

Sys. 1897623

Escola Politécnica da Universidade de São Paulo
Departamento de Engenharia Mecânica

ESTUDO DO ESCOAMENTO COMPRESSÍVEL
UNIDIMENSIONAL GENERALIZADO

Rogério Rodrigues Jr.
William Arakaki

Orientador:
Prof. Dr. Marcos de M. Pimenta

São Paulo
1993

Aos nossos pais pelo apoio e
compreensão neste importante
período de nossas vida.

Agradecimentos

Ao orientador Prof. Dr. Marcos de M. Pimenta pelos ensinamentos e pela disposição para a elaboração deste trabalho.

A todos que direta ou indiretamente colaboraram na execução deste trabalho.

SUMÁRIO

1. INTRODUÇÃO.....	1
2. DESENVOLVIMENTO DO EQUACIONAMENTO.....	2
2.1 Modelagem do sistema físico.....	2
2.2 Equacionamento	3
2.3 Análise das equações.....	7
2.3.1 Caso 1: o valor de G é sempre negativo	8
2.3.2 Caso 2: o valor de G é sempre nulo	8
2.3.3 Caso 3: o valor de G é sempre positivo.....	8
2.3.4 Caso 4: o valor de G muda de sinal negativo para positivo	8
2.3.5 Caso 5: o valor de G muda de sinal positivo para negativo	9
3. DESCRIÇÃO DO ALGORITMO	10
4. IMPLEMENTAÇÃO DO PROGRAMA	12
5. RESULTADOS.....	13
6. COMENTÁRIOS E CONCLUSÕES	41
ANEXO 1: MÉTODO DE RUNGE-KUTTA.....	43
ANEXO 2: LISTAGEM DO PROGRAMA.....	45
ANEXO 3: DETALHES DO PROGRAMA.....	70
BIBLIOGRAFIA.....	72

1. INTRODUÇÃO

Na maioria das análises feitas em sistemas que utilizam fluidos compressíveis como meio de energia é considerada apenas a variação de um determinado parâmetro como, por exemplo, a variação de área em um escoamento permanente unidimensional. Estes escoamentos são denominados *Escoamentos Simples*. Como exemplos, pode-se citar:

- Escoamento isentrópico com variação de área;
- Escoamento de Fanno;
- Escoamento de Rayleigh.

Para estes escoamentos as soluções numéricas são simples.

Na vida real, porém, muitos dos sistemas de interesse são combinações de diversos escoamentos simples. A combinação destes escoamentos gera o chamado *Escoamento Compressível Unidimensional Generalizado*¹. O objetivo deste trabalho é fazer um estudo deste escoamento e implementar um algoritmo computacional para seu cálculo. Para isto, será utilizado o *Método dos Coeficientes de Influência* introduzido por SHAPIRO [1], o qual foi desenvolvido pela aplicação dos conceitos de conservação de massa, quantidade de movimento e energia para um volume de controle infinitesimal onde são considerados simultaneamente os efeitos de variação de área, troca de calor, atrito e injeção de massa.

Primeiramente será feito um estudo do escoamento compressível e a partir deste será desenvolvido o equacionamento de um algoritmo para sua solução numérica. Além disso, serão desenvolvidas rotinas de manipulação de dados e gráficos com o objetivo de facilitar a simulação e análise de problemas.

Finalmente serão analisados alguns problemas com base em resultados obtidos através de simulações feitas com o programa.

¹Por um motivo de simplificação, será sempre citado no texto como *escoamento compressível*.

2. DESENVOLVIMENTO DO EQUACIONAMENTO

2.1 Modelagem do sistema físico:

Considerando o seguinte volume de controle:

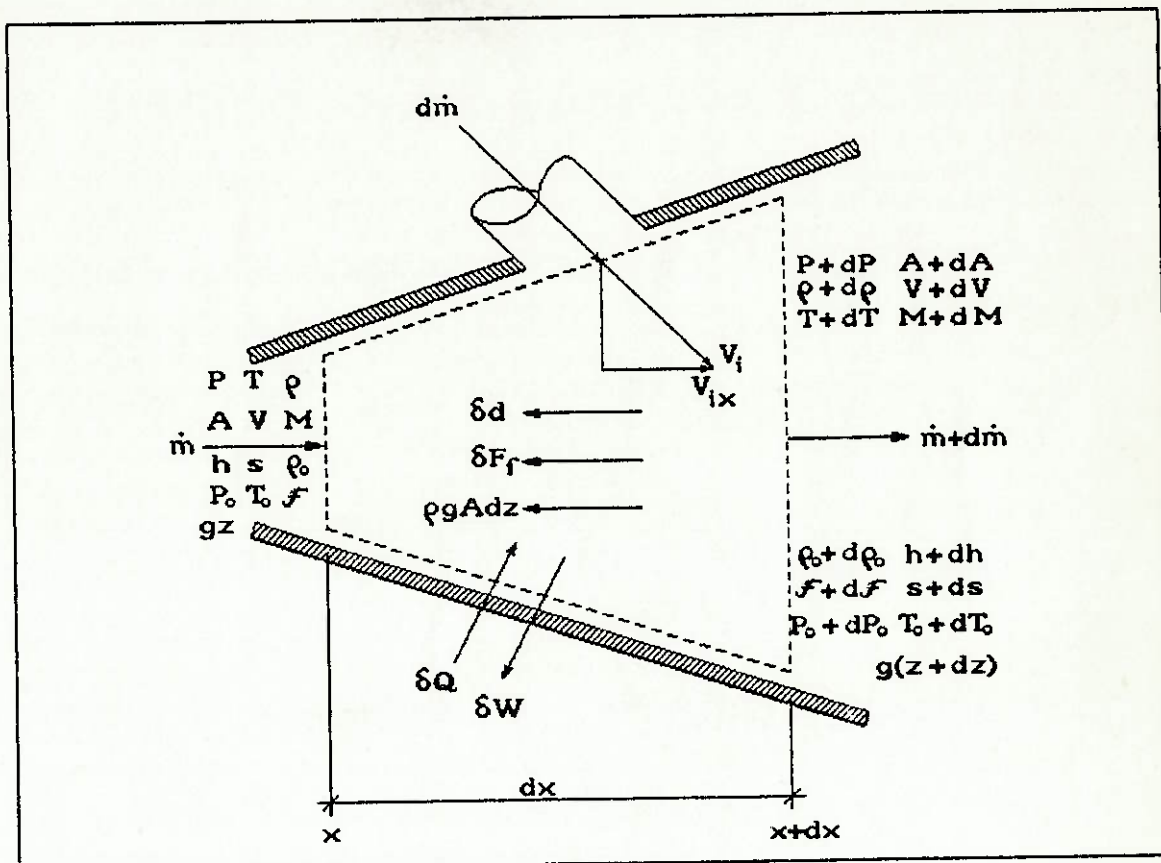


Figura 1 - volume de controle

onde:

- A área
- P_0 pressão de estagnação
- P pressão estática
- T_0 temperatura de estagnação
- T temperatura estática
- M número de Mach
- V velocidade

h	entalpia
ρ	densidade
s	entropia
z	cota
J	função impulso
δQ	transferência de calor
δW	trabalho de eixo
δd	arrasto
$\dot{m}$	vazão mássica
$\dot{dm}$	injeção de massa
δF_f	força de atrito

Hipóteses:

- escoamento unidimensional;
- regime permanente;
- gás perfeito;
- mudanças nas propriedades do escoamento são contínuas.

2.2 Equacionamento

Aplicando os conceitos de conservação de massa, quantidade de movimento e energia para o volume de controle, obtém-se²:

Conservação da massa:

$$\frac{\dot{dm}}{\dot{m}} = \frac{d\rho}{\rho} + \frac{dV}{V} + \frac{dA}{A} \quad (1)$$

²Detalhes são descritos por SHAPIRO [1].

Conservação da quantidade de movimento:

$$\frac{dp}{p} + kM^2 \frac{dM}{M} + k \frac{M^2}{2} \frac{dT}{T} + k \frac{M^2}{2} \left[\frac{f}{D} dx + \frac{2\delta d}{kM^2 PA} \right] + kM^2(1-y) \frac{\dot{dm}}{\dot{m}} = 0 \quad (2)$$

onde:

fcoeficiente de atrito

Ddiâmetro

$$y = \frac{V_w}{V_i}$$

$$k = \frac{C_p}{C_v}$$

Conservação da energia:

$$\frac{\delta Q - \delta W - dH_i}{C_p T} = \left[1 + \frac{(k-1)}{2} M^2 \right] \frac{dT_o}{T_o} \quad (3)$$

As expressões abaixo completam o sistema de equações.

Equação de estado:

$$\frac{dP}{P} = \frac{d\rho}{\rho} + \frac{dT}{T} \quad (4)$$

Número de Mach:

$$\frac{dM}{M} = \frac{dV}{V} - \frac{1}{2} \frac{dT}{T} \quad (5)$$

Temperatura estática e de estagnação:

$$\frac{dT_o}{T_o} = \frac{dT}{T} + \frac{(k-1)M^2}{1 + \frac{(k-1)M^2}{2}} \frac{dM}{M} \quad (6)$$

Pressão estática e de estagnação:

$$\frac{dP_o}{P_o} = \frac{dP}{P} + \frac{kM^2}{1 + \frac{(k-1)M^2}{2}} \frac{dM}{M} \quad (7)$$

Função impulso:

$$\frac{d\mathcal{J}}{\mathcal{J}} = \frac{dP}{P} + \frac{dA}{A} + \frac{2kM^2}{1 + kM^2} \frac{dM}{M} \quad (8)$$

Entropia:

$$\frac{ds}{C_p} = \frac{dT}{T} - \frac{k-1}{k} \frac{dP}{P} \quad (9)$$

Com as equações (1), (2), (4) a (9) tem-se um sistema de 8 equações no qual as variáveis dependentes dP/P , $d\rho/\rho$, dT/T , dV/V , dP_o/P_o , $d\mathcal{J}/\mathcal{J}$ e ds/C_p são expressas em termos de quatro variáveis independentes ou *driving potentials*: dA/A , dTo/To , $d\dot{m}/\dot{m}$ e $[f dx/D + 2\delta d/(kM^2 PA)]$. A solução deste sistema de equações lineares resulta em oito equações diferenciais, uma para cada variável dependente, que são funções do número de Mach local e dos *driving potentials*.

A distribuição do número de Mach na direção x pode ser obtida através da integração da expressão abaixo:

$$\begin{aligned} \frac{1}{M} \frac{dM}{dx} = & -\frac{\psi}{A(1-M^2)} \frac{dA}{dx} + \frac{kM^2 \psi}{2(1-M^2)} \left[\frac{f}{D} + \frac{2}{kM^2 PA} \frac{\delta d}{dx} \right] + \\ & + \frac{(1+kM^2) \psi}{2(1-M^2) To} \frac{dTo}{dx} + \frac{\psi[(1+kM^2) - ykM^2]}{(1-M^2) \dot{m}} \frac{d\dot{m}}{dx} \end{aligned} \quad (10)$$

onde:

$$\psi = 1 + \frac{k-1}{2} M^2$$

A integração das outras sete equações diferenciais não é necessária pois com aplicação da equação da energia e da continuidade entre duas seções de escoamento e com a introdução das definições do número de Mach e da função impulso juntamente com a equação de estado dos gases perfeitos obtém-se relações para o cálculo das outras variáveis dependentes:

$$\frac{T_2}{T_1} = \frac{T_{o2}}{T_{o1}} \frac{1 + \frac{k-1}{2} M_1^2}{1 + \frac{k-1}{2} M_2^2} \quad (11)$$

$$\frac{V_2}{V_1} = \frac{M_2}{M_1} \sqrt{\frac{T_2}{T_1}} \quad (12)$$

$$\frac{P_2}{P_1} = \frac{\dot{m}_2}{\dot{m}_1} \frac{A_1}{A_2} \frac{M_1}{M_2} \sqrt{\frac{T_2}{T_1}} \quad (13)$$

$$\frac{\rho_2}{\rho_1} = \frac{\dot{m}_2}{\dot{m}_1} \frac{A_1}{A_2} \frac{V_1}{V_2} \quad (14)$$

$$\frac{f_2}{f_1} = \frac{P_2}{P_1} \frac{A_2}{A_1} \frac{1 + kM_2^2}{1 + kM_1^2} \quad (15)$$

$$s_2 = s_1 + C_p \ln \frac{T_2}{T_1} - R \ln \frac{P_2}{P_1} \quad (16)$$

$$\frac{P_{o2}}{P_{o1}} = \frac{P_2}{P_1} \left[\frac{1 + \frac{k-1}{2} M_2^2}{1 + \frac{k-1}{2} M_1^2} \right]^{k/(k-1)} \quad (17)$$

Através das equações (10) a (17) pode-se desenvolver um procedimento para resolver qualquer escoamento unidimensional de fluido compressível em regime permanente.

2.3 Análise das equações

Observa-se através da equação (10) que o *Método dos Coeficientes de Influência* apresenta um ponto de singularidade quando $M=1$ (o denominador torna-se nulo). Este fato torna-se um inconveniente quando se deseja solucionar um problema em que há uma passagem contínua do regime subsônico para o supersônico, ou vice-versa³. Logo, para uma análise do comportamento do escoamento nesta região, algumas considerações devem ser feitas.

A equação (10) pode ser escrita na forma:

$$\frac{dM}{dx} = M\psi \frac{G(x)}{1-M^2}$$

$$G(x) = -\frac{1}{A} \frac{dA}{dx} + \frac{kM^2}{2} \left[\frac{f}{D} + \frac{2}{kM^2 PA} \frac{\delta d}{dx} \right] +$$

$$+ \frac{(1+kM^2)}{2To} \frac{dT_o}{dx} + \frac{[(1+kM^2) - \gamma kM^2]}{\dot{m}} \frac{d\dot{m}}{dx}$$

A partir destas equações pode-se montar uma tabela para os valores de dM/dx :

	$M < 1$	$M = 1$	$M > 1$
$G < 0$	negativo	∞	positivo
$G = 0$	zero	indeterminado	zero
$G > 0$	positivo	∞	negativo

Tabela 1 - Valores de dM/dx

Cada caso será melhor detalhado a seguir.

³Por exemplo, escoamento em um bocal convergegente/divergente. Discussões mais detalhadas sobre este aspecto podem ser encontradas em ZUCROW & HOFFMAN [2].

2.3.1 CASO 1: o valor de G é sempre negativo

Se o escoamento é inicialmente subsônico, M decresce continuamente. Por outro lado, se o escoamento é inicialmente supersônico, M cresce continuamente. Exemplos: escoamento isentrópico em uma passagem divergente e escoamento com resfriamento.

2.3.2 CASO 2: o valor de G é sempre nulo

O número de Mach sempre permanece constante, independente do regime de escoamento. Exemplo: escoamento adiabático com atrito em uma passagem divergente

$$\text{com } \frac{d(\ln A)}{dx} = \frac{2kM^2 f}{D}.$$

2.3.3 CASO 3: o valor de G é sempre positivo

Se o escoamento é inicialmente subsônico, M tende a crescer continuamente. Se é supersônico, M tende a decrescer continuamente. Em resumo, o número de Mach tende sempre para transição de um regime para outro. Entretanto, se for atingido $M=1$ e G continuar positivo para um incremento de posição, ocorre *blocagem* e não é possível a existência de um escoamento em regime permanente. Depois de um período transitório de reajuste, uma nova condição de regime permanente será estabelecida para que não ocorra *blocagem* (isto envolve geralmente uma redução do número de Mach inicial e/ou o aparecimento de ondas de choque). Exemplos: escoamento isentrópico em uma passagem convergente, escoamento com aquecimento e escoamento adiabático com atrito em uma passagem convergente.

2.3.4 CASO 4: o valor de G muda de sinal negativo para positivo

Se o escoamento é inicialmente subsônico, M primeiro decresce, passa por um mínimo onde $G=0$ e então cresce, podendo ocorrer *blocagem* (situação limite). Entretanto, se o escoamento é supersônico, M cresce, passa por um máximo ($G=0$) e então decresce (novamente pode haver *blocagem*). Exemplo: escoamento isentrópico através de um bocal divergente-convergente.

2.3.5 CASO 5: o valor de G muda de sinal positivo para negativo

Serão apenas considerados escoamentos inicialmente subsônicos, mas deve ficar claro que existem situações similares para escoamentos supersônicos. Assim, para escoamentos com $M < 1$ no início:

- Se M é bastante baixo, irá crescer, passar por um máximo ($G=0$) e então decrescerá continuamente;
- Se M é suficientemente alto, o escoamento *bloca* antes de atingir $G=0$ e então a solução não pode continuar;
- Se para uma dada variação de $G(x)$, o escoamento atingir $M=1$ no ponto exato onde $G=0$, ocorrerá uma passagem contínua para o regime supersônico ou uma volta ao regime subsônico.

Estas considerações serão muito importantes para a fase de implementação do algoritmo, como poderá ser visto a seguir.

3. DESCRIÇÃO DO ALGORITMO

A partir das equações, será desenvolvido o seguinte procedimento de cálculo:

1. Utilizando as condições iniciais do fluido (parâmetros do escoamento) e as variações para os *driving potentials* (condições de contorno), calcula-se o valor do número de Mach para o próximo incremento de posição do conduto através da resolução da equação (10) por um método numérico (no caso, foi utilizado o Método de Runge-Kutta⁴);

2. Com o valor de Mach calculado e as equações (11) a (17), pode-se determinar as demais variáveis do escoamento;

3. Calcula-se o valor da função $G(x)$ para cada passo, verificando as condições do escoamento segundo os casos possíveis para $G(x)$;

4. Prossegue-se calculando desta forma até o final do conduto, caracterizando todo o escoamento.

Caso se atinja $M=1$ em uma região onde $G(x)$ se mantém positivo, significa que o escoamento foi *blocado*. O número de Mach inicial é então reduzido, e todo o cálculo é refeito, até que se consiga percorrer todo o conduto.

De acordo com o algoritmo descrito, tem-se o seguinte fluxograma:

⁴Mais detalhes sobre este método serão apresentados no ANEXO 1.

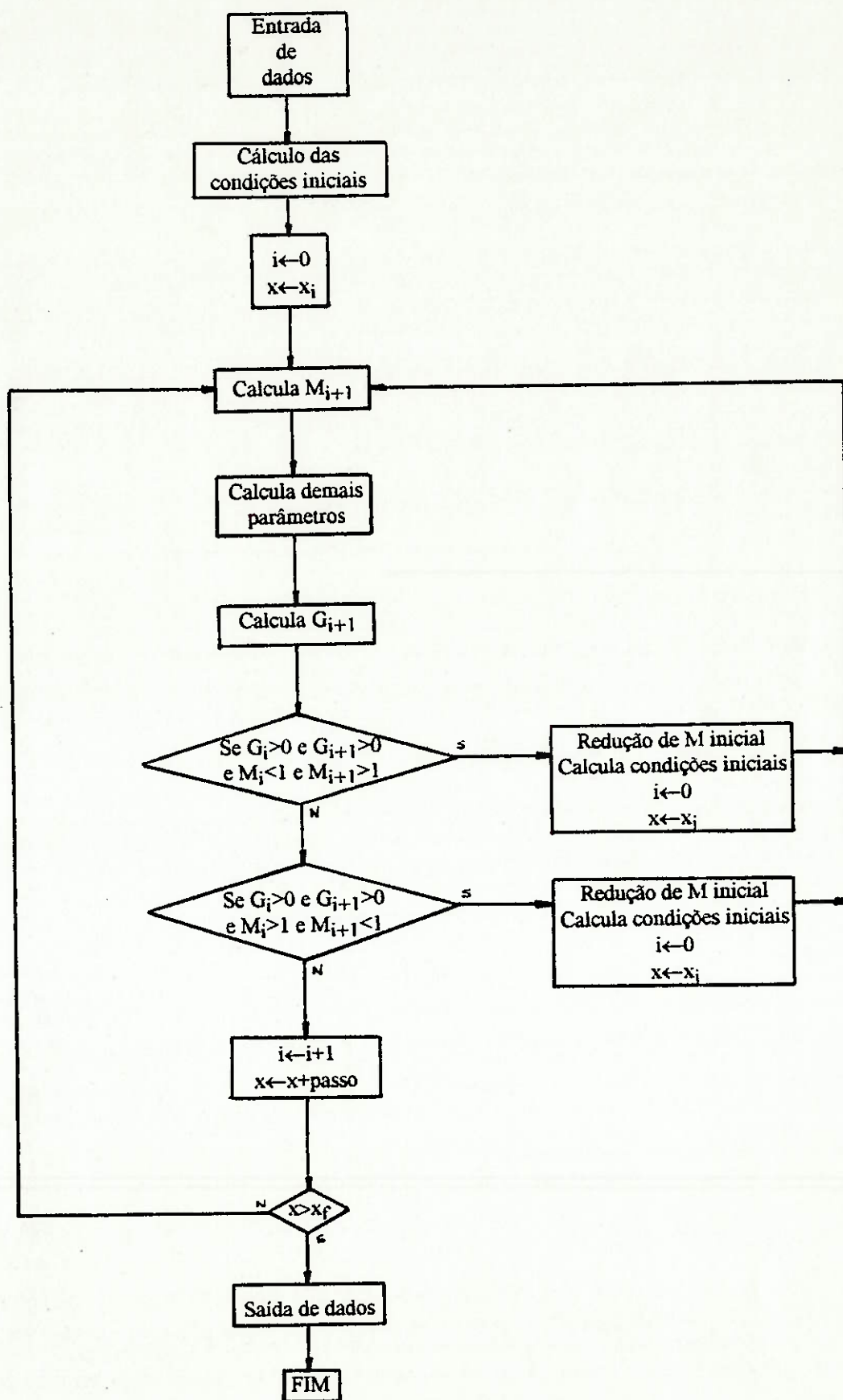


Figura 2 - Fluxograma do programa

4. IMPLEMENTAÇÃO DO PROGRAMA

O programa será feito utilizando-se como linguagem de programação o Turbo Pascal e para que possa ser executados é necessário um microcomputador do tipo PC⁵. Foram desenvolvidas rotinas para manipulação de dados:

- leitura de dados via teclado ou recuperação de arquivos;
- armazenamento de dados via arquivo texto e real;
- plotagem dos pontos calculados através de rotinas gráficas.

