

UNIVERSIDADE DE SÃO PAULO
DEPARTAMENTO DE ENGENHARIA MECÂNICA

SIMULAÇÃO DE ESCOAMENTO REATIVO
COMBUSTÃO DE MONÓXIDO DE CARBONO

Luiz Tobaldini Neto

Orientador: Euryale de Jesus Zerbini



São Paulo
2002

UNIVERSIDADE DE SÃO PAULO
DEPARTAMENTO DE ENGENHARIA MECÂNICA

SIMULAÇÃO DE ESCOAMENTO REATIVO
COMBUSTÃO DE MONÓXIDO DE CARBONO

Luiz Tobaldini Neto

Orientador: Euryale de Jesus Zerbini

São Paulo
2002

Aos meus pais
Célia e Luiz Carlos
sempre presentes...

Agradecimentos

Agradeço ao Prof. Zerbini pela orientação indispensável à realização deste trabalho, pelas conversas e principalmente por me ajudar a entender o fascínio das ciências térmicas.

A meus colegas engenheiros Rui, Tafs, Nicolau e Marcelo pelas idéias, consultas e principalmente pelo estímulo a fazer bem feito.

À minha mãe e à minha irmã pela paciência e apoio nas inumeráveis horas dedicadas a essa Escola.

Resumo

Neste trabalho estamos interessados em desenvolver um modelo para simulação de escoamentos reativos. Particularizamos para o caso da oxidação do monóxido de carbono de modo que, sem perda de generalidade possamos compreender alguns fenômenos básicos deste tipo de problema, possibilitando posteriormente o aperfeiçoamento do código para contemplar outros mecanismos de reação. Utilizamos um mecanismo de 4 reações para a modelar a cinética química e fazemos um modelo do escoamento plugado (Plug-flow). A oxidação do monóxido de carbono não ocorre sem a catálise por uma substância hidrogenada, que no nosso caso será a água. Dificuldades para a convergência do algoritmo foram encontradas e a análise de cada uma das reações isoladamente permitiu identificar que uma provável causa para esses problemas está nos coeficientes do modelo cinético.

Abstract

In the present work we intend to develop and simulate a reactive flow model. We will particularize this model to the carbon monoxide oxidation mechanism, so we can understand some basic phenomena of this sort of problem without losing generality. The source code can be easily modified so that it would be able to model other reactions mechanisms as well. We will use a 4 reactions mechanism to model chemical kinetics and the flow will be modelled as a Plug-flow reactor. The oxidation of carbon monoxide does not occur at low temperatures without the presence of a catalytical hydrogenous substance, that in our case would be water. Some convergence difficulties came up during the simulations and the analysis of each reaction separately allowed us identify that these problems can be related to the chemical kinetics modelling strategy.

Sumário

1	Introdução	1
1.1	Motivação	2
2	Introdução Teórica à Cinética Química	3
2.1	A expressão de Arrhenius para a taxa de reação	3
3	Reator Plug-Flow	6
3.1	Aplicação das leis de conservação	7
4	O mecanismo da reação do CO	12
4.1	Taxas de reação e constantes utilizadas	13
4.2	Notação matricial para solução da cinética química	13
5	Simulações e Resultados	16
5.1	Caso 1 (baseline)	16
5.2	Caso 2 - variando UA	19
5.3	Caso 3 - variando UA e ϕ	21
5.3.1	Discussão sobre os resultados	24
5.4	Casos para reações isoladas	25
5.4.1	Conclusões sobre testes com reações isoladas	25
6	Conclusões Gerais	32
	Bibliografia	34

Lista de Figuras

3.1	Discretização do escoamento - Volumes de Controle	8
5.1	Temperatura - 1	17
5.2	Frações Molares - 1	17
5.3	Frações Molares - detalhe - 1	18
5.4	Taxas de reação - 1	18
5.5	Temperatura - 2	19
5.6	Frações Molares - 2	20
5.7	Frações Molares - detalhe - 2	20
5.8	Taxas de reação - 2	21
5.9	Temperatura - 3	22
5.10	Frações Molares - 3	22
5.11	Frações Molares - detalhe - 3	23
5.12	Taxas de reação - 3	23
5.13	Temperatura - 4	26
5.14	Frações Molares - 4	26
5.15	Taxas de reação - 4	27
5.16	Temperatura - 5	27
5.17	Frações Molares - 5	28
5.18	Taxas de reação - 5	28
5.19	Temperatura - 6	29
5.20	Frações Molares - 6	29
5.21	Taxas de reação - 6	30
5.22	Temperatura - 7	30
5.23	Frações Molares - 7	31
5.24	Taxas de reação - 7	31

Capítulo 1

Introdução

Neste trabalho será feito um estudo do modelo cinético do mecanismo de oxidação do CO em mistura com água. Este mecanismo é composto por quatro reações básicas como veremos no decorrer do trabalho.

Partindo-se das quatro equações de conservação (massa, energia, momento e espécies) para um volume de controle será desenvolvido um modelo para um reator Plug-Flow. Este modelo será implementado em um código computacional para integração das equações diferenciais que o descrevem. Este código foi implementado utilizando-se a linguagem de programação FORTRAN. Esta opção foi feita pois a linguagem se mostra bastante eficiente na resolução de problemas matemáticos, além de possuir alguns recursos que tornam bastante clara e direta a leitura do código fonte.

Utilizamos primeiramente integração via método de Euler, ou seja, fazemos uma aproximação em primeira ordem da integral das equações diferenciais. Veremos também no momento oportuno que a alternativa utilizando-se métodos de ordem superior, como o método de Runge-Kutta de quarta ordem, não traz benefícios significativos frente ao custo computacional que representa.

O objetivo deste trabalho como proposto no início deste ano é a iniciação ao estudo de *escoamentos reativos*. Esta é uma área cuja variedade de aplicações é indiscutivelmente vasta, indo desde problemas de combustão, como o aqui estudado, às inúmeras possibilidades existentes na indústria química e de processos, por exemplo.

A integração entre cinética química, mecânica dos fluidos e termodinâmica propicia um campo de estudos que além de interessante pode-se dizer até fascinante. Nestas áreas encontramos fenômenos que ainda constituem campo de

intensa investigação e que ainda são pouco compreendidos por nós, ou melhor dizendo, que oferecem ainda muito por se descobrir.

1.1 Motivação

Tendo definido que o objetivo do trabalho seria desenvolver um modelo numérico de um escoamento reativo, escolheu-se a reação de oxidação do monóxido de carbono. O mecanismo da reação é relativamente simples sendo composto por apenas quatro reações, que nos leva a um número conveniente de equações, permitindo a integração do sistema de EDO's sem grandes custos computacionais mas ainda mantendo certa complexidade que permita efetivamente o entendimento dos processos de interação entre as diversas reações do modelo cinético.

Este é um problema de grande interesse para a indústria siderúrgica. A operação racional e econômica dos fornos de reaquecimento de placas exige que o processo seja conduzido de forma a maximizar sua economia de energia e seu rendimento metálico, sem contudo afetar a homogeneidade do material.

Uma das ferramentas mais poderosas no sentido de se otimizar a condução deste processo são modelos matemáticos eficazes e comprovadamente aferidos em relação à realidade operacional da planta. Os modelos matemáticos de fornos de reaquecimento de placas podem ser utilizados, viabilizando a simulação de diversas condições operacionais e seus reflexos na qualidade do encharque do material, sua condição superficial, consumo de energia, emissão de poluentes, etc. Os resultados obtidos por esses modelos permitem definir as condições otimizadas de processo.

Suas paredes internas são constituídas de revestimento refratário sílico-aluminoso, com cerca de 50% de alumina. Estes fornos utilizam como combustíveis gás de coqueria, gás natural ou óleo combustível. Seus recuperadores são do tipo cerâmico e a chaminé possui tiragem natural. Sua potência de aquecimento é da ordem de 370 Mcal/t, considerando-se um ritmo médio de produção igual a 120 t/h de placas.

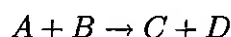
Como o gás de coqueria é um gás altamente rico em monóxido de carbono, o entendimento do fenômeno de combustão deste gás é de extrema importância para a análise do desempenho deste tipo de fornos

Capítulo 2

Introdução Teórica à Cinética Química

2.1 A expressão de Arrhenius para a taxa de reação

A maioria das reações químicas têm suas taxas de reação TR dominadas por colisões de moléculas de duas das espécies envolvidas, que devam ter a capacidade de reagir. Assim, o tipo mais simples de reação são as de segunda ordem (duas espécies reagentes). Outras reações são dominadas por um estágio de quebra da ligação molecular, sendo assim chamadas de reações de primeira ordem. A maioria das reações deste último tipo caem na classe dos processos de decomposição. Em uma reação de segunda ordem arbitrária podemos escrever:



Sendo todos os coeficientes estequiométricos iguais a 1, como no exemplo, temos:

$$TR = \frac{d[A]}{dt} = -k[A][B] = -\frac{d[C]}{dt}$$

A convenção usada neste trabalho é que os colchetes envolvendo um símbolo químico se referem à concentração molar daquela espécie por metro cúbico. Especificando a reação da maneira acima não significa que toda colisão entre os reagentes A e B levariam à formação de produtos ou a sua decomposição.

Arrhenius deu início a uma teoria que leva em conta este fato e atribui a k uma dependência da temperatura. Ele afirma que apenas as moléculas que possuem energia acima de uma quantidade E irão reagir. Moléculas adquirem a energia

adicional necessária das colisões induzidas por condições térmicas existentes. Estas moléculas com alta energia levam à formação de produtos. O postulado de Arrhenius pode ser escrito da seguinte forma:

$$TR = -Z_{AB} \exp\left(\frac{-E}{RT}\right) \quad (2.1)$$

ou, expandindo Z

$$TR = \alpha[A][B] \exp\left(\frac{-E}{RT}\right)[A][B] \quad (2.2)$$

onde Z_{AB} é frequência cinética de colisão do gás que é também função das concentrações dos reagentes e de uma constante α . $\exp(-E/RT)$ é o fator de Boltzmann. A teoria cinética mostra que o fator de Boltzmann representa a fração de colisões que têm energia superior a E .

Experimentalmente, o valor de E é encontrado plotando-se os valores medidos de $\ln k$ em função de $1/T$ e computando-se a inclinação da curva que melhor interpola o conjunto de dados. A inclinação da curva corresponde ao valor $-E/R$. Uma vez que obtivemos o valor de E , Z poderá ser calculado a partir da equação 2.1 e dos valores medidos de k . Frequentemente E pode ser obtido dentro de uma pequena margem de erro, no entanto as dificuldades inerentes ao processo de medição da taxa de reação produzem incertezas em α que podem exceder o fator de 2. Se a reação não for uma reação elementar, então o valor de E obtido desta maneira será chamado de *energia de ativação geral*.

O entendimento das magnitudes e da dependência da temperatura das taxas de reação em reações bimoleculares entre gases é de importância fundamental para a modelagem de reações de combustão. A interpretação teórica da dependência da temperatura dos coeficientes de reação bimoleculares é um tópico de alta complexidade. Isto se deu principalmente pelo desenvolvimento de técnicas experimentais mais apuradas e conseqüentemente pela geração de bancos de dados mais precisos. Atualmente dados já foram colhidos para uma faixa muito mais larga de temperaturas. Este desenvolvimento tem sido contínuo. Para algumas reações as expressões para suas taxas já estão bem definidas e até comprovadas por métodos teóricos. Este não é, porém, o caso geral para reações bimoleculares em fase gasosa. Para grande parte das reações importantes temos um grande desvio entre os dados existentes. Avanços experimentais e teóricos são esperados, permitindo ao menos que as reações mais utilizadas em sistemas de combustão

sejam discutidas com um grau de confiabilidade maior.

Capítulo 3

Reator Plug-Flow

O acoplamento da termodinâmica com a mecânica dos fluidos e a cinética química nos permite descrever a evolução detalhada do sistema desde o estado inicial dos reagentes até o estado final dos produtos, que podem ou não estar em equilíbrio químico. Ou seja, poderemos assim calcular a concentração das espécies como função do tempo à medida que a reação acontece.

