 and incremented in equal steps = FILL (m). Default = 0.0

MYPOS — Slope of the y-position linear calibration curve. If no total temperature data is contained in file NAMEIN, MYPOS applies only to the y-position of the pitot probe. (Converted y-position has the units of m)

BYPOS — Offset of the y-position linear calibration curve. If no total temperature data is contained in file NAMEIN, MYPOS applies only to the y-position of the pitot probe. (Converted y-position has the units of m)

YERR — Offset correction to physical y-position after input data has been converted to physical units. Default = 0.0 (m)

FILLT — 0.0 - a data fill for T_t' data is not used and a linear calibration equation described by MYTPOS and BYTPOS is used to convert input data to physical units.
$\neq$ 0.0, A data fill is used to construct the y-position data of the T_t' probe. y-position is started at 0.0 and incremented in equal steps = FILLT (m). Default = 0.0

MYTPOS — Slope of the y-position linear calibration curve for the total temperature probe. This value is not used if total temperature data is contained in file NAMEIN. In that case, MYPOS is used. (Converted y-position has the units of m)

BYTPOS — Offset of the y-position linear calibration curve for the total temperature probe. See comments on MYTPOS. (Converted y-position has the units of m)

YTERR — Offset correction to physical y-position for total temperature probe after input data has been converted to physical units. Default = 0.0 (m)

MPTP — Slope of the pressure transducer linear calibration curve. (Transducer is assumed to measure pitot pressure.) (Converted pressure has the units of mmHg)

BPTP — Offset of the pressure transducer linear calibration curve. (Transducer is assumed to measure pitot pressure.) (Converted pressure has the units of mmHg)

PSTAT — Static pressure calculation option.
PSTAT > 0: P_e specified by user. BPTP adjusted such that

P_e=PSTAT
PSTAT = 0: First pressure data point = P_e. BPTP adjusted such that "edge" pressure equals pressure at the first data point.
PSTAT < 0: P_e calculated directly from edge condition and P_o. No modifications are made to BPTP.
Default = -1

MTTP — Slope of the linear thermocouple calibration. Note that the voltage calculated from this calibration is then converted to temperature using the Type-J thermocouple conversion equation.

BTTP — Offset of the linear thermocouple calibration. See note on MTTP description.

MODTT — Total temperature model #. MODTT = 0: $T_t = T_o$ through b.l.;
MODTT=1: T_t = linear interpolation between specified T_w and T_o @ b.l. edge.;
MODTT=2: recovery factor linearly dependent upon M_e,
$rf = (T_t'-T)/(T_t-T)$
MODTT=3: recovery factor linearly dependent upon M,
$rf = (T_t'-T)/(T_t-T)$
MODTT=4: recovery factor linearly dependent upon Re_{TTP},
$rf = (T_t'-T)/(T_t-T)$
MODTT=5: recovery factor linearly dependent upon M_e, $rf = T_t'/T_t$
MODTT=6: recovery factor linearly dependent upon M, $rf = T_t'/T_t$
MODTT=7: recovery factor linearly dependent upon Re_{TTP},
$rf = T_t'/T_t$
Default = 1

MRF — Slope of the recovery factor curve. Calculated in program if MODTT = 3,4,6,7 Default = 0.0

BRF — Offset of the recovery factor curve. Default = 0.90

HTTP — Frontal Height of the T_t' probe. Used to calculate Re_{TTP} on which the recovery factor is based if MODTT = 4,7. (m) Default = 7.62e-5

NORDER — Order of y (1), P_t' (2), and T_t' (3), in the input file. NORDER = 213, signifies data is in columnar order of P_t', y, T_t'. If no T_t' data is specified (e.g. NORDER = 12) The file is assumed to contain no total temperature data. Default = 213

NTORDER — Order of y (1) and T_t' (2), in the input file, NAMETIN. NORDER = 12, signifies data is in columnar order of y, T_t'. This option not used if NAMETIN = '*NOFILE*'. Default = 213

NEDGE — An array of two integers noting the range of data points which define the boundary layer "edge." The later is the location at which the free-stream conditions as seen by the boundary layer are calculated from. NEDGE(1) = starting data point in range; NEDGE(2) = ending data point in range. Edge data is taken as an average value of data points contained in this range. NEDGE(1) = NEDGE(2) = -1 signifies the code to determine the "edge" automatically. Default = -1

NEAVG — Number of data points used in the average to determine the "edge" properties. Used only if NEDGE(1) and NEDGE(2) are not specified (i.e. = -1). Default = 5

NTW — An array of two integers noting the range of data points which define the "wall" values for the T_t' probe. Similar to NEDGE( ). NTW(1) = starting data point in range; NTW(2) = ending data point in range. Wall data is taken as an average value of data points contained in this range. NTW(1) = NTW(2) = -1 signifies the code to determine the wall points automatically. Default = -1

NTWAVG — Number of data points averaged to calculate T_w as measured by the overhead probe. Similar to NEAVG. Default = 5

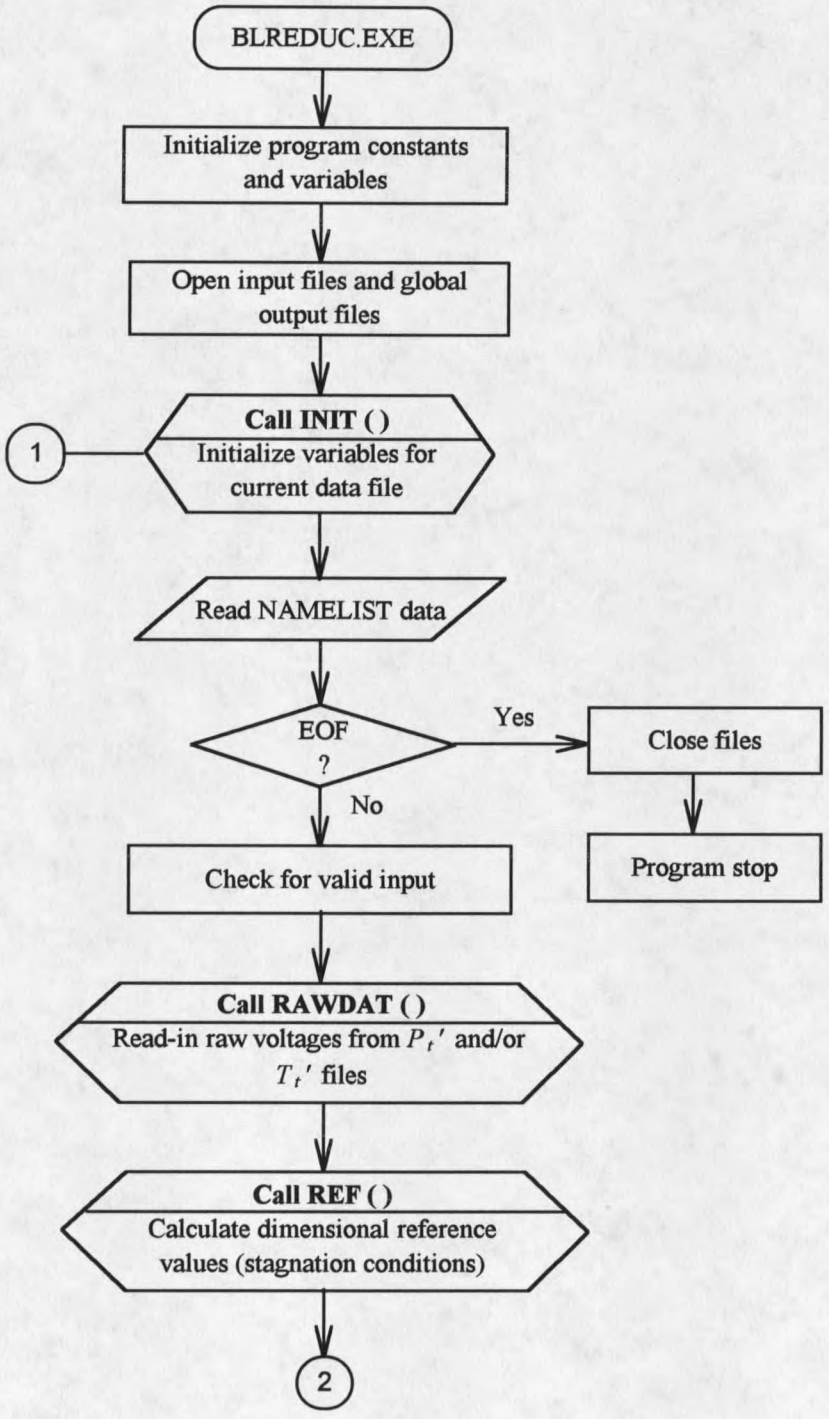
BLE — Value of u/u_e at which the boundary layer thickness, d, is computed. Default = 0.99

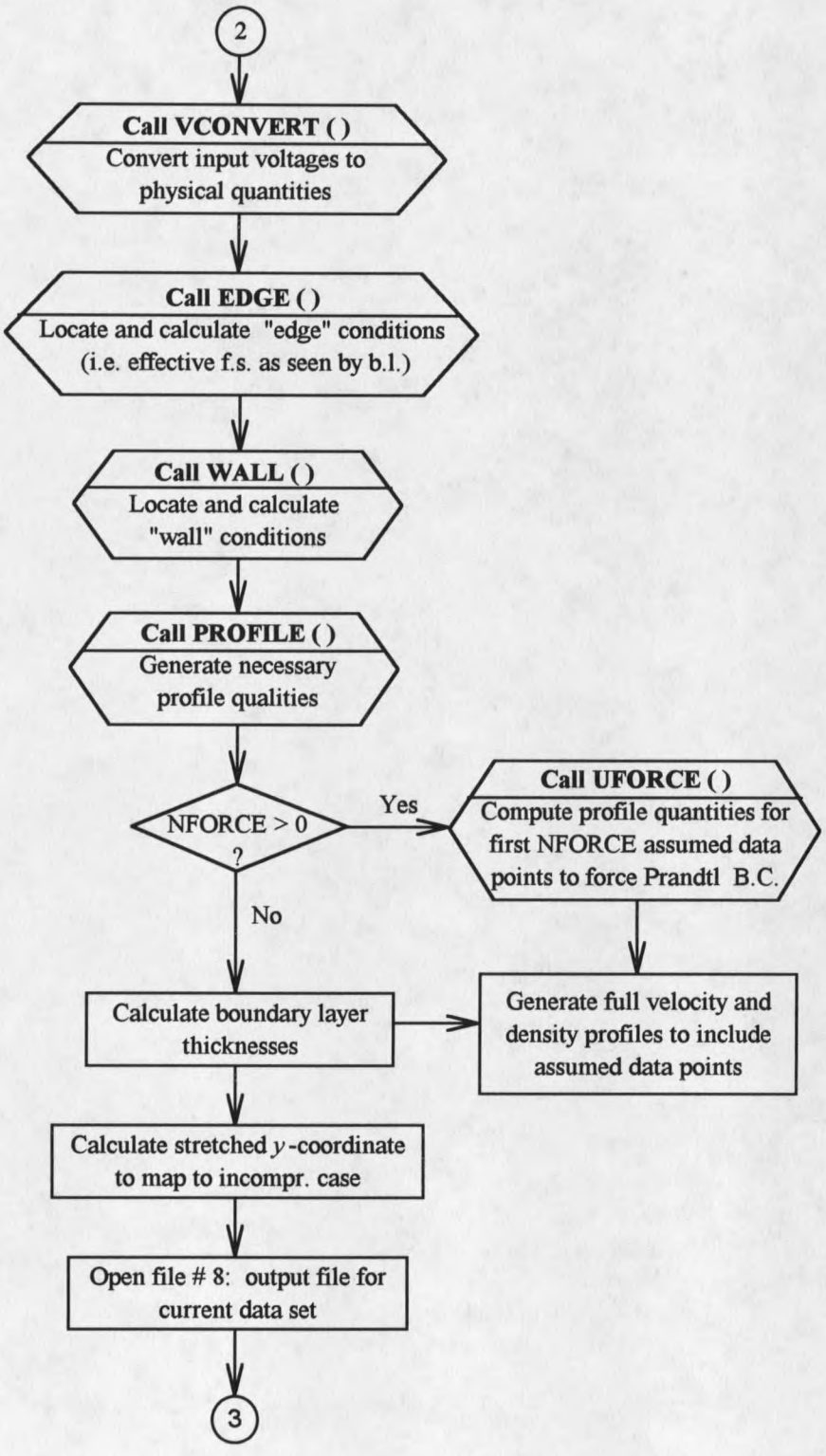
NFORCE — Number of data points to map onto a straight line (for u/u_e vs. y profile) which forces the velocity at the wall to be 0.0 as required by the Prandtl boundary condition. The straight line is defined by u/u_e = 0 @ y = 0 and $u/u_e = (u/u_e)_{\text{measured}}$ @ $(y)_{\text{data point \# NFORCE}}$. Default = 0

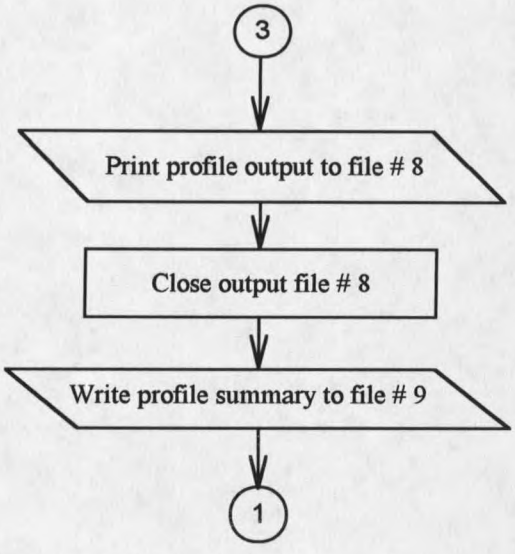
VTWLOC — Un-amplified thermocouple voltage from thermocouple which measures T_w at the location of the profile. Reprinted in output file for comparison with T_w measured by overhead probe. Default = 0.0

KODTW — 0 - use T_w of that measured by the T_t' overhead probe
1 - use T_w calculated from by VTWLOC signal. Default = 0

Flow Chart for BLREDUC.EXE: Mainline




2
Call VCONVERT ( )
Convert input voltages to physical quantities
Call EDGE ( )
Locate and calculate "edge" conditions (i.e. effective f.s. as seen by b.l.)
Call WALL ( )
Locate and calculate "wall" conditions
Call PROFILE ( )
Generate necessary profile qualities
NFORCE > 0 ?
Yes
Call UFORCE ( )
Compute profile quantities for first NFORCE assumed data points to force Prandtl B.C.
No
Calculate boundary layer thicknesses
Generate full velocity and density profiles to include assumed data points
Calculate stretched y-coordinate to map to incompr. case
Open file # 8: output file for current data set
3


3
Print profile output to file # 8
Close output file # 8
Write profile summary to file # 9
1

```
C  This program reduces data from a profile output into physical
C  parameters.
C  Written by T. Brogan, 9/94

      Program Begin
C
C     Declare variables
C
      parameter(mdata=250,nfiles=100)
      integer nedge(2), ntw(2),
     &  ndata,  norder,  ntoff, ierr,JY,JT,JP,neavg,ntwavg,
     & nforce
      real V(3,mdata),ypos(mdata),   PTP(mdata), ttp(mdata),dens(mdata),
     &       U(mdata),  MA(mdata),    TT(mdata),   T(mdata),  mu(mdata),
     &      PT(mdata),  Uf(mdata), DENSf(mdata), Maf(mdata), Ptf(mdata),
     &      Tf(mdata), muf(mdata),  PTPf(mdata),VT(2,mdata),  rf(mdata),
     &  TTPW(2,mdata), URe(mdata),ytilda(mdata), TTf(mdata), rff(mdata),
     &    URef(mdata),
     &    ME,    PE,    TE,    TTE,  xpos,   zpos,  fill,  mptp,  bptp,
     & MTTP,  BTTP,  MYPOS,  BYPOS,   mue,     UE, DENSE,   muo, denso,
     &soundo, lstar, MYTPOS, BYTPOS, gamma, vtwloc,  yerr, yterr, delta,
     & dstar, theta,   TTPE,   mrf,    brf,   rfe, Pstat, Pshift
      character namein*50, nameout*50,  title*80, fitdesc*30,
     &          nofile*6, nametin*50, pfname*12
      COMMON /REFRNC/ Po, To, gamma
C
C     name1 is a list of initial values read in for each data file.
C
      NAMELIST /name1/    Po,     To,   xpos,   zpos,    fill,    yerr,
     &          mptp,   bptp,  bypos,  mypos, norder,  namein, nameout,
     &         title,  nedge,  neavg,    ble, nforce, upforce,
     &         modtt, vtwloc,   mttp,   bttp, bytpos,  mytpos,     mrf,
     &           brf,  yterr, ntwavg,nametin,    ntw, ntorder,   fillt,
     &          http,  Pstat,  kodtw
C
C     Open files
C
      open(unit=6,file='cntrl.dat',status='old',form='formatted')
      open(unit=9,file='summary.out',status='unknown',form='formatted')
C
C     Write header to summary file
C
      call getdat(iyr,imon,iday)
      call gettim(ihr,imin,isec,i100th)
      write(9,9974) imon,iday,iyr,ihr,imin
      write(9,9984)
C
C     Read data
C
      do 1000 ifile = 1,nfiles
C
```