O programa é composto por uma unidade principal, denominada TF.EXE, e por duas *Units*: CALCULUS.TPU e GRAFICUS.TPU⁶. Além disso, para executar as rotinas gráficas do programa (plotagem de gráficos das variáveis do escoamento) são necessários os arquivos com extensão BGI e CHR do Turbo Pascal.

Apesar do trabalho objetivar o desenvolvimento de um programa para o cálculo do escoamento compressível, uma vez descrito o algoritmo e as equações básicas, não há necessidade de se aprofundar em detalhes do programa como, por exemplo, as rotinas gráficas, que dependem de técnicas de programação e características da linguagem de programação utilizada.

Através do programa pode-se então fazer análises sobre o escoamento compressível e comparações com casos calculados teoricamente para se verificar a validade dos resultados obtidos.

⁵Devido ao grande número de cálculos, aconselha-se que a configuração mínima seja um PC 386. Além disso, devido às saídas gráficas, seria conveniente a utilização de um monitor de padrão VGA.

⁶Uma listagem do programa principal e das *Units* encontra-se no ANEXO

5. RESULTADOS

Para que possa ser feita uma avaliação do programa, será resolvido um problema de escoamento de Rayleigh para comparar os resultados obtidos com o programa e os encontrados analiticamente. O problema utilizado para a comparação é o exemplo 8.3 de PIMENTA [7], cujo sistema físico é apresentado abaixo (deseja-se determinar as condições no ponto 3):

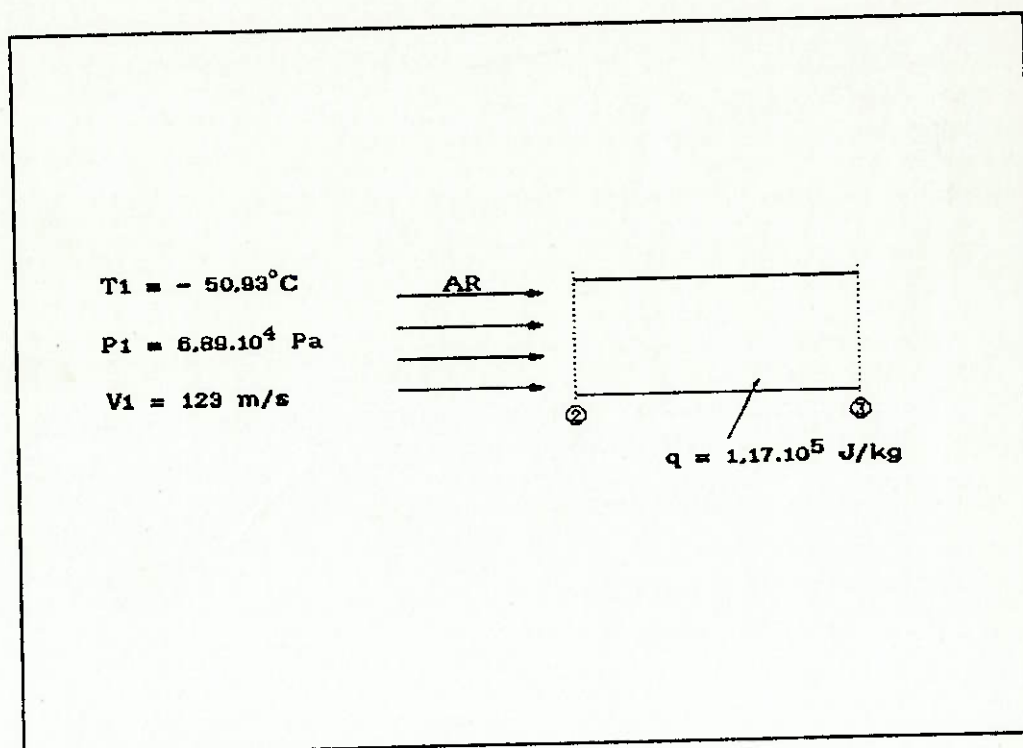


Figura 3 - sistema físico

Resultados obtidos analiticamente:

$$M_3=0.605 \quad P_3=56300 \text{ Pa} \quad T_3=322.4 \text{ K}$$

Resultados obtidos através do programa:

$$M_3=0.6046 \quad P_3=56330 \text{ Pa} \quad T_3=322.48 \text{ K}$$

O programa também calcula a distribuição do número de Mach entre as seções 2 e 3, e este resultado é apresentado no gráfico seguinte :

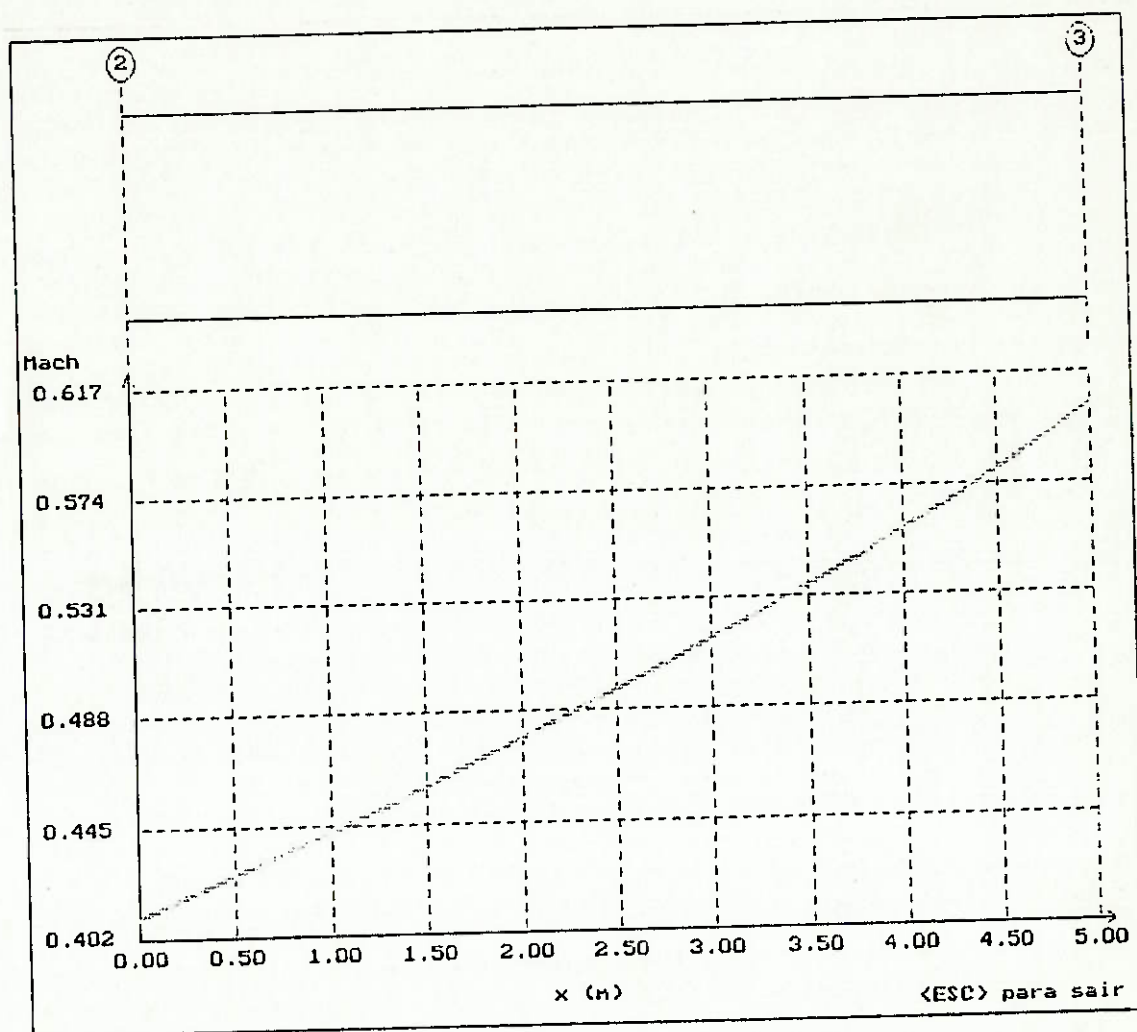


Gráfico 1 - Mach=f(x)

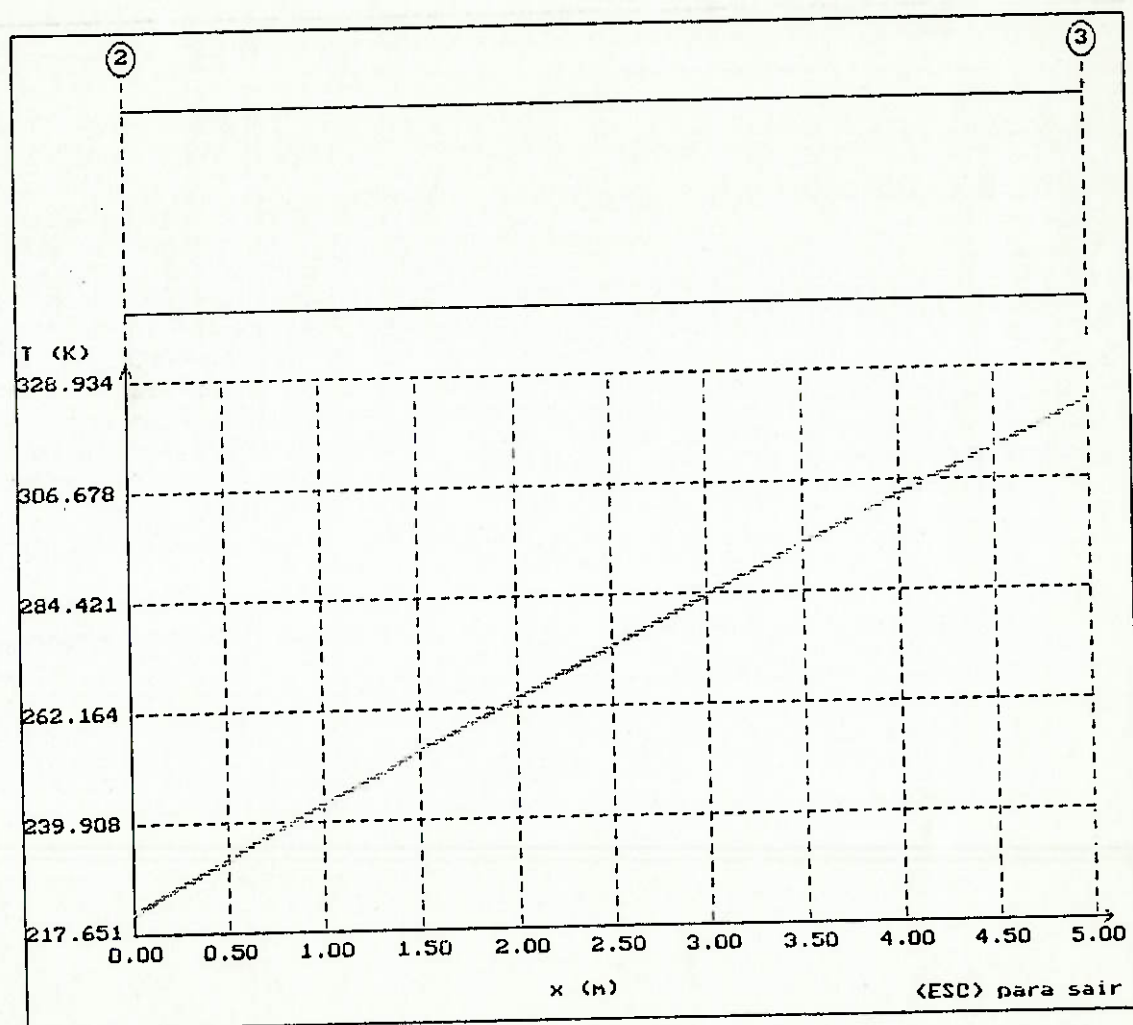


Gráfico 2 - $T=f(x)$

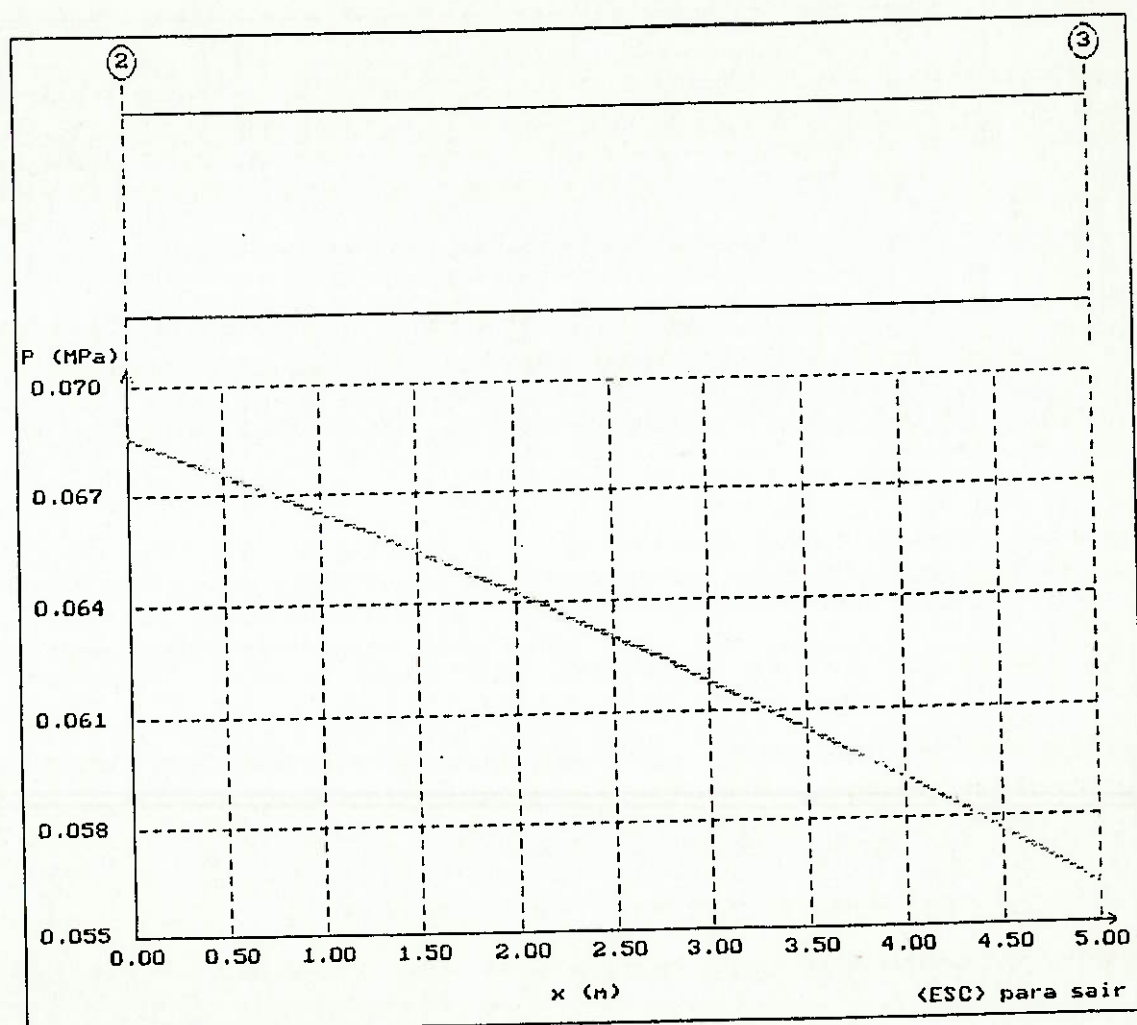


Gráfico 3 - $P=f(x)$

O gráfico 4 mostra a variação do número de Mach num bocal divergente para regimes de escoamento subsônico e supersônico. As características do bocal e as condições de contorno são apresentados a seguir :

- $k = 1.4$
- $R = 287 \text{ N.m/kg.K}$
- Fator de atrito : $f = 0.005$
- Diâmetro : $D(x) = 0.2 + 0.1.x$
- Comprimento : 2 m
- Condições na entrada do bocal : $P_o = 0.100 \text{ MPa}$
 $T_o = 300 \text{ K}$

Escoamento em Bocal Divergente

Numero de Mach

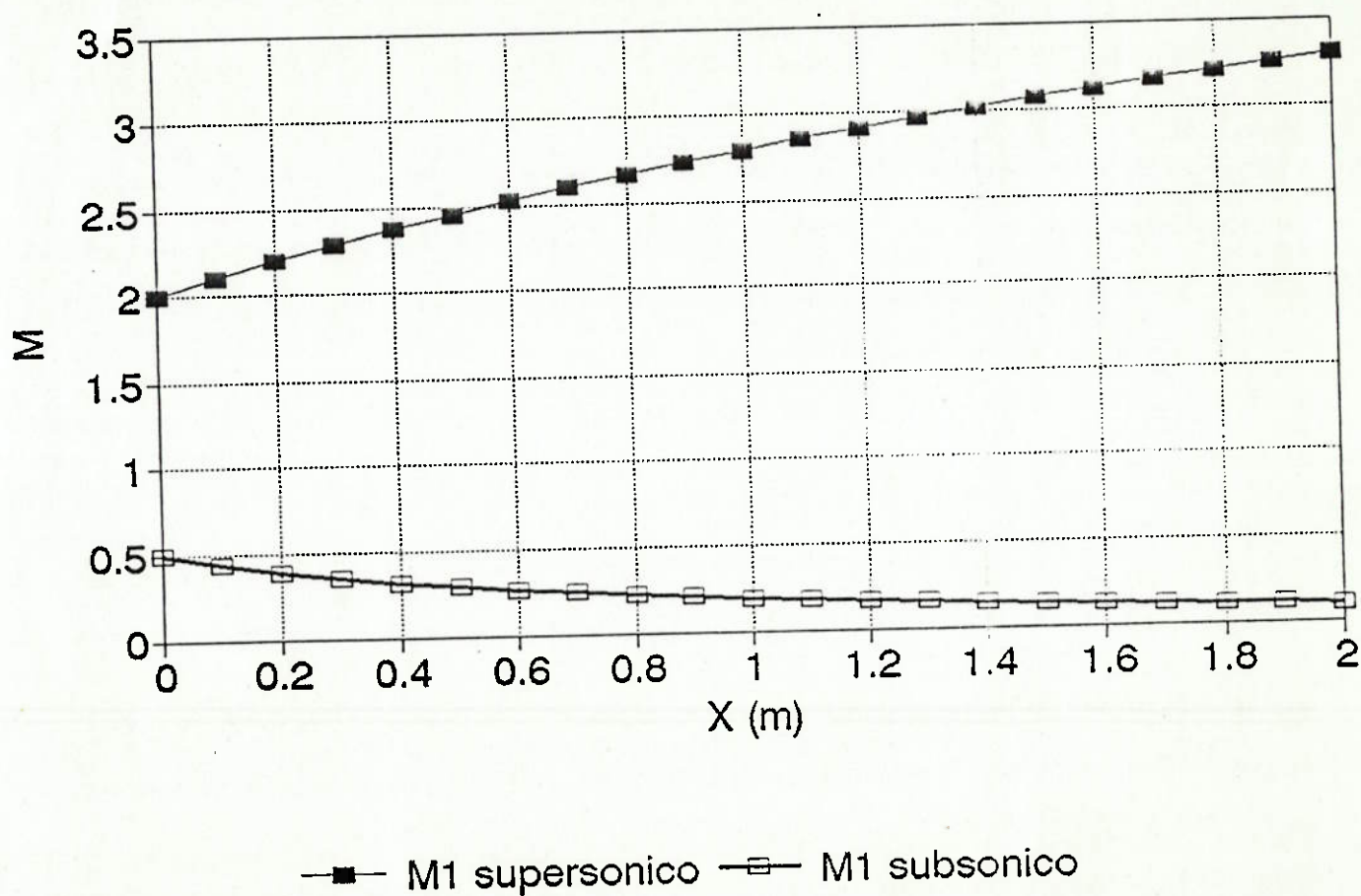


Gráfico 4

Observa-se que o escoamento neste tipo de bocal nunca muda de regime, afastando-se sempre do regime sônico ($M=1$), ou seja, caso o fluido entre a uma velocidade subsônica no bocal, permanecerá subsônico (com Mach diminuindo) e vice-versa.