Um reator Plug-Flow é um tipo idealizado de reator que possui os seguintes atributos:

- O escoamento se dá em regime permanente;
- Não existe mistura na direção axial. Isso significa que a difusão turbulenta ou molecular de massa é desprezível na direção do escoamento;
- As propriedades são constantes na direção perpendicular ao escoamento, isto é, temos um escoamento unidimensional. Isto significa que em qualquer seção transversal, uma única velocidade, temperatura, composição etc. caracterizam completamente o escoamento. Para esse problema em especial usaremos primeiramente um tubo de seção cilíndrica de área constante ao longo da direção do escoamento;
- Escoamento ideal invíscido;
- Comportamento de gás ideal. Esta hipótese permite que relações de estado simples possam ser usadas para relacionar temperatura, pressão, densidade, frações molares e entalpia;

3.1 Aplicação das leis de conservação

O objetivo é desenvolver um sistema de equações diferenciais de primeira ordem cuja solução descreva as propriedades do escoamento no reator, incluindo a composição, em função da distância (x) percorrida no interior do reator. A definição da geometria e do sistema de coordenadas está indicado na figura. Aplicando-se as equações de conservação ao sistema obteremos o sistema de EDO's desejado. Os seguintes parâmetros são considerados conhecidos: a vazão mássica $\dot{m}$, k_i , que é a taxa de progresso da reação, a temperatura (T), $A(x)$ e W'' . A função de área $A(x)$ permite utilizar a mesma modelagem em qualquer tipo de reator unidimensional, podendo representar um bocal ou um difusor, por exemplo, sem limitar-se a um duto cilíndrico. O fluxo de calor $W(x)$, apesar de explicitamente indicar que o fluxo de calor na parede é conhecido, permite também indicar que o fluxo de calor pode ser determinado a partir de uma distribuição de temperatura dada na parede. Do controle de volume representado na figura 3.1, podemos derivar as seguintes relações:

- Conservação de massa

$$\frac{d(\rho v_x A)}{dx} = 0 \quad (3.1)$$

- Conservação de momento em X

$$\frac{dP}{dx} + \rho v_x \frac{dv_x}{dx} = 0 \quad (3.2)$$

- Conservação de energia

$$\frac{dh + \frac{v_x^2}{2}}{dx} + \frac{W'' \zeta}{\dot{m}} = 0 \quad (3.3)$$

- Conservação de espécies

$$\frac{dY_i}{dx} - \frac{\dot{\omega}_i M W_i}{\rho v_x} \quad (3.4)$$

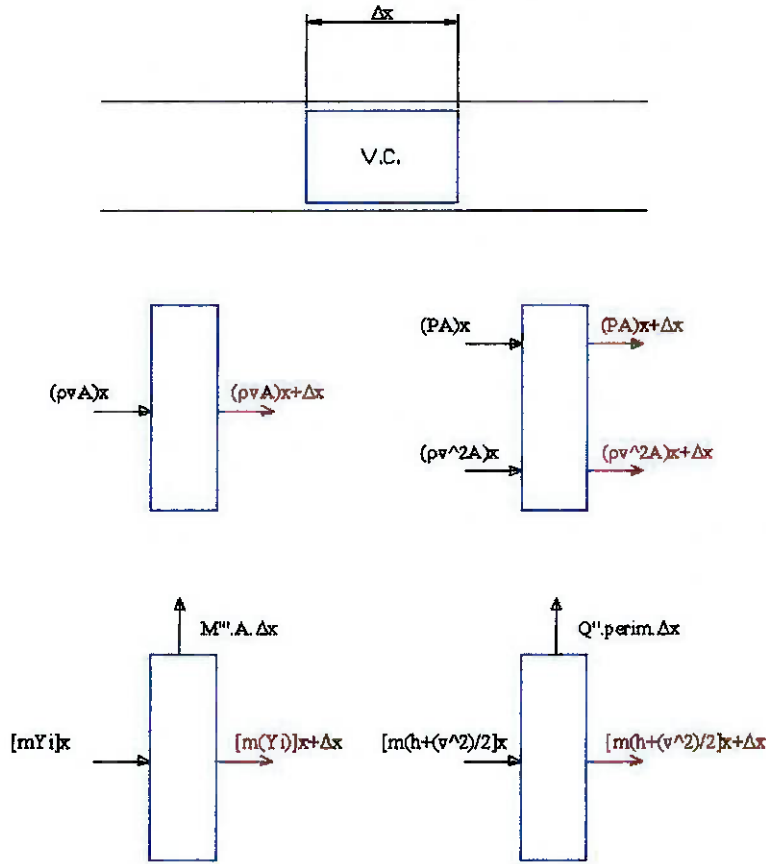


Figura 3.1: Discretização do escoamento - Volumes de Controle

Os símbolos v_x e ζ representam a velocidade axial e o perímetro do reator, respectivamente. Todas as outras quantidades já foram definidas previamente. Rearranjando as equações 3.1 e 3.3 iniciamos o processo de isolamento das variáveis desejadas:

$$\frac{1}{\rho} \frac{d\rho}{dx} + \frac{1}{v_x} \frac{dv_x}{dx} + \frac{1}{A} \frac{dA}{dx} = 0 \quad (3.5)$$

$$\frac{dh}{dx} + v_x \frac{dv_x}{dx} + \frac{W\zeta}{\dot{m}} = 0 \quad (3.6)$$

Os $\dot{\omega}_i$ da equação 3.4 podem ser expressados usando equação 4.9 com o $[X]_i$ transformado para Y_i .

Para gás ideal a relação funcional,

$$h = h(t, Y_i), i = 1, 2, \dots, N \quad (3.7)$$

pode ser expandida usando a regra da cadeia para relacionar dh/dx e dT/dx , levando a

$$\frac{dh}{dx} = c_p \frac{dT}{dx} + \sum_{i=1}^N h_i \frac{dY_i}{dx} \quad (3.8)$$

Nesta equação temos que as entalpias h_i são as entalpias de formação dos componentes à temperatura T , de modo que esses valores devem ser negativos na nossa convenção de que calor positivo é aquele

para completar nossa descrição matemática do reator plug-flow, nós diferenciamos a equação de estado de gás ideal,

$$P = \frac{\rho RT}{MW_{mix}} \quad (3.9)$$

obtendo:

$$\frac{1}{P} \frac{dP}{dx} = \frac{1}{\rho} \frac{d\rho}{dx} + \frac{1}{T} \frac{dT}{dx} - \frac{1}{MW_{mix}} \frac{dMW_{mix}}{dx} \quad (3.10)$$

onde o peso molecular da mistura deriva da sua definição em termos de frações mássicas:

$$\frac{dMW_{mix}}{dx} = -MW_{mix}^2 \sum_{i=1}^N \frac{1}{MW_i} \frac{dY_i}{dx} \quad (3.11)$$

O número de equações pode ser reduzido eliminando algumas das derivadas acima por substituição. Uma escolha lógica possível é manter as derivadas $d\rho/dx$, dT/dx e dY_i/dx .

Sabemos que:

$$\frac{dY_i}{dx} = \frac{\dot{\omega}_i MW_i}{\rho v_x} \quad (3.12)$$

Da equação 3.5 tiramos:

$$\frac{dv_x}{dx} = -\frac{v_x}{A} \frac{dA}{dx} - \frac{v_x}{\rho} \frac{d\rho}{dx} \quad (3.13)$$

Substituindo-se a equação acima e a equação 3.8 em 3.6 temos:

$$c_p \frac{dT}{dx} + \frac{1}{v_x \rho} \sum_{i=1}^N h_i \dot{\omega}_i MW_i + v_x \left(-\frac{v_x}{A} \frac{dA}{dx} - \frac{v_x}{\rho} \frac{d\rho}{dx} \right) + \frac{\dot{Q}'' \zeta}{\dot{m}} = 0 \quad (3.14)$$

Podemos ainda, calcular o fluxo de calor por unidade de comprimento de duto q' :

$$\frac{\dot{Q}'' \zeta}{\dot{m}} = q' \quad (3.15)$$

Dividindo-se a equação 3.14 por c_p , e substituindo nela a relação para o fluxo unitário de calor acima:

$$\frac{dT}{dx} = \frac{v_x^2}{\rho c_p} \frac{d\rho}{dx} + \frac{v_x^2}{c_p} \left(\frac{1}{A} \frac{dA}{dx} \right) - \frac{1}{v_x \rho c_p} \sum_{i=1}^N h_i \dot{\omega}_i MW_i - \frac{q'}{c_p} \quad (3.16)$$

Da equação de conservação do momento na direção x 3.2 temos:

$$\frac{dv_x}{dx} = -\frac{1}{\rho v_x} \frac{dP}{dx} \quad (3.17)$$

Substituindo-se a equação acima e a equação 3.11 na equação 3.5 obteremos:

$$\frac{d\rho}{dx} = \frac{1}{v_x^2} \frac{dP}{dx} - \frac{\rho}{A} \frac{dA}{dx} = 0 \quad (3.18)$$

Isolando-se o termo dP/dx da equação 3.10 e substituindo na equação acima teremos:

$$\frac{d\rho}{dx} = \frac{P}{\rho v_x^2} \frac{d\rho}{dx} + \frac{P}{v_x^2 T} \frac{dT}{dx} - \frac{P}{MW_{mix} v_x^2} \frac{dMW_{mix}}{dx} - \rho \left(\frac{1}{A} \frac{dA}{dx} \right) \quad (3.19)$$

Fazendo-se as devidas manipulações algébricas e substituindo-se o termo dT/dx da equação 3.20 na equação anterior obtemos:

$$\frac{d\rho}{dx} = \frac{\left(1 - \frac{R}{c_p MW_{mix}} \right) \rho^2 v_x^2 \left(\frac{1}{A} \frac{dA}{dx} \right) + \frac{q'' \zeta \rho}{T c_p v_x} + \frac{\rho R}{v_x c_p MW_{mix}} \sum_{i=1}^N MW_i \dot{\omega}_i \left(h_i - \frac{MW_{mix}}{MW_i} c_p T \right)}{P \left(1 + \frac{v_x^2}{c_p T} \right) - \rho v_x^2} \quad (3.20)$$

Com esta escolha, as equações 3.20, 3.16 e 3.12 compõem o nosso sistema de EDO's, que deverá ser integrado a partir de condições iniciais apropriadas.

O tempo de residência t_R pode ser definido por:

$$\frac{dt_R}{dx} = \frac{1}{v_x} \quad (3.21)$$

e também será incluído no nosso sistema de EDO's.

Uma quinta equação para dA/dx será incluída, de modo a generalizarmos a análise para várias geometrias axissimétricas. Para a análise de um cilindro teremos naturalmente que:

$$\frac{dA}{dx} = 0 \quad (3.22)$$

As condições iniciais necessárias para resolver as equações 3.20-3.21 são:

$$T(0) = T_0$$

$$\rho(0) = \rho_0$$

$$Y_i(0) = Y_{i,0}$$

$$t_R(0) = 0$$

Quando da implementação todas unidades foram utilizadas no sistema SI/mol/K.

Capítulo 4

O mecanismo da reação do CO

Apesar da oxidação do monóxido de carbono ter uma grande importância por si só, como já citado na Introdução, sendo essa importância nossa principal motivação neste trabalho, este mecanismo é também muito importante na oxidação de hidrocarbonetos. A combustão de hidrocarbonetos pode ser resumidamente caracterizada por duas fases: a quebra do combustível em monóxido de carbono e a oxidação propriamente deste a dióxido de Carbono.

Os calores de combustão do CO e do H_2 são respectivamente 67,6 e 57,8 kcal/mol, mostrando que a energia latente de uma mistura de CO e O_2 é maior que a energia de uma mistura correspondente de H_2 e O_2 . No entanto isso não significa que misturas de CO e O_2 são mais reativas do que misturas de H_2 e O_2 . Ao contrário, como será discutido a seguir, tem-se mostrado que misturas contendo apenas CO e O_2 e nenhum traço de impurezas hidrogenadas podem ser virtualmente não reagentes. Não é possível identificar em um sistema químico contendo apenas CO e O_2 um mecanismo que leve à reação em cadeia, como ocorre com H_2 e O_2 . Porém, pequenas quantidades de H_2O ou H_2 podem ter um efeito muito significativo na taxa de oxidação do CO . Com a presença do hidrogênio os dois sistemas químicos são conectados em um mecanismo de reação em cadeia. O Isto acontece porque a oxidação do CO envolvendo o radical hidroxila é muito mais rápida que os passos envolvendo apenas O_2 ou O . [2]

Sendo a água a substância hidrogenada a ser considerada podemos descrever a reação de oxidação do CO através do seguinte mecanismo: [6]





A reação 4.1 é lenta e não contribui significativamente para a formação de CO_2 , mas serve apenas para iniciar o processo em cadeia. O processo de oxidação do CO efetivamente é descrito pela reação 4.3. Essa reação também dá continuidade ao processo liberando átomos de hidrogênio, H , que reagem com O_2 para formar OH e O na reação 4.4 realimentando as reações 4.2 e 4.3.

4.1 Taxas de reação e constantes utilizadas

Pesquisando-se na literatura disponível sobre o tema pode-se observar que os valores existentes para as taxas de reação são bastante diferentes, dependendo da fonte da pesquisa. Isso se dá porque temos um sem-número de possibilidades experimentais para se levantar estes valores. Os coeficientes da expressão de Arrhenius

4.2 Notação matricial para solução da cinética química

Em mecanismos que envolvam diversas equações elementares, como o aqui proposto, é conveniente utilizar-se de uma notação compacta adequada para representar tanto o mecanismo como a taxa de produção das espécies. Podemos então escrever:

$$\sum_{i=1}^N \nu'_{ji} X_j \Leftrightarrow \sum_{j=1}^N \nu''_{ji} X_j \quad (4.5)$$

para $i = 1, 2 \dots L$

onde ν''_{ji} e ν'_{ji} são os coeficientes estequiométricos do elemento j na reação i do lado dos reagentes e dos produtos respectivamente. Vamos tomar como exemplo o nosso sistema, com as reações de 4.1 a 4.4. Nele incluíremos uma quinta reação fictícia, para representar a presença de N_2 na mistura. Como o N_2 será por nós considerado inerte, a quinta reação será:



Este recurso facilitará a implementação do programa, de maneira a manter o algoritmo bem estruturado e generalizável.

Definiremos então j e i como segue:

j	Componente	i	Reação
1	CO	1	$CO + O_2 \rightarrow CO_2 + O$
2	O ₂	2	$O + H_2O \rightarrow OH + OH$
3	CO ₂	3	$CO + OH \rightarrow CO_2 + H$
4	O	4	$H + O_2 \rightarrow OH + O$
5	H ₂ O	5	$N_2 \rightarrow N_2$
6	OH		
7	H		
8	N ₂		

Usando-se o j como índice das colunas e i como o índice para as linhas, podemos então escrever as matrizes

$$\nu'_{ji} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.88 \end{bmatrix} \quad (4.7)$$

e

$$\nu''_{ji} = \begin{bmatrix} 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1.88 \end{bmatrix} \quad (4.8)$$

Os coeficientes para o N_2 são aqui indiferentes, desde que sejam iguais tanto do lado dos reagentes como do lado dos produtos. Utilizaremos o coeficiente 1,88 para relembrar que estamos utilizando então 0,5 mol de ar por mol de CO, na reação estequiométrica.

com as três relações que seguem completamos a notação compacta utilizada:

$$\omega_j = \sum_{i=1}^L \nu_{ji} q_i \quad (4.9)$$

para $j=1,2,\dots,N$

$$\nu_{ji} = \nu''_{ji} - \nu'_{ji} \quad (4.10)$$

$$q_i = k_{fi} \prod_{j=1}^N [X_j]^{\nu'_{ji}} - k_{ri} \prod_{j=1}^N [X_j]^{\nu''_{ji}} \quad (4.11)$$

O termo $\omega_j \equiv \frac{d[X_j]}{dt}$ é a **taxa de produção da espécie j** para o mecanismo completo. A equação 4.11 define a variável **taxa de progresso** para a reação elementar i .