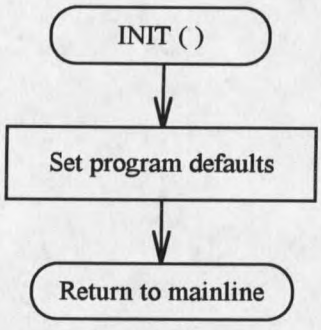
```
C     Initialize defaults
C
      call init(zpos,fill,yerr,yterr,norder,ntwavg,nedge,gamma,ntw,
     &   neavg,vtwloc,modtt,kodtw,ble,nforce,upforce,nametin,nofile,
     &   ntoff,ierr,ntorder,fillt,xref,mrf,brf,pfname,http, Pstat)
C
C     Read input variables from control file
C
      read(6,nml=name1,end=1020,iostat=ierr)
      if (ierr .gt. 0) then
          write(*,9990) ifile,ierr
          nferr = nferr+1
          goto 1000
      end if
C
C     Check for valid input (this is only a partial check for common
C     errors)
C
      if (int(norder/10) .eq. 0) then
          write(*,'(A,I3,2x,A,I3)')
     &     'ERROR:  VARIABLE NORDER SET INCORRECTLY.  NORDER =', norder,
     &     'FILE #',ifile
          nferr = nferr+1
          goto 1000
      end if
      if ((int(norder/100) .gt. 0) .and. (nametin .ne. nofile)) then
          write(*,'(1X,A,/1X,A,A,2X,A,I3,2x,A,I3)')
     &     'ERROR: VARIABLE NORDER OR NAMETIN SET INCORRECTLY.',
     &     'NAMETIN = ',NAMETIN,'NORDER =',NORDER,'FILE #',IFILE
          nferr = nferr+1
          goto 1000
      end if
      if (((modtt .ge. 2) .and. (nametin .eq. nofile)) .and.
     &    (int(norder/100) .eq. 0)) then
          write(*,'(1X/1X,A,A/1X,A,A,/1X,A,A,/1X,A,I2,/1X,A,I3)')
     &     'ERROR:  MODTT SET INCORRECTLY OR NO TEMPERATURE DATA FILE ',
     &     'SPECIFIED','NAMEIN = ',NAMEIN,'NAMETIN = ', NAMETIN,
     &     'MODTT = ',MODTT,'NORDER = ', NORDER
          nferr = nferr+1
          goto 1000
      end if
      xbar = xpos/xref
C
C     Read raw profile data
C
      call rawdat(namein,nametin,norder,mdata,fill,fillt,
     &     yterr,ntoff,V,VT,ndata,ntdata,nofile,nferr,*1000)
C
      write(*,9999) namein
      if (nametin.ne.nofile) write(*,9978) nametin
C
```

```
C     Calculate dimensional reference values
C
      call ref(Po, To, gamma, UReo,soundo,muo,denso)
C
C     Convert voltages to physical quantities
C
      call vconvert(V,VT,YPOS,PTP,TTP,norder,ndata,fill,JY,JP,JT,
     & MPTP,BPTP,MTTP,BTTP,BYPOS,MYPOS,Po,To,yerr,yterr,ntorder,BYTPOS,
     & MYTPOS,FILLT,NTOFF,ntdata,TTPW)
C
C     Determine boundary layer "stream" location and compute "free-
C     stream" quantities:
C
      call edge(PTP,TTP,ME,PE,TE,TTE,DENSE,ndata,nedge,gamma,
     &          neavg,ue,mue,rfacte,To,PTPE,modtt,mrf,brf,ttpe,rfe,URee,
     &          Reettp,UReo,http,Pstat,BPTP, Po,Pshift)
      lstar = sqrt(xpos/(URee*UReo))
C
C     Determine wall values
C
      Call wall(TTPW,vtwloc,ntwavg,ntw,Twp,Twt,ntoff,To,kodtw,Tw)
C
C     Generate profile quantities
C
      call profile(YPOS,PTP,TTP,ME,PE,PTPE,nedge,
     &             Pt,Ma,TT,T,U,DENS,MU,modtt,Tw,TE,ndata,ble,rf,
     &             mrf,brf,URe,Reettp)
C
C     Adjust profile quantities to force Prandtl boundary condition
C     @ wall. And calculate integral properties.
C
      if(nforce .gt. 0) then
     &    call uforce(YPOS,U,Uf,Maf,Ptf,DENSf,Tf,PE,ME,TE,TTE,
     &                TT,TTF,rff,muf,gamma,nforce,ndata,To,fitdesc,
     &                URef,mrf,brf,Reettp,TTP,modtt)
          do 70 i=nforce+1,ndata
              uf(i) = u(i)
   70       densf(i) = dens(i)
          nbldat=ndata
          delta=d1(uf,ypos,ble,nbldat,nedge(2))
          dstar=d2(uf,densf,ypos,nbldat,nedge(1))
          theta=d3(uf,densf,ypos,nbldat,nedge(1))
      else
          nbldat=ndata
          delta=d1(u,ypos,ble,nbldat,nedge(2))
          dstar=d2(u,dens,ypos,nbldat,nedge(1))
          theta=d3(u,dens,ypos,nbldat,nedge(1))
      end if
C
C     Transform y-coordinate to a stretched incompressible coordinate.
C
```

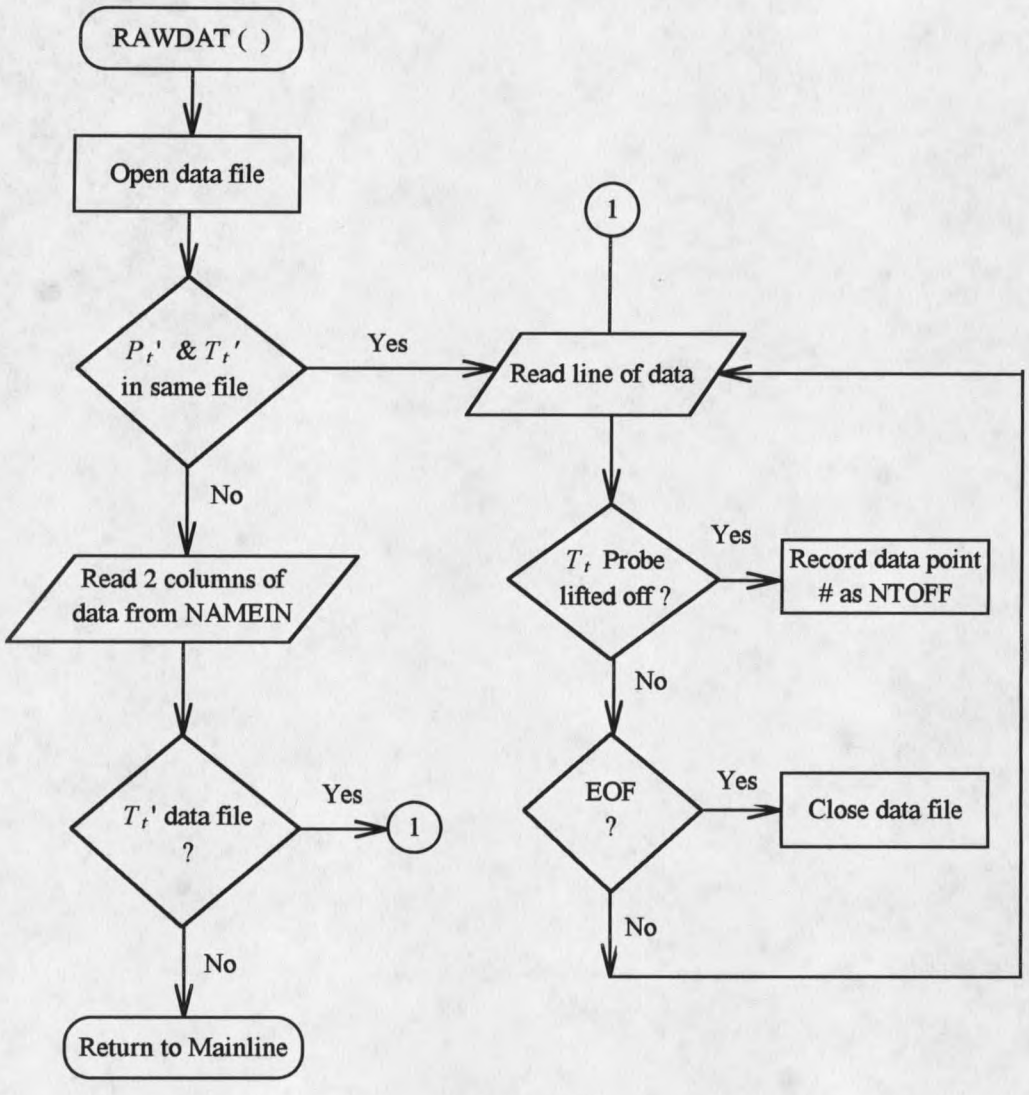
```
      call ystretch(ytilda,densf,TE/Tw,ypos,ndata)
C
C     Output
C
      call getdat(iyr,imon,iday)
      call gettim(ihr,imin,isec,i100th)
      call fname(namein,50,pfname)
C
      open(unit=8,file=nameout,status='unknown',form='formatted',
     &   iostat=ierr)
      if (ierr .gt. 0.0) then
           write(*,9987) nameout,ierr
           nferr = nferr+1
           goto 1000
      end if
C
      Rex = URee*UReo*xpos
C
      write(8,9997) title,namein,nametin,imon,iday,iyr,ihr,imin,Po,To,
     &              xpos,zpos,mptp,bptp,Pstat,mttp,bttp,MRF,BRF,mypos,
     &              bypos,mytpos,bytpos,fill,yerr,fillt,yterr
      write(8,9985) denso,muo,soundo,UReo,http,xref
      write(8,9996) ndata,norder,nforce,ble,ntdata,ntorder,ntoff,modtt
      write(8,9995) Rex,delta,dstar,theta
      write(8,9986) ntw(1),ntw(2),twp,vtwloc,Twt
      write(8,9994) nedge(1),nedge(2),ypos(nedge(1)),ypos(nedge(2)),
     &              Me,Pe,Te,TTE,rfacte,densé,ue,mue,PTPE,URee,TTPE,
     &              Reettp
      write(8,9993)
      if (nforce.gt.0) then
         write(8,9981) nforce,fitdesc
         if(YPOS(1) .ne. 0.0)
     &     write(8,9980) 0,'-----   ','-----   ','-----   ',0.0,
     &        '-----   ',Twp,PE,Tw,Tw/TE,0.0,0.0,0.0,brf
         if (upforce .eq. 'u' .or. upforce .eq. 'U') then
           do 40 k = 1,nforce
40           write(8,9982) k,V(JY,k),'-----   ',V(JT,k),YPOS(k)/DELTA,
     &         '-----   ',TTP(k), PTf(k), TTf(k), Tf(k), Uf(k),
     &         Maf(k),URef(k), rff(k)
         end if
         write(8,*) ' '
      else
         write(8,9980) 0,'-----   ','-----   ','-----   ',0.0,
     &        '-----   ',Twp,PE,Tw,Tw/TE,0.0,0.0,0.0,brf
      end if
      do 60 k = 1,ndata
60      write(8,9992) k,V(JY,k),V(JP,k),V(JT,k),YPOS(k)/delta,PTP(k),
     &         TTP(k), PT(k), TT(k), T(k), U(k), Ma(k),
     &         URe(k),rf(k)
      if (ntoff .gt. 1) then
            write(8,9976)
```

```
            write(8,9975) (k,TTPW(1,k)/delta,TTPW(2,k), k=1,ntoff)
      end if
      close (8)
C
      YE = (ypos(nedge(1))+ypos(nedge(2)))/2.0
      write(9,9983) ifile,pfname,xpos/xref,Po,To,Ye,Me,Pe,Te,TTPE,
     &        URee*UReo,rfacte,Rex,delta,dstar,theta,lstar
C
 1000 continue
C
 1020 close (6)
      close (9)
      write(*,9989) IFILE-1,NFERR
C
C     Output format statements:
C
 9999 format(1x/1x,'REDUCING DATA FROM FILE: ',A)
 9997 format(1X,'TITLE:  ',A80,/
     &1x,'INPUT DATA FILE:  ',A,/
     &1X,'TEMP. DATA FILE:  ',A,/
     &1X,'DATA REDUCED:  ',2(I2.2,'/'),I4,2X,I2,':',I2.2,//
     &1X,'**** INPUT AND OUTPUT IN S.I. UNITS W/ PRESSURES IN mmHg ****'
     &//
     &1X,'TUNNEL CONDITIONS:  ','Po = ',1PE10.3,2X,'To = ',1PE10.3,/
     &1X,'PROFILE LOCATION:   ','X  = ',1PE10.3,2X,'Z  = ',1PE10.3,/1x/
     &1X,'CALIBRATIONS: (Y = M*X + B)  ',/1x/
     &1X,'           PRESSURE:     M = ',1PE11.4,3X,'B = ',1PE11.4,3X,
     &          'PSTAT = ',1PE11.4,/
     &1X,'        TEMPERATURE:     M = ',1PE11.4,3X,'B = ',1PE11.4,/
     &1X,'   RECOVERY FACTOR:     M = ',1PE11.4,3X,'B = ',1PE11.4,/
     &1X,'         Y-POSITION:    M = ',1PE11.4,3X,'B = ',1PE11.4,/
     &1x,'        Yt-POSITION:    M = ',1PE11.4,3X,'B = ',1PE11.4,/
     &1X,T7,'         FILL = ',1PE10.3,3X,' YERR = ',1PE10.3,3X,/
     &1X,T7,'        FILLT = ',1PE10.3,3X,'YTERR = ',1PE10.3,3X)
 9985 format(1x/1x, 'REFERENCE VALUES: ',/1X/1X,'DENSo =',1PE11.4,2X,
     &         ' MUo =',1PE11.4,2X,'SOUNDo =',1PE11.4,2X,'UReo =',
     &         1PE11.4,/1x,
     &        ' HTTP =',1PE11.4,2X,'XREF =',1PE11.4)
 9996 format(1x/1x, 'PROFILE SUMMARY:  ',/1X/
     &1X,' NDATA = ',I3,2X,' NORDER = ',I3,2X,'NFORCE = ',I3,2X,
     &          'BLEDGE =',F6.3,/
     &1X,'NTDATA = ',I3,2X,'NTORDER = ',I3,2X,' NTOFF = ',I3,2X,
     &          ' MODTT = ',I2)
 9995 format(1X,'Rex =',1PE10.3,3X,'DELTA =',1PE10.3,3X,'DSTAR =',
     &         1PE10.3,3X,'THETA =',1PE10.3)
 9994 format(1X/1X,'EDGE VALUES:'/1X/2X,'NEDGE(1) = ',I3,2x,
     &          'NEDGE(2) = ',I3,2X,'YEDGE(1) =',1PE10.3,2X,'YEDGE(2) ='
     &          ,1PE10.3,/
     &2X,'  ME =',1PE11.4,2X,'   PE =',1PE11.4,2X,'  TE =',1PE11.4,2X,
     &          ' TTE =',1PE11.4,/
     &2X,' RFE =',1PE11.4,2X,'DENSE =',1PE11.4,2X,'  UE =',1PE11.4,2X,
```

```
     &          ' MUE =',1PE11.4,/
     &2X,'PTPE =',1PE11.4,2x,' UREE =',1PE11.4,2x,'TTPE =',1PE11.4,2X,
     &          'RETE =',1PE11.4)
 9993 format(1x/1x, 16HPROFILE LISTING: ,/1x/1x,
     & T2,'Point',T11,'Y_v',T22,'Pt_v',T34 'Tt_v',T45,'y/delta',T58,
     &'Pt''/Po',
     &T70,'Tt''/To',T82,'Pt/Po',T94'Tt/To',T107,'T/Te',T119,'U/Ue',T131,
     &'M/Me',T141,'URe/URee',T155,'rf'/,
     &'-------------------------------------------------------------------
     &-------------------------------------------------------------------
     &------------------------------')
 9992 format(2x,I3,13(1X,1PE11.4))
 9990 format(1x,'WARNING: ERROR IN READING CURRENT RECORD IN ''CNTRL.DAT
     &''',/1x,'CHECK VALIDITY OF NAMELIST VARIABLES',2X,'RECORD = ',I3,
     &2x,'ERROR # ',I4)
 9989 format(1x/1x,'END PROGRAM.  TOTAL RECORDS =',I3,2x,'RECORD ERRORS
     &=',I3)
 9987 format(1x,'ERROR IN OPENING FILE:  ',A,1X,'(ERROR # ',I4,')')
 9986 format(1x/1x'WALL VALUES:'/1X/1X,'NTW = ',I3,',',I3,2X,'Tw_p =',
     &1PE10.3,2X,'Tw_tv =',1pe10.3,2x'Tw_t =',1PE10.3)
 9984 format(1X,' FILE # ',' Pt'' FILE NAME',
     &'   XBAR    ','     Po    ','     To    ','   YEDGE   ',
     &'     Me    ','   Pe/Po   ','   Te/To   ','  Tt''e/To  ',
     &'    URee   ','   rfacte  ','    Rex    ','   DELTA   ',
     &'   DSTAR   ','   THETA   ','    LSTAR  ')
 9983 format(1x,I4,5x,A12,15(1x,1pe10.3))
 9982 format(2x,I3,1X,1PE11.4,1X,A11,2(1X,1PE11.4),1X,A11,8(1X,1PE11.4))
 9981 format(1x,'DATA POINTS 1-',I2,' ADJUSTED TO FORCE PRANDTL BOUND
     &ARY CONDITION @ WALL',1X,A)
 9980 format(2x,I3,3(1X,A11),1X,1PE11.4,1X,A11,8(1X,1PE11.4))
 9979 format(2x,I3,1X,1PE11.4,1X,A11,10(1X,1PE11.4))
 9978 format(1x,'       TEMPERATURE FILE: ',A)
 9977 format (a40,3(1x,1pe10.3),2(1x,i3),2(1x,1pe10.3),1x,0PF7.3,1x,
     &          1pe10.3)
 9976 format(1x/1x/1x,'WALL TEMPERATURE PROBE VALUES:',/1X/1X,
     &' Point',' y/delta   ','   Tt''/To  ',/1X,
     &'------------------------------')
 9975 format((2x,I4,2(1x,1pe11.4)))
 9974 format(1x,'DATA SUMMARY FILE'/
     &1X,'DATA REDUCED:  ',2(I2.2,'/'),I4,2X,I2,':',I2.2,//
     &1X,'**** INPUT AND OUTPUT IN S.I. UNITS W/ PRESSURES IN mmHg ****'
     &/)
C
      end
```

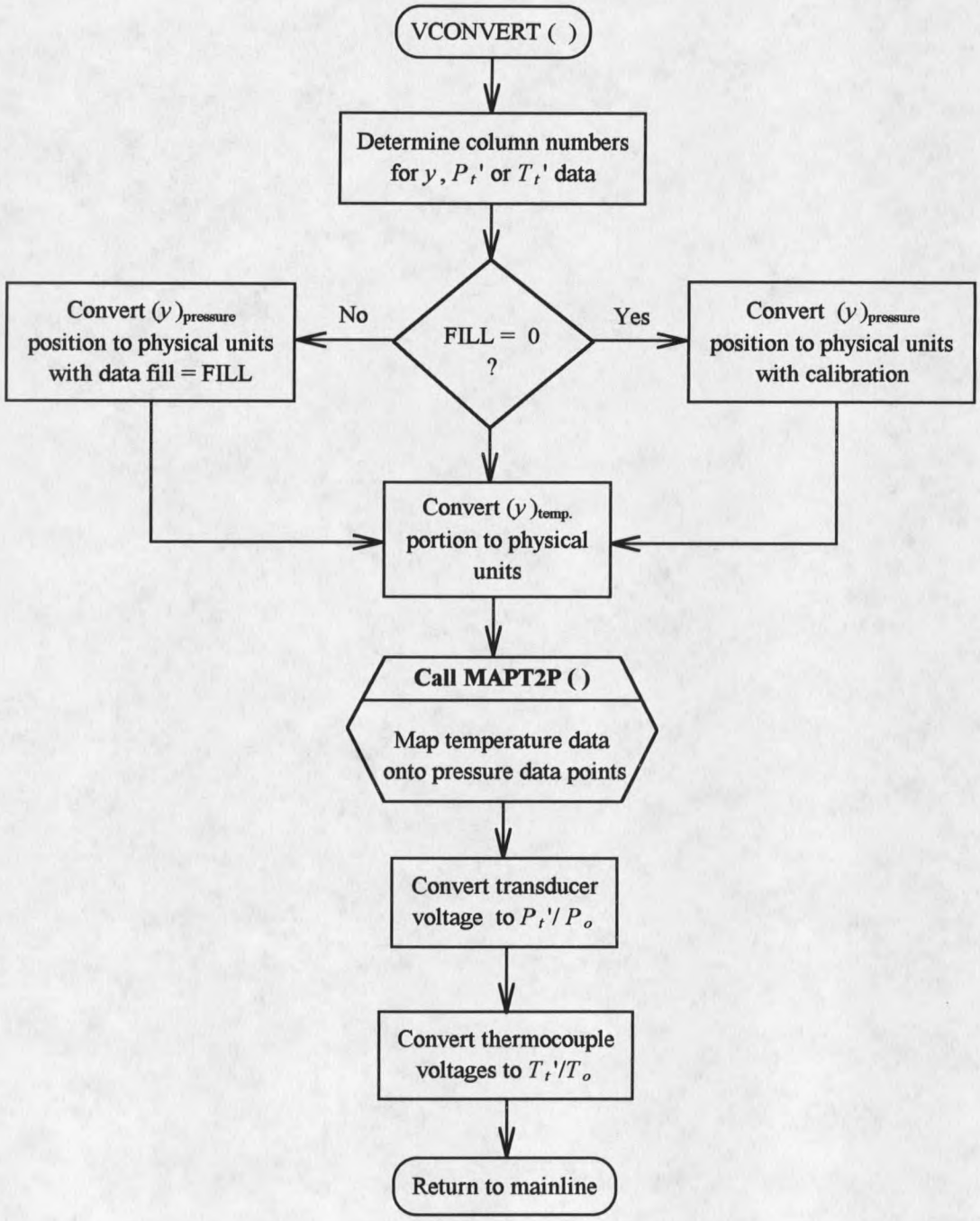

INIT ( )
Set program defaults
Return to mainline

```
C*********************************************************************
      subroutine init(zpos,fill,yerr,yterr,norder,ntwavg,nedge,gamma,
     &   ntw,neavg,vtwloc,modtt,kodtw,ble,nforce,upforce,nametin,nofile,
     &   ntoff,ierr,ntorder,fillt,xref,mrf,brf,pfname,http,Pstat)
C*********************************************************************
C
C     Subroutine INIT() initializes variables for each data file being
C     reduced.
C
      real zpos, fill,   yerr,    yterr,  gamma, vtwloc,   ble,   xref,
     &     mrf,brf,http, Pstat
      integer  norder, ntwavg, nedge(2), ntw(2),  modtt, kodtw, nforce,
     &          ntoff,   ierr,  ntorder
      character nametin*50, nofile*6, pfname*12
C
      gamma = 1.4
      zpos = 0.0
      fill = 0.0
      fillt = 0.0
      yerr = 0.0
      yterr = 0.0
      norder = 213
      ntwavg = 5
      nedge(1) = -1
      nedge(2) = -1
      ntw(1) = -1
      ntw(2) = -1
      neavg = 5
      ntoff = 1
      vtwloc = 0.0
      modtt = 1
      kodtw = 0
      ble = 0.99
      nforce = 0
      nofile = '*NONE*'
      nametin = nofile
      ierr = 0
      ntorder = 12
      xref = .384584
      mrf = 0.0
      brf = 0.90
      pfname = '
      http = 7.62e-5
      Pstat = -1
C
      return
      end
```