O gráfico 5 mostra a variação do número de Mach num bocal convergente para regimes de escoamento subsônico e supersônico. As características do bocal e as condições de contorno são apresentados a seguir :

- $k = 1.4$
- $R = 287 \text{ N.m/kg.K}$
- Fator de atrito : $f = 0.005$
- Diâmetro : $D(x) = 0.4 - 0.1.x$
- Comprimento : 2 m
- Condições na entrada do bocal : $P_o = 0.100 \text{ MPa}$
 $T_o = 300 \text{ K}$

Escoamento em Bocal Convergente Numero de Mach

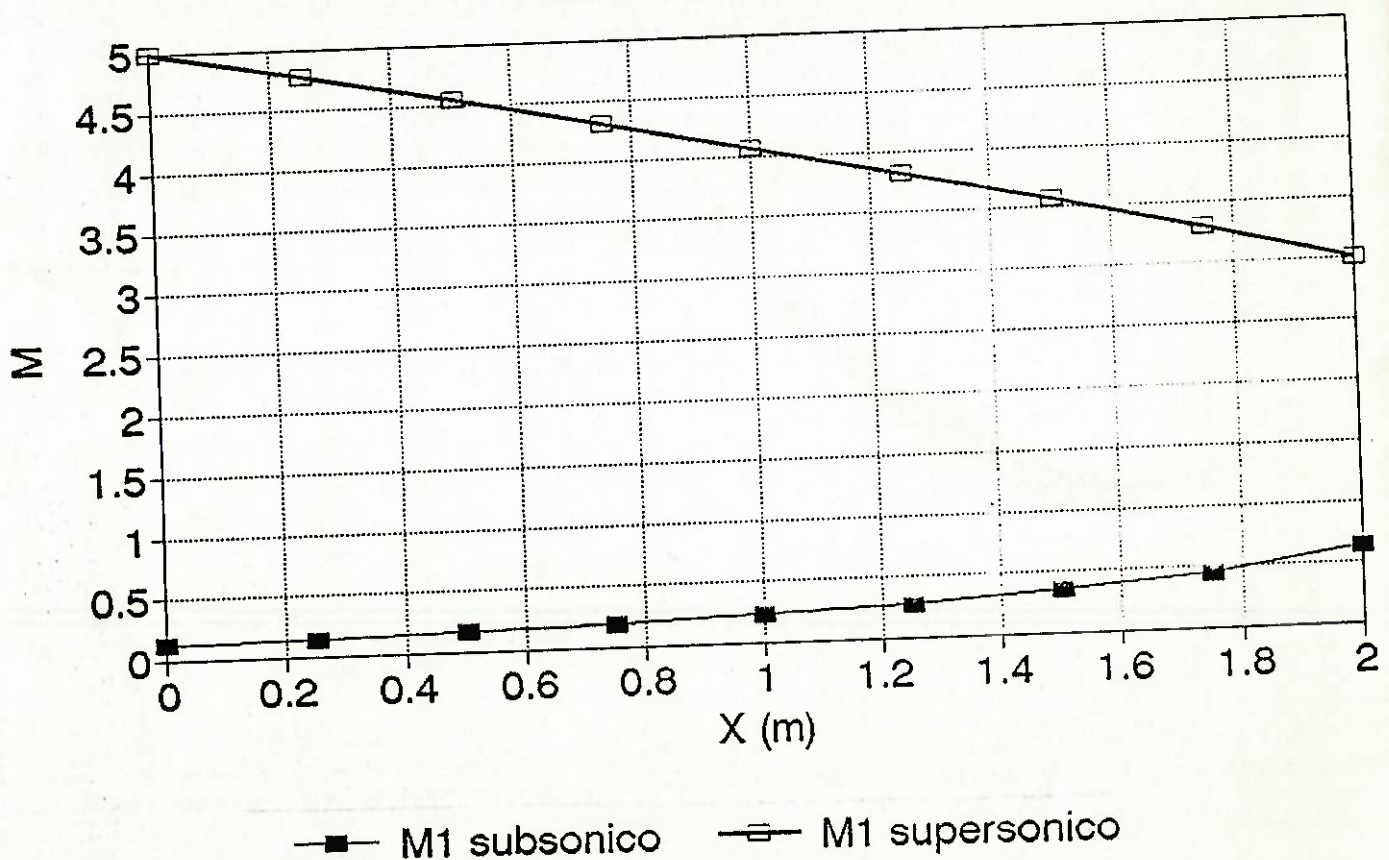


Gráfico 5

Nota-se neste caso que, independentemente do número de Mach de entrada, o escoamento tende a se aproximar do regime sônico ($M=1$), entretanto, é importante ressaltar que esta condição limite poderia ser atingida somente na saída do bocal (quando se diz que o mesmo está *bloqueado*).

Apresenta-se, a seguir, a variação do número de Mach ao longo de um conduto com diâmetro constante, para diferentes fatores de atrito (*Escoamento de Fanno*). Os dados do problema são os seguintes :

- $k = 1.4$
- $R = 287 \text{ N.m/kg.K}$
- Diâmetro do conduto : 0.2 m
- Comprimento do conduto : 5.0 m
- Condições na entrada do conduto : $M = 0.2$
 $P_o = 0.100 \text{ MPa}$
 $T_o = 300 \text{ K}$

Influencia do Atrito no Escoamento Numero de Mach

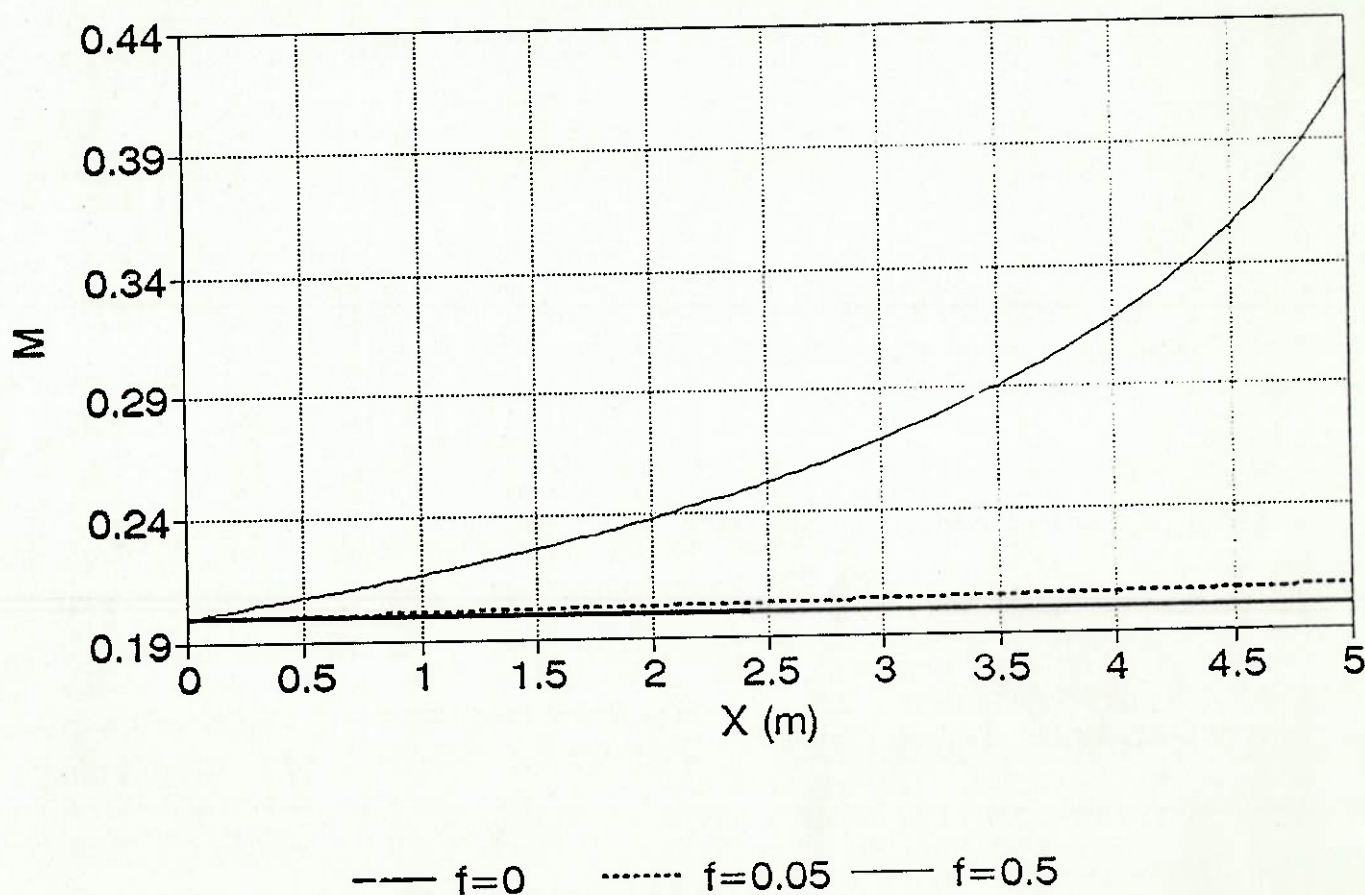


Gráfico 6

Observa-se, através do gráfico 6, que o crescimento do número de Mach ao longo do conduto acentua-se com o aumento do fator de atrito (f), como esperado.

Os gráficos 7 e 8 mostram respectivamente as variações do número de Mach e da temperatura estática ao longo de um conduto de diâmetro constante, para diferentes taxas de transferência de calor. A seguir são apresentadas as características do conduto, bem como as condições de contorno para cada um dos casos :

- $k = 1.4$
- $R = 287 \text{ N.m/kg.K}$
- Comprimento do conduto : 5.0 m
- Diâmetro do conduto : 0.2 m
- Condições na entrada do conduto : $M = 0.5$
 $P_o = 0.500 \text{ MPa}$

Adota-se uma variação linear de temperatura de estagnação para representar a transferência de calor no conduto : $T_o(x) = 300 + C.x$

Caso 1 : $C = 0 \text{ K/m}$ (Escoamento Adiabático)

Caso 2 : $C = 10 \text{ K/m}$

Caso 3 : $C = 20 \text{ K/m}$

Caso 4 : $C = 25 \text{ K/m}$

Observa-se que o aumento da quantidade de calor transferida ao fluido provoca aumento do número de Mach e da temperatura estática na saída do conduto, ou seja, a velocidade do escoamento aumenta. Além disso, nota-se que para escoamentos onde se atinge número de Mach próximo da unidade, a temperatura estática chega a um valor máximo, onde $M = 0.845$ (Caso 4), apresentando uma tendência de queda logo em seguida. Consultando-se uma tabela para escoamento de Rayleigh comprova-se que o ponto de máxima temperatura estática realmente acontece para $M = 0.845$.

Este fato se deve à grande influência do número de Mach sobre a temperatura estática nesta região, já que a temperatura estática é inversamente proporcional ao quadrado do número de Mach e diretamente proporcional à temperatura de estagnação, que aumenta linearmente enquanto M cresce de forma exponencial.

Influencia da Troca de Calor Numero de Mach

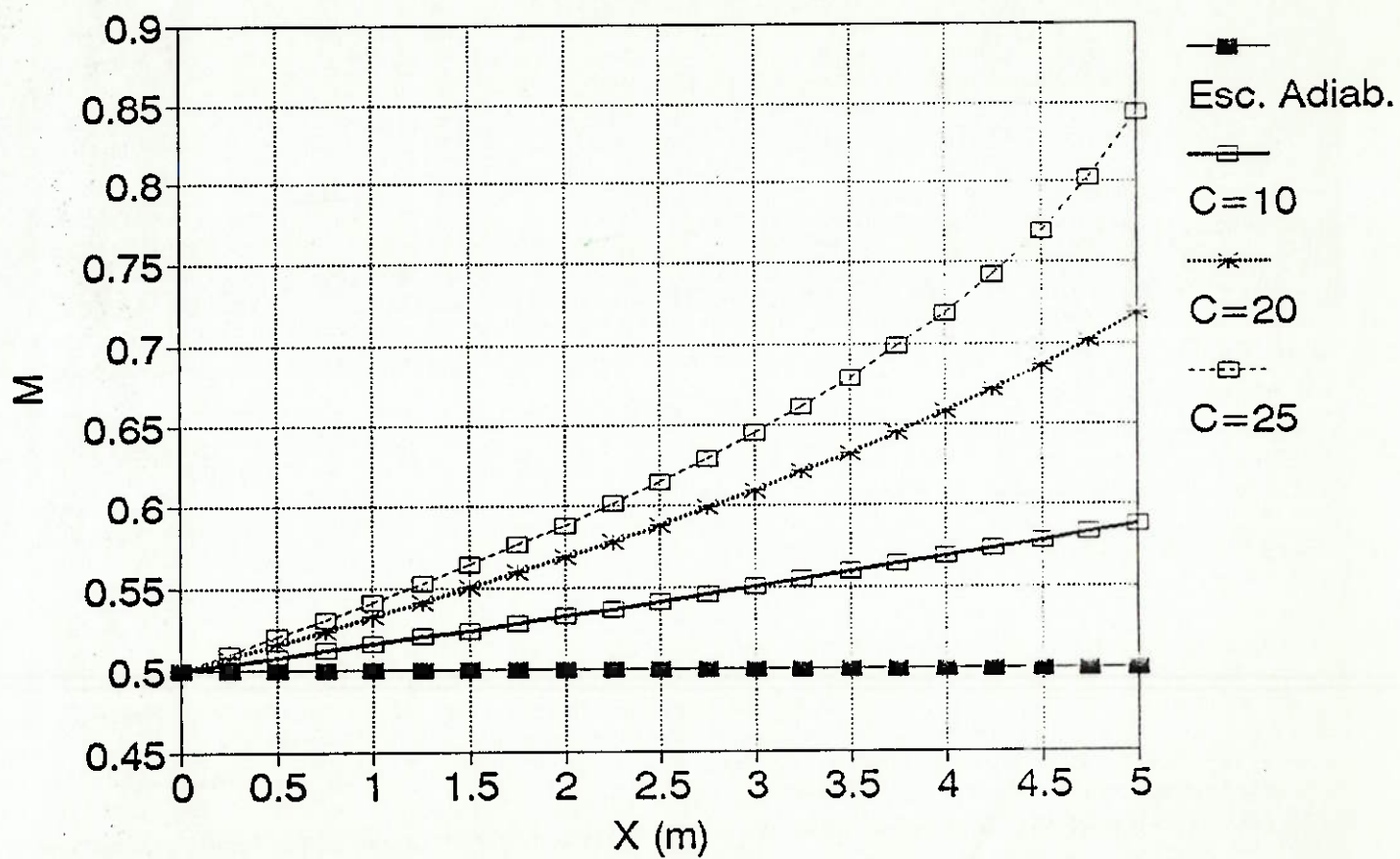


Gráfico 7

Influencia da Troca de Calor Temperatura Estática

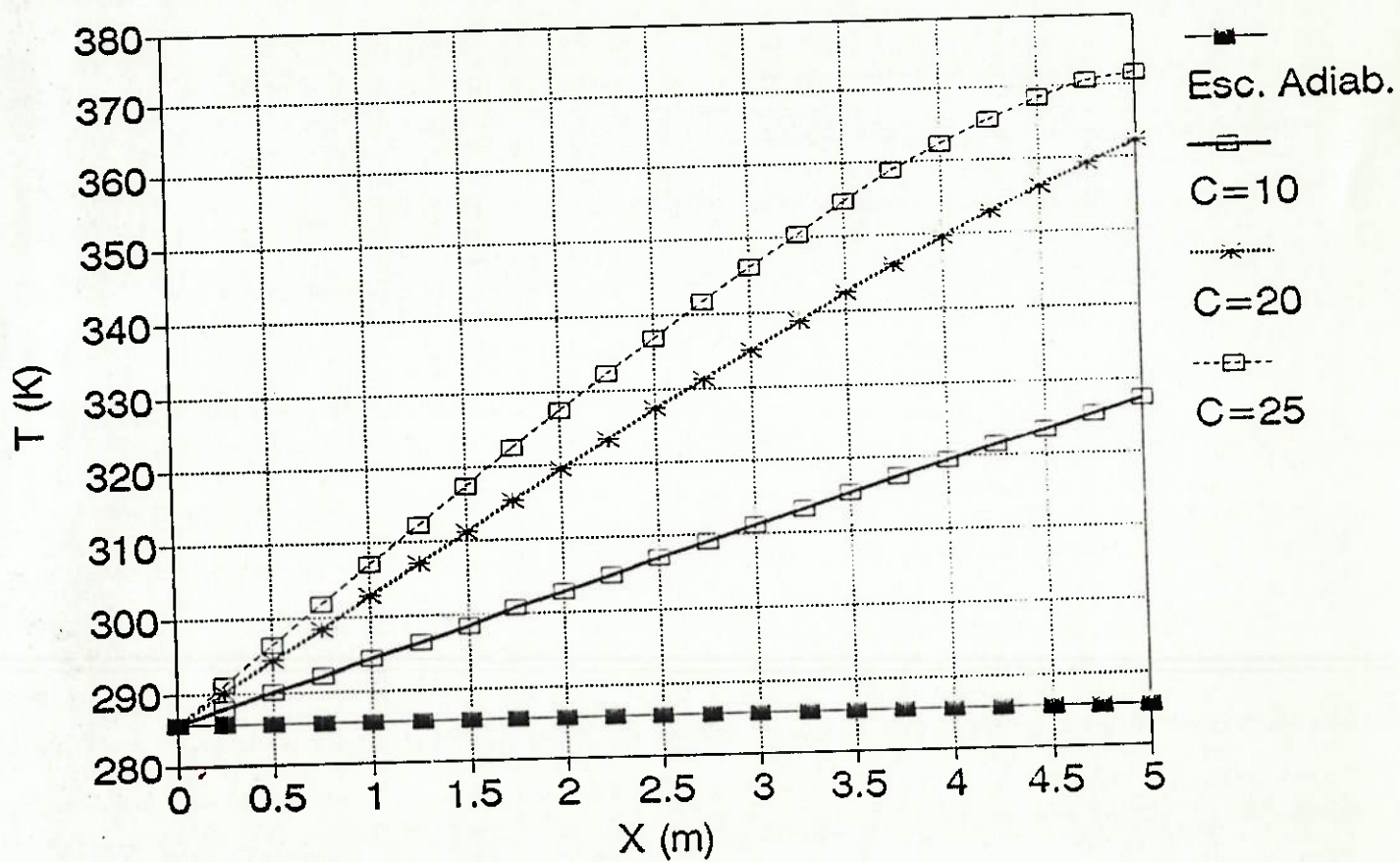


Gráfico 8

Será realizada agora a análise de um bocal convergente com troca de calor. As características do bocal e as condições de contorno são apresentadas a seguir :

- $k = 1.4$
- $R = 287 \text{ N.m/kg.K}$
- Comprimento do bocal : 0.5 m
- Diâmetro do bocal : $D(x) = 0.4 - 0.1.x$
- Condições na entrada do bocal : $P_o = 0.500 \text{ MPa}$
 $M = 0.5$

Adota-se uma variação linear de temperatura de estagnação para representar a transferência de calor no conduto : $T_o(x) = 600 + C.x$

- Caso 1 : $C = 37 \text{ K/m}$
- Caso 2 : $C = 0 \text{ K/m}$ (Escoamento Adiabático)
- Caso 3 : $C = -50 \text{ K/m}$
- Caso 4 : $C = -350 \text{ K/m}$
- Caso 5 : $C = -1100 \text{ K/m}$

Observa-se através do gráfico 9, que a tendência seria o fluido acelerar no bocal convergente caso o escoamento fosse isentrópico (Caso 2), já que temos Mach subsônico na entrada. A adição de calor (Caso 1) acentua o efeito de aceleração sobre o fluido, enquanto que a retirada de calor (Casos 3, 4 e 5) age de forma oposta. Além disso, nota-se que há uma taxa de retirada de calor para qual o escoamento teria número de Mach constante.

Os gráficos 10 e 11 mostram que a adição de calor aumenta a queda de pressão no bocal. Enquanto isso, a retirada de calor atenua o efeito de queda de pressão no mesmo, podendo em casos extremos até aumentar a pressão. No gráfico 12 observa-se que a retirada de calor acentua, como era de se esperar, o efeito de queda de temperatura estática que normalmente ocorreria num bocal convergente.

Desta forma conclui-se que a adição de calor no bocal convergente acentua a queda de pressão e o aumento de Mach, enquanto atenua a queda de temperatura estática. A retirada de calor atua de forma oposta. Entretanto, observa-se que para números de Mach próximos da unidade o efeito de queda da temperatura estática acentua-se, da mesma forma que ocorre num escoamento de Rayleigh.

