

UNIVERSIDADE DE SÃO PAULO
ESCOLA POLITÉCNICA
DEPARTAMENTO DE ENGENHARIA MECÂNICA

Simulação de Turbina a Gás

Olivia Terence Saa

Orientadores: Marcos de Mattos Pimenta
Euryale de Jesus Zerbini

São Paulo
2009

UNIVERSIDADE DE SÃO PAULO
ESCOLA POLITÉCNICA
DEPARTAMENTO DE ENGENHARIA MECÂNICA

Simulação de Turbina a Gás

Trabalho de formatura apresentado à Escola
Politécnica da Universidade de São Paulo para
obtenção do título de Graduação em Engenharia

Olivia Terence Saa

Orientadores: Marcos de Mattos Pimenta
Euryale de Jesus Zerbini

Área de Concentração:
Engenharia Mecânica

São Paulo

2009

FICHA CATALOGRÁFICA

Saa, Olivia Terence
Simulação de turbina a gás / O.T. Saa. – São Paulo, 2009.
35 p. + apêndice.

Trabalho de Formatura - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia Mecânica.

1. Escoamento (Simulação numérica) I. Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia Mecânica II. t.

AGRADECIMENTOS

Agradeço ao Pimenta, ao Zerbini e ao Beto pela ajuda e motivação.

RESUMO

Este trabalho é uma simulação numérica unidimensional de uma turbina a gás. Foi utilizado o método de Runge-Kutta para a solução numérica de equações diferenciais ordinárias e todos os programas foram feitos em Fortran. O modelo utilizado inclui tanto os aspectos de mecânica e termodinâmica dos fluidos quanto os de cinética química. É um modelo bastante completo, com reações, mudança de área, adição de massa, perda de carga por atrito e transferência de calor.

ABSTRACT

This work is a unidimensional numerical simulation of a gas turbine. Runge-Kutta method was used to solve ordinary differential equations and all programs were made in Fortran. The proposed model includes the mechanical and thermodynamical aspects and also the chemical kinetics. It is quite a complete model, taking into account chemical reactions, area and mass changes, friction pressure loss and heat transfer.

Conteúdo

1	INTRODUÇÃO	1
2	REVISÃO BIBLIOGRÁFICA	2
2.1	Equação da continuidade	2
2.2	Equação do momento	3
2.3	Equação da energia	4
2.4	Equação da continuidade das espécies químicas	6
2.5	Equação de estado dos gases perfeitos	9
3	MATERIAIS E MÉTODOS	14
4	VALIDAÇÃO DO MODELO	15
4.1	Mecânica dos fluidos	15
4.1.1	Aumento e redução de área	15
4.1.2	Aumento e redução de massa	17
4.1.3	Transferência de calor e trabalho	17
4.1.4	Atrito	17
4.2	Cinética química	23
4.3	Comparação com resultados analíticos	25
4.3.1	Escoamento isoentrópico	26
4.3.2	Escoamento com perdas	30
4.3.3	Transição do regime subsônico para o supersônico	32
5	CONCLUSÕES	34

Lista de Figuras

1	Efeito do aumento da área (caso subsônico)	16
2	Efeito do aumento da área (caso supersônico)	16
3	Efeito da redução da área (caso subsônico)	18
4	Efeito da redução da área (caso supersônico)	18
5	Efeito do aumento da área (compressibilidade desprezível) . . .	19
6	Efeito da redução da área (compressibilidade desprezível) . . .	19
7	Modelo de adição de massa pontual	20
8	Efeito da adição de massa (caso subsônico)	20
9	Efeito da adição de massa (caso supersônico)	21
10	Efeito da adição de massa (compressibilidade desprezível) . . .	21
11	Efeito da transferência de calor (caso subsônico)	22
12	Efeito do atrito (caso subsônico)	22
13	Concentração de HF na reação $H_2 + F_2$	23
14	Concentrações das espécies presentes na reação $H_2 + F_2$. . .	24
15	Efeito da reação $H_2 + F_2$ (caso subsônico)	24
16	Escoamento em bocal convergente-divergente	33

NOTAÇÃO UTILIZADA

A	área da seção transversal ao escoamento
C_i	fração mássica da espécie i
cp_f	calor específico a pressão constante “frozen” da mistura
cp_i	calor específico a pressão constante da espécie i
cp_i^0	calor específico a pressão constante “frozen” da espécie i
D	diâmetro hidráulico
f	fator de fricção
g	aceleração da gravidade
h	entalpia da mistura
h_i	entalpia da espécie i
h_i^0	entalpia de formação da espécie i
$k = \frac{cp}{cv}$	do fluido
K_{fj}	coeficiente da reação direta j
K_{bj}	coeficiente da reação inversa j
K_{pj}	coeficiente de equilíbrio da reação j
m	número de reações presentes
M_i	número de Mach na seção i
\dot{m}	vazão mássica total

\dot{m}_i vazão mássica da espécie i

\bar{m}_i massa molecular da espécie i

n número de espécies presentes

N número de moles presentes

N_i número de moles da espécie i

p pressão total

Q calor transferido ao fluido

R constante da mistura

\bar{R} constante universal dos gases

R_i constante R da espécie i

S_i entropia do escoamento na seção i

T_i temperatura do escoamento na seção i

V velocidade do escoamento

W trabalho realizado pelo fluido

X_i fração molar da espécie i

z altura do centróide da seção transversal ao escoamento

ρ massa específica da mistura

ρ_i massa específica da espécie i

σ_i função fonte da espécie i

ν'_{ij} coeficiente estequiométrico da espécie i nos reagentes da reação j

ν''_{ij} coeficiente estequiométrico da espécie i nos produtos da reação j

1 INTRODUÇÃO

O objetivo deste trabalho é a simulação numérica de uma câmara de combustão de uma turbina a gás, seja ela aeronáutica ou de uso termelétrico. Com esse modelo, podem ser feitos diversos ajustes para melhorar o funcionamento do equipamento. Foi utilizado o método de Runge-Kutta para a solução numérica de equações diferenciais ordinárias e todos os programas foram feitos em Fortran.

2 REVISÃO BIBLIOGRÁFICA

O ponto inicial deste trabalho consiste na obtenção de equações diferenciais ordinárias envolvendo as derivadas da velocidade, pressão, temperatura, massa específica e concentrações mássicas das espécies presentes para poder ser aplicado o método numérico de Runge-Kutta ou outro similar. Para isso, partiu-se de equações usuais, como a Primeira Lei da Termodinâmica e a equação da continuidade [1]. Como seu objetivo é a análise do escoamento em regime permanente, este trabalho só considera variações de parâmetros no espaço, nunca no tempo. Ao admitir escoamento unidimensional, só se consideram variações em 1 direção, neste caso, estipulada como x .

2.1 Equação da continuidade

$$\dot{m} = \rho AV$$

onde:

\dot{m} → vazão mássica total

ρ → massa específica da mistura

A → área da seção transversal ao escoamento

V → velocidade do escoamento

Derivando esta equação em x , obtém-se:

$$\frac{d\dot{m}}{dx} = \rho V \frac{dA}{dx} + \rho A \frac{dV}{dx} + V A \frac{d\rho}{dx}$$

Rearranjando:

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

2.2 Equação do momento

$$dp + \rho V dV + \rho g dz + \frac{\rho V^2}{2} \left(\frac{4f dx}{D} \right) + \frac{\partial D}{A} = 0$$

onde:

p → pressão total

g → aceleração da gravidade

z → altura do centróide da seção transversal ao escoamento

f → fator de atrito

D → diâmetro hidráulico

Derivando em x e rearranjando:

$$\frac{dp}{dx} + \rho V \frac{dV}{dx} = -\rho g \frac{dz}{dx} - \frac{\rho V^2}{2} \left(\frac{4f}{D} \right) - \frac{1}{A} \frac{\partial D}{\partial x} \quad (2)$$

2.3 Equação da energia

$$\partial W - \partial Q + dh + d \left(\frac{V^2}{2} \right) + gdz = 0$$

onde:

$W \rightarrow$ trabalho realizado pelo fluido

$Q \rightarrow$ calor transferido ao fluido

$h \rightarrow$ entalpia da mistura

Derivando em x :

$$\frac{\partial W}{\partial x} - \frac{\partial Q}{\partial x} + \frac{dh}{dx} + V \frac{dV}{dx} + g \frac{dz}{dx} = 0 \quad (3)$$

Sabendo que

$$h = \sum_{i=1}^n C_i h_i = \sum_{i=1}^n C_i \left[\int_{T_o}^T c_{pi} dT + h_i^0 \right]$$

tem-se:

$$\frac{dh}{dx} = \sum_{i=1}^n \left(C_i \frac{dh_i}{dx} + h_i \frac{dC_i}{dx} \right) = \sum_{i=1}^n \left(h_i \frac{dC_i}{dx} \right) + \sum_{i=1}^n \left(C_i \frac{d}{dx} \left[\int_{T_o}^T c_{pi} dT + h_i^0 \right] \right)$$

onde:

C_i → fração mássica da espécie i

h_i → entalpia da espécie i

c_{pi} → calor específico a pressão constante da espécie i

h_i^0 → entalpia de formação da espécie i

Como h_i^0 é constante:

$$\frac{dh}{dx} = \sum_{i=1}^n \left(h_i \frac{dC_i}{dx} \right) + \sum_{i=1}^n C_i c_{pf} \frac{dT}{dx} = \sum_{i=1}^n \left(h_i \frac{dC_i}{dx} \right) + c_{pf} \frac{dT}{dx} \quad (4)$$

onde:

$$c_{pf} = \sum_{i=1}^n X_i c_{pi}^0$$

X_i → fração molar da espécie i

c_{pf} → calor específico “frozen” da mistura

c_{pi}^0 → calor específico “frozen” da espécie i

De (3) e (4):

$$\frac{\partial W}{\partial x} - \frac{\partial Q}{\partial x} + c_{pf} \frac{dT}{dx} + \sum_{i=1}^n \left(h_i \frac{dC_i}{dx} \right) + V \frac{dV}{dx} + g \frac{dz}{dx} = 0 \quad (5)$$

2.4 Equação da continuidade das espécies químicas

$$d\dot{m}_i = \sigma_i A dx$$

onde:

\dot{m}_i → vazão mássica da espécie i

σ_i → função fonte da espécie i

Derivando em x e rearranjando:

$$\frac{dC_i}{dx} = \frac{\sigma_i A}{\dot{m}} \quad (6)$$

onde:

$$\sigma_i = \bar{m}_i \sum_{j=1}^m \Delta \nu_{ij} \left[K_{fj} \prod_{i=1}^n \left(\frac{\rho C_i}{\bar{m}_i} \right)^{\nu'_{ij}} - K_{bj} \prod_{i=1}^n \left(\frac{\rho C_i}{\bar{m}_i} \right)^{\nu''_{ij}} \right]$$

$$\Delta \nu_{ij} = (\nu''_{ij} - \nu'_{ij})$$

$$\Delta \nu_j = \sum_{i=1}^n \Delta \nu_{ij}$$

$$K_{fj} = K_{bj} K_{pj} (\bar{R}T)^{-\Delta \nu_j}$$

$$K_{pj} = e^{\frac{-\Delta(E_g)}{RT}}$$

$$C_i = \frac{\dot{m}_i}{\dot{m}} = \frac{\rho_i}{\rho}$$

$$\bar{m}_i = \frac{m_i}{N_i}$$

$$X_i = \frac{N_i}{N}$$

onde:

\bar{m}_i → massa molecular da espécie i

ν'_{ij} → coeficiente estequiométrico da espécie i nos reagentes da reação j

ν''_{ij} → coeficiente estequiométrico da espécie i nos produtos da reação j

\bar{R} → constante universal dos gases

n → número de espécies presentes

K_{fj} → coeficiente da reação direta j

K_{bj} → coeficiente da reação inversa j

K_{pj} → coeficiente de equilíbrio da reação j

ρ_i → massa específica da espécie i

E_g → energia livre de gibbs

N_i → número de moles da espécie i na mistura

N → número de moles total

Multiplicando (6) por h_i e somando de $i = 1$ a n :

$$\sum_{i=1}^n h_i \frac{dC_i}{dx} = \sum_{i=1}^n h_i \frac{\sigma_i A}{\dot{m}} \quad (7)$$

Fazendo $\alpha = (7)$ e juntando a (5):

$$\frac{\partial W}{\partial x} - \frac{\partial Q}{\partial x} + c_{pf} \frac{dT}{dx} + \alpha + V \frac{dV}{dx} + g \frac{dz}{dx} = 0$$

Rearranjando:

$$V \frac{dV}{dx} + c_{pf} \frac{dT}{dx} = \frac{\partial Q}{\partial x} - \frac{\partial W}{\partial x} - g \frac{dz}{dx} - \alpha \quad (8)$$

2.5 Equação de estado dos gases perfeitos

$$p = \rho RT$$

Derivando em x :

$$\frac{dp}{dx} = \rho R \frac{dT}{dx} + \rho T \frac{dR}{dx} + RT \frac{d\rho}{dx}$$

onde:

$R \rightarrow$ constante da mistura

Rearranjando:

$$\frac{dT}{dx} = \frac{T}{p} \frac{dp}{dx} - \frac{T}{\rho} \frac{d\rho}{dx} - \frac{T}{R} \frac{dR}{dx} \quad (9)$$

Sabendo que $R = \sum_{i=1}^n C_i R_i$

$$\frac{dR}{dx} = \sum_{i=1}^n \left(C_i \frac{dR_i}{dx} + R_i \frac{dC_i}{dx} \right) = \sum_{i=1}^n R_i \frac{dC_i}{dx} \quad (10)$$

onde:

$R_i \rightarrow$ constante R da espécie i

Multiplicando (6) por R_i e somando de $i = 1$ a n :

$$\sum_{i=1}^n R_i \frac{dC_i}{dx} = \sum_{i=1}^n R_i \frac{\sigma_i A}{\dot{m}} \quad (11)$$

Fazendo $\beta = (11)$ e juntando as equações (9), (10) e (11):

$$\frac{dT}{dx} = \frac{T}{p} \frac{dp}{dx} - \frac{T}{\rho} \frac{d\rho}{dx} - \frac{T}{R} \beta$$

Rearranjando:

$$\frac{dT}{dx} + \frac{T}{\rho} \frac{d\rho}{dx} - \frac{T}{p} \frac{dp}{dx} = -\frac{T}{R}\beta \quad (12)$$

