

LIN CHAU JEN

MODELAGEM DO ESCOAMENTO E DA TROCA DE CALOR POR
RADIAÇÃO DE UM RECUPERADOR DE CALOR SIDERÚRGICO

TESE APRESENTADA À ESCOLA POLITÉCNICA DA
UNIVERSIDADE DE SÃO PAULO PARA A OBTENÇÃO
DO TÍTULO DE DOUTOR EM ENGENHARIA

ORIENTADOR: PROF. DR. CLEMENTE GRECO

PROFESSOR DOUTOR DO DEPARTAMENTO
DE ENGENHARIA MECÂNICA DA EPUSP

SÃO PAULO

1998

OK

Agradecimentos

Ao meu orientador, pelo interesse, incentivo e acompanhamento do meu programa de doutorado. Já no meu mestrado, mesmo em outro departamento, o apoio do Prof. Greco foi-me muito precioso.

A Francisco D. A. Souza, pelo interesse constante e auxílios valiosos. Seu apoio foi fundamental para a realização deste trabalho.

Ao Prof. Euryale J. Zerbini, pelas proveitosas conversas e sugestões.

Aos colegas do Agrupamento de Engenharia Térmica do IPT, pela possibilidade de ter-me dedicado a este trabalho.

À minha esposa, Leninha, e aos meus filhos, Lucas e Daniel, pela compreensão.

DEDALUS - Acervo - EPMN



31600010402

Modelagem do Escoamento e da Troca de Calor por Radiação de um Recuperador de Calor

Errata

<i>localização</i>	<i>expressão incorreta</i>	<i>expressão correta</i>
pg. 1, 16 ^a linha	difundido	bem difundido
pg. 2, 24 ^a linha	para poder-se utilizar	para se poder utilizar
pg. 4, 15 ^a linha	parâmetros de operação	parâmetros de projeto
pg. 4, 21 ^a linha	dados operacionais	dados de projeto
pg. 5, figura	2 ^a “saída do ar”	entrada do ar
pg. 8, 10 ^a linha	para utilizar-se	para se utilizar
pg. 9, rodapé	como pode-se perceber	como se pode perceber
pg. 13, 24 ^a linha	que é esse método	que esse método
pg. 13, rodapé	fortemente condicionada	fortemente condicionado
pg. 20, 6 ^a linha	$S =$	$S_{\phi} =$
pg. 22, 4 ^a linha	pefil achatado	perfil uniforme
pg. 23, 14 ^a /15 ^a linha	retirar “e à taxa de liberação de calor internamente ao volume de controle (Q)	
pg. 26, 21 ^a linha	temperatura do volume de controle	temperatura média da seção do fluido
pg. 27, 12 ^a linha	fluxo de potência	fluxo de energia
pg. 27, 14 ^a linha	fluxo de potência	fluxo de energia

pg. 28, 20 ^a linha	produto vetorial	produto escalar
pg. 34, 11 ^a linha	expressão (5.28)	expressão (5.32)
pg. 34, 14 ^a linha	expressão (5.28)	expressão (5.33)
pg. 42, figura	2 ^a “saída do ar”	entrada do ar
pg. 55, 2 ^a linha	q_c	$-q_c$
pg. 57, 7 ^a linha	entre 1 e 4	arbitrário. Utilizou-se valores entre 1 e 4.
pg. 61, 10 ^a linha	ao longo do reator	ao longo do trocador
pg. 65, figura	Fluxo de potência	Fluxo de energia
pg. 66, figura	Fluxo de potência	Fluxo de energia
pg. 67, figura	Fluxo de potência	Fluxo de energia
pg. 68, figura	Fluxo de potência	Fluxo de energia

Sumário

1. Introdução	1
2. Objetivo proposto e formulação do problema	4
2.1. Objetivo proposto	4
2.2. Formulação do problema do recuperador de calor	4
3. Revisão da literatura	8
4. Técnica “CFD”: equacionamento básico e método numérico	17
4.1. Preâmbulo	17
4.2. A variante do método empregada na técnica CFD	18
4.3. Forma geral das equações diferenciais de conservação	19
4.4. Turbulência	20
4.5. Comentários sobre a condição de contorno na seção de saída das variáveis do escoamento	21
5. A equação de energia e o método das ordenadas discretas	23
5.1. A equação de energia	23
5.2. Modelagem da troca térmica radiativa	29
5.2.1. A equação de transferência radiativa	29
5.2.2. O método das ordenadas discretas	31
5.2.3. Seleção do número e das direções das ordenadas discretas	40
5.3. Condições de contorno para a simulação do trocador de calor	41
5.3.1. Para a equação da intensidade de radiação	41
5.3.2. Para a equação de energia	47
6. Validação do método das ordenadas discretas	48
7. Resultados da simulação do recuperador de calor	54
7.1. Perfis de temperaturas no tubo que separa o gás do ar	54
7.2. Resultados	58
7.2.1. Campo de velocidade dos gases	58
7.2.2. Campo de temperatura dos gases	60
7.2.3. Temperaturas ao longo do trocador	61
7.2.4. Fluxos térmicos transferidos do gás para o ar	65

7.2.5. A importância da radiação como mecanismo de troca térmica dos gases	66
7.2.6. Fluxos de potência radiativos	67
8. Conclusões e considerações finais	69
Anexo A: Fórmula da derivada direcional em coordenadas cilíndricas	70
Anexo B: Detalhes dos programas	74
Referências bibliográficas	78
Apêndice: Listagem dos programas	

Índice de figuras

Figura 2.1. Desenho esquemático do trocador de calor.	5
Figura 5.1. Esquema de volume de controle mostrando o fluxo térmico na direção y .	24
Figura 5.2. Esquema de feixes de radiação incidente e emergente de uma superfície.	27
Figura 5.3. Desenho esquemático apresentando a área como o cálculo de integral.	31
Figura 5.4. Esquema do sistema de coordenadas e ângulos locais na geometria cilíndrica.	33
Figura 5.5. Esquema apresentando os contornos Sul e Oeste do domínio.	38
Figura 5.6. Desenho esquemático do trocador de calor.	42
Figura 5.7. Esquema apresentando as parcelas da intensidade de radiação emergente de uma superfície sólida	42
Figura 5.8. Desenho esquemático de dois feixes de radiação simétricos.	43
Figura 5.9. Desenho esquemático de um orifício de uma cavidade esférica.	44
Figura 5.10. Dimensões do plenum e do conduto de ar.	45
Figura 6.1. Radiação entre duas placas.	48
Figura 6.2. Comparação dos resultados: (a) SIEGEL; HOWELL; (b) S_4 .	52
Figura 7.1. Esquema de um elemento da parede apresentando os fluxos térmicos relacionados a ele.	55
Figura 7.2. Alguns perfis de temperaturas no tubo de separação até sua convergência.	57
Figura 7.3. Seções transversais à direção do escoamento.	58
Figura 7.4. Perfis da componente axial da velocidade nas seções transversais.	59
Figura 7.5. Eixos na direção r , definidos para apresentação dos resultados.	59
Figura 7.6. Perfis da componente axial da velocidade nos eixos r definidos.	60
Figura 7.7. Perfis de temperatura nas seções transversais definidas.	60
Figura 7.8. Perfis da temperatura ao longo do trocador, nos eixos definidos.	61
Figura 7.9. Temperaturas ao longo do recuperador de calor.	62
Figura 7.10. Temperaturas das faces interna e externa do tubo de separação, e sua diferença.	64

Figura 7.11. Perfis de fluxos térmicos que atravessam o tubo de separação ar/gás.	65
Figura 7.12. Fluxos de potência transferidos dos gases.	66
Figura 7.13. Perfis de $(q_r^+ - q_r^-)$ ao longo do trocador.	67
Figura 7.14. Perfis de $(q_z^+ - q_z^-)$ ao longo do trocador.	68
Figura A1 – Esquema do sistema de coordenadas com um feixe de radiação na direção.	70
Figura A2 – Detalhe do plano perpendicular ao eixo z.	71

Nomenclatura

		Equação de referência
A	área	5.10
a	coeficiente	4.2
a	coeficiente de absorção	5.19
C	constante	6.6
c	coeficiente	
F	função genérica	5.24
h	coeficiente de troca de calor	5.10
I	intensidade de radiação total	5.11
k	energia cinética turbulenta	
k	condutividade térmica	5.6
n	versor normal efluente de uma superfície	5.16
n	índice de refração do meio	6.2
p	pressão	5.1
Q	potência	5.11
q	fluxo térmico	5.1
R	raio interno do tubo que confina os gases de combustão	
r	coordenada radial	5.29
S	termo fonte	4.1
s	direção genérica	5.11
T	temperatura	5.1
t	tempo	5.1
V	volume do volume de controle	4.2
v	vetor velocidade	4.1
x,y,z	coordenadas retangulares	5.3

w	fator de ponderação	5.25
z	coordenada axial	5.29
α	coeficiente	5.32
β	coeficiente de expansão térmica	5.1
Γ	coeficiente difusivo	4.1
γ	fator da expressão de interpolação	5.41
ε	taxa de dissipação turbulenta	
ε	emissividade	5.50
ϕ	variável genérica	4.1
ϕ	função probabilidade de espalhamento	5.19
ζ	coseno diretor	5.29
ψ	coordenada angular	5.29
η	coseno diretor	5.28
θ	ângulo	5.11
μ	viscosidade	5.1
μ	coseno diretor	5.28
ξ	coseno diretor	5.28
ρ	densidade	4.1
ρ	refletividade	5.50
σ	parâmetro de similaridade	5.9
σ	coeficiente	5.19
σ	constante de Stefan-Boltzmann	6.2
τ	espessura ótica	6.3
ϕ	função dissipação	5.1
Ω	ângulo sólido	5.11

Índices subscritos

<i>a</i>	relativo ao ar
<i>b</i>	relativo à superfície negra
<i>c</i>	relativo ao grupo independente do termo fonte
<i>c</i>	relativo a condução de calor
<i>c</i>	relativo à coordenada angular do sistema ortogonal cilíndrico
<i>D</i>	relativo a valor pré-definido
<i>e</i>	relativo a radiação emitida
<i>g</i>	relativo aos gases de combustão
<i>h</i>	relativo ao coeficiente de película
<i>i</i>	relativo aos feixes incidentes
<i>k</i>	relativo à ordem de iteração
<i>m</i>	relativo à direção <i>m</i>
<i>o</i>	relativo aos feixes efluentes
<i>p</i>	relativo ao ponto nodal, centro do volume de controle
<i>p</i>	relativo ao coeficiente da variável principal no termo fonte
<i>r</i>	relativo a radiação
<i>s</i>	relativo a espalhamento da radiação
<i>w</i>	relativo à superfície adjacente ao volume
<i>ef</i>	relativo a efetivo
<i>ref</i>	relativo à radiação refletida
<i>x,y,z</i>	relativo às coordenadas retangulares
<i>n,s,e,w</i>	relativo aos pontos nodais adjacentes a norte, a sul, a este e a oeste

Superscritos

- + relativo ao sentido positivo do eixo
- relativo ao sentido negativo do eixo
- a relativo a valor admitido
- c relativo a valor calculado

Sobre-linha

- indicação de variável vetorial

Resumo

Este trabalho apresenta o método das ordenadas discretas para o cálculo das trocas de calor por radiação. Este método foi selecionado pela possibilidade de se integrar a um programa CFD (Computational Fluid Dynamics).

Inicialmente, o método foi experimentado na simulação de um problema puro de radiação, unidimensional, que possui solução analítica.

A seguir, foi aplicado na simulação de um trocador de calor de forno siderúrgico, constituído por dois tubos concêntricos, onde escoam, em fluxos turbulentos e co-correntes, gases de combustão no tubo interno, e ar no espaço anular. O método foi aplicado tanto no compartimento do ar quanto do gás, sendo que, nesse último, sua integração com a técnica CFD foi testada.

Os resultados da simulação foram comparados com dados de projeto.

Este trabalho mostra a simplicidade conceitual e a facilidade operacional do método.

Abstract

This work presents the discrete ordinates method for radiation heat transfer calculation. This method has been selected aiming for an integration with a CFD program.

The method has been tested to solve a radiative unidimensional problem which has an analytical solution.

Then, the method has been applied to simulate a two concentric tubes, with combustion gases flow in the inside tube and air flow in the cylindrical annulus. The flows are turbulent and cocurrent. The method has been applied on air and gas sections; and in the later, its integration with a CFD program was tested.

The simulation results have been compared with project parameters.

This work shows the conceptual simplicity and the operational facility of the method.

1. Introdução

As técnicas empregadas para projeto e desenvolvimento de sistemas de combustão, assim como para diagnóstico de problemas em equipamentos existentes têm sofrido mudanças consideráveis nos últimos anos.

As técnicas tradicionais baseadas na utilização intensiva de dados experimentais e correlações empíricas globais, obtidas em equipamentos e condições de operação semelhantes aos do projeto têm sido substituídas por programas computacionais que simulam os fenômenos complexos e interrelacionados que ocorrem numa câmara de combustão.

Os principais programas para simulação de equipamentos de combustão tiveram impulso a partir da desenvoltura que adquiriu a área da engenharia denominada CFD (Computational Fluid Dynamics), isto é, Dinâmica dos Fluidos Computacional.

Atualmente, o conhecimento de um conjunto de rotinas capazes de simular numericamente escoamentos mono ou multicomponente, mono ou bifásico, com transferência de calor, com reações químicas, já está consolidado e é relativamente difundido.

Os programas comerciais disponíveis para tal finalidade e mesmo programas desenvolvidos em universidades têm pequenas diferenças no conjunto de rotinas de base. Essas rotinas baseiam-se na solução das equações diferenciais de conservação da massa e da quantidade de movimento através de uma técnica de discretização¹.

Mas, se é verdade que as rotinas para a simulação de escoamentos estão relativamente consagradas e disseminadas no meio acadêmico brasileiro, não se pode fazer tal afirmativa para os métodos de cálculo das trocas de calor por radiação.

Apesar da importância fundamental desse tema para processos a altas temperaturas, são pouquíssimos os pesquisadores que, nas universidades ou centros de pesquisas do país, dedicam-se ao estudo desse tema.

¹ Por essa razão, pode-se falar de um método ou uma técnica CFD.

Alguns fatos ajudam a explicar essa situação:

- a difícil compatibilidade do equacionamento da troca de calor por radiação com as equações básicas de conservação. Radiação é um fenômeno que relaciona um volume de controle com todos os demais elementos (de volume e de superfície) do domínio, enquanto que as equações de conservação (massa, quantidade de movimento e energia, considerando-se trocas de calor por condução e convecção), relacionam um volume apenas com os elementos de volume ou de superfície imediatamente vizinhos;
- a necessidade de manipulações matemáticas trabalhosas para obter-se solução mesmo nos casos onde o cálculo da radiação térmica está dissociado do cálculo dos perfis das pressões e das velocidades.

O advento dos programas computacionais comerciais importados que incorporam rotinas para o cálculo da radiação térmica, presente nos dias de hoje, não torna prescindível o trabalho de pesquisadores brasileiros no tema da radiação, pelo contrário, há que se ter conhecimento do fenômeno, da aplicabilidade e limitações de modelos matemáticos desenvolvidos para sua simulação. Os modelos de radiação têm seu respectivos campos de aplicação e, assim, generalizações de alguns modelos podem levar a resultados incorretos.

Esses programas comerciais apenas facilitam o trabalho matemático de se alcançar a convergência de soluções, o que também deve ser valorizado. Mas, apresentam a modelagem de importantes fenômenos, da qual a radiação é apenas um exemplo, como “caixas-pretas”, isto é, não é possível fazer alterações no código fonte que equaciona o fenômeno.

Assim, para poder-se utilizar plenamente esses programas, há que se ter especialistas em radiação, nos métodos de cálculo das trocas radiativas e nos outros fenômenos físicos presentes no problema em estudo, além de experiência em simulação numérica na área CFD.

Essas razões, associadas ao fato da especialização profissional do autor situar-se na área de combustão industrial, levaram à escolha do tema do presente trabalho.

Anteriormente a este trabalho, o autor já detinha conhecimento de assuntos correlatos, na medida em que o equacionamento e o tratamento matemático de escoamento turbulento confinado foi objeto de sua dissertação de mestrado.

2. Objetivo proposto e formulação do problema

2.1. Objetivo proposto

O objetivo proposto do programa de doutorado foi o estudo das trocas de calor por radiação em equipamentos de uso industrial. Além disso, propunha-se a seleção de um método matemático capaz de calcular essas trocas e capaz de se integrar num programa CFD.

O método denominado de “ordenadas discretas” foi selecionado por atender às duas especificações mencionadas acima.

Inicialmente, verificou-se a validade do método aplicando-o num problema que tem solução conhecida.. Esse problema tratou exclusivamente de trocas de calor por radiação, sem a influência de condução ou convecção.

Em seguida, o método foi, então, incorporado ao programa CFD desenvolvido pelo autor no seu programa de mestrado e aplicado na simulação de um recuperador de calor de forno da indústria siderúrgica. Esse caso foi escolhido por representar uma aplicação industrial do qual se obteve parâmetros de operação.

2.2. Formulação do problema do recuperador de calor

O objeto do estudo é um recuperador de calor de um forno de laminação de uma siderúrgica. Este recuperador foi objeto de estudo da equipe do atual Laboratório de Combustão e Gaseificação do IPT em 1982. Naquela época, essa equipe, da qual o autor faz parte, detinha apenas conhecimentos incipientes sobre o cálculo da troca de calor por radiação. Desenhos e dados operacionais desse recuperador, fornecidos pela siderúrgica naquela época, foram guardados e serão utilizados nesse trabalho.

O recuperador de calor se compõe de dois dutos circulares concêntricos. Escoando co-currentemente tem-se, no duto interno, gases quentes obtidos pela combustão de óleo no forno e, na coroa circular, ar que irá alimentar os queimadores do forno.

O problema proposto é a simulação da operação desse equipamento obtendo-se a distribuição de temperaturas dos gases no duto interno e do ar na coroa circular e comparar as temperaturas de saída desses gases e do ar com os valores fornecidos pelo seu fabricante.

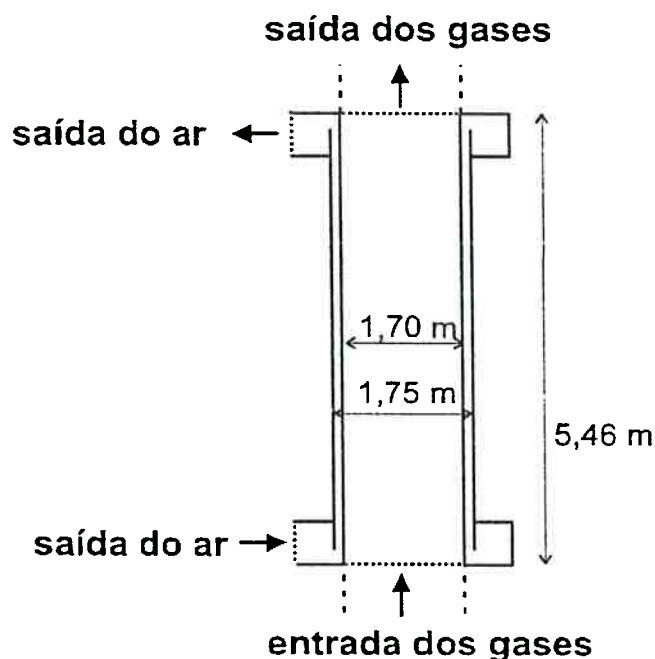


Figura 2.1. Desenho esquemático do trocador de calor.

Foram adotadas as seguintes hipóteses

- regime estacionário
- fluido incompressível
- escoamento turbulento¹ e axissimétrico

¹ Os escoamentos dos gases e do ar têm números de Reynolds da ordem de 107.000 e 64.000 respectivamente.

É simples perceber que o sistema de coordenadas que melhor se adapta a este tipo de escoamento é o cilíndrico.

A solução desse problema, rigorosamente, exige a solução de dois problemas acoplados:

- determinar os perfis de temperaturas e de trocas de calor para o escoamento de ar na coroa circular;
- determinar os perfis de velocidades, temperaturas e trocas de calor para o escoamento dos gases de combustão no duto interno do recuperador de calor.

A separação nesses dois problemas ocorre naturalmente por existirem dois escoamentos distintos, confinados em seções fisicamente separadas.

Ambos os problemas foram resolvidos utilizando-se o método das ordenadas discretas para o cálculo das trocas de calor por radiação.

O primeiro problema, do escoamento de ar na coroa circular, foi resolvido admitindo as seguintes premissas:

- o escoamento turbulento tem perfil de velocidades achatado (“plug-flow”).
- o meio, por ser ar atmosférico, não é participativo com relação à radiação².
- a radiação emitida pelas superfícies são isotrópicas, isto é, sua intensidade independe da direção do feixe de radiação.

O segundo problema, do escoamento de gases no duto interno, foi resolvido admitindo as seguintes premissas:

- o meio não é espalhador de radiação, já que se pode considerar desprezível a concentração de material particulado, que é o principal agente de espalhamento da radiação em gases originados de câmaras de combustão;

² Esse fato se traduziu no algoritmo de cálculo das trocas de calor por radiação da seguinte maneira:

- desprezou-se os termos relativo a espalhamento da radiação;
- admitiu-se um valor muito pequeno para o coeficiente de absorção total do meio (0,001).