Bocal Convergente com Troca de Calor

Numero de Mach

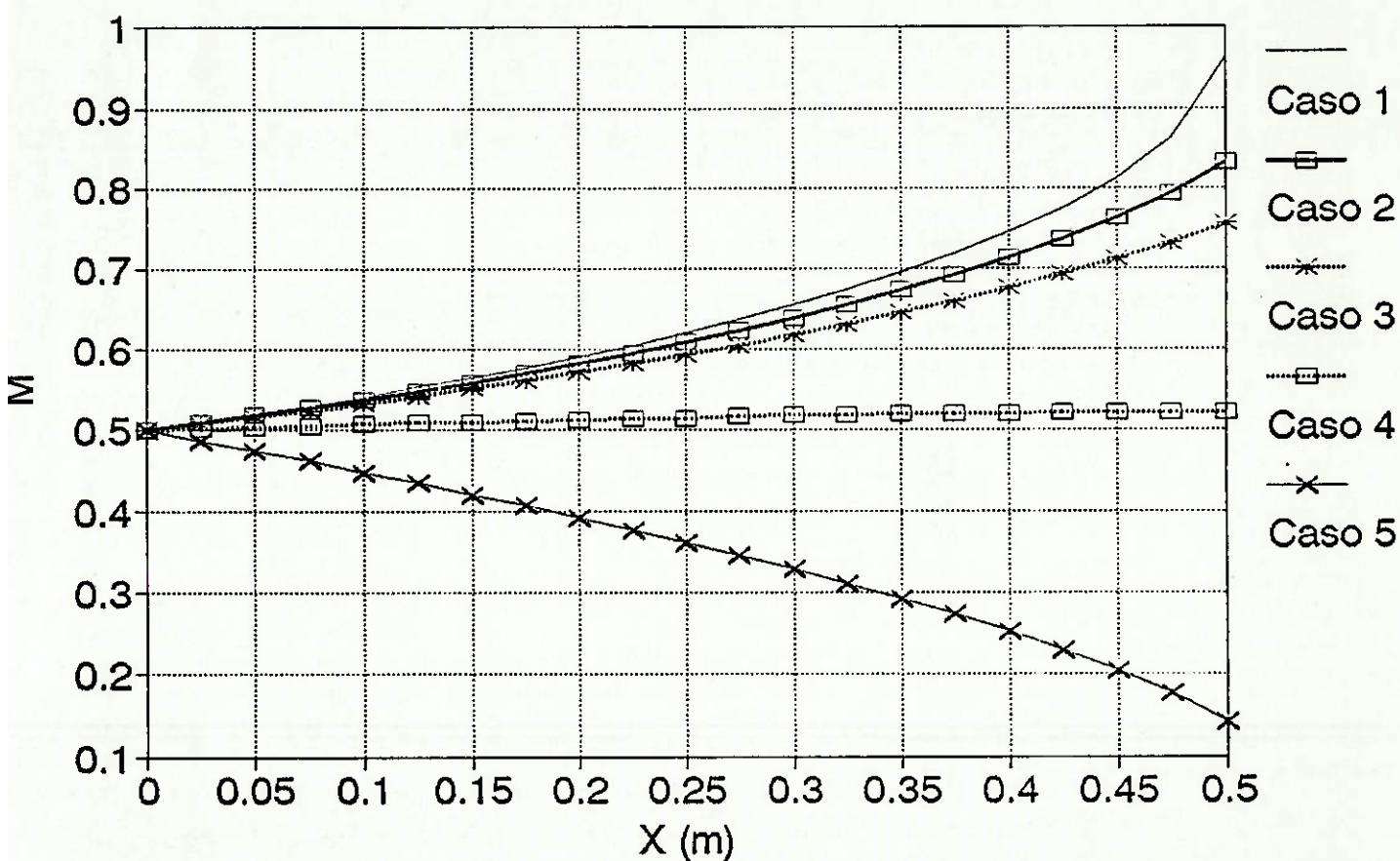


Gráfico 9

Bocal Convergente com Troca de Calor

Pressao Estatica

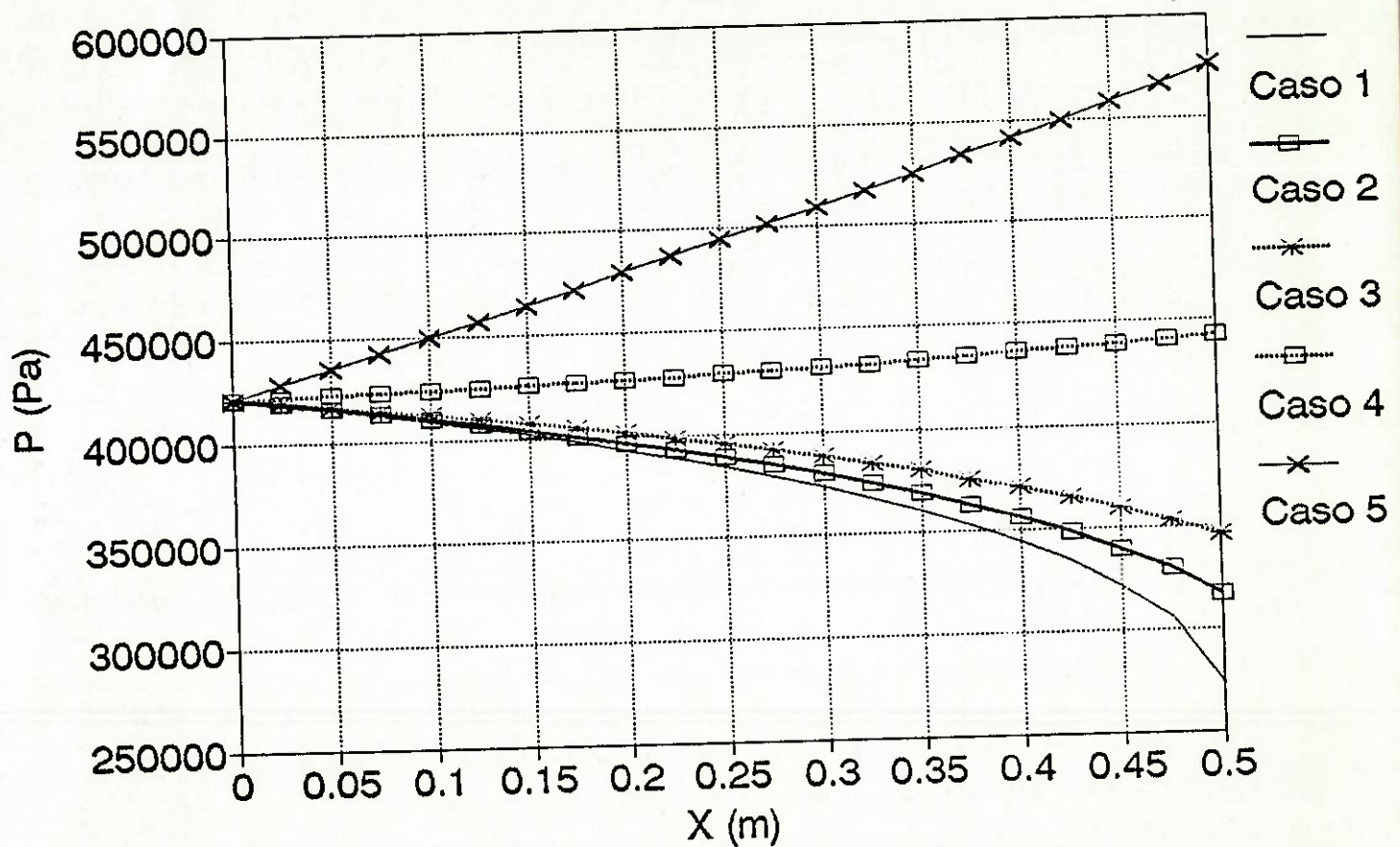


Gráfico 10

Bocal Convergente com Troca de Calor

Pressao de Estagnacao

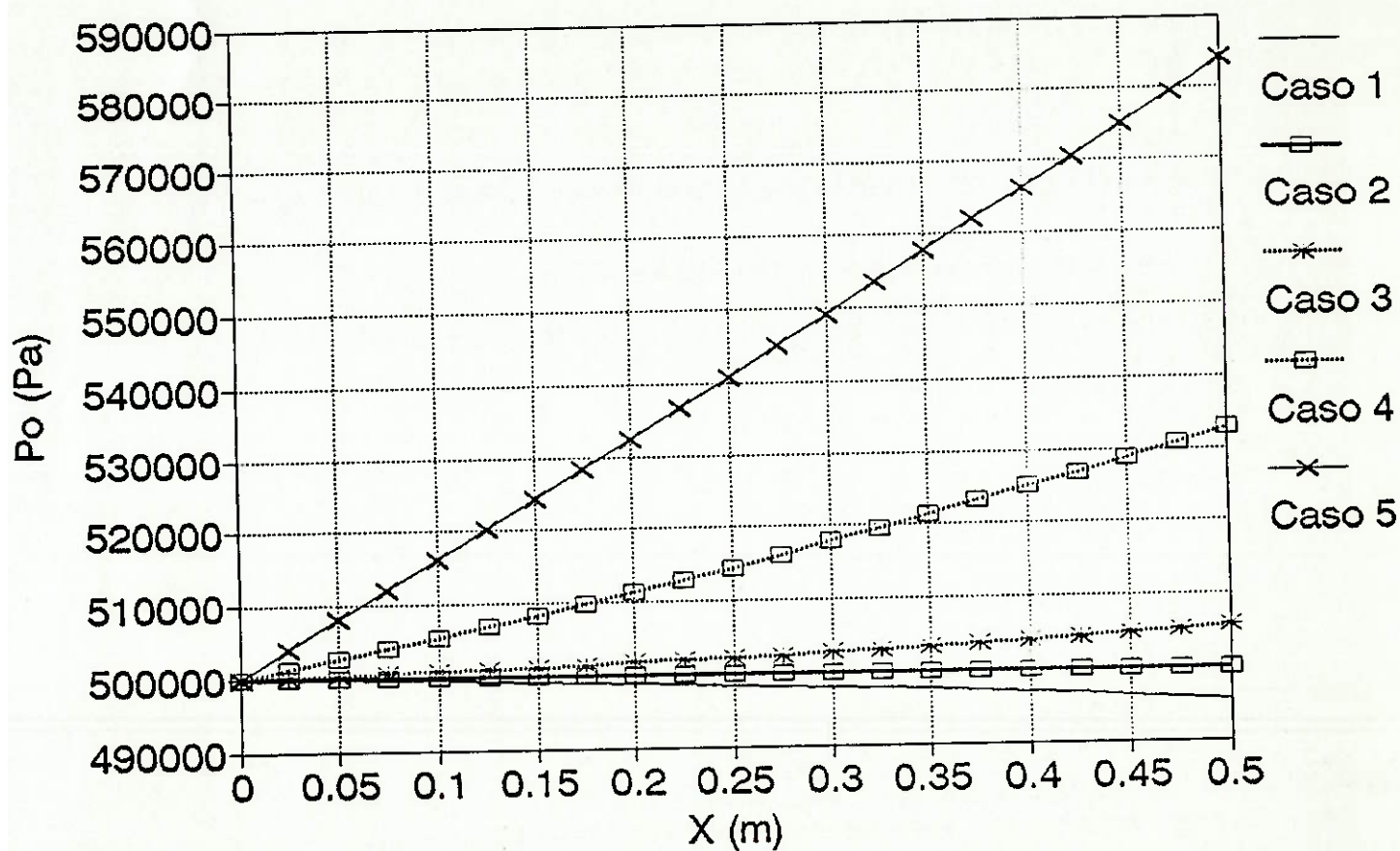


Gráfico 11

Bocal Convergente com Troca de Calor Temperatura Estática

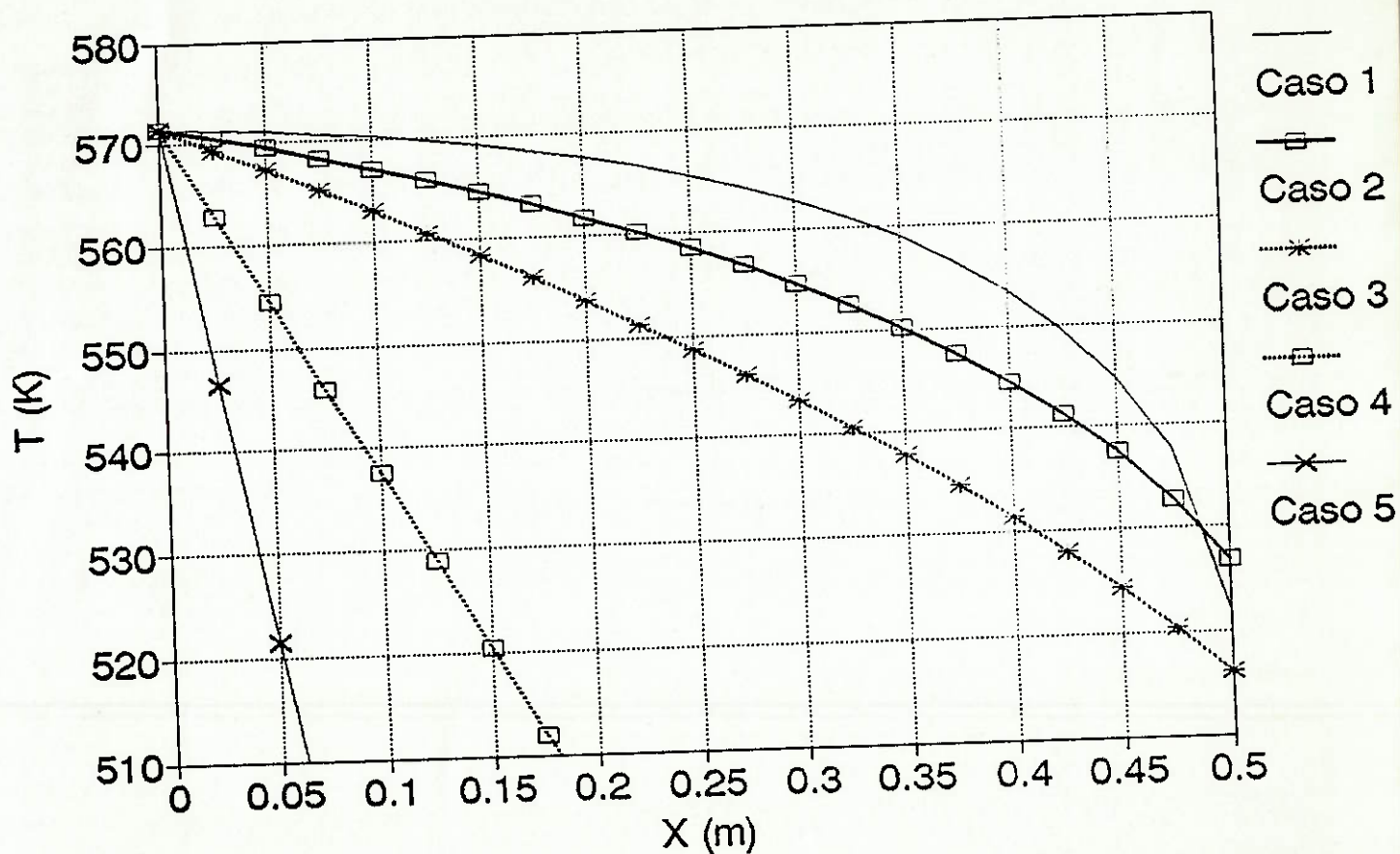


Gráfico 12

Bocal Convergente com Troca de Calor

Temperatura de Estagnacao

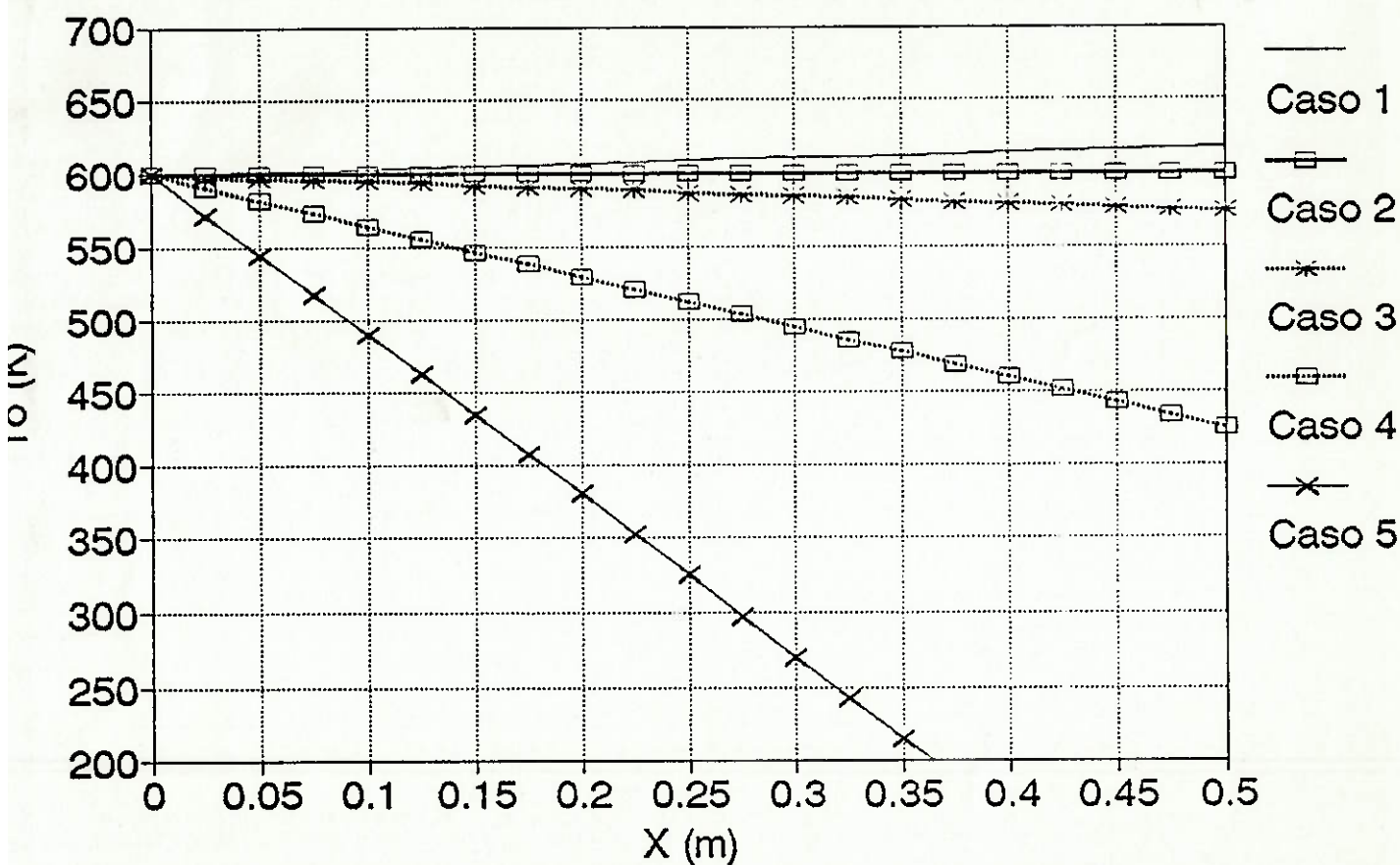


Gráfico 13

O próximo exemplo consiste no estudo do escoamento através de um conduto de diâmetro constante com atrito, no qual varia-se a taxa de transferência de calor. As condições de contorno e as características do fluido e do conduto são apresentadas abaixo:

- $k = 1.4$
- $R = 287 \text{ N.m/kg.K}$
- Comprimento do conduto : 6.0 m
- Diâmetro do conduto : 0.2 m
- Fator de atrito : $f = 0.005$
- Condições na entrada do conduto : $P_o = 0.500 \text{ MPa}$
 $M = 0.5$

Adota-se uma variação linear de temperatura de estagnação para representar a transferência de calor no conduto : $T_o(x) = 600 + C.x$

Caso 1 : $C = 0 \text{ K/m}$ (Escoamento Adiabático)

Caso 2 : $C = 20 \text{ K/m}$

Caso 3 : $C = 35 \text{ K/m}$

Caso 4 : $C = -5 \text{ K/m}$

Caso 5 : $C = -10 \text{ K/m}$

A tendência natural para o escoamento num conduto de diâmetro constante com atrito (escoamento de Fanno) é de que, para números de Mach subsônicos, M aumente, a pressão (estática e de estagnação) e a temperatura estática diminuam ao longo do conduto. Já a temperatura de estagnação permanece constante.

A medida que retira-se calor do fluido, os efeitos provocados pelo atrito tendem a atenuar-se no caso da pressão (gráficos 15 e 16) e do número de Mach (gráfico 14), enquanto que para a temperatura estática (gráfico 17) o efeito se acentua. No caso de adição de calor temos a situação inversa.

Observando os gráficos de temperatura de estática (gráfico 17) ao longo do conduto, nota-se que as taxas de troca de calor que aproximam o sistema da blocagem (Caso 3) fazem com que a temperatura, inicialmente, cresça quase linearmente. Entretanto, quando o número de Mach aproxima-se da unidade há uma visível alteração nesta situação, pois a temperatura estática tende a atingir um máximo, e começa a decrescer após este fato. Efeito semelhante ocorre no escoamento de Rayleigh. Os resultados obtidos neste exemplo podem ser comparados com os obtidos por SHAPIRO [1], comprovando-se a sua validade.