As equações (1), (2), (6), (8) e (12) formam um sistema de $(4 + n)$ equações e $(4 + n)$ incógnitas $\left(\frac{dV}{dx}, \frac{d\rho}{dx}, \frac{dp}{dx}, \frac{dT}{dx}, \frac{dC_i}{dx}\right)$:

$$\begin{aligned} \rho A \frac{dV}{dx} + VA \frac{d\rho}{dx} &= \frac{d\dot{m}}{dx} - \rho V \frac{dA}{dx} \\ \frac{dp}{dx} + \rho V \frac{dV}{dx} &= -\rho g \frac{dz}{dx} - \frac{\rho V^2}{2} \left(\frac{4f}{D}\right) - \frac{1}{A} \frac{\partial D}{\partial x} \\ \frac{dC_i}{dx} &= \frac{\sigma_i A}{\dot{m}} - \frac{C_i}{\dot{m}} \frac{d\dot{m}}{dx} \\ V \frac{dV}{dx} + c_{pf} \frac{dT}{dx} &= \frac{\partial Q}{\partial x} - \frac{\partial W}{\partial x} - g \frac{dz}{dx} - \alpha \\ \frac{dT}{dx} + \frac{T}{\rho} \frac{d\rho}{dx} - \frac{T}{p} \frac{dp}{dx} &= -\frac{T}{R}\beta \end{aligned}$$

Rearranjando o sistema matricialmente:

$$\begin{bmatrix} \rho A & VA & 0 & 0 & 0 \\ \rho V & 0 & 1 & 0 & 0 \\ V & 0 & 0 & c_{pf} & 0 \\ 0 & \frac{T}{\rho} & -\frac{T}{p} & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{dV}{dx} \\ \frac{d\rho}{dx} \\ \frac{dp}{dx} \\ \frac{dT}{dx} \\ \frac{dC_i}{dx} \end{bmatrix} = \begin{bmatrix} \frac{d\dot{m}}{dx} - \rho V \frac{dA}{dx} \\ -\rho g \frac{dz}{dx} - \frac{\rho V^2}{2} \frac{4f}{D} - \frac{1}{A} \frac{\partial D}{\partial x} \\ \frac{\partial Q}{\partial x} - \frac{\partial W}{\partial x} - g \frac{dz}{dx} - \alpha \\ -\frac{T}{R}\beta \\ \frac{\sigma_i A}{\dot{m}} - \frac{C_i}{\dot{m}} \frac{d\dot{m}}{dx} \end{bmatrix}$$

Tem-se, então, um sistema da forma

$$[A][x] = [B]$$

ou, então:

$$[A]^{-1}[A][x] = [A]^{-1}[B]$$

$$[x] = [A]^{-1}[B]$$

Resolvendo:

$$\frac{dV}{dx} = -\frac{TC_{pf}p \left(\frac{dn}{dx} - \frac{\rho V dA}{dx} \right)}{\rho A (-TC_{pf}p + \rho V^2 TC_{pf} - V^2 p)} + \frac{V TC_{pf} \left(-\rho g \frac{dz}{dx} - \frac{2\rho V^2 f}{D} - \frac{1}{A} \frac{\partial D}{\partial x} \right)}{-TC_{pf}p + \rho V^2 TC_{pf} - V^2 p}$$

$$-\frac{Vp \left(\frac{dQ}{dx} - \frac{dW}{dx} - g \frac{dz}{dx} - \alpha \right)}{-TC_{pf}p + \rho V^2 TC_{pf} - V^2 p} - \frac{V C_{pf} p T \beta}{(-TC_{pf}p + \rho V^2 TC_{pf} - V^2 p) R}$$

$$\frac{dp}{dx} = \frac{V (\rho TC_{pf} - p) \left(\frac{dn}{dx} - \frac{\rho V dA}{dx} \right)}{A (-TC_{pf}p + \rho V^2 TC_{pf} - V^2 p)} - \frac{\rho TC_{pf} \left(-\rho g \frac{dz}{dx} - \frac{2\rho V^2 f}{D} - \frac{1}{A} \frac{\partial D}{\partial x} \right)}{-TC_{pf}p + \rho V^2 TC_{pf} - V^2 p}$$

$$+ \frac{\rho p \left(\frac{dQ}{dx} - \frac{dW}{dx} - g \frac{dz}{dx} - \alpha \right)}{-TC_{pf}p + \rho V^2 TC_{pf} - V^2 p} + \frac{\rho C_{pf} p T \beta}{(-TC_{pf}p + \rho V^2 TC_{pf} - V^2 p) R}$$

$$\frac{dp}{dx} = \frac{V TC_{pf} p \left(\frac{dn}{dx} - \frac{\rho V dA}{dx} \right)}{A (-TC_{pf}p + \rho V^2 TC_{pf} - V^2 p)} - \frac{(TC_{pf} + V^2) p \left(-\rho g \frac{dz}{dx} - \frac{2\rho V^2 f}{D} - \frac{1}{A} \frac{\partial D}{\partial x} \right)}{-TC_{pf}p + \rho V^2 TC_{pf} - V^2 p}$$

$$+ \frac{\rho V^2 p \left(\frac{dQ}{dx} - \frac{dW}{dx} - g \frac{dz}{dx} - \alpha \right)}{-TC_{pf}p + \rho V^2 TC_{pf} - V^2 p} + \frac{\rho V^2 C_{pf} p T \beta}{(-TC_{pf}p + \rho V^2 TC_{pf} - V^2 p) R}$$

$$\frac{dT}{dx} = \frac{VTp \left(\frac{d\dot{m}}{dx} - \frac{\rho V dA}{dx} \right)}{\rho A (-TC_{pf}p + \rho V^2 TC_{pf} - V^2 p)} - \frac{V^2 T \left(-\rho g \frac{dz}{dx} - \frac{2\rho V^2 f}{D} - \frac{1}{A} \frac{\partial D}{\partial x} \right)}{-TC_{pf}p + \rho V^2 TC_{pf} - V^2 p}$$

$$+ \frac{T(-p + \rho V^2) \left(\frac{dQ}{dx} - \frac{dW}{dx} - g \frac{dz}{dx} - \alpha \right)}{-TC_{pf}p + \rho V^2 TC_{pf} - V^2 p} + \frac{V^2 p T \beta}{(-TC_{pf}p + \rho V^2 TC_{pf} - V^2 p) R}$$

$$\frac{dC_i}{dx} = \frac{\sigma_i A}{\dot{m}} - \frac{C_i d\dot{m}}{\dot{m} dx}$$

Finalmente, obteve-se um conjunto de equações diferenciais ordinárias que podem ser resolvidas numericamente por diversos métodos.

3 MATERIAIS E MÉTODOS

Para este trabalho, somente foram utilizados métodos numéricos. A simulação foi feita com um programa em linguagem Fortran, usando o método de Runge-Kutta de 4ª ordem:

$$y_{n+1} = y_n + \frac{h}{6}(m_1 + 2m_2 + 2m_3 + m_4)$$

$$m_1 = f(x_n, y_n)$$

$$m_2 = f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2}m_1\right)$$

$$m_3 = f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2}m_2\right)$$

$$m_4 = f(x_n + h, y_n + hm_3)$$

Para encontrar as propriedades dos gases estudados, usaram-se aproximações do tipo

$$\frac{C_p(T)}{R} = a_1 + a_2T + a_3T^2 + a_4T^3 + a_5T^4$$

$$\frac{H(T)}{RT} = a_1 + a_2\frac{T}{2} + a_3\frac{T^2}{3} + a_4\frac{T^3}{4} + a_5\frac{T^4}{5} + \frac{b_1}{T}$$

$$\frac{S(T)}{R} = a_1\ln(T) + a_2T + a_3\frac{T^2}{2} + a_4\frac{T^3}{3} + a_5\frac{T^4}{4} + b_2$$

cujos coeficientes foram obtidos em [2].

As constantes das reações testadas foram obtidas em [3].

4 VALIDAÇÃO DO MODELO

4.1 Mecânica dos fluidos

Ao desconsiderarmos qualquer tipo de reação química, isolamos o problema da mecânica e termodinâmica dos fluidos no problema. Foram feitas diversas simulações, variando parâmetros críticos, sempre testando tanto o caso subsônico quanto o supersônico. Também foram feitas simulações com velocidades muito baixas.

4.1.1 Aumento e redução de área

O aumento de área causa efeitos diferentes para o escoamento subsônico e o supersônico. No caso subsônico, um aumento da área resulta em uma diminuição da velocidade, um aumento da pressão e da massa específica e um aumento na temperatura (figura (1)). Já no escoamento supersônico, um aumento na área tem um efeito contrário do efeito no escoamento subsônico (figura (2)). As figuras (3) e (4), analogamente, representam a redução constante de área no caso subsônico e no supersônico. A variação de área em escoamentos com velocidades muito baixas tem um efeito um pouco diferente, pois o efeito da compressibilidade se torna desprezível. Esse caso está representado nas figuras (5) e (6).

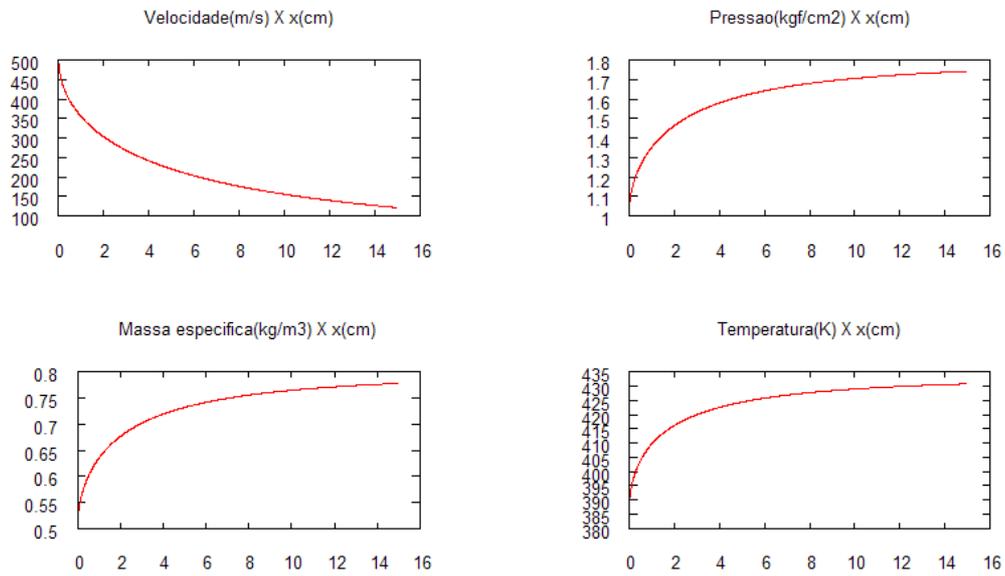


Figura 1: Efeito do aumento da área (caso subsônico)

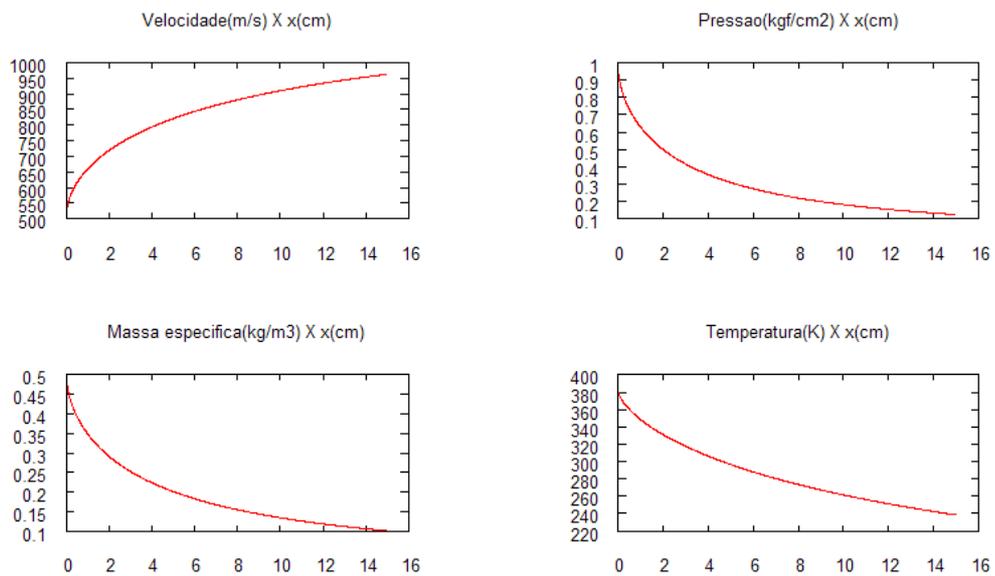


Figura 2: Efeito do aumento da área (caso supersônico)

4.1.2 Aumento e redução de massa

O aumento e redução puros do fluxo mássico no escoamento (vide figura (7)) devem ter efeitos equivalentes à redução e ao aumento de área. As figuras (8), (9) e (10) mostram o efeito da adição de massa pontual no escoamento subsônico, supersônico e muito lento, respectivamente.

4.1.3 Transferência de calor e trabalho

A transferência de calor para o fluido tem um efeito contrário à realização de trabalho pelo fluido. O primeiro está representado na figura (11) (transferência de $1 J/(kg \cdot cm)$ em área de $132 cm^2$).

4.1.4 Atrito

A presença de atrito no caso subsônico resulta em uma diminuição na pressão, temperatura e massa específica e aumento da velocidade, representados na figura (12).