RAWDAT ( )
Open data file
P_t' & T_t' in same file
Yes
No
Read 2 columns of data from NAMEIN
T_t' data file ?
Yes
1
No
Return to Mainline
1
Read line of data
T_t Probe lifted off ?
Yes
Record data point # as NTOFF
No
EOF ?
Yes
Close data file
No

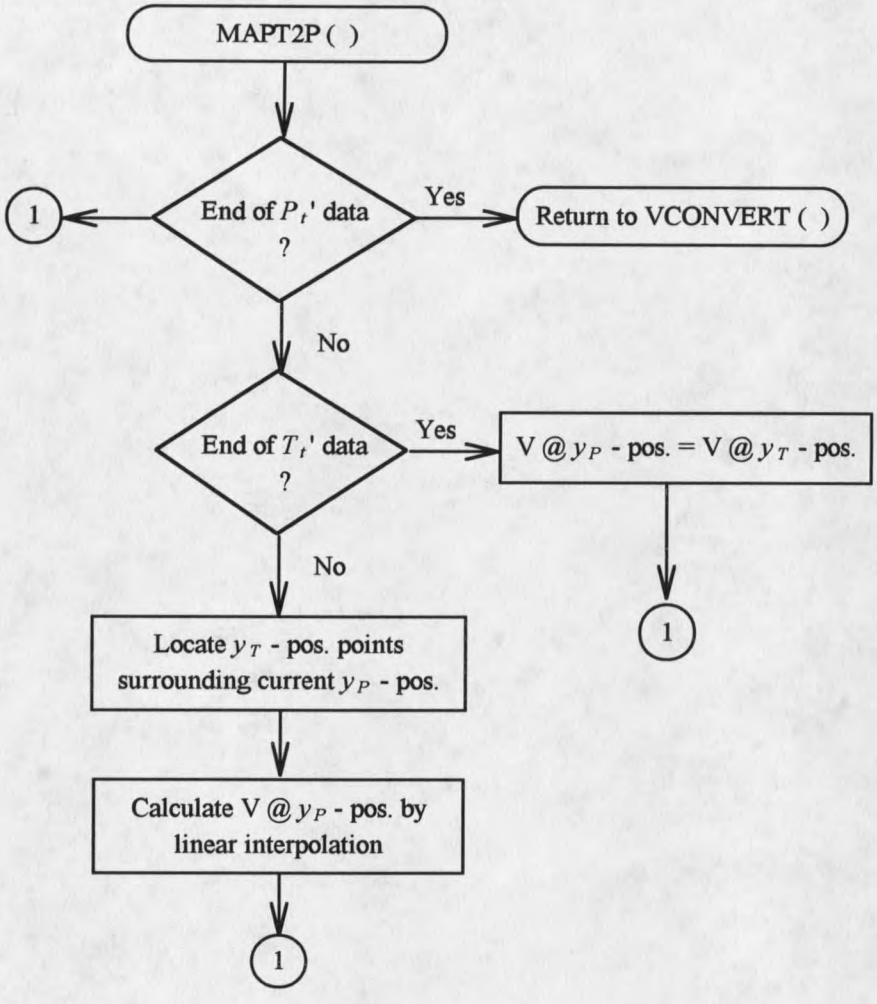
```
C*********************************************************************
      subroutine rawdat(namein,nametin,norder,mdata,fill,fillt,
     &     yterr,ntoff,V,VT,ndata,ntdata,nofile,nferr,*)
C*********************************************************************
C
C     Subroutine rawdata() reads in the raw voltage data from user
C     specified data files.  Voltages are returned in the array V(). If
C     temperature data is in a separate file, temperature and y-
C     position for Tt probe are stored in the array VT().
C
      integer mdata,ndata,ntdata,ntoff
      real V(3,*), VT(2,*)
      character namein*50, totline*80, nofile*6, nametin*50
C
      open(unit=7,file=namein,status='old',form='formatted',
     &   iostat=ierr)
      if (ierr .gt. 0) then
           write(*,9987) namein,ierr
           nferr = nferr+1
           return 1
      end if
C
      if (int(norder/100) .gt. 0) then
        do 10 i=1,mdata
           read(7,8999,end=15,err=15,iostat=ierr) totline
           do 5 j=1,20,2
               if (totline(j:j+1) .eq. '!!') then
                   if ((fill .ne. 0.0) .and. (yterr .ne. 0.0)) then
                      ntoff = i+yterr/fill
                      if (ntoff .lt. 1) write(*,9988) ntoff,yterr,fill
                   else
                      ntoff = i
                   end if
                   goto 10
               end if
    5      continue
           backspace (7)
   10    read(7,*,err=15,iostat=ierr) V(1,i),V(2,i),V(3,i)
      else
        do 80 i=1,mdata
   80     read(7,*,err=15,end=15,iostat=ierr) V(1,i),V(2,i)
      end if
C
   15 close (7)
      ndata=i-1
      if (int(norder/100) .gt. 0) ntdata = ndata
      if (ierr.gt.0) write(*,9998) ierr,ndata
      ierr = 0
C
C     If there is a file that contains total temperature probe data
C     separately, read temperature data.
```

```
      if (nametin .ne. nofile) then
         open(unit=5,file=nametin,status='old',form='formatted',
     &        iostat=ierr)
         if (ierr .gt. 0.0) then
                write(*,9987) nametin,ierr
                nferr = nferr+1
                return 1
         end if
         do 90 i=1,mdata
             read(5,8999,end=95) totline
             do 85 j=1,20,2
                if (totline(j:j+1) .eq. '!!') then
                     if ((fillt .ne. 0.0) .and. (yterr .ne. 0.0)) then
                        ntoff = i+yterr/fillt
                        if (ntoff .lt. 1) write(*,9988) ntoff,yterr,fillt
                     else
                        ntoff = i
                     end if
                     goto 90
                end if
    85       continue
             backspace (5)
C
C       The following 'dummy' variable is added only for use with
C       the data profiles used with the Brogan Experiments.  It assumes
C       a file containing pressure data in the 2nd column.
C
    90   read(5,*,err=95,iostat=ierr) VT(1,i),dummy,VT(2,i)
    95   close (5)
         ntdata=i-1
         if (ierr.gt.0) write(*,9998) ierr,ntdata+1
         ierr = 0
       end if
C
C      Read format statements:
C
 8999 format(A80)
C
C      Ouput format statements:
C
 9998 format(1x,'ERROR IN READING CURRENT DATA FILE:  (ERROR #',I4,
     &') DATA POINT: ',i4)
 9988 format(1x,'WARNING: NTOFF = ',I3,'.  (NTOFF MUST BE GREATER THAN 0
     &)',
     &/1X,'NTOFF = I+YTERR/FILL; WHERE I=DATA POINT AT WHICH RAW DATA IN
     &DICATES',/1X,'TEMPERATURE PROBE LIFT OFF.',/1X,'YTERR = ',1PE9.2,
     &2X,'FILL = ',1PE9.2)
 9987 format(1x,'ERROR IN OPENING FILE:  ',A,1X,'(ERROR # ',I4,')')

      return
      end
```

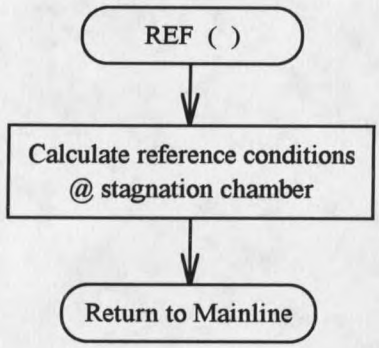

VCONVERT ( )
Determine column numbers for y, P_t' or T_t' data
FILL = 0 ?
No
Yes
Convert $(y)_{pressure}$ position to physical units with data fill = FILL
Convert $(y)_{pressure}$ position to physical units with calibration
Convert $(y)_{temp.}$ portion to physical units
Call MAPT2P ( )
Map temperature data onto pressure data points
Convert transducer voltage to P_t'/P_o
Convert thermocouple voltages to T_t'/T_o
Return to mainline

```
C*********************************************************************
      subroutine vconvert(V,VT,YPOS,PTP,TTP,norder,ndata,fill,JY,JP,JT,
     & MPTP,BPTP,MTTP,BTTP,BYPOS,MYPOS,Po,To,yerr,yterr,ntorder,BYTPOS,
     & MYTPOS,FILLT,NTOFF,ntdata,TTPW)
C*********************************************************************
C
C     Subroutine VCONVERT() converts rawdata read as voltages to
C     physical quantities.  Total temperature data is mapped onto
C     pressure data points.
C
      parameter (mdata = 250)
      integer ndata,norder,ntorder,JY,JP,JT,JTT,JYT,j(3)
      real V(3,*), YPOS(*),YTPOS(mdata),PTP(*),TTP(*),
     &    VT(2,*), TTPW(2,*),
     & MPTP,BPTP,MTTP,BTTP,BYPOS,MYPOS,MYTPOS,BYTPOS,Po,To,yerr,yterr
C
C     first determine the order of the data i.e. y,pressure,temperature
C     data columns, then convert voltages to physical quantities
C
      j(1) = norder/100
      j(2) = (norder - j(1)*100)/10
      j(3) = norder - j(1)*100 - j(2)*10
      do 5 i=1,3
          if (j(i) .eq. 1) jy = i
          if (j(i) .eq. 2) jp = i
          if (j(i) .eq. 3) jt = i
    5 continue
      if (j(1) .eq. 0) then
          jy = jy-1
          jp = jp-1
          jt = 3
      end if
C
C     Convert y-position voltages in primary data file (contains
C     pressure data and possibly temperature)
C
      YPOS(1) = 0.0
      do 10 i=1,ndata
            if (FILL .eq. 0.0) then
              YPOS(i) = V(jy,i)*MYPOS+BYPOS
            else
              YPOS(i) = (i-1)*FILL
            endif
   10       YPOS(i) = YPOS(i) + YERR
C           if (ypos(i) .lt. 0.) ypos(i) = 0.
C
C     Construct temperature data y-positions (ytpos) if temperature data
C     is in the primary data file. Note ndata = ntdata.
C
      if (j(1) .gt. 0) then
```

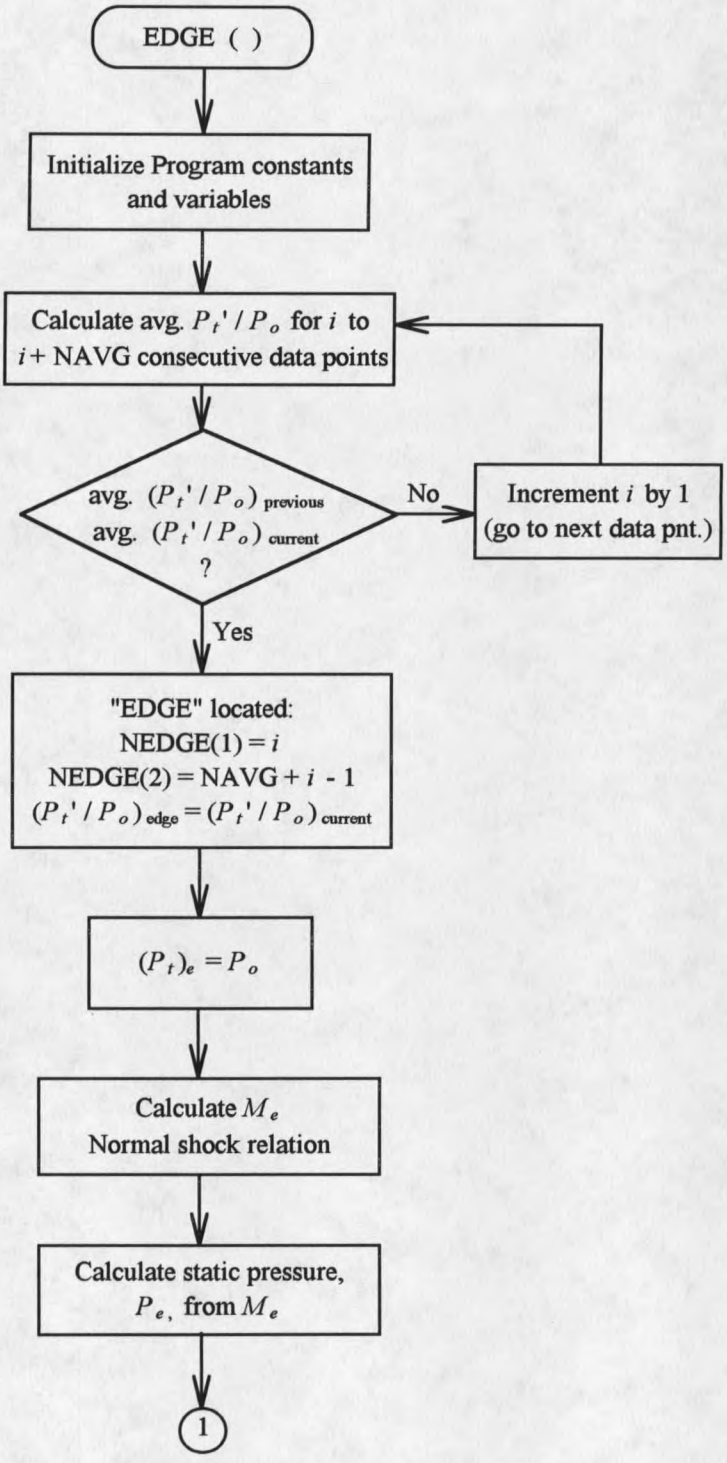
```
            jtt = 1
            do 15 i=1,ndata
                VT(jtt,i) = V(jt,i)
   15           ytpos(i) = ypos(i) - ypos(ntoff)
        end if
C
C       Convert data from a secondary data file (containing ytpos() and
C       temperature data) if necessary
C
        if ((j(1) .eq. 0) .and. (ntorder .gt. 0)) then
C
C         Pressure and temperature data must be in separate arrays
C
          j(1) = ntorder/10
          j(2) = ntorder-j(1)*10
          do 20 i=1,2
                if (j(i) .eq. 1) jyt=i
                if (j(i) .eq. 2) jtt=i
   20     continue
C
C         Construct y- position arrays for temperature and pressure data
C
          do 30 i=1,ntdata
              if (fillt .eq. 0.0) then
                  ytpos(i) = VT(jyt,i)*MYTPOS + BYTPOS + yterr
              else
                  ytpos(i) = (i-1)*FILLT + yterr
              end if
   30     continue
          ytoff = ytpos(ntoff)
          do 35 i=1,ntdata
   35         ytpos(i) = ytpos(i) - ytoff
        end if
C
C       Now map temperature data onto pressure data points
C
        call mapt2p(ndata,jstart,ntdata,ytpos,ypos,V,VT,jt,jtt,ntoff)
C
        do 40 i=1,ndata
            PTP(i) = (V(jp,i)*MPTP+BPTP)/Po
            V(jt,i) = V(jt,i)*MTTP+BTTP
   40       TTP(i) = (273 + THERMOJ(V(jt,i)))/To
        do 50 i=1,ntoff
            VT(jtt,i) = VT(jtt,i)*MTTP+BTTP
            TTPW(1,i) = ytpos(i)
   50       TTPW(2,i) = (273 + THERMOJ(VT(jtt,i)))/To
        return
        end
```

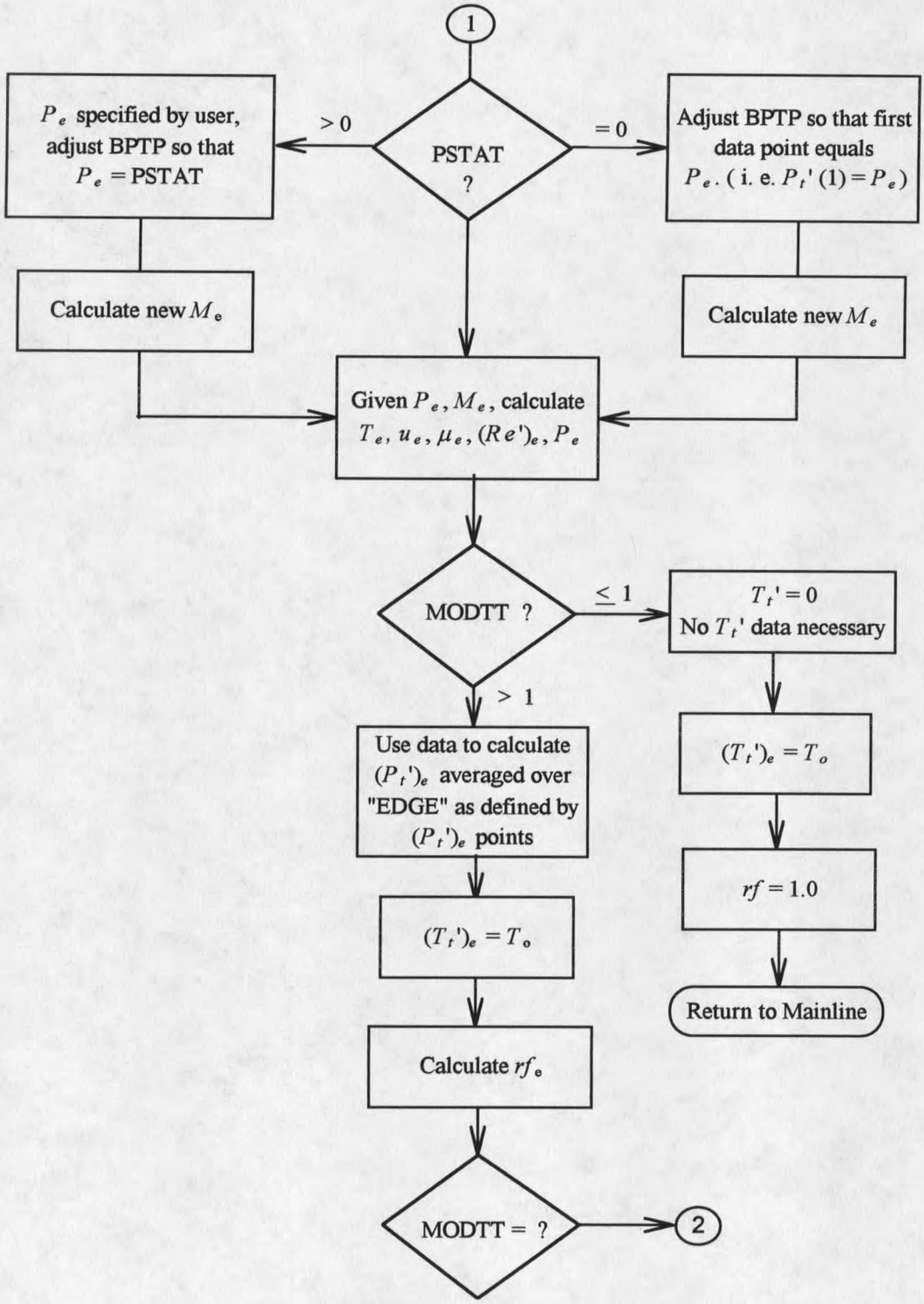

MAPT2P ( )
End of P_t' data ?
Yes
Return to VCONVERT ( )
1
No
End of T_t' data ?
Yes
V @ y_P - pos. = V @ y_T - pos.
1
No
Locate y_T - pos. points surrounding current y_P - pos.
Calculate V @ y_P - pos. by linear interpolation
1