Conduto com Atrito e Troca de Calor Número de Mach

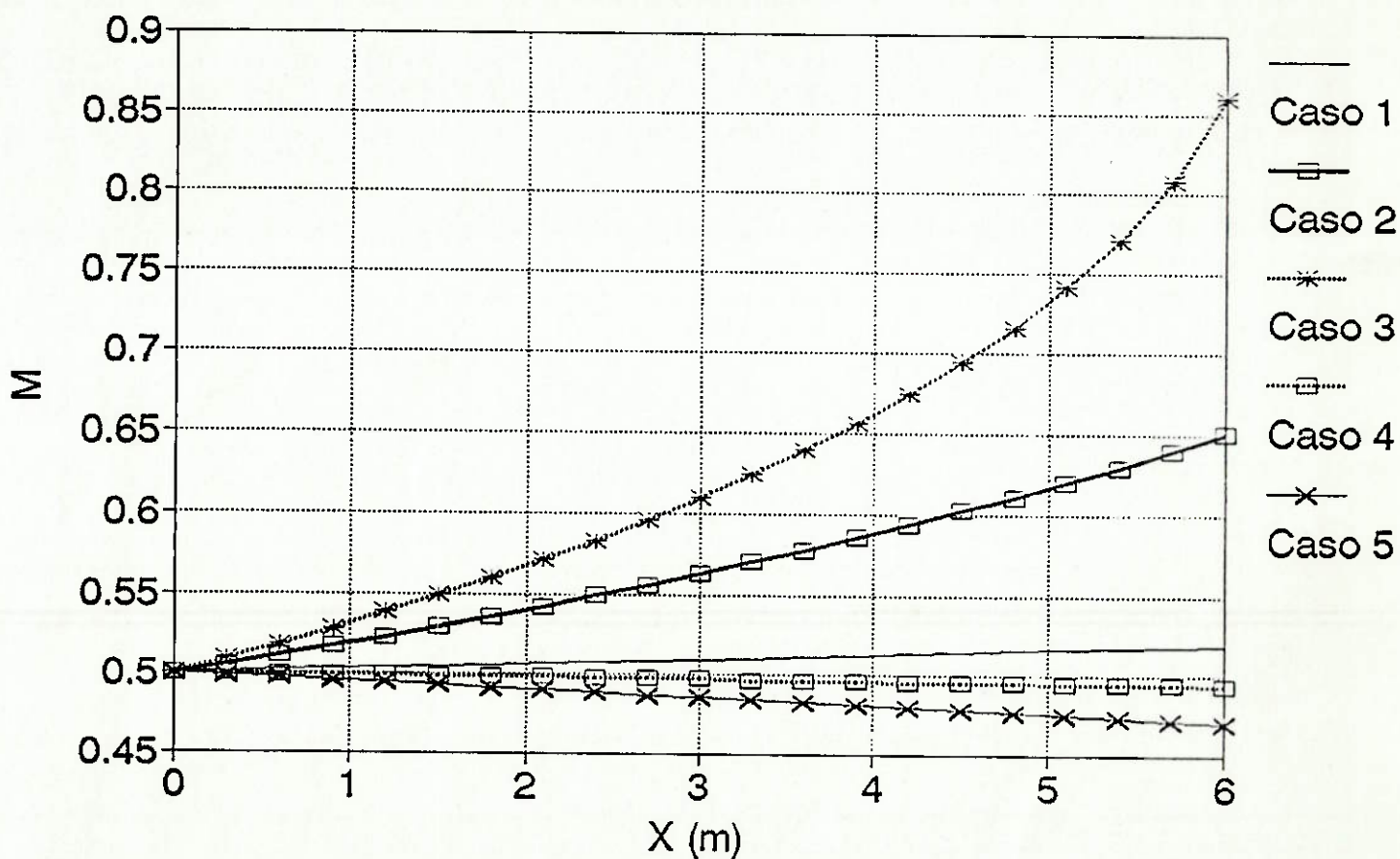


Gráfico 14

Conduto com Atrito e Troca de Calor

Pressao Estatica

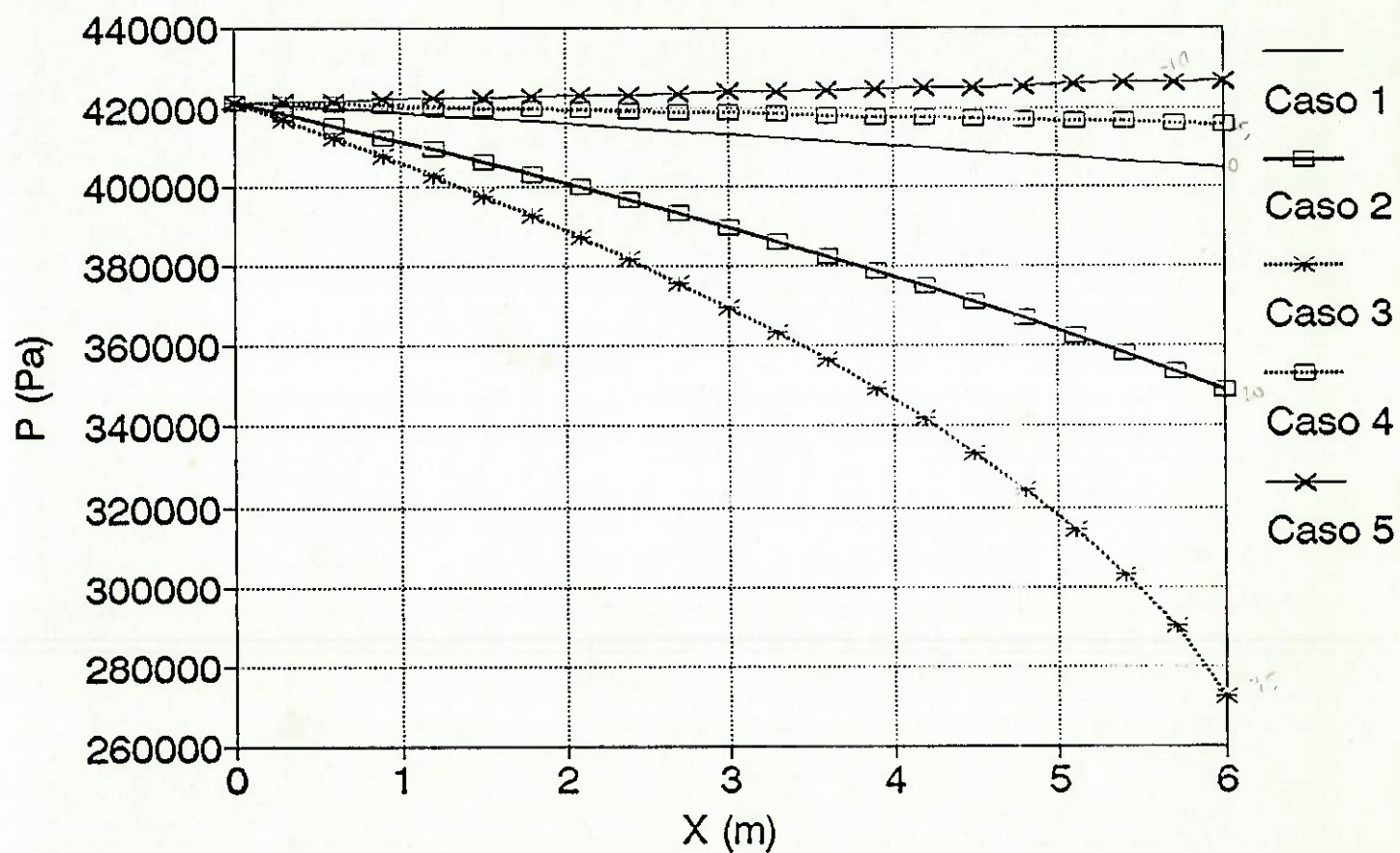


Gráfico 15

Conduto com Atrito e Troca de Calor

Pressao de Estagnacao

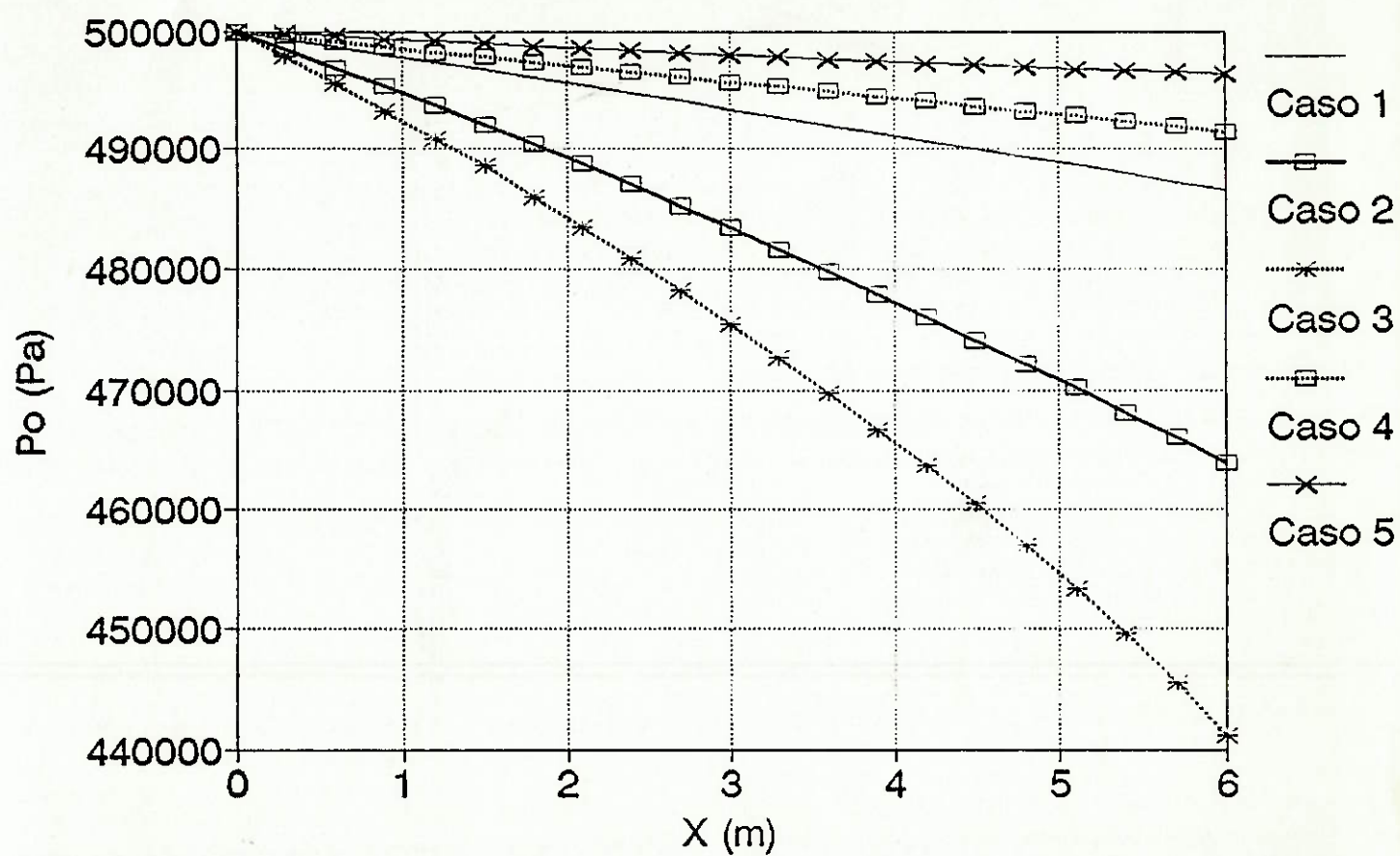


Gráfico 16

Conduto com Atrito e Troca de Calor

Temperatura Estática

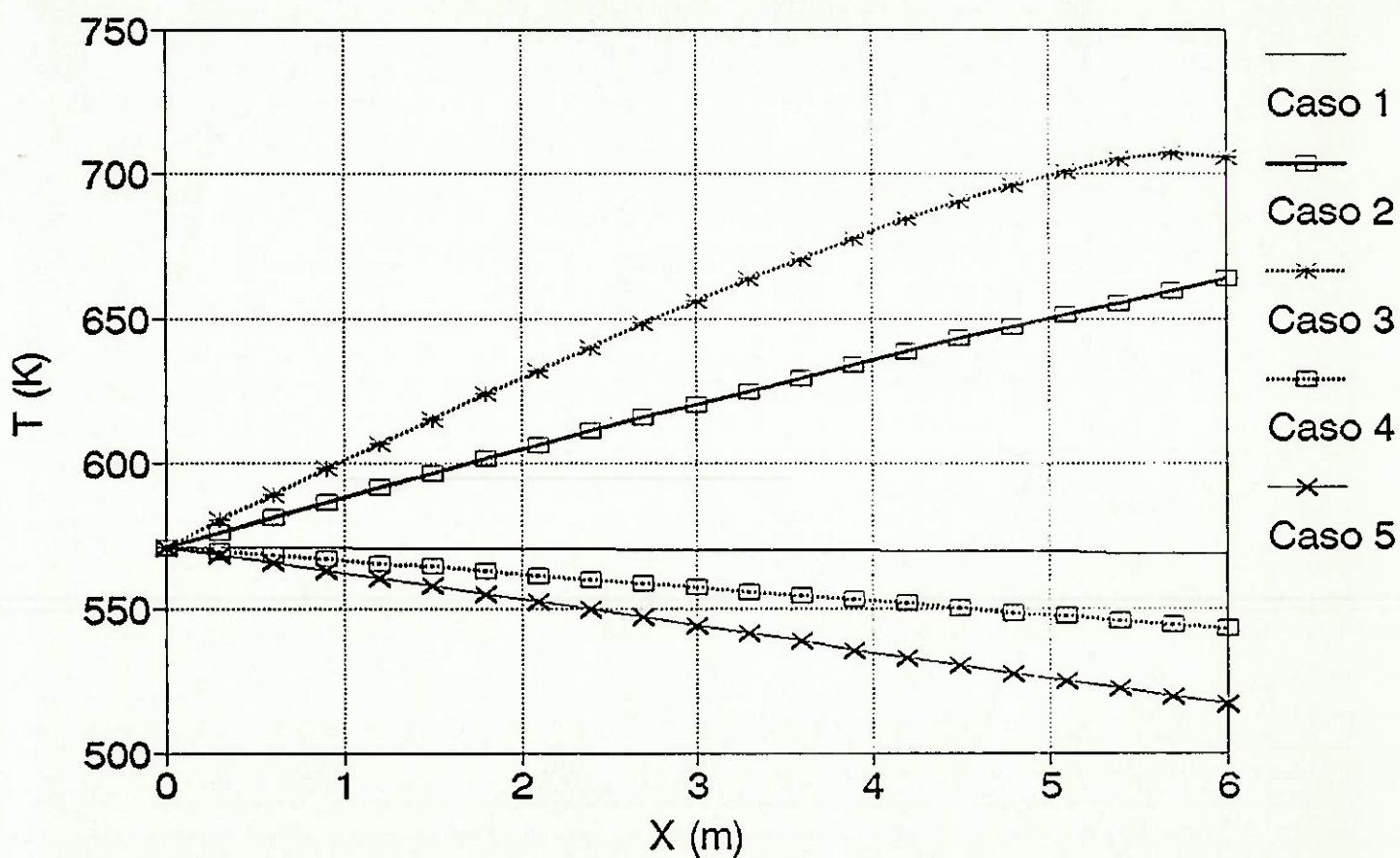


Gráfico 17

Conduto com Atrito e Troca de Calor

Temperatura de Estagnação

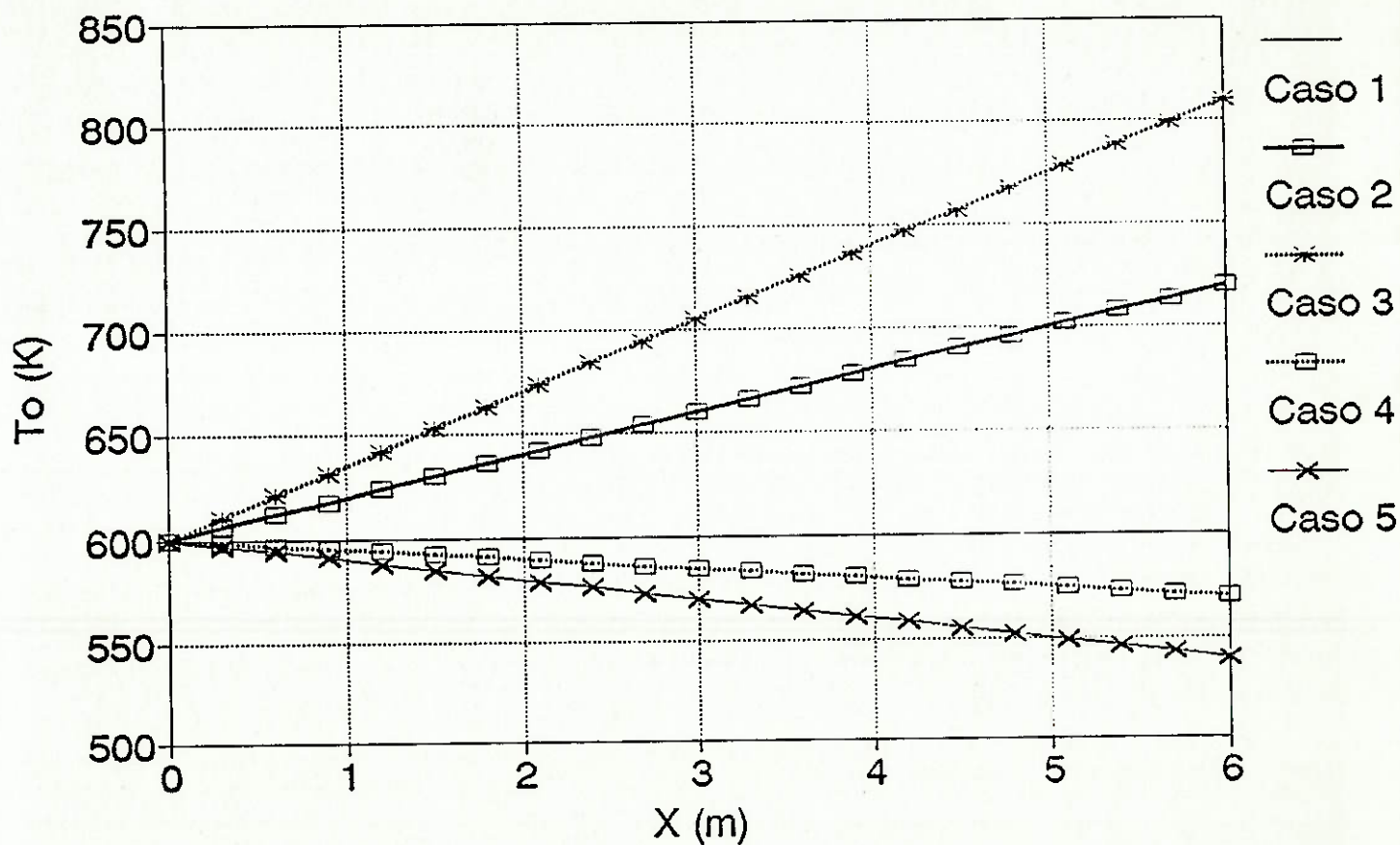


Gráfico 18

O programa também permite o estudo de condutos com configuração mais complexa.