- a radiação emitida pelo gás é isotrópica.
- o meio é cinzento, isto é, o coeficiente de absorção independe do comprimento de onda da radiação³.

Os dois problemas foram resolvidos de maneira acoplada, isto é, ambos foram resolvidos para o mesmo perfil de temperaturas da parede entre eles.

³ Grande parte da literatura encontrada na pesquisa bibliográfica realizada relativa a processo de combustão adota essa hipótese. Essa hipótese é válida especialmente quando a presença de fuligem ou material particulado for expressiva. Essa hipótese permite uma grande simplificação no tratamento matemático comparativamente ao caso de equacionar as trocas de calor por radiação em cada intervalo do espectro de comprimentos de onda.

3. Revisão da literatura

Neste trabalho, a pesquisa bibliográfica esteve orientada basicamente ao estudo da radiação térmica e à sua modelagem matemática.

O tema da turbulência e os métodos constitutivos de uma técnica CFD foram temas de estudo por ocasião do mestrado do autor, e assim, a pesquisa bibliográfica desses tópicos, bem como comentários acerca dela, foram apresentados naquela dissertação de mestrado [JEN (1989)].

A pesquisa bibliográfica não se restringiu ao método das ordenadas discretas, empregado para o cálculo das trocas de calor por radiação que o autor deste trabalho considera como um dos mais adequados para utilizar-se acoplado a um programa CFD. Mesmo porque essa opinião foi uma das conclusões da pesquisa bibliográfica como será comentado mais adiante.

No início da pesquisa bibliográfica procurou-se recuperar e estudar os trabalhos numa ordem próxima da cronológica para ter uma visão da evolução dos métodos de simulação da radiação térmica. Assim, a apresentação do trabalho realizado no levantamento, recuperação e estudo da literatura relacionada ao tema tem também algum caráter de resenha.

Inicialmente é conveniente comentar alguns dos poucos livros que tratam exclusivamente do tema da radiação. Nos cursos de graduação, radiação é estudado na disciplina “Transferência de Calor” ou “Fenômenos de Transporte” e, assim, comparece nos livros textos dessas disciplinas em pequenos capítulos.

Um dos primeiros livros que tratou exclusivamente da radiação foi escrito por HOTTEL; SAROFIM (1967); nele são apresentados detalhadamente os conceitos e a técnica envolvidos no método das zonas para o cálculo das trocas de calor por radiação.

No entanto, aparentemente, o livro que se consagrou como livro texto nesta área foi o escrito por SIEGEL; HOWELL (1972). Tem entre seus méritos, uma apresentação dos tópicos bastante didática e adoção de uma cuidadosa notação para expressar as equações

relacionadas no fenômeno. Sua terceira edição, de 1992, apresenta inúmeros acréscimos em relação à edição anterior.

Outro livro que merece destaque foi escrito por MODEST (1993); nele encontram-se vários detalhes de desenvolvimento dos modelos elaborados para o cálculo da troca de calor por radiação.

Contendo uma exposição resumida dos conceitos necessários para a modelagem da troca de calor por radiação, o artigo de VISKANTA (1987) foi muito proveitoso ao autor desse texto, assim como o seminário interno de SOUSA (1989).

Método das zonas

O método das zonas, pioneiro nas aplicações de engenharia, está detalhadamente descrito em HOTTEL; SAROFIM (1967). Nesse método, a superfície e o volume do domínio de interesse são divididos num certo número de zonas, em cada qual admite-se distribuição uniforme de propriedades. As áreas diretas de troca de calor entre os elementos de superfície e de volume são calculadas e, a seguir, as áreas totais são obtidas através de operações matriciais. Assim, esse método reduz o problema da transferência radiativa à solução de um conjunto de equações algébricas não lineares.

Originalmente, o método das zonas foi aplicado em casos nos quais os perfis de velocidade eram conhecidos ou estabelecidos e admitidos como independentes das trocas de calor por radiação. Mais recentemente, o método das zonas foi conjugado com uma técnica CFD, isto é, o cálculo do perfil de velocidades também faz parte do problema, mas admite que o meio é “transparente” à radiação, isto é, que as trocas de calor por radiação só ocorrem entre os elementos de superfície presentes no domínio de interesse. Dessa maneira, a radiação não participa nas equações de conservação dos elementos de volumes internos do domínio¹.

VAN DONGEN (1983) apresentou um trabalho no qual empregou um método semi-empírico, que combinava o método das zonas para o cálculo das trocas radiativas e uma

¹ É dessa maneira que os primeiros programas comerciais CFD simulavam os casos de combustão, como pode-se perceber da leitura da notas do programa PHOENIX versão 1.5.

técnica integral para o cálculo dos perfis de escoamento, para a simulação de fornalhas. O parâmetro empírico foi a viscosidade turbulenta. A comparação com dados medidos foi satisfatória para os casos onde o escoamento advindo do queimador não tinha componente tangencial.

Método de MonteCarlo

O método de Monte Carlo enfoca a radiação como o trânsito de “pacotes de energia” (fótons). Esse método realiza o acompanhamento de um conjunto de fótons desde os seus pontos de emissão, passando por suas iterações com partículas de matéria do meio até os pontos onde são completamente absorvidos. As trocas de calor por radiação são calculadas através do balanço estatístico do conjunto de fótons. A resolução de problemas matemáticos por métodos estatísticos envolve sempre o uso de números randômicos².

O método de Monte Carlo apresenta as deficiências de requerer grande tempo computacional e estar sujeito ao erro estatístico associado à geração de números randômicos.

STEWART; CANNON (1971) apresentaram um trabalho baseado na aplicação do método de “Monte Carlo” para o cálculo do fluxo radiativo em fornalha cilíndrica, em casos cujos perfis de escoamento eram estabelecidos “a priori”.

Os métodos das zonas e o de Monte Carlo distinguem-se dos demais métodos pelo tipo de equacionamento adotado para o cálculo da troca radiativa. O primeiro método, equaciona esse cálculo a partir de um balanço entre duas zonas fazendo uso das “áreas de troca”. O segundo obtém esse cálculo a partir do acompanhamento de um conjunto de fótons.

Ambos os métodos apresentam equações para a troca radiativa diferentes da forma diferencial. Assim, eles são incompatíveis com os métodos que partem das equações diferenciais de conservação, como os da técnica CFD.

Método dos fluxos

Os métodos que se conjugam com essa técnica são os chamados métodos de fluxos. A equação empregada para o cálculo da troca radiativa é obtida a partir do balanço diferencial da intensidade de radiação (espectral ou total) realizado num volume de controle. Essa equação diferencial, denominada de “equação de transferência radiativa” contém uma integral que representa a influência das intensidades de radiação incidentes no volume de controle vindas de todas as direções³.

Assim, todos os métodos de fluxos admitem que a intensidade de radiação é função da posição e da direção de propagação, e do comprimento de onda, no caso de considerar-se radiação espectral. Usualmente, a dependência funcional com a direção complica o problema já que todas as direções possíveis devem ser consideradas. Por essa razão, é desejável separar a dependência direcional da dependência espacial para simplificar o equacionamento matemático.

Se for assumido que a intensidade de radiação seja uniforme em certos intervalos do ângulo sólido, a equação de transferência radiativa é significativamente simplificada reduzindo-se as equações integro-diferenciais a um conjunto de equações diferenciais lineares acopladas em termos das intensidades ou fluxos de radiação médios. Variando-se o número de ângulos sólidos para os quais a intensidade de radiação é assumida como constante, obtém-se diferentes métodos de fluxos, tais como de 2 fluxos, de 4 ou 6 fluxos.

Uma outra maneira de evitar expressões complicadas na equação de transferência radiativa, devido à dependência angular da intensidade, é integrar aquela equação no espaço depois de multiplicá-la por certos cosenos diretores. Essa técnica origina os chamados métodos “dos momentos”.

O método dos “harmônicos esféricos” é desenvolvido similarmente, mas um método matemático mais sofisticado de integração da equação de transferência radiativa é empregado.

² Números randômicos podem ser associados aos “jogos de azar”, presentes em cassinos. Por essa razão, esses métodos de amostragem receberam o nome de Monte Carlo, lembrando o famoso casino europeu de Mônaco.

³ Por essa razão, essa é uma equação integro-diferencial.

O método das “ordenadas discretas” é obtido discretizando-se todo o ângulo sólido num número finito de direções ordenadas com seus fatores de ponderação correspondentes. A equação de transferência radiativa é aplicada para cada direção ordenada e a integral presente naquela equação é substituída por uma quadratura numérica.

Um outro método que parte da equação de transferência radiativa mas combina técnicas do método das zonas e de Monte Carlo é o chamado método da “transferência discreta”, desenvolvido por LOCKWOOD; SHAH (1976). Eles admitiram que os feixes de radiação têm suas direções estabelecidas deterministicamente e não estatisticamente, e que todas as direções constituem um conjunto ortogonal, obtendo-se uma técnica que requer menor tempo de computação que o método das zonas e o de Monte Carlo.

Métodos dos múltiplos fluxos

GIBSON; MONAHAN (1971) foram os autores de um trabalho que pode-se colocar como um dos precursores do método dos fluxos. Eles utilizaram hipóteses simplificadoras assumidas por estudiosos da radiação entre estrelas para aplicá-las na simulação de uma chama de combustível pulverizado de geometria axissimétrica. Tem como base um modelo unidimensional para a equação de transferência da radiação, que, após manipulações matemáticas, apresenta o termo da troca de calor por radiação como resultado de um sistema composto por duas equações diferenciais ordinárias. Essas manipulações matemáticas não são fáceis de se acompanhar, pelo contrário, pode-se dizer que elas são relativamente obscuras. Não se conseguiu chegar à mesma expressão que aqueles autores. Para o mesmo caso, MODEST (1993) apresenta uma formulação alternativa, rigorosamente correta no seu desenvolvimento. Essa formulação alternativa é a base de trabalhos posteriores enfocando o uso do método dos fluxos, em especial, o das ordenadas discretas.

LOWES et al. (1974) estenderam a equação final obtida por Gibson e Monahan para um domínio bidimensional, desenvolvendo um modelo de quatro fluxos para um sistema axissimétrico e compararam os resultados obtidos por esse modelo com os obtidos pelo método das zonas e com os valores medidos de experimentos com uma fornalha da International Flame Research Foundation (IFRF).

GOSMAN; LOCKWOOD (1973) foram pioneiros em inserir um modelo de quatro fluxos para transferência radiativa num programa CFD. Naquela época, os modelos de turbulência ainda não estavam consolidados e variavam bastante entre si. Assim, não é de estranhar as discrepâncias encontradas comparando-se os resultados obtidos com os valores medidos numa fornalha a gás da IFRF.

DEMARCO; LOCKWOOD (1975) criaram um método de seis fluxos e o aplicaram no cálculo das trocas de calor de uma fornalha retangular. A base desse método é a admissão de uma expressão para a distribuição angular da intensidade de radiação baseada na sua expansão em série de Taylor, truncando-se após o segundo termo nas direções coordenadas. Essa distribuição foi, então, substituída na equação de transferência radiativa e a equação resultante integrada sobre ângulos sólidos tendo os eixos coordenados cartesianos como eixos de simetria para produzir as seis aproximações da equação de transferência.⁴

Esse mesmo método é o centro dos artigos de LOCKWOOD; SHAH (1976) e HJERTAGER; MAGNUSSEN (1982). O primeiro desses artigos, reportou a sua aplicação em casos que possuem soluções exatas, portanto dando ênfase a casos unidimensionais. O segundo reportou a aplicação desse método acoplado a um programa CFD, simulando, em três dimensões, uma fornalha retangular IFRF.

VARMA (1979) apresentou um método de quatro fluxos cuja contribuição foi o tratamento dado ao termo de espalhamento da equação de transferência radiativa. Tal termo não pode ser desprezado em chamas de carvão pulverizado, que são os casos abordados pelo artigo.

Método das ordenadas discretas

Pelo estudo da literatura recolhida, pode-se perceber que é esse método vem ganhando impulso mais recentemente. As razões que explicam esse fato são:

⁴ Esse método está fortemente condicionada à geometria retangular. Uma indicação disso é que não se encontra na literatura nenhum registro de sua aplicação em casos com outros tipos de geometria.

- a simplicidade para resolver a equação de transferência radiativa, incluindo-se o termo de espalhamento da radiação, que pode ser considerado tanto isotrópico quanto anisotrópico. Essa simplicidade contrasta com outros métodos que exigem análises matemáticas sofisticadas;
- a facilidade de resolver o sistema de equações diferenciais ordinárias. Para isso, pode-se lançar mão de pacotes de rotinas matemáticas disponíveis comercialmente, dispensando o pesquisador de algum trabalho computacional;
- a facilidade de integrar suas rotinas computacionais em códigos computacionais CFD.

Os pesquisadores Fiveland e Truelove foram os pioneiros em aplicar o método das ordenadas discretas a problemas da área de transmissão de calor. Historicamente esse método foi proposto para o estudo da troca de radiação entre estrelas e, para o caso bidimensional em coordenadas cilíndricas, recebeu uma grande contribuição de físicos no estudo do fluxo de nêutrons em reatores atômicos.

FIVELAND (1982) apresenta uma aplicação do método para geometrias cilíndricas com simetria axial e simulou a operação de duas fornalhas cilíndricas. Os resultados obtidos foram comparados com os obtidos nas aplicações de outros métodos numéricos aos mesmos casos, sem uma confrontação com valores medidos. No desenvolvimento apresentado, ele utiliza uma aproximação do termo da dependência angular da intensidade de radiação sugerido por CARLSON; LATHROP (1968) em seu estudo de reatores atômicos.

O desenvolvimento que Fiveland elaborou serviu de base ao cálculo das trocas radiativas do problema do presente trabalho, na medida em que poucas modificações a esse desenvolvimento foram introduzidas por pesquisadores até os dias de hoje. Esse desenvolvimento é apresentado no capítulo 5.

FIVELAND (1984,1987) apresentou a aplicação do método para domínios retangulares. Particularmente, o primeiro desses artigos reporta as aplicações do método que ele fez em vários casos, comparando seus resultados com os de outros métodos, numa espécie de “benchmark”, isto é, tendo como critério tempo de processamento de computador. E

concluiu que esse método requer menor tempo de processamento que o método das zonas e dos harmônicos esféricos.

TRUELOVE (1987) modificou o trabalho de FIVELAND (1984) apenas num detalhe da técnica da quadratura numérica utilizada para o cálculo da integral sobre as direções em 4π estereo-radianos.

Em outro trabalho, TRUELOVE (1988) aplicou o método simulando uma fornalha tridimensional, de formato retangular, com meio participativo. O caso estudado foi idealizado, isto é, não correspondeu a ensaio real onde tenham sido feitas medições. Truelove comparou os resultados com outros obtidos através dos métodos de harmônicos esféricos e das zonas. Adotando o último como aquele que produz resultados mais próximos da realidade, concluiu que o método das ordenadas discretas produz resultados superiores aos do método dos harmônicos esféricos.

JAMALUDDIN; SMITH (1988) aplicaram o método das ordenadas discretas no estudo da combustão de carvão pulverizado em fornalha cilíndrica. Como as incertezas dos resultados numéricos podem ser devidos à falta de conhecimento das propriedades radiativas e também devido a imprecisões inerentes ao modelo de radiação, esses pesquisadores fizeram um estudo de sensibilidade do modelo em relação a variáveis, como as propriedades radiativas e parâmetros relevantes do sistema (tamanho das partículas e temperatura das partículas). Neste estudo concluíram que:

- tanto as aproximações de segunda e quarta ordem, S_2 e S_4 , predizem a transferência radiativa num meio participativo com precisão aceitável, não sendo necessário empregar aproximações de maior ordem⁵;
- imprecisões na determinação da concentração e temperatura das partículas acarretam maiores desvios dos cálculos da transferência radiativa do que aqueles causados por imprecisões na determinação dos coeficientes de absorção e de espalhamento do meio.

⁵ A ordem do método das ordenadas discretas está associada ao número de direções que representa todo o ângulo sólido em torno do volume elementar (ver item 5.2.3).

KUMAR et al. (1990) apresentaram um trabalho no qual compararam vários esquemas de quadratura no cálculo da transferência radiativa através de meio participativo num domínio unidimensional.

4. Técnica “CFD”: equacionamento básico e método numérico

4.1. *Preâmbulo*

Nos dias de hoje, é conhecido como CFD a área da engenharia que, através de simulação numérica, estuda problemas nos quais a influência do perfil de escoamento é fundamental. É uma área bastante abrangente em relação tanto à diversidade de problemas abordados (como exemplos, formação de vórtices na esteira de um corpo rígido imerso num fluxo ou escoamento supersônico), quanto aos métodos numéricos empregados na elaboração do programa de simulação.

Considerando-se que:

- esta área é hoje relativamente conhecida, sendo objeto de cursos de pós-graduação em todas as principais escolas de engenharia do país;
- os tópicos fundamentais abordados por este programa de doutorado são a troca de calor por radiação e o método das ordenadas discretas para o seu cálculo e, assim, a técnica CFD é apenas um “ambiente envoltório” ao qual se incorporou a subrotina computacional do método,

serão descritas apenas sucintamente, as características básicas do método CFD empregado, reservando-se uma descrição mais detalhada para a modelagem da troca de calor por radiação.

O programa CFD utilizado emprega o método denominado “dos volumes finitos”, também conhecido como o método das “diferenças finitas com enfoque no volume de controle”. Ele se distingue do método das diferenças finitas convencional por discretizar as equações obtidas após a integração das equações diferenciais num volume elementar, enquanto que no método das diferenças finitas a discretização é obtida partindo-se diretamente das equações diferenciais.

4.2. A variante do método empregada na técnica CFD

Pode-se dizer que o método dos volumes finitos possui inúmeras variantes, em função dos diferentes modos de abordagem de algumas especificidades do método. Algumas dessas especificidades são comentadas a seguir:

- a seleção das variáveis a serem determinadas.

Por este critério distingue-se duas linhas: aquela que baseia o equacionamento do escoamento nas funções “linha de corrente” e “vorticidade” e a linha conhecida com “das variáveis primitivas”, que baseia o equacionamento do escoamento nas variáveis: pressão, componentes do vetor velocidade etc.

O método escolhido baseia o equacionamento nas “variáveis primitivas”.

- a utilização de malhas “desencontradas” para as componentes da velocidade.

Método tradicionalmente utilizado, adota uma malha para as variáveis principais do fenômeno (pressão, temperatura, concentração, “k e epsilon” etc) e malhas específicas para as componentes da velocidade. Tais malhas são especificadas de maneira que essas componentes sejam calculadas nas interfaces dos volumes de controle da malha das variáveis principais.

- as funções de interpolação utilizadas para se calcular os valores das variáveis, ou de suas derivadas, entre dois pontos nodais.

Os métodos da diferença central, “upwind” e “híbridos” estão entre os métodos utilizados para isso. Adotou-se um método “híbrido” descrito por MALISKA (1995).

- a forma da equação para atualização em cada iteração do campo de pressões na assim denominada formulação “incompressível”.

Nesta formulação, uma equação de estado é empregada para atualização em cada iteração do campo de densidade e o método pelo qual se determina uma expressão para atualizar os valores do campo de pressão é o que diferencia uma variante de outra. Os primeiros métodos reportados em literatura e, portanto relativamente conhecidos, são SIMPLE, SIMPLER, SIMPLEC entre outros. Adotou-se o método denominado PRIME, também descrito por MALISKA (1995).

A seguir serão comentados tópicos especiais do método escolhido.

4.3. Forma geral das equações diferenciais de conservação

As equações diferenciais de conservação da massa, quantidade de movimento e energia constituem as equações básicas nas quais se estrutura a técnica CFD.

Essas equações podem ser representadas pela seguinte expressão geral:

$$\rho \bar{v} \cdot \nabla \phi = \nabla \cdot (\Gamma_{\phi} \nabla \phi) + S_{\phi} \quad (4.1)$$

onde os parâmetros ϕ , Γ_{ϕ} e S_{ϕ} assumem grupos diferentes para representar as diferentes equações¹.

Discretizando-se as equações diferenciais de conservação obtém-se a seguinte expressão:

$$a_p \phi_p = \sum_i a_i \phi_i + S_c I_p \quad (4.2)$$

¹ A hipótese de regime estacionário está subentendida nessa expressão.

onde
$$a_p = \sum_i a_i - S_p V_p \quad (4.3)$$

ϕ_p é o valor da variável ϕ no volume de controle em consideração

ϕ_i é o valor da variável ϕ no volume de controle adjacente i

a_i é o coeficiente que traduz a influência do volume adjacente i ao volume em consideração

$S = S_p \phi_p + S_c$ é o termo fonte da variável ϕ

V_p é o volume do volume de controle em consideração

4.4. Turbulência

O programa desenvolvido no programa de mestrado admite os escoamentos como turbulentos como são a maior parte dos escoamentos tratados pela engenharia. Escoamentos laminares também podem ser simulados pois são contemplados pelo equacionamento mais abrangente dos escoamentos turbulentos (isto é, o equacionamento do escoamento laminar é formalmente igual ao do caso turbulento, porém com simplificações nos coeficientes).