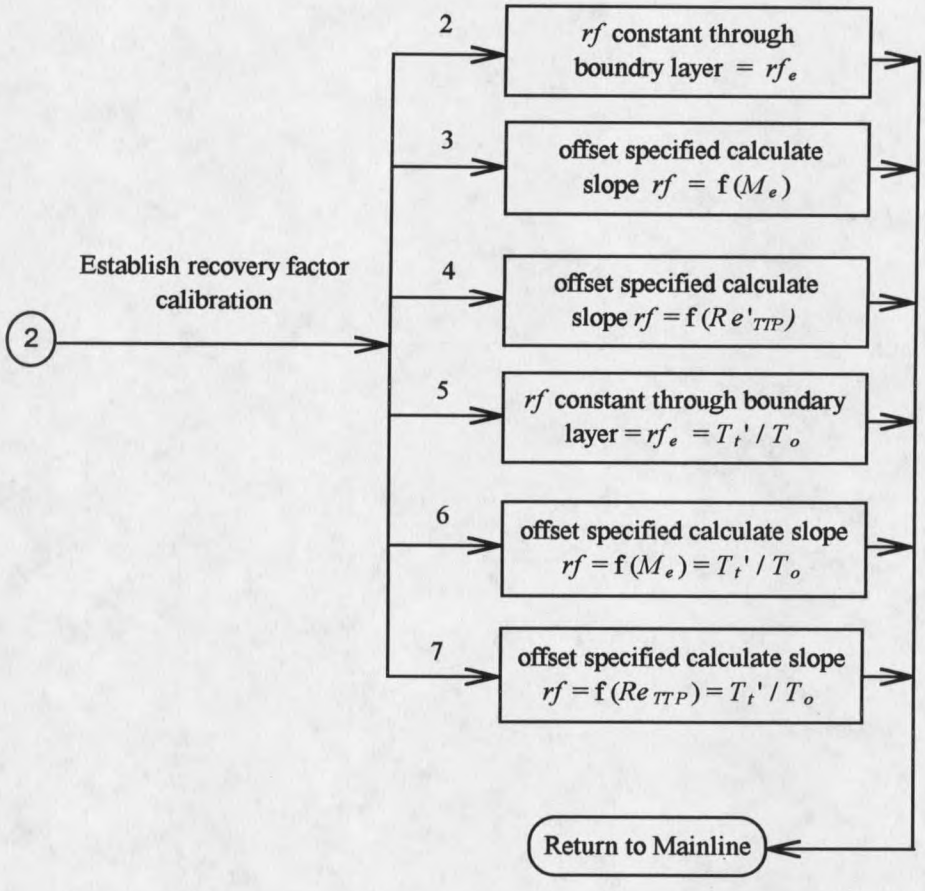
```
C*********************************************************************
      subroutine mapt2p(ndata,jstart,ntdata,ytpos,ypos,V,VT,jt,jtt,
     &   ntoff)
C*********************************************************************
C
C     Subroutine MAPT2P() maps temperature data onto the pressure data
C     points by linear interpolation.
C
      integer ndata, ntdata,jstart,jt,jtt
      real V(3,*),VT(2,*),ytpos(*),ypos(*)
C
C     Hunt for the corresponding ytpos() and ypos() values.  Assume both
C     arrays are in ascending order.
C
      jstart = 2
      do 10 i=1,ndata
        if (ypos(i) .lt. ytpos(ntdata)) then
          do 20 j=jstart,ntdata
              if (ytpos(j) .gt. ypos(i)) then
                  V(jt,i) = VT(jtt,j-1) + (VT(jtt,j)-VT(jtt,j-1))*
     &             (ypos(i)-ytpos(j-1))/(ytpos(j)-ytpos(j-1))
                  goto 10
              end if
   20     continue
        else
          V(jt,i) = VT(jtt,ntdata)
        end if
   10 jstart=j-1
C
      return
      end
```




```
C**********************************************************************
      subroutine ref(Po, To, gamma, UReo,soundo,muo,denso)
C**********************************************************************
C
C     Subroutine REF() computes the reference quantities in the
C     stagnation chamber.
C
      real Po, To, UReo, soundo, Rair, gamma, muo, denso, visc0
      Rair = 287.00
      visc0 = 1.717e-5
      denso = Po*133.3224/(To*Rair)
      muo = viscos(To/273.15,273.15)*visc0
      soundo = sqrt(gamma*Rair*To)
      UReo = denso*soundo/muo
      return
      end
```


EDGE ( )
Initialize Program constants and variables
Calculate avg. P_t' / P_o for i to i + NAVG consecutive data points
avg. $(P_t' / P_o)_{previous}$
avg. $(P_t' / P_o)_{current}$
?
No
Increment i by 1 (go to next data pnt.)
Yes
"EDGE" located:
NEDGE(1) = i
NEDGE(2) = NAVG + i - 1
$(P_t' / P_o)_{edge} = (P_t' / P_o)_{current}$
$(P_t)_e = P_o$
Calculate M_e Normal shock relation
Calculate static pressure, P_e, from M_e
1


1
PSTAT ?
> 0
= 0
P_e specified by user, adjust BPTP so that P_e = PSTAT
Adjust BPTP so that first data point equals P_e. (i. e. $P_t'(1) = P_e$)
Calculate new M_e
Calculate new M_e
Given P_e, M_e, calculate $T_e, u_e, \mu_e, (Re')_e, P_e$
MODTT ?
≤ 1
$T_t' = 0$ No T_t' data necessary
> 1
Use data to calculate $(P_t')_e$ averaged over "EDGE" as defined by $(P_t')_e$ points
$(T_t')_e = T_o$
$rf = 1.0$
$(T_t')_e = T_o$
Return to Mainline
Calculate rf_o
MODTT = ?
2

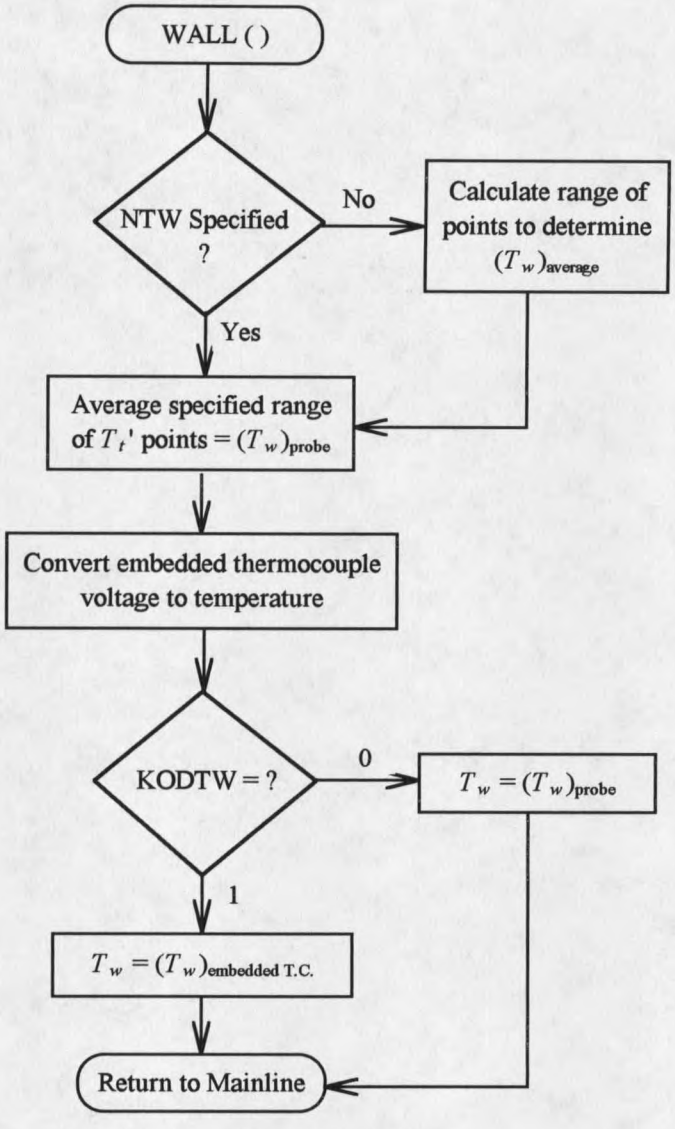

2
rf constant through boundry layer = rf_e
3
offset specified calculate slope $rf = \mathrm{f}(M_e)$
Establish recovery factor calibration
2
4
offset specified calculate slope $rf = \mathrm{f}(Re'_{TTP})$
5
rf constant through boundary layer = $rf_e = T_t' / T_o$
6
offset specified calculate slope $rf = \mathrm{f}(M_e) = T_t' / T_o$
7
offset specified calculate slope $rf = \mathrm{f}(Re_{TTP}) = T_t' / T_o$
Return to Mainline

```
************************************************************************
      Subroutine EDGE(PTP,TTP,ME,PE,TE,TTE,DENSE,ndata,nedge,
     &   gamma,navg,UE,MUE,RFACTE,To,PTPE,modtt,mrf,brf,ttpe,rfe,URee,
     &   Reettp,UReo,http,Pstat,BPTP,Po,Pshift)
************************************************************************
C
C     Subroutine EDGE() computes the edge quantities for use through
C     the rest of the program.  By default, EDGE() locates the boundary
C     layer edge by a NAVG point moving average.  The user can also
C     specify the range of points which define the boundary layer edge.
C
      integer   nedge(2),
     & ndata, navg, nstart, max
C
      real PTP(ndata), TTP(*),
     &    PE,  PTPEN,    PTPE,   TTE,   TE,  ME, Mhisq,  Mlosq,   Msq,
     & accur,      A,       B,     C,    D,   E,     F,  gamma, DENSE,
     &    UE,    MUE,  RFACTE,  TTPE,  mrf, brf,   rfe,   URee, Reettp,
     &  http,   UReo,   Pstat,  BPTP, beta, MachPTPsq, Po
C
C     This subroutine EDGE() assumes a smooth profile in both pressure
c     and temperature data.  The data goes from y = 0 (wall) to the
C     free-stream.
C     Determine "edge" location as the location of maximum pitot
C     pressure (if the edge data points are not specified by input).
C     Note that this criteria is only valid if the thermal boundary
C     layer is smaller than or the same thickness as the velocity
C     boundary layer.
C
      nstart = 20
      PTPE = 0.0
C
      if ((nedge(1) .eq. -1) .and. (nedge(2) .eq. -1)) then
        do 10 i=nstart,ndata-navg
          PTPEn = 0.0
          do 20 j=0,navg-1
   20       PTPEn = PTPEn+PTP(i+j)
          PTPEn = PTPEn/navg
          if (PTPEn .lt. PTPE) goto 30
   10     PTPE = PTPEn
   30   nedge(1) = i-1
        nedge(2) = i+navg-2
      else
        navg = nedge(2)-nedge(1)+1
        do 40 j=0,navg-1
   40     PTPE = PTPE + PTP(nedge(1)+j)
        PTPE = PTPE/navg
      endif
C
C     Determine edge mach number.  The following equation is for
```

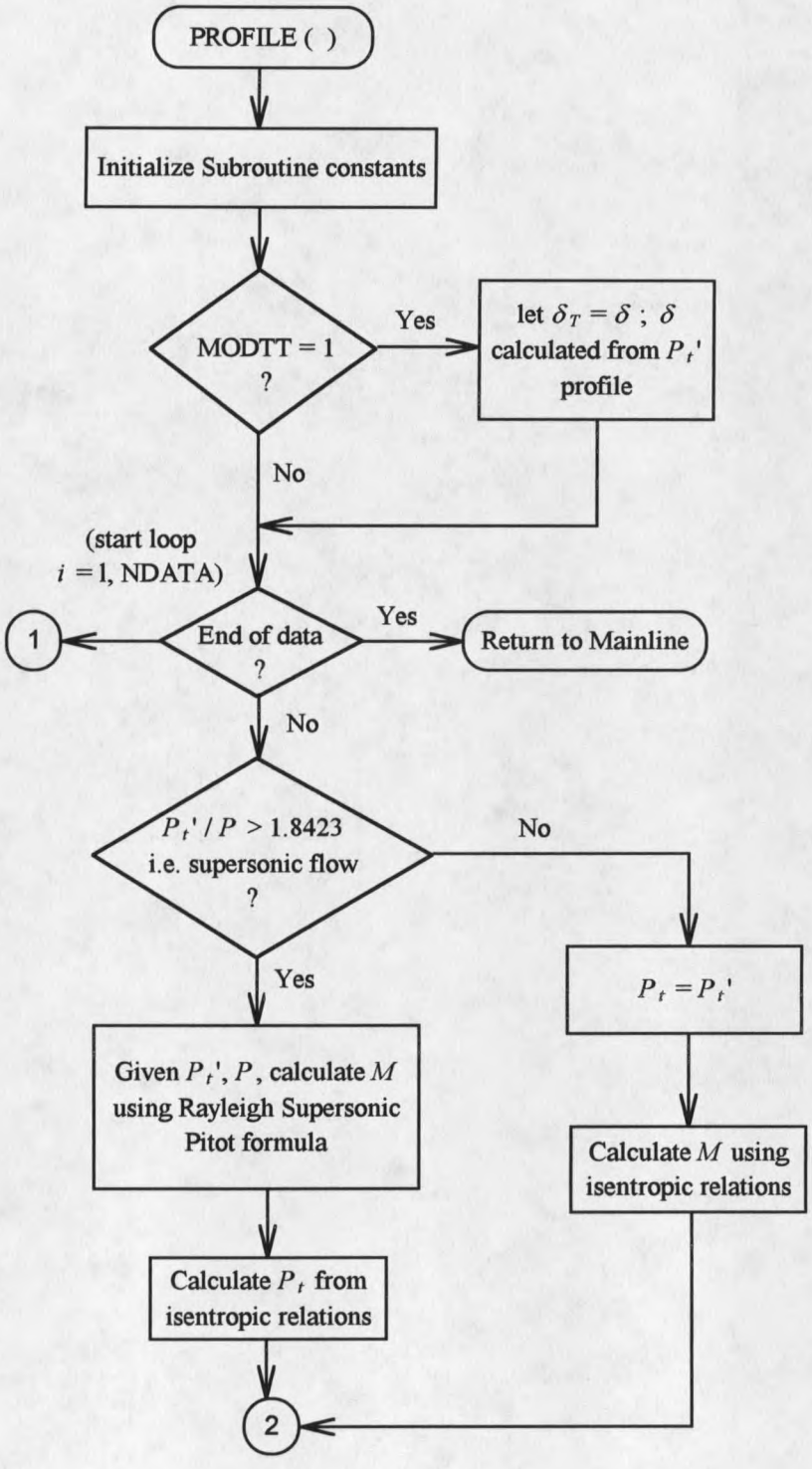
```
C     determining the mach number from a pitot measurement knowing
C     the tunnel stagnation pressure.  Reference:  Eqn 6-1, pg. 148,
C     Liepmann & Roshko, Elements of Gas Dynamics, copyright 1957,
C     John Wiley & Sons.
C
      A=2.0*gamma/(gamma+1.0)
      B=(gamma-1.0)/(gamma+1.0)
      C=(gamma-1.0)/2.0
      D=(gamma+1)/2.0
      E=1.0/(1.0-gamma)
      F=gamma/(1.0-gamma)
C
      accur = 1.e-6
      max = 30
      Mhisq = 16.0
      Mlosq = 1.001

      Msq = MachPTPsq(PTPE,Mhisq,Mlosq,A,B,C,D,E,F,accur,max)
      PE = (1.0+C*Msq)**F
C
C     By default (Pstat < 0) the static pressure is taken as the
C     measurement in the free-stream and no correction is made to the
C     Encapsulated Transducer output.
C
      If (Pstat .eq. 0) then
C
C         Static Pressure is taken as the wall measurment.  Calibration
C         offset is adjusted accordingly.
C
          PTP1 = PTP(1)
          do 60 i=1,10
              diff = PE-PTP1
              If (abs(diff) .lt. 1.e-4) goto 70
              PTPE=PTPE+diff
              PTP1=PTP1+diff
              Mhisq = 12.3
              Mlosq = 2.0
              Msq = MachPTPsq(PTPE,Mhisq,Mlosq,A,B,C,D,E,F,accur,max)
60            PE = (1.0+C*Msq)**F
70        continue
          Pshift = PTP1-PTP(1)
          BPTP = BPTP+Pshift*Po
          do 80 i=1,ndata
80            PTP(i) = PTP(i)+Pshift
      else if (Pstat .gt. 0) then
C
C         Static Pressure is input by user. Calibration is offset
C         to force free-stream measurement equal to static pressure
C         accordingly.
C
          PE1 = PE
```