Exemplo 1:**PARÂMETROS DO ESCOAMENTO:**

$$\text{Mach inicial} = 0.22$$

$$\text{Temperatura de estagnação inicial} = 600.00 \text{ K}$$

$$\text{Pressão de estagnação inicial} = 100000.00 \text{ Pa}$$

$$k = 1.40$$

$$R = 287.00 \text{ N.m/kg.K}$$

$$\text{Comprimento do conduto: } X_i = 0.00 \text{ m}$$

$$X_f = 5.00 \text{ m}$$

$$\text{Passo de iteração} = 0.0600 \text{ m}$$

CONDIÇÕES DE CONTORNO:**Trecho 1**

$$\text{Início : } 0.00 \text{ m}$$

$$\text{Fim : } 3.00 \text{ m}$$

$$f = 0.000$$

$$T_o(x) = 0.00 \cdot x^2 + 0.00 \cdot x + 600.00$$

$$D(x) = 0.000 \cdot x^2 + -0.050 \cdot x + 0.400$$

CONDIÇÕES DE CONTORNO:**Trecho 2**

$$\text{Início : } 3.00 \text{ m}$$

$$\text{Fim : } 5.00 \text{ m}$$

$$f = 0.000$$

$$T_o(x) = 0.00 \cdot x^2 + 0.00 \cdot x + 600.00$$

$$D(x) = 0.000 \cdot x^2 + 0.200 \cdot x + 0.250$$

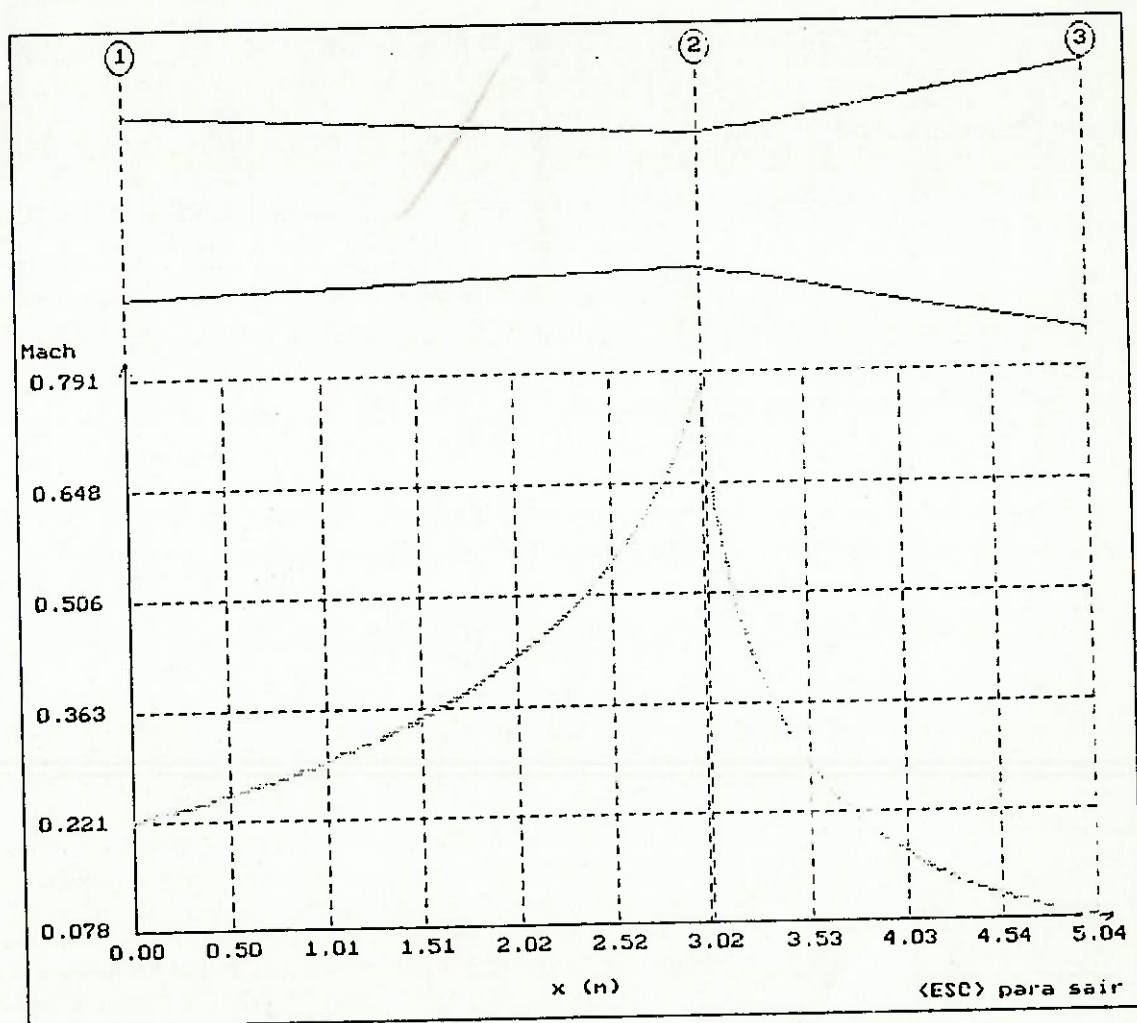


Gráfico 19

Exemplo 2:**PARÂMETROS DO ESCOAMENTO:**

Mach inicial = 1.10
Temperatura de estagnação inicial = 600.00 K
Pressão de estagnação inicial = 100000.00 Pa
 $k = 1.40$
 $R = 287.00 \text{ N.m/kg.K}$

Comprimento do conduto: $X_i = 0.00 \text{ m}$
 $X_f = 5.00 \text{ m}$

Passo de iteração = 0.0069 m

CONDIÇÕES DE CONTORNO:**Trecho 1**

Início : 0.00 m
Fim : 2.00 m
 $f = 0.000$
 $To(x) = 0.00 \cdot x^2 + 0.00 \cdot x + 600.00$
 $D(x) = 0.000 \cdot x^2 + 0.050 \cdot x + 0.400$

CONDIÇÕES DE CONTORNO:**Trecho 2**

Início : 2.00 m
Fim : 3.00 m
 $f = 0.000$
 $To(x) = 0.00 \cdot x^2 + 0.00 \cdot x + 600.00$
 $D(x) = 0.000 \cdot x^2 + 0.000 \cdot x + 0.500$

CONDIÇÕES DE CONTORNO:**Trecho 3**

Início : 3.00 m
Fim : 5.00 m
 $f = 0.000$
 $To(x) = 0.00 \cdot x^2 + 0.00 \cdot x + 600.00$
 $D(x) = 0.000 \cdot x^2 + 0.050 \cdot x + 0.500$

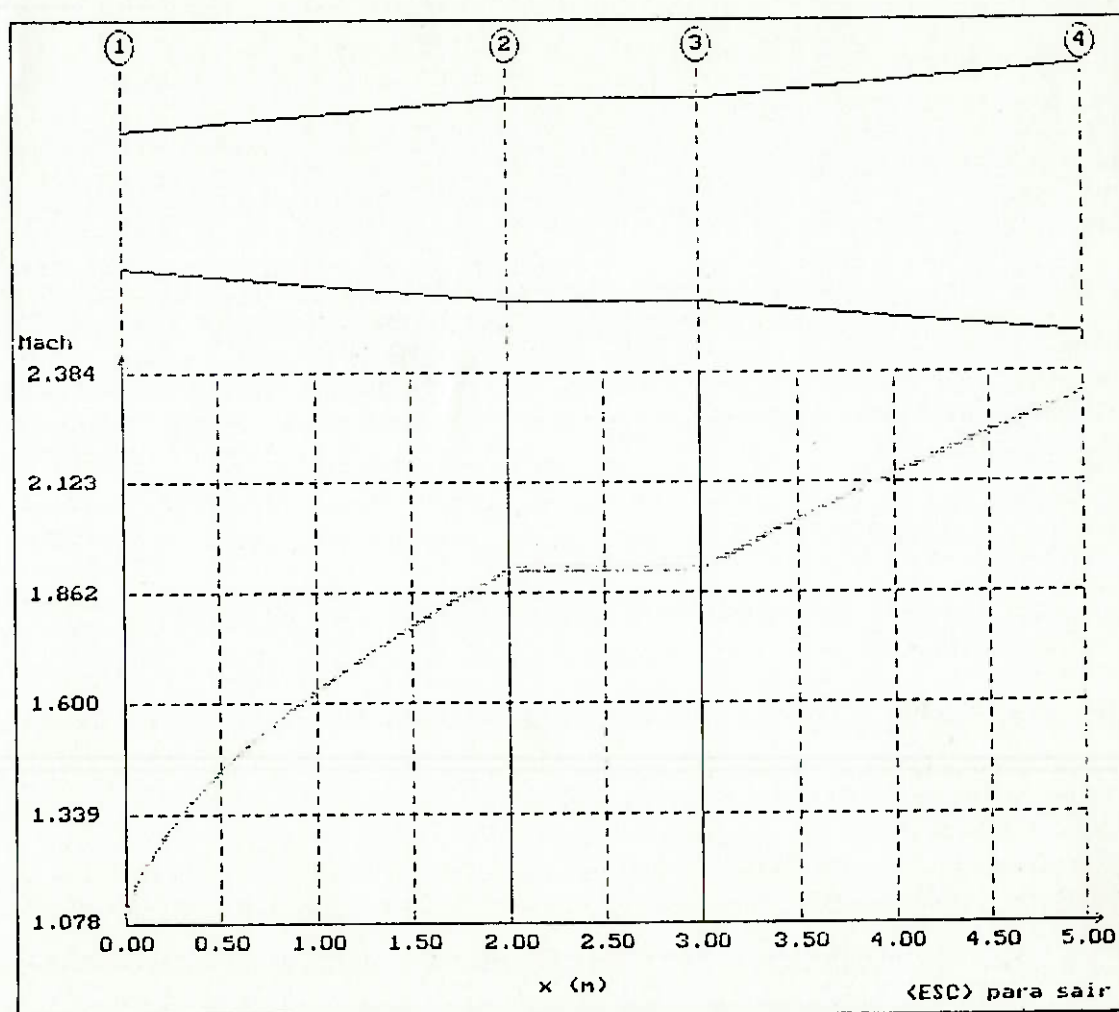


Gráfico 20

Para finalizar, há a opção de se plotar o diagrama $T_x S^7$.

Exemplo 3:

PARÂMETROS DO ESCOAMENTO:

Mach inicial = 0.10

Temperatura de estagnação inicial = 600.00 K

Pressão de estagnação inicial = 500000.00 Pa

$k = 1.40$

$R = 287.00 \text{ N.m/kg.K}$

Comprimento do conduto: $X_i = 0.00 \text{ m}$

$X_f = 6.00 \text{ m}$

Passo de iteração = 0.3000 m

CONDIÇÕES DE CONTORNO:

Trecho 1

Início : 0.00 m

Fim : 6.00 m

$f = 0.005$

$T_o(x) = 5.00 \cdot x^2 + -10.00 \cdot x + 600.00$

$D(x) = 0.000 \cdot x^2 + 0.000 \cdot x + 0.200$

⁷Os gráficos dos exemplos 1, 2 e 3 (gráficos 19, 20 e 21 respectivamente) foram imprimidos a partir da tela gráfica do programa.

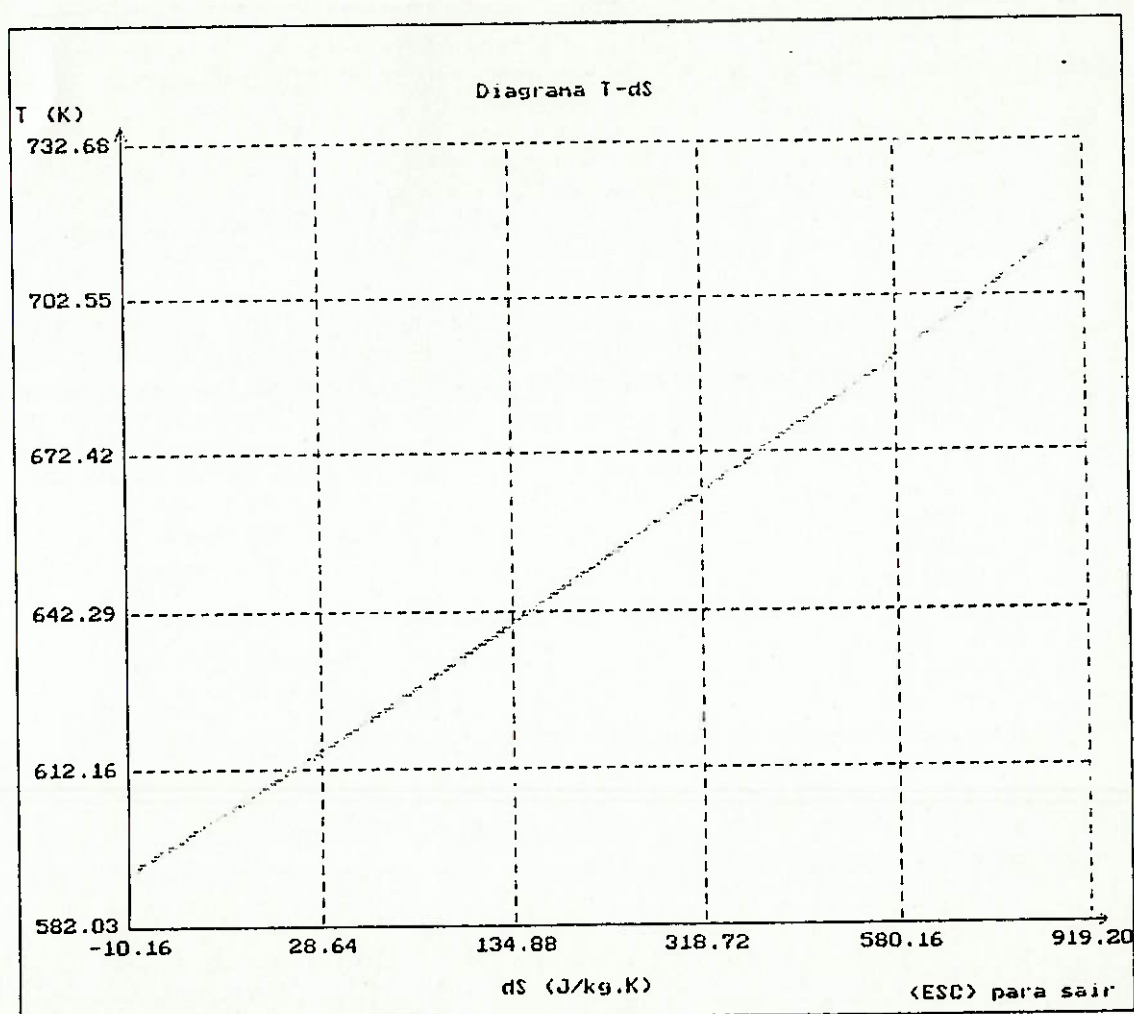


Gráfico 21

6. COMENTÁRIOS E CONCLUSÕES

Observa-se que os resultados obtidos com o programa são bastante próximos dos calculados teoricamente, o que mostra a eficácia do algoritmo implementado.

Devem ser feitas algumas considerações a respeito do programa:

- Foi adotada uma estrutura de dados estática (vetores de números reais) para o armazenamento dos valores calculados, isto implica numa limitação em termos de memória. Desta forma restringe-se o passo de interação que pode ser adotado, diminuindo a precisão dos resultados. Com o objetivo de solucionar este problema, pretende-se substituir a estrutura de dados estática por uma estrutura de alocação dinâmica (pointers).
- Após feita uma simulação, para se modificar um determinado parâmetro ou condição de contorno e refazer os cálculos é necessário redigitar quase todos os valores iniciais. Uma melhoria a ser feita seria colocar estes valores na tela na forma de células de uma planilha e, então, através do *mouse* ou de movimentos do cursor via teclado, se alterar apenas o parâmetro desejado.
- O gráfico TxdS considera nula a entropia inicial e, portanto, para se obter o valor da entropia em uma determinada posição é necessário acrescentar o valor da entropia para as condições iniciais do escoamento ao calculado. Apesar disto, o resultado qualitativo é representativo do caso analisado. A função implementada funciona perfeitamente para condutos simples (um único trecho). Para condutos mais complexos, a rotina não consegue apresentar dados compatíveis com o fenômeno. Uma solução seria plotar um diagrama para cada trecho do conduto.
- Os gráficos utilizados para as análises que contém mais de uma curva foram feitos com o auxílio de uma planilha. Isto causa certa perda de tempo pois é necessário sair do programa para se plotar tais gráficos. O ideal seria implementar5 rotinas para permitir que isso fosse feito dentro do próprio programa de simulação.

- Como o programa é baseado em uma solução numérica de equação diferencial, a escolha do passo é muito importante. Passos menores produzem resultados melhores, porém, às vezes, diminuir excessivamente o passo não trará grandes vantagens para a análise dos valores calculados, além de consumir um maior tempo de processamento e memória. Além disso, dependendo do passo pode haver o aparecimento de alguma descontinuidade nos gráficos. Isto ocorre principalmente quando o passo coincide ou se aproxima muito do ponto de mudança de um trecho para outro. Para se evitar este problema, pode-se variar um pouco o valor do passo para que os cálculos contornem estes pontos de instabilidade.

- Para estudo de problemas mais complexos os autores sugerem que se implemente um equacionamento mais abrangente, no qual considera-se a variação do calor específico e do peso molecular do fluido ao longo do escoamento. Este equacionamento é apresentado em detalhes em SHAPIRO [1]. Além disso, a inclusão de choques normais também seria de grande interesse.

Apesar de todas as modificações que ainda podem ser feitas, é inegável a grande utilidade do software desenvolvido para o estudo de escoamentos unidimensionais generalizados, pois este propicia ao usuário a obtenção de resultados de forma rápida e precisa, inclusive com a apresentação de gráficos na impressora ou na tela do computador.

ANEXO 1: MÉTODO DE RUNGE-KUTTA

Existem vários métodos numéricos para integração de equações diferenciais. Neste programa será utilizado o algoritmo de Runge-Kutta. Dada uma função $y(x)$, os Métodos de Runge-Kutta têm a seguinte forma:

$$y_{i+1} = y_i + h\phi(x_i, y_i, h) \quad i = 0, 1, 2, \dots$$

onde:

iinteração

hpasso de interação

ϕfunção incremento

Para uma melhor aproximação será utilizado o Método de Runge-Kutta de Ordem 4. As fórmulas para o algoritmo de 4ª Ordem são do tipo

$$y_{i+1} = y_i + h(ak_1 + bk_2 + ck_3 + dk_4)$$

A fórmula atribuída a Kutta, que aqui será usada, tem a seguinte forma:

Dado $\frac{dy}{dx} = f(x, y)$

A solução é:

$$y_{i+1} = y_i + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

onde:

$$k_1 = f(x_i, y_i)$$

$$k_2 = f\left(x_i + \frac{h}{2}, y_i + \frac{k_1 h}{2}\right)$$

$$k_3 = f\left(x_i + \frac{h}{2}, y_i + \frac{k_2 h}{2}\right)$$

$$k_4 = f(x_i + h, y_i + k_3 h)$$

Outros detalhes podem ser vistos em CARNAHAN [8].

ANEXO 2: LISTAGEM DO PROGRAMA

PROGRAM Trabalho_de_Formatura;

USES Graficus, Calculus, Dos, Crt;

VAR

Nome,
Caminho : string;
ch : char;
freq : byte;
index : integer;

Procedure Salva (Caminho, Nome: string; pontos: integer; frequencia: byte);

Var i, j : integer;

arq : Text;

arq2 : File of Real;

Begin

Cursor(1);

x[0]:=xi;

i:=0;

j:=0;

assign(arq2, Caminho+Nome+'.VAL');

rewrite(arq2);

gotoxy(30,13);

write('Salvando');

repeat

gotoxy(39,13);

write(Nome, '...');

write(arq2, x[i]);

write(arq2, M[i]);

write(arq2, P0[i]);

write(arq2, P[i]);

write(arq2, T0[i]);

write(arq2, T[i]);

write(arq2, A[i]);

write(arq2, s[i]);

i:=i+frequencia;

j:=j+1;

until (i>pontos);

close(arq2);

assign(arq, Caminho+Nome+'.DAT');

rewrite(arq);

writeln(arq);

writeln(arq);

writeln(arq, 'ARQUIVO : ', Nome+'.DAT');

writeln(arq);

writeln(arq, 'PARAMETROS DO ESCOAMENTO:');

writeln(arq);

writeln(arq, 'Mach inicial = ', M[0]:5:2;

writeln(arq, 'Temperatura de estagnacao inicial = ', T0[0]:5:2, ' K');

writeln(arq, 'Pressao de estagnacao inicial = ', P0[0]:8:2, ' Pa');

writeln(arq, 'k = ', k:4:2;

writeln(arq, 'R = ', R:5:2, ' N.m/kg.K');

```

writeln(arq);
writeln(arq,'      Comprimento do conduto: Xi = ',Xi:5:2,' m');
writeln(arq,'      Xf = ',Xf:5:2,' m');
writeln(arq);
writeln(arq,'      Passo de iteracao = ',h:6:4,' m');
for i:=1 to trechos do
  begin
    writeln(arq);
    writeln(arq,'CONDICOES DE CONTORNO:');
    writeln(arq);
    writeln(arq,'      Trecho ',i);
    writeln(arq);
    writeln(arq,'      Inicio : ',x1[i]:5:2,' m');
    writeln(arq,'      Fim    : ',x2[i]:5:2,' m');
    writeln(arq,'      f = ',f[i]:5:3);
    writeln(arq,'      To(x) = ',A2[i]:5:2,'*x^2 + ',B2[i]:5:2,'*x + ',C2[i]:5:2);
    writeln(arq,'      D(x) = ',A1[i]:5:3,'*x^2 + ',B1[i]:5:3,'*x + ',C1[i]:5:3);
    writeln(arq,'      dm/dx = ',A3[i]:5:2,'*x + ',B3[i]:5:2);
  end;
close(arq);
assign(arq,Caminho+Nome+'.CON');
rewrite(arq);
writeln(arq,j);
writeln(arq,M[0]);
writeln(arq,T0[0]);
writeln(arq,P0[0]);
writeln(arq,k);
writeln(arq,R);
writeln(arq,Xi);
writeln(arq,Xf);
writeln(arq,h);
writeln(arq,trechos);
for i:=1 to trechos do
  begin
    writeln(arq,A1[i]);
    writeln(arq,A2[i]);
    writeln(arq,A3[i]);
    writeln(arq,B1[i]);
    writeln(arq,B2[i]);
    writeln(arq,B3[i]);
    writeln(arq,C1[i]);
    writeln(arq,C2[i]);
    writeln(arq,X1[i]);
    writeln(arq,X2[i]);
    writeln(arq,f[i]);
  end;
close(arq);
gotoxy(30,15);
write(^g,'Arquivo salvo');
Delay(1000);
Cursor(0);
End;

Procedure Recupera(Caminho,Nome:string);
var i,j,code : integer;
    aux      : string;
    arq      : Text;

```

```
    arq2    : File of real;
Begin
  assign(arq, Caminho+Nome+'.CON');
  reset(arq);
  readln(arq, aux);
  Val(aux, j, code);
  readln(arq, aux);
  Val(aux, M[0], code);
  readln(arq, aux);
  Val(aux, T0[0], code);
  readln(arq, aux);
  Val(aux, P0[0], code);
  readln(arq, aux);
  Val(aux, k, code);
  readln(arq, aux);
  Val(aux, R, code);
  readln(arq, aux);
  Val(aux, Xi, code);
  readln(arq, aux);
  Val(aux, Xf, code);
  readln(arq, aux);
  Val(aux, h, code);
  readln(arq, aux);
  Val(aux, trechos, code);
  for i:=1 to trechos do
    begin
      readln(arq, aux);
      Val(aux, A1[i], code);
      readln(arq, aux);
      Val(aux, A2[i], code);
      readln(arq, aux);
      Val(aux, A3[i], code);
      readln(arq, aux);
      Val(aux, B1[i], code);
      readln(arq, aux);
      Val(aux, B2[i], code);
      readln(arq, aux);
      Val(aux, B3[i], code);
      readln(arq, aux);
      Val(aux, C1[i], code);
      readln(arq, aux);
      Val(aux, C2[i], code);
      readln(arq, aux);
      Val(aux, X1[i], code);
      readln(arq, aux);
      Val(aux, X2[i], code);
      readln(arq, aux);
      Val(aux, f[i], code);
    end;
  close(arq);
  assign(arq2, Caminho+Nome+'.VAL');
  reset(arq2);
  for i:=0 to j-1 do
    begin
      read(arq2, x[i]);
      read(arq2, M[i]);
      read(arq2, P0[i]);
```

```

    read(arq2,P[i]);
    read(arq2,T0[i]);
    read(arq2,T[i]);
    read(arq2,A[i]);
    read(arq2,s[i]);
    end;
close(arq2);
index:=j-1;
End;