O enfoque dado à turbulência, neste trabalho, é o enfoque mais tradicional, baseado nos valores médios, no tempo, das grandezas envolvidas no escoamento. São exatamente tais valores médios aqueles obtidos pelos instrumentos usuais de medição como, por exemplo, o tubo de Pitot, o manômetro tipo Bourdon, o termopar etc.

Por esse enfoque, os parâmetros principais do escoamento são decompostos nos seus componentes médio e flutuante e aplicados nas equações diferenciais de conservação. Nessas equações, os termos são reagrupados, com considerações quanto aos produtos de flutuações, obtendo-se novas equações diferenciais cujas variáveis principais são os valores médios dos parâmetros principais do escoamento.

Na equação de conservação da quantidade de movimento, o termo das tensões de Reynolds é modelado segundo o modelo de Boussinesq, que o calcula como o produto do gradiente da velocidade média no tempo e de um parâmetro que se denomina viscosidade turbulenta, fazendo uma analogia com a lei de Newton da viscosidade.

Dessa maneira, chega-se a uma expressão formalmente igual àquela que representa a conservação da quantidade de movimento para o escoamento laminar, escrita em termos dos valores médios das variáveis e com seu coeficiente de transferência modificado para levar em conta a influência da turbulência.

A introdução da viscosidade turbulenta fornece um meio de fechar o equacionamento baseado na equação de conservação da quantidade de movimento pois, assim, o efeito da turbulência resume-se no cálculo do perfil dessa viscosidade. As demais equações de conservação (energia, espécies químicas...) têm seus coeficientes de transferência alterados por similaridade em relação à viscosidade turbulenta.

A viscosidade turbulenta é propriedade do escoamento e não do fluido e, mesmo entre dois pontos adjacentes, pode variar bastante dependendo da intensidade da turbulência.

Para introduzir o efeito da turbulência utilizou-se o modelo de dois parâmetros denominado “ k e ϵ ”, provavelmente o mais utilizado na técnica CFD. As equações de “ k e ϵ ” também podem ser expressas pela expressão geral, equação (4.1).

Junto à superfícies sólidas, as equações de “ k e ϵ ”, possuem condições de contorno obtidas a partir das assim chamadas “funções de parede”. Essas têm como base expressões empíricas levantadas para a distribuição das velocidades no escoamento turbulento confinado por superfícies sólidas.

4.5. Comentários sobre a condição de contorno na seção de saída das variáveis do escoamento

Os gases produto da combustão percorrem um trecho de 5,46 metros no duto cujo diâmetro mede 1,7 metros. Nessa condição, na qual o percurso é pouco maior do que

três diâmetros do conduto, seu perfil de velocidades da seção transversal ao escoamento pouco variará a partir do perfil inicial². As variações ocorrerão somente nas proximidades da parede do duto.

Admitindo que os gases entram no recuperador de calor com perfil achatado (“plug-flow”), pode-se classificar o escoamento como parabólico, isto é, escoamento sem correntes de recirculação.

Considerando-se ainda, que a jusante do recuperador de calor situa-se um “recuperador de convecção” e que, esse último, constituído por um banco de tubos imersos no escoamento do gás, atuará como retificador do fluxo de gás, a condição de contorno mais adequada para a seção de saída é a do perfil de velocidade estacionário.

² O perfil do “escoamento plenamente desenvolvido” só é alcançada após cerca de 10 a 12 diâmetros de distância da seção de entrada.

5. A equação de energia e o método das ordenadas discretas

5.1. A equação de energia

A forma geral da equação diferencial de conservação de energia para um fluido em movimento pode ser escrita, conforme SPARROW; CESS (1978) como¹:

$$\rho c_p \frac{DT}{Dt} = \rho c_p \left(\frac{\partial T}{\partial t} + \bar{v} \cdot \nabla T \right) = -\nabla \cdot \bar{q} + \beta T \frac{Dp}{Dt} + \mu \Phi \quad (5.1)$$

onde \bar{q} é o vetor fluxo térmico total

Φ é a função dissipação

Preferiu-se essa expressão, que tem a temperatura como variável dependente, do que uma expressão em função da entalpia de estagnação. Assim, a variação da energia interna devido à variação da energia cinética está sendo desprezada, o que é bastante razoável no escoamento sem altos gradientes de velocidade do duto central do recuperador de calor.

Os termos associados ao trabalho realizado pelas forças de pressão $\left(\beta T \frac{Dp}{Dt} \right)$, à dissipação viscosa $(\mu \Phi)$ e à taxa de liberação de calor internamente ao volume de controle (Q) foram desprezados no desenvolvimento realizado por serem irrelevantes.

Efetuada essas simplificações e admitindo regime estacionário, a equação anterior se reduz a:

$$\rho \bar{v} \cdot \nabla T = \frac{-\nabla \cdot \bar{q}}{c_p} \quad (5.2)$$

¹ Não se considera outros efeitos da radiação, por serem insignificantes face ao seu efeito como mecanismo adicional de troca de calor.

Como as trocas por radiação apresentam-se no termo $\nabla \cdot \vec{q}$, vale a pena comentá-lo. Esse termo expressa as trocas térmicas que o volume do fluido realiza com as suas vizinhanças e sua forma matemática foi obtida através do balanço dessas trocas num volume de controle. Por exemplo, em coordenadas retangulares, tem-se o volume de controle representado na Figura 5.1.

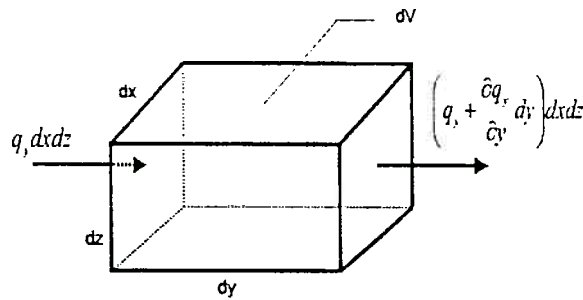


Figura 5.1. Esquema de volume de controle mostrando o fluxo térmico na direção y.

O termo que expressa a potência que o volume de controle recebe das vizinhanças é:

$$\left[q_x - \left(q_x + \frac{\partial q_x}{\partial x} dx \right) \right] dy dz + \left[q_y - \left(q_y + \frac{\partial q_y}{\partial y} dy \right) \right] dx dz + \left[q_z - \left(q_z + \frac{\partial q_z}{\partial z} dz \right) \right] dx dy \quad (5.3)$$

Portanto, q_x , por exemplo, representa o fluxo térmico total que incide na face $x = 0$ do volume de controle.

A expressão (5.3) pode ser dividida por $dV = dx dy dz$, resultando no termo presente na equação (5.1):

$$-\left(\frac{\partial q_x}{\partial x} + \frac{\partial q_y}{\partial y} + \frac{\partial q_z}{\partial z} \right) = -\nabla \cdot \vec{q} \quad (5.4)$$

Se o meio é participante quanto à radiação, isto é, se emite, absorve e espalha radiação, o vetor fluxo térmico total pode ser desmembrado em dois termos como:

$$\vec{q} = \vec{q}_c + \vec{q}_r \quad (5.5)$$

onde \vec{q}_c é o vetor fluxo de calor devido à condução

\bar{q}_r é o vetor fluxo térmico devido à radiação

Os componentes de \bar{q}_r representam os fluxos de radiação incidentes em cada uma das faces do volume de controle e, \bar{q}_r representa o fluxo total de radiação incidente no volume, chamado de vetor fluxo térmico por radiação.

O vetor fluxo de calor por condução é usualmente representado como:

$$\bar{q}_c = -k \nabla T \quad (5.6)$$

Assim, utilizando-se as equações (5.5) e (5.6), a expressão (5.2) pode ser reescrita como:

$$\rho \bar{v} \cdot \nabla T = \nabla \cdot \left(\frac{k}{c_p} \nabla T \right) - \frac{\nabla \cdot \bar{q}_r}{c_p} \quad (5.7)$$

Essa equação será transformada para considerar a influência da turbulência nas trocas térmicas. Essa transformação é análoga àquela realizada na equação de conservação da quantidade de movimento, quando se substituiu a viscosidade pela viscosidade efetiva. No caso presente, a transformação consiste na substituição do coeficiente difusivo por um coeficiente efetivo, que será relacionado à viscosidade efetiva por similaridade. Assim:

$$\rho \bar{v} \cdot \nabla T = \nabla \cdot (\Gamma_{T,ef} \nabla T) - \frac{\nabla \cdot \bar{q}_r}{c_p} \quad (5.8)$$

onde $\Gamma_{T,ef} = \frac{\mu_{ef}}{\sigma_T}$ (5.9)

$\Gamma_{T,ef}$ é o coeficiente difusivo efetivo da equação de energia

μ_{ef} é a viscosidade turbulenta

σ_T é o parâmetro de similaridade entre $\Gamma_{T,ef}$ e μ_{ef} , e é igual a 0.7

A expressão (5.8) encontra-se escrita segundo a forma geral das equações de conservação, equação (4.1), e, portanto, uma expressão discretizada semelhante à equação (4.2) pode ser derivada a partir dela. Essa equação resultante será válida para todos os pontos do domínio onde forem válidos os coeficientes difusivos efetivos, o que exclui os pontos próximos às camadas limites adjacentes a superfícies sólidas. O uso desses coeficientes não produz bons resultados nessas regiões por estas não apresentarem a turbulência plenamente desenvolvida.

Essa restrição pode ser contornada com o uso das expressões empíricas que calculam os coeficientes locais de troca de calor (“coeficientes de película”), que, por sua vez, permitem o cálculo dos fluxos de potência trocados entre a superfície e o fluido.

Assim, a equação discretizada resultante, válida para os volumes de controle adjacentes às superfícies sólidas, sofre duas alterações:

- o coeficiente (α_i) que traduz a influência da região próxima da camada limite ao volume de controle em consideração é anulado,
- é adicionado um termo fonte correspondente ao fluxo de potência trocado entre a superfície sólida e o volume de controle. Esse termo é equacionado como:

$$q_h = \frac{h A_r (T_w - T_p)}{c_p V_p} \quad (5.10)$$

onde h é o coeficiente de película

A_r é a área de troca de calor entre o volume de controle e a superfície que lhe é adjacente

T_w é a temperatura da superfície

T_p é a temperatura do volume de controle

V_p é o volume do volume de controle.

Voltando à equação (5.8), deve-se agora, relacionar \bar{q}_r com as intensidades de radiação que incidem e/ou partem do volume de controle.

A intensidade de radiação pode ser definida como o fluxo de radiação, associado a um feixe que se propaga em determinada direção, por unidade de ângulo sólido centrado na direção de propagação do feixe.

Considerando que a energia radiativa apresenta-se segregada por diferentes intervalos de comprimento de onda, pode-se distinguir entre intensidade de radiação espectral (associada a um determinado intervalo do espectro de comprimentos de onda) e a intensidade de radiação total, isto é, considerando todo o espectro de comprimentos de onda.

Assim:

Intensidade espectral $I_\lambda \equiv$ fluxo de potência / ângulo sólido / comprimento de onda

Intensidade total $I \equiv$ fluxo de potência / ângulo sólido

No desenvolvimento subsequente será utilizado somente o conceito de intensidade de radiação total.

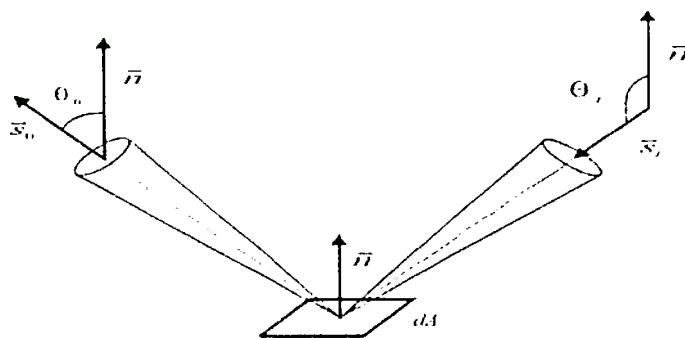


Figura 5.2. Esquema de feixes de radiação incidente e emergente de uma superfície.

A partir dessas definições, observando-se a Figura 5.2, a potência que atinge a superfície dA devida a um feixe de radiação vindo da direção \bar{s}_1 é:

$$dQ_i = I(\bar{s}_1) d\Omega_i dA \cos\theta_1 = I(\bar{s}_1) d\Omega_i (\cos\theta_1 dA) \quad (5.11)$$

Da mesma maneira, a potência que sai da superfície dA devida a um feixe de radiação pela direção \vec{s}_o é:

$$dQ_o = I(\vec{s}_o) d\Omega_o \cos\theta_o dA \quad (5.12)$$

Pode-se perceber que fluxos associados a feixes de radiação que saem da superfície dA são positivos já que $\cos\theta_o > 0$ enquanto que os fluxos associados a feixes de radiação que atingem a superfície dA são negativos ($\cos\theta_i < 0$).

O fluxo de radiação que o volume de controle recebe devido aos feixes que o atingem, vindos de todas as direções, é:

$$q_{r,i} = \int_{4\pi} I(\vec{s}_i) \cos\theta_i d\Omega_i \quad (5.13)$$

Analogamente, o fluxo de radiação que o volume de controle perde, devido aos feixes que o abandonam por todas as direções, é:

$$q_{r,o} = \int_{4\pi} I(\vec{s}_o) \cos\theta_o d\Omega_o \quad (5.14)$$

O fluxo líquido de radiação pode ser calculado pela soma de ambas as parcelas:

$$q_r = q_{r,i} + q_{r,o} = \int_{4\pi} I(\vec{s}) \cos\theta d\Omega \quad (5.15)$$

Como $\cos\theta = \vec{n} \cdot \vec{s}$, a expressão anterior pode ser reescrita como:

$$q_r = \int_{4\pi} I(\vec{s}) \vec{n} \cdot \vec{s} d\Omega = \vec{n} \cdot \int_{4\pi} I(\vec{s}) \vec{s} d\Omega \quad (5.16)$$

Considere-se agora um volume de controle convexo qualquer envolvido por um número qualquer de faces de dimensões arbitrárias. A expressão (5.16) é válida para cada face do volume de controle.

Lembrando que $\vec{n} \cdot \vec{a} = \text{comp}_{\vec{n}} \vec{a}$ (isto é, o produto vetorial entre um vetor e um versor resulta no componente do vetor na direção do versor), q_r pode ser interpretado como a

parcela do fluxo térmico líquido que o volume troca com suas vizinhanças correspondente à face de normal emergente \vec{n} . Assim,

$$q_r = \vec{n} \cdot \vec{q}_r \quad (5.17)$$

Conclui-se que o vetor fluxo térmico devido à radiação é dado por:

$$\vec{q}_r = \int_{4\pi} I(\vec{s}) \vec{s} d\Omega \quad (5.18)$$

5.2. Modelagem da troca térmica radiativa

5.2.1. A equação de transferência radiativa

Para o cálculo de $\nabla \cdot \vec{q}_r$, deve-se partir da chamada equação de transferência radiativa.

Essa equação expressa o balanço de energia de um feixe de radiação que se desloca na direção \vec{s} ao atravessar um volume de controle de um meio material, considerando os efeitos de emissão, absorção e espalhamento de radiação.

Ela pode ser expressa como, conforme SIEGEL; HOWELL (1972):

$$\frac{dl}{ds} = \vec{s} \cdot \nabla I = -(a + \sigma_s)I + aI_b + \frac{\sigma_s}{4\pi} \int_{4\pi} I(\vec{s}_i) \phi(\vec{s}_i, \vec{s}) d\Omega, \quad (5.19)$$

onde

$\frac{dl}{ds}$ é a derivada direcional da intensidade de radiação na direção de propagação;

$(a + \sigma_s)I$ é a atenuação devida à absorção e ao espalhamento pelo meio;

$a I_b$ é a contribuição devida à emissão do meio

$\frac{\sigma_s}{4\pi} \int_{4\pi} I(\bar{s}_i) \phi(\bar{s}_i, \bar{s}) d\Omega_i$ é a contribuição devida ao espalhamento na direção \bar{s} , da radiação vinda de outras direções que incide no volume de controle.

A equação de transferência radiativa expressa o balanço de um único feixe de radiação; portanto, para obter-se o balanço do volume de controle, deve-se integrá-la sobre todos os ângulos sólidos, resultando:

$$\int_{4\pi} \bar{s} \cdot \nabla I d\Omega = - \int_{4\pi} (a + \sigma_s) I d\Omega + \int_{4\pi} a I_b d\Omega + \int_{4\pi} \frac{\sigma_s}{4\pi} \int_{4\pi} I(\bar{s}_i) \phi(\bar{s}_i, \bar{s}) d\Omega_i d\Omega \quad (5.20)$$

No termo do lado esquerdo da equação anterior, os operadores matemáticos podem ser intercambiados já que a direção \bar{s} e as coordenadas espaciais são independentes entre si.

$$\nabla \cdot \int_{4\pi} I \bar{s} d\Omega = - \int_{4\pi} (a + \sigma_s) I d\Omega + \int_{4\pi} a I_b d\Omega + \int_{4\pi} \frac{\sigma_s}{4\pi} \int_{4\pi} I(\bar{s}_i) \phi(\bar{s}_i, \bar{s}) d\Omega_i d\Omega \quad (5.21)$$

Combinando com a equação (5.18), pode-se concluir que:

$$\nabla \cdot \bar{q}_r = - \int_{4\pi} (a + \sigma_s) I d\Omega + \int_{4\pi} a I_b d\Omega + \int_{4\pi} \frac{\sigma_s}{4\pi} \int_{4\pi} I(\bar{s}_i) \phi(\bar{s}_i, \bar{s}) d\Omega_i d\Omega \quad (5.22)$$

Essa equação indica que, se as intensidades de radiação de todas as direções, assim como as propriedades do meio, forem conhecidas é possível calcular $\nabla \cdot \bar{q}_r$ em cada volume de controle para, a seguir, proceder-se ao cálculo da equação de energia.

Para o caso em estudo, a equação anterior pode ser simplificada admitindo que o meio emite e absorve mas não espalha radiação. Essa simplificação se justifica por ser desprezível a concentração volumétrica dos sólidos particulados no gás produto de combustão, principal agente espalhador de radiação em meios semelhantes. Assim, a equação fica.

$$\nabla \cdot \vec{q}_r = - \int_{4\pi} a I d\Omega + \int_{4\pi} a I_b d\Omega \quad (5.23)$$

5.2.2. O método das ordenadas discretas

Referindo-se à equação anterior, percebe-se, obviamente, que não há método que permita determinar as intensidades de radiação em todas as direções. Um problema análogo a este consiste em calcular a integral de uma função $F(x)$ num intervalo Δx . Para o cálculo dessa integral, numericamente igual à área sob a curva (Figura 5.3), não são calculados todos os valores de $F(x)$ no intervalo considerado.

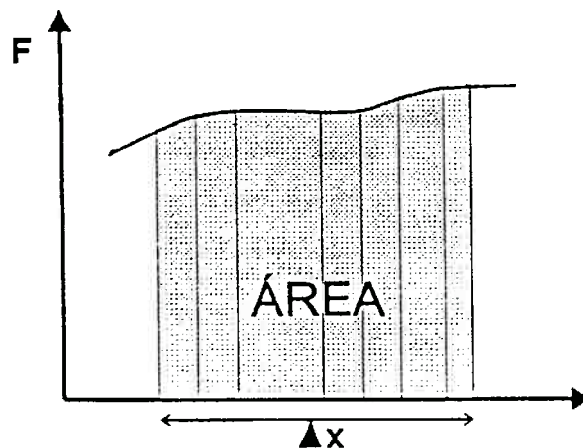


Figura 5.3. Desenho esquemático apresentando a área como o cálculo de integral.

Considere-se novamente a Figura 5.3. É simples perceber que:

$$\int_{\Delta x} F(x) dx \cong \sum_{m=1}^N F(x_m) \Delta x_m \quad (5.24)$$

ou, dito em palavras, a integral pode ser aproximada por uma quadratura numérica.

O método das ordenadas discretas baseia-se numa aproximação semelhante, ou seja:

$$\int_{4\pi} F(\vec{s}) d\Omega \cong \sum_{m=1}^N F(s_m) w_m \quad (5.25)$$

isto é, integrais sobre direções são substituídas por quadraturas numéricas, baseadas num número limitado de direções (N) com seus respectivos fatores de ponderação (w_m).

$$\text{Obviamente, } \sum_{m=1}^N w_m = 4\pi \quad (5.26)$$

Assim, a determinação de $\nabla \cdot \vec{q}_r$, pode ser obtida, partindo da equação (5.23). Considerando que a radiação emitida por um corpo negro é isotrópica, isto é, independe da direção de propagação, a equação mencionada pode ser reescrita como:

$$\nabla \cdot \vec{q}_r = \alpha \left(- \sum_{m=1}^N I_m w_m + 4\pi I_b \right) \quad (5.27)$$

isto é, a troca térmica líquida por radiação de um volume de controle pode ser calculada a partir das intensidades de radiação de um número limitado de direções.

O cálculo dessas intensidades nas direções escolhidas pode ser feito também com a utilização da equação de transferência radiativa, pois ela nada mais é do que uma equação diferencial em termos das coordenadas espaciais necessitando tão somente de condição(ões) de contorno(s) para a sua resolução.