Capítulo 5

Simulações e Resultados

Um caso base (baseline) foi escolhido para simulação do programa, a partir do qual deverão ser feitas as comparações variando-se alguns dos parâmetros da simulação, permitindo assim entender a influência destes no comportamento do nosso reator.

5.1 Caso 1 (baseline)

Utilizamos as seguintes entradas:

$$\dot{m} = 0,1kg/s$$

$$diam = 0,1m$$

$$p = 101325Pa - \text{Pressão atmosférica}$$

$$T_{wall} = 1100K$$

$$T_0 = 300K$$

$$\phi = 1,0 \text{ excesso de ar em relação à estequiométrica}$$

$$UA = 0,007kg_{agua}/kg_{arseco}$$

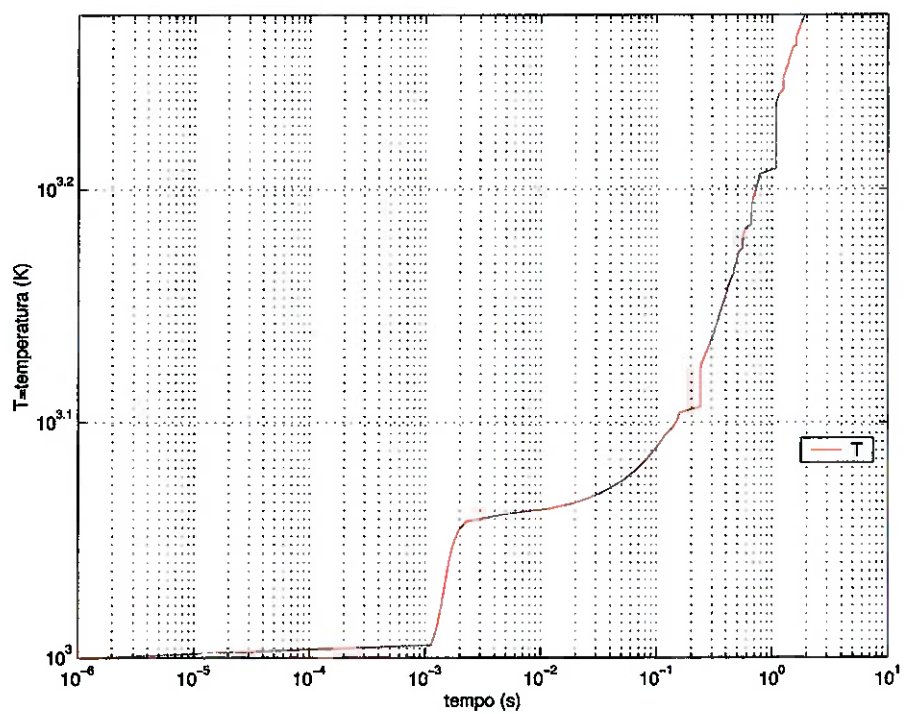


Figura 5.1: Temperatura - 1

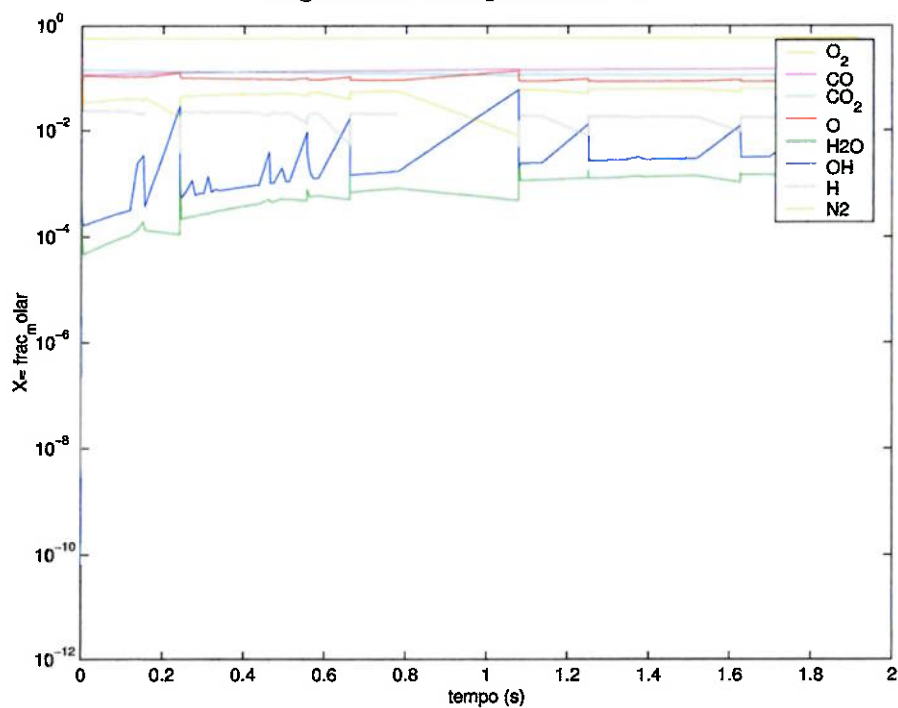


Figura 5.2: Frações Molares - 1

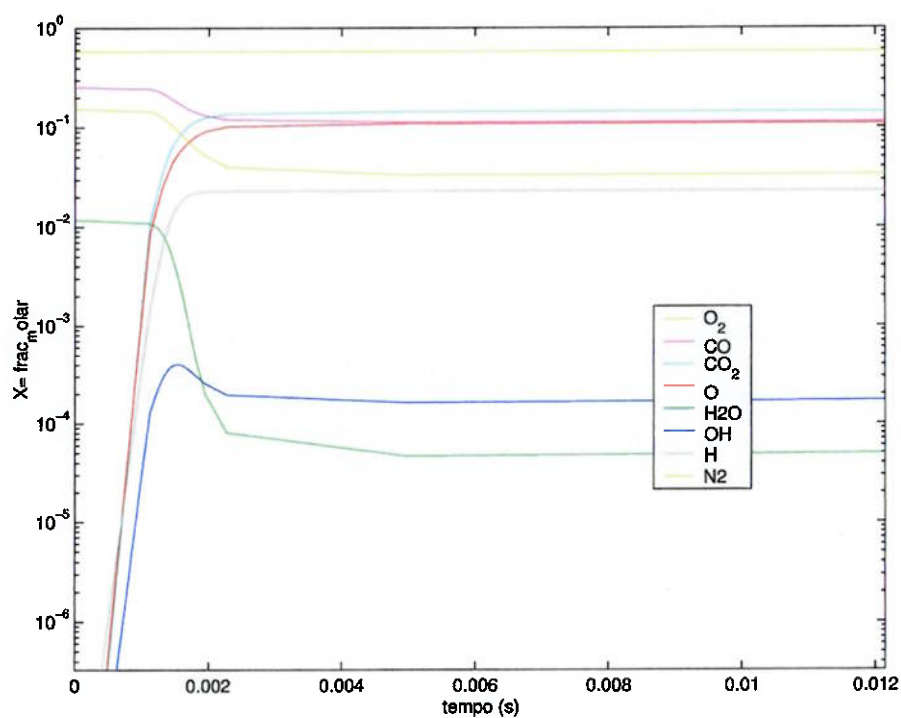


Figura 5.3: Frações Molares - detalhe - 1

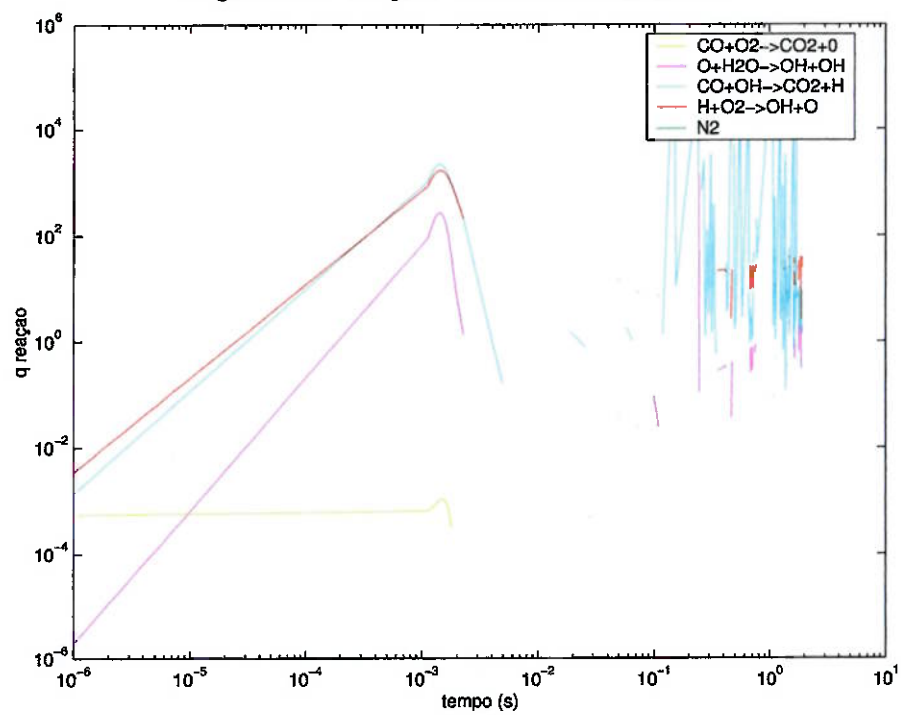


Figura 5.4: Taxas de reação - 1

5.2 Caso 2 - variando UA

Utilizamos as seguintes entradas:

$$\dot{m} = 0,1 \text{ kg/s}$$

$$\text{diam} = 0,1 \text{ m}$$

$$p = 101325 \text{ Pa} - \text{Pressão atmosférica}$$

$$T_{\text{wall}} = 1100 \text{ K}$$

$$T_0 = 300 \text{ K}$$

$$\phi = 1,0 \text{ excesso de ar em relação à estequiométrica}$$

$$UA = 0,015 \text{ kg}_{\text{agua}}/\text{kg}_{\text{ar seco}}$$

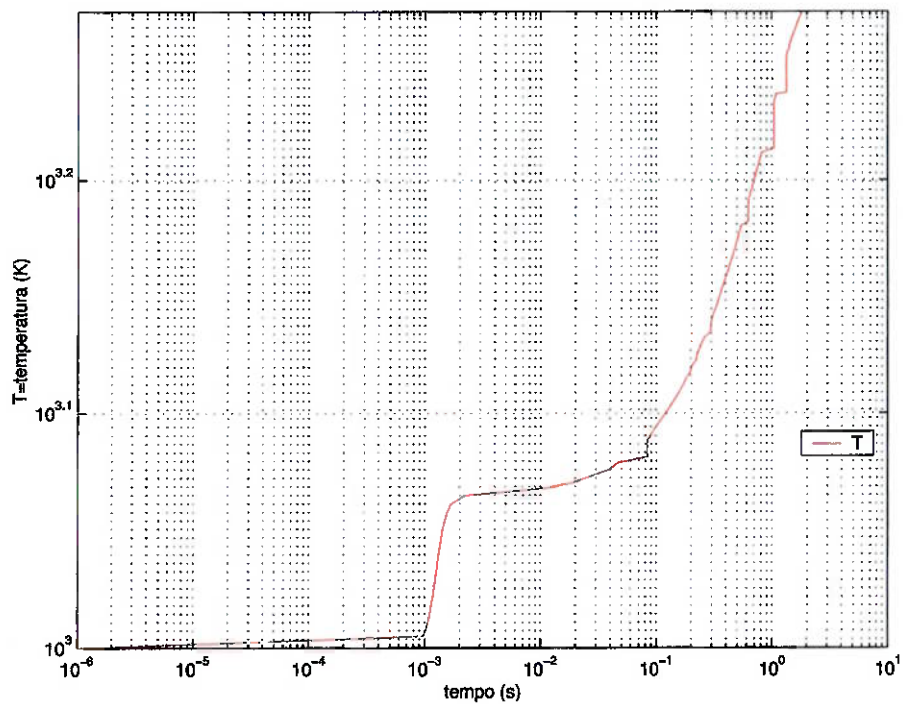


Figura 5.5: Temperatura - 2

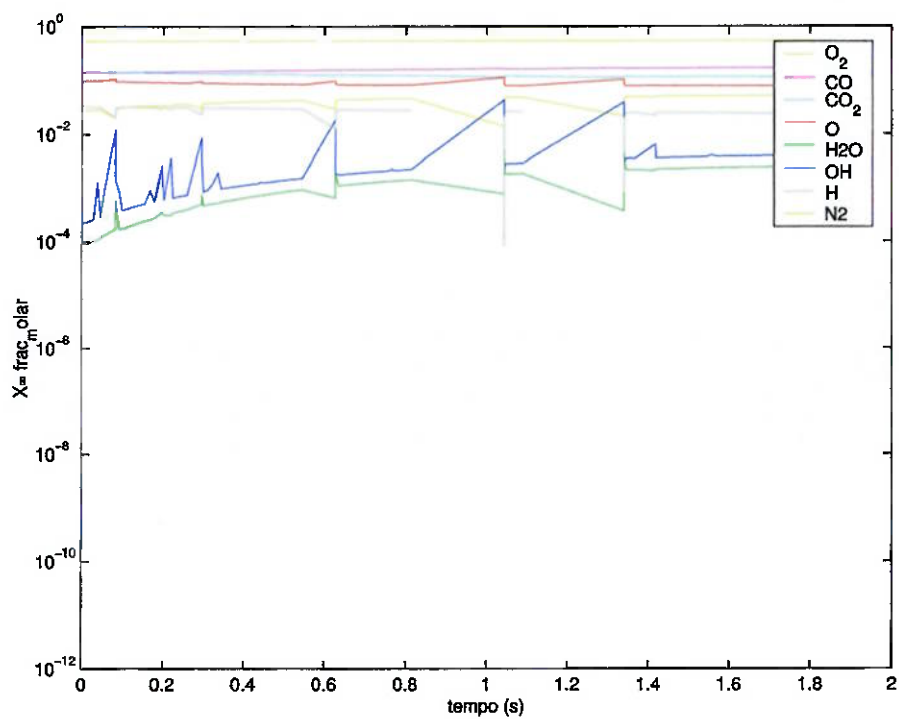


Figura 5.6: Frações Molares - 2

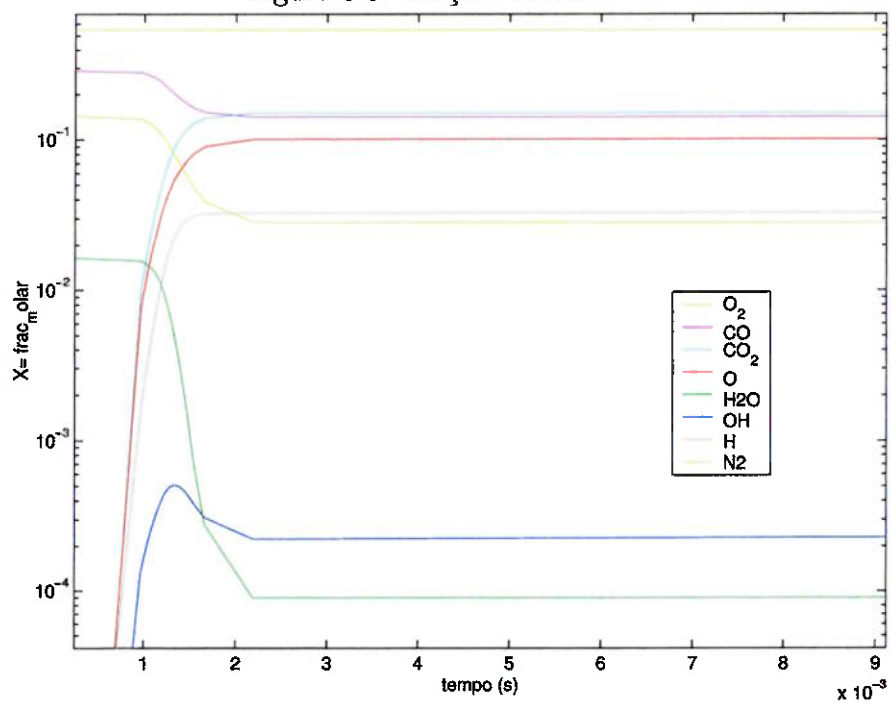


Figura 5.7: Frações Molares - detalhe - 2

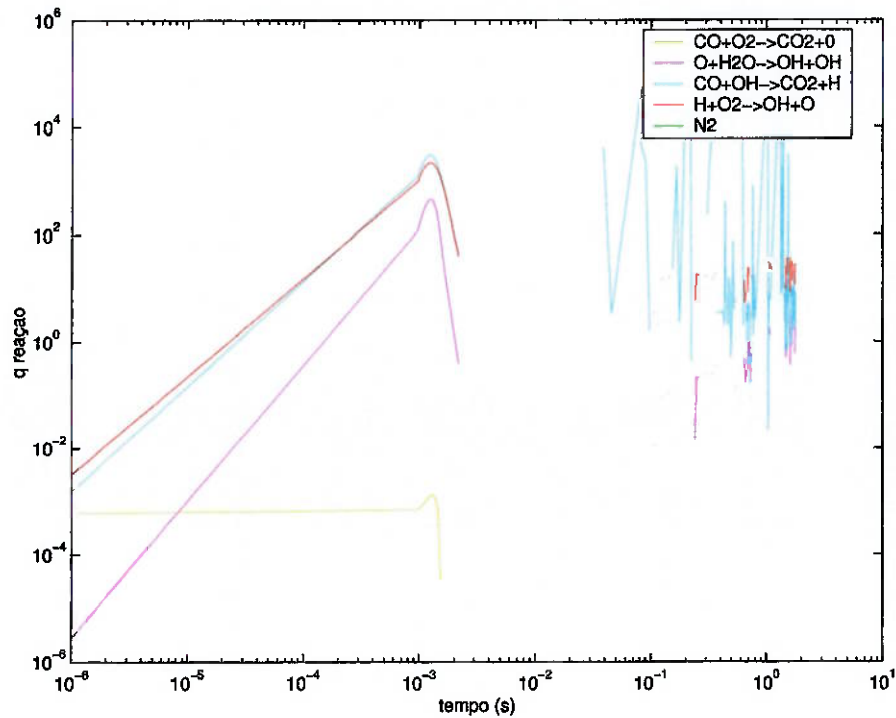


Figura 5.8: Taxas de reação - 2

5.3 Caso 3 - variando UA e ϕ

Utilizamos as seguintes entradas:

$$\dot{m} = 0,1 \text{ kg/s}$$

$$\text{diam} = 0,1 \text{ m}$$

$$p = 101325 \text{ Pa} - \text{Pressão atmosférica}$$

$$T_{\text{wall}} = 1100 \text{ K}$$

$$T_0 = 300 \text{ K}$$

$$\phi = 1,2 \text{ excesso de ar em relação à estequiométrica}$$

$$UA = 0,007 \text{ kg}_{\text{agua}}/\text{kg}_{\text{ar seco}}$$

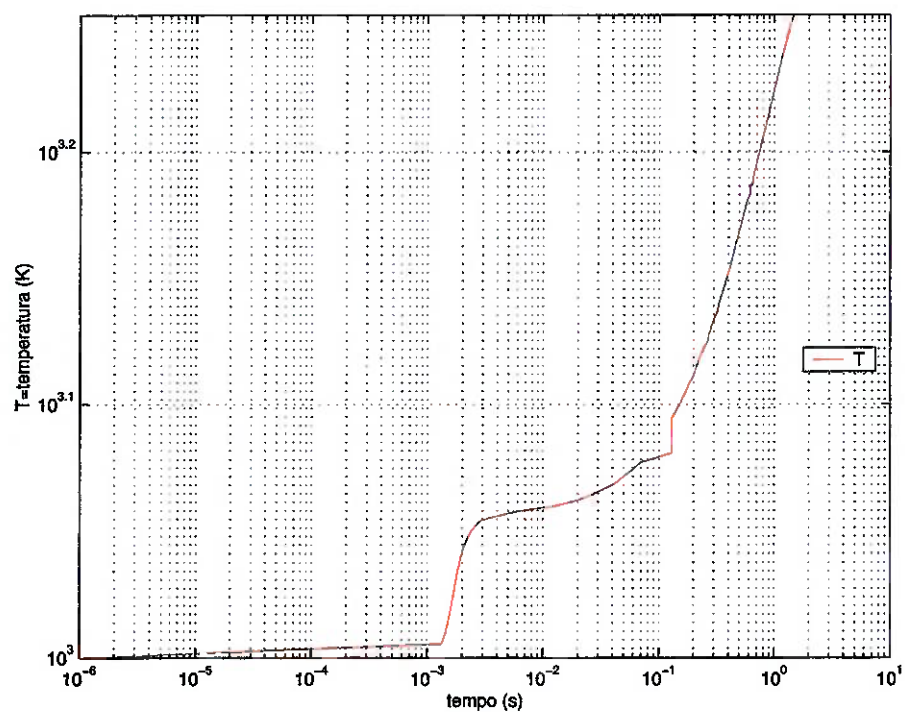


Figura 5.9: Temperatura - 3

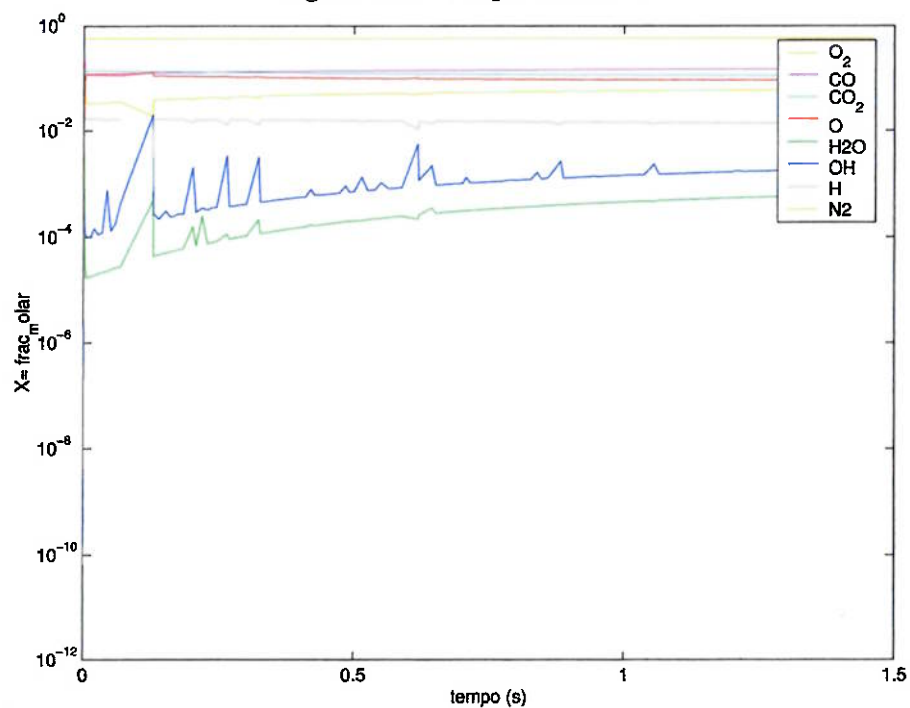


Figura 5.10: Frações Molares - 3

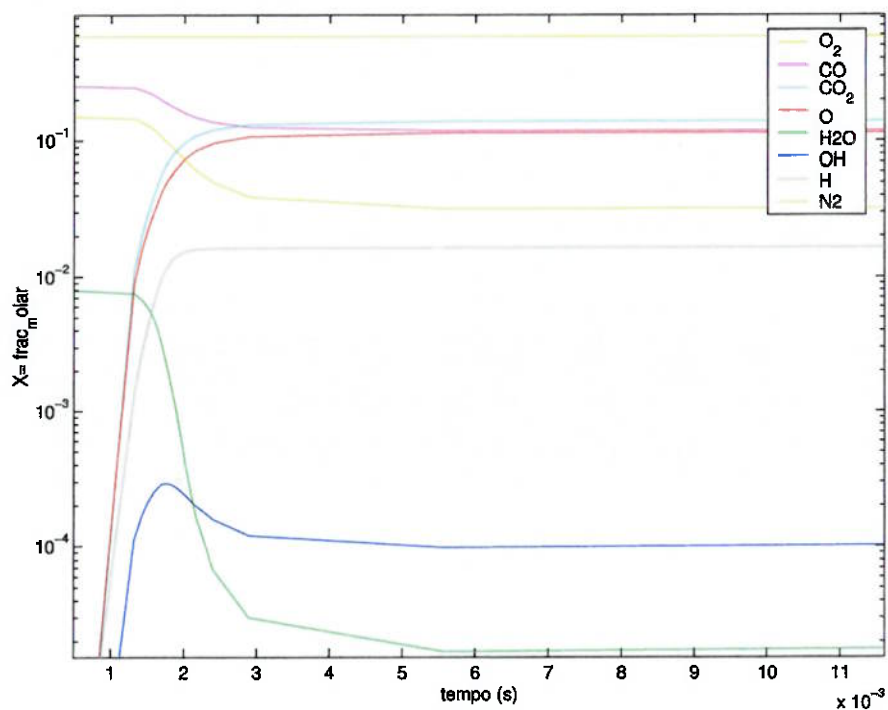


Figura 5.11: Frações Molares - detalhe - 3

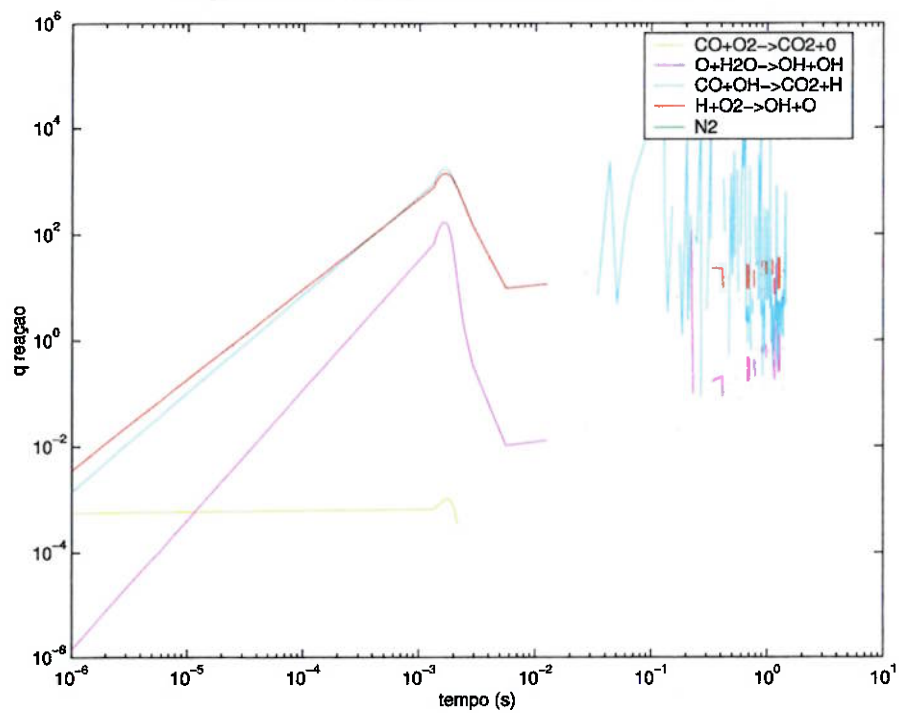


Figura 5.12: Taxas de reação - 3

5.3.1 Discussão sobre os resultados

A principal observação a ser feita a respeito dos resultados é o fato de, nos três casos apresentados, existir uma certa dificuldade do programa de identificar o ponto de estabilização da temperatura quando do consumo de O_2 . Isto acontece porque mesmo com a diminuição da concentração de O_2 , as outras reações principalmente a 4.4 continua ocorrendo, e o calor por ela liberado continua interferindo na temperatura do sistema. A não convergência do método leva a algumas reflexões sobre a natureza do problema. Duas causas principais são consideradas:

- a formulação não foi bem feita;
- o modelo cinético apresenta falhas;

A primeira hipótese cairia por terra se considerarmos que alguns testes de consistência foram realizados para a formulação, mostrando que as equações de conservação estão sendo atendidas. Estes testes foram feitos comparando-se as concentrações e temperaturas finais e iniciais. Uma alternativa aqui é testar-se novas estratégias de integração. Métodos adaptativos também foram utilizados, fazendo o passo inversamente proporcional à derivada da temperatura. Ou seja, quanto maior a tendência do sistema de divergir menor o passo, no sentido de diminuir a velocidade de avanço e atentar para possíveis variações bruscas que venham a ocorrer.