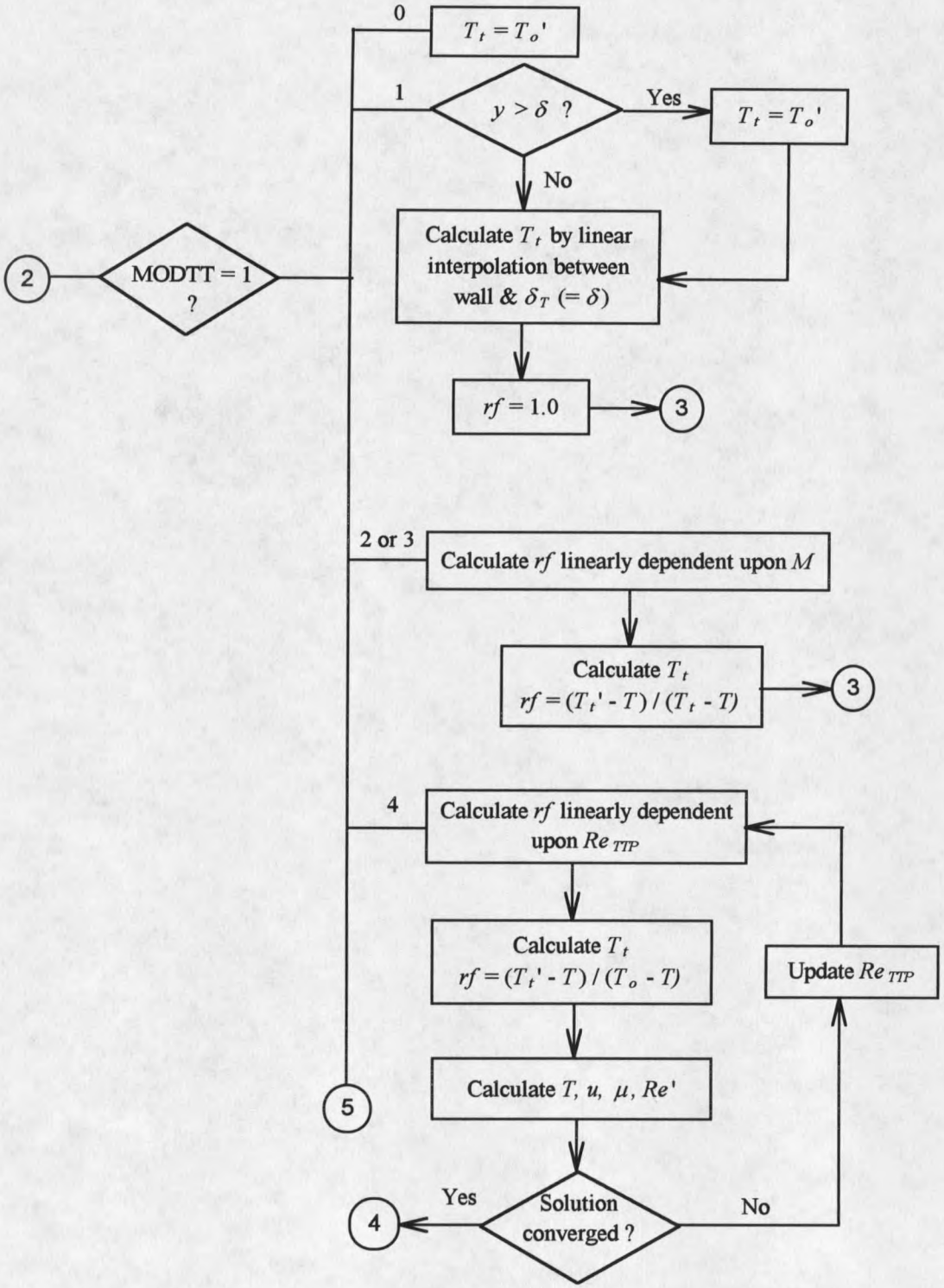
```
            beta = 2.5
C               beta is a weighting factor to control convergence.
            do 90 i=1,10
                diff = Pstat-PE
                If (abs(diff) .lt. 1.e-4) goto 100
                PTPE=PTPE+beta*diff
                Mhisq = 12.3
                Mlosq = 2.0
                Msq = MachPTPsq(PTPE,Mhisq,Mlosq,A,B,C,D,E,F,accur,max)
   90           PE = (1.0+C*Msq)**F
  100       continue
            Pshift = PE1-PTP(1)
            BPTP = BPTP+Pshift*Po
            do 110 i=1,ndata
  110           PTP(i) = PTP(i)+Pshift
        end if
C
C       Calculate other edge conditions.  Note that from boundary
C       layer theory this static pressure, PE, is constant through the
C       boundary layer.  Reference:  Isentropic relations.
C
        ME = sqrt(Msq)
        DENSE = PE**(1.0/GAMMA)
        TE = 1.0/(1.0+C*Msq)
        UE = ME*sqrt(TE)
        mue = viscos(TE,To)
        URee = (dense*ue/mue)
        Reettp = URee*UReo*http
C
C       Calculate total temperature
C
        if (modtt .le. 1) then
            TTE = 1.0
            TTPE = 0.0
            rfacte = 1.0
        else
            TTPE = 0.0
            do 50 j=0,navg-1
   50         TTPE = TTPE + TTP(nedge(1)+j)
            TTPE = TTPE/navg
            rfacte = rfact(TTPE,1.0,TE)

            rfe = rfacte
            TTE = 1.0
            if (modtt .eq. 2) then
                mrf = 0.0
                brf = rfe
            end if
            if (modtt .eq. 3) mrf = (rfe-brf)/ME
            if (modtt .eq. 4) mrf = (rfe-brf)/Reettp
            if (modtt .eq. 5) then
```

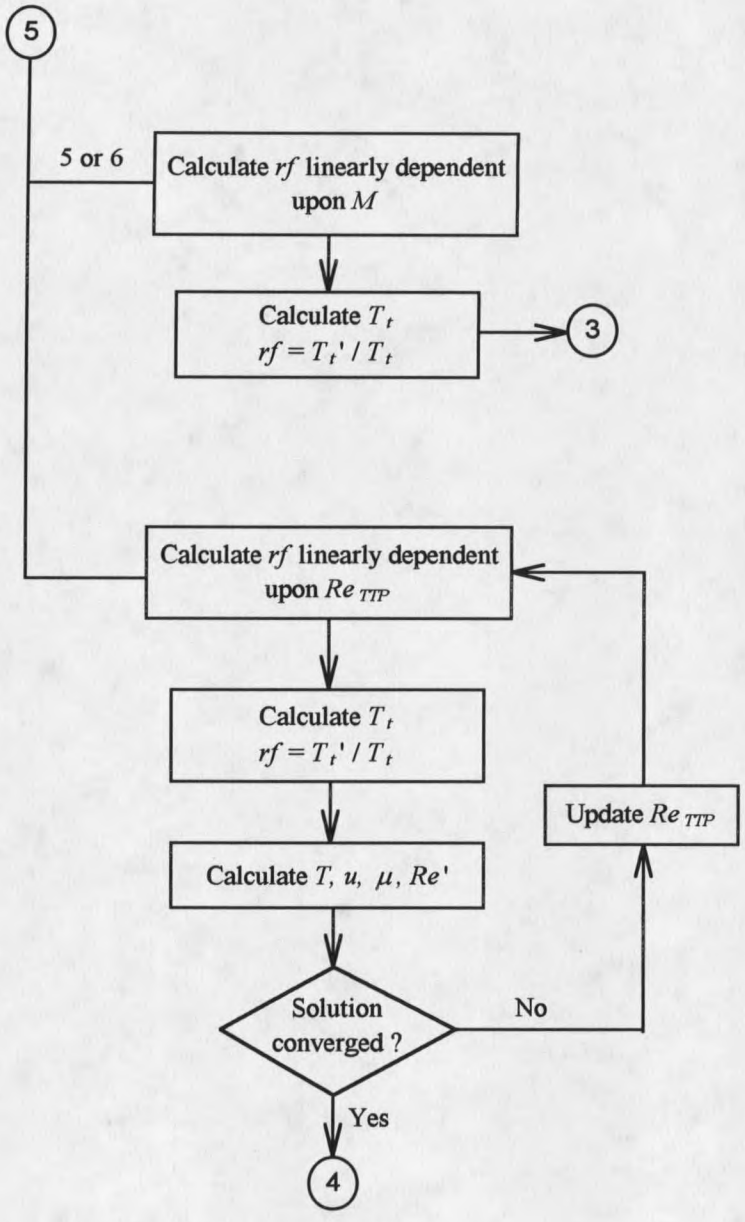
```
            mrf = 0.0
            brf = ttpe
        end if
        if (modtt .eq. 6) mrf = (ttpe-brf)/ME
        if (modtt .eq. 7) mrf = (ttpe-brf)/Reettp
    end if
C
    return
    end
```

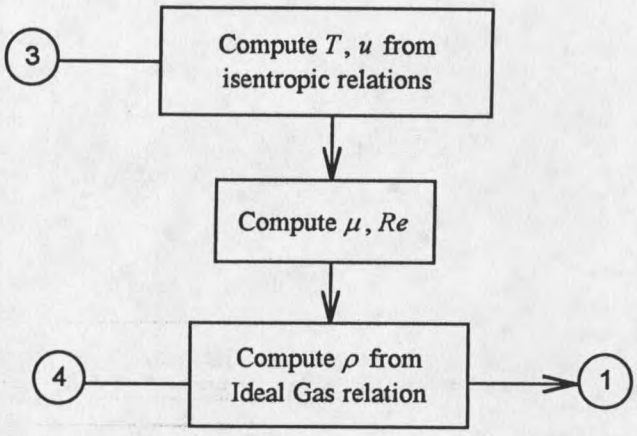

WALL ( )
NTW Specified ?
No
Calculate range of points to determine $(T_w)_{average}$
Yes
Average specified range of T_t' points = $(T_w)_{probe}$
Convert embedded thermocouple voltage to temperature
KODTW = ?
0
$T_w = (T_w)_{probe}$
1
$T_w = (T_w)_{embedded\ T.C.}$
Return to Mainline

```
C*********************************************************************
      subroutine wall(TTPW,vtwloc,ntwavg,ntw,Twp,Twt,ndata,To,kodtw,Tw)
C*********************************************************************
C
C     Computes wall values.  Currently, only surface temperature values.
C
      integer ndata,ntwavg,ntw(2),kodtw
      real TTPW(2,ndata),vtwloc,Twp,Twt,To,Tw
C
      if ((ntw(1) .gt. 0.0) .and. (ntw(2) .gt. 0.0)) then
          ntwavg = ntw(2) - ntw(1)
      else
          ntw(1) = 1
          ntw(2) = 1+ntwavg
      end if
C
      Twp = 0.0
      do 10 j=0,ntwavg-1
   10   Twp = Twp + TTPW(2,ntw(1)+j)
      Twp = Twp/ntwavg
C
      Twt = (Thermoj(vtwloc)+273.15)/To
C
      if (kodtw .eq. 0) Tw = Twp
      if (kodtw .eq. 1) Tw = Twt
      return
      end
```


PROFILE ( )
Initialize Subroutine constants
MODTT = 1 ?
Yes
let $\delta_T = \delta$; δ calculated from P_t' profile
No
(start loop i = 1, NDATA)
1
End of data ?
Yes
Return to Mainline
No
$P_t' / P > 1.8423$ i.e. supersonic flow ?
No
Yes
$P_t = P_t'$
Given P_t', P, calculate M using Rayleigh Supersonic Pitot formula
Calculate M using isentropic relations
Calculate P_t from isentropic relations
2


0
$T_t = T_o'$
1
$y > \delta$?
Yes
$T_t = T_o'$
No
Calculate T_t by linear interpolation between wall & δ_T (= δ)
2
MODTT = 1 ?
rf = 1.0
3
2 or 3
Calculate rf linearly dependent upon M
Calculate T_t
$rf = (T_t' - T) / (T_t - T)$
3
4
Calculate rf linearly dependent upon Re_{TTP}
Calculate T_t
$rf = (T_t' - T) / (T_o - T)$
Update Re_{TTP}
Calculate T, u, μ, Re'
5
Yes
Solution converged ?
No
4


5
5 or 6
Calculate rf linearly dependent upon M
Calculate T_t
$rf = T_t' / T_t$
3
Calculate rf linearly dependent upon Re_{TTP}
Calculate T_t
$rf = T_t' / T_t$
Calculate T, u, μ, Re'
Update Re_{TTP}
Solution converged ?
No
Yes
4

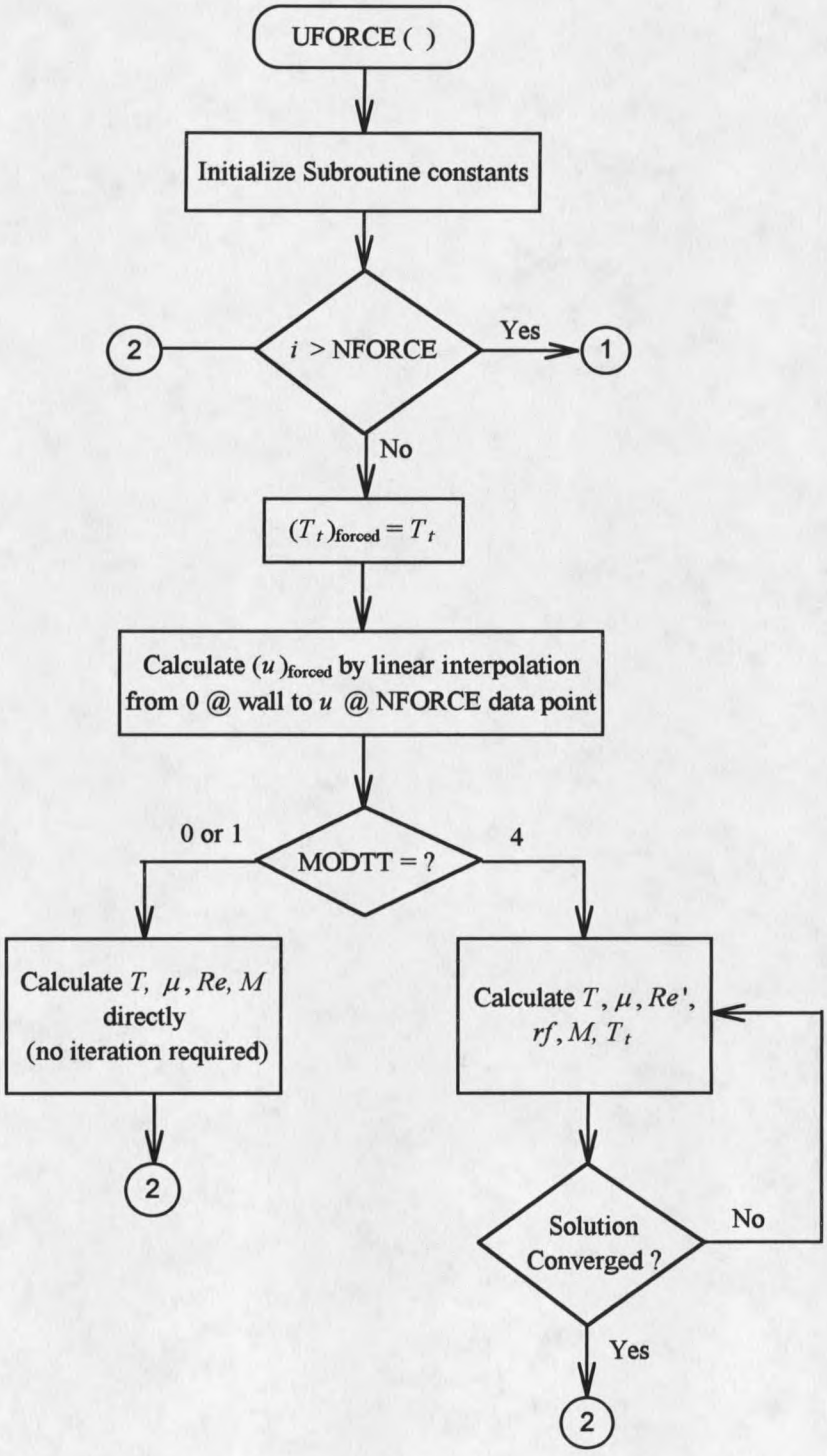

3
Compute T, u from isentropic relations
Compute μ, Re
4
Compute ρ from Ideal Gas relation
1

```
C*********************************************************************
      subroutine profile(YPOS,PTP,TTP,ME,PE,PTPE,nedge,
     &     Pt,Ma,TT,T,U,DENS,MU,modtt,Tw,TE,ndata,ble,rf,mrf,brf,URe,
     &     Reettp)
C*********************************************************************
C
C     The subroutine profile() calculates the profile quantities of a
C     data set constisting of y-, Pitot pressure, and optional Total
C     temperature values.  The routine uses either a linear fit
C     between the wall and edge total temperature or Total temperature
C     can be calculated using total temperature probe data and a
C     recovery factor data as a function of Re' and M.
C
      integer modtt,ndata,nedge(2)
      real ypos(ndata), PTP(ndata), ttp(ndata), dens(ndata), Pt(ndata),
     &        U(ndata),  MA(ndata),  TT(ndata),    T(ndata), mu(ndata),
     &       rf(ndata), URe(ndata),
     &    ME,     PE,     TE,  PTPE, gamma,  mrf,  brf, Reettp,
     &    To,     Po,  PTPI, Mhisq, Mlosq,  Msq,   Tw,     PO2, mblesq
      COMMON /REFRNC/ Po, To, gamma
C
      A=2.0*gamma/(gamma+1.0)
      B=(gamma-1.0)/(gamma+1.0)
      C=(gamma-1.0)/2.0
      D=(gamma+1)/2.0
      E=1.0/(1.0-gamma)
      F=gamma/(1.0-gamma)
      accur = 1.e-6
      max = 30
      rmrf = mrf*Reettp
      URe1 = 0.0
C
C     If a linear total temperature model is used (modtt = 1) calculate
C     the thermal boundary layer thickness from pitot pressure profile
C     use velocity boundary layer thickness = thermal boundary layer
C     thickness.
C
      if (modtt .eq. 1) then
          blesq = ble*ble
          mblesq = 1./(((1.-blesq)/blesq)*(1/TE) + 1.)
          tble = PTPE*((A*ME*ME-B)/(A*mblesq*Me*Me-B))**2.5*mblesq**3.5
          delta = d1(ptp,ypos,tble,ndata,nedge(1))
      end if
C
      do 10 i=1,ndata
C
C     Determine mach number and total pressure using the Rayleigh
C     Supersonic Pitot formula for supersonic flow and the isentropic
C     relation with Pt = Ptp for subsonic flow.
C     Reference:  pg. 149, Liepmann & Roshko, Elements of Gas
```

```
C     Dynamics, copyright 1957, John Wiley & Sons.
C
      PTPI = PTP(i)/PE
      if (PTPI .gt. 1.8923) then
         Mhisq=16.0
         Mlosq=1.0001
C
         do 60 j=1,max
              Msq=(Mhisq+Mlosq)/2.0
              Po2 = (A*Msq-B)**E / (D*Msq)**F
              If (abs(1.0 - Po2/PTPI) .lt. accur) goto 70
              if (po2 .gt. ptpi) Mhisq = Msq
              if (po2 .lt. ptpi) Mlosq = Msq
   60    continue
         write(*,9999) 'WARNING:  SOLUTION FOR ME NOT CONVERGED AFTER ',
     &           J,' ITERATIONS.', i,sqrt(Mlosq),sqrt(Mhisq),
     &           sqrt(Msq),accur*100.0
   70    MA(i) = sqrt(Msq)/ME
C
C        Calculate static pressure & temperature.  Note that from
C        boundary layer theory this static pressure, PE, is constant
C        through the boundary layer.  Reference:  Isentropic relations
C
         PT(i) = (1.0+C*Msq)**(-F)*PE
      else
         PT(i) = PTPI*PE
         if(PTPI .lt. 1.0) PTPI = 1.0
         Msq = (PTPI**(-1.0/F)-1.0)/C
         MA(i) = sqrt(Msq)/ME
      end if
C
C     Now calculate total temperature, static temperature and velocity
C     using a linear model or temperature data with recovery factor
C     data.
C
      if (modtt .eq. 0) then
C
          TT(i) = 1.0
C
      elseif (modtt .eq. 1) then
C
         if (ypos(i) .lt. delta) then
           TT(i)=Tw+(1.0-Tw)*YPOS(i)/delta
         elseif (ypos(i) .ge. delta) then
           TT(i) = 1.0
         end if
         rf(i) = 1.0
C
      elseif ((modtt .eq. 2).or.(modtt .eq. 3)) then
C
C        use a recovery factor (linearly dependent upon Mach #) and
```

```
C        temperature probe data to determine total temperature, velocity
C        and static temperature.
C
         rf(i) = mrf*MA(i)*ME + brf
         TT(i) = TTP(i)*(1+C*Msq)/(rf(i)*C*Msq+1)
C
      elseif (modtt .eq. 4) then
C
C        use a recovery factor (linearly dependent upon URee) and
C        temperature probe data to determine total temperature, velocity
C        and static temperature.
C
         do 80 j = 1,max
             rf(i) = rmrf*URe1 + brf
             TT(i) = TTP(i)*(1+C*Msq)/(rf(i)*C*Msq+1)
             T(i) = TT(i)/(1+C*Msq)/TE
             u(i) = MA(i)*sqrt(T(i))
             mu(i) = viscos(T(i),TE*To)
             URe(i) = u(i)/mu(i)/T(i)
             if (abs(URe(i)-URe1) .le. URe1*accur) goto 10
   80        URe1 = URe(i)
C
      elseif ((modtt .eq. 5) .or. (modtt .eq. 6)) then
C
C         Identical to modtt .eq. 3 but rf is defined as Tt'/Tt
C
          rf(i) = mrf*MA(i)*ME + brf
          TT(i) = TTP(i)/rf(i)
C
      elseif (modtt .eq. 7) then
C
C         Identical to modtt .eq. 4 but rf is defined as Tt'/Tt
C
         do 90 j = 1,max
             rf(i) = rmrf*URe1 + brf
             TT(i) = TTP(i)/rf(i)
             T(i) = TT(i)/(1+C*Msq)/TE
             u(i) = MA(i)*sqrt(T(i))
             mu(i) = viscos(T(i),TE*To)
             URe(i) = u(i)/mu(i)/T(i)
             if (abs(URe(i)-URe1) .le. URe1*accur) goto 10
   90        URe1 = URe(i)
      end if
C
      T(i) = TT(i)/(1+C*Msq)/TE
      u(i) = MA(i)*sqrt(T(i))
      mu(i) = viscos(T(i),TE*To)
      URe(i) = u(i)/mu(i)/T(i)
   10 dens(i) = 1.0/T(i)
C
C
```

```
C      format statements
C
 9999 FORMAT(1x,A,i2,A,/1x,'CALCULATING M FOR DATA POINT ',I3,/1x,
     &     'LOWER BOUND =',1pe10.3,/1x,'UPPER BOUND =',1pe10.3,/1x,
     &     'M =',1pe10.3,/1x,'% ACCURACY =',1pe10.3)
C
      return
      end
```