Para um problema, em meio não espalhador de radiação, equacionado em coordenadas retangulares, a equação de transferência radiativa pode ser escrita, a partir da equação (5.19), como:

$$\vec{s}_m \cdot \nabla I_m = \xi_m \frac{\partial I_m}{\partial x} + \mu_m \frac{\partial I_m}{\partial y} + \eta_m \frac{\partial I_m}{\partial z} = -a I_m + a I_b \quad (5.28)$$

onde ξ_m , μ_m e η_m são os cossenos diretores da direção \vec{s}_m

Fazendo desenvolvimento análogo para casos equacionados em coordenadas ortogonais curvilíneas (cilíndricas e esféricas), observa-se que, para uma direção \vec{s}_m , seus cossenos diretores variam com a cota do ponto considerado nessa direção, diferentemente do que ocorre com o equacionamento em coordenadas ortogonais retangulares. Para o caso de geometria cilíndrica, essa dificuldade adicional pode ser solucionada através do procedimento apresentado a seguir.

Inicialmente, deve-se observar que a derivada direcional pode ser escrita como (com notação de acordo com a Figura 5.4)²:

$$\vec{s}_m \cdot \nabla I_m = \frac{\mu_m}{r} \frac{\partial}{\partial r} (r I_m) - \frac{1}{r} \frac{\partial}{\partial \psi} (\eta_m I_m) + \zeta_m \frac{\partial I_m}{\partial z} \quad (5.29)$$

onde $\mu_m = \text{sen } \theta \cos \psi$

$$\eta_m = \text{sen } \theta \text{ sen } \psi \quad (5.30)$$

$$\zeta_m = \cos \theta$$

Portanto, a equação de transferência radiativa pode ser escrita como:

$$\frac{\mu_m}{r} \frac{\partial}{\partial r} (r I_m) - \frac{1}{r} \frac{\partial}{\partial \psi} (\eta_m I_m) + \zeta_m \frac{\partial I_m}{\partial z} = -a I_m + a I_b \text{ para } m = 1, 2, \dots, N \quad (5.31)$$

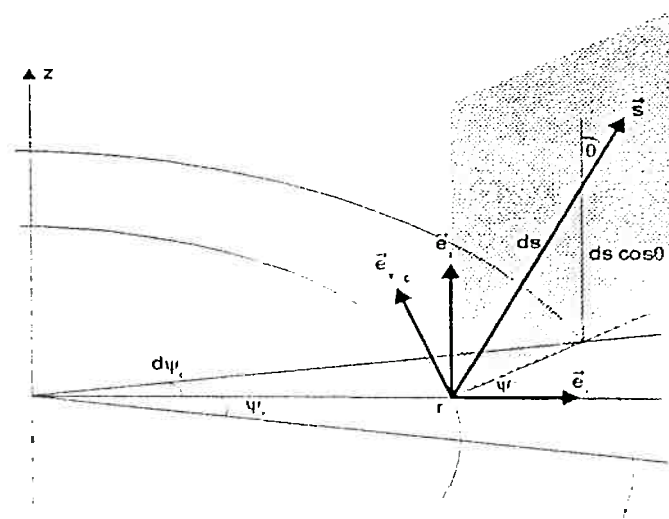


Figura 5.4. Esquema do sistema de coordenadas e ângulos locais na geometria cilíndrica.

² A demonstração da expressão (5.29) encontra-se no Anexo 1.

CARLSON; LATHROP (1968) propuseram substituir a derivada angular por uma aproximação de diferenças finitas:

$$\left[\frac{1}{r} \frac{\partial}{\partial \psi} (\eta_m I_m) \right]_{\psi=\psi_m} = \frac{1}{r} \frac{(\alpha_{m+1/2} I_{m+1/2} - \alpha_{m-1/2} I_{m-1/2})}{w_m} \quad (5.32)$$

onde as direções $(m + 1/2)$ e $(m - 1/2)$ definem as bordas do intervalo angular w_m e os coeficientes α são calculados de maneira a preservar a generalidade do esquema de diferenças finitas face ao campo de intensidades de radiação.

Assim, assumindo campo I_m uniforme, obtém-se a fórmula de recorrência:

$$\alpha_{m+1/2} - \alpha_{m-1/2} = w_m \left[\frac{\partial \eta}{\partial \psi} \right]_{\psi=\psi_m} = w_m \mu_m, \quad m=1,2..N \quad (5.33)$$

que pode ser usada para calcular $\alpha_{m+1/2}$ se um valor para $\alpha_{1/2}$ puder ser calculado.

Essa condição poder ser conseguida utilizando-se o seguinte artifício, lembrando que a expressão (5.28) representa a variação de uma componente de I_m devido à variação do ângulo ψ que ocorre numa propagação diferencial do feixe:

- para $m=1$, escolhe-se uma direção onde não haja variação desse ângulo, o que implica em, conforme (5.28):

$$\alpha_{1/2} = \alpha_{3/2} = 0 \quad (5.34)$$

- para compatibilizar-se com a fórmula de recorrência, essa direção não deve fazer parte do conjunto de direções que representa o ângulo sólido de 4π estéreo-radianos, o que implica em:

$$w_1 = 0 \quad (5.35)$$

A escolha da direção $\mu_1 = -1$, com $w_1 = 0$ atende os requisitos, permitindo a utilização da fórmula de recorrência com os valores iniciais mencionados.

Combinando as expressões anteriores, obtém-se a forma final da equação de transferência radiativa em coordenadas cilíndricas:

$$\frac{\mu_m}{r} \frac{\partial}{\partial r} (r I_m) - \frac{1}{r} \frac{(\alpha_{m-1/2} I_{m-1/2} - \alpha_{m+1/2} I_{m+1/2})}{W_m} + \zeta_m \frac{\partial I_m}{\partial z} = -a I_m + a I_b \quad (5.36)$$

Essa é uma equação diferencial que deve ser discretizada, isto é, transformada para uma equação algébrica válida para os pontos nodais do domínio. Para tanto, pode-se aplicar a técnica dos volumes finitos.

Como ilustração, apresenta-se a transformação do primeiro termo da equação. Integrando-o em $dV' = r d\psi_c dr dz$ obtém-se:

$$\int_z^{z+dz} \int_{\psi_c}^{\psi_c+d\psi_c} \int_{r_s}^{r_n} \frac{\mu_m}{r} \frac{\partial}{\partial r} (r I_m) r dr d\psi_c dz = \mu_m \int_z^{z+dz} \int_{\psi_c}^{\psi_c+d\psi_c} \left[\int_{r_s}^{r_n} \frac{\partial}{\partial r} (r I_m) dr \right] d\psi_c dz =$$

$$\mu_m \int_z^{z+dz} \int_{\psi_c}^{\psi_c+d\psi_c} (r_n I_{m,n} - r_s I_{m,s}) d\psi_c dz = \mu_m (A_n I_{m,n} - A_s I_{m,s}) \quad (5.37)$$

onde: $A_{r,n}, A_{r,s}$ são as áreas das faces do volume perpendiculares à direção r , ao norte e ao sul respectivamente.

$I_{m,n}, I_{m,s}$ são as intensidades de radiação nas faces do volume de controle, faces norte e sul respectivamente.

Aplicando essa técnica a toda a equação (5.36), obtém-se:

$$\mu_m (A_{r,n} I_{m,n} - A_{r,s} I_{m,s}) - (A_{r,n} - A_{r,s}) \frac{\alpha_{m+1/2} I_{m+1/2} - \alpha_{m-1/2} I_{m-1/2}}{W_m} +$$

$$+ \zeta_m (A_{z,e} I_{m,e} - A_{z,w} I_{m,w}) = -a I_{m,p} + a I_b \quad (5.38)$$

onde: $A_{z,e}, A_{z,w}$ são as áreas das faces do volume perpendiculares à direção z , a este e a oeste respectivamente.

$I_{m,e}, I_{m,w}$ são as intensidades de radiação nas faces do volume de controle, faces este e oeste respectivamente.

$I_{m,p}$ é a intensidade de radiação no volume de controle considerado, na direção m .

Essa equação será utilizada para calcular a intensidade de radiação de determinada direção no volume de controle ($I_{m,p}$) a partir dos valores das intensidades em duas das suas interfaces.

Ela assumirá desenvolvimentos algébricos diferentes considerando:

a) a função de interpolação da intensidade de radiação

A equação anterior deve ser simplificada pela adoção de uma função de interpolação que associe o valor da variável no ponto nodal com seus valores nas duas interfaces adjacentes numa direção coordenada.

Para efeito de discussão, considerem-se as funções de interpolação:

- Diferença central

$$\text{Por exemplo: } I_{m,p} = \frac{1}{2}(I_{m,n} + I_{m,s}) \quad (5.39)$$

- “Upwind”

Por exemplo, quando o feixe de radiação desloca-se, na direção Norte-Sul, do Sul para o

$$\text{Norte: } I_{m,p} = I_{m,s} \quad (5.40)$$

A interpolação por diferença central, apesar de produzir a aproximação de uma derivada com menor erro de truncamento, não é a escolha mais adequada para muitos escoamentos. Muitas vezes, a adoção desse esquema acarreta a instabilidade do procedimento de cálculo ou a sua convergência em solução fisicamente inconsistente.

O uso da função “upwind” (quando a aproximação adota o valor da variável de um lado só) resolve o problema das instabilidades mas acarreta o problema conhecido como “difusão numérica”, quando a solução numérica atenua os gradientes internos do problema real.

A escolha entre essas opções deve considerar não apenas as dificuldades numéricas mas também as características do problema real. Em escoamento de fluidos, a adoção do esquema “upwind” tem sentido quando o efeito convectivo sobrepuja o difusivo; em caso contrário, a interpolação por diferença central é a escolha mais adequada.

No caso da radiação, a discussão é absolutamente semelhante.

A adoção da diferença central pode levar a instabilidades, muitas vezes impossibilitando a convergência numa solução, por acarretar valores negativos para as intensidades de radiação nos cálculos iterativos (que não têm significado físico, diga-se de passagem).

A adoção da função “upwind” garante que não haja valores negativos para a intensidade de radiação nos cálculos iterativos.

Sob a ótica da realidade física, o esquema “upwind” para a interpolação da intensidade de radiação tem sentido para meios cujos coeficientes de extinção (absorção mais espalhamento) forem baixos; isto é, para meios pouco participativos. Em caso contrário, isto é, para meios de altos valores para o coeficientes de extinção, o mais adequado é a utilização da interpolação por diferença central.

O programa desenvolvido possibilita o emprego das seguintes opções: “upwind” e “diferença ponderada”. Esta última apresenta uma função do seguinte tipo:

$$I_{m,p} = \gamma I_{m,n} + (1 - \gamma) I_{m,s} \quad \text{com } 0 < \gamma < 1 \quad (5.41)$$

O esquema da diferença central está incorporado à expressão anterior como um caso particular ($\gamma=0.5$)

b) direção e sentido de deslocamento do feixe de radiação

Como a equação final é uma equação diferencial parcial de primeira ordem em termos das duas coordenadas espaciais, basta conhecer-se o valor da intensidade de radiação em duas faces para se determinar o seu valor no ponto nodal. A escolha dessas duas faces dependerá da direção e sentido de deslocamento do feixe de radiação.

Por exemplo, para feixes de radiação deslocando-se no sentido Sudoeste para Nordeste, a aplicação das condições de contorno das fronteiras Sul e Oeste do domínio permitirá

calcular, em primeiro lugar, a intensidade de radiação do elemento de volume do canto sudoeste (“inferior esquerdo”) e, em seguida, as intensidades de radiação nas faces Norte e Este desse volume são determinadas, vide Figura 5.5. Feito isso, pode-se determinar as intensidades de radiação nos pontos nodais adjacentes a esse volume de controle. Prosseguindo-se com esse esquema as intensidades de radiação de todos os volumes elementares do domínio são calculados.

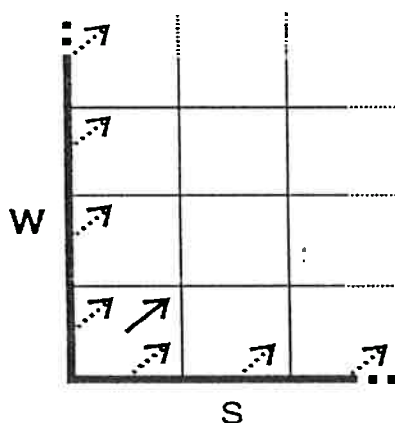


Figura 5.5. Esquema apresentando os contornos Sul e Oeste do domínio.

Desenvolvimentos semelhantes aplicam-se a feixes de radiação que se deslocam nas demais direções.

Observe-se, portanto, que diferentes condições de contorno são aplicadas para diferentes feixes de radiação dependendo da direção e sentido em que se deslocam e que, para contemplar um conjunto de direções representativo de todas as direções, condições de contorno em todas as fronteiras devem ser especificadas.

Para finalizar a exposição deste tópico, apenas a título de exemplo, apresenta-se a simplificação da equação (5.38) advinda das seguintes considerações:

- direção de deslocamento do feixe de radiação: de Sudoeste para Nordeste.
- função de interpolação: diferença ponderada.

$$I_{m,p} = \gamma I_{m-1/2} + (1 - \gamma) I_{m-1/2} = \gamma I_{m,n} + (1 - \gamma) I_{m,s} = \gamma I_{m,c} + (1 - \gamma) I_{m,w} \quad (5.42)$$

chega-se a:

$$I_{m,p} = \frac{\mu_m A'_r I_{m,s} + \zeta_m A'_z I_{m,w} - \Pi \alpha'_m I_{m-1/2} + \gamma a I_b V_p}{\mu_m A_{r,n} + \zeta_m A_{z,e} - \Pi \alpha_{m-1/2} + \gamma a V_p} \quad (5.43)$$

onde $A'_r = (1 - \gamma) A_{r,n} + \gamma A_{r,s}$

$$A'_z = A_{z,e} = A_{z,w} \quad (5.44)$$

$$\alpha'_m = (1 - \gamma) \alpha_{m+1/2} + \gamma \alpha_{m-1/2}$$

$$\Pi = \frac{A_{r,n} - A_{r,s}}{w_m}$$

Assim, através dessa equação (e outras semelhantes, elaboradas considerando as outras direções de deslocamento) pode-se calcular as intensidades de radiação e, em seguida, através da equação (5.23), calcular o divergente do vetor fluxo térmico por radiação, o que permite resolver a equação de conservação de energia em todos os volumes elementares do domínio.

Para a interpretação da solução de problema que envolve radiação, é conveniente, apresentar os campos dos fluxos de potência radiativa nas direções coordenadas ao invés dos campos de intensidades de radiação.

Considere-se o fluxo de potência radiativa que incide num ponto nodal vindo dos feixes que possuam componente no sentido positivo do eixo \vec{e}_r . Esses feixes possuem o coseno diretor μ positivo. Esse fluxo pode ser calculado como:

$$q_r^+ = \sum_m w_m \mu_m I_m, \text{ para todo } m \text{ tal que } \mu_m > 0 \quad (5.45)$$

O fluxo de potência radiativa que incide no ponto nodal vindo dos feixes que possuam componente no sentido negativo do eixo \vec{e}_r , analogamente, pode ser calculado como:

$$q_r^- = \sum_m w_m \mu_m I_m, \text{ para todo } m \text{ tal que } \mu_m < 0 \quad (5.46)$$

Os fluxos de potência radiativa nos dois sentidos do eixo \bar{e}_z possuem expressões análogas:

$$q_z^+ = \sum_{m_i} w_m \zeta_m I_m, \text{ para todo } m \text{ tal que } \zeta_m > 0 \quad (5.47)$$

$$q_z^- = \sum_m w_m \zeta_m I_m, \text{ para todo } m \text{ tal que } \zeta_m < 0 \quad (5.48)$$

5.2.3. Seleção do número e das direções das ordenadas discretas

Embora a escolha do esquema da quadratura numérica seja arbitrária, foram feitas as seguintes considerações:

Quanto ao número de direções

O método das ordenadas discretas também é chamado de “método S_N ” ou “aproximação S_N ”, onde N indica o número de valores diferentes que cada coseno diretor assume. Por exemplo, na denominada “aproximação S_2 simétrica”, cada coseno diretor pode assumir os valores 0.5773503 e -0.5773503.

Pode-se calcular o número de direções (n) que devem ser consideradas a partir da ordem N através da seguinte expressão:

$$n = N(N + 2) \quad (5.49)$$

embora, várias direções podem ser desnecessárias para problemas uni e bidimensional em razão de simetria.

JAMALUDDIN; SMITH (1988) constataram que aproximações de ordem 2 ou 4 fornecem resultados próximos, desaconselhando a utilização de aproximações de maior ordem.

Neste trabalho, adotou-se a aproximação de ordem 4, o que significa a representação do ângulo sólido de 4π estéreos-radianos através de 24 direções, sendo necessário, no entanto, considerar-se apenas 12 em razão da simetria axial do problema.

Quanto à seleção das direções

Podem ser encontradas na literatura vários esquemas de quadratura. Neste trabalho, adotou-se as direções mencionadas por LATHROP; CARLSON (1968).

5.3. Condições de contorno para a simulação do trocador de calor

5.3.1. Para a equação da intensidade de radiação

Para ambos os problemas, dos escoamentos de ar na coroa circular e do gás produto de combustão no duto interno, deve-se resolver a equação diferencial da intensidade de radiação de cada direção ordenada.

Considere-se um único feixe de radiação para efeito de raciocínio.

A condição de contorno determina o valor da intensidade de cada feixe de radiação no ponto do contorno pelo qual essa radiação entra no domínio do problema.

Se o ponto pelo qual esse feixe entra no domínio do problema pertence a um contorno sólido, a determinação do valor de sua intensidade se faz computando a soma de duas parcelas: a intensidade da radiação emitida e a da radiação refletida pelo ponto. O contorno sólido constitui-se como uma barreira à radiação proveniente do meio externo ao domínio.

Tal não é o caso se o ponto pertencer a um contorno gasoso. Nesse caso, deve-se considerar a radiação proveniente do meio externo que entra no domínio por esse ponto.

Assim, observando o desenho esquemático do trocador de calor, novamente apresentado na Figura 5.6, e considerando que o domínio se reduz à metade em razão da simetria

axial do fenômeno, pode-se distinguir as condições de contorno para a equação da intensidade de radiação para ambos os problemas entre as seguintes categorias, definidas em função do contorno pelo qual a radiação entra no domínio:

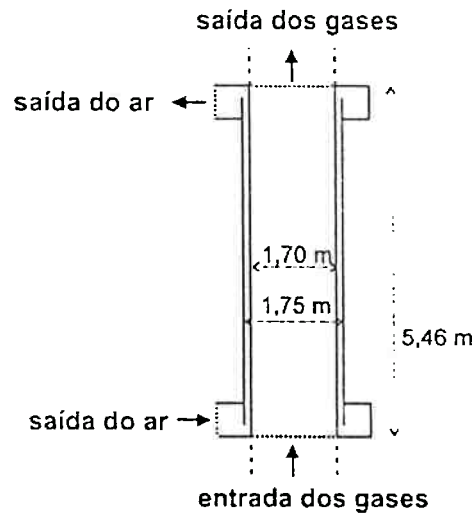


Figura 5.6. Desenho esquemático do trocador de calor.

a) contorno sólido

$$I_m = \varepsilon I_b + \frac{\rho q'}{\pi} \quad (5.50)$$

onde εI_b é a intensidade da radiação emitida

$\frac{\rho q'}{\pi}$ é a intensidade da radiação refletida (q' é o fluxo de radiação incidente no ponto vindo de todas as direções de um semi espaço)

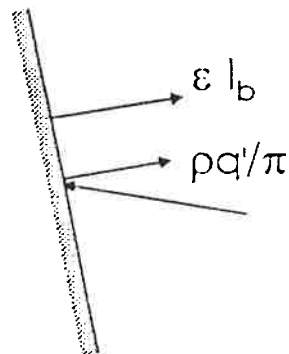


Figura 5.7. Esquema apresentando as parcelas da intensidade de radiação emergente de uma superfície sólida.

A expressão (5.50) é a forma abrangente de equacionar a intensidade emitida por uma superfície. Ela se simplifica se admitir-se que a superfície seja negra, como é o caso das paredes do compartimento do gás. Essa hipótese baseia-se no fato de que tais superfícies, usualmente, encontram-se recobertas por uma camada de fuligem. Assim, para essas superfícies, vale a seguinte expressão:

$$I_m = I_b \quad (5.51)$$

b) contorno gasoso correspondente ao eixo de simetria axial

Esse caso é verificado no escoamento do gás no duto interno do recuperador de calor.

A hipótese da simetria axial indica que se um feixe de radiação é emitido num ponto qualquer da superfície interna do duto (ponto A) em direção a um ponto qualquer situado no eixo de simetria (ponto B), um outro feixe de igual intensidade também será emitido, partindo do ponto A' (situado na mesma seção transversal do ponto A, porém em posição diametralmente oposta) em direção ao ponto B.

Ambos os feixes sofrem igual variação no percurso (atenuação devida à absorção e incremento devido à emissão do meio) e, por consequência, possuirão a mesma intensidade ao atingir o eixo de simetria (Figura 5.8).