A segunda hipótese pode também ser defendida argumentando-se com o fato que já destacamos anteriormente, de que as constantes utilizadas na formulação das equações de reação na forma de Arrhenius variam muito em função da fonte de pesquisa.

O trabalho agora concentra-se no entendimento dos fenômenos que levaram à não convergência do programa, tendo postas as duas hipóteses acima.

5.4 Casos para reações isoladas

Partindo-se dos resultados mostrados na seção anterior resolveu-se investigar o comportamento do código para cada uma das 4 reações do mecanismo de maneira independente. Seguem aqui os resultados obtidos. Consideramos sempre as mesmas frações mássicas usadas como condição inicial no caso 3, de modo a entender cada reação sob condições similares às que estarão sujeitas no caso geral. A única ressalva é que não teremos obviamente a interferência das outras reações. Outro detalhe é que uma fração mássica inicial muito pequena ($1.e-10$) de cada componente é considerada na entrada apenas para que existam reagentes, para que as reações secundárias ocorram. Por reações secundárias entenda-se as reações 4.2, 4.3 e 4.4, aqui chamadas de 2, 3 e 4 respectivamente, enquanto a reação 4.1 será a reação 1.

Referenciaremos aqui os 4 casos restantes (4, 5, 6 e 7) conforme critério abaixo:

- Caso 4 - apenas reação 1
- Caso 5 - apenas reação 2
- Caso 6 - apenas reação 3
- Caso 7 - apenas reação 4

5.4.1 Conclusões sobre testes com reações isoladas

Do apresentado a seguir veremos que o mecanismo cinético exhibe muitas oscilações em função das temperaturas e das concentrações consideradas. Como os gráficos apresentados estão em escala logarítmica vemos que a ordem de grandeza dos fenômenos são realmente muito diferentes, dificultando a convergência do método numérico. As reações 3 e 4, são bastante rápidas oscilando muito para determinadas temperaturas conforme podemos ver nas figuras 5.21 e 5.24. Conter numericamente o desenvolvimento destas reações pode constituir um problema numérico bastante difícil.