UFORCE ( )
Initialize Subroutine constants
2
i > NFORCE
Yes
1
No
$(T_t)_{\text{forced}} = T_t$
Calculate $(u)_{\text{forced}}$ by linear interpolation from 0 @ wall to u @ NFORCE data point
0 or 1
MODTT = ?
4
Calculate T, μ, Re, M directly (no iteration required)
2
Calculate T, μ, Re', rf, M, T_t
Solution Converged ?
No
Yes
2

```
C*********************************************************************
      subroutine uforce(YPOS,U,Uf,Maf,Ptf,DENSf,Tf,PE,ME,TE,TTE,
     &    TT,TTF,rff,muf,gamma,nforce,ndata,To,fitdesc,URef,mrf,brf,
     &    Reettp,TTP,modtt)
C*********************************************************************
C
C     Subroutine UFORCE() forces the data to satisfy Prandtl boundary
C     condition.  Velocities are determined by a linear profile between
C     data point NFORCE and the wall.  All other quantities are computed
C     using the velocity, Total temperature (as measured by probe) and
C     static pressure as the independent variables.
C
      integer ndata,nforce, max,modtt
C
      real u(ndata), uf(ndata), densf(ndata), Maf(ndata),  Ptf(ndata),
     &  YPOS(ndata), Tf(ndata),    TT(ndata), MUf(ndata), URef(ndata),
     &    rff(ndata), TTf(ndata), TTP(ndata),
     &    ME,     PE, TE, TTE, slope, C, F, To, accur, gamma, rmrf, brf,
     & URef1, Reettp, mrf

      character fitdesc*30

      slope = u(nforce)/YPOS(nforce)
      C=(gamma-1.0)/2.0
      F=gamma/(gamma-1.0)
      accur = 1.e-6
      rmrf = mrf*Reettp
      URef1 = 0.
      max = 30

      do 10 i=1,nforce
          TTf(i) = TT(i)
          if (modtt.eq.4) then
C
C             Compute forced data -- note the dependence of the total
C             temperature on the velocity due to the recovery factor.
C
              do 80 j = 1,max
                  Uf(i) = YPOS(i)*slope
                  Tf(i) = ((TE/TTE - 1.0)*Uf(i)*Uf(i) + 1.0)*TTf(i)/TE
                  MUf(i) = viscos(Tf(i),TE*To)
                  URef(i) = Uf(i)/muf(i)/Tf(i)
                  rff(i) = rmrf*URef(i)+brf
                  Maf(i) = Uf(i)*sqrt(1.0/Tf(i))
                  TTf(i) = TTP(i)*(1+C*Maf(i)*Maf(i))/(rff(i)*C*Maf(i)*
     &                      Maf(i)+1)
c                  write(*,'(1x,A,/1x,I2,3(1x,1pe14.7))')
c     &' test: j,delta URef(i),TTf(i), accur =',
c     &j,URef(i)-URef1,TTf(i), accur
                  if (abs(URef(i)-URef1) .le. URef1*accur) goto 10
```

```
   80              URef1 = URef(i)
C
        elseif ((modtt .eq. 0) .or. (modtt .eq. 1)) then
C
C           Compute artificial data -- no total temperature dependence
C           upon the total temperature.
C
                Uf(i) = YPOS(i)*slope
                Tf(i) = ((TE/TTE - 1.0)*Uf(i)*Uf(i) + 1.0)*TTf(i)/TE
                MUf(i) = viscos(Tf(i),TE*To)
                URef(i) = Uf(i)/muf(i)/Tf(i)
                rff(i) = 1.0
                Maf(i) = Uf(i)*sqrt(1.0/Tf(i))

        end if
   10   DENSf(i) = 1/Tf(i)
        Ptf(i) = (1+C*(Maf(i)*ME)**2.0)**F*PE
      fitdesc = '(LINEAR FIT ON VELOCITY)'
      return
      end
```

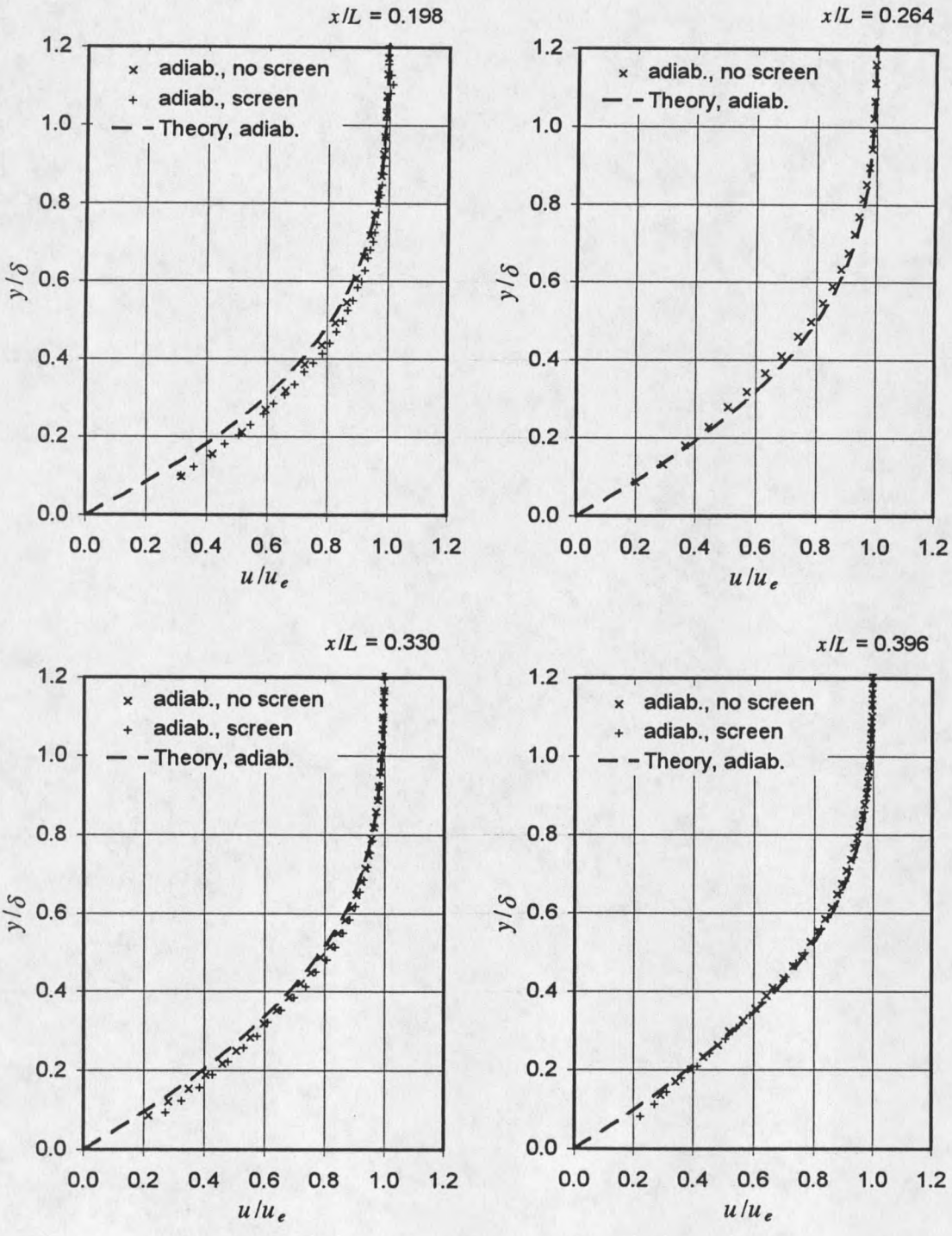
```
                    MINOR SUBROUTINES AND FUNCTIONS
C*********************************************************************
      subroutine ystretch(ytilda,dens,densw,ypos,ndata)
C*********************************************************************
C
C     Subroutine YSTRETCH() transforms the y-coordinate to a stretched
C     "incompressible" y-coordinate.  Trapezoidal rule is used for
C     integration.  REF:  Schlichting, Illingsworth-Stewartson
C     transformation.
C
      integer ndata
      real ytilda(*),dens(*),ypos(*),densw
      ytilda(1) = (dens(1)+densw)*ypos(1)/2
      do 10 i=2,ndata
   10     ytilda(i) = ytilda(i-1)+(ypos(i)-ypos(i-1))*
     &                             (dens(i)+dens(i-1))/2
      return
      end

C*********************************************************************
      subroutine fname(namec,nchar,pfnc)
C*********************************************************************
C
C     Subroutine FNAME() extracts the base filename from a filename
C     which includes the path.
C
      integer nchar
      character namec*1(80),pfnc*1(12)
      nstart = 1
      do 10 i=1,nchar
          if (namec(i) .eq. '\') nstart = i+1
          if (namec(i) .eq. ' ') goto 20
   10 continue
   20 nstop = i-1
      j = 1
      do 30 i = nstart,nstop
          pfnc(j) = namec(i)
   30     j=j+1
      return
      end

C*********************************************************************
      real function rfact(TTP,TT,T)
C*********************************************************************
C
C     Function RFACT computes the recovery factor given temperatures.
C
      real Ttp,Tt,T
      rfact = (TTp - T)/(TT - T)
      if(rfact .gt. 1.0) write(*,9991) rfact,ttp,tt,t
 9991 format(1x,'WARNING: TEMPERATURE PROBE RECOVERY FACTOR GREATER THA
```

```
     &N 1.0',1X/1X,'r_t = ',g10.3,3x,'TTP = ',g10.3,3x,'T = ',g10.3,3x,
     &'TT = ',g10.3,1x/1x,/1X,'POSSIBLY THE CALIBRATION CONSTANTS ARE',
     &' SET INCORRECTLY FOR THE TEMPERATURE'/1x,'PROBE CALIBRATION.')
      return
      end

C*********************************************************************
      real function d1(u,y,ble,ndata,nedge)
C*********************************************************************
C
C     Function D1() determines the boundary layer edge.
C
      integer ndata,nedge
      real u(ndata),y(ndata),ble
      DO 10 I=1,NEDGE
        IF(u(i) .gt. ble) then
          D1 = Y(I) - (Y(I)-Y(I-1))*(u(i)-ble)/(u(i)-u(i-1))
          return
        end if
   10 continue
      write (*,9999) 'WARNING: NO DELTA FOUND.','ATTEMPTED TO COMPUTE ',
     &'DELTA AT U/UE = ',BLE,'DELTA SET TO -1'
      D1 = -1
C
 9999 FORMAT (1X,A,/1X,A,A,F6.3,/1X,A)
C
      return
      end

C*********************************************************************
      real function d2(u,dens,y,ndata,nedge)
C*********************************************************************
C
C     function D2() computes the displacement thickness.  Trapezoidal
C     rule is used for integration.
C
      integer ndata
      real u(ndata),y(ndata),dens(ndata)
      d2 = (1-u(1)*dens(1)/2.0)*y(1)
      do 10 i=2,NEDGE
          rhou = (u(i)*dens(i) + u(i-1)*dens(i-1))/2.0
   10     d2 = d2+(1-rhou)*(y(i)-y(i-1))
      return
      end

C*********************************************************************
      real function d3(u,dens,y,ndata,nedge)
C*********************************************************************
C
C     Computes the momentum thickness of a boundary layer profile.
C     Trapezoidal rule is used for integration.
```

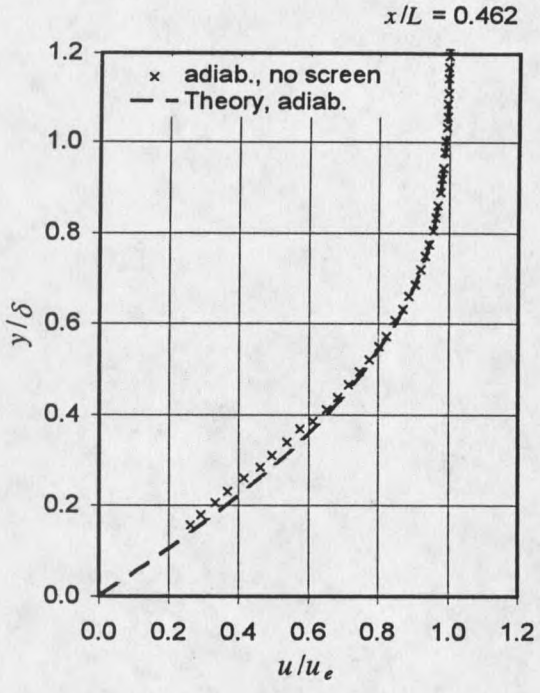
```
C
      integer ndata,nedge
      real u(ndata),y(ndata),dens(ndata)
      d3= (1-u(1))*dens(1)*u(1)*y(1)/2.0
      do 10 i=2,NEDGE
          rhouu = (u(i)*u(i)*dens(i) + u(i-1)*u(i-1)*dens(i-1))/2.0
          rhou = (u(i)*dens(i) + u(i-1)*dens(i-1))/2.0
   10     d3 = d3+(rhou-rhouu)*(y(i)-y(i-1))
      return
      end

C*********************************************************************
      real function MachPTPsq(PTPE,Mhisq,Mlosq,A,B,C,D,E,F,accur,max)
C*********************************************************************
C
C     Function MachPTPsq() computes the square of the Mach number given
C     the pitot pressure over the Stagnation pressure.  Iteration is
C     performed using the bisection method to converge.
C
      integer max
      real PTPE,Mhisq,Mlosq,A,B,C,D,E,F,accur, Po2, Msq
C
      do 60 i=1,max
          Msq=(Mhisq+Mlosq)/2.0
          Po2 = (A*Msq-B)**E * ((1+C*Msq)/(D*Msq))**F
          If (abs(1.0 - Po2/PTPE) .lt. accur) goto 70
          if (po2 .lt. PTPE) Mhisq = Msq
          if (po2 .gt. PTPE) Mlosq = Msq
   60 continue
      write(*,9999) 'WARNING:  SOLUTION FOR ME NOT CONVERGED AFTER ',I,
     &            ' ITERATIONS.', sqrt(Mlosq),sqrt(Mhisq),sqrt(Msq),
     &            accur*100
   70 MachPTPsq = Msq
      return
C
 9999 FORMAT(1x,A,i2,A,/1x,'LOWER BOUND =',1pe10.3,/1x,'UPPER BOUND =',
     &    1pe10.3,/1x,'ME =',1pe10.3,/1x,'% ACCURACY =',1pe10.3)
C
      end

C*********************************************************************
      real function THERMOJ(V)
C*********************************************************************
C
C     The function THERMOJ() calculates the temperature for a J-type
C     thermocouple given the measured voltage, V, and a reference
C     temperature of 0 degrees Celcius.
C
C     The thermocouple function is programmed in nested polynomial form
C     in order to reduce computation time.
C
```

```
      real V,a0,a1,a2,a3,a4,a5
C
C     Values for the polynomial coefficients a0 - a5 are values
C     for a Type J (Iron-Constantan) valid for the temperature range
C     0-760 deg. C.  The polynomial is reported +/- 0.1 deg. C from the
C     ANSI standard tabulated values.  Values obtained from Omega, "The
C     Temperature Handbook," vol. 28, copyright 1992, Omega Engineering,
C     Inc.
C
      DATA a0/-0.048868252/,a1/19873.14503/,a2/-218614.5353/,
     &     a3/ 11569199.78/,a4/-264917531.4/,a5/2018441314./
      THERMOJ = a0+V*(a1+V*(a2+V*(a3+V*(a4 +V*a5))))
      return
      end
C*********************************************************************
      real function viscos(T,Tref)
C*********************************************************************
C
C     Function viscos computes the dynamic viscosity normalized with a
C     reference viscosity (viscosity at Tref) according to Sutherlands
C     Viscosity Law.  Value for S1 came from Schlichting, Boundary Layer
C     Theory, pg 328.   S1 is in deg. Kelvin.  Also note that T is
C     already normalized with Tref.  viscos() produces the dimensional
C     viscosity normalized by the viscosity at Tref.
C
      real T, Tref, S1
      S1 = 110.3
      viscos = T**1.5*(1.0 + S1/Tref)/(T + S1/Tref)
      return
      end
```