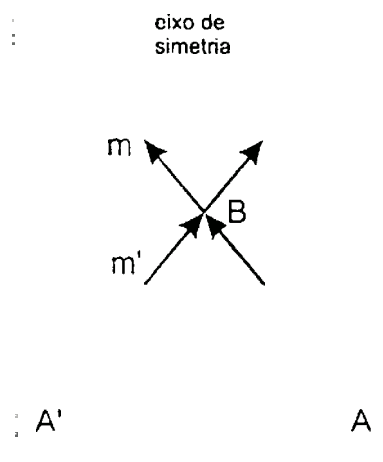


Figura 5.8. Desenho esquemático de dois feixes de radiação simétricos.

Matematicamente, essa condição pode ser expressa como:

$$I_m = I_{m'} \quad (5.52)$$

onde m e m' são direções tais que:

$$\mu_m = -\mu_{m'} \quad (5.53)$$

$$\zeta_m = \zeta_{m'}$$

onde μ e ζ são os cossenos diretores de uma direção em relação a \vec{e}_r e \vec{e}_z respectivamente.

c) contorno gasoso correspondente às seções de entrada dos fluidos

- Para o escoamento do gás produto de combustão no duto interno:

Pode-se supor que a radiação que entra no domínio, por esse contorno, é proveniente de uma cavidade negra com temperatura igual à dos gases.

Essa hipótese se justifica pela consideração de que a relação (diâmetro do duto interno do recuperador dividido pelo diâmetro equivalente do volume do forno siderúrgico) é pequena. Neste caso, a seção de entrada dos gases no recuperador de calor comporta-se como uma superfície negra cuja temperatura é a temperatura da superfície interna das paredes do forno.

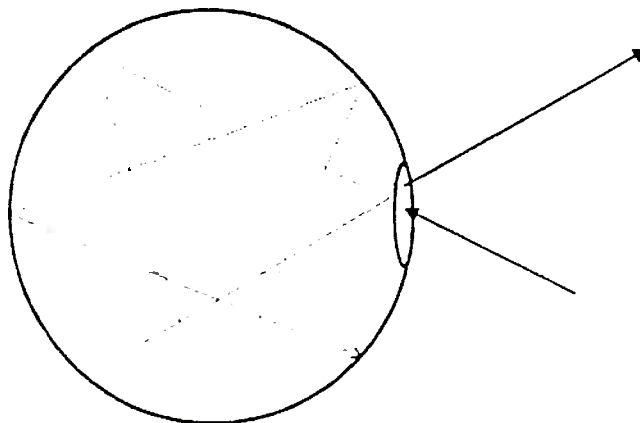


Figura 5.9. Desenho esquemático de um orifício de uma cavidade esférica.

Observando a Figura 5.9, pode-se perceber que o orifício é um “absorvedor perfeito” porque praticamente toda a radiação vinda do meio externo, que incide nele, é absorvida pelas paredes internas da cavidade esférica; pode-se demonstrar que o orifício também é um “emissor perfeito”³ (apresentando assim as duas propriedades de um corpo negro). Esse comportamento do orifício de uma cavidade esférica é o que justifica a utilização de fornos esféricos para calibração de medidores.

Assim:

$$I_m = I_b \quad (5.54)$$

- Para o escoamento do ar na coroa circular:

Pode-se supor que a radiação que entra no domínio por esse contorno é proveniente de uma cavidade negra que se encontra à temperatura de entrada do ar (temperatura ambiente).

Essa hipótese se justifica novamente pela relação (diâmetro equivalente do canal de ar)/(diâmetro equivalente do plenum para distribuição do ar), inferior àquela que justifica o comportamento da cavidade como uma superfície negra, conforme pode ser observado na Figura 5.10.

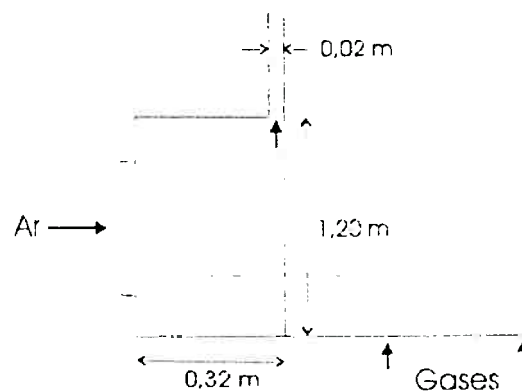


Figura 5.10. Dimensões do plenum e do conduto de ar

³ Conforme PIKMAN (1987).

d) contorno gasoso correspondente às seções de saída dos fluidos

- Para o escoamento do gás produto de combustão no duto interno:

Pode-se supor que a radiação que entra no domínio por esse contorno é proveniente de superfícies sólidas negras que se encontram à temperatura de saída do gás.

Essa hipótese se justifica pelo fato do recuperador em estudo ser o assim chamado recuperador inferior e, colocado a jusante dele, situar-se o recuperador superior. Este, também um pré-aquecedor de ar, é um recuperador “de convecção”, com tubos de ar imersos no escoamento do gás.

A seção inferior do recuperador “de convecção”, em particular, em função das tomadas de saída do ar e das junções dos tubos de ar, possui considerável parte da sua seção transversal preenchida pela superfícies dos tubos, cujas temperaturas externas encontram-se bastante próximas à temperatura do gás nessa seção. Esse fato justifica que a radiação emergente dessa seção seja proveniente de superfícies sólidas.

Essas superfícies podem ser caracterizadas como negras porque, regra geral, estão impregnadas de fuligem.

- Para o escoamento do ar na coroa circular:

Pode-se supor que a radiação que entra ao domínio por esse contorno é proveniente de uma cavidade negra que se encontra à temperatura de saída do ar.

Essa hipótese se justifica novamente pela relação (diâmetro equivalente do canal de ar)/(diâmetro equivalente do plenum de captação do ar). Esse plenum tem as mesmas dimensões que o de distribuição, esquematicamente apresentado na Figura 5.10.

5.3.2. Para a equação de energia

Considerando-se ambos os problemas, dos escoamentos de ar e do gás, as condições de contorno para a equação de energia são:

a) seção de entrada do fluido

A temperatura nessa seção é a temperatura de entrada do fluido.

$$T = T_0 \quad (5.55)$$

b) contorno correspondente às paredes dos tubos

Em contornos dessa natureza, dois casos ocorrem:

- parede isolada: derivada radial da temperatura nula

$$\frac{\partial T}{\partial r} = 0 \quad (5.56)$$

É o caso do tubo externo que confina o escoamento do ar na coroa circular.

- temperatura pré-definida

$$T = T_D \quad (5.57)$$

É o caso do tubo que separa o ar do gás. Deve-se entender “pré-definida” como um valor assumido para rodar o programa, valor que será reiterado até sua convergência (ver item 7.1).

c) seção de saída do fluido

Nenhuma condição de contorno foi adotada.

6. Validação do método das ordenadas discretas

Antes de adaptar uma subrotina de cálculo das trocas de calor radiativas segundo o método das ordenadas discretas num programa CFD, foi executada uma etapa de validação do método com os objetivos de verificar os resultados que o método determina frente a resultados teóricos e adquirir alguma experiência na execução do algoritmo de cálculo.

Foi selecionado um caso que tem solução apresentada na literatura, envolvendo exclusivamente trocas de calor por radiação, isto é, sem a interferência de condução e/ou convecção.

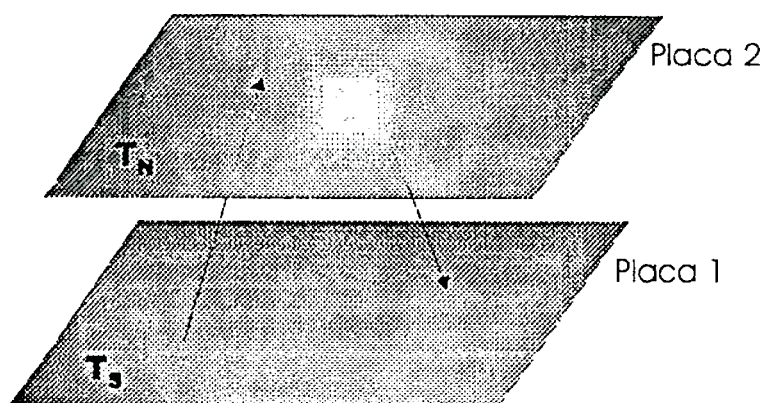


Figura 6.1. Radiação entre duas placas.

O problema escolhido foi o de determinar o perfil de temperatura do regime estacionário (isto é, em equilíbrio radiativo) entre duas placas paralelas, negras e infinitas a temperaturas diferentes e constantes, separadas por um gás cinzento, não espalhador de radiação, de coeficiente de absorção constante.¹

Nesse caso, o equacionamento segundo o método das ordenadas discretas possibilita pequenas simplificações em relação ao equacionamento desenvolvido no item 5.2.2.

¹ SIEGEL; HOWELL (1972), item 14-6.2, página 446.

Partindo-se da equação (5.28), admitido a hipótese do gás ser cinzento e não espalhador tem-se:

$$\frac{dI_m}{ds} = \bar{s} \cdot \nabla I_m = aI_b - aI_m \quad (6.1)$$

onde $I_b = \frac{n^2 \sigma T^4}{\pi}$ (6.2)

O problema é unidimensional, isto é, as variações ocorrem apenas em z , direção perpendicular entre as placas. Chamando de $d\tau = a dz$ e lembrando que μ é o coseno diretor da direção \bar{s} em relação a \bar{e}_z , pode-se reescrever a equação anterior como:

$$\mu_m \frac{dI_m}{d\tau} + I_m = I_b \quad (6.3)$$

que, por conveniência, pode ser separada nas seguintes equações:

$$\mu'_m \frac{dI_m^+}{d\tau} + I_m^+ = I_b \quad (6.4)$$

$$-\mu'_m \frac{dI_m^-}{d\tau} + I_m^- = I_b \quad (6.5)$$

A expressão (6.4) é válida para os feixes de radiação que se deslocam da placa inferior em direção à placa superior, isto é, conforme o sentido positivo de z (nesse caso, $\mu' = \mu > 0$); enquanto que a expressão (6.5) vale para os feixes de radiação que partem da placa superior em direção à placa inferior, isto é, que se deslocam conforme o sentido negativo de z (nesse caso, $\mu' = -\mu > 0$).

O objetivo dessa notação é obter alguma simplificação no algoritmo de cálculo, derivada da simetria existente entre as direções que são escolhidas para integrar o método das ordenadas discretas. Assim, para uma direção de coseno diretor μ , haverá outra cujo coseno diretor será $-\mu$, e ambas possuem o mesmo fator de ponderação w .

As equações (6.4) e (6.5) são equações diferenciais ordinárias de primeira ordem e, portanto, possuem como solução:

$$I_m^- = I_b + C^- e^{-\tau \mu_m} \quad (6.6)$$

$$I_m^+ = I_b + C^+ e^{-\tau \mu_m} \quad (6.7)$$

onde as constantes de integração C^+ e C^- podem ser calculadas a partir das condições de contorno. Essas condições estabelecem:

- junto à placa Sul

$$\tau = 0 \Rightarrow I^+ = I_{b,S} \quad (6.8)$$

- junto à placa Norte

$$\tau = \tau_L \Rightarrow I^- = I_{b,N} \quad (6.9)$$

onde
$$I_{b,S} = \frac{n^2 \sigma T_S^4}{\pi} \quad (6.10)$$

$$I_{b,N} = \frac{n^2 \sigma T_N^4}{\pi} \quad (6.11)$$

Assim, chega-se às seguintes equações:

$$I_m^- = I_b + (I_{b,S} - I_b) e^{-\tau \mu_m} \quad (6.12)$$

$$I_m^+ = I_b + (I_{b,N} - I_b) e^{-(\tau_L - \tau) \mu_m} \quad (6.13)$$

As equações (6.12) e (6.13) calculam as intensidades de radiação em qualquer ponto entre as placas para todas as direções selecionadas.

A temperatura num determinado ponto pode ser calculada a partir da hipótese do regime estar estacionário, isto é, em equilíbrio radiativo. Nessa condição, a equação de energia se simplifica para:

$$\nabla \cdot \vec{q}_r = 0 \quad (6.14)$$

Combinando com a equação (5.27) vem:

$$\nabla \cdot q_r = a \left(- \int_{4\pi} I d\Omega + 4\pi I_b \right) = 0 \quad (6.15)$$

o que implica que o termo entre parênteses é nulo.

Pelo método das ordenadas discretas, essa integral pode ser aproximada como:

$$\int_{4\pi} I d\Omega = \sum_{m=1}^N I_m w_m \quad (6.16)$$

ou fazendo uso da notação mencionada anteriormente:

$$\int_{4\pi} I d\Omega = \sum_{m=1}^{N/2} (I_m^+ + I_m^-) w_m \quad (6.17)$$

Dessa maneira:

$$- \sum_{m=1}^{N/2} (I_m^+ + I_m^-) w_m + 4\pi I_b = 0 \quad (6.18)$$

Combinando com a equação (6.2) resulta:

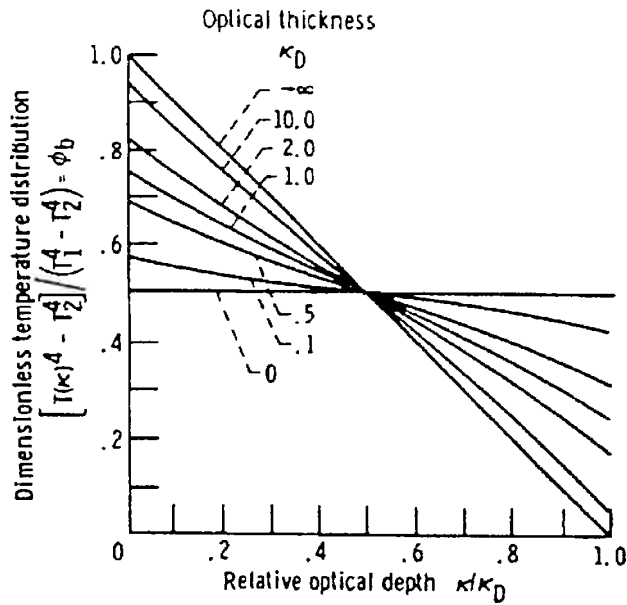
$$T = \left(\frac{1}{4n^2 \sigma} \sum_{m=1}^{N/2} (I_m^+ + I_m^-) w_m \right)^{1/4} \quad (6.19)$$

Essa equação relaciona a temperatura do volume de controle com as intensidades de radiação que nele incidem advindo de todas as direções selecionadas.

Dessa maneira, esse problema foi resolvido e sua solução² pôde ser comparada com a solução fornecida por SIEGEL; HOWELL (1972). Essa comparação pode ser observada na Figura 6.2:

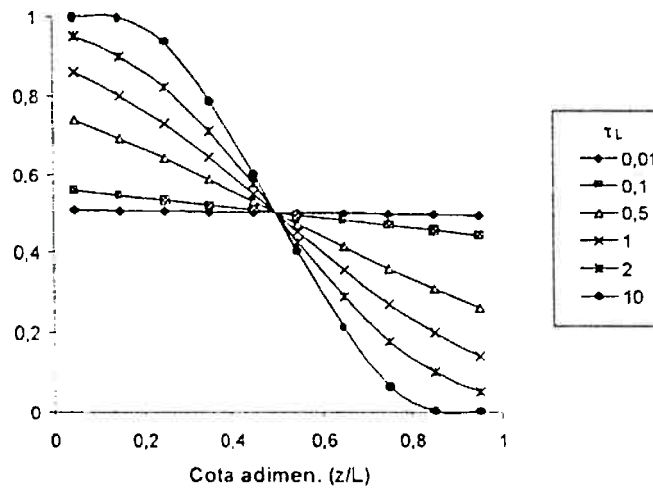
² Foi utilizada a aproximação de ordem 4, S₄.

Em primeiro lugar, observa-se que os resultados da aplicação do método das ordenadas discretas reproduzem as descontinuidades entre as temperaturas das placas e as temperaturas do meio próximas às placas.



(a)

Perfis do grupo adimen. da Temperatura



(b)

Figura 6.2. Comparação dos resultados³: (a) SIEGEL; HOWELL; (b) S₄.

³ Os perfis estão em função da espessura ótica, que pode ser caracterizada como o produto entre o coeficiente de extinção (admitido constante) e a distância característica percorrida pelos feixes de radiação.

Para o caso limite em que a espessura ótica fosse zero, o perfil de temperaturas pode não ser uma reta horizontal; a rigor, qualquer perfil seria aceitável, na medida em que o meio não interage com a radiação das placas.

Entretanto, verificam-se desvios entre as curvas da solução analítica e as obtidas pela aplicação do método S_4 para os casos onde os meios são opticamente densos, isto é, que possuem coeficientes de extinção⁴ altos. Nesses casos, o perfil correto aproxima-se do linear.

Para meios de coeficientes de extinção baixos, a aplicação da aproximação S_4 apresenta resultados bastante próximos dos valores teóricos.]

A grande maioria dos problemas na área de combustão relacionam-se com meios não opticamente densos. Por exemplo, problemas relacionados com chamas, fornos, caldeiras, trocadores de calor etc.

Meios opticamente densos são encontrados apenas em casos especiais envolvendo combustão em meios porosos e em leitos fluidizados. Para esses casos, o equacionamento das trocas de calor não representa nenhum grande problema, já que os feixes radiativos são trocados entre volumes muito próximos, permitindo simplificar a equação de transferência radiativa para a forma de uma equação de difusão. Para essa equação, a utilização de métodos matemáticos bem simples é capaz de fornecer bons resultados.

Assim, o domínio do método das ordenadas discretas permite tratar a grande maioria dos problemas da área de combustão industrial.

No caso do trocador de calor, os gases de combustão apresentam espessura ótica da ordem de 0,4 m. Nesse caso, portanto, o método das ordenadas discretas deve fornecer resultados bastante satisfatórios.

⁴ Coeficiente de extinção é o resultado da soma entre os coeficientes de absorção e espalhamento.

7. Resultados da simulação do recuperador de calor

7.1. Perfis de temperaturas no tubo que separa o gás do ar

Como mencionado anteriormente no capítulo 2, a simulação do recuperador de calor foi executada através do processamento de dois programas computacionais:

O primeiro programa calcula todas as variáveis envolvidas no escoamento do gás, sendo inicialmente admitido um perfil de temperaturas para a face interna do tubo que o separa do ar.

O segundo programa calcula todas as variáveis envolvidas no escoamento do ar, sendo inicialmente admitido um perfil de temperaturas para a face externa do tubo que o separa do gás.

A hipótese do regime estacionário, permitiu realizar o acoplamento necessário entre os dois programas para simular o recuperador de calor. Esse acoplamento baseou-se no equilíbrio dos fluxos térmicos líquidos em cada ponto do tubo de separação dos fluidos, isto é, no fato de que devem ser iguais os seguintes perfis:

- do fluxo térmico líquido transferido do gás para a face interna do tubo de separação;
- do fluxo de calor transferido por condução entre as faces interna e externa do tubo;
- do fluxo térmico líquido transferido da face externa do tubo de separação para o ar e para o tubo que confina o ar externamente.

Com a notação apresentada na Figura 7.1, as igualdades mencionadas podem ser expressas como:

$$q_c = q_{t,x} - q_{c,x} + q_{h,x} \quad (7.1)$$

¹ Como comentado anteriormente, a superfície da parede interna do tubo de separação foi admitida como negra (coberta por camada de fuligem).

$$q_c = k \cdot \nabla T \quad (7.2)$$

$$q_c = q_{r,a} - q_{ref} - q_{e,a} - q_{h,a} \quad (7.3)$$

onde q_c é o fluxo de calor por condução através da parede

q_r é o fluxo de radiação que incide na parede

$q_{ref} = \rho q_r$ é o fluxo de radiação refletido

q_e é o fluxo de radiação emitido

q_h é o fluxo térmico trocado, exceto a parcela radiativa, entre o fluido e a parede

ρ é a refletividade da parede

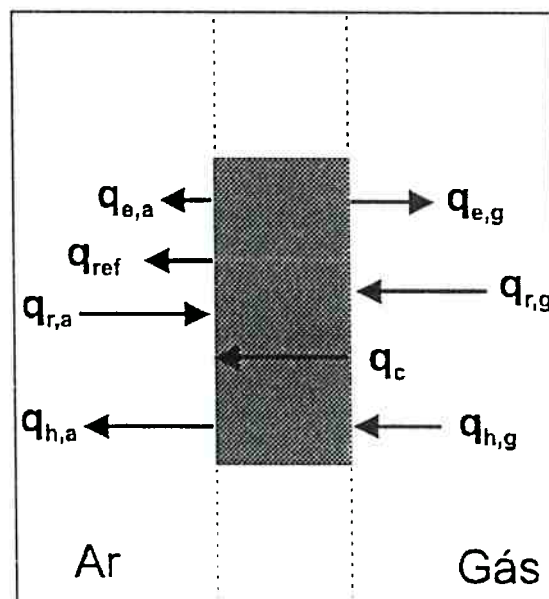


Figura 7.1. Esquema de um elemento da parede apresentando os fluxos térmicos relacionados a ele.