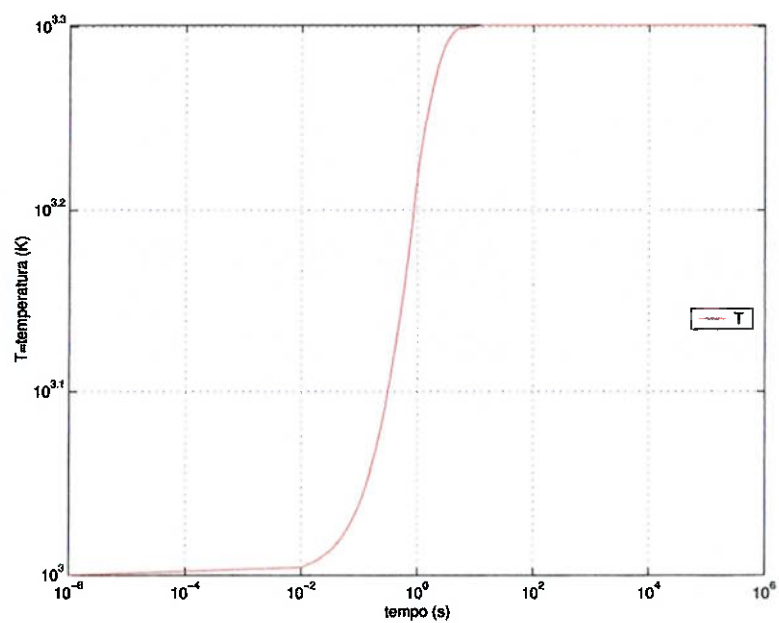


Figura 5.13: Temperatura - 4

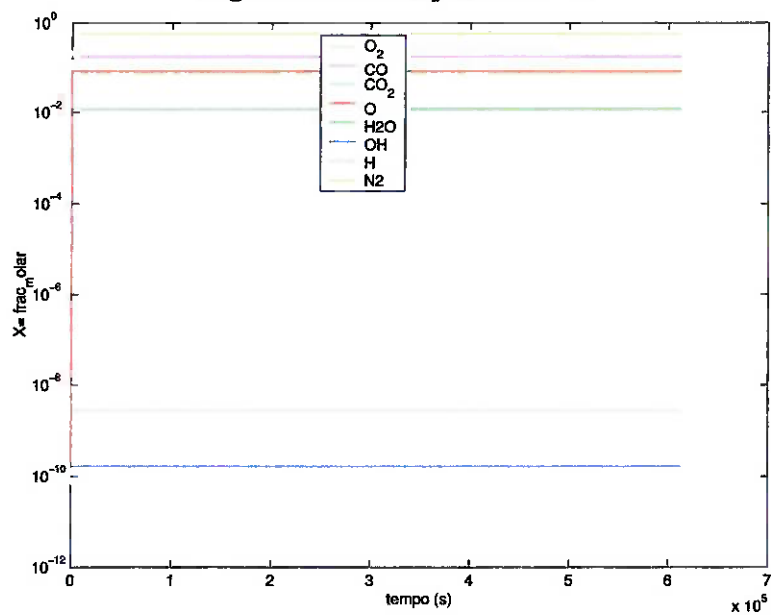


Figura 5.14: Frações Molares - 4

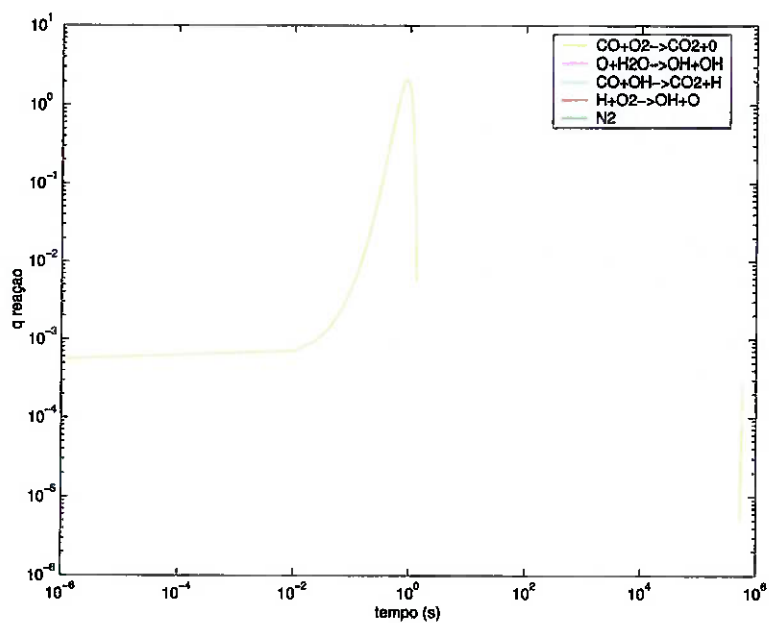


Figura 5.15: Taxas de reação - 4

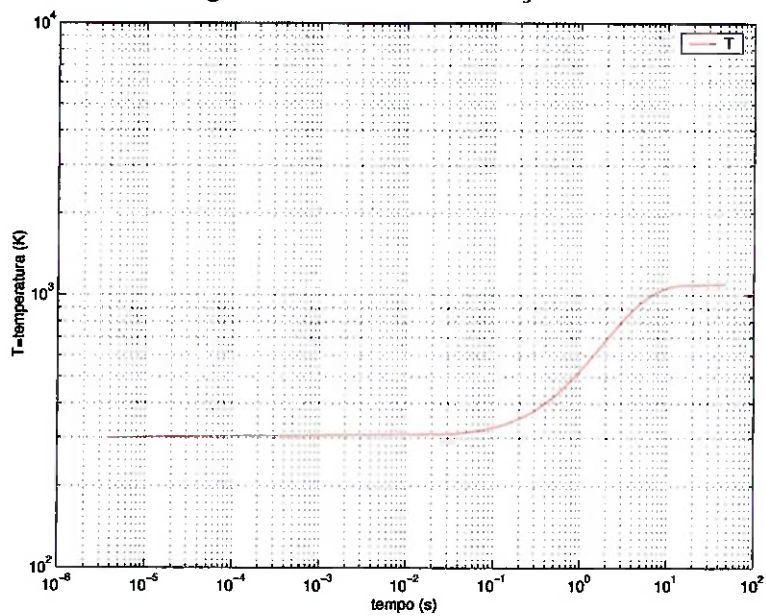


Figura 5.16: Temperatura - 5

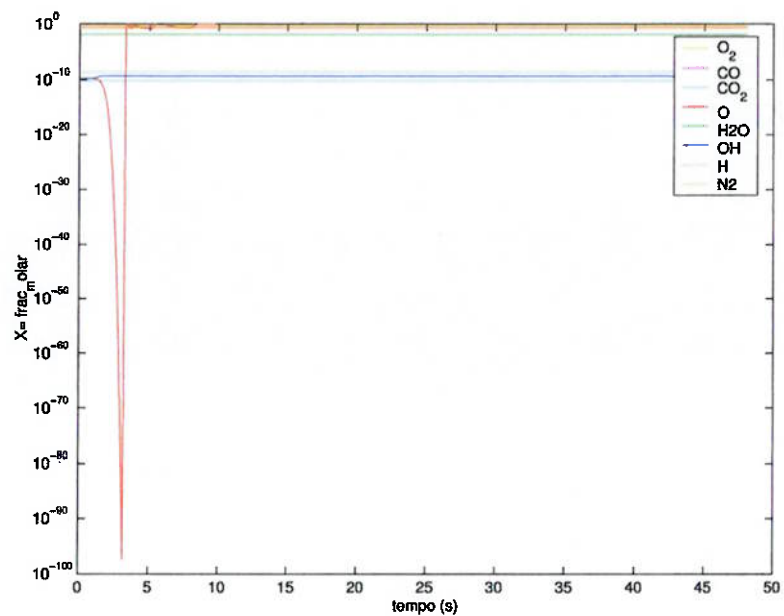


Figura 5.17: Frações Molares - 5

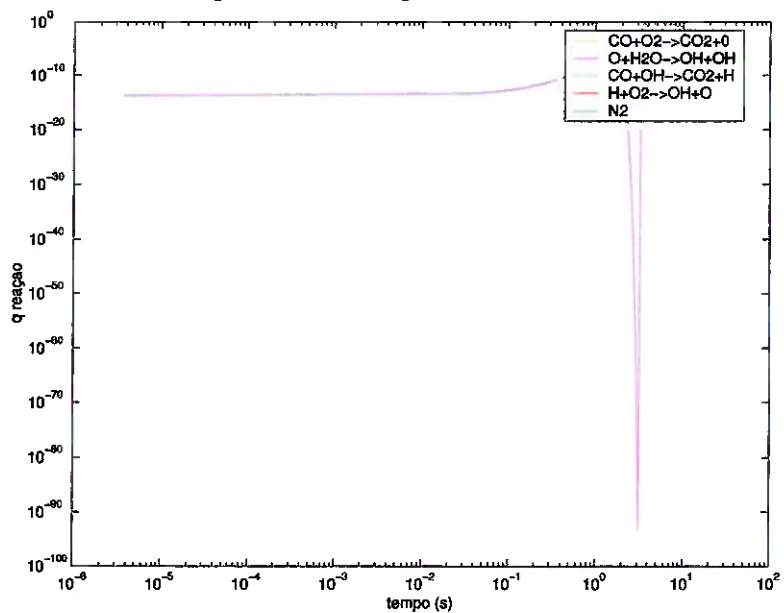


Figura 5.18: Taxas de reação - 5

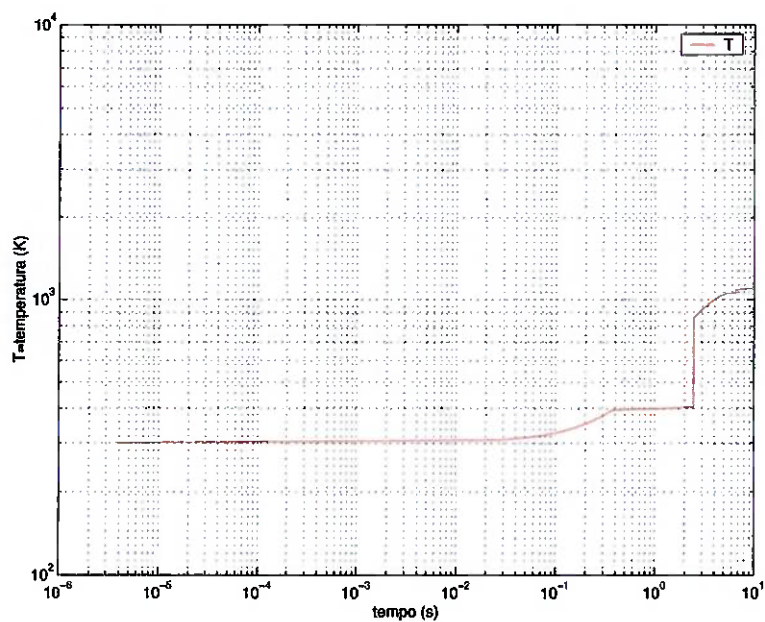


Figura 5.19: Temperatura - 6

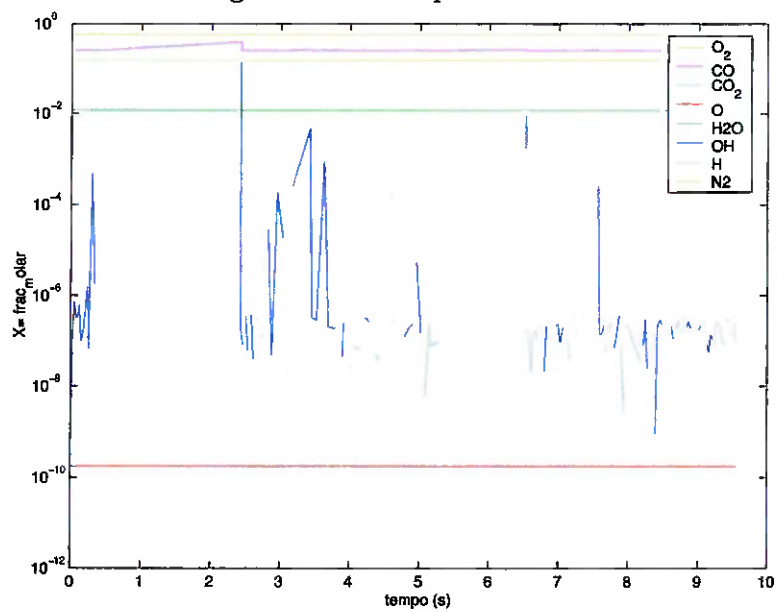


Figura 5.20: Frações Molares - 6

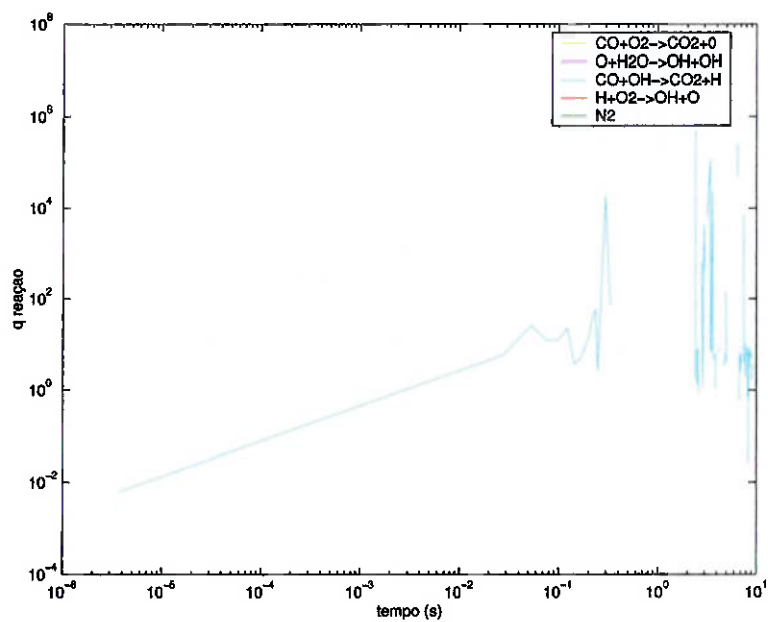


Figura 5.21: Taxas de reação - 6

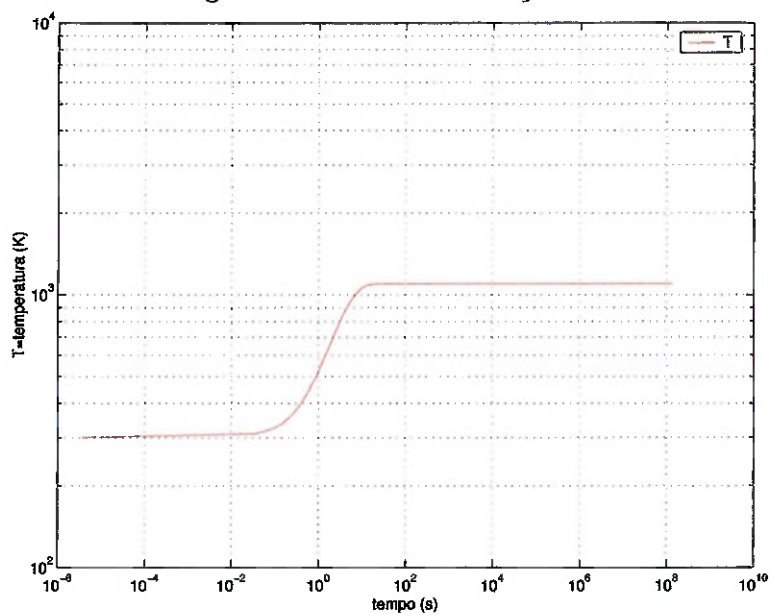


Figura 5.22: Temperatura - 7

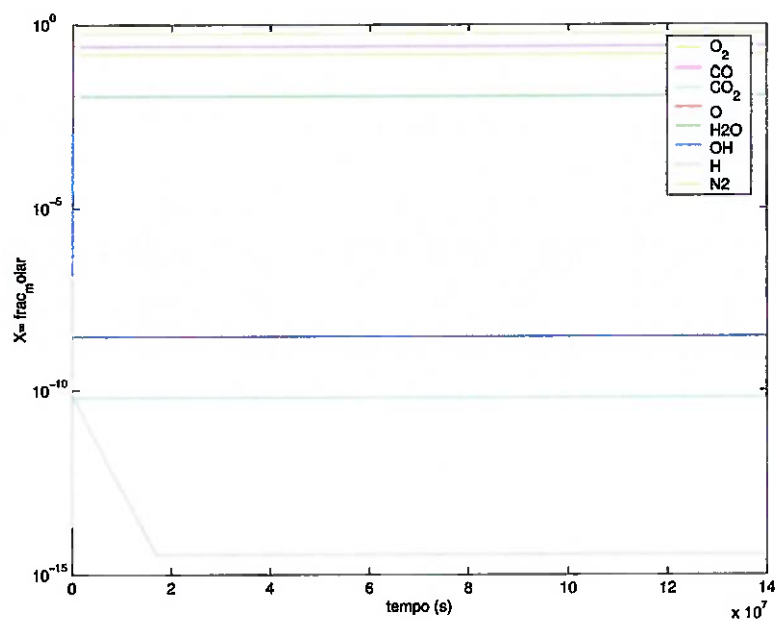


Figura 5.23: Frações Molares - 7

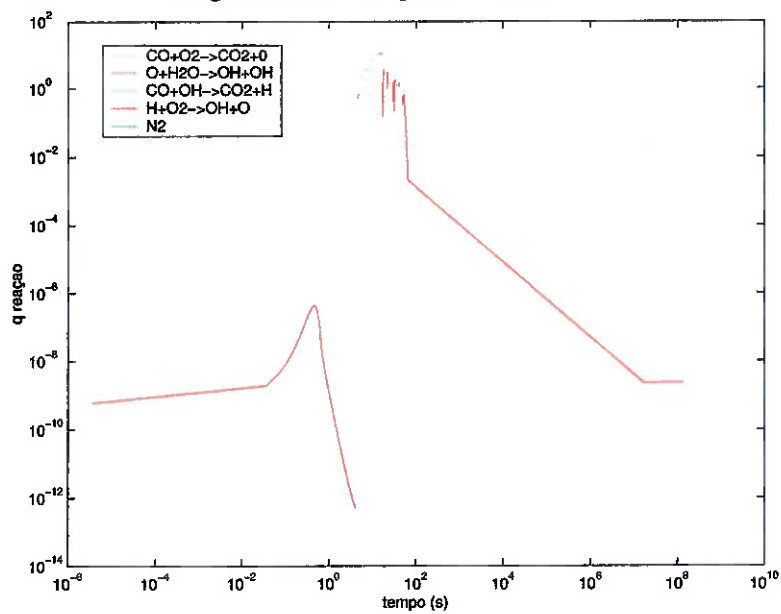


Figura 5.24: Taxas de reação - 7

Capítulo 6

Conclusões Gerais

Do apresentado até agora pudemos perceber que o estudo de escoamentos reativos constitui efetivamente um campo de estudos que ao mesmo tempo que exige um conhecimento multidisciplinar muito extenso, possibilita inúmeras aplicações.

A utilização de simulações numéricas nesta área é um campo muito já é um recurso consagrado e uma ferramenta imprescindível hoje para o engenheiro que atua nesta área. Problemas diversos envolvendo, envolvendo inclusive aspectos ambientais são alvo deste ramo das ciências térmicas.

No problema estudado aqui tivemos uma pequena amostra dos grandes desafios que constituem o estudo de escoamentos reativos.

O primeiro ponto a ser destacado é que o engenheiro deve manter um olhar crítico com relação às suas bases de dados e sempre que possível providenciar múltiplas fontes que permitam o confrontamento destes dados, principalmente no tocante a dados experimentais. Podemos citar como elementos que merecem atenção especial:

- **Valores interpolados de constantes de reação** - para os quais devemos atentar para as faixas de temperatura a que se aplicam. Nas referências bibliográficas utilizadas - [3],[7] e [2], principalmente - assim como no banco de dados do NIST - National Institute of Standards and Technology, encontramos diversos valores para estas constantes, provenientes de pesquisadores e experimentos diferentes. Estes valores conforme visto no capítulo 2.1, divergem até mesmo em ordens de grandeza de um autor para outro.
- **Coefficientes para polinômios interpoladores** - a utilização de uma fonte de dados com respaldo no meio científico é o primeiro passo. Tabelas

JANAF e NIST por exemplo. Atenção especial deve ser dada para o problema da mudança de faixas de temperatura de aplicabilidade do polinômio, onde os coeficientes mudam e pode não haver uma interpolação contínua da curva.

Também característicos de sistemas reativos são os sistemas de equações diferenciais "stiff", para os quais temos escalas de tempo diferentes. Este é um elemento bastante crítico do ponto de vista da solução numérica, pois limita o passo de integração à ordem de grandeza dos fenômenos de escala menor. algoritmos especiais para este tipo de problema podem ser encontrados na literatura sobre métodos numéricos.

O algoritmo da maneira como foi implementado apresenta diversas vantagens principalmente no tocante à versatilidade. Ele pode ser facilmente modificado para solução de qualquer outro mecanismo cinético, bastando apenas a inclusão de novos dados modificando apenas:

- polinômios interpoladores
- as matrizes dos coeficientes estequiométricos

Essa modularização do algoritmo é um recurso interessante pois facilitaria a inclusão de outros modelos básicos de reatores (bem-misturado, pressão constante-massa constante e volume constante-massa constante), bastando incluir subrotinas equivalentes à subrotina "Plug", que pode ser vista no código fonte anexo. Contemplando estes diversos tipos de reatores modelagens mais complexas são também possibilitadas, nas quais podemos utilizar combinações entre estes modelos básicos.

Outra vantagem da modularização é que em etapas futuras uma interface gráfica utilizando alguma biblioteca visual de programação poderia ser facilmente implementada, pois temos um programa que poderia se comunicar sem dificuldades com arquivos textos que conteriam tanto os parâmetros de entrada, como também poderiam conter as saídas do programa para posterior processamento.

É importante destacar aqui também que o algoritmo implementado tem um caráter genérico também no que diz respeito à forma do nosso reator. Desde que tenhamos uma simetria axial podemos descrever esta geometria através da sua variação ao longo do comprimento dA/dx .

Referências Bibliográficas

- [1] Van Wylen, G. J., Borgnakke, C., Sonntag, R. E., “*Fundamentos da Termodinâmica Clássica*”, Edgard Blücher LTDA., 5ª edição, (1998).
- [2] Turns, Stephen R., “*An Introduction to Combustion*”, McGraw-Hill, Inc., 1ª edição, (1996).
- [3] Gardiner Jr., W.C., “*Combustion Chemistry*”, Springer Verlag, N. York, (1984).
- [4] Incropera, F.P., De Witt, D.P., “*Fundamentals of Heat and Mass Transfer*”, John Wiley and Sons, N. York, 4ª edição, (1996).
- [5] Carnahan, B., Luther, H.A., Wilkes, J.O., “*Ordinary Differential Equations*”, John Wiley and Sons, N. York, 1ª edição, (1969).
- [6] Glassman, I., “*Combustion*”, Academic Press, Inc. N. York, 2ª edição (1987).
- [7] Williams, F.A., “*Combustion Theory*”, The Benjamin/Cummings Publishing Company, Inc. Menlo Park, 2ª edição (1987).