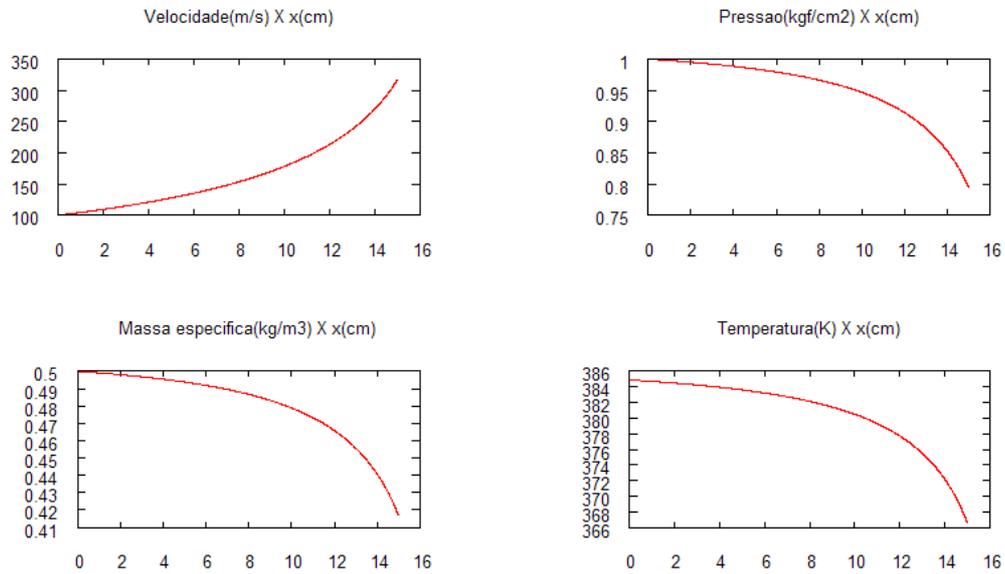


Figura 3: Efeito da redução da área (caso subsônico)

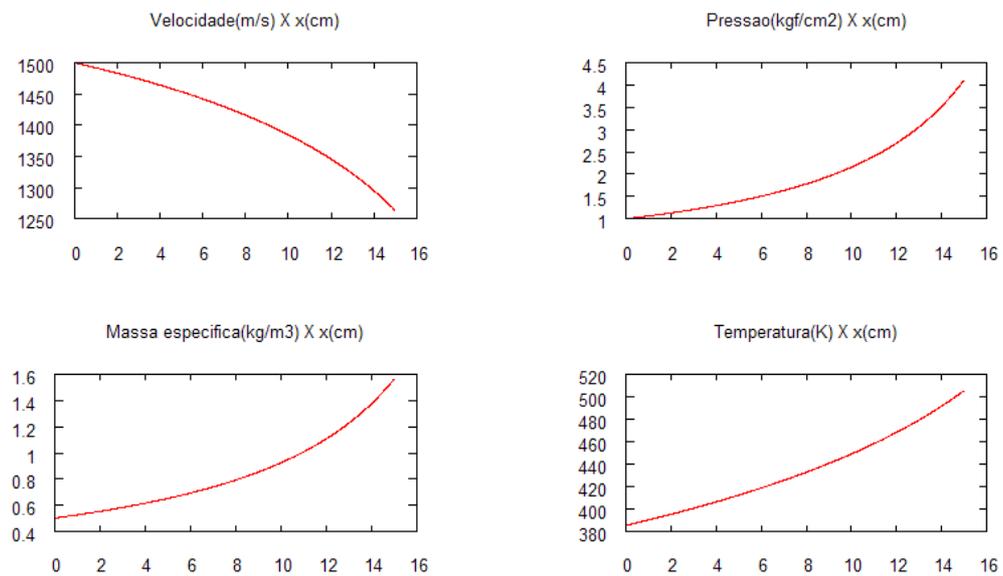


Figura 4: Efeito da redução da área (caso supersônico)

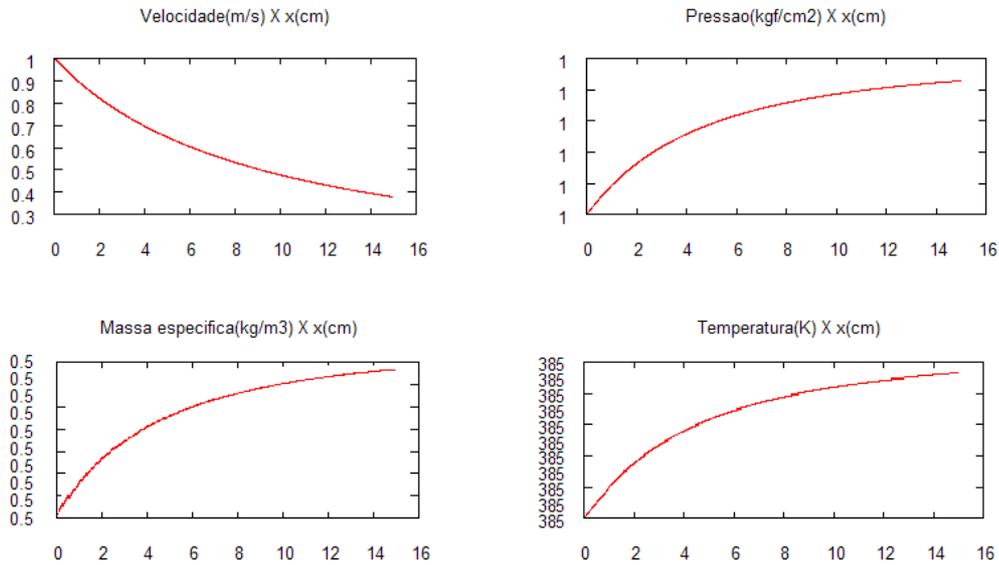


Figura 5: Efeito do aumento da área (compressibilidade desprezível)

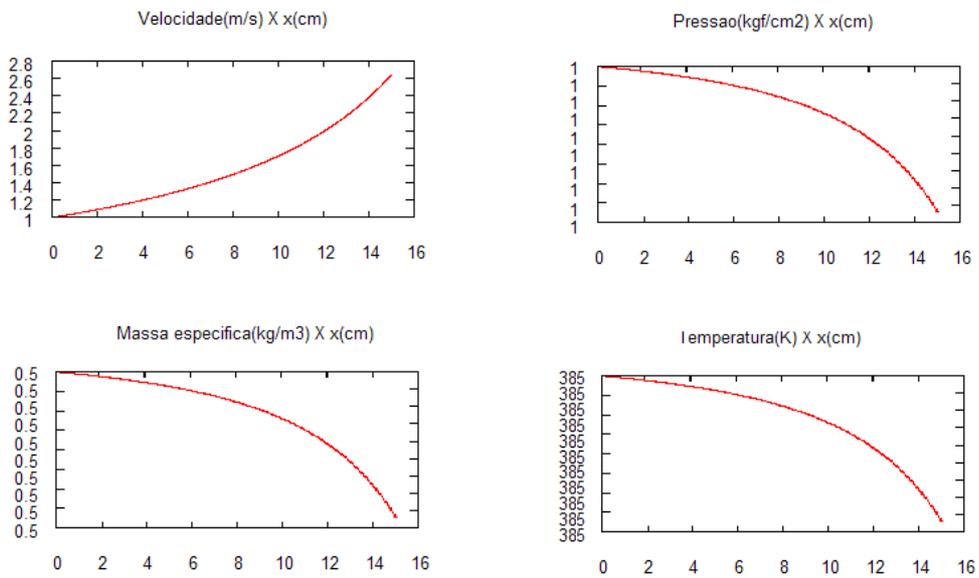


Figura 6: Efeito da redução da área (compressibilidade desprezível)

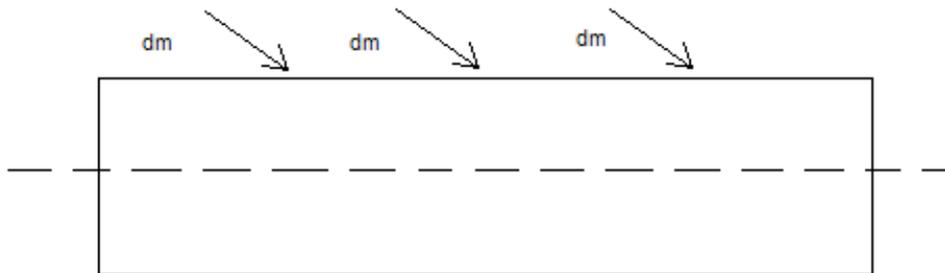


Figura 7: Modelo de adição de massa pontual

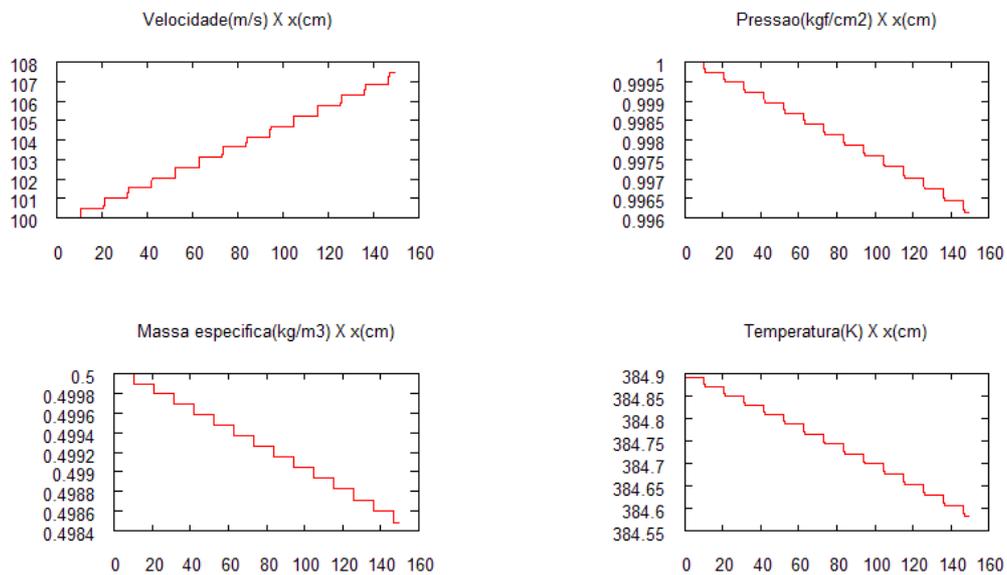


Figura 8: Efeito da adição de massa (caso subsônico)

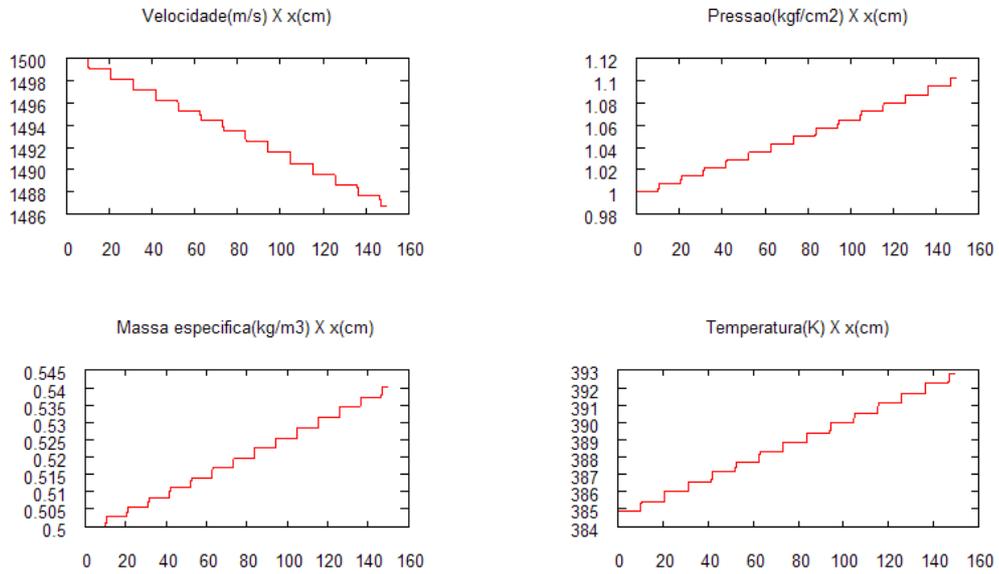


Figura 9: Efeito da adição de massa (caso supersônico)

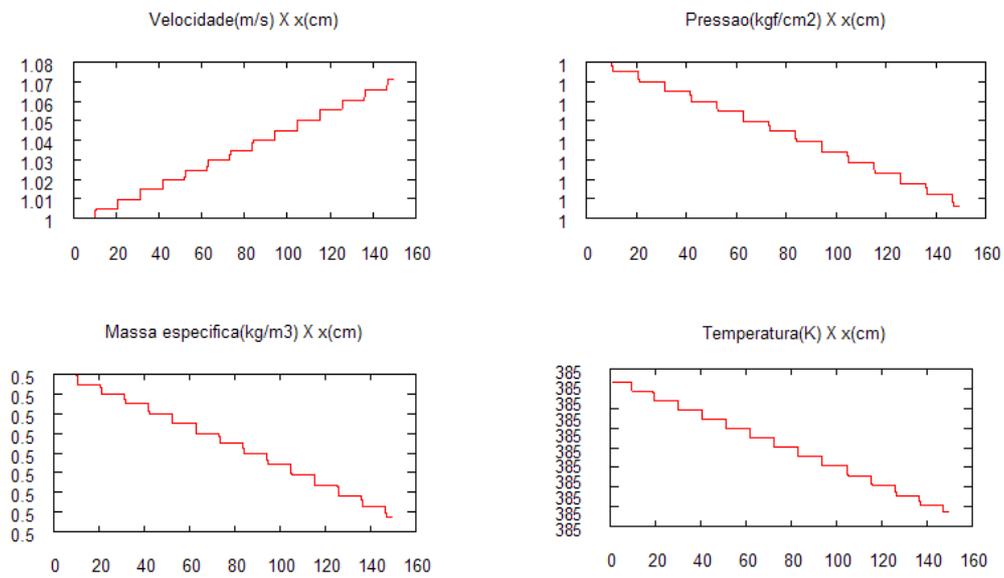


Figura 10: Efeito da adição de massa (compressibilidade desprezível)