Estabeleceu-se o seguinte procedimento para o processamento dos programas computacionais desenvolvidos:

1. admite-se um perfil de temperaturas para a face interna do tubo de separação;

2. processa-se o programa relativo ao escoamento do gás, quando se calculam os fluxos térmicos líquidos que o gás transfere ao tubo;
3. calcula-se o perfil de temperaturas para a face externa do tubo, a partir dos perfis das temperaturas da face interna e dos fluxos líquidos transferidos calculados no item anterior;
4. processa-se o programa relativo ao escoamento do ar, a partir do perfil das temperaturas da face externa do tubo, quando se calcula os fluxos térmicos líquidos transferidos do tubo para o ar e para o tubo que confina o ar externamente;
5. obtém-se um perfil calculado de temperaturas para a face interna do tubo de separação a partir dos perfis das temperaturas da face externa do tubo e dos fluxos calculados no item anterior;
6. combina-se os perfis calculado e admitido de temperaturas da face interna do tubo de separação, obtendo-se um novo perfil admitido e repete-se o ciclo de processamento a partir do item (2) até a convergência dos perfis de fluxos térmicos calculados pelos dois programas.

Com base nesse procedimento, definiu-se os seguintes passos para convergir o perfil de temperaturas da face interna do tubo de separação:

- para dar início ao procedimento descrito, foi selecionado o perfil linear de temperaturas que apresentasse o menor “*desvio médio*” calculado como:

$$\text{desvio médio} = \frac{\sum_{j=1}^{N_z} \frac{|T_j^a - T_j^c|}{T_j^a} \Delta z_j}{\sum_{j=1}^{N_z} \Delta z_j} \quad (7.4)$$

onde T_j^a e T_j^c são os valores admitido e calculado da temperatura da superfície interna do tubo de separação no ponto j

Δz_j é o comprimento do lado no eixo $\frac{y}{z}$ do volume de controle

- a partir dos perfis calculado e admitido, a obtenção de um novo perfil admitido obedeceu à seguinte expressão:

$$T_{j,k+1}^a = T_{j,k}^a + a(T_{j,k}^a - T_{j,k}^c) \quad (7.5)$$

onde os índices subscritos “ k ” e “ $k+1$ ” referem-se à ordem da iteração e a é um fator entre 1 e 4.

- considerou-se que os perfis de fluxos térmicos calculados pelos dois programas convergiram quando a diferença entre seus valores tornou-se menor que 8 %, em qualquer cota (como o fluxo térmico é função da temperatura elevada à quarta potência, tal variação no fluxo leva a correções na temperatura da parede menores que 0,02 %, em cada cota).

A Figura 7.2 apresenta, como ilustração, alguns perfis de temperaturas da face interna do tubo de separação empregados no procedimento da sua convergência. A cota 0 corresponde à seção de entrada dos gases, enquanto que a cota 5,5 m corresponde à de saída. A legenda do gráfico apresenta a ordem de obtenção dos perfis, assim, a curva n° 1 é o perfil linear utilizado para dar início ao procedimento, as curvas n°s 20 e 30 são perfis intermediários e a curva n° 59 é o perfil final.

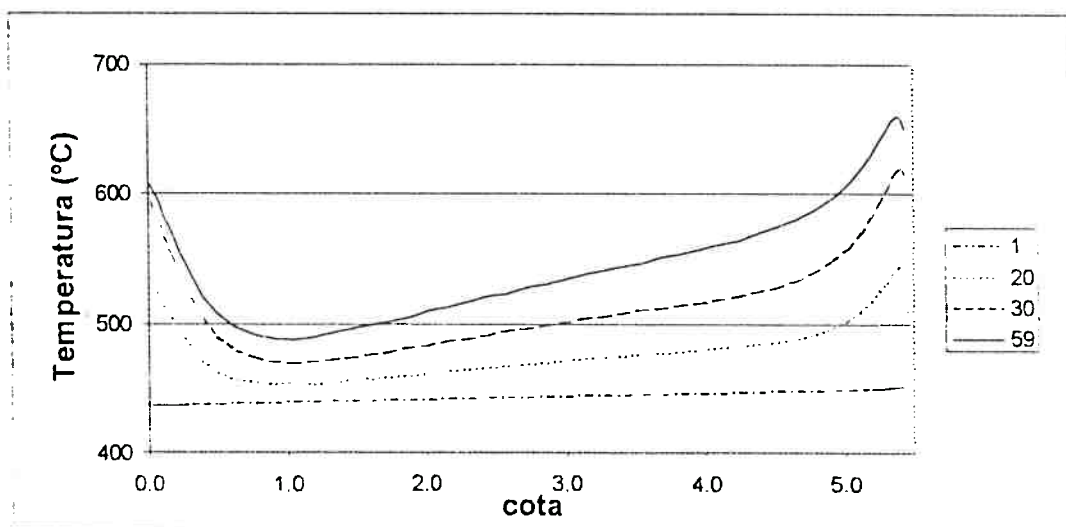


Figura 7.2. Alguns perfis de temperaturas no tubo de separação até sua convergência.

Os perfis de fluxos térmicos calculados a partir desse perfil de temperaturas para a face interna do tubo de separação pelos dois programas estão apresentados mais adiante.

7.2. Resultados

Foi processado os dois programas a partir do perfil de temperaturas convergido e seus resultados são apresentados a seguir.

7.2.1. Campo de velocidade dos gases

Conforme comentado no item 4.5, ocorrem variações de velocidade apenas nas proximidades da parede do duto. Para representar graficamente esse fato, foram definidas cinco seções transversais ao sentido de escoamento do gás, mostradas na Figura 7.3, cujos perfis de velocidade axial estão representados na Figura 7.4.

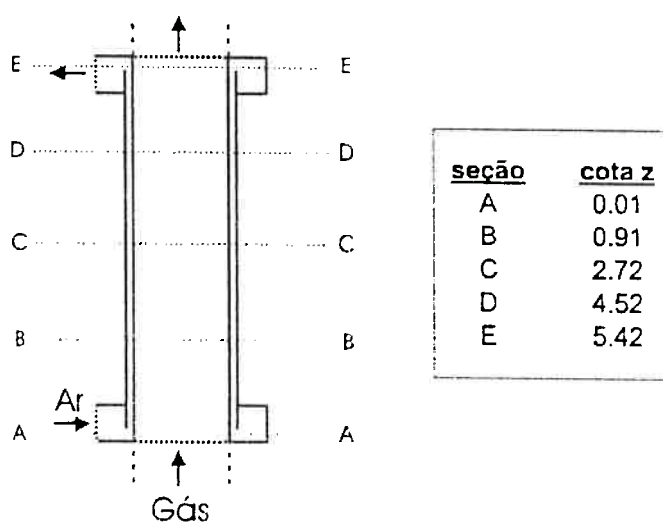


Figura 7.3. Seções transversais à direção do escoamento.

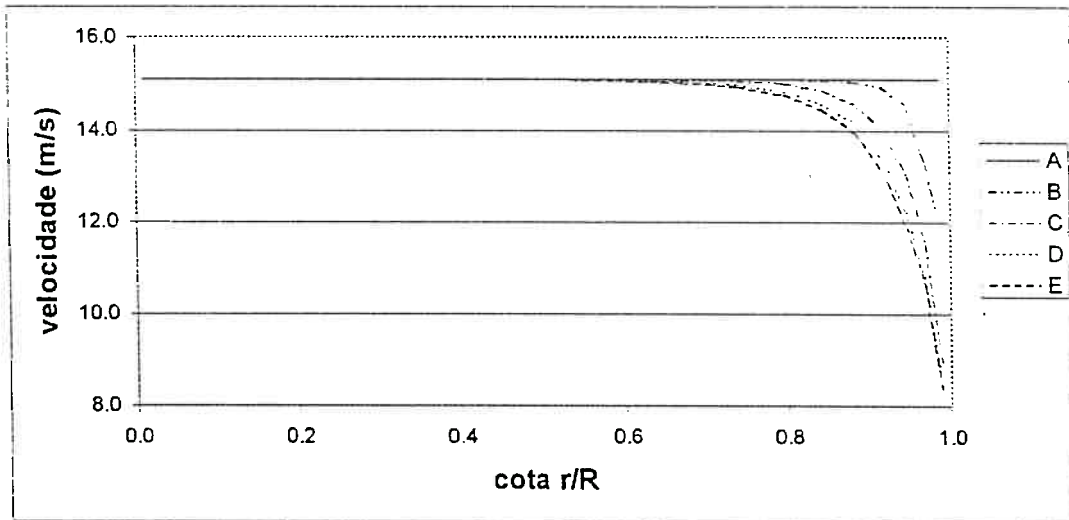


Figura 7.4. Perfis da componente axial da velocidade nas seções transversais.

Outra maneira de apresentar essa variação é apresentar os perfis da componente axial da velocidade ao longo do trocador, a determinadas distâncias do eixo do tubo, ou seja, considerando a representação do trocador num domínio bidimensional, em determinados eixos r . Foram definidos os seguintes eixos:

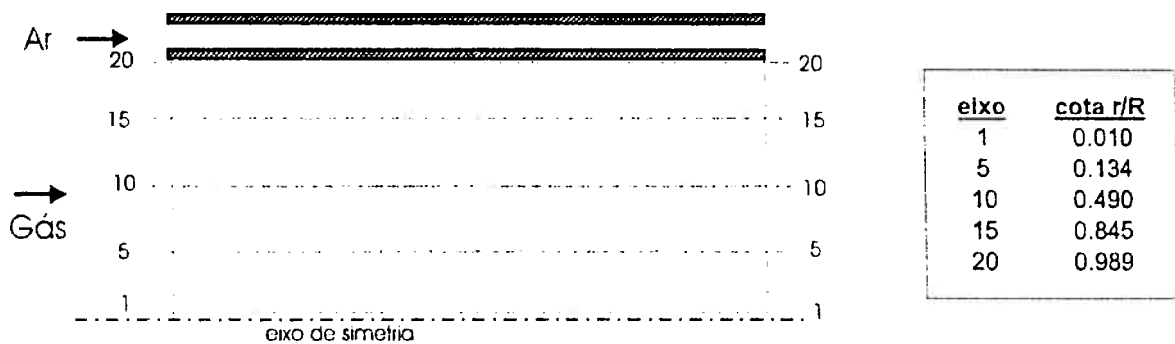


Figura 7.5. Eixos na direção r , definidos para apresentação dos resultados.

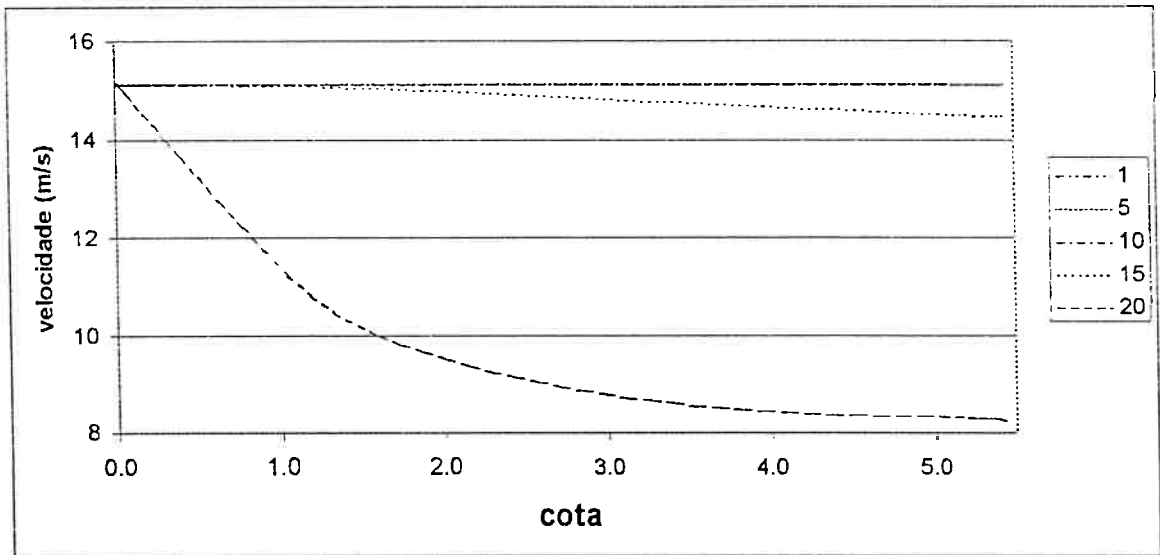


Figura 7.6. Perfis da componente axial da velocidade nos eixos r definidos.

Como se pode perceber, observando os dois gráficos anteriores, a velocidade varia pouco ao longo do trocador a menos das proximidades da parede.

Os perfis da componente radial apresentam variações insignificantes, de modo que prescindiu-se de suas apresentações.

7.2.2. Campo de temperatura dos gases

Apresenta-se o campo de temperatura dos gases através de gráficos análogos aos utilizados para apresentar o campo de velocidade dos gases.

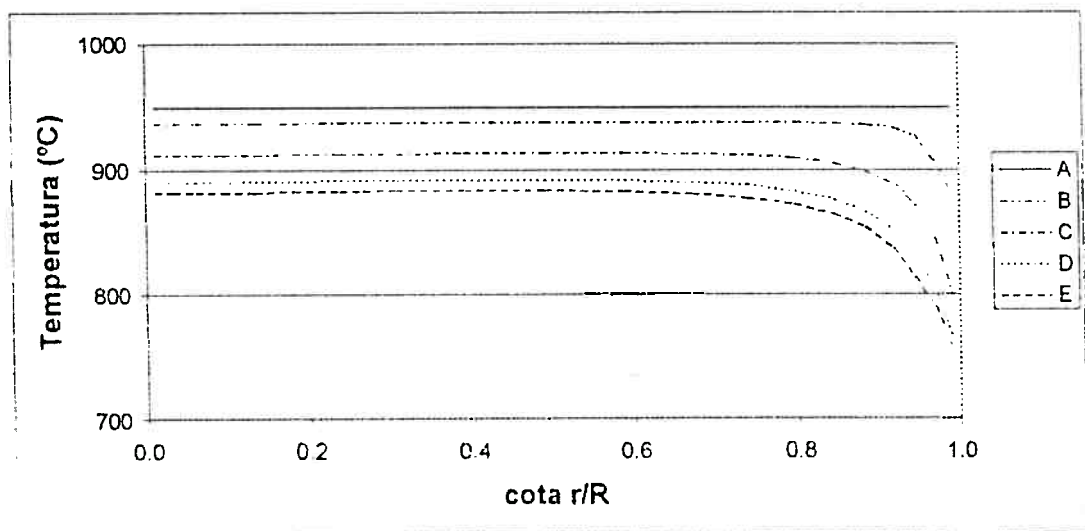


Figura 7.7. Perfis de temperatura nas seções transversais definidas.

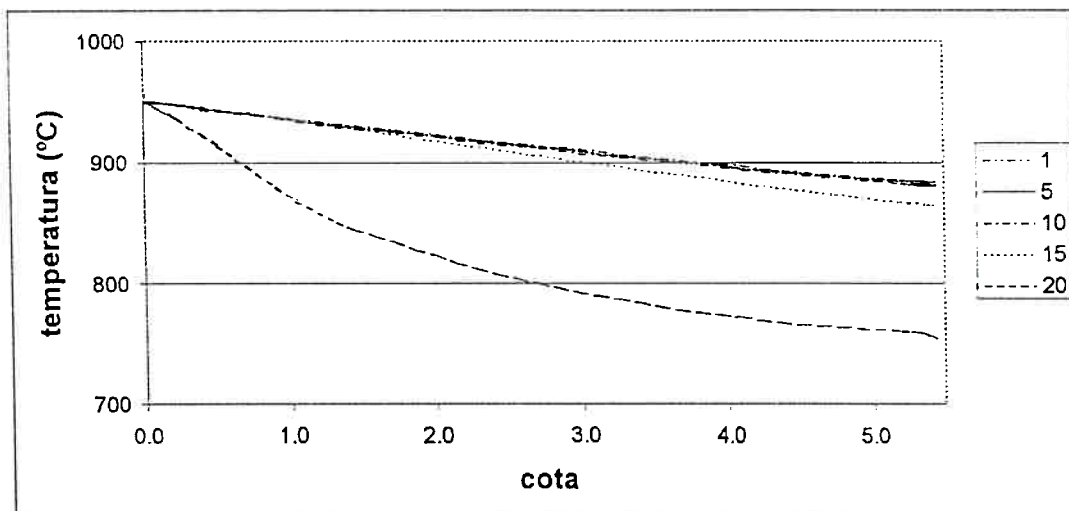


Figura 7.8. Perfis da temperatura ao longo do trocador, nos eixos definidos.

As temperaturas médias do gás ao longo do trocador decrescem pela troca térmica para o ar. Assim, os perfis de temperatura ao longo do trocador, nos eixos r definidos, decrescem com o aumento da cota do ponto, como pode ser observado na Figura 7.8. Esse fato, também, pode ser identificado através da Figura 7.7, onde os perfis de temperatura nas seções transversais ao escoamento “rebaixam-se” à medida que se afastam da entrada do escoamento

7.2.3. Temperaturas ao longo do trocador

A Figura 7.9 permite uma visão geral de como evoluem as temperaturas. Nela são apresentados os principais perfis ao longo do reator.

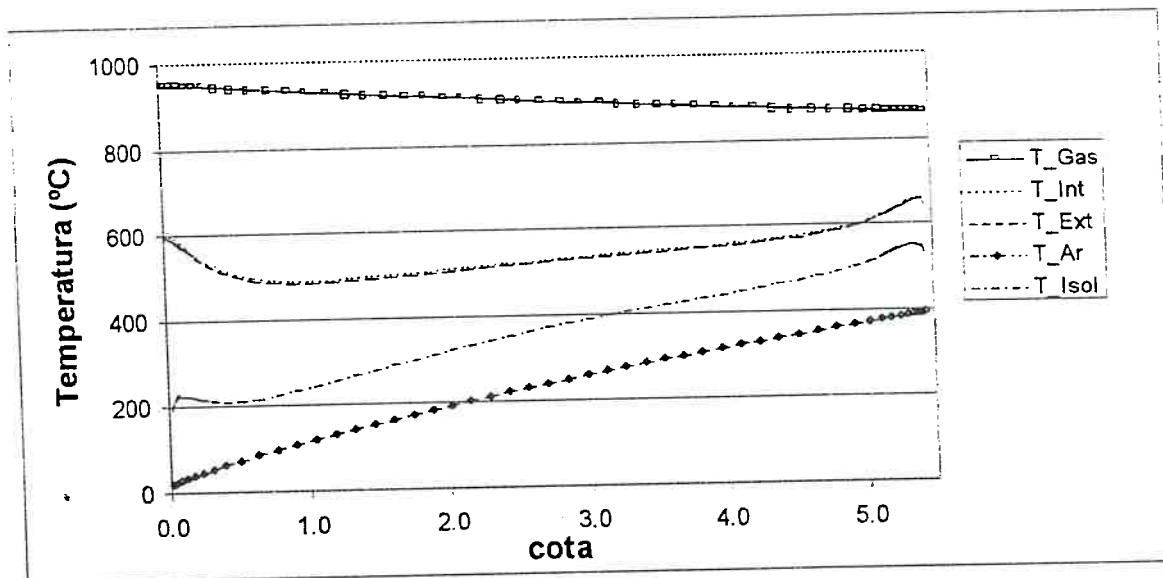


Figura 7.9. Temperaturas ao longo do recuperador de calor.

Na Figura 7.9, a legenda “T_Int” representa a temperatura da parede interna do tubo que separa o gás do ar, “T_Ext” representa a temperatura da parede externa desse tubo, “T_Isol” representa a temperatura do tubo que confina o ar externamente, “T_Gas” e “T_Ar” representam as temperaturas médias do gás e do ar, em cada seção transversal, respectivamente².

Os parâmetros de projeto do recuperador de calor, fornecidos pelo fabricante, de renome internacional, “Stein Surface Combustion”, são:

Vazão de Gás	Vazão de Ar	Temperatura do Gás		Temperatura do Ar	
		entrada	saida	entrada	saida
27605 Nm ³ /h	8943 Nm ³ /h	950 °C	875 °C	20 °C	310 °C

A simulação indicou que, através do trocador de calor, o gás se resfria de 950 a 861 °C, e que o ar se aquece de 20 a 393 °C. Os resultados da simulação são cerca de 16 % superiores aos dados de projeto calculado em termos da potência trocada entre o gás e o ar. Outros comentários são feitos acerca dessa comparação no capítulo 8.

O tubo isolado que confina o ar externamente tem temperaturas superiores às do ar, conforme era esperado. Esse tubo recebe um fluxo térmico por meio da radiação

² As variações da temperatura do ar numa seção transversal são pequenas. Assim, optou-se por apresentar apenas o perfil da sua temperatura média.

oriunda da superfície externa do tubo de separação ar/gases e o transfere por convecção ao ar.

Pode-se perceber que os perfis de temperaturas das faces interna e externa do tubo que separa o gás do ar e do tubo que confina o ar externamente apresentam, na sua grande parte central, curvas ascendentes em relação à cota do trocador de calor. Esse fato se deve ao aquecimento da corrente de ar, que apresenta gradiente, em valor absoluto, muito superior ao do resfriamento do gás.