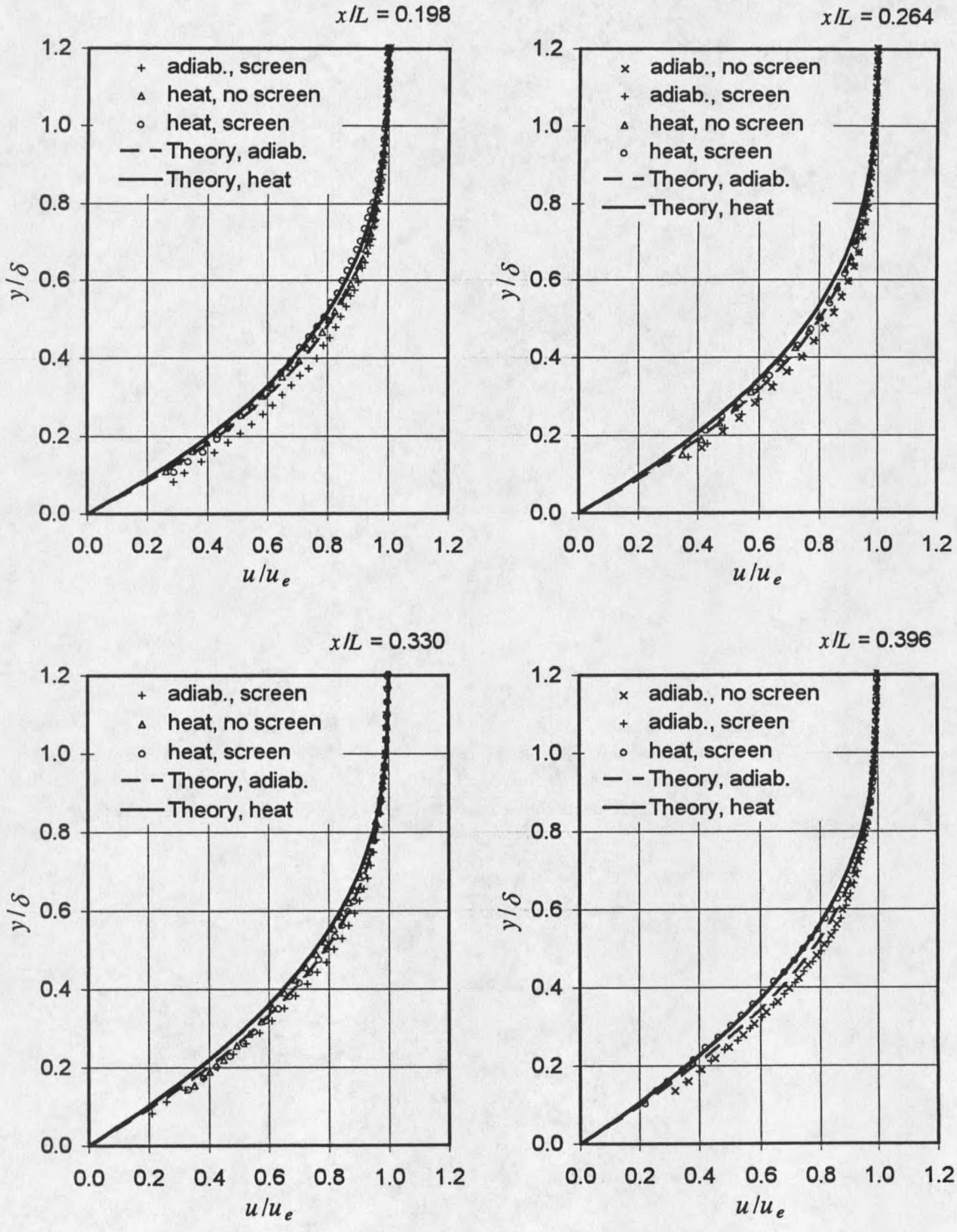
APPENDIX B: Complete Set of Velocity and Temperature Plots



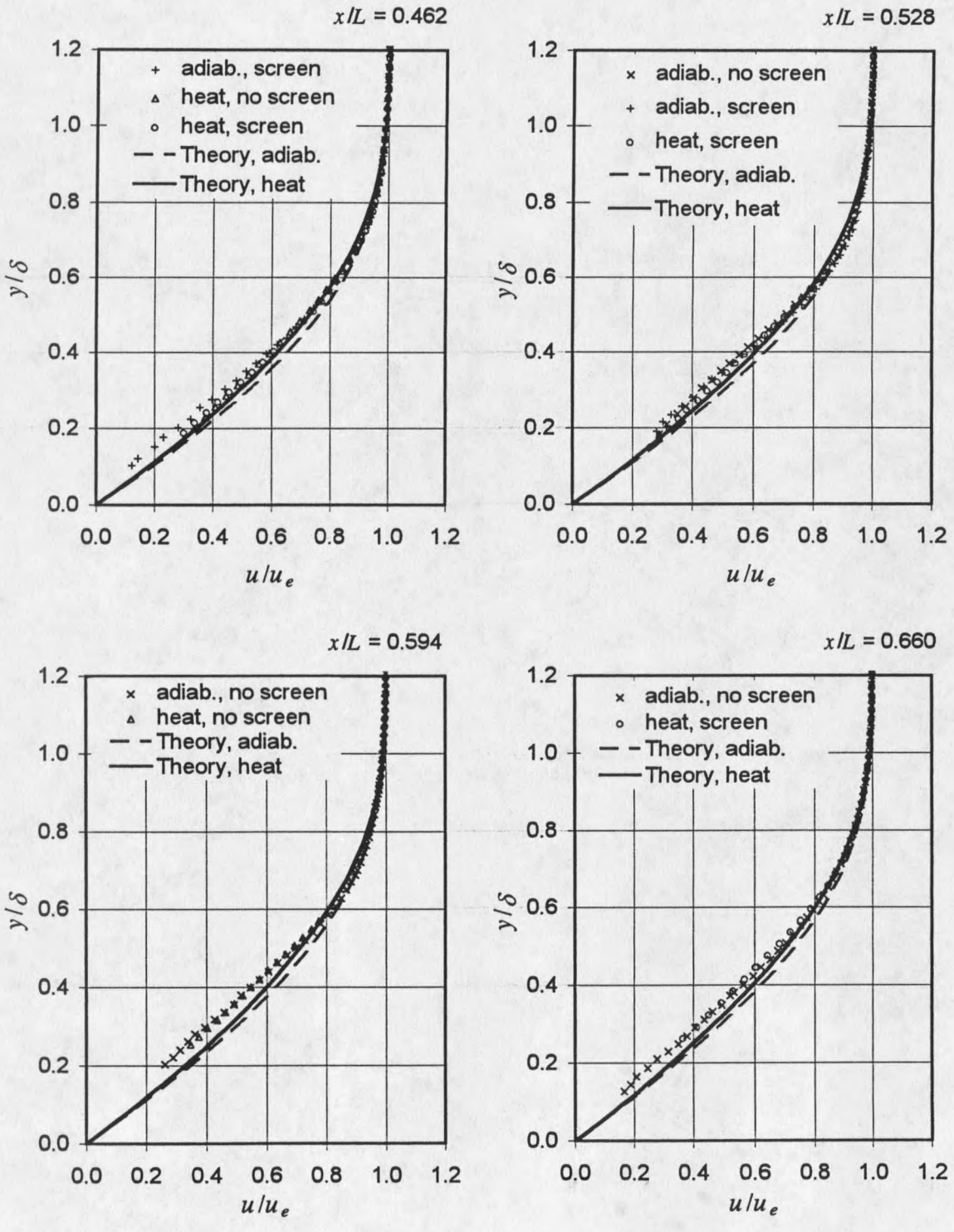
Velocity, Case 1a - 1b, P_o = 400 mmHg, T_o = 32 °C, Unheated



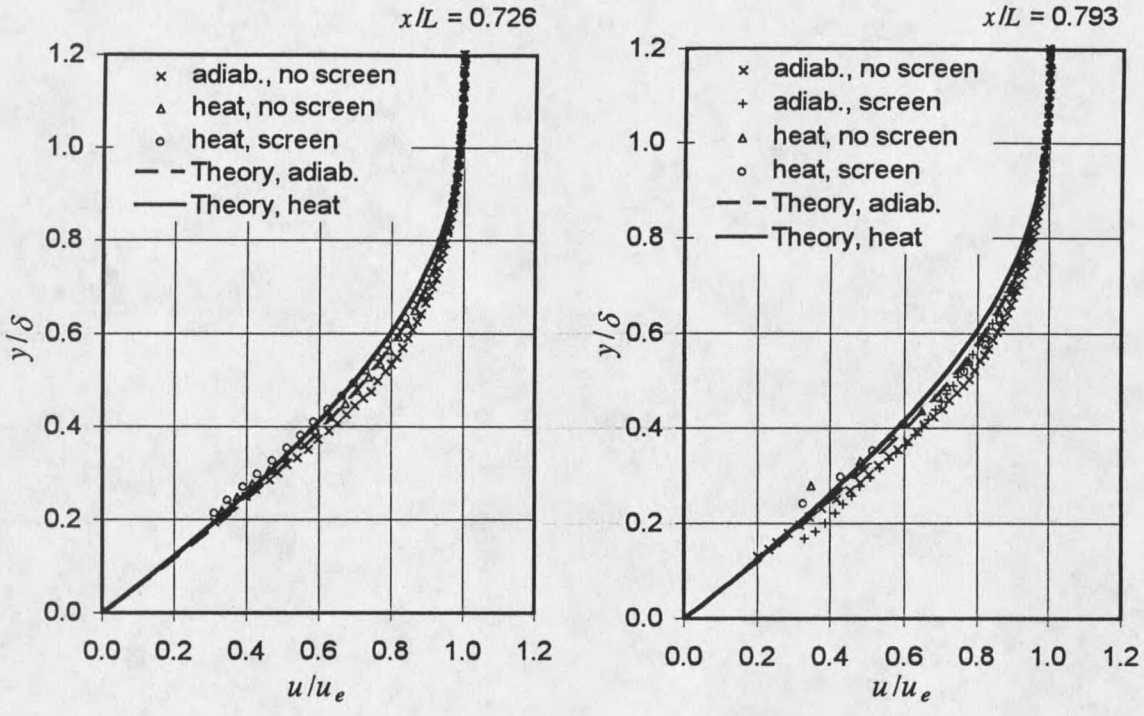
Velocity, Case 1a - 1b, P_o = 400 mmHg, T_o = 32 °C, Unheated



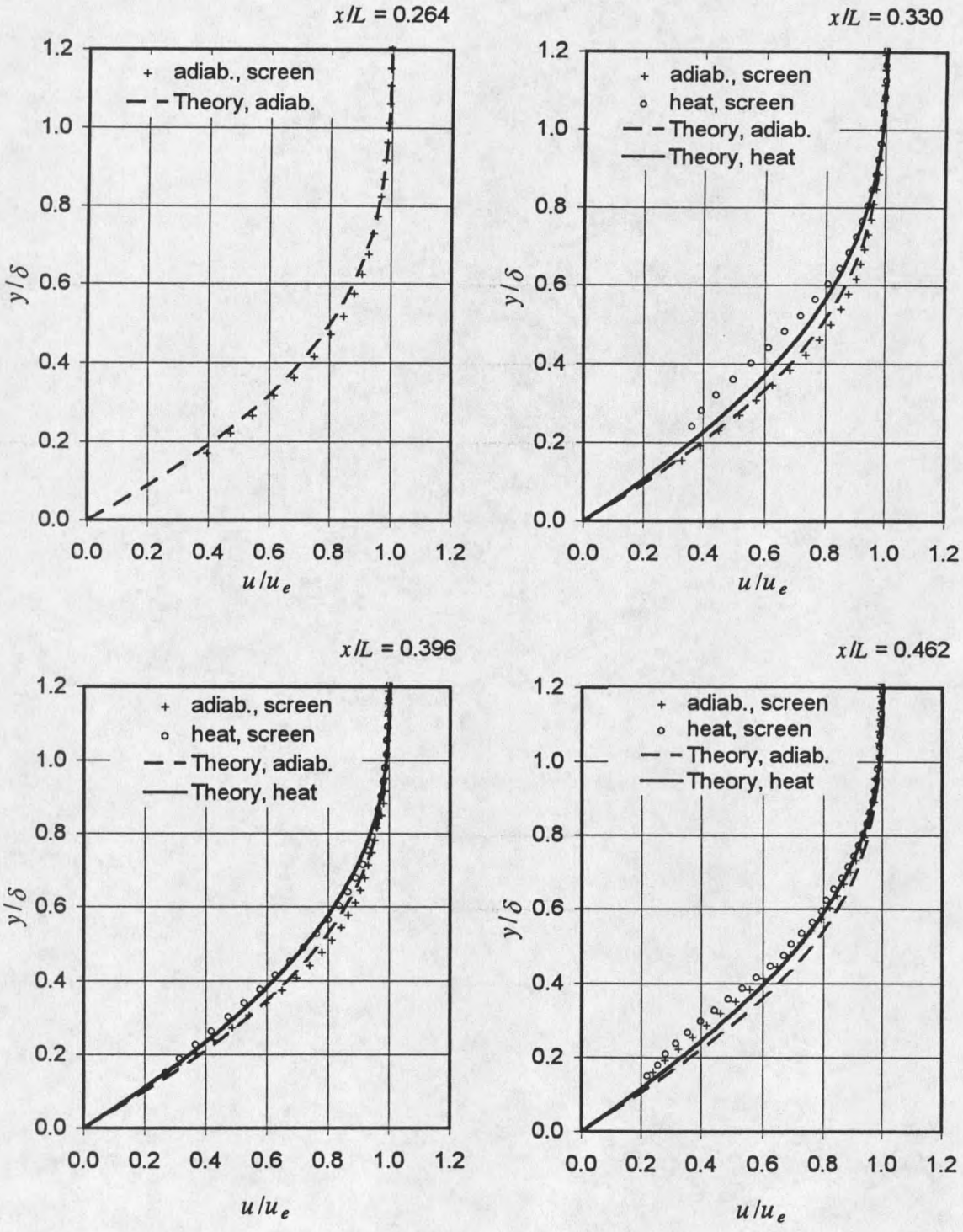
Velocity, Case 2a - 2d, P_o = 400 mmHg, T_o = 66 °C, Heated condition is $(T_w)_{x=0}$ = 97 °C



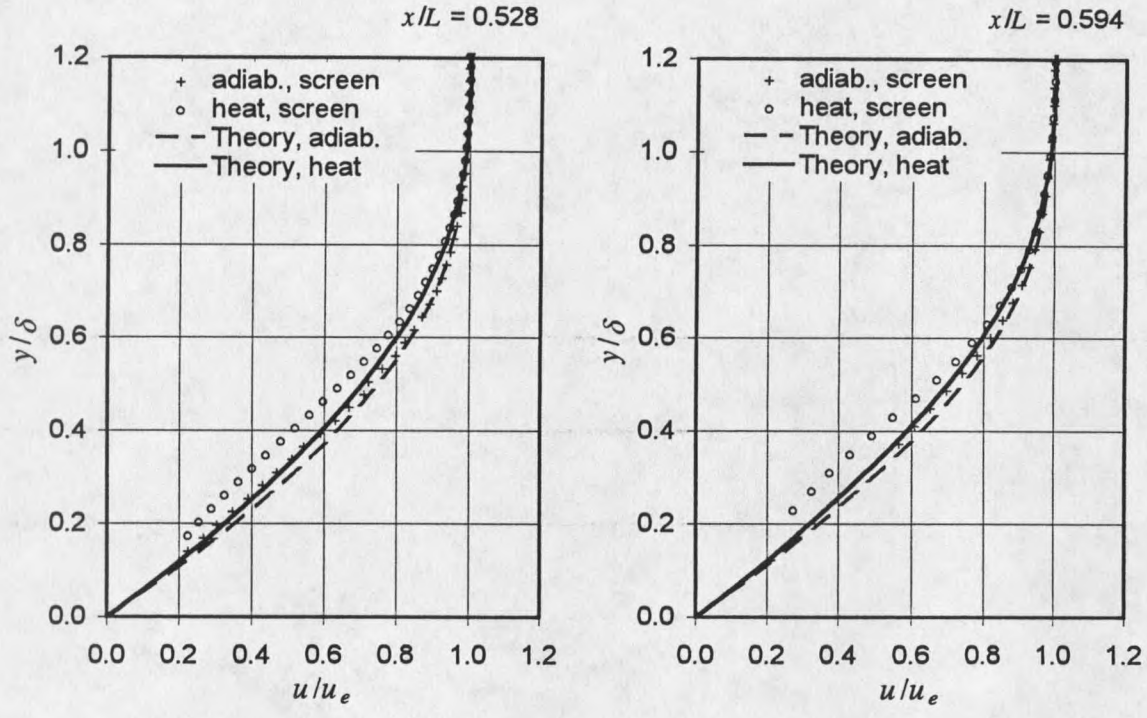
Velocity, Case 2a - 2d, P_o = 400 mmHg, T_o = 66 °C, Heated condition is $(T_w)_{x=0}$ = 97 °C



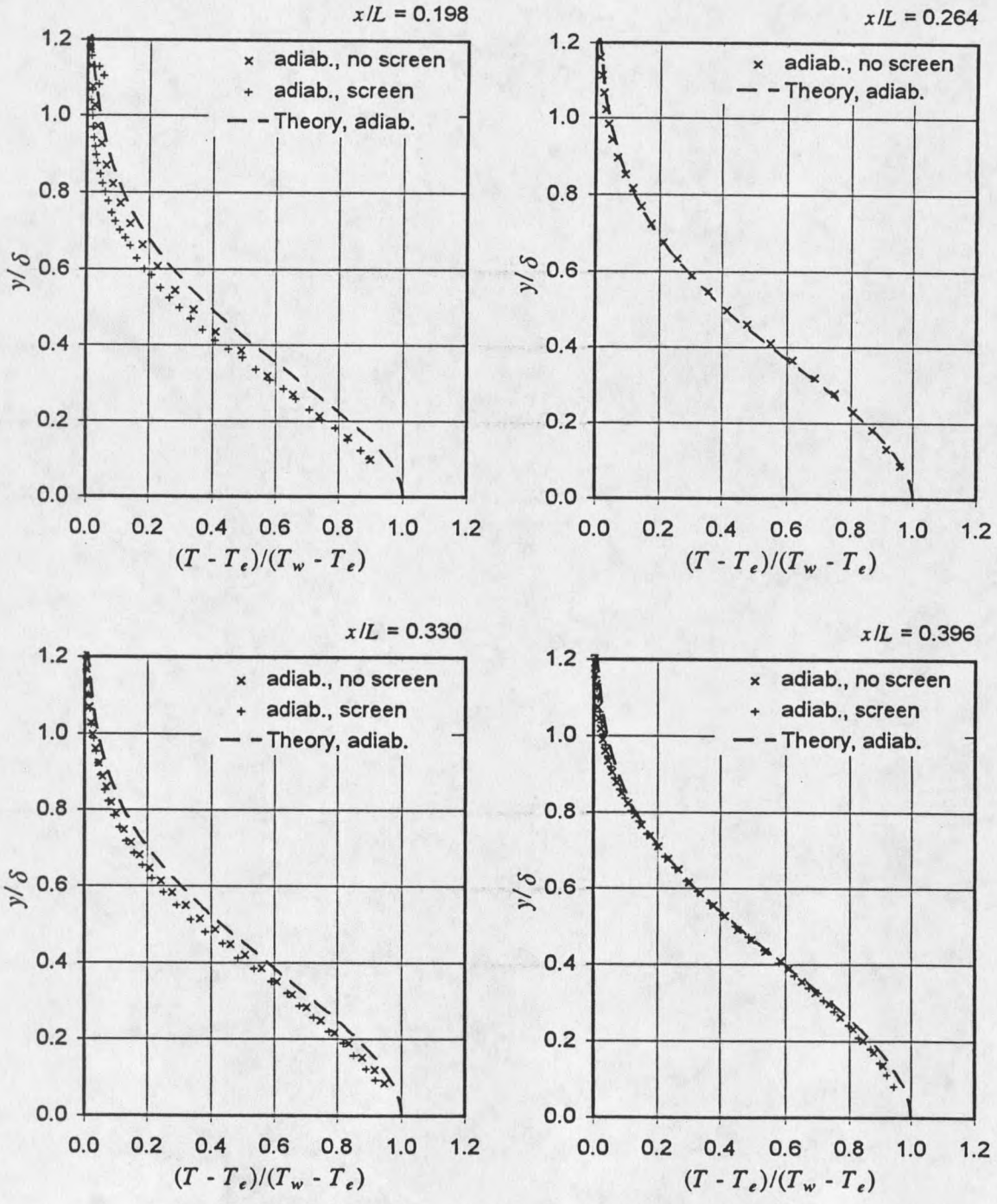
Velocity, Case 2a - 2d, P_o = 400 mmHg, T_o = 66 °C, Heated condition is $(T_w)_{x=0}$ = 97 °C