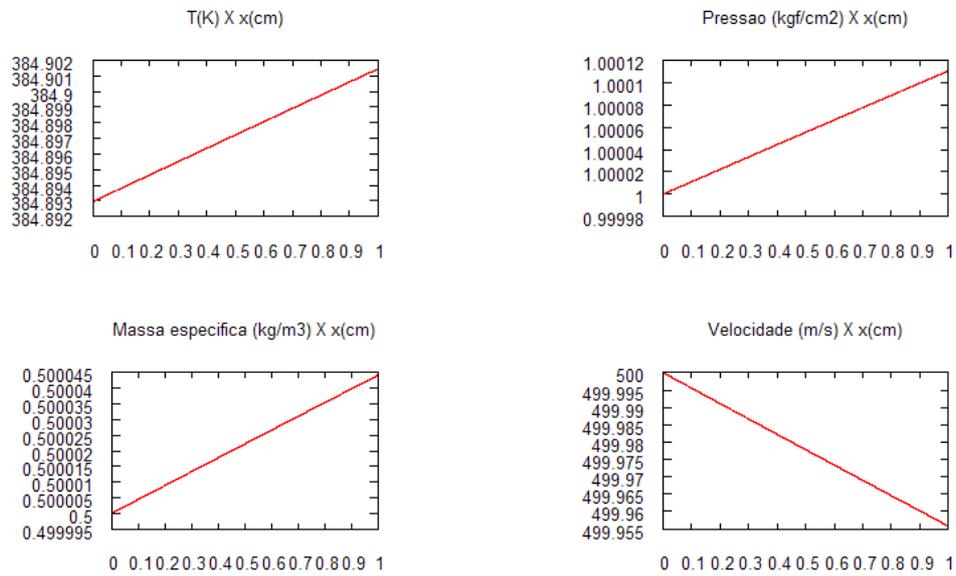


Figura 11: Efeito da transferência de calor (caso subsônico)

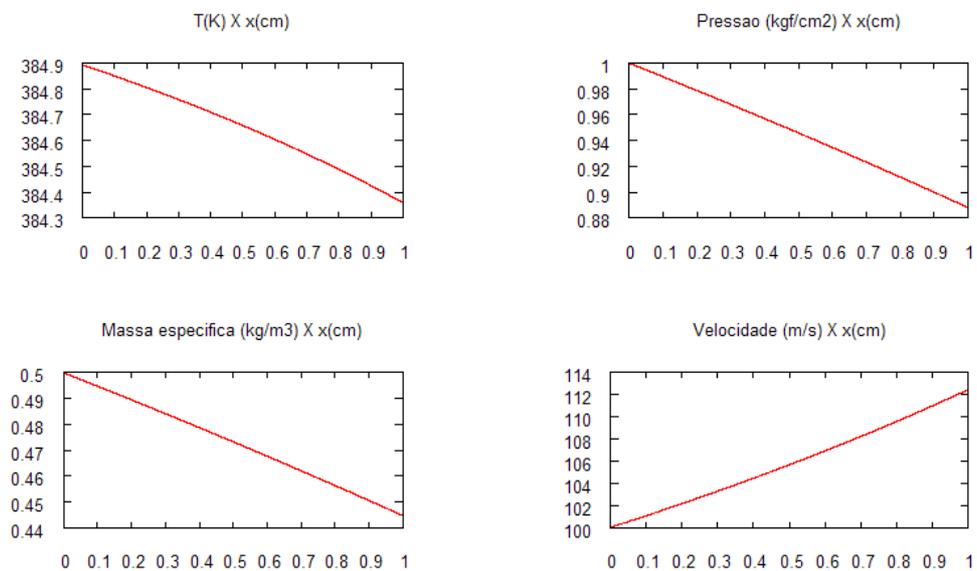
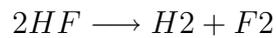


Figura 12: Efeito do atrito (caso subsônico)

4.2 Cinética química

Considerando área e fluxo mássico constantes e escoamento isoentrópico, isolamos o problema da cinética química do modelo.

Usando o modelo de reação de $H_2 + F_2$



com as constantes de reação de [3], temos como resultado as figuras (13), (14) e (15).

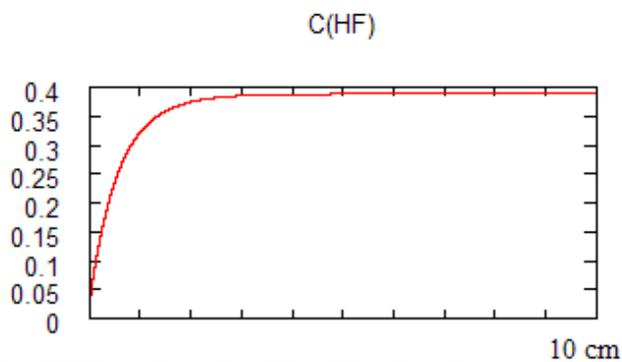
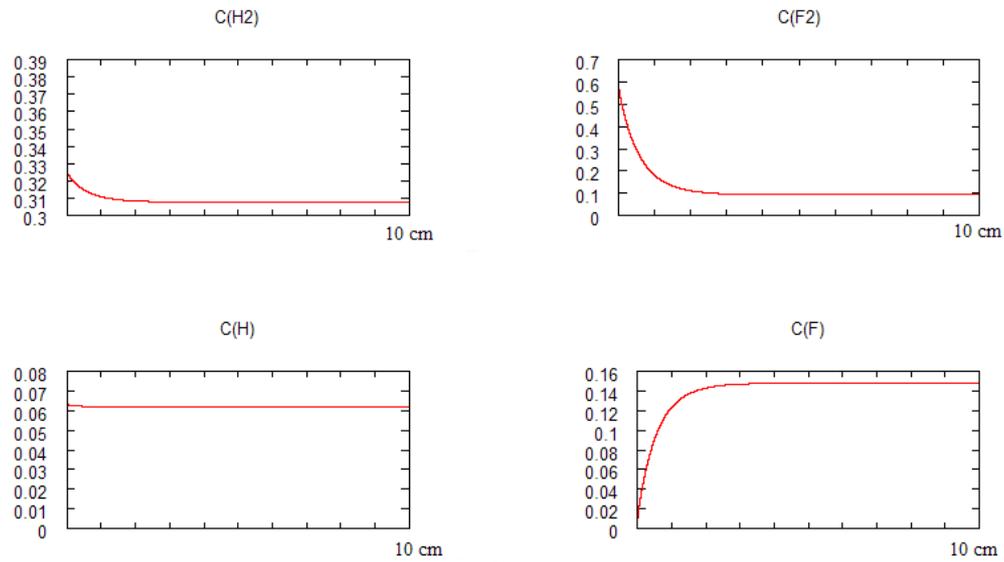
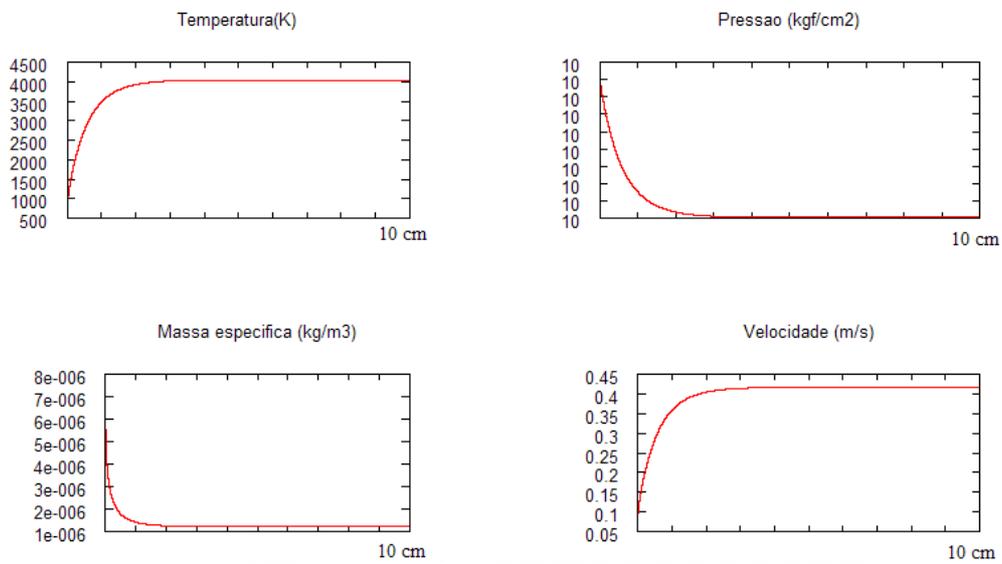


Figura 13: Concentração de HF na reação $H_2 + F_2$

Figura 14: Concentrações das espécies presentes na reação $H_2 + F_2$ Figura 15: Efeito da reação $H_2 + F_2$ (caso subsônico)

4.3 Comparação com resultados analíticos

Sabe-se que, em um escoamento com mudança de área, o número de Mach pode ser calculado com a seguinte equação [4]:

$$\frac{A_2}{A_1} = \frac{M_1}{M_2} \left[\frac{1 + \left(\frac{k-1}{2}\right)M_2^2}{1 + \left(\frac{k-1}{2}\right)M_1^2} \right]^{\frac{k+1}{2(k-1)}} e^{\frac{\Delta S}{R}} \quad (13)$$

Onde:

A_i → área da seção i

M_i → número de Mach na seção i

k → $\frac{C_p}{C_v}$ do fluido

R → constante da mistura

$\Delta S = S_2 - S_1$ → variação da entropia no escoamento

Considerando k constante, as propriedades do escoamento (p , T e ρ) podem ser calculadas com as seguintes expressões:

$$\frac{T_2}{T_1} = \frac{1 + \left(\frac{k-1}{2}\right)M_1^2}{1 + \left(\frac{k-1}{2}\right)M_2^2} \quad (14)$$

$$\frac{p_1}{p_2} = \left[\frac{1 + \left(\frac{k-1}{2}\right)M_2^2}{1 + \left(\frac{k-1}{2}\right)M_1^2} \right]^{\frac{k}{k-1}} e^{\frac{\Delta S}{R}} \quad (15)$$

$$\frac{\rho_2}{\rho_1} = \left[\frac{1 + \left(\frac{k-1}{2}\right)M_1^2}{1 + \left(\frac{k-1}{2}\right)M_2^2} \right]^{\frac{1}{k-1}} e^{\frac{-\Delta S}{R}} \quad (16)$$

4.3.1 Escoamento isoentrópico

No escoamento isoentrópico, $\Delta S = 0$ e as equações (13), (14), (15) e (16) ficam:

$$\frac{A_2}{A_1} = \frac{M_1}{M_2} \left[\frac{1 + \left(\frac{k-1}{2}\right)M_2^2}{1 + \left(\frac{k-1}{2}\right)M_1^2} \right]^{\frac{k+1}{2(k-1)}} \quad (17)$$

$$\frac{T_2}{T_1} = \frac{1 + \left(\frac{k-1}{2}\right)M_1^2}{1 + \left(\frac{k-1}{2}\right)M_2^2} \quad (18)$$

$$\frac{p_1}{p_2} = \left[\frac{1 + \left(\frac{k-1}{2}\right)M_2^2}{1 + \left(\frac{k-1}{2}\right)M_1^2} \right]^{\frac{k}{k-1}} \quad (19)$$

$$\frac{\rho_2}{\rho_1} = \left[\frac{1 + \left(\frac{k-1}{2}\right)M_1^2}{1 + \left(\frac{k-1}{2}\right)M_2^2} \right]^{\frac{1}{k-1}} \quad (20)$$

Assumindo o caso em que:

$$k = 1.393 \text{ (O}_2\text{)}$$

$$A_1 = 2 \text{ cm}^2$$

$$A_2 = 5 \text{ cm}^2$$

$$M_1 = 0.998799$$

$$T_1 = 315.4860 \text{ K}$$

$$\rho_1 = 1.22 \cdot 10^{-6} \frac{\text{kg}}{\text{cm}^3}$$

$$p_1 = 1.00 \text{ atm}$$

Analicamente, utilizando as equações (17), (18), (19) e (20) resulta:

$$M_2 = 0.239717$$

$$T_2 = 373.1171 \text{ K}$$

$$\rho_2 = 1.8697 \cdot 10^{-6} \frac{\text{kg}}{\text{cm}^3}$$

$$p_2 = 1.812 \text{ atm}$$

Usando o programa desenvolvido, temos:

$$M_2 = 0.240173$$

$$T_2 = 372.7354 \text{ K}$$

$$\rho_2 = 1.8722 \cdot 10^{-6} \frac{\text{kg}}{\text{cm}^3}$$

$$p_2 = 1.813 \text{ atm}$$

Que traz um erro comparativo com a solução analítica de:

$$\xi(M) = \frac{0.240173 - 0.239717}{0.240173} = 0.190\%$$

$$\xi(T) = \frac{373.1171 - 372.7354}{373.1171} = 0.134\%$$

$$\xi(\rho) = \frac{1.8722 - 1.8697}{1.8722} = 0.032\%$$

$$\xi(p) = \frac{1.813 - 1.812}{1.813} = 0.102\%$$

Mudando o número de iterações (*itot*) do programa temos:

Itot Erro médio

$$10^6 = 0.063\%$$

$$10^7 = 0.063\%$$

Como o erro mudou muito pouco, pode-se assumir que ele já chegou em sua parte assintótica, o que indica que pode-se aumentar o passo:

Itot Erro médio

$$10^5 = 0.063\%$$

$$10^4 = 0.190\%$$

$$10^3 = 1.545\%$$

$$10^2 = 16.049\%$$

Tendo como base esses dados, para este exemplo, o número de iterações ideal é de aproximadamente 10^5 .