Anexo 1 - Código fonte comentado

```
c-----
c
c Código fonte comentado do programa implementado no trabalho de
c formatura de Luiz Tobaldini Neto, entregue à Escola Politécnica
c em dezembro de 2002
c-----
c 1
c subrotina para transpor matriz a em bc
      subroutine trans(a,b,lin,col)
      integer lin,col
      real*8 a(lin,col), b(col,lin)

      do 11 i=1,lin
        do 10 j=1,col
          b(i,j) =a(j,i)
10      continue
11      continue
      return
      end
c-----
c 2
c subrotina para preencher vetor x com zerosc
      subroutine zvec(x,tam)
      integer tam
      real*8 x(tam)

      do 10 i=1,tam
        x(i) =0.
10      continue
      return
      end
c-----
c 3
c subrotina para preencher matriz a com zerosc
      subroutine zmat(a,lin,col)
      integer lin,col
      real*8 a(lin,col)

      do 11 i=1,lin
        do 10 j=1,col
          a(i,j) =0.
10      continue
11      continue
```

```
        return
    end
c-----
c 4
c subrotina para multiplicar elementos de vetores um a umc
    subroutine mvec(w,x,y,tam)
        integer tam
        real*8 x(*),y(*),w(*)
        do 10 i=1,tam
            w(i)=x(i)*y(i)
        10 continue
        return
    end
c-----
c 5
c subrotina para somar elementos de um vetorc
    subroutine totalvec(ytotal,y,tam)
        integer tam
        real*8 y(*),ytotal

        ytotal=0.
        do 10 i=1,tam
            ytotal=ytotal+y(i)
        10 continue
        return
    end
c-----
c 6
c subrotina para multiplicar uma matriz por uma constante realc
    subroutine rxmat(c,a,lin,col,num)
        integer lin,col
        real*8 c(lin,col),a(lin,col)

        real num

        do 11 i=1,lin
            do 10 j=1,col
                c(i,j)=num*a(i,j)
            10 continue
        11 continue
        return
    end
c-----
c 7
c subrotina para multiplicar uma matriz por uma constante realc
    subroutine mmat(c,a,b,lina,cola,linb,colb)
        integer lina,cola,linb,colb
        real*8 a(lina,cola),b(linb,colb),c(lina,colb)

        call zmat(c,lina,colb)
        do 12 i=1,lina
            do 11 j=1,colb
                do 10 k=j,cola
                    c(i,j)=c(i,j)+a(i,k)*b(k,j)
                10 continue
            11 continue
        12 continue
        return
    end
c-----
c 8
c subrotina para somar duas matrizes
    subroutine smat(c,a,b,lin,col)
        integer lin,col
```

```

      real*8 a(lin,col),b(lin,col),c(lin,col)

      do 11 i=1,lin
        do 10 j=1,col
          c(i,j)=a(i,j)+b(i,j)
10      continue
11      continue
      return
      end

c-----
c 9
c subrotina para cálculo das funções a serem integradas na modelagem
c de um escoamento Plug-Flow conforme visto na descrição teórica
c
c Essa rotina recebe como parâmetros:
c
c R=constante universal dos gases
c der=vetor que recebe as derivadas aqui calculadas
c ncomp=número de componentes
c MW=massa molar
c omdot=derivada das concentrações de cada componente calculada na
c   subrotina "dconc"
c h=entalpia absoluta na base mássica de cada componente
c cpmix = calor específico da mistura
c mwmix = massa molar da mistura
c area = área da seção transversal
c vx = velocidade do escoamento
c T= temperatura da mistura gasosa na seção
c p = pressão (atmosférica)
c rho= densidade da mistura
c uf=fluxo de calor por unidade comprimento de tubo na seção

      subroutine plug(R,der,ncomp,MW,omdot,h,cpmix,mwmix,area,
& vx,T,p,rho,uf)

      integer ncomp
      real*8 t1,t2,t3,t4,t5,t6,der(12)
      real*8 MW(ncomp),omdot(ncomp),h(ncomp)
      real*8 cpmix,mwmix,area,vx,T,p,rho,R,uf

C se for cilíndrico:
      der(1)=0.
      *****
      t1=(1-(R/(cpmix*mwmix))) * (rho**2)*(vx**2)*(1/area)*(der(1))

      t2=0
      do 10 i=1,ncomp
        t2 = t2 + ( MW(i)*omdot(i)*(h(i)-(mwmix/MW(i))*cpmix*T))
10      continue

      t3=(rho*R/(vx*cpmix*mwmix))*(t2)
      t4=p*(1+(vx**2)/(cpmix*T))-rho*(vx**2)
      t5=(uf*p*rho)/(T*cpmix*vx)

      der(2)=(t1+t3+t6)/t4
      *****

      t5=0
      do 20 i=1,ncomp
        t5=t5+h(i)*omdot(i)*MW(i)
20      continue

      der(3)=((vx**2)/(rho*cpmix))*der(2)+((vx**2)/cpmix)*

```

```

      & ((1/area)*der(1))-(1/(vx*rho*cpmix))*(+t5)+(uf/cpmix)
*****
      do 30 i=1,ncomp
        der(i+3)=omdot(i)*MW(i)/(rho*vx)
      30 continue

*****
      der(12)=1./vx

      return
      end
c-----
c 10
c subrotina para calculo da integração pelo método de Euler
      subroutine euler(var,der,tam,passo)
        integer tam
        real*8 var(tam)
        real*8 passo,var_nova(tam),der(tam)

        do 11 i=1,tam
          var(i) = var(i) + passo*der(i)
        11 continue

        return
        end
c-----
c 11
c subrotina para calculo da integração pelo método de Runge-Kutta 4
c os parâmetros k1,k2,k3,k4 são calculados no main

      subroutine rk4(k1,k2,k3,k4,var,tam)

        integer tam
        real*8 k1(tam),k2(tam),k3(tam),k4(tam),var(tam)

        do 11 i=1,tam
          var(i)=var(i)+k1(i)/6+k2(i)/3+k3(i)/3+k4(i)/6
        11 continue

        end
c-----
c 12
c subrotina auxiliar para calculo dos k's do método de Runge-Kutta 4
      subroutine k(kj,f,tam,passo)

        integer tam
        real*8 var(tam)
        real*8 passo,f(tam),kj(tam)

        do 11 i=1,tam
          kj(i) = passo*f(i)
        11 continue

        return
        end
c-----
c 13
c subrotina para cálculo de omdot=variação das concentrações de cada
c componente
c
c Os parâmetros utilizados nessa subrotina são
c
c omdot=vetor que recebe os valores calculados
c conc=concentração volumétrica molar de cada componente
c nreaction = número de reações do modelo

```

```

c ncomp = número de componentes envolvidos
c vreacts = coeficientes estequiométricos de cada componente, em
c          cada reação (lado dos reagentes)
c vprods = coeficientes estequiométricos de cada componente, em
c          cada reação (lado dos produtos)
c kf = vetor com k's "pra frente" de cada reação
c kr = vetor com k's "pra trás" de cada reação
c q = vetor com saldo de produção de cada componente conforme
c notação apresentada na teoria

      subroutine dconc(omdot,conc,nreaction,ncomp,vreacts,
& vprods,kf,kr,q)

      integer comp,reaction,nreaction,ncomp

      real*8 conc(ncomp),q(nreaction)
      real*8 prodtf(nreaction),prodtr(nreaction)
      real*8 vreacts(ncomp,nreaction),vprods(ncomp,nreaction)
      real*8 kf(nreaction),kr(nreaction)
      real*8 nvreacts(ncomp,nreaction),v(ncomp,nreaction)
      real*8 omdot(ncomp)

      do 10 reaction=1,nreaction
        prodtf(reaction)=1.
        prodtr(reaction)=1.
      10 continue

      do 12 reaction=1,nreaction

        do 11 comp=1,ncomp
          prodtf(reaction)=prodtf(reaction)*
& (conc(comp)**vreacts(comp,reaction))
          prodtr(reaction)=prodtr(reaction)*
& (conc(comp)**vprods(comp,reaction))
        11 continue

        q(reaction)=kf(reaction)*prodtf(reaction)-
& kr(reaction)*prodtr(reaction)
      12 continue

      call rxmat(nvreacts,vreacts,ncomp,nreaction,-1.)
      call smat(v,vprods,nvreacts,ncomp,nreaction)

      call zvec(omdot,ncomp)

      do 14 comp=1,ncomp
        do 13 reaction=1,nreaction
          omdot(comp)= omdot(comp)+v(comp,reaction)*q(reaction)
        13 continue
      14 continue

      return
      end

-----
c 14
c subrotina para cálculo das propriedades termodinâmicas h,cp e s
c na base molar para cada componente a determinada temperatura T
c de acordo com os polinômios de interpolação fornecidos pelo
c NIST - National Institute of Standards and Technology

```

```

subroutine pprts(a,ncomp,h0bar,cpbar,g0bar,s0bar,T)
integer i,ncomp
real*8 h0bar(ncomp),cpbar(ncomp),s0bar(ncomp),g0bar(ncomp)
real*8 T
real te
real*8 a(9,ncomp)

te=T/1000.

do 10 i=1,ncomp

  h0bar(i)= 1000.*(a(1,i)*te + a(2,i)*(te**2)/2 + a(3,i)*(te**3)/3
& + a(4,i)*(te**4)/4 - a(5,i)/te + a(6,i) )

  cpbar(i)= a(1,i) + a(2,i)*te + a(3,i)*(te**2) + a(4,i)*(te**3)
& + a(5,i)/(te**2)

  s0bar(i)= a(1,i)*alog(te) + a(2,i)*(te) + a(3,i)*(te**2)/2
& + a(4,i)*(te**3)/3 - a(5,i)/(2*(te**2)) + a(7,i)

10 continue

return
end

c-----
c 15
c subrotina para cálculo da função de Gibbs e cálculo dos k's "de volta"
c das reações: Não foi Utilizada

C subroutine gibbs(reaction,ncomp,h0bar,cpbar,g0barf,s0bar,T)
C integer i,j,ncomp
C real*8 h0bar(ncomp),cpbar(ncomp),s0bar(ncomp),g0barf(ncomp),
c real*8 hobarf(ncomp)
C real*8 T,t1,t2
C do 11 i=1,ncomp
C   t1=0
C   t2=0
C   do 10 j=1,ncomp
C     t1 = t1 + vreact(j,reaction)*h0bar(i)
C     t2 = t2 + (-T*s0bar(j))
C 10 continue
C   h0barf(i)=h0bar(i)-t1
C   g0barf(i) = h0barf(i)-T*s0bar(i)-t2
C 11 continue
C   return
C   end

*****
*****
*****

c 16
c PROGRAMA PRINCIPAL
  program main

C PARÂMETROS
c ncomp= número de componentes
c nreaction = número de reações

```

```

c cont1 = contador para limite de iterações
c lim1 = limite máximo de iterações
c printinter = intervalo de impressão (tela e texto)

    integer ncomp, nreaction
    integer cont1,lim1,cont2,lim2,printinter

c pi = constante pi
c R = constante universal dos gases (J.mol.K)
c Avog = número de Avogrado

    real*8 pi
    real*8 R
    real*8 Avog

    parameter (ncomp = 8)
    parameter (nreaction = 5)
    parameter (pi = 3.14159)
    parameter (Avog = 6.02e23)
    parameter (R=8.3145)
    parameter (lim1= 3e6)

*****
C VARIÁVEIS
c flags
    integer coflg,co2flg,h2oflg,ohflg,backflg,parada

c diam = diametro da tubulacao
c a = matriz com 9 termos do polinômio da NIST e com ncomp linhas
c kf, kr = vetores com k's de ida e volta das reações
c var = vetores que guarda variáveis integradas
c der = derivadas das variáveis
c varold = guarda var do passo anterior para restauracao
c k1,k2,k3,k4 = parâmetros do método RK-4
c passo = passo de integracao
    real*8 diam
    real*8 a(9,ncomp),kf(nreaction),kr(nreaction),var(12),der(12)
    real*8 dercor(12),k1(12),k2(12),k3(12),k4(12),varold(12)
    real*8 passo

c MW = massa molar de cada componente
c omdot = variacao da concentracao de cada componente
c conc = concentracao de cada componente
c v = volume específico da mistura
c mwmix = massa molar da mistura
c Y = fracao massica de cada componente
c X = fracao molar de cada componente
    real*8 MW(ncomp),omdot(ncomp),conc(ncomp)
    real*8 v,mwmix
    real*8 Y(ncomp), X(ncomp)

c h0bar = entalpia absoluta na base molar de cada componente
c g0bar = valor da funcao de Gibbs para cada componente
c s0bar = entropia
c hmass = entalpia absoluta na base mássica para cada componente
c cpbar = calor específico de cada componente na base molar
c cpmix = calor específico da mistura na base mássica
    real*8 h0bar(ncomp),g0bar(ncomp),s0bar(ncomp),hmass(ncomp)
    real*8 cpbar(ncomp),cpmix

c vx = velocidade do escoamento
c mponto = vazao massica
c t = temperatura

```

```

c p= pressao (atmosferica)
c rho = massa especifica
c comprimento = distancia percorrida na tubulacao
c tempo = tempo total da simulacao
c ytotal = somatoria dos elementos do vetor Y para conferencia
c xtotal = somatoria dos elementos do vetor X para conferencia
  real*8 vx
  real*8 mponto
  real*8 t,p,rho
  real*8 comprimento,tempo,ytotal,xtotal,t2

c area = area da secao transversal
c excar = excesso de ar na entrada
c mtotal = massa total para 1 mol de combustivel na entrada
c mh2o = massa de agua na entrada
c omega = unidade absoluta do ar na entrada
  real*8 area,excar,mtotal,mh2o,omega

c vreacts, vprods= coeficientes estequiométricos de cada componente
c   conforme notacao apresentada na teoria
c ur = faotr de sub-relaxacao
c q = saldo da reação (para esquerda ou direita)
  real*8 vreacts(8,5),vprods(8,5),ur,q(nreaction)

c vetores com coeficientes dos polinomios NIST para cada componente
c   vetores com indice B sao para segunda faixa de temperaturas
c   ajustadas
  real*8 co(9),coB(9),o2(9),co2(9),co2B(9)
  real*8 o(9),h2o(9),h2oB(9),oh(9),ohB(9),h(9),n2(9)

c unit_flux= fluxo de calor por unidade de comprimento na entrada
c hconv = coeficiente de transferencia de calor por conveccao
c twall = temperatura da parede na entrada - para iginicao
c perim = perimetro da tubulacao
c flux = fluxo de calor da parede pro fluido
  real*8 unit_flux,hconv,twall,perim,flux
  real*8 q1,q2,q3,q4

*****
C INICIALIZACOES

  p=1.013e5
  mponto=0.1
  tempo=0.
  comprimento=0.
  T=300.
  twall=1100.
  diam=0.1
  perim=pi*diam
  ur=5
  hconv=100.

  passo = 1.e-4
  printinter=1.e4

  coflg=0
  co2flg=0
  h2oflg=0
  ohflg=0

  ytotal=0.
  xtotal=0.

  data vreacts/1.,1.,0.,0.,0.,0.,0.,0.,
&              0.,0.,0.,1.,1.,0.,0.,0.,

```

```

&          1.,0.,0.,0.,0.,1.,0.,0.,
&          0.,1.,0.,0.,0.,0.,1.,0.,
&          0.,0.,0.,0.,0.,0.,0.,1./

```

```

data vprods/0.,0.,1.,1.,0.,0.,0.,0.,
&          0.,0.,0.,0.,0.,2.,0.,0.,
&          0.,0.,1.,0.,0.,0.,1.,0.,
&          0.,0.,0.,1.,0.,1.,0.,0.,
&          0.,0.,0.,0.,0.,0.,0.,1./

```

```

data co/25.56759, 6.096130, 4.054656, -2.671301, 0.131021,
& -118.0089, 227.3665, -110.5271, 28.01/

```

```

data coB/35.15070, 1.300095, -0.205921, 0.013550, -3.282780,
& -127.8375, 231.7120, -110.5271, 28.01/