Pode-se observar também que esses perfis apresentam nas suas extremidades elevações de temperatura, vide Figura 7.9. Isso se deve à conjugação dos seguintes fatos:

- à medida que os elementos de superfície se situam mais próximos das seções de entrada ou saída, maior é a influência dessas seções naqueles elementos (maior é o “fator de forma” entre a seção e o elemento de superfície);
- as seções de entrada e saída dos fluidos foram modeladas como superfícies negras à temperatura do fluido naquelas seções, conforme apresentado no item 5.3.

Uma superfície negra emite radiação com intensidade superior à emitida por um volume gasoso, à mesma temperatura, pois, conforme a equação (5.16), as intensidades de radiação que um volume gasoso e uma superfície negra emitem, à mesma temperatura, relacionam-se como:

$$I = a I_b \quad (7.6)$$

onde: I é a intensidade de radiação emitida pelo volume gasoso

I_b é a intensidade de radiação emitida pela superfície negra

a é o coeficiente de absorção do meio gasoso

Para o meio constituído pelos gases de combustão, o coeficiente de absorção vale 0,25.

Mesmo que uma superfície negra se encontre a temperatura ligeiramente inferior que um volume gasoso, ela pode emitir radiação com intensidade maior que a emitida pelo volume de gás.

Assim, as elevações da temperatura das seções dos tubos próximas às extremidades refletem o aumento do fluxo de radiação que essas seções recebem, devido à incidência de um número maior de feixes originados da parede, e que são de maiores intensidades que os originados dos volumes gasosos.

A elevação da temperatura nas regiões do tubo próximas das extremidades detectada pela simulação tem correspondência com o fenômeno real. Medições de temperatura de superfícies refratárias de fornos industriais têm identificado o mesmo comportamento.

O fato da elevação das curvas de temperatura das superfícies sólidas, próximas à seção de entrada ser maior que a elevação próxima à de saída, deve-se ao fato de que a temperatura da primeira seção é superior à temperatura dos gases nas seções do recuperador de calor adjacentes a ela, enquanto que ocorre o contrário para a seção de saída.

A Figura 7.10 apresenta, em destaque, as curvas das temperaturas das faces interna e externa do tubo de separação. A diferença de seus valores, para cada cota, também está apresentada. Como foi adotado um valor constante para a condutividade térmica do tubo, o perfil da diferença é semelhante ao perfil do fluxo de calor que atravessa o tubo, que é apresentado a seguir.

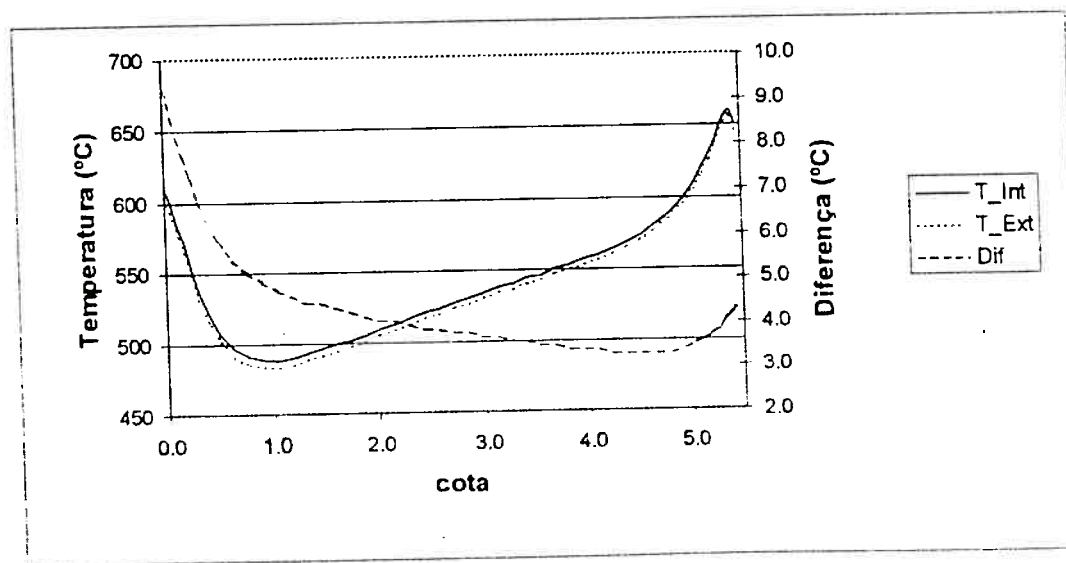


Figura 7.10. Temperaturas das faces interna e externa do tubo de separação, e sua diferença.

7.2.4. Fluxos térmicos transferidos do gás para o ar

A Figura 7.11 apresenta os perfis dos fluxos térmicos que atravessam o tubo de separação ar/gás, calculados pelos dois programas.

Como comentado no item 7.1, a diferença entre os valores calculados pelos dois programas diferem menos de 8 % para qualquer cota.

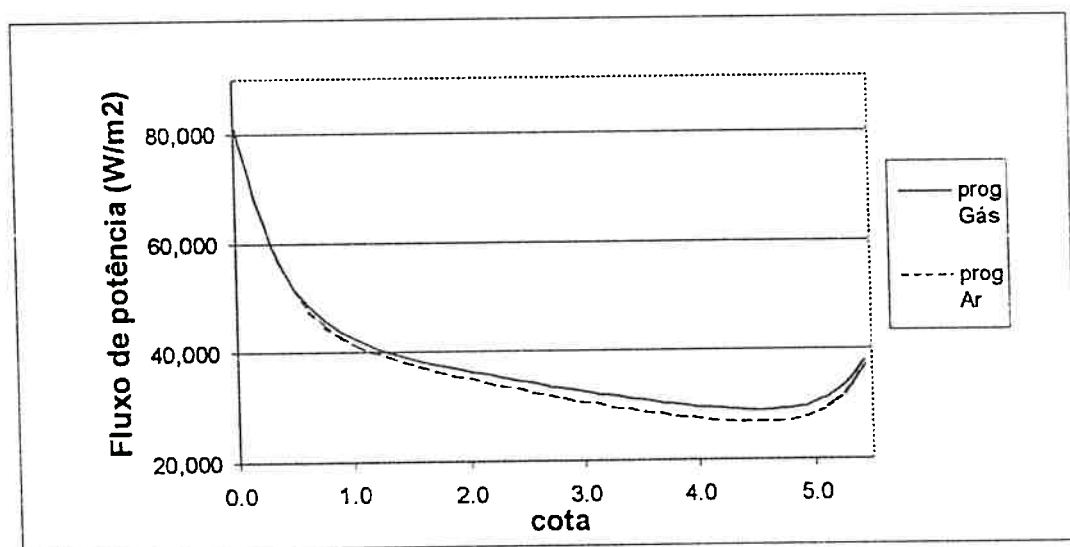


Figura 7.11. Perfis de fluxos térmicos que atravessam o tubo de separação ar/gás.

O comportamento descendente do perfil de fluxo térmico com relação à cota z do recuperador é o comportamento esperado de um recuperador de calor com fluxos em escoamento co-corrente, pois, de maneira geral, o fluxo térmico transferido é função da diferença entre as temperaturas médias do gás e do ar, e esta é máxima à entrada do recuperador e diminui com o aumento da cota z .

Os aumentos dos fluxos transferidos nas extremidades do recuperador reflete a maior iteração que as regiões da superfície do tubo próximas às seções de entrada e saída dos gases têm com elas, comentado no item 7.2.3.

7.2.5. A importância da radiação como mecanismo de troca térmica dos gases

A Figura 7.12 procura apresentar quantitativamente a importância da radiação na troca térmica global do recuperador. A legenda “q cond” representa fluxo térmico líquido transferido dos gases para a parede do tubo de separação, “q rad” representa o fluxo devido à radiação, enquanto que, “q pelicula” representa o fluxo devido à convecção combinada com condução (calculado com o uso do coeficiente de película, de acordo com o item 5.1).

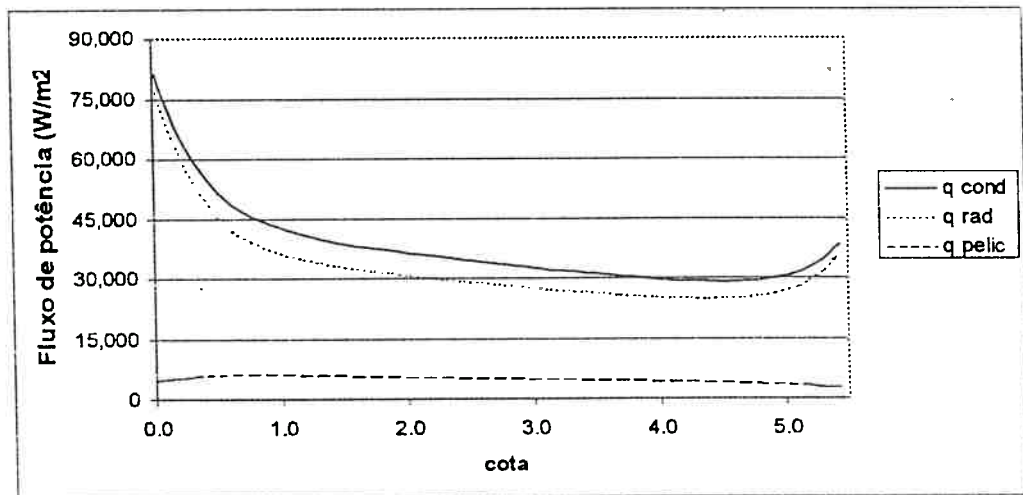


Figura 7.12. Fluxos de potência transferidos dos gases.

Pode-se perceber claramente que o mecanismo da radiação é o grande responsável pela troca térmica dos gases para o tubo de separação. Entretanto, no conduto do ar, este é aquecido quase que exclusivamente por convecção, já que o ar é transparente à radiação.

7.2.6. Fluxos de potência radiativos

A Figura 7.13 apresenta os perfis de $(q_r^+ - q_r^-)$ ao longo do trocador nos eixos r definidos na Figura 7.5. O parâmetro $(q_r^+ - q_r^-)$ representa, em cada ponto, o balanço das componentes radiais dos fluxos de potência radiativos que nele incidem. O fato de $(q_r^+ - q_r^-)$ ser positivo em praticamente todos os pontos do recuperador, indica que potência é transferida das regiões centrais (isto é, próximas ao eixo axial) para as regiões próximas às superfícies do tubo, em praticamente todo o recuperador.

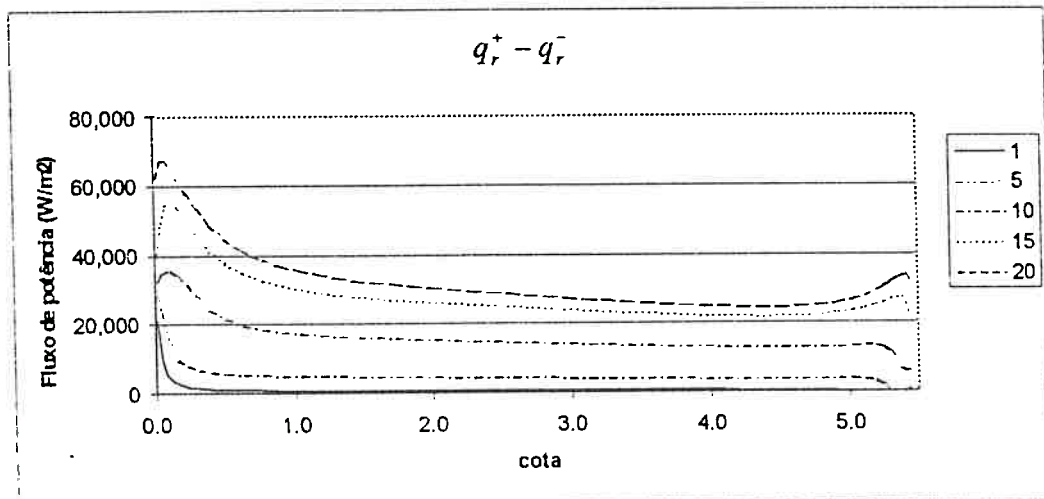


Figura 7.13. Perfis de $(q_r^+ - q_r^-)$ ao longo do trocador.

A Figura 7.13 permite observar que os perfis “rebaixam-se” quando a cota do seu eixo r diminui. Isto é, numa mesma seção transversal, o fluxo líquido de radiação radial aumenta ao se aproximar da parede do tubo. Esse fato é explicado, novamente, pelo aumento da influência da parede à medida em que diminui a distância entre o volume de controle e a parede. Como a temperatura da superfície é muito menor que a dos gases, a intensidade que ela emite é menor que as dos volumes de gases adjacentes, aumentando a diferença $(q_r^+ - q_r^-)$.

O perfil de $(q_r^+ - q_r^-)$, no eixo 20, apresenta formato muito semelhante ao perfil do fluxo térmico transferido dos gases para a parede (Figura 7.12), pois a componente radial é a

principal componente do vetor fluxo térmico por radiação, o que pode ser deduzido da observação dos perfis de $(q_r^+ - q_r^-)$ e $(q_z^+ - q_z^-)$, Figuras 7.13 e 7.14.

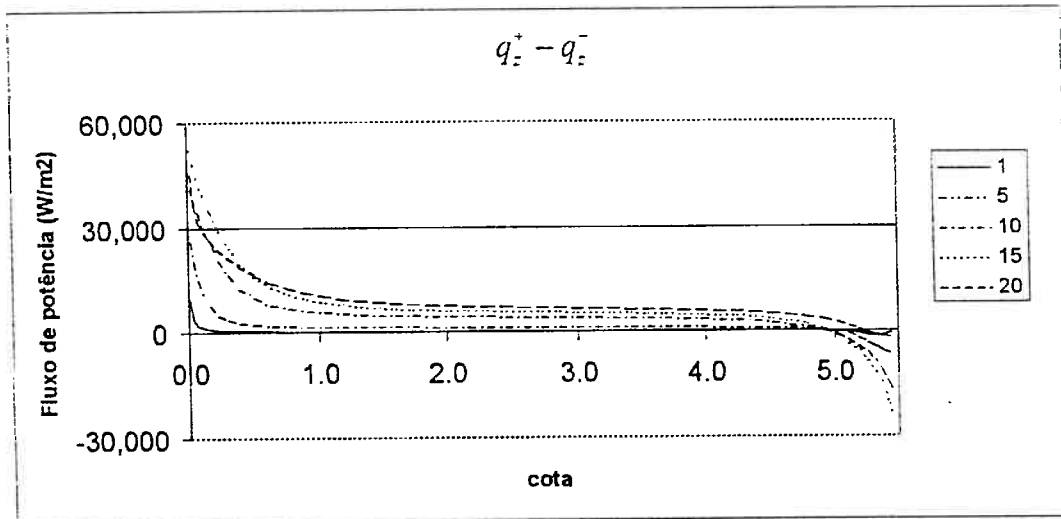


Figura 7.14. Perfis de $(q_z^+ - q_z^-)$ ao longo do trocador.

Quanto ao fluxo líquido de radiação na direção axial, pode-se dizer que é menos expressivo que o fluxo na direção radial, através da observação das figuras 7.13 e 7.14.

A Figura 7.14 mostra que $(q_z^+ - q_z^-)$ é positivo na maior parte do trocador, indicando que, como resultado do balanço líquido de potência radiativa na direção axial, potência é transferida das regiões iniciais do trocador para as regiões finais, o que não é válido apenas na região próxima à saída dos gases. Nessa região, predomina a influência da seção de saída, que mesmo a temperatura inferior do que dos volumes gasosos a montante, emite intensidades de radiação superiores àquelas emitidas por esses volumes. São as intensidades emitidas pela superfície de saída dos gases que determinam os valores negativos de $(q_z^+ - q_z^-)$.

As elevações dos fluxos de potência radiativa nas extremidades do recuperador são, novamente, reflexos da influência das seções de entrada e saída dos gases.

8. Conclusões e considerações finais

A simulação do recuperador de calor apresentou resultados bastante coerentes. Prova disso, são os perfis de temperatura das superfícies sólidas que apresentaram características verificadas experimentalmente, como por exemplo, a elevação das temperaturas nas regiões próximas à saída dos gases.

A comparação numérica dos resultados obtidos com os de projeto, apesar de poder ser considerada satisfatória, está prejudicada. Isso porque se desconhece se os dados fornecidos pelo fabricante correspondem a valores medidos, ou, se não trazem algum fator de atenuação implícito de maneira a garantir determinado nível de desempenho ao usuário do equipamento.

Os modestos recursos computacionais necessários para a simulação do caso considerado, provam a versatilidade do modelo de radiação implantado. Recursos de maior porte podem ser necessários, no entanto, se complexidades forem adicionadas ao problema.

Pode-se afirmar, portanto, que o método das ordenadas discretas deve ser considerado quando se desejar o cálculo das trocas de calor por radiação, especialmente se esse problema for combinado com o da determinação dos padrões de escoamento de um fluido.

Anexo A: Fórmula da derivada direcional em coordenadas cilíndricas

Apresenta-se o desenvolvimento matemático do termo da derivada direcional em coordenadas cilíndricas para geometrias com simetria axial. Tal desenvolvimento tem sido omitido nos artigos recolhidos e não é complicado, desde que sejam escolhidos os ângulos convenientes.

Antes de lidar propriamente com o equacionamento da radiação, deve-se estabelecer algumas relações geométricas. Considere-se o esboço da Figura A1, apresenta um feixe de radiação percorrendo a direção \vec{s} .

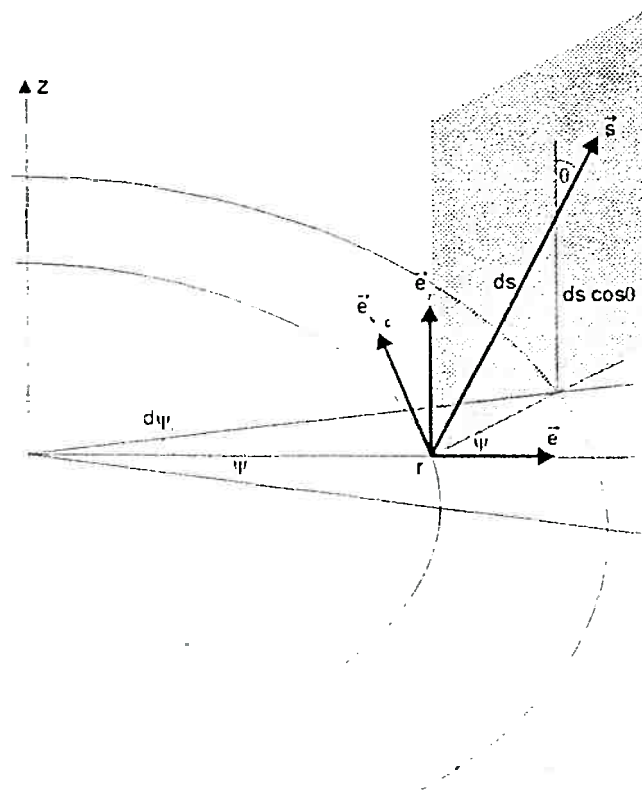


Figura A1 – Esquema do sistema de coordenadas com um feixe de radiação na direção

O ângulo ψ_c corresponde ao ângulo do sistema de coordenadas cilíndricas.

O ângulo θ corresponde ao ângulo entre a direção s e o eixo \bar{e}_z ;

O ângulo ψ corresponde ao ângulo entre a projeção da direção s no plano perpendicular a \bar{e}_z e o eixo \bar{e}_r . Observe que este ângulo varia de acordo com o ponto considerado na direção s , como pode-se perceber na Figura A.2, que apresenta um detalhamento do plano perpendicular ao eixo \bar{e}_z da Figura anterior.

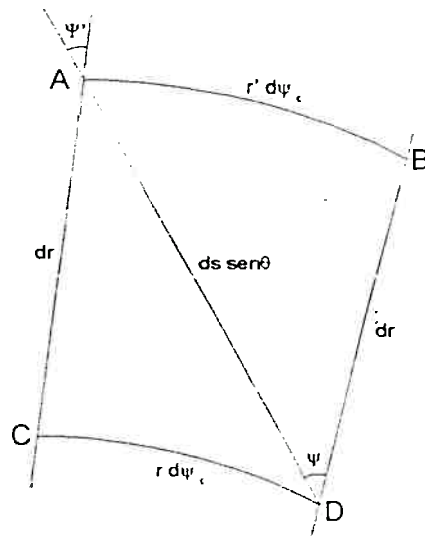


Figura A2 – Detalhe do plano perpendicular ao eixo z .

Aplicando-se a lei dos senos para os triângulos ACD e ABD, obtém-se:

$$r \operatorname{sen} \psi = r' \operatorname{sen} \psi'$$

de onde se conclui que ψ pode ser vista como função exclusiva de r , podendo escrever:

$$d(r \operatorname{sen} \psi) = 0 \Rightarrow \frac{d\psi}{dr} = -\frac{\operatorname{tg} \psi}{r}$$

Por outro lado, pode-se escrever:

$$\frac{d\psi}{ds} = \frac{d\psi}{dr} \frac{dr}{ds}$$

que combinada com a expressão anterior fornece:

$$\frac{d\psi}{ds} = -\frac{\text{sen } \psi \text{ sen } \theta}{r}$$

Considere-se o caso unidimensional, no qual as variáveis variam apenas com r , isto é:

$$\frac{\partial I}{\partial z} = \frac{\partial I}{\partial \psi_c} = 0$$

Adotando as coordenadas θ e ψ , reconhece-se que I varia com r , θ e ψ .