```

Procedure Planilha(Caminho,Nome:string;pontos:integer;frequencia:byte);

```

var i,j : integer;
    arq,
    arq1,
    arq2,
    arq3,
    arq4 : Text;
Begin
    Cursor(1);
    x[0]:=xi;
    i:=0;
    j:=0;
    assign(arq,Caminho+Nome+'.M');
    assign(arq1,Caminho+Nome+'.P0');
    assign(arq2,Caminho+Nome+'.P');
    assign(arq3,Caminho+Nome+'.T0');
    assign(arq4,Caminho+Nome+'.T');
    rewrite(arq);
    rewrite(arq1);
    rewrite(arq2);
    rewrite(arq3);
    rewrite(arq4);
    gotoxy(30,13);
    write('Salvando');
    repeat
        gotoxy(39,13);
        write(Nome,'...');
        writeln(arq,x[i]:8:5,' ',M[i]:8:5);
        writeln(arq1,x[i]:8:5,' ',P0[i]:10:2);
        writeln(arq2,x[i]:8:5,' ',P[i]:10:2);
        writeln(arq3,x[i]:8:5,' ',T0[i]:10:3);
        writeln(arq4,x[i]:8:5,' ',T[i]:10:3);
        i:=i+frequencia;
        j:=j+1;
    until (i>pontos);
    close(arq);
    close(arq1);
    close(arq2);
    close(arq3);
    close(arq4);
    Cursor(0);
End;

```

Procedure EntradaDeDados;

```

var i : integer;
Begin

```

```

y:=1;
if (k<>0)
  then begin
    Clrscr;
    gotoxy(30,12);
    write('Novo Caso ? <S/N> ');
    repeat
      ch:=Ucase(readkey);
    until (ch='S') or (ch='N');
    end
  else ch:='S';
if (ch='S')
  then begin
    Clrscr;
    writeln('PARAMETROS DO ESCOAMENTO');
    writeln;
    write('      Mach inicial = ');
    readln(M[0]);
    write('      Pressao de estagnacao inicial [Pa] = ');
    readln(P0[0]);
    write('      k = ');
    readln(k);
    write('      R [N.m/kg.K] = ');
    readln(R);
    write('      Xi [m] = ');
    readln(xi);
    write('      Xf [m] = ');
    readln(xf);
    write('      Passo (passo min. = '1.01*(xf-xi)/MaxPont:6:4,') [m] = ');
    readln(h);
    writeln;
    if (h<(1.01*(xf-xi)/MaxPont))
      then repeat
        write('      Passo incorreto. Digite novo valor : ');
        readln(h);
        writeln;
      until (h>(1.01*(xf-xi)/MaxPont));
    write('      Numero de Trechos (no max ',maxtrec,') : ');
    readln(trechos);
    if (trechos>maxtrec)
      then repeat
        writeln;
        write('      Numero de trechos incorreto. Digite novo valor : ');
        readln(trechos);
      until (trechos<=maxtrec);
    for i:=1 to trechos do
      begin
        A1[i]:=0; A2[i]:=0; A3[i]:=0;
        B1[i]:=0; B2[i]:=0; B3[i]:=0;
        C1[i]:=0; C2[i]:=0;
        f[i]:=0;
      end;
    for i:=1 to trechos do
      begin
        Clrscr;
        writeln('CONDICOES DE CONTORNO');
        if i=1

```

```

    then x1[i]:=xi
    else x1[i]:=x2[i-1];
if i=trechos
    then x2[i]:=xf
    else begin
        writeln;
        write('      Final do trecho ',i,' : ');
        readln(x2[i]);
    end;
writeln;
writeln('      Fator de atrito (Trecho ',i,')');
writeln;
writeln('      f = ',f[i]:5:5);
writeln;
write('      Deseja alterar ? (s/n)');
ch:=readkey;
if (ch='S') or (ch='s')
    then begin
        writeln;
        write('      f = ');
        readln(f[i]);
    end;
writeln;
writeln;
writeln('      Temperatura de Estagnacao [K] (Trecho ',i,')');
writeln;
writeln('      To (x) = ',A2[i]:5:1,'*x^2 + ',B2[i]:5:1,'*x + ',C2[i]:5:1);
writeln;
write('      Deseja alterar ? (s/n)');
ch:=readkey;
if (ch='S') or (ch='s')
    then begin
        writeln;
        writeln;
        writeln('      To (x) = A*x^2 + B*x + C');
        write('      A = ');
        readln(A2[i]);
        write('      B = ');
        readln(B2[i]);
        write('      C = ');
        readln(C2[i]);
    end;
writeln;
writeln;
writeln('      Diametro (Trecho ',i,')');
writeln;
writeln('      D (x) = ',A1[i]:5:3,'*x^2 + ',B1[i]:5:3,'*x + ',C1[i]:5:3);
writeln;
write('      Deseja alterar ? (s/n)');
ch:=readkey;
if (ch='S') or (ch='s')
    then begin
        writeln;
        writeln;
        writeln('      D (x) = A*x^2 + B*x + C');
        write('      A = ');
        readln(A1[i]);

```

```

        write('      B = ');
        readln(B1[i]);
        write('      C = ');
        readln(C1[i]);
    end;
    writeln;
    writeln;
    writeln('      Injecao de Massa (Trecho ',i,')');
    writeln;
    writeln('      dm/dx = ',A3[i]:5:2,'*x + ',B3[i]:5:2);
    writeln;
    write('      Deseja alterar ? (s/n)');
    ch:=readkey;
    if (ch='S') or (ch='s')
    then begin
        writeln;
        writeln;
        writeln('      dm/dx = A*x + B');
        write('      A = ');
        readln(A3[i]);
        write('      B = ');
        readln(B3[i]);
    end;
    end;
    ClrScr;
    writeln;
    writeln;
    writeln('CONTROLE DE BLOCAGEM');
    writeln;
    write('      Reducao de Mach (em %): ');
    readln(porcentagem);
    writeln;
    writeln;
    end
else begin
    ClrScr;
    writeln('      PARAMETROS EXISTENTES :');
    writeln;
    writeln('      Mach inicial = ',M{0}:6:3);
    writeln('      Pressao de estagnacao inicial [Pa] = ',P0{0}:6:3);
    writeln('      k = ',k:6:2);
    writeln('      R [N.m/Kg.K] = ',R:6:2);
    writeln('      Xi [m] = ',Xi:6:2);
    writeln('      Xf [m] = ',Xf:6:2);
    writeln('      Passo [m] = ',h:6:4);
    writeln('      Numero de Trechos : ',trechos);
    window(11,12,80,25);
    writeln('Deseja alterar algum parametro ? <S/N> ');
    repeat
        ch:=upcase(readkey);
    until (ch='S') or (ch='N');
    if ch='S'
    then begin
        writeln;
        writeln('NOVOS PARAMETROS :');
        writeln;
        write('Mach inicial = ');

```

```

    readln(M[0]);
    write('Pressao de estagnacao inicial [Pa] = ');
    readln(P0[0]);
    write('k = ');
    readln(k);
    write('R [N.m/kg.K] = ');
    readln(R);
    write('Passo (passo min. = ',xf/MaxPont:6:4,') [m] = ');
    readln(h);
    end;
for i:=1 to trechos do
begin
  ClrScr;
  writeln('CONDICOES DE CONTORNO :');
  writeln;
  writeln('Fator de atrito (Trecho ',i,')');
  writeln;
  writeln('f = ',f[i]:5:5);
  writeln;
  write('Deseja alterar ? (s/n)');
  ch:=readkey;
  if (ch='S') or (ch='s')
  then begin
    writeln;
    write('f = ');
    readln(f[i]);
  end;
  ClrScr;
  writeln('CONDICOES DE CONTORNO :');
  writeln;
  writeln('Temperatura de Estagnacao [K] (Trecho ',i,')');
  writeln;
  writeln('To (x) = ',A2[i]:5:1,'*x^2 + ',B2[i]:5:1,'*x + ',C2[i]:5:1);
  writeln;
  write('Deseja alterar ? (s/n)');
  ch:=readkey;
  if (ch='S') or (ch='s')
  then begin
    writeln;
    writeln;
    writeln('To (x) = A*x^2 + B*x + C');
    write('A = ');
    readln(A2[i]);
    write('B = ');
    readln(B2[i]);
    write('C = ');
    readln(C2[i]);
  end;
  ClrScr;
  writeln('CONDICOES DE CONTORNO :');
  writeln;
  writeln('Diametro (Trecho ',i,')');
  writeln;
  writeln('D (x) = ',A1[i]:5:3,'*x^2 + ',B1[i]:5:3,'*x + ',C1[i]:5:3);
  writeln;
  write('Deseja alterar ? (s/n)');
  ch:=readkey;

```

```

if (ch='S') or (ch='s')
then begin
    writeln;
    writeln;
    writeln('D (x) = A*x^2 + B*x + C');
    write('A = ');
    readln(A1[i]);
    write('B = ');
    readln(B1[i]);
    write('C = ');
    readln(C1[i]);
end;
ClrScr;
writeln('CONDICOES DE CONTORNO ');
writeln;
writeln('Injecao de Massa (Trecho ',i,')');
writeln;
writeln('dm/dx = ',A3[i]:5:2,'*x + ',B3[i]:5:2);
writeln;
write('Deseja alterar ? (s/n)');
ch:=readkey;
if (ch='S') or (ch='s')
then begin
    writeln;
    writeln;
    writeln('dm/dx = A*x + B');
    write('A = ');
    readln(A3[i]);
    write('B = ');
    readln(B3[i]);
end;
end;
ClrScr;
writeln('CONTROLE DE BLOCAGEM');
writeln;
write('Reducao de Mach (em %): ');
readln(porcentagem);
writeln;
writeln;
end;
window(1,1,80,25);
ClrScr;
gotoxy(14,12);
write('Deseja salvar novos parametros do escoamento ? <S/N> ');
repeat
    ch:=upcase(readkey);
until (ch='S')or(ch='N');
if (ch='S')
then begin
    ClrScr;
    gotoxy(35,5);
    writeln('ARQUIVO');
    writeln;
    write('    Path : ');
    readln(Caminho);
    writeln;
    write('    Nome do arquivo : ');

```

```
        readln(Nome);
        writeln;
        Salva(Caminho,Nome,0,1);
    end;
End;

BEGIN
    Apresentacao;
    T0[0]:=0;
    k:=0;
    repeat
        ClrScr;
        window(30,7,55,18);
        ClrScr;
        writeln('Opcoes :');
        writeln;
        writeln('<C>alculo;');
        writeln('<S>alvar arquivo;');
        writeln('<P>lotar grafico;');
        writeln('<R>ecuperar dados;');
        writeln('<D>ados para planilha;');
        writeln('<T>erminar. ');
        writeln;
        write('Digite opcao : ');
        repeat
            gotoxy(16,10);
            write(' ');
            gotoxy(16,10);
            readln(ch);
            ch:=uppercase(ch);
        until (ch='C')or(ch='S')or(ch='P')or(ch='T')or(ch='R')or(ch='D');
        window(1,1,80,25);
        case ch of
            'C' : begin
                EntradaDeDados;
                Calculo(index);
            end;
            'S' : begin
                ClrScr;
                gotoxy(35,5);
                writeln('ARQUIVO');
                writeln;
                write('  Path : ');
                readln(Caminho);
                writeln;
                write('  Nome do arquivo : ');
                readln(Nome);
                writeln;
                write('  Frequencia de pontos salvos : ');
                readln(freq);
                Salva(Caminho,Nome,index,freq);
            end;
            'D' : begin
                ClrScr;
                gotoxy(35,5);
                writeln('ARQUIVO');
                writeln;
```

```
write(' Path : ');
readln(Caminho);
writeln;
write(' Nome do arquivo : ');
readln(Nome);
writeln;
write(' Frequencia de pontos salvos : ');
readln(freq);
Planilha(Caminho,Nome,index,freq);
end;
'R' : begin
  ClrScr;
  gotoxy(35,5);
  writeln('DADOS');
  writeln;
  write(' Path : ');
  readln(Caminho);
  writeln;
  write(' Nome do arquivo : ');
  readln(Nome);
  Recupera(Caminho,Nome);
end;
'P' : begin
  repeat
    ClrScr;
    gotoxy(35,5);
    writeln('GRAFICO');
    writeln;
    writeln;
    writeln(' [1] Mach x comprimento;');
    writeln(' [2] Pressao estatica x comprimento;');
    writeln(' [3] Pressao de estagnacao x comprimento;');
    writeln(' [4] Temperatura estatica x comprimento;');
    writeln(' [5] Temperatura de estagnacao x comprimento;');
    writeln(' [6] Temperatura x Entropia;');
    writeln;
    writeln(' [S] Sair');
    writeln;
    write(' Digite a opcao : ');
    repeat
      ch:=upcase(readkey);
    until ((ch='1')or(ch='2')or(ch='3')or(ch='4')or(ch='5')or(ch='6')or(ch='S'));
    writeln;
    case ch of
      '1': begin
        Plota('1',index);
      end;
      '2': begin
        Plota('2',index);
      end;
      '3': begin
        Plota('3',index);
      end;
      '4': begin
        Plota('4',index);
      end;
      '5': begin
```

```
        Plota('S'.index);
    end;
    '6': begin
        TS(index,GridAbs,GridOrd);
    end;
end;
until (ch='S');
end;
end;
until (ch='T');
ClrScr;
END.
```

UNIT Calculus;

INTERFACE

USES Dos,Crt;

Procedure Calculo (var i:integer);

Procedure Cursor(status:byte);

CONST

MaxPont = 800;

MaxTrec = 5;

TYPE vetor = array [0..MaxPont] of real;

vector = array [1..MaxTrec] of real;

VAR

```

M,      { * Numero de Mach      * }
P,      { * Pressao estatica     * }
P0,     { * Pressao de estagnacao * }
T,      { * Temperatura estatica * }
T0,     { * Temperatura de estagnacao * }
A,      { * Area                 * }
dmassa, { * Vazao massica        * }
drag,   { * Forca de arrasto     * }
s,      { * Entropia Especifica  * }
G,
X      : vetor;
k,      { * k=Cp/Cv              * }
R,      { * Constante dos gases  * }
fi,     { * fi=1+(k-1)*M^2/2     * }
y,      { * y=Vix/Vi            * }
h,      { * Passo de interacao   * }
xi,     { * Posicao inicial       * }
xf,     { * Posicao final         * }
porcentagem : real;
A1,A2,A3,
B1,B2,B3,
C1,C2,
x1,x2,f : vetor;
i      : integer;
trechos,
w      : byte;

```

IMPLEMENTATION

Procedure Cursor(status:byte);

const startline=7;

endline=7;

var regs : registers;

begin

with regs do

begin

AH:=\$01;

CL:=endline;

CH:=\$20*status+startline;

```

    Intr($10,regs);
end;
end;

Function dA_dx (x:real):real;
Begin
    dA_dx:=0.5*pi*(A1[w]*sqr(x-x1[w])+B1[w]*(x-x1[w])+C1[w])*(2*A1[w]*(x-x1[w])+B1[w]);
End;

Function dT0_dx (x:real):real;
Begin
    dT0_dx:=2*A2[w]*(x-x1[w])+B2[w];
End;

Function CalcArea (x:real):real;
Begin
    CalcArea:=0.25*pi*sqr(A1[w]*sqr(x-x1[w])+B1[w]*(x-x1[w])+C1[w]);
End;

Function CalcT0 (x:real):real;
Begin
    CalcT0:=A2[w]*sqr(x-x1[w])+B2[w]*(x-x1[w])+C2[w];
End;

Function ddrag_dx (x:real):real;
Begin
    ddrag_dx:=0;
End;

Function dmassa_dx (x:real):real;
Begin
    dmassa_dx:=A3[w]*(x-x1[w])+B3[w];
End;

Procedure CondicoesIniciais;
Begin
    for i:=0 to MaxPont do
        begin
            G[i]:=0;
            A[i]:=0;
            P[i]:=0;
            T[i]:=0;
            dmassa[i]:=0;
            s[i]:=0;
        end;
    T0[0]:=CalcT0(xi);
    P[0]:=P0[0]/exp((k/(k-1)*ln(1+(k-1)*sqr(M[0])/2)));
    T[0]:=T0[0]/(1+(k-1)*sqr(M[0])/2);
    A[0]:=CalcArea(xi);
    dmassa[0]:=P[0]*A[0]*sqr(k*R*T[0])*M[0]/(R*T[0]);
    s[0]:=0;
End;

procedure CalculaG (i:integer);
begin
    G[i]:=-dA_dx(x[i])/A[i] +
        0.5*k*sqr(M[i])*((4*f[w]/sqr(4*A[i]/pi))+(2*ddrag_dx(x[i]))/(k*sqr(M[i])*p[i]*A[i]))+

```

```

    0.5*(1+k*sqr(M[i]))*dT0_dx(x[i])/T0[i] +
    (1+k*sqr(M[i])-y*k*sqr(M[i]))*dmassa_dx(x[i])/dmassa[i];
end;

```

```

Function funcao (x,M,A,p,T0,masa:real):real;

```

```

Var

```

```

    M2,f1,f2,f3,f4,f5,f6 : real;

```

```

Begin

```

```

    fi:=1+(k-1)*sqr(M)/2;

```

```

    M2:=sqr(M);

```

```

    f1:=-fi/((1-M2)*A);

```

```

    f2:=k*M2*f1/(2-2*M2);

```

```

    f3:=f1*w/sqrt(4*A/pi);

```

```

    f4:=2/(k*M2*p*A);

```

```

    f5:=(1+k*M2)*fi/(2*T0-2*T0*M2);

```

```

    f6:=fi*((1+k*M2)-y*k*M2)/(masa-massa*M2);

```

```

    funcao:=M*(f1*dA_dx(x)+f2*(f3+f4*ddrag_dx(x))+f5*dT0_dx(x)+f6*dmassa_dx(x));

```

```

End;

```

```

Procedure Calculo (var i:integer);

```

```

Var

```

```

    o1,o2,o3,o4,

```

```

    haux,dM,dMa : real;

```

```

    flag      : boolean;

```

```

    j         : byte;

```

```

Begin

```

```

    Cursor(1);

```

```

    w:=1;

```

```

    CondiçõesIniciais;

```

```

    ClrScr;

```

```

    gotoxy(30,13);

```

```

    write('Interacoes :');

```

```

    x[0]:=xi;

```

```

    flag:=false;

```

```

    i:=0;

```

```

    haux:=h;

```

```

    CalculaG(0);

```

```

    repeat

```

```

        o1:=funcao(x[i],M[i],A[i],P[i],T0[i],dmassa[i]);

```

```

        o2:=funcao(x[i]+h/2,M[i]+o1*h/2,CalcArea(x[i]+h/2),P[i],CalcT0(x[i]+h/2),dmassa[i]);

```

```

        o3:=funcao(x[i]+h/2,M[i]+o2*h/2,CalcArea(x[i]+h/2),P[i],CalcT0(x[i]+h/2),dmassa[i]);

```

```

        o4:=funcao(x[i]+h,M[i]+o3*h,CalcArea(x[i]+h),P[i],CalcT0(x[i]+h),dmassa[i]);

```

```

        dM:=h*(o1+2*o2+2*o3+o4)/6;

```

```

        if i=0 then dMa:=dM;

```

```

        M[i+1]:=M[i]+dM;

```

```

        x[i+1]:=x[i]+h;

```

```

        for j:=1 to trechos do

```

```

            if ((x[i+1]>=x1[j]) and (x[i+1]<=x2[j]))

```

```

                then w:=j;

```

```

        A[i+1]:=CalcArea(x[i+1]);

```

```

        T0[i+1]:=CalcT0(x[i+1]);

```

```

        drag[i+1]:=drag[i]+h*ddrag_dx(x[i+1]);

```

```

        dmassa[i+1]:=dmassa[i]+h*dmassa_dx(x[i+1]);

```

```

        T[i+1]:=T[i]*(T0[i+1]/T0[i])*(1+(k-1)*sqr(M[i])/2)/(1+(k-1)*sqr(M[i+1])/2);

```

```

        P[i+1]:=P[i]*(A[i]/A[i+1])*(M[i]/M[i+1])*sqrt(T[i+1]/T[i])*(dmassa[i+1]/dmassa[i]);

```

```

        P0[i+1]:=P0[i]*(P[i+1]/P[i])*exp((k/(k-1))*ln((1+(k-1)*sqr(M[i+1])/2)/(1+(k-1)*sqr(M[i])/2)));

```

```

        CalculaG(i+1);