```

```

data o2/29.6590, 6.137261, -1.186521, 0.095780, -0.219663,
& -9.861391, 237.9480, 0., 32.00/

```

```

data co2/24.99735, 55.18696, -33.69137, 7.948387, -0.136638,
& -403.6075, 228.2431, -393.5224, 44.01/

```

```

data co2B/58.16639, 2.720074, -0.492289, 0.038844, -6.447293,
& -425.9186, 263.6125, -393.5224, 44.01/

```

```

data o/21.18610, -0.502314, 0.168694, -0.008962, 0.075664,
& 243.1306, 187.2591, 249.1731, 16.00/

```

```

data h2o/30.09200, 6.832514, 6.793435, -2.534480, 0.082139,
& -250.8810, 223.3967, -241.8264, 18.02/

```

```

data h2oB/41.96426, 8.622053, -1.499780, 0.098119, -11.15764,
& -272.1797, 219.7809, -241.8264, 18.02/

```

```

data oh/32.27768, -11.136291, 13.60545, -3.846486, -0.001335,
& 29.75113, 225.5783, 38.98706, 17.01/

```

```

data ohB/28.74701, 4.714489, -0.814725, 0.054748, -2.747829,
& 26.41439, 214.1166, 38.98706, 17.01/

```

```

data h/20.78603, 4.850638e-10, -1.582916e-10, 1.525102e-11,
& 3.196347e-11, 211.8020, 139.8711, 217.9994, 1.01/

```

```

data n2/26.09200, 8.2118801, -1.976141, 0.159274, 0.044434,
& -7.989230, 221.0200, 0, 28.01/

```

```

do 10 i=1,9

```

```

a(i,1)=co(i)
a(i,2)=o2(i)
a(i,3)=co2(i)
a(i,4)=o(i)
a(i,5)=h2o(i)
a(i,6)=oh(i)
a(i,7)=h(i)
a(i,8)=n2(i)

```

```

10 continue

```

```

do 11 i=1,ncomp

```

```

MW(i) = a(9,i)*0.001

```

```

11 continue

```

```

*****
c definicao dos índices de 1 a 9 correspondentes a cada

```

```

c componente
c 1=CO
c 2=O2
c 3=CO2
c 4=O
c 5=H2O
c 6=OH
c 7=H
c 8=N2
*****

c CALCULO DAS FRACOES MASSICAS
c fazendo-se a reação de 1mol de CO:

  omega=0.005
  excar= 1.

  mttotal = 28. + excar*(16.+52.6)
  mh2o= omega*excar*(16.+52.6)
  mttotal=mh2o+mttotal

c caso seja queima com oxigenio puro
c  mttotal = 28. + excar*(16.)
c  mh2o= omega*excar*(16.)
c  mttotal=mh2o+mttotal

  Y(1)=28./mttotal
  Y(2)=excar*16/mttotal
  Y(3)=0.
  Y(4)=0.
  Y(5)=(mh2o/mttotal)
  Y(6)=0.
  Y(7)=0.
  Y(8)=excar*52.6/mttotal
  call totalvec(ytotal,y,ncomp)
*****

  area =(pi*diam*diam)/4

  call zvec(omdot,ncomp)
  call zvec(der,ncomp)
  call zvec(q,nreact)

  mwmix=0.
  do 12 i=1,ncomp
    mwmix = mwmix + (Y(i)/MW(i))
12  continue
  mwmix = 1/mwmix

  do 13 i=1,ncomp
    X(i)=Y(i)*mwmix/MW(i)
13  continue

  v=(R*T)/(p*mwmix)
  rho=1./v

  backflg=0
  cont1=0

  der(3)=3
  parada=0

*****
C ITERAÇÕES até limite lim1

```

```

do while ((cont1.lt.lim1))
    q1=q(1)
    q2=q(2)
    q3=q(3)
    q4=q(4)

c Cálculo da massa molar da mistura para novo Y calculado na
c iteracao anterior
    mwmix=0.
    do 20 i=1,ncomp
        mwmix = mwmix + (Y(i)/MW(i))
    20 continue
    mwmix = 1./mwmix

c Atualizacao de X,v,vx,diam (caso haja variacao de area) e perim
    do 21 i=1,ncomp
        X(i)=Y(i)*mwmix/MW(i)
    21 continue

    v=1./rho
    vx=mponto/(rho*area)
    diam=(area*4/pi)**0.5
    perim=pi*diam

c Correcao dos coeficientes dos polinomios NIST em funcao do
c range de temperatura

    if ((T.GE.1300.).AND.(coflg.EQ.0).AND.(ohflg.EQ.0)) then
        do 22 i=1,ncomp
            a(i,1)=coB(i)
            a(i,6)=ohB(i)
            coflg=1
            ohflg=1
        22 continue
    end if

    if ((T.GE.1200.).AND.(co2flg.EQ.0))then
        do 23 i=1,ncomp
            a(i,3)=co2B(i)
            co2flg=1
        23 continue
    end if

    if ((T.GE.1200.).AND.(h2oflg.EQ.0)) then
        do 24 i=1,ncomp
            a(i,5)=h2oB(i)
            h2oflg=1
        24 continue
    end if

c Calculo das propriedades termodinâmicas de subrotina "pprts"
    call pprts(a,ncomp,h0bar,cpbar,g0bar,s0bar,T)

    cpmix=0.
    do 25 i=1,ncomp
        cpmix=cpmix+(X(i)*cpbar(i))
    25 continue

c Conversao de cp e h para base massica
    cpmix=cpmix/mwmix

    do 30, i=1,ncomp

```

```

      hmass(i)=h0bar(i)/MW(i)
30  continue

c Definicao dos valores dos k's de ida e volta para reacoes de
c 1 ate 5 conforme definido no corpo do trabalho

      kf(1)=2.53e6*exp(-199547/(R*T))
      kf(2)=1.11e7*((T/298.)**0.95)*exp(-71255./(R*T))
      kf(3)=2.26e4*((T/298.)**1.55)*exp(3342./(R*T))
      kf(4)=9.75e7*exp(-62109./(R*T))
      kf(5)=0

      kr(1)=1.69e7*exp(-220334/(R*T))
      kr(2)=9.95e5*((T/298.)**1.14)*exp(-416./(R*T))
      kr(3)=1.51e8*exp(-110583./(R*T))
      kr(4)=2.74e6*((T/298.)**0.40)*exp(3093./(R*T))
      kr(5)=0

c k's alternativos
c      kf(1)=2.53e6*exp(-199547./(R*T))
c      kf(2)=1.11e7*((T/298.)**0.95)*exp(-71255./(R*T))
c      kf(3)=4.0e6*exp(-33000/(R*T))
c      kf(4)=9.75e7*exp(-62109./(R*T))
c      kf(5)=0.

c      kr(1)=1.69e7*exp(-220334./(R*T))
c      kr(2)=3.1e1*((T/298.)**4.77)*exp(37997./(R*T))
c      kr(3)=1.51e8*exp(-110583./(R*T))
c      kr(4)=6.92e8*((T/298.)**(-0.92))*exp(-69343./(R*T))
c      kr(5)=0.

c k's zerados para desconsiderar reacao "i"
c      kf(1)=0.
c      kf(2)=0.
c      kf(3)=0.
c      kf(4)=0.
c      kf(5)=0

c      kr(1)=0.
c      kr(2)=0.
c      kr(3)=0.
c      kr(4)=0.
c      kr(5)=0.

c Calculo da concentracao molar de cada componente
  do 31 i=1,ncomp
    conc(i)=Y(i)*rho/MW(i)
31  continue

c Chamada da funcao "dconc" para calculo da derivada da
c concentracao

      call dconc(omdot,conc,nreaction,ncomp,vreacts,
&      vprods,kf,kr,q)

c Calculo do calor transferido por conveccao da parede pro
c escoamento
      if(twall.gt.t) then

        flux=(twall-t)*hconv
        unit_flux=flux*perim

```

```

    else
        unit_flux=0.
    end if

c chamada da funcao para calculo de derivadas a serem
c integradas
    call plug(R,der,ncomp,MW,omdot,hmass,
        &          cpmix,mwmix,area,vx,T,p,rho,unit_flux)

c vetor var recebe variaveis correspondentes a serem integradas
c ela existe apenas por conveniencia de notacao e para facilidade
c de passagem às funcoes
    var(1)=area
    var(2)=rho
    var(3)=T
    var(4)=Y(1)
    var(5)=Y(2)
    var(6)=Y(3)
    var(7)=Y(4)
    var(8)=Y(5)
    var(9)=Y(6)
    var(10)=Y(7)
    var(11)=Y(8)
    var(12)=tempo

c atribuicao de valores a varold para restituicao
c usado caso calculo da variavel de valor muito alto
c apos integracao. Volta-se, altera-se o passo e
c continuamos com passo novo e varold

    do 32 i=1,ncomp
        varold(i)=var(i)
    32 continue

c Correcao do passo caso reacao (1 a 4) esteja ocorrendo muito
c rapidamente

    if ((dmax1(q1,q2,q3,q4).gt.300).and.(backflg.eq.0))then
        passo=1.e-3/abs(der(3))
        printinter= 1.e4

        var(1)=varold(1)
        var(2)=varold(2)
        var(3)=varold(3)
        var(4)=varold(4)
        var(5)=varold(5)
        var(6)=varold(6)
        var(7)=varold(7)
        var(8)=varold(8)
        var(9)=varold(9)
        var(10)=varold(10)
        var(11)=varold(11)
        var(12)=varold(12)

        backflg=1
    end if

    if (backflg.eq.1)then
        passo=1.e-3/abs(der(3))
    endif

```

```

c  if (der(3).gt.1e8)then
c
c      passo=(1.e-8)/der(3)
c      backflag=2
c
c  endif
c
c      if (dmax1(q1,q2,q3,q4).lt.5)then
c          passo=1.e-3/abs(der(3))
c      end if
c
c      call euler(var,der,12,passo)
c Comandos alternativo para uso de RK4
c
c  call plug(R,der,ncomp,MW,omdot,hmass,
c      &      cpmix,mwmix,area,vx,T,p,rho,unit_flux)
c      call k(k1,der,12,passo)
c
c  do 33 i=1,12
c      var(i) = var(i)+.5*k1(i)
c 33  continue
c
c      area = var(1)
c      rho = var(2)
c      T = var(3)
c
c      do 34 i=1,ncomp
c          Y(i) = var(i+3)
c 34  continue
c      tempo = var(12)
c
c      do 35 i=1,ncomp
c          conc(i)=Y(i)*rho/MW(i)
c 35  continue
c
c      call dconc(omdot,conc,nreaction,ncomp,vreacts,
c      &  vprods,kf,kr,q)
c
c      call plug(R,der,ncomp,MW,omdot,hmass,
c      &      cpmix,mwmix,area,vx,T,p,rho,unit_flux)
c      call k(k2,der,12,passo)
c *****
c      do 43 i=1,12
c          var(i) = var(i)+.5*k2(i)
c c 43  continue
c
c      area = var(1)
c      rho = var(2)
c      T = var(3)
c
c      do 44 i=1,ncomp
c          Y(i) = var(i+3)
c 44  continue
c      tempo = var(12)
c
c      do 45 i=1,ncomp
c          conc(i)=Y(i)*rho/MW(i)
c 45  continue
c
c      call dconc(omdot,conc,nreaction,ncomp,vreacts,
c      &  vprods,kf,kr,q)
c
c      call plug(R,der,ncomp,MW,omdot,hmass,

```

```

c      &          cpmix,mwmix,area,vx,T,p,rho,unit_flux)
c
c      call k(k3,der,12,passo)
c*****
c      do 53 i=1,12
c          var(i) = var(i)+.5*k3(i)
c 53  continue
c
c      area = var(1)
c      rho = var(2)
c      T = var(3)
c
c      do 54 i=1,ncomp
c          Y(i) = var(i+3)
c 54  continue
c      tempo = var(12)
c
c      do 55 i=1,ncomp
c          conc(i)=Y(i)*rho/MW(i)
c 55  continue
c*****
c      call dconc(omdot,conc,nreaction,ncomp,vreacts,
c      &  vprods,kf,kr,q)
c
c      call plug(R,der,ncomp,MW,omdot,hmass,
c      &          cpmix,mwmix,area,vx,T,p,rho,unit_flux)
c
c      call k(k4,der,12,passo)
c*****
c      call rk4(k1,k2,k3,k4,var,12)
c*****
c      area = (ur*var(1)+(1-ur)*area)
c      rho = (ur*var(2)+(1-ur)*rho)
c      T = (ur*var(3)+(1-ur)*T)
c
c      do 32 i=1,ncomp
c          Y(i) = (ur*var(i+3)+(1-ur)*Y(i))
c 32  continue
c      tempo = (ur*var(12)+(1-ur)*tempo)
c
c*****
C ATUALIZACAO DAS VARIAVEIS INTEGRADAS POR EULER
c      area = var(1)
c      rho = var(2)
c      T = var(3)
c
c      do 61 i=1,ncomp
c          Y(i) = var(i+3)
c 61  continue
c      tempo = var(12)
c
c      t2=tempo
c      comprimento=comprimento+passo
c*****
C IMPRESSOES DAS SAIDAS PARA POS-PROCESSAMENTO
c
c      if (mod((cont1),printinter).eq.0) then
c
c      open (3, file = 'saida106.txt')
c          write(3,69) t,tempo,comprimento,cpmix,mwmix,passo
c
c      open (4, file = 'saida206.txt')
c          write(4,69) x

```

```
open (5, file = 'saida306.txt')
write(5,69) q

open (6, file = 'saida406.txt')
write(6,69) h0bar

call totalvec(ytotal,y,ncomp)
call totalvec(xtotal,x,ncomp)

write(*,69) T,mwmix,tempo,comprimento,der(3)
write(*,*)cont1,passo,rho,vx
end if

c Atualiza contador e encerra loop (se necessario)
cont1=cont1+1

end do

c Verifica qual foi o criterio de parada (controle de erro)
if (cont1.ge.lim1) then
parada=1
end if

if (der(3).le.0.0001) then
parada=2
end if

69 format (1x, 8e13.5)

write (*,*) 'fim da simulação',parada

end
end
```