Assumindo agora:

$$M_1 = 1.000279$$

$$Itot = 10^5$$

Resulta analiticamente:

$$M_2 = 2.435686$$

$$T_2 = 177.3109 \text{ K}$$

$$\rho_2 = 2.6962 \cdot 10^{-7} \frac{\text{kg}}{\text{cm}^3}$$

$$p_2 = 1.2210 \cdot 10^{-1} \text{ atm}$$

Com o programa:

$$M_2 = 2.440912$$

$$T_2 = 173.0179 \text{ K}$$

$$\rho_2 = 2.6953 \cdot 10^{-7} \frac{\text{kg}}{\text{cm}^3}$$

$$p_2 = 1.2116 \cdot 10^{-1} \text{ atm}$$

Que traz um erro de

$$\xi(M) = 0.214\%$$

$$\xi(T) = 0.033\%$$

$$\xi(\rho) = 0.782\%$$

$$\xi(p) = 0.747\%$$

$$\mathbf{Erro\ médio} = 0.44\%$$

Com itot = 10^6

$$\mathbf{Erro\ médio} = 0.44\%$$

4.3.2 Escoamento com perdas

Inserindo uma perda distribuída da forma

$$\Delta p = f \frac{\rho V^2}{2}$$

Com $f = 10^{-5}$ e

$$A_1 = 2 \text{ cm}^2$$

$$A_2 = 5 \text{ cm}^2$$

$$M_1 = 0.998799$$

$$T_1 = 315.4860 \text{ K}$$

$$\rho_1 = 1.22 \cdot 10^{-6} \frac{\text{kg}}{\text{cm}^3}$$

$$p_1 = 1.00 \text{ atm}$$

Resulta analiticamente, usando as equações (13), (14), (15) e (16):

$$M_2 = 0.244670$$

$$T_2 = 372.9432 \text{ K}$$

$$\rho_2 = 1.8323 \cdot 10^{-6} \frac{\text{kg}}{\text{cm}^3}$$

$$p_2 = 1.775 \text{ atm}$$

Usando o programa desenvolvido, temos:

$$M_2 = 0.243979$$

$$T_2 = 372.6045 \text{ K}$$

$$\rho_2 = 1.8433 \cdot 10^{-6} \frac{\text{kg}}{\text{cm}^3}$$

$$p_2 = 1.784 \text{ atm}$$

Que traz um erro de:

$$\xi(M) = 0.283\%$$

$$\xi(T) = 0.090\%$$

$$\xi(\rho) = 0.595\%$$

$$\xi(p) = 0.505\%$$

$$\mathbf{Erro\ médio} = 0.369\%$$

4.3.3 Transição do regime subsônico para o supersônico

Para a passagem pela condição sônica, adotou-se $M = 1.001$ sempre que $M = 0.999$ e $\frac{dM}{dx} > 0$. Usando esse artifício e as seguintes condições para o escoamento de O_2 por um bocal convergente-divergente:

$$A_1 = 5.000cm^2 \text{ (área da seção de entrada do bocal)}$$

$$A_2 = 5.000cm^2 \text{ (área da seção de saída do bocal)}$$

$$M_1 = 0.6430 \text{ (número de Mach na seção de entrada do bocal)}$$

$$A_{min} = 4.375cm^2 \text{ (área da menor seção do bocal)}$$

Obteve-se com o programa:

$$M_2 = 1.447014 \text{ (número de Mach na seção de saída do bocal)}$$

Usando a equação (13) temos como resultado:

$$M_2 = 1.446464$$

Que resulta em um erro de:

$$\xi = 0.038\%$$

A figura (16) mostra o caso acima e sua transição do regime subsônico para o supersônico.

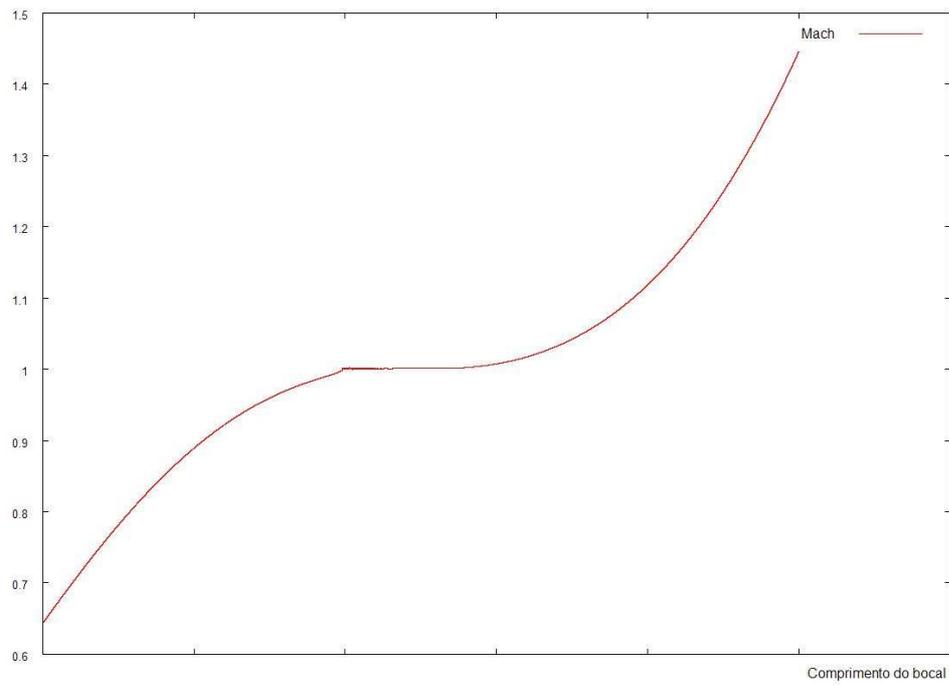


Figura 16: Escoamento em bocal convergente-divergente

5 CONCLUSÕES

Neste trabalho foi feita a modelagem e simulação numérica de um escoamento compressível com reação química. Para isso, foi usado o método de Runge-Kutta de 4ª ordem em um programa de autoria própria em linguagem Fortran.

Na fase de equacionamento, a resolução do sistema linear que descreve o modelo se mostrou muito útil, já que nas simulações se perdeu a necessidade de resolver o sistema linear a cada passo após atualizar a matriz $[A]$.

O método de Runge-Kutta, associado com um passo adequado, foi uma escolha certa para o sistema proposto, apesar de haver existido a necessidade de utilizar-se uma pequena perturbação para evitar a singularidade da transição do regime subsônico para o supersônico.

Tendo em vista as simulações feitas, pode-se concluir que o modelo está correto e é adequado para o problema proposto, com um erro aceitável.

Um possível desenvolvimento desde trabalho é a aplicação do modelo de gas real para uma boa precisão em uma maior faixa de estados termodinâmicos. Outro desdobramento seria o desenvolvimento de um modelo axissimétrico, que requereria um novo equacionamento.

Referências

- [1] Van Wylen, Gordon, *Fundamentos da termodinâmica*, Edgard Blücher, 2003.
- [2] Mc Bride, Bonnie, *Coefficients for Calculating Thermodynamic and Transport Properties of Individual Species*, NASA, 1993.
- [3] Zucrow, Maurice, *Gas Dynamics*, John Wiley and Sons, 1976.
- [4] Pimenta, Marcos, *Introdução à dinâmica dos gases*, 1982.

Apêndice – Código Fonte do programa

```

program          metanosemreacoes
implicit        none
integer         i, itot, b, j, k, l, reacoes, especies, gg

real*8          dm, dA, m, A, dz, g, Cpftotal, f, D, dD,
dQ, dW

real*8          func, prod, Rglobal, Cantes
real*8          V, rho, p, T

real*8          s, propriedadesCH4, propriedadesCH3
real*8          propriedadesCH2O, propriedadesHCO,
propriedadesCO
real*8          propriedadesO, propriedadesO2,
propriedadesH
real*8          propriedadesH2, propriedadesOH,
propriedadesH2O
real*8          propriedadesCO2, Energia
real*8          h, cp, hf, entalpiatotal

real*8          xis, passo, e, somakp
real*8          somah, somaR, somamol, reactionrate
real*8          gibbscarbono, hcarbono, scarbono

real*8          m1, m2, m3, m4, fu, Ee, sigma, Rm,
reagentes
real*8          R, deltanu, X, produtos, Kr, mm, mol, mi
real*8          C, C1, C2, C3, C4, C5, C6, C7, C8, C9, C10, C11
real*8          C12
real*8          lnkpfi, gibbs, hformref, reagentesformacao
real*8          produtosformacao, hform
real*8          mach, cpsobrecv

common /dados1/ sigma
common /dados2/ dm, dA, m, A, dz, g
common /dados3/ Cpftotal, f, D, dD, dQ, dW, somah, somaR, R
    
```

```

! CCCCCCCCCCCCCCCCCCCCCCCCCCCCC
! C Tenho
! C 24 reções
! C 11 especies
! CCCCCCCCCCCCCCCCCCCCCCCCCCCCC
    
```

```

! CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
! C Reações :
! C CH4      +      O      <----->      CH3      +      OH
(j=1)
! C CH4      +      H      <----->      CH3      +      H2
(j=2)
! C CH4      +      OH     <----->      CH3      +      H2O
(j=3)
! C CH4      <----->      CH3      +      H
(j=4)
! C CH3      +      O      <----->      CH2O     +      H
(j=5)
! C CH3      +      O2     <----->      CH2O     +      OH
(j=6)
! C CH2O     +      O      <----->      HCO      +      OH
(j=7)
! C CH2O     +      H      <----->      HCO      +      H2
(j=8)
! C CH2O     +      OH     <----->      HCO      +      H2O
(j=9)
! C CH2O     <----->      HCO      +      H
(j=10)
! C CH2O     <----->      CO       +      H2
(j=11)
! C HCO      +      H      <----->      CO       +      H2
(j=12)
! C HCO      +      OH     <----->      CO       +      H2O
(j=13)
! C HCO      <----->      CO       +      H
    
```

```
(j=14)
! C CO      +      O      <----->      CO2
(j=15)
! C CO      +      O2     <----->      CO2      +      O
(j=16)
! C CO      +      OH     <----->      CO2      +      H
(j=17)
! C O       +      O       <----->      O2
(j=18)
! C O       +      H2     <----->      OH      +      H
(j=19)
! C O       +      OH     <----->      O2      +      H
(j=20)
! C O2      +      H2     <----->      2OH
(j=21)
! C H       +      H       <----->      H2
(j=22)
! C H       +      OH     <----->      H2O
(j=23)
! C H2      +      OH     <----->      H2O      +      H
(j=24)
! C
! CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
```

```
! CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
! C Espécies químicas:
! C CH4      (i=1)
! C CH3      (i=2)
! C CH2O     (i=3)
! C HCO      (i=4)
! C CO       (i=5)
! C O        (i=6)
! C O2       (i=7)
! C H        (i=8)
! C H2       (i=9)
! C OH       (i=10)
! C H2O      (i=11)
! C CO2      (i=12)
! C
! CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
```

```
! CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
! C Mudança nas matrizes:
! C m1, m2, m3, m4 e fu(especies+4)
! C Ee, sigma, Rm, mm, X, C, h, mol, mi, s, lnkphi, Cantes, gibbs e
cp(especies)
! C deltanu(reações)
! C reagentes e produtos(especies,reações)
! C Kr(reações,3)
! C prod(reacoes,2)
! CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
```

```
dimension m1(16), m2(16), m3(16), m4(16), fu(16)
dimension Ee(12), sigma(12), Rm(12), mm(12), X(12), C(12),h(13)
dimension mol(12), mi(12), cp(12), s(13), lnkphi(12), gibbs(12)
dimension reagentes(12,24),produtos(12,24),Kr(24,3),deltanu(24)
dimension prod(24,2), hf(13), reagentesformacao(13,13)
dimension produtosformacao(13,13), hform(13)
```

```
open(1,file='metano.dat',status='unknown')
```

```
! CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
! C
C
! C
C
! C          COMEÇO DO PROGRAMA          C
! C
C
! C
```

```

C
! CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

! CCCCCCCCCCCCCCCCCCCCCC
! C Parametros
! CCCCCCCCCCCCCCCCCCCCCC
   reacoes = 24
   especies = 12

   dm = 0.0           ! 100 kg/(cm*s)
   dA = 0.0           ! cm2/cm
   g = 0.1            ! 100 m/s2
   f = 0.0            ! 1E4 kg/s2
   D = 1.0            ! 100 m*kg/s2
   A = 1.0            ! cm2
   dD = 0.0           ! 1E4 kg/s2
   dQ = 0.0           ! J/(kg*cm)
   dW = 0.0           ! J/(kg*cm)
   Rglobal = 8.3140   ! J/(mol*K)
   dz = 0.0           ! cm/cm
   mm(1) = 16.0E-3    ! kg/mol
   mm(2) = 15.0E-3    ! kg/mol
   mm(3) = 30.0E-3    ! kg/mol
   mm(4) = 29.0E-3    ! kg/mol
   mm(5) = 28.0E-3    ! kg/mol
   mm(6) = 16.0E-3    ! kg/mol
   mm(7) = 32.0E-3    ! kg/mol
   mm(8) = 1.0E-3     ! kg/mol
   mm(9) = 2.0E-3     ! kg/mol
   mm(10) = 17.0E-3   ! kg/mol
   mm(11) = 18.0E-3   ! kg/mol
   mm(12) = 44.0E-3   ! kg/mol

   do l = 1, especies
   Rm(l) = Rglobal/mm(l) ! J/(kg*K)
   enddo

! CCCCCCCCCCCCCCCCCCCCCC
! C Iterações
! CCCCCCCCCCCCCCCCCCCCCC

   passo = 1E-3 ! cm
   itot = 10000

! CCCCCCCCCCCCCCCCCCCCCC
! C Condições iniciais
! CCCCCCCCCCCCCCCCCCCCCC

   C1 = 0.1
   C2 = 0.0
   C3 = 0.0
   C4 = 0.0
   C5 = 0.0
   C6 = 0.0
   C7 = 0.9
   C8 = 0.0
   C9 = 0.0
   C10 = 0.0
   C11 = 0.0
   C12 = 0.0
   C(1) = C1
   C(2) = C2
   C(3) = C3
   C(4) = C4
   C(5) = C5
   C(6) = C6
   C(7) = C7
   C(8) = C8
   C(9) = C9
   C(10) = C10
   C(11) = C11

```

```

C(12) = C12

V = 915                                ! m/s
rho = 1E-7                             ! kg/cm3
p = 0.1                                ! J/cm3 = 10 kgf/cm2
m = rho*V*A                             ! 100 kg/s

R = 0
do i = 1, especies
R = R + C(i)*Rm(i)                    ! J/(kg*K)
enddo

T = p/(R*rho)                          ! K
xis = 0.0                               ! cm

write(1,*) T, C1, C2, C3, C4, C5, C6, C7, C8, C9, C10, C11
#, C12

! Matriz reagentes

do i = 1, especies
do j = 1, reacoes
reagentes(i, j)=0
enddo
enddo

reagentes(1,1) = 1
reagentes(6,1) = 1
reagentes(1,2) = 1
reagentes(8,2) = 1
reagentes(1,3) = 1
reagentes(10,3) = 1
reagentes(1,4) = 1
reagentes(2,5) = 1
reagentes(6,5) = 1
reagentes(2,6) = 1
reagentes(7,6) = 1
reagentes(3,7) = 1
reagentes(6,7) = 1
reagentes(3,8) = 1
reagentes(8,8) = 1
reagentes(3,9) = 1
reagentes(10,9) = 1
reagentes(3,10) = 1
reagentes(3,11) = 1
reagentes(4,12) = 1
reagentes(8,12) = 1
reagentes(4,13) = 1
reagentes(10,13) = 1
reagentes(4,14) = 1
reagentes(5,15) = 1
reagentes(6,15) = 1
reagentes(5,16) = 1
reagentes(7,16) = 1
reagentes(5,17) = 1
reagentes(10,17) = 1
reagentes(6,18) = 2
reagentes(6,19) = 1
reagentes(9,19) = 1
reagentes(6,20) = 1
reagentes(10,20) = 1
reagentes(7,21) = 1
reagentes(9,21) = 1
reagentes(8,22) = 2