```

```
if ((G[i]*G[i+1]>0) and (M[i]>1) and (M[i+1]<1)) or
    ((G[i]*G[i+1]>0) and (M[i]<1) and (M[i+1]>1)) or
    ((G[i]>0) and (G[i+1]>0) and ((dM*dMa)<0))
then begin
    i:=-1;
    M[0]:=M[0]-porcentagem*M[0];
    x[0]:=xi;
    CondiçõesIniciais;
    CalculaG(0);
end
else s[i+1]:=s[i]+((R*k)/(k-1))*ln(T[i+1]/T[i])-R*ln(P[i+1]/P[i]);
i:=i+1;
dMa:=dM;
gotoxy(44,13);
write(i:5);
until (x[i]>=xf) or (i>=MaxPont);
gotoxy(30,15);
write('^g','Calculo efetuado');
Delay(1000);
Cursor(0);
h:=haux;
End;
Begin
End.
```

UNIT Graficus;**INTERFACE**

USES Graph, Calculus, Crt;

CONST

GridAbs=10;

GridOrd=5;

Procedure Initialize;

Procedure Apresentacao;

Procedure Plota (gratype:char; pontos:integer);

Procedure TS (pontos,abscissa,ordenada:integer);

IMPLEMENTATION**CONST**

Fonts : array[0..4] of string[13] =

('DefaultFont', 'TriplexFont', 'SmallFont', 'SansSerifFont', 'GothicFont');

LineStyle : array[0..4] of string[9] =

('SolidLn', 'DottedLn', 'CenterLn', 'DashedLn', 'UserBitLn');

FillStyles : array[0..11] of string[14] =

('EmptyFill', 'SolidFill', 'LineFill', 'LtSlashFill', 'SlashFill',
'BkSlashFill', 'LtBkSlashFill', 'HatchFill', 'XHatchFill',
'InterleaveFill', 'WideDotFill', 'CloseDotFill');

TextDirect : array[0..1] of string[8] = ('HorizDir', 'VertDir');

HorizJust : array[0..2] of string[10] = ('LeftText', 'CenterText', 'RightText');

VertJust : array[0..2] of string[10] = ('BottomText', 'CenterText', 'TopText');

TYPE

politype=array [0..MaxPont] of pointype;

VAR

GraphDriver : integer;

GraphMode : integer;

MaxX, MaxY : word;

ErrorCode : integer;

MaxColor : word;

OldExitProc : Pointer;

Linha : politype;

Procedure Initialize;

Var

InGraphicsMode : boolean;

PathToDriver : string;

Begin

DirectVideo := False;

PathToDriver := "";

repeat

```
{IFDEF Use8514}
  GraphDriver := IBM8514;
  GraphMode := IBM8514Hi;
{$ELSE}
  GraphDriver := Detect;
{$ENDIF}

InitGraph(GraphDriver, GraphMode, PathToDriver);
ErrorCode := GraphResult;
if ErrorCode <> grOK then
begin
  Writeln('Erro grafico: ', GraphErrorMsg(ErrorCode));
  if ErrorCode = grFileNotFound then
  begin
    Writeln('Entre "Path" completo para o driver BGI ou tecla <Ctrl-Break> para finalizar :');
    Readln(PathToDriver);
    Writeln;
  end
  else
  begin
    Halt(1);
  end;
end;
until ErrorCode = grOK;
Randomize;
MaxColor := GetMaxColor;
MaxX := GetMaxX;
MaxY := GetMaxY;
End;

Procedure Apresentacao;
Var ch : char;
Begin
  Initialize;
  ClearDevice;
  SetColor(white);
  Rectangle(10,10,MAXx-10,MAXy-10);
  Rectangle(15,15,MAXx-15,MAXy-15);
  SetTextJustify(CenterText,CenterText);
  SetTextStyle(1,0,3);
  OutTextXY(MAXx div 2,MAXy div 5,'Departamento de Engenharia Mecanica');
  OutTextXY(MAXx div 2,(MAXy div 5)+40,'da Escola Politecnica da USP');
  OutTextXY(MAXx div 2,4*(MAXy div 5),'Sao Paulo');
  OutTextXY(MAXx div 2,4*(MAXy div 5)+40,'-1993-');
  ch:=readkey;
  ClearDevice;
  Rectangle(10,10,MAXx-10,MAXy-10);
  Rectangle(15,15,MAXx-15,MAXy-15);
  SetTextStyle(1,0,5);
  OutTextXY(MAXx div 2,MAXy div 4,'TRABALHO');
  OutTextXY(MAXx div 2,2*(MAXy div 4),'DE');
  OutTextXY(MAXx div 2,3*(MAXy div 4),'FORMATURA');
  ch:=readkey;
  ClearDevice;
  Rectangle(10,10,MAXx-10,MAXy-10);
  Rectangle(15,15,MAXx-15,MAXy-15);
  SetTextStyle(1,0,4);
  OutTextXY(MAXx div 2,MAXy div 4,'Estudo do');
```

```

OutTextXY(MAXx div 2, 2*(MAXy div 4), 'Escoamento Compressível');
OutTextXY(MAXx div 2, 3*(MAXy div 4), 'Unidimensional Generalizado');
ch:=readkey;
ClearDevice;
Rectangle(10, 10, MAXx-10, MAXy-10);
Rectangle(15, 15, MAXx-15, MAXy-15);
SetTextStyle(1, 0, 3);
OutTextXY(MAXx div 2, MAXy div 3, 'Orientador : Prof. Dr. Marcos de M. Pimenta');
OutTextXY(MAXx div 2, 2*(MAXy div 3), 'Alunos : Rogério Rodrigues Jr.');
OutTextXY((MAXx div 2)+18, 2*(MAXy div 3)+30, 'William Arakaki');
ch:=readkey;
CloseGraph;
RestoreCrtMode;
End;

```

```

Procedure Min_Max (vet:vetor; pontos:integer; var min:real; var max:real);
Var i : integer;
Begin
  max:=0;
  min:=1e7;
  for i:=0 to pontos do
    begin
      if vet[i] > max
      then max:=vet[i];
      if vet[i] < min
      then min:=vet[i];
    end;
  max:=1.02*max;
  min:=0.98*min;
End;

```

```

Procedure Eixos;
Var iniciox, inicioy : byte;
    fimx, fimy : integer;
Begin
  Rectangle(0, 0, MAXx, MAXy);          { * Plota borda do grafico * }
  SetLineStyle(0, 1, 2);
  SetColor(white);
  iniciox:=trunc(0.10*MAXx);             { * Determinacao da area do grafico * }
  inicioy:=MAXy div 3;
  fimx:=MAXx-15;
  fimy:=MAXy-45;
  Line(iniciox, inicioy, iniciox, fimy); { * Plota eixo das ordenadas * }
  Line(iniciox-3, inicioy+7, iniciox, inicioy); { * Plota seta do eixo das * }
  Line(iniciox, inicioy, iniciox+3, inicioy+7); { * ordenadas * }
  Line(iniciox, fimy, fimx, fimy);         { * Plota eixo das abscissas * }
  Line(fimx-5, fimy-3, fimx, fimy);        { * Plota seta do eixo das * }
  Line(fimx, fimy, fimx-5, fimy+3);        { * abscissas * }
End;

```

```

Procedure Legenda(Str1, Str2:string);
Begin
  SetTextStyle(0, 0, 1);
  SetTextJustify(CenterText, BottomText);
  OutTextXY(MAXx div 2, MAXy-10, 'x (m)'); { * Legenda das abscissas * }
  SetTextJustify(LeftText, CenterText);
  OutTextXY(4, (MAXy div 3)-5, Str1+Str2); { * Legenda das ordenadas * }

```

```

SetTextJustify(RightText, BottomText);
OutTextXY(MAXx-10, MAXy-5, '<ESC> para sair');
End;

```

Procedure Grid (pontos:integer;abscissa.ordenada:byte;min,max:real);

```

Var coord,i,
    iniciox,
    inicioy,
    fimx,fimy    : integer;
    valor,xval,
    dx,dy        : real;
    S            : string[10];
Begin
    SetLineStyle(3,1,0);
    SetTextJustify(CenterText, TopText);
    iniciox:=trunc(0.10*MAXx);
    inicioy:=(MAXy div 3)+10;
    fimx:=MAXx-23;
    fimy:=MAXy-45;
    dx:=(fimx-iniciox)/abscissa;
    xval:=(x[pontos]-x[0])/abscissa;
    for i:=1 to abscissa do
        begin
            coord:=trunc(i*dx)+iniciox;
            Line(coord,inicioy,coord,fimy);
            Str(xval*i:4:2,S);
            OutTextXY(coord,fimy+6,S);
        end;
    Str(xi:4:2,S);
    OutTextXY(iniciox,fimy+6,S);
    SetTextJustify(LeftText, CenterText);
    dy:=(fimy-inicioy)/ordenada;
    for i:=0 to (ordenada-1) do
        begin
            coord:=trunc(i*dy)+inicioy;
            Line(iniciox,coord,fimx,coord);
            valor:=max-i*(max-min)/ordenada;
            Str(valor:6:3,S);
            OutTextXY(1,coord,S);
        end;
    valor:=min;
    Str(valor:6:3,S);
    OutTextXY(1,fimy-1,S);
End;

```

Procedure Duto(pontos:integer);

```

Var iniciox,
    fimx,fimy,
    i,
    coord    : integer;
    dx       : real;
    s        : string[1];
Begin
    iniciox:=trunc(0.10*MAXx);
    fimx:=MAXx-23;
    fimy:=MAXy-45;
    dx:=(fimx-iniciox)/(X[pontos]-X[0]);

```

```

SetColor(lightgray);
SetWriteMode(CopyPut);
SetLineStyle(3,0,2);
SetTextJustify(CenterText,CenterText);
Line(iniciox,(MAXy div 3)-1,iniciox,20);
Circle(iniciox,12,8);
OutTextXY(iniciox,12,'1');
for i:=2 to trechos do
begin
  coord:=trunc(X1[i]*dx)+iniciox;
  Line(coord,fimy-1,coord,20);
  Circle(coord,12,8);
  Str(i,s);
  OutTextXY(coord,12,s);
end;
dx:=(fimx-iniciox)/GridAbs;
coord:=trunc(GridAbs*dx)+iniciox;
Line(coord,(MAXy div 3)-1,coord,20);
Circle(coord,12,8);
Str(trechos+1,s);
OutTextXY(coord,12,s);
End;

```

```

Procedure Conduto(pontos:integer);
Var i,centro,campo,
    iniciox,fimx,
    yf      : integer;
    max,min,dx,dy,
    D       : real;
Begin
  Min_Max(A,pontos,min,max);
  max:=SQRT(max/PI);
  min:=SQRT(min/PI);
  iniciox:=trunc(0.10*MAXx);
  fimx:=MAXx-23;
  yf:=(MAXy div 3)-5;
  campo:=yf-22;
  centro:=(campo div 2)+20;
  dx:=(fimx-iniciox)/(x[pontos]-x[0]);
  dy:=(campo div 2);
  for i:=0 to pontos do
    begin
      Linha[i].x:=trunc(X[i]*dx)+iniciox;
      D:=SQRT(A[i]/PI);
      Linha[i].y:=centro-trunc(((D-min)/(max-min)+1)*dy/2);
    end;
  SetWriteMode(CopyPut);
  SetColor(lightgray);
  SetLineStyle(0,0,0);
  for i:=0 to (pontos-1) do
    line(Linha[i].x,Linha[i].y,Linha[i+1].x,Linha[i+1].y);
  for i:=0 to pontos do
    begin
      Linha[i].x:=trunc(X[i]*dx)+iniciox;
      D:=SQRT(A[i]/PI);
      Linha[i].y:=centro+trunc(((D-min)/(max-min)+1)*dy/2);
    end;
  end;

```

```

for i:=0 to (pontos-1) do
  line(Linha[i].x,Linha[i].y,Linha[i+1].x,Linha[i+1].y);
Duto(pontos);
SetColor(white);
End;

Procedure Grafico (vet:vetor;pontos:integer;min,max:real);
Var dx,dy : real;
    iniciox,
    inicioy,
    fimx,
    fimy,
    i : integer;
Begin
  SetWriteMode(CopyPut);
  SetColor(cyan);
  SetLineStyle(0,1,3);
  iniciox:=trunc(0.10*MAXx);
  inicioy:=(MAXy div 3)+10;
  fimx:=MAXx-23;
  fimy:=MAXy-45;
  dx:=(fimax-iniciox)/(x[pontos]-x[0]);
  dy:=(fimy-inicioy)/(max-min);
  for i:=0 to pontos do
    begin
      Linha[i].x:=trunc(X[i]*dx)+iniciox;
      Linha[i].y:=fimy-trunc((vet[i]-min)*dy);
    end;
  for i:=0 to (pontos-1) do line(Linha[i].x,Linha[i].y,Linha[i+1].x,Linha[i+1].y);
  SetColor(white);
  SetLineStyle(0,0,0);
End;

Procedure TS (pontos,abscissa,ordenada:integer);
Var maxS,minS,
    maxT,minT,
    dx,dy,
    valor : real;
    iniciox,inicioy,
    fimx,fimy,
    coord,i : integer;
    st : string[10];
Begin
  abscissa:=abscissa div 2;
  Initialize;
  ClearDevice;
  SetWriteMode(CopyPut);
  Rectangle(0,0,MAXx,MAXy);    { * Plota borda do grafico *}
  iniciox:=trunc(0.1*MAXx);
  inicioy:=trunc(0.1*MAXy);
  fimx:=MAXx-23;
  fimy:=MAXy-45;
  Min_Max(T,pontos,minT,maxT);
  Min_Max(S,pontos,minS,maxS);
  minS:=minS/0.8;
  maxS:=maxS/1.02;
  SetTextStyle(0,0,1);

```

```

SetTextJustify(CenterText,BottomText);
OutTextXY(MAXx div 2,inicioy-10,'Diagrama T-dS');
SetTextJustify(CenterText,BottomText);
OutTextXY(MAXx div 2,MAXy-10,'dS (J/kg.K)');  { * Legenda das abscissas *}
SetTextJustify(LeftText,CenterText);
OutTextXY(4,inicioy-5,'T (K)');  { * Legenda das ordenadas *}
SetTextJustify(RightText,BottomText);
OutTextXY(MAXx-10,MAXy-5,'<ESC> para sair');

```

```

SetLineStyle(0,1,2);
SetColor(white);
Line(iniciox,inicioy,iniciox,fimy); { * Plota eixo das ordenadas *}
Line(iniciox-3,inicioy+7,iniciox,inicioy); { * Plota seta do eixo das *}
Line(iniciox,inicioy,iniciox+3,inicioy+7); { * ordenadas *}
Line(iniciox,fimy,MAXx-15,fimy); { * Plota eixo das abscissas *}
Line(MAXx-20,fimy-3,MAXx-15,fimy); { * Plota seta do eixo das *}
Line(MAXx-15,fimy,MAXx-20,fimy+3); { * abscissas *}

```

```

inicioy:=inicioy+10;
dx:=(fimx-iniciox)/abscissa;
dy:=(fimy-inicioy)/ordenada;
SetLineStyle(3,1,0);
SetTextJustify(CenterText,TopText);
for i:=1 to abscissa do
begin
  coord:=trunc(i*dx)+iniciox;
  Line(coord,inicioy,coord,fimy);
  valor:=i*(maxS-minS)/abscissa+minS;
  Str(valor*10:2,st);
  OutTextXY(coord,fimy+6,st);
end;
Str(minS:5:2,st);
OutTextXY(iniciox,fimy+6,st);
SetTextJustify(RightText,CenterText);
for i:=0 to (ordenada-1) do
begin
  coord:=trunc(i*dy)+inicioy;
  Line(iniciox,coord,fimx,coord);
  valor:=maxT-i*(maxT-minT)/ordenada;
  Str(valor:6:2,st);
  OutTextXY(iniciox-5,coord,st);
end;
Str(minT:6:2,st);
OutTextXY(iniciox-5,fimy-1,st);

```

```

SetColor(cyan);
SetLineStyle(0,1,3);
dx:=(fimx-iniciox)/(maxS-minS);
dy:=(fimy-inicioy)/(maxT-minT);
for i:=0 to pontos do
begin
  Linha[i].x:=trunc((S[i]-minS)*dx)+iniciox;
  Linha[i].y:=fimy-trunc((T[i]-minT)*dy);
end;
for i:=0 to (pontos-1) do line(Linha[i].x,Linha[i].y,Linha[i+1].x,Linha[i+1].y);
SetColor(white);
SetLineStyle(0,0,0);

```

```
repeat
  st[0]:=readkey;
until (st[0]#27);

CloseGraph;
RestoreCrtMode;
End;

Procedure Plota (graftype:char;pontos:integer);
Var Min,Max : real;
    vet      : vetor;
    Mensagem,
    unidade  : string[7];
    i        : integer;
    ch       : char;
Begin
  Initialize;
  ClearDevice;
  Eixos;
  case graftype of
    '1' : begin
      vet:=M;
      mensagem:='Mach';
      unidade:='';
      end;
    '2' : begin
      for i:=0 to pontos do vet[i]:=P[i]/1e6;
      mensagem:='P';
      unidade:=' (MPa)';
      end;
    '3' : begin
      for i:=0 to pontos do vet[i]:=P0[i]/1e6;
      mensagem:='Po';
      unidade:=' (MPa)';
      end;
    '4' : begin
      vet:=T;
      mensagem:='T';
      unidade:=' (K)';
      end;
    '5' : begin
      vet:=T0;
      mensagem:='To';
      unidade:=' (K)';
      end;
  end;
  Min_Max(vet,pontos,Min,Max);
  Legenda(mensagem,unidade);
  Grid(pontos,GridAbs,GridOrd,Min,Max);
  Grafico(vet,pontos,Min,Max);
  Min_Max(A,pontos,Min,Max);
  Conduto(pontos);
  repeat
    ch:=readkey;
  until (ch#27);
  CloseGraph;
```

```
RestoreCrtMode;  
End;
```

```
begin  
end.
```

ANEXO 3: DETALHES DO PROGRAMA

Alguns detalhes devem ser observados para a utilização do programa:

1. Após a entrada de dados em na opção *Cálculo*, o programa permite que se salve os parâmetros do e as condições de contorno do escoamento antes de realizar o cálculo propriamente dito. É aconselhável que se aceite esta opção, pois se houver algum problema durante os cálculos, não haverá necessidade de se digitar todos os valores novamente. É importante lembrar que só serão salvos os dados iniciais e que após o cálculo é necessário atualizar o arquivo através da opção *Salvar arquivo*.

2. Sempre que for solicitado o PATH e o NOME é necessário se assegurar de que o o diretório existe. Além disso, não se deve esquecer de colocar a barra invertida ao final do PATH, antes de se digitar o nome do arquivo. Por exemplo, para o arquivo EXEMPLO do diretório DEMO do drive A:

PATH: A:\DemoB\

Nome do arquivo: EXEMPLO

Quando se deseja salvar arquivos, o programa irá pedir a *Frequência de pontos salvos*. Por exemplo, se forem calculados 50 pontos (de 0 a 50) e se digitar frequência=2, serão salvos apenas os pontos 0, 2, 4... Isto foi feito para que não se ocupasse muito espaço de armazenamento em disco.

3. Não há necessidade de se digitar extensões para os arquivos pois o programa já as têm pré-definidas. Quando se utiliza a opção *Dados para planilha*, serão criados arquivos texto contendo os valores calculados para as diversas variáveis (P, Po, T, To...). Estes arquivos são constituídos da seguinte forma: a primeira coluna contém os valores de x (posição do conduto) e a segunda contém os valores de uma determinada propriedade. Pode-se utilizar um editor de texto ou uma planilha para se manipular estes dados. A tabela a seguir mostra como são os arquivos criados pelo programa.

Opção	Extensão	Tipo de arquivo	Conteúdo
<D>ados para planilha	*.M	texto	valores de x e Mach
	*.T		valores de x e Temp. estática
	*.T0		valores de x e Temp. de estag.
	*.P		valores de x e Pressão estática
	*.P0		valores de x e Pressão de estag.
<S>alvar arquivo	*.DAT	texto	parâmetros e cond. de contorno do esc.
	*.CON		valores calculados
	*.VAL	real	valores calculados

BIBLIOGRAFIA

- [1] SHAPIRO, A. H., *The Dynamics and Thermodynamics of Compressible Fluid Flow*, Vol. I, Ronald Press, New York, 1953.
- [2] ZUCROW, M. J. and HOFFMAN, J. D., *Gas Dynamics*, Wiley, New York, 1976.
- [3] ROTT, R. M., *Introduction to Gas Dynamics*, John Wiley & Sons Inc., New York, 1962.
- [4] BINDER, R. C., *Advanced Fluid Dynamics and Fluid Machinery*, Prentice-Hall Inc., New York, 1951.
- [5] OSWATITSCH, Klaus, (English version by Gustav Kuerti), *Gas Dynamics*, Academic Press Inc., New York, 1956.
- [6] VON MISES, Richard, *Mathematical Theory of Compressible Fluid Flow*, Academic Press Inc., New York, 1958.
- [7] PIMENTA, M. M., *Introdução à Dinâmica dos Gases*, notas de aula, Departamento de Engenharia Mecânica da E.U.S.P., São Paulo, 1982.
- [8] CARNAHAN, B.; LUTHER, H. A. and WILKES, J. O., *Applied Numerical Methods*, John Wiley, New York, 1969.