Assim, pela teoria das funções implícitas:

$$\vec{s} \cdot \nabla I = \frac{dI}{ds} = \frac{\partial I}{\partial r} \frac{dr}{ds} + \frac{\partial I}{\partial \theta} \frac{d\theta}{ds} + \frac{\partial I}{\partial \psi} \frac{d\psi}{ds}$$

Utilizando-se as relações geométricas desenvolvidas, e reconhecendo-se que $\frac{d\theta}{ds} = 0$,

chega-se a

$$\frac{dI}{ds} = \text{sen } \theta \cos \psi \frac{\partial I}{\partial r} - \frac{\text{sen } \theta \text{ sen } \psi}{r} \frac{dI}{d\psi}$$

chamando-se de $\mu = \text{sen } \theta \cos \psi$

$$\eta = \text{sen } \theta \text{ sen } \psi$$

e observando-se que $\mu = \frac{\partial \eta}{\partial \psi}$

a equação anterior pode ser escrita como:

$$\frac{dI}{ds} = \frac{\mu}{r} \frac{\partial}{\partial r} (rI) - \frac{1}{r} \frac{\partial}{\partial \psi} (\eta I)$$

Essa equação é o desenvolvimento do termo da derivada direcional da intensidade de radiação para o caso unidimensional.

Desenvolver-se-á em seguida para o caso tridimensional axissimétrico. Inicialmente, considere a extrapolação da equação anterior para casos tridimensionais. Para isso, basta adicionar as dependências de I em relação a z e ψ_c .

Assim, $I = I(r, \psi, \psi_c, z)$, o que leva a:

$$\frac{dI}{ds} = \frac{\partial I}{\partial r} \frac{dr}{ds} + \frac{\partial I}{\partial \psi} \frac{d\psi}{ds} + \frac{\partial I}{\partial \psi_c} \frac{d\psi_c}{ds} + \frac{\partial I}{\partial z} \frac{dz}{ds}$$

Observando-se a Figura A.1, pode-se concluir as seguintes relações:

$$\frac{dz}{ds} = \cos \theta \quad , \text{ variável que chamaremos de } \zeta, \text{ e}$$

$$\frac{d\psi_c}{ds} = \frac{\sin \theta \sin \psi}{r} \quad , \text{ termo que se anula no caso axissimétrico,}$$

Assim, obtém-se:

$$\frac{dI}{ds} = \frac{\mu}{r} \frac{\partial}{\partial r} (rI) - \frac{1}{r} \frac{\partial}{\partial \psi} (\eta I) + \zeta \frac{\partial I}{\partial z}$$

Anexo B: Detalhes dos programas

1. Determinação dos coeficientes de película

Foram empregadas as expressões relacionadas abaixo, conforme recomendações de BENNETT; MYERS (1978), para a determinação dos coeficientes de película. Todas as propriedades são calculadas na temperatura média do fluido em questão.

a) Para o escoamento do gás, equação de Dittus e Boelter:

$$\frac{hD}{k} = 0.023 \left(\frac{Dv\rho}{\mu} \right)^{0.8} \left(\frac{c_p\mu}{k} \right)^{0.3}$$

b) Para o escoamento do ar

- Para a parede externa do tubo interno, equação de Wiegand:

$$\frac{hD_{eq}}{k} = 0.023 \left(\frac{D_{eq}v\rho}{\mu} \right)^{0.8} \left(\frac{c_p\mu}{k} \right)^{0.4} \left(\frac{D_2}{D_1} \right)^{0.45}$$

onde D_{eq} é o diâmetro hidráulico do espaço anular

D_1 é o diâmetro externo do tubo interno

D_2 é o diâmetro interno do tubo externo

- Para a parede interna do tubo externo, equação de Sieder e Tate:

$$\frac{h}{v\rho c_p} \left(\frac{c_p\mu}{k} \right)^{2/3} \left(\frac{\mu_s}{\mu} \right)^{0.14} = 0.023 \left(\frac{D_{eq}v\rho}{\mu} \right)^{-0.2}$$

onde μ_s é calculado na temperatura da parede.

2. Determinação das propriedades radiativas

a) Índice de refração do meio

Tanto o ar como o gás têm o índice de refração igual a 1, conforme SIEGEL; HOWELL (1972), Tabela 21-1, página 700.

b) Coeficiente de absorção do meio

O ar é transparente à radiação, assim utilizou-se um valor pequeno para seu coeficiente de absorção (0,001).

Para a mistura dos gases de combustão, calculou-se seu coeficiente utilizando-se as tabelas de HOTTEL; SAROFIM (1967) a partir da composição dos gases. Os gráficos utilizados foram os das figuras 6.9 (pg. 229), 6.11 (pg. 232) e 6.12 (pg. 233). O valor encontrado para coeficiente de absorção dos gases foi de 0,25.

c) Emissividade da superfície dos dutos no compartimento do ar

Considerou-se que o compartimento do ar é constituído por “aço com superfície coberta com camada de óxido”, atribuindo-se assim, conforme SIEGEL; HOWELL (1972), Apêndice D, pg. 794, o valor de 0,8 para sua emissividade.

O compartimento dos gases é constituído por superfícies negras.

3. Determinação das demais propriedades dos fluidos

Para a determinação das propriedades, foram utilizadas relações recomendadas por REID et al. (1987), exceto para a densidade. Para esta, foi empregada a expressão para gás perfeito.

a) viscosidade

- para os componentes individualmente: método de Lucas (REID et al., página 400).
- para a mistura dos componentes: método de Wilke (REID et al., página 407).

b) condutividade térmica

- para os componentes individualmente: Tabela 10.3 (REID et al., página 515).
- para a mistura dos componentes: equação de Wassiljewa (REID et al., página 530).

c) calor específico a pressão constante

- para os componentes individualmente: Appendix A – Property Data Bank (REID et al., página 656).
- para a mistura dos componentes: equação 5-5.1 (REID et al., página 121).

4. Detalhes computacionais

a) malhas utilizadas

para o programa dos gases: 20 divisões na direção radial e 50 na axial

para o programa do ar: 5 divisões na direção radial e 50 na axial

b) critérios de convergência e fatores de relaxação

Variável	Critério de convergência	Fator de relaxação
u e v	0,005 m/s	4.
P	1. Pa	---
k e ε	0,1 %	1.
I e T	0,1 %	0,6 ¹

c) Software para simulação: MicroSoft Fortran PowerStation for Windows.

d) Processador: Pentium 100 MHz.

e) Tempo de processamento: variável entre 2 a 8 minutos, dependendo das condições iniciais.

¹ Os fatores de relaxação estão referidos a fórmulas diferentes. Para I e T , vale a seguinte relação $\phi_{new} = \alpha \phi_{calc} + (1 - \alpha) \phi_{old}$, enquanto que as demais variáveis utilizam a formulação "E", descrito em JEN (1989).

Referências bibliográficas

- BENNETT, C.O.; MYERS, J.E.: Fenômenos de Transporte – quantidade de movimento, calor e massa, Editora McGraw-Hill do Brasil Ltda, 1978.
- CARLSON, B.G; LATHROP, K.D.: Transport Theory: The Method of Discrete Ordinates, in Computing Methods in Reactor Physics, edited by Greenspan, Kelber and Okrent, Gordon and Breach, New York, 1968.
- DE MARCO, A. G.; LOCKWOOD, F. C.: A New Flux Model for the Calculation of Radiation in Furnaces. La Rivista dei Combustibili. vol. 29,pp. 184-196, 1975.
- FIVELAND, W.A.: A Discrete Ordinates Method for Predicting Radiative Heat Transfer in Axisymmetric Enclosures, ASME Paper 82-HT-20, 1982.
- FIVELAND, W.A.: Discrete Ordinate Methods for Radiative Heat Transfer in Isotropically and Anisotropically Scattering Media, ASME Journal of Heat Transfer, vol. 109, pp. 809-812, 1987.
- FIVELAND, W.A.: Discrete Ordinates Solutions of the Radiative Transport Equation for Rectangular Enclosures, ASME Journal of Heat Transfer, vol. 106, pp. 699-706, 1984.
- GIBSON, M. M.; MONAHAN, J. A.: A Simple Model of Radiation Heat Transfer from a Cloud of Burning Particles in a Confined Gas Stream. Int. J. Heat Mass Transfer, vol. 14, pp. 141-147, 1971.
- GOSMAN, A. D.; LOCKWOOD, F. C.: Incorporation of a Flux Model for Radiation into a Finite-Difference Procedure for Furnace Calculations. 14th Symposium International on Combustion, The Combustion Institute, pp. 661-671, 1973.
- GUPTA, A.K.; LILLEY, D.G.: Flowfield Modeling and Diagnostics. Abacus Press. USA. 1985.

- HJERTAGER, B. H.; MAGNUSSEN, B. F.: Computer Simulation of Flow, Heat Transfer and Combustion in Three-Dimensional Furnaces. Physico Chemical Hydrodynamics, Vol. 3, nº 3/4, pp. 231-250, 1982.
- HOTTEL, H.C.; SAROFIM, A.F.: Radiative Transfer. McGraw-Hill, 1967.
- JAMALUDDIN, A.S.; SMITH, P.J.: Predicting Radiative Transfer in Axisymmetric Cylindrical Enclosures Using the Discrete Ordinates Method. Combustion Science and Technology, vol. 62, pp. 173-186, 1988.
- JEN, L.C.: Modelo matemático para escoamento turbulento confinado, visando a configuração de queimadores para gases de petróleo e outros gases industriais. Dissertação de mestrado submetida à Escola Politécnica da Universidade de São Paulo. 1989.
- KUMAR, S.; MAJUMDAR, A.; TIEN, C.L.: The Differential-Discrete-Ordinate Method for Solutions of the Equation of Radiative Transfer, ASME Journal of Heat Transfer, vol. 112, nº 2, pp. 424-429, 1990.
- LOCKWOOD, F. C.; SHAH, N. G.: An Improved Flux Model for the Calculation of Radiation Heat Transfer in Combustion Chambers. Proceedings of 16th National Heat Transfer Conference, pp. 2-7, Aug. 1976.
- LOWES, T. M.; BARTELD, H.; HEAP, M. P.; MICHELFELDER, S.; PAI, B. R.: Prediction of Radiant Heat Flux Distribution. Heat Transfer in Flames. Scripta Book Company. pp. 179-190, 1974.
- MALISKA, C.R.: Transferência de Calor e Mecânica dos Fluidos Computacional, in APOSTILA DE CURSO. PROPESA/CTA. 1988.
- MALISKA, C.R.: Transferência de Calor e Mecânica dos Fluidos Computacional – Fundamentos e Coordenadas Generalizadas. Livros Técnicos e Científicos Editora S. A. 1995.
- MODEST, M. F.: Radiative Heat Transfer. McGraw-Hill Inc., 1993.

- PIKMAN, B.: Forno Esférico para Calibração de Radiômetros, in Relatório IPT “Apoio Tecnológico à Utilização Industrial de Gás Natural”, coordenação de Souza, F.D.A., 1987.
- REID, R.C. et al.: The Properties of Gases & Liquids, 4th. Edition. McGraw Hill Inc., 1987.
- SIEGEL, R.; HOWELL, J. R.: Thermal Radiation Heat Transfer. McGraw-Hill Kogakusha L.d., 1972.
- SPARROW, E.M.; CESS, R.D.: Radiation Heat Transfer. Hemisphere Publishing Co. 1978.
- SOUSA, F.D.A.: Método dos fluxos para cálculo das Trocas de Calor entre chamas e superfície. Seminário Interno AET-DME-IPT. São Paulo, 1989
- STEWARD, F. R.; CANNON, P.: The Calculation of Radiative Heat Flux in a Cylindrical Furnace Using the Monte Carlo Method, Int. J. Heat Mass Transfer, vol. 14, pp. 245-262, 1971.
- TRUELOVE, J.S.: Discrete-Ordinate Solutions of the Radiation Transport Equation, ASME Journal of Heat Transfer, vol. 109, nº 4, pp.1048-1051, 1987.
- TRUELOVE, J.S.: Three-Dimensional Radiation in Absorbing-Emitting-Scattering Media Using the Discrete-Ordinates Approximations. Journal of Quantitative Spectroscopy and Radiative Transfer, vol. 39, nº 1, pp. 27-31, 1988.
- VAN DONGEN, F. G.: Heat Transfer in gas-fired furnaces - a semi-empirical model combining the zone method for radiation and an integral technique for the flow pattern, Journal of the Institute of Energy, vol. 56, nº 429, pp. 184-190, 1983.
- VARMA, S.A.: Radiative Heat Transfer in a Pulverized-Coal Flame. Pulverized-Coal Combustion and Gasification. Plenum Press, pp. 83-106, 1979.
- VISCANTA, R. and Mengüç, M.P.: Radiation Heat Transfer in Combustion Systems. Prog. Energy Combust. Sc., vol. 13, pp. 97-160, 1987.

Apêndice: Listagem dos programas

```

*****
*** Arquivo 1: Programa "tese03_c.f90" ***
*** Escoamento dos gases ***
*****
!MS$REAL8
! *** Programa final: interior do tubo central, escoamento dos gases
PROGRAM MAIN
parameter(Nr=20,Nr1p=21,Nr1m=19,Nz=50,Nz1p=51,Nz1m=49,Nm=13)
dimension Coef_Mi(Nm),Coef_Zeta(Nm),W(Nm),AlphaMenos(Nm),AlphaMais(Nm)
integer M_Simetrico(Nm)
dimension R(0:Nr1p),Dr(0:Nr1p),Dr_N(Nr1m),Dr_S(2:Nr),VTurbK(Nr),VEpsilon(Nr), &
& Visc_Entrada(Nr),Ro_Entrada(Nr),Gama_Entrada(Nr),AreaZ(Nr),TWall_W(Nr),TWall_E(Nr)
dimension Z(0:Nz1p),Dz(0:Nz1p),Dz_E(Nz1m),Dz_W(2:Nz),ATdma(Nz),BTdma(Nz),CTdma(Nz),D
Tdma(Nz), &
& XTdma(Nz),BeTdma(Nz),GaTdma(Nz),TalW(Nz),TWall_S(Nz),TWall_N(Nz),CoefHeatTransfer
(Nz), &
& QToNorthBoundary(Nz),Temp_Media(Nz),Temp_Externa(Nz),U_Medio(Nz),FluxoCalorCond(N
z)
dimension P(Nr,Nz),U(0:Nr1p,Nz1p),Uch(Nr,Nz1p),V(Nr1p,0:Nz1p),Vch(Nr1p,Nz), &
& VolDzAp(Nr,2:Nz),VolDrAp(2:Nr,Nz),AuxMtx(0:Nr1p,0:Nz1p),ViscE(Nr,Nz), &
& TurbK(0:Nr1p,0:Nz1p),Epsilon(0:Nr1p,0:Nz1p),Can(Nr,Nz),Cas(Nr,Nz),Cae(Nr,Nz), &
& Caw(Nr,Nz),Gk(Nr,Nz),Ro(Nr,Nz),Viscosidade(Nr,Nz),Temperatura(Nr,Nz), &
& Cp(Nr,Nz),Gama(Nr,Nz),AreaR_N(Nr,Nz),AreaR_S(Nr,Nz),Volume(Nr,Nz)
dimension AIntensity(Nr,Nz,2:Nm),Pizao(Nr,Nz,2:Nm)
logical flaguv,flagimp,Flag_TFinal
Integer Upwind_NS,Upwind_EW,Upwind_Virtual
common/fdebug/flagimp
common/simpr/niter2,npress,dnpress
common/sturb/RlxK,rlxeps,epsk,epseps
common/sturb2/cmi,te,tk
common/sneww/rlxuv,epsu,epsv,epsj,niter
common/simpr2/ipressi,ipressj
common/cmml/AMol(5),Tc(5),Pc(5),Zc(5),AMi(5),CpA(5),CpB(5),CpC(5),CpD(5),Y(5)
common/cmml2/AK_A(5),AK_B(5),AK_C(5),AK_D(5),PesoMolecularMistura,Dr_Total
common/tuel/Pi,Sigma
common/tue2/Refraction,Extinction,Absorption,ReflectWall,EmissWall
common/tue3/Relax_Phi,Epsilon_Energy,Epsilon_Temp,MaxIterEnergy,MaxIterTemp,Relax_Temp

common/tue4/Diamond,Diamond_NS,Diamond_EW,Upwind_Virtual,Upwind_EW
open(5,file='tese03b_geral_in.dat',form='formatted',mode='read')
open(6,file='tese03c_out.dat',status='replace')
open(7,file='campos_iniciais_in.dat',form='formatted',mode='read')
open(9,file='TempDuto_Externa.dat',status='replace')
open(10,file='novo_perfil.dat',form='formatted',mode='read')
open(11,file='campos_iniciais_out.dat',status='replace')
data y,PesoMolecularMistura/0.119,0.093,0.753,0.033,0.002,29.18/ !Gas
data AMol(1),Tc(1),Pc(1),Zc(1),AMi(1),CpA(1),CpB(1),CpC(1),CpD(1) &
/44.01,304.1,73.8,0.274,0.,1.980E+01,7.344E-02,-5.602E-05,1.715E-08/
data AMol(2),Tc(2),Pc(2),Zc(2),AMi(2),CpA(2),CpB(2),CpC(2),CpD(2) &
/18.015,647.3,221.2,0.235,1.8,3.224E+01,1.924E-03,1.055E-05,-3.596E-09/
data AMol(3),Tc(3),Pc(3),Zc(3),AMi(3),CpA(3),CpB(3),CpC(3),CpD(3) &
/28.013,126.2,33.9,0.29,0.,3.115E+01,-1.357E-02,2.680E-05,-1.168E-08/
data AMol(4),Tc(4),Pc(4),Zc(4),AMi(4),CpA(4),CpB(4),CpC(4),CpD(4) &
/31.999,154.6,50.4,0.288,0.,2.811E+01,-3.680E-06,1.746E-05,-1.065E-08/
data AMol(5),Tc(5),Pc(5),Zc(5),AMi(5),CpA(5),CpB(5),CpC(5),CpD(5) &
/64.0E3,430.8,78.8,0.269,1.6,2.385E+01,6.699E-02,-4.961E-05,1.328E-08/
data AK_A(1),AK_B(1),AK_C(1),AK_D(1)/-7.215e-3,8.015e-5,5.477e-9,-1.053e-11/
data AK_A(2),AK_B(2),AK_C(2),AK_D(2)/7.341e-3,-1.013e-5,1.801e-7,-9.1e-11/
data AK_A(3),AK_B(3),AK_C(3),AK_D(3)/3.919e-4,9.816e-5,-5.067e-8,1.504e-11/
data AK_A(4),AK_B(4),AK_C(4),AK_D(4)/-3.273e-4,9.966e-5,-3.743e-8,9.732e-12/
data AK_A(5),AK_B(5),AK_C(5),AK_D(5)/-8.086e-3,6.344e-5,-1.382e-8,2.303e-12/
data Coef_Mi(1),Coef_Zeta(1),W(1)/-1.,0.,0./
data Coef_Mi(2),Coef_Zeta(2),W(2)/-0.9082483,-0.2958759,1.047197551/
data Coef_Mi(3),Coef_Zeta(3),W(3)/-0.9082483,0.2958759,1.047197551/
data Coef_Mi(4),Coef_Zeta(4),W(4)/-0.2958759,-0.9082483,1.047197551/
data Coef_Mi(5),Coef_Zeta(5),W(5)/-0.2958759,-0.2958759,1.047197551/
data Coef_Mi(6),Coef_Zeta(6),W(6)/-0.2958759,0.2958759,1.047197551/
data Coef_Mi(7),Coef_Zeta(7),W(7)/-0.2958759,0.9082483,1.047197551/
data Coef_Mi(8),Coef_Zeta(8),W(8)/0.2958759,-0.9082483,1.047197551/
data Coef_Mi(9),Coef_Zeta(9),W(9)/0.2958759,-0.2958759,1.047197551/
data Coef_Mi(10),Coef_Zeta(10),W(10)/0.2958759,0.2958759,1.047197551/
data Coef_Mi(11),Coef_Zeta(11),W(11)/0.2958759,0.9082483,1.047197551/
data Coef_Mi(12),Coef_Zeta(12),W(12)/0.9082483,-0.2958759,1.047197551/
data Coef_Mi(13),Coef_Zeta(13),W(13)/0.9082483,0.2958759,1.047197551/
data M_Simetrico/0,0,0,0,0,0,4,5,6,7,2,3/
Niter_Main=1
Flag_TFinal=.true.
call INICIO(Nr,Nz,Nr1p,Nz1p,Nr1m,Nz1m,Nm,AlphaMenos,AlphaMais,Coef_Mi,W, &

```



```

& R,Dr,Dr_S,Dr_N,AreaZ,VTurbK,VEpsilon,Visc_Entrada,Ro_Entrada,Gama_Entrada, &
& TWall_W,TWall_E, &
& Z,Dz,Dz_W,Dz_E,TalW,TWall_S,TWall_N,CoefHeatTransfer,