```

```

reagentes(8,23) = 1
reagentes(10,23) = 1
reagentes(9,24) = 1
reagentes(10,24) = 1

```

```
! Matriz produtos
```

```

do i = 1,especies
do j = 1,reacoes
produtos(i,j)=0
enddo
enddo

```

```

produtos(2,1) = 1
produtos(10,1) = 1
produtos(2,2) = 1
produtos(9,2) = 1
produtos(2,3) = 1
produtos(11,3) = 1
produtos(2,4) = 1
produtos(8,4) = 1
produtos(3,5) = 1
produtos(8,5) = 1
produtos(3,6) = 1
produtos(10,6) = 1
produtos(4,7) = 1
produtos(10,7) = 1
produtos(4,8) = 1
produtos(9,8) = 1
produtos(4,9) = 1
produtos(11,9) = 1
produtos(4,10) = 1
produtos(8,10) = 1
produtos(5,11) = 1
produtos(9,11) = 1
produtos(5,12) = 1
produtos(9,12) = 1
produtos(5,13) = 1
produtos(11,13) = 1
produtos(5,14) = 1
produtos(8,14) = 1
produtos(12,15) = 1
produtos(12,16) = 1
produtos(6,16) = 1
produtos(12,17) = 1
produtos(8,17) = 1
produtos(7,18) = 1
produtos(10,19) = 1
produtos(8,19) = 1
produtos(7,20) = 1
produtos(8,20) = 1
produtos(10,21) = 2
produtos(9,22) = 1
produtos(11,23) = 1
produtos(11,24) = 1
produtos(8,24) = 1

```

```

do j=1,reacoes
deltanu(j)=0
do l=1,especies
deltanu(j) = deltanu(j) + produtos(l,j) - reagentes(l,j)
end do
end do

```

```
! Matrizes de formação
```

```

do l=1,especies+1
do j=1,especies+1

```

```

reagentesformacao(j,1) = 0
enddo
enddo

do l=1,especies+1
do j=1,especies+1
produtosformacao(j,l) = 0
enddo
enddo

do j=1,especies+1
produtosformacao(j,j) = 1
enddo

!reações de formação
! C          +          2H2          >          CH4
(j = 1)
! C          +          1,5H2        >          CH3
(j = 2)
! C          +          H2           +          0,5O2  >
CH2O      (j = 3)
! C          +          0,5H2      +          0,5O2  >          HCO
(j = 4)
! C          +          0,5O2        >          CO
(j = 5)
! 0,5O2      >          O
(j = 6)
! O2
O2          (j = 7)
! 0,5H2      >          H
(j = 8)
! H2
H2          (j = 9)
! 0,5H2      +          0,5O2        >          OH
(j = 10)
! H2          +          0,5O2        >          H2O
(j = 11)
! C          +          0,5O2        >          CO2
(j = 12)
! C
C          (j = 13)

reagentesformacao(1,13) = 1
reagentesformacao(1,9) = 2
reagentesformacao(2,13) = 1
reagentesformacao(2,9) = 1.5
reagentesformacao(3,13) = 1
reagentesformacao(3,9) = 1
reagentesformacao(3,7) = 0.5
reagentesformacao(4,13) = 1
reagentesformacao(4,7) = 0.5
reagentesformacao(4,9) = 0.5
reagentesformacao(5,13) = 1
reagentesformacao(5,7) = 0.5
reagentesformacao(6,7) = 0.5
reagentesformacao(7,7) = 1
reagentesformacao(8,9) = 0.5
reagentesformacao(9,9) = 1
reagentesformacao(10,7) = 0.5
reagentesformacao(10,9) = 0.5
reagentesformacao(11,9) = 1
reagentesformacao(11,7) = 0.5
reagentesformacao(12,13) = 1
reagentesformacao(12,7) = 0.5
reagentesformacao(13,13) = 1

```

! CCCCCCCCCCCCCCCCCCCCCC

```

! C h
! CCCCCCCCCCCCCCCCCCCCCCCCCC

      l = 2

      h(1) = propiedadesCH4(T,l)           ! J/mol
      h(2) = propiedadesCH3(T,l)           ! J/mol
      h(3) = propiedadesCH2O(T,l)          ! J/mol
      h(4) = propiedadesHCO(T,l)           ! J/mol
      h(5) = propiedadesCO(T,l)            ! J/mol
      h(6) = propiedadesO(T,l)             ! J/mol
      h(7) = propiedadesO2(T,l)            ! J/mol
      h(8) = propiedadesH(T,l)             !
J/mol
      h(9) = propiedadesH2(T,l)            ! J/mol
      h(10) = propiedadesOH(T,l)           ! J/mol
      h(11) = propiedadesH2O(T,l)          ! J/mol
      h(12) = propiedadesCO2(T,l)          ! J/mol
      h(13) = hcarbono(T)                  !
J/mol

      entalpiatotal = 0
      do l=1,especies
      entalpiatotal = entalpiatotal+ h(l)/mm(l)
      enddo

! CCCCCCCCCCCCCCCCCCCCCCCCCC
! C
C
! C
C
! C          CORPO DO PROGRAMA          C
! C
C
! C
C
! CCCCCCCCCCCCCCCCCCCCCCCCCC

      do i=1,itot

      dm = 0.0                               ! 100 kg/(cm*s)

      if (i<5000) then
      dA = -1E-3*(A-0.5)                     ! cm2/cm
      else
      dA = 1E-3*(A-0.5)                     ! cm2/cm
      endif

! CCCCCCCCCCCCCCCCCCCCCCCCCC
! C cp
! CCCCCCCCCCCCCCCCCCCCCCCCCC
      l = 1

      cp(1) = propiedadesCH4(T,l)/mm(1)      ! J/(kg*K)
      cp(2) = propiedadesCH3(T,l)/mm(2)      ! J/(kg*K)
      cp(3) = propiedadesCH2O(T,l)/mm(3)      ! J/(kg*K)
      cp(4) = propiedadesHCO(T,l)/mm(4)      ! J/(kg*K)
      cp(5) = propiedadesCO(T,l)/mm(5)        ! J/(kg*K)
      cp(6) = propiedadesO(T,l)/mm(6)         ! J/(kg*K)
      cp(7) = propiedadesO2(T,l)/mm(7)        ! J/(kg*K)
      cp(8) = propiedadesH(T,l)/mm(8)         ! J/(kg*K)
      cp(9) = propiedadesH2(T,l)/mm(9)        ! J/(kg*K)
      cp(10) = propiedadesOH(T,l)/mm(10)      ! J/(kg*K)
      cp(11) = propiedadesH2O(T,l)/mm(11)     ! J/(kg*K)
      cp(12) = propiedadesCO2(T,l)/mm(12)     ! J/(kg*K)

```

```

! CCCCCCCCCCCCCCCCCCCCCC
! C h
! CCCCCCCCCCCCCCCCCCCCCC

l = 2

h(1) = propiedadesCH4(T,l) ! J/mol
h(2) = propiedadesCH3(T,l) ! J/mol
h(3) = propiedadesCH2O(T,l) ! J/mol
h(4) = propiedadesHCO(T,l) ! J/mol
h(5) = propiedadesCO(T,l) ! J/mol
h(6) = propiedadesO(T,l) ! J/mol
h(7) = propiedadesO2(T,l) ! J/mol
h(8) = propiedadesH(T,l) !
J/mol
h(9) = propiedadesH2(T,l) ! J/mol
h(10) = propiedadesOH(T,l) ! J/mol
h(11) = propiedadesH2O(T,l) ! J/mol
h(12) = propiedadesCO2(T,l) ! J/mol
h(13) = hcarbono(T) !
J/mol

```

```

! CCCCCCCCCCCCCCCCCCCCCC
! C s
! CCCCCCCCCCCCCCCCCCCCCC

l = 3

s(1) = propiedadesCH4(T,l) !
J/(mol*K)
s(2) = propiedadesCH3(T,l) !
J/(mol*K)
s(3) = propiedadesCH2O(T,l) !
J/(mol*K)
s(4) = propiedadesHCO(T,l) !
J/(mol*K)
s(5) = propiedadesCO(T,l) !
J/(mol*K)
s(6) = propiedadesO(T,l) !
J/(mol*K)
s(7) = propiedadesO2(T,l) !
J/(mol*K)
s(8) = propiedadesH(T,l) !
J/(mol*K)
s(9) = propiedadesH2(T,l) !
J/(mol*K)
s(10) = propiedadesOH(T,l) !
J/(mol*K)
s(11) = propiedadesH2O(T,l) !
J/(mol*K)
s(12) = propiedadesCO2(T,l) !
J/(mol*K)
s(13) = scarbono(T) !
J/(mol*K)

```

```

! CCCCCCCCCCCCCCCCCCCCCC
! C massa, mol, X
! CCCCCCCCCCCCCCCCCCCCCC

do l = 1,especies
mi(l) = C(l)*m ! kg/s
enddo

do l = 1,especies
mol(l) = mi(l)/mm(l) ! mol/s

```

```

enddo

somamol = 0
do l = 1, especies
somamol = somamol + mol(l)           ! mol/s
enddo

do l = 1, especies
X(l) = mol(l)/somamol
enddo

! CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
! C   R da mistura e cpf da mistura, MACH
! CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

R = 0
do l = 1, especies
R = R + C(l)*Rm(l)                   ! J/(kg*K)
enddo

Cpftotal = 0
do l = 1, especies
Cpftotal = Cpftotal + X(l)*cp(l)     ! J/(kg*K)
enddo

cpsobrecv = Cpftotal/(Cpftotal-R)
mach = V/((T*cpsobrecv*R)**0.5)

! CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
! C
! C   coeficientes da reação
! C
! C   Kf/Kb = Kp(Rglobal*T)^deltanu
! C
! C
! CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

! CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
! C   Kf = forward
! CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

!
!   do j=1, reacoes
!   Kr(j,1) = reactionrate(T,j)       ! cm, mol, K, s
!   enddo

!
!   do j=1, reacoes
!   Kr(j,1) = 1                       ! cm, mol, K, s
!   enddo

! CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
! C   Kp = equilibrio
! CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

!
!   do l=1, especies+1
!   hf(l) = hformref(l)                ! J/mol
!   enddo
!
!
!
!   do j=1, especies+1
!   hform(j) = hf(j)
!   do l=1, especies+1
!   hform(j) = hform(j) + (h(l) - hf(l)) * (produtosformacao(j,l) -
! #reagentesformacao(j,l))
!   enddo
!   enddo
!
!
!   do j=1, especies+1

```

```

!      gibbs(j) = hform(j)
!      do l=1,especies+1
!      gibbs(j) = gibbs(j) - T*s(l)*(produtosformacao(j,l) -
!      #reagentesformacao(j,l))          ! J/mol
!      enddo
!      enddo

!
!      do l=1,especies
!      lnkpfi(l) = -gibbs(l)/(Rglobal*T)      ! []
!      enddo
!
!      do j = 1, reacoes
!      somakp = 0
!      do l = 1, especies
!      somakp = somakp + (produtos(l,j)-reagentes(l,j))*lnkpfi(l)
!      enddo
!      Kr(j,3) = exp(somakp)
!      enddo

      do j = 1, reacoes
        Kr(j,3) = 1
      enddo

! CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
! C   Kb = backward
! CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

!      do j = 1, reacoes
!      Kr(j,2) = Kr(j,1)/(Kr(j,3) * (Rglobal*T)**(-deltanu(j)))
!      enddo

      do j = 1, reacoes
        Kr(j,2) = 1
      enddo

! CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
! C   Ee para simplificar o sigma
! CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

      do l = 1, especies
        Ee(l) = rho*C(l)/mm(l)          ! mol/cm3
      enddo

! CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
! C   matriz sigma
! CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

!      do j = 1, reacoes
!      prod(j,1) = 1
!      prod(j,2) = 1
!      do l = 1, especies
!      if (reagentes(l,j).ne.0) then
!      prod(j,1) = prod(j,1) * (Ee(l)**reagentes(l,j))
!      endif
!      if (produtos(l,j).ne.0) then
!      prod(j,2) = prod(j,2) * (Ee(l)**produtos(l,j))
!      endif
!      enddo
!      enddo

!      do k =1, especies
!      sigma(k) = 0
!      do j = 1, reacoes
!      sigma(k) = sigma(k) + mm(k)*(produtos(k,j)-reagentes(k,j))*
!      #(Kr(j,1)*prod(j,1)-Kr(j,2)*prod(j,2))      ! kg/(cm3*s)
!      enddo
!      enddo