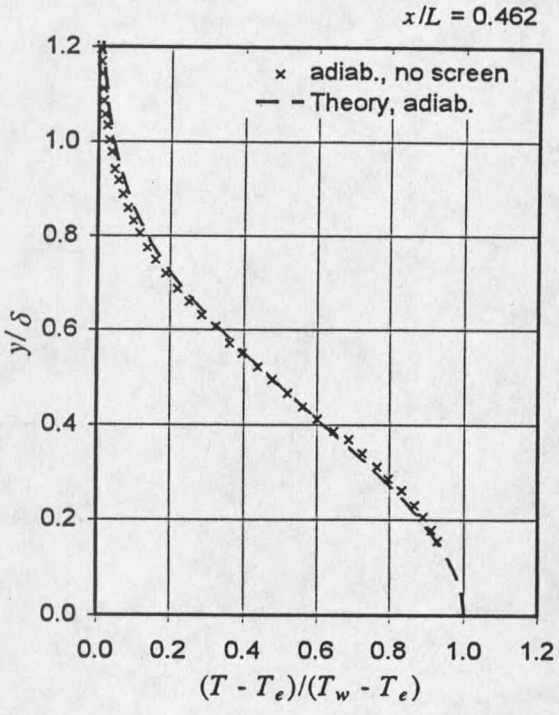
Velocity, Case 3a - 3b, P_o = 500 mmHg, T_o = 21 °C, Heated condition is $(T_w)_{x=0}$ = 60 °C



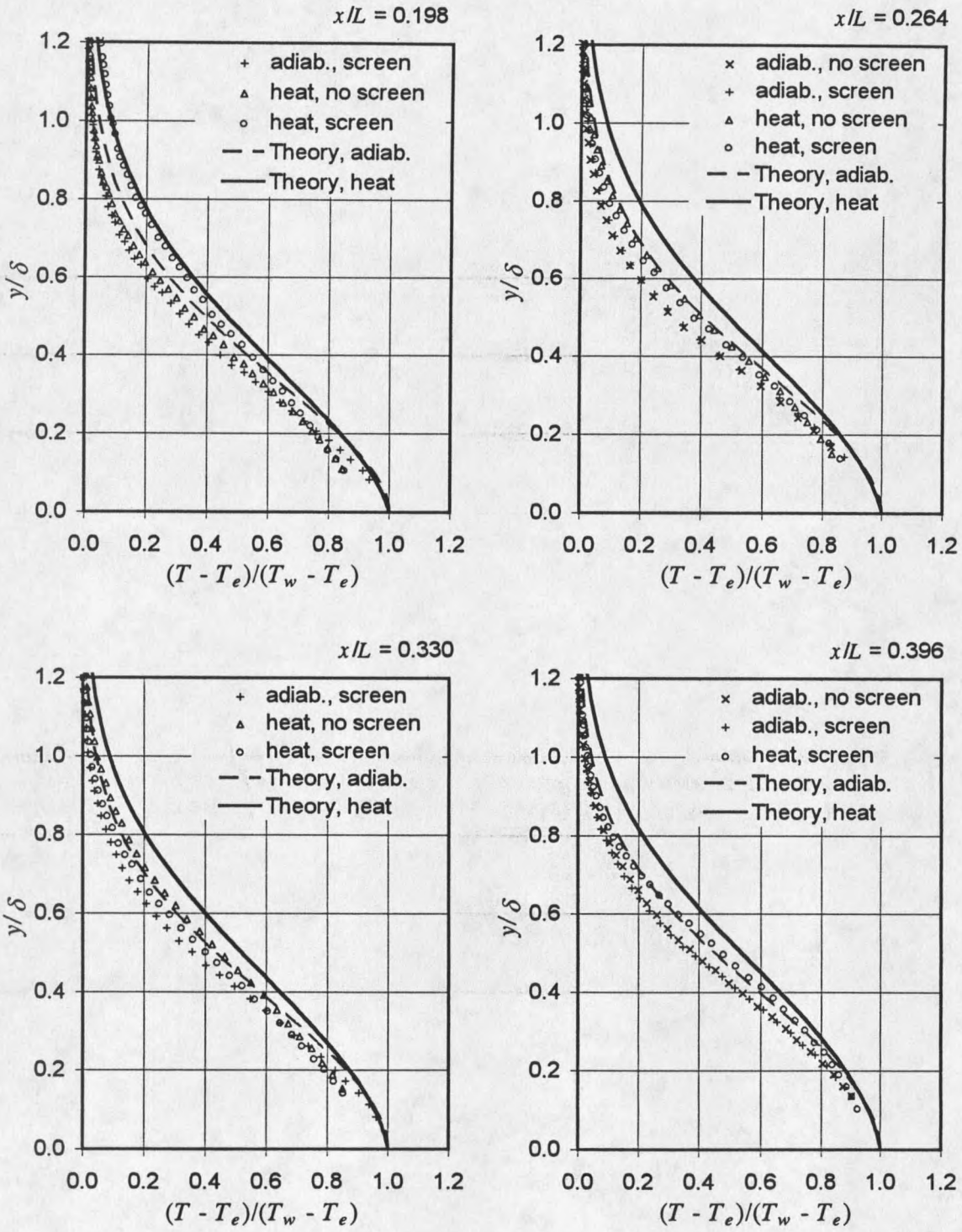
Velocity, Case 3a - 3b, P_o = 500 mmHg, T_o = 21 °C, Heated condition is $(T_w)_{x=0}$ = 60 °C



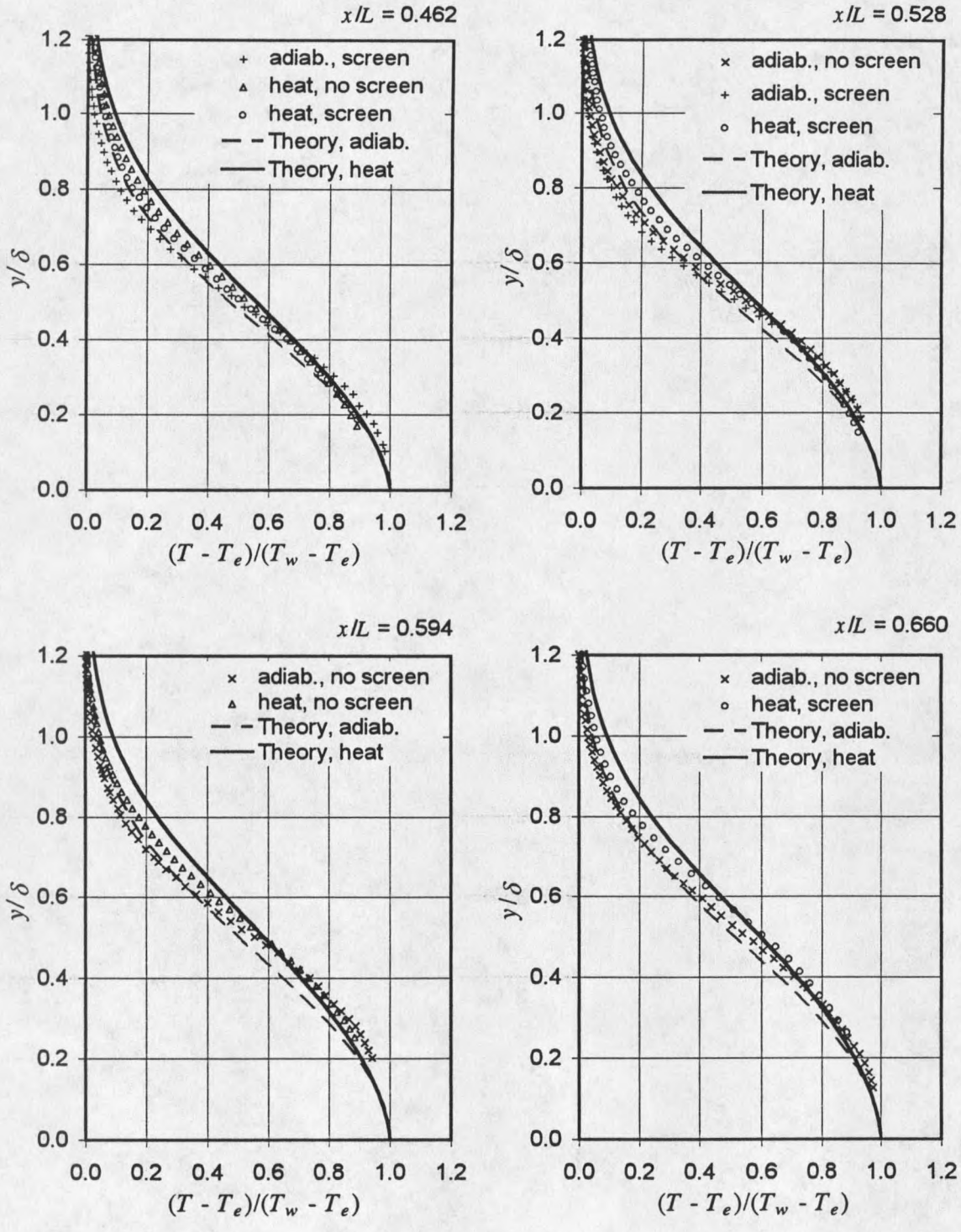
Temperature, Case 1a - 1b, P_o = 400 mmHg, T_o = 32 °C, Unheated



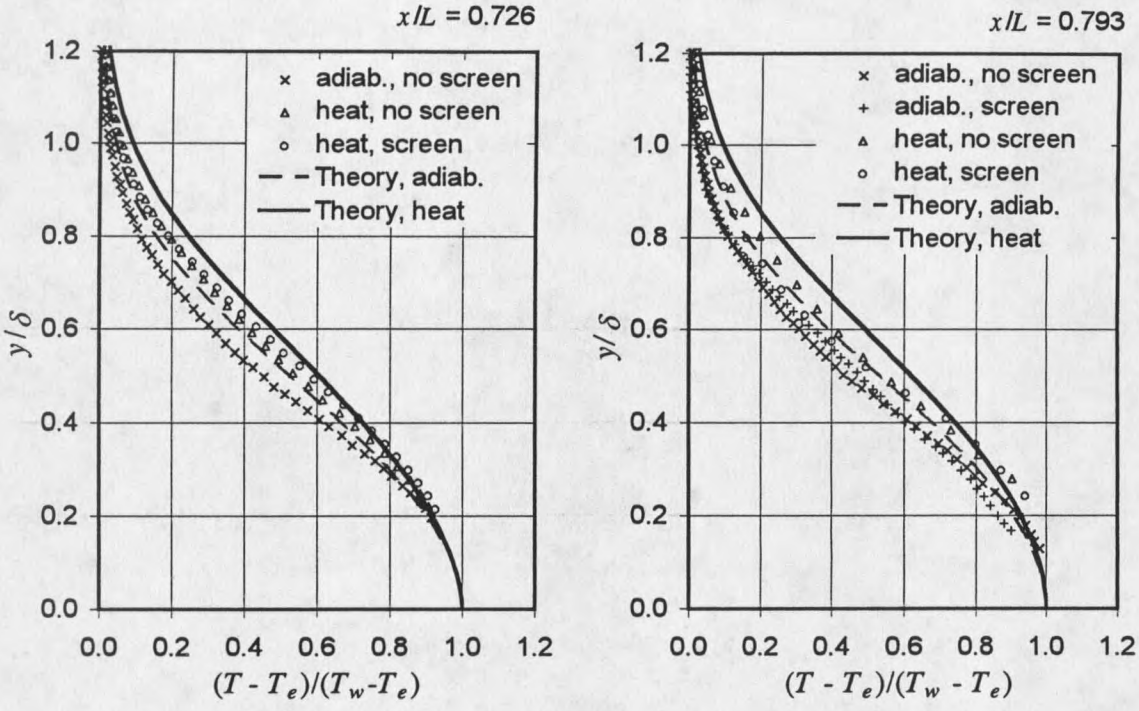
Temperature, Case 1a - 1b, P_o = 400 mmHg, T_o = 32 °C, Unheated



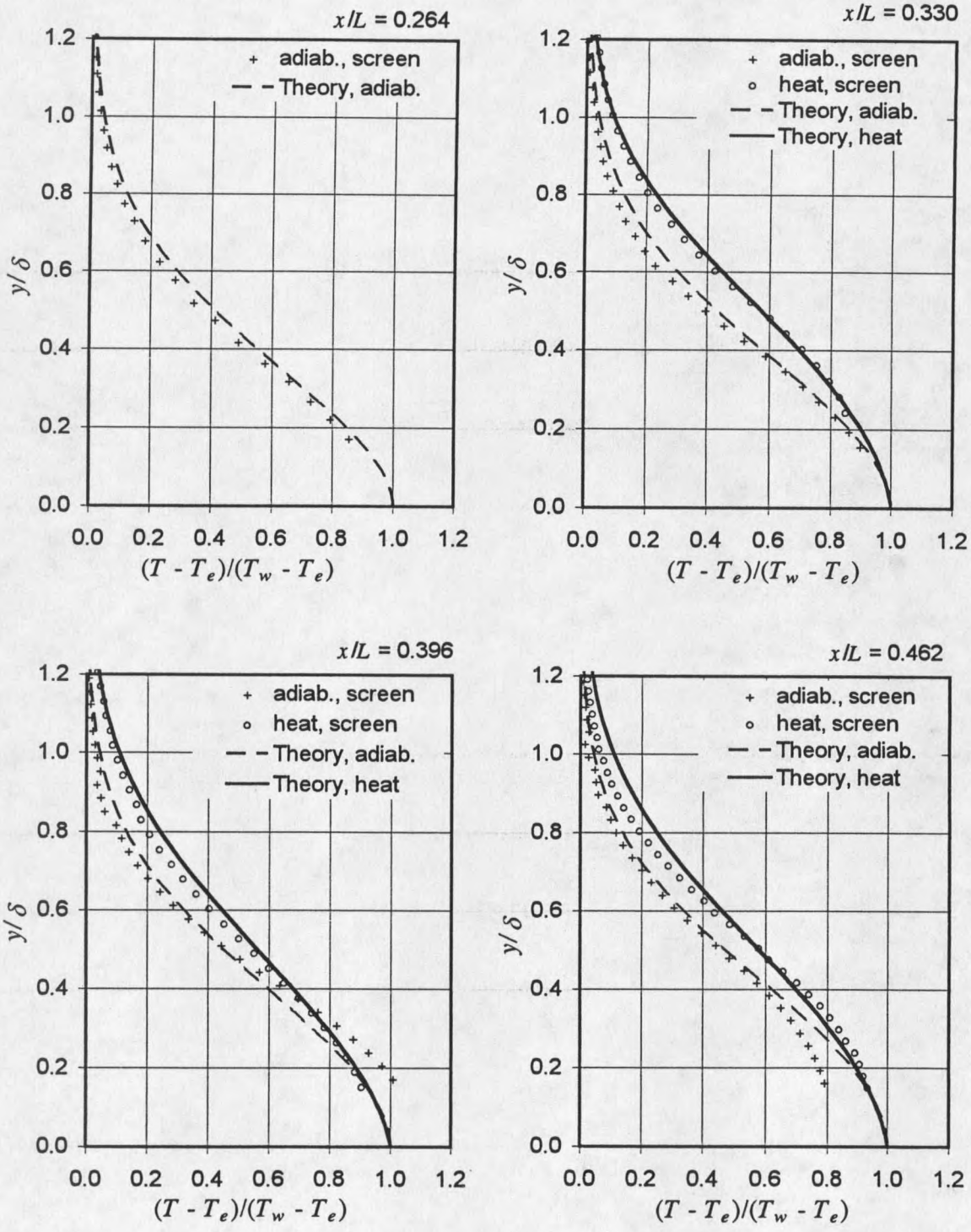
Temperature, Case 2a - 2d, P_o = 400 mmHg, T_o = 66 °C, Heated condition is $(T_w)_{x=0}$ = 97 °C



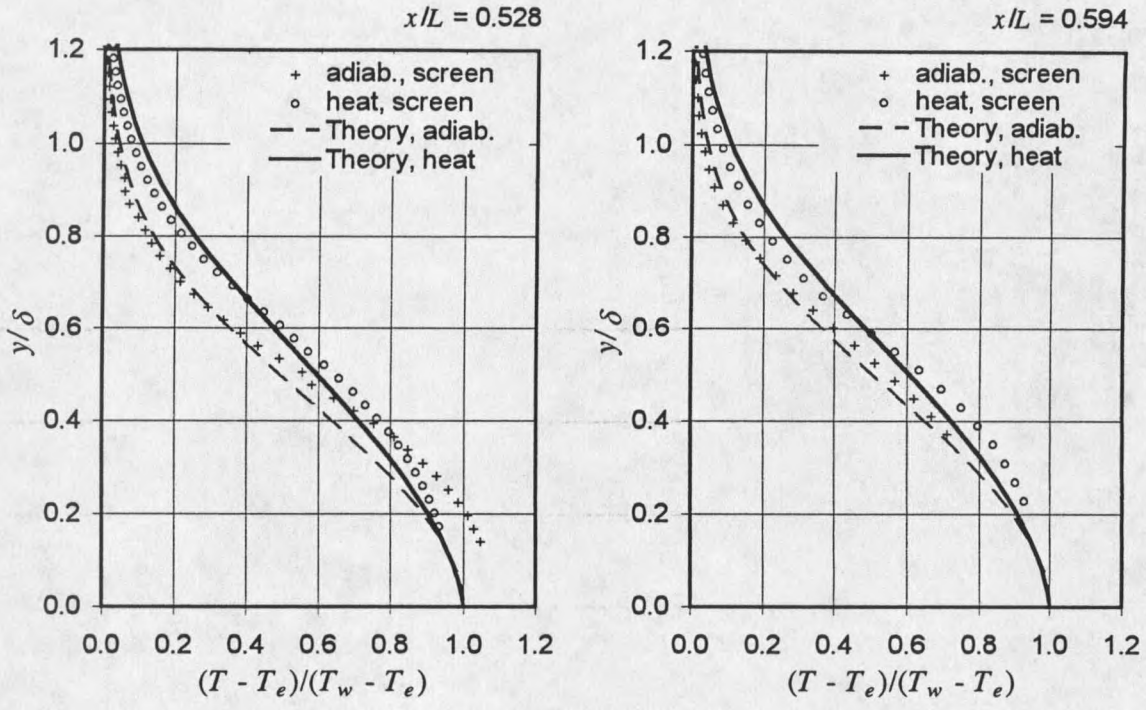
Temperature, Case 2a - 2d, P_o = 400 mmHg, T_o = 66 °C, Heated condition is $(T_w)_{x=0}$ = 97 °C



Temperature, Case 2a - 2d, P_o = 400 mmHg, T_o = 66 °C, Heated condition is $(T_w)_{x=0}$ = 97 °C



Temperature, Case 3a - 3b, P_o = 500 mmHg, T_o = 21 °C, Heated condition is $(T_w)_{x=0}$ = 60 °C



Temperature, Case 3a - 3b, P_o = 500 mmHg, T_o = 21 °C, Heated condition is $(T_w)_{x=0}$ = 60 °C

MONTANA STATE UNIVERSITY LIBRARIES
3 1762 10248950 5

HOUGHTEN
BINDERY LTD
UTICA/OMAHA
NE.