```

```

do k =1, especies
sigma(k) = 0
do j = 1, reacoes
sigma(k) = 0
enddo
enddo

! CCCCCCCCCCCCCCCCCCCCCC
! C somaR
! CCCCCCCCCCCCCCCCCCCCCC

somaR = 0
do l = 1, especies
somaR = somaR + Rm(l)*sigma(l)*A/m - Rm(l)*C(l)*dm/m
!J/(kg*K*cm)
enddo

! CCCCCCCCCCCCCCCCCCCCCC
! C somah
! CCCCCCCCCCCCCCCCCCCCCC

somah = 0
do l = 1, especies
somah = somah+(h(l)/mm(l))*sigma(l)*A/m-(h(l)/mm(l))*C(l)*dm/m
enddo
!J/(kg*cm)

! CCCCCCCCCCCCCCCCCCCCCC
! C Runge Kutta
! CCCCCCCCCCCCCCCCCCCCCC

do b=1,(4+especies)
m1(b)=func(V,rho,p,T,C1,C2,C3,C4,C5,C6,C7,C8,C9,C10,C11,C12,b)
enddo

do b=1,(4+especies)
m2(b) = func(V+passo*m1(1)/2,rho+passo*m1(2)/2,p+passo*m1(3)
#/2,T+passo*m1(4)/2,C1+passo*m1(5)/2,C2+passo*m1(6)/2,
#C3+passo*m1(7)/2,C4+passo*m1(8)/2,C5+passo*m1(9)/2,
#C6+passo*m1(10)/2,C7+passo*m1(11)/2,C8+passo*m1(12)/2,
#C9+passo*m1(13)/2,C10+passo*m1(14)/2,C11+passo*m1(15)/2,
#C12+passo*m1(16)/2,b)
enddo

do b=1,(4+especies)
m3(b) = func(V+passo*m2(1)/2,rho+passo*m2(2)/2,p+passo*m2(3)
#/2,T+passo*m2(4)/2,C1+passo*m2(5)/2,C2+passo*m2(6)/2,
#C3+passo*m2(7)/2,C4+passo*m2(8)/2,C5+passo*m2(9)/2,
#C6+passo*m2(10)/2,C7+passo*m2(11)/2,C8+passo*m2(12)/2,
#C9+passo*m2(13)/2,C10+passo*m2(14)/2,C11+passo*m2(15)/2,
#C12+passo*m2(16)/2,b)
enddo

do b=1,(4+especies)
m4(b) = func(V+passo*m3(1),rho+passo*m3(2),p+passo*m3(3),
#T+passo*m3(4),C1+passo*m3(5),C2+passo*m3(6),C3+passo*m3(7),
#C4+passo*m3(8),C5+passo*m3(9),C6+passo*m3(10),C7+passo*
#m3(11),C8+passo*m3(12),C9+passo*m3(13),C10+passo*m3(14),
#C11+passo*m3(15),C12+passo*m3(16),b)
enddo

V = V + passo*(m1(1)+2*m2(1)+2*m3(1)+m4(1))/6
rho = rho + passo*(m1(2)+2*m2(2)+2*m3(2)+m4(2))/6
p = p + passo*(m1(3)+2*m2(3)+2*m3(3)+m4(3))/6
T = T + passo*(m1(4)+2*m2(4)+2*m3(4)+m4(4))/6
C1 = C1 + passo*(m1(5)+2*m2(5)+2*m3(5)+m4(5))/6
C2 = C2 + passo*(m1(6)+2*m2(6)+2*m3(6)+m4(6))/6
C3 = C3 + passo*(m1(7)+2*m2(7)+2*m3(7)+m4(7))/6

```

```
      C4 = C4 + passo*(m1(8)+2*m2(8)+2*m3(8)+m4(8))/6
      C5 = C5 + passo*(m1(9)+2*m2(9)+2*m3(9)+m4(9))/6
C6 = C6 + passo*(m1(10)+2*m2(10)+2*m3(10)+m4(10))/6
      C7 = C7 + passo*(m1(11)+2*m2(11)+2*m3(11)+m4(11))/6
      C8 = C8 + passo*(m1(12)+2*m2(12)+2*m3(12)+m4(12))/6
      C9 = C9 + passo*(m1(13)+2*m2(13)+2*m3(13)+m4(13))/6
      C10 = C10 + passo*(m1(14)+2*m2(14)+2*m3(14)+m4(14))/6
      C11 = C11 + passo*(m1(15)+2*m2(15)+2*m3(15)+m4(15))/6
      C12 = C12 + passo*(m1(16)+2*m2(16)+2*m3(16)+m4(16))/6

      C(1) = C1
      C(2) = C2
      C(3) = C3
      C(4) = C4
      C(5) = C5
      C(6) = C6
      C(7) = C7
      C(8) = C8
      C(9) = C9
      C(10) = C10
      C(11) = C11
      C(12) = C12

      xis = xis + passo
      A = A + dA
!      m = m + dm

      write(1,*) T, p, rho, V, mach

!      write(1,*) A, dA

      end do

      close(1)
      end
```

```

! CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC
! C
C
! C
C
! C          FUNÇÕES          C
! C
C
! C
C
! CCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCCC

      function func(V,rho,p,T,C1,C2,C3,C4,C5,C6,C7,C8,C9,C10,C11,C12
#,b)
      implicit none
      integer b, i

      real*8      V, rho, p, T, func
      real*8      sigma, dm, dA, m, A, g
      real*8      Cpftotal, f, D, dD, dQ, dW, somah, somaR, R, dz
      real*8      C1, C2, C3, C4, C5, C6, C7, C8, C9, C10, C11, C12

      common /dados1/ sigma
      common /dados2/ dm, dA, m, A, dz, g
      common /dados3/ Cpftotal, f, D, dD, dQ, dW, somah, somaR, R

      dimension sigma(12)

      if (b.eq.1) then
        func = (-T*Cpftotal*p*(dm-rho*V*dA)/(rho*A)+V*T*Cpftotal*
#(-rho*g*dz-2*rho*V*V*f/D-dD/A)-V*p*(dQ-dW-g*dz-somah)-(V*
#Cpftotal*p*T*somaR)/R)/(-T*Cpftotal*p+rho*V*V*T*Cpftotal-V*V*p)

!      func = -T / rho * Cpftotal / A / (-T * Cpftotal * p + rho * V
!      *** 2 * T * Cpftotal- V ** 2 * p) * p * (dm - rho * V * dA) +
!      #V * T * Cpftotal / (-T * Cpftotal * p + rho * V ** 2 * T *
!      #Cpftotal - V ** 2 * p) * (-rho * g * dz - 2 * rho * V ** 2 *
!      #f / D - 1 / A * dD) - V / (-T * Cpftotal * p + rho * V ** 2 *
!      #T * Cpftotal - V ** 2 * p) * p * (dQ - dW - g * dz - somah) -
!      #V * Cpftotal / (-T * Cpftotal * p + rho * V ** 2 * T * Cpftotal
!      # - V ** 2 * p) * p * T / R * somaR
!
!      elseif (b.eq.2) then

          func = (V*(rho*T*Cpftotal-p)*(dm-rho*V*dA)/A-rho*T*Cpftotal*
#(-rho*g*dz-2*rho*V*V*f/D-dD/A)+rho*p*(dQ-dW-g*dz-somah)+(rho*
#Cpftotal*p*T*somaR)/R)/(-T*Cpftotal*p+rho*V*V*T*Cpftotal-V*V*p)

!
!      func = V * (rho * T * Cpftotal - p) / A / (-T * Cpftotal
* p +
!      #rho * V **2 * T * Cpftotal - V ** 2 * p) * (dm - rho * V * dA)
!      #- rho * T * Cpftotal / (-T * Cpftotal * p + rho * V ** 2 * T *
!      #Cpftotal - V ** 2 * p) * (-rho * g * dz - 2 * rho * V ** 2 * f
!      # / D - 1 / A * dD) + rho / (-T * Cpftotal * p + rho * V ** 2 *
!      #T * Cpftotal - V ** 2 * p) * p * (dQ - dW - g * dz - somah) +
!      #rho * Cpftotal / (-T * Cpftotal * p + rho * V ** 2 * T *
!      #Cpftotal - V ** 2 * p) * p * T / R * somaR

      elseif (b.eq.3) then

          func = (V*T*Cpftotal*p*(dm-rho*V*dA)/A-(T*Cpftotal+V*V)*p*(
#-rho*g*dz-2*rho*V*V*f/D-dD/A)+rho*p*V*V*(dQ-dW-g*dz-somah)+
#(rho*V*V*Cpftotal*p*T*somaR)/R)/(-T*Cpftotal*p+rho*V*V*T*
#Cpftotal-V*V*p)

!
!      func = V * T * Cpftotal / A / ( (-T) * Cpftotal * p + rho * V
!      *** 2 * T * Cpftotal - V ** 2 * p) * p * (dm - rho * V * dA) -

```

```

!      #(T * Cpfttotal + V ** 2) * p / (-T * Cpfttotal * p + rho * V **
!      #2 * T * Cpfttotal - V ** 2 * p) * (-rho * g * dz - 2 * rho * V
!      #** 2 * f / D - 1 / A * dD) + rho * V ** 2 / (-T * Cpfttotal * p
!      # + rho * V ** 2 * T * Cpfttotal - V ** 2 * p) * p * (dQ - dW -
!      #g * dz - somah) + rho * V ** 2 * Cpfttotal / (-T * Cpfttotal * p
!      # + rho * V ** 2 * T * Cpfttotal - V ** 2 * p) * p * T / R *
!      #somaR

      elseif (b.eq.4) then

          func = (V*T*p*(dm-rho*V*dA)/(rho*A)-T*V*V*(-rho*g*dz-2*rho*V*V*
#f/D-dD/A)+T*(rho*V*V-p)*(dQ-dW-g*dz-somah)+(V*V*p*T*somaR)/R)/
#(-T*Cpfttotal*p+rho*V*V*T*Cpfttotal-V*V*p)

!      func = V * T / rho / A / (-T * Cpfttotal * p + rho * V ** 2 * T
!      # *Cpfttotal - V ** 2 * p) * p * (dm - rho * V * dA) - V ** 2 *
!      #T / ( -T * Cpfttotal * p + rho * V ** 2 * T * Cpfttotal - V ** 2
!      # * p) * (-rho * g * dz - 2 * rho * V ** 2 * f / D - 1 / A * dD)
!      # + T * (-p + rho * V ** 2) / (-T * Cpfttotal * p + rho * V ** 2
!      # * T * Cpfttotal - V ** 2 * p) * (dQ - dW - g * dz - somah) + V
!      # ** 2 / (-T * Cpfttotal* p + rho * V ** 2 * T * Cpfttotal - V **
!      # 2 * p) * p * T / R *somaR
!

!      elseif (b.eq.5) then
!      func = sigma(1) * A / (rho*A*V)

!      elseif (b.eq.6) then
!      func = sigma(2) * A / (rho*A*V)

!      elseif (b.eq.7) then
!      func = sigma(3) * A / (rho*A*V)

!      elseif (b.eq.8) then
!      func = sigma(4) * A / (rho*A*V)

!      elseif (b.eq.9) then
!      func = sigma(5) * A / (rho*A*V)

!      elseif (b.eq.10) then
!      func = sigma(6) * A / (rho*A*V)

!      elseif (b.eq.11) then
!      func = sigma(7) * A / (rho*A*V)

!      elseif (b.eq.12) then
!      func = sigma(8) * A / (rho*A*V)
!
!      elseif (b.eq.13) then
!      func = sigma(9) * A / (rho*A*V)

!      elseif (b.eq.14) then
!      func = sigma(10) * A / (rho*A*V)

!      elseif (b.eq.15) then
!      func = sigma(11) * A / (rho*A*V)

!      else
!      func = sigma(12) * A / (rho*A*V)
!      endif

      else
      func = 0
      endif

      end

```



```

Ea(1) = 33.01E3
Ea(2) = 36.49E3
Ea(3) = 13.0E3
Ea(4) = 451.0E3
Ea(5) = 0.4E3
Ea(6) = 13.4E3
Ea(7) = 11.55E3
Ea(8) = 14.85E3
Ea(9) = -35.4E3
Ea(10) = 418.0E3
Ea(11) = 206.0E3
Ea(12) = 1.08E3
Ea(13) = -1.12E3
Ea(14) = 70.81E3
Ea(15) = 6.69E3
Ea(16) = 206.0E3
Ea(17) = 0.98E3
Ea(18) = -9.97E3
Ea(19) = 29.85E3
Ea(20) = -9.7E2
Ea(21) = 287.0E3
Ea(22) = 0.52E3
Ea(23) = -1.84E3
Ea(24) = 18.29E3

```

```

reactionrate = A(j)*(T/Tref)**n(j)*exp(-Ea(j)/(R*T))

```

```

end

```

```

function propiedadesCH4(T,l)

```

```

!      l = 1   cp
!      l = 2   h
!      l = 3   s

```

```

integer    l
real*8     T, a1, a2, a3, a4, a5, b1, b2, R

```

```

R = 8.314

```

```

!      if (T>1000) then
a1 = 7.48514950E-02
a2 = 1.33909467E-02
a2 = 1.33909467E-05
a3 = -5.73285809E-06
a4 = 1.22292535E-09
a5 = -1.01815230E-13
b1 = -9.46834459E+03
b2 = 1.84373180E+01
      else
a1 = 5.14987613
a2 = -1.36709788E-2
a3 = 4.91800599E-5
a4 = -4.84743026E-8
a5 = 1.66693956E-11
b1 = -1.02466476E4
b2 = -4.64130376
      endif

```

```

      if (l.eq.1) then
propiedadesCH4 = (a1 + a2*T + a3*T**2 + a4*T**3 + a5*T**4)*R
      elseif (l.eq.2) then
propiedadesCH4 = (a1 + a2*T/2 + a3*(T**2)/3 + a4*(T**3)/4 +
#a5*(T**4)/5 + b1/T)*R*T
      else
propiedadesCH4 = (a1*log(T) + a2*T + a3*(T**2)/2 + a4*(T**3)/

```

```

#3 + a5*(T**4)/4 + b2)*R
endif

end

function propiedadesCH3(T,l)

!      l = 1    cp
!      l = 2    h
!      l = 3    s

integer  l
real*8   T, a1, a2, a3, a4, a5, b1, b2, R

R = 8.314

if (T>1000) then
a1 = 2.96866033
a2 = 5.80717646E-3
a3 = -1.97778534E-6
a4 = 3.07278752E-10
a5 = -1.78853897E-14
b1 = 1.65388869E4
b2 = 4.77944503
else
a1 = 3.67359040
a2 = 2.01096176E-3
a3 = 5.73021856E-6
a4 = -6.87117425E-9
a5 = 2.54385734E-12
b1 = 1.64449988E4
b2 = 1.60456433
endif

if (l.eq.1) then
propiedadesCH3 = (a1 + a2*T + a3*T**2 + a4*T**3 + a5*T**4)*R
elseif (l.eq.2) then
propiedadesCH3 = (a1 + a2*T/2 + a3*(T**2)/3 + a4*(T**3)/4 +
#a5*(T**4)/5 + b1/T)*R*T
else
propiedadesCH3 = (a1*log(T) + a2*T + a3*(T**2)/2 + a4*(T**3)/
#3 + a5*(T**4)/4 + b2)*R
endif

end

function propiedadesCH2O(T,l)

!      l = 1    cp
!      l = 2    h
!      l = 3    s

integer  l
real*8   T, a1, a2, a3, a4, a5, b1, b2, R

R = 8.314

if (T>1000) then
a1 = 3.16952664
a2 = 6.19320583E-3
a3 = -2.25056377E-6
a4 = 3.66975680E-10

```

```

a5 = -2.20149470E-14
b1 = -1.44784444E4
b2 = 6.04209449
else
a1 = 4.79372315
a2 = -9.90833369E-3
a3 = 3.73220008E-5
a4 = -3.79285261E-8
a5 = 1.31772652E-11
b1 = -1.43089667E4
b2 = 6.02812900E-1
endif

if (l.eq.1) then
propiedadesCH2O = (a1 + a2*T + a3*T**2 + a4*T**3 + a5*T**4)*R
elseif (l.eq.2) then
propiedadesCH2O = (a1 + a2*T/2 + a3*(T**2)/3 + a4*(T**3)/4 +
#a5*(T**4)/5 + b1/T)*R*T
else
propiedadesCH2O = (a1*log(T) + a2*T + a3*(T**2)/2 + a4*(T**3)
#/3 + a5*(T**4)/4 + b2)*R
endif

end

function propiedadesHCO(T,l)

!      l = 1    cp
!      l = 2    h
!      l = 3    s

integer  l
real*8   T, a1, a2, a3, a4, a5, b1, b2, R

R = 8.314

if (T>1000) then
a1 = 3.64896209
a2 = 3.08090819E-3
a3 = -1.12429876E-6
a4 = 1.86308086E-10
a5 = -1.13951828E-14
b1 = 3.71209048E3
b2 = 5.06147406
else
a1 = 4.22118584
a2 = -3.24392532E-3
a3 = 1.37799446E-5
a4 = -1.33144093E-8
a5 = 4.33768865E-12
b1 = 3.83956494E3
b2 = 3.39437243
endif

if (l.eq.1) then
propiedadesHCO = (a1 + a2*T + a3*T**2 + a4*T**3 + a5*T**4)*R
elseif (l.eq.2) then
propiedadesHCO = (a1 + a2*T/2 + a3*(T**2)/3 + a4*(T**3)/4 +
#a5*(T**4)/5 + b1/T)*R*T
else
propiedadesHCO = (a1*log(T) + a2*T + a3*(T**2)/2 + a4*(T**3)
#/3 + a5*(T**4)/4 + b2)*R
endif

end

function propiedadesCO(T,l)

```

```

!      l = 1   cp
!      l = 2   h
!      l = 3   s

integer    l
real*8     T, a1, a2, a3, a4, a5, b1, b2, R

R = 8.314

if (T>1000) then
a1 = 3.04848583
a2 = 1.35172818E-3
a3 = -4.85794075E-7
a4 = 7.88536486E-11
a5 = -4.69807489E-15
b1 = -1.42661171E4
b2 = 6.01709790
else
a1 = 3.57953347
a2 = -6.10353680E-4
a3 = 1.01681433E-6
a4 = 9.07005884E-10
a5 = -9.04424499E-13
b1 = -1.4344086E4
b2 = 3.50840928
endif

if (l.eq.1) then
propriedadesCO = (a1 + a2*T + a3*T**2 + a4*T**3 + a5*T**4)*R
elseif (l.eq.2) then
propriedadesCO = (a1 + a2*T/2 + a3*(T**2)/3 + a4*(T**3)/4 +
#a5*(T**4)/5 + b1/T)*R*T
else
propriedadesCO = (a1*log(T) + a2*T + a3*(T**2)/2 + a4*(T**3)/
#3 + a5*(T**4)/4 + b2)*R
endif

end

```

```
function  propriedadesO(T,l)
```

```

!      l = 1   cp
!      l = 2   h
!      l = 3   s

integer    l
real*8     T, a1, a2, a3, a4, a5, b1, b2, R

R = 8.314

if (T>1000) then
a1 = 2.54363697
a2 = -2.73162486E-5
a3 = -4.1902952E-9
a4 = 4.95481845E-12
a5 = -4.79553694E-16
b1 = 2.9226012E4
b2 = 4.92229457
else
a1 = 3.1682671
a2 = -3.27931884E-3
a3 = 6.64306396E-6
a4 = -6.12806624E-9
a5 = 2.11265971E-12
b1 = 2.91222592E4
b2 = 2.05193346

```

```

endif

    if (l.eq.1) then
        propiedades0 = (a1 + a2*T + a3*T**2 + a4*T**3 + a5*T**4)*R
    elseif (l.eq.2) then
        propiedades0 = (a1 + a2*T/2 + a3*(T**2)/3 + a4*(T**3)/4 +
#a5*(T**4)/5 + b1/T)*R*T
    else
        propiedades0=(a1*log(T) + a2*T + a3*(T**2)/2 + a4*(T**3)/3 +
#a5*(T**4)/4 + b2)*R
    endif

end

function propiedadesO2(T,l)

!       l = 1   cp
!       l = 2   h
!       l = 3   s

integer    l
real*8     T, a1, a2, a3, a4, a5, b1, b2, R

R = 8.314

if (T>1000) then
a1 = 3.66096083
a2 = 6.56365523E-4
a3 = -1.41149485E-7
a4 = 2.05797658E-11
a5 = -1.29913248E-15
b1 = -1.21597725E3
b2 = 3.41538184
else
a1 = 3.78245636
a2 = -2.99673415E-3
a3 = 9.847302E-6
a4 = -9.68129508E-9
a5 = 3.24372836E-12
b1 = -1.06394356E3
b2 = 3.65767573
endif

if (l.eq.1) then
propiedadesO2 = (a1 + a2*T + a3*T**2 + a4*T**3 + a5*T**4)*R
elseif (l.eq.2) then
propiedadesO2 = (a1 + a2*T/2 + a3*(T**2)/3 + a4*(T**3)/4 +
#a5*(T**4)/5 + b1/T)*R*T
else
propiedadesO2=(a1*log(T) + a2*T + a3*(T**2)/2 + a4*(T**3)/3 +
#a5*(T**4)/4 + b2)*R
endif

end

function propiedadesH(T,l)

!       l = 1   cp
!       l = 2   h
!       l = 3   s

```

```

integer    l
real*8     T, a1, a2, a3, a4, a5, b1, b2, R

R = 8.314

if (T>1000) then
a1 = 2.50000286
a2 = -5.65334214E-9
a3 = 3.63251723E-12
a4 = -9.19949720E-16
a5 = 7.95260746E-20
b1 = 2.54736589E4
b2 = -4.46698494E-1
else
a1 = 2.5
a2 = 0.0
a3 = 0.0
a4 = 0.0
a5 = 0.0
b1 = 2.54736599E4
b2 = -4.46682853E-1
endif

if (l.eq.1) then
propriedadesH = (a1 + a2*T + a3*T**2 + a4*T**3 + a5*T**4)*R
elseif (l.eq.2) then
propriedadesH = (a1 + a2*T/2 + a3*(T**2)/3 + a4*(T**3)/4 +
#a5*(T**4)/5 + b1/T)*R*T
else
propriedadesH=(a1*log(T) + a2*T + a3*(T**2)/2 + a4*(T**3)/3 +
#a5*(T**4)/4 + b2)*R
endif

end

function  propiedadesH2(T,l)

!         l = 1    cp
!         l = 2    h
!         l = 3    s

integer    l
real*8     T, a1, a2, a3, a4, a5, b1, b2, R

R = 8.314

if (T>1000) then
a1 = 2.93286579
a2 = 8.2660796E-4
a3 = -1.46402335E-7
a4 = 1.54100359E-11
a5 = -6.88804432E-16
b1 = -8.13065597E2
b2 = -1.02432887
else
a1 = 2.34433112
a2 = 7.98052075E-3
a3 = -1.9478161E-5
a4 = 2.01572094E-8
a5 = -7.37611781E-12
b1 = -9.17935173E2
b2 = 6.83010238E-1
endif

if (l.eq.1) then
propriedadesH2 = (a1 + a2*T + a3*T**2 + a4*T**3 + a5*T**4)*R
elseif (l.eq.2) then
propriedadesH2 = (a1 + a2*T/2 + a3*(T**2)/3 + a4*(T**3)/4 +

```

```

#a5*(T**4)/5 + b1/T)*R*T
else
  propiedadesH2=(a1*log(T) + a2*T + a3*(T**2)/2 + a4*(T**3)/3 +
#a5*(T**4)/4 + b2)*R
endif

end

function propiedadesOH(T,l)

!      l = 1    cp
!      l = 2    h
!      l = 3    s

integer  l
real*8   T, a1, a2, a3, a4, a5, b1, b2, R

R = 8.314

if (T>1000) then
a1 = 2.83864507
a2 = 1.10725586E-3
a3 = -2.93914978E-7
a4 = 4.20524257E-11
a5 = -2.42159092E-15
b1 = 3.94395852E3
b2 = 5.84452662
else
a1 = 3.99201543
a2 = -2.40131752E-3
a3 = 4.61793841E-6
a4 = -3.88113333E-9
a5 = 1.3641147E-12
b1 = 3.61508056E3
b2 = -1.03925458E-1
endif

if (l.eq.1) then
  propiedadesOH = (a1 + a2*T + a3*T**2 + a4*T**3 + a5*T**4)*R
elseif (l.eq.2) then
  propiedadesOH = (a1 + a2*T/2 + a3*(T**2)/3 + a4*(T**3)/4 +
#a5*(T**4)/5 + b1/T)*R*T
else
  propiedadesOH = (a1*log(T) + a2*T + a3*(T**2)/2 + a4*(T**3)/
#3 + a5*(T**4)/4 + b2)*R
endif

end

function propiedadesH2O(T,l)

!      l = 1    cp
!      l = 2    h
!      l = 3    s

integer  l
real*8   T, a1, a2, a3, a4, a5, b1, b2, R

R = 8.314

if (T>1000) then
a1 = 2.67703787
a2 = 2.97318329E-3
a3 = -7.7376969E-7
a4 = 9.44336689E-11
a5 = -4.26900959E-15
b1 = -2.98858938E4
b2 = 6.88255571

```

```

else
a1 = 4.19864056
a2 = -2.0364341E-3
a3 = 6.52040211E-6
a4 = -5.48797062E-9
a5 = 1.77197817E-12
b1 = -3.02937267E4
b2 = -8.49032208E-1
endif

if (l.eq.1) then
propiedadesH2O = (a1 + a2*T + a3*T**2 + a4*T**3 + a5*T**4)*R
elseif (l.eq.2) then
propiedadesH2O = (a1 + a2*T/2 + a3*(T**2)/3 + a4*(T**3)/4 +
#a5*(T**4)/5 + b1/T)*R*T
else
propiedadesH2O = (a1*log(T) + a2*T + a3*(T**2)/2 + a4*(T**3)
#/3 + a5*(T**4)/4 + b2)*R
endif

end

function propiedadesCO2(T,l)

!      l = 1    cp
!      l = 2    h
!      l = 3    s

integer    l
real*8     T, a1, a2, a3, a4, a5, b1, b2, R

R = 8.314

if (T>1000) then
a1 = 4.63659493
a2 = 2.74131991E-3
a3 = -9.95828531E-7
a4 = 1.60373011E-10
a5 = -9.16103468E-15
b1 = -4.90249341E4
b2 = -1.93534855
else
a1 = 2.35677352
a2 = 8.98459677E-3
a3 = -7.12356269E-6
a4 = 2.45919022E-9
a5 = -1.43699548E-13
b1 = -4.83719697E4
b2 = 9.90105222
endif

if (l.eq.1) then
propiedadesCO2 = (a1 + a2*T + a3*T**2 + a4*T**3 + a5*T**4)*R
elseif (l.eq.2) then
propiedadesCO2 = (a1 + a2*T/2 + a3*(T**2)/3 + a4*(T**3)/4 +
#a5*(T**4)/5 + b1/T)*R*T
else
propiedadesCO2 = (a1*log(T) + a2*T + a3*(T**2)/2 + a4*(T**3)
#/3 + a5*(T**4)/4 + b2)*R
endif

end

function hcarbono(T)

integer    l

```

```

real*8      T, a1, a2, a3, a4, a5, b1, b2, R

R = 8.314

if (T>1000) then
a1 = 1.45571829
a2 = 1.71702216E-3
a3 = -6.97562786E-7
a4 = 1.35277032E-10
a5 = -9.67590652E-15
b1 = -6.951388144E2
b2 = -8.52583033
else
a1 = -3.10872072E-1
a2 = 4.40353686E-3
a3 = 1.90394118E-6
a4 = -6.38546966E-9
a5 = 2.98964248E-12
b1 = -1.08650794E2
b2 = 1.11382953
endif

hcarbono = (a1 + a2*T/2 + a3*(T**2)/3 + a4*(T**3)/4 +
#a5*(T**4)/5 + b1/T)*R*T

end

function scarbono(T)

integer      l
real*8      T, a1, a2, a3, a4, a5, b1, b2, R

R = 8.314

if (T>1000) then
a1 = 1.45571829
a2 = 1.71702216E-3
a3 = -6.97562786E-7
a4 = 1.35277032E-10
a5 = -9.67590652E-15
b1 = -6.951388144E2
b2 = -8.52583033
else
a1 = -3.10872072E-1
a2 = 4.40353686E-3
a3 = 1.90394118E-6
a4 = -6.38546966E-9
a5 = 2.98964248E-12
b1 = -1.08650794E2
b2 = 1.11382953
endif

scarbono = (a1*log(T) + a2*T + a3*(T**2)/2 + a4*(T**3)/3 +
#a5*(T**4)/4 + b2)*R

end

function hformref(l)

real*8      R

R = 8.314

```

```
if (l.eq.1) then
hformref = R * (-8.97226656E3)
elseif (l.eq.2) then
hformref = R * 1.76679083E4
elseif (l.eq.3) then
hformref = R * (-1.30590979E4)
elseif (l.eq.4) then
hformref = R * 5.05141013E3
elseif (l.eq.5) then
hformref = R * (-1.32936276E4)
elseif (l.eq.6) then
hformref = R * 2.99687009E4
elseif (l.eq.7) then
hformref = R * 0.0
elseif (l.eq.8) then
hformref = R * 2.62190349E4
elseif (l.eq.9) then
hformref = R * 0.0
elseif (l.eq.10) then
hformref = R * 4.73234213E3
elseif (l.eq.11) then
hformref = R * (-2.90848168E4)
elseif (l.eq.12) then
hformref = R * (-4.73281047E4)
else
hformref = R * 0.0
endif
```

```
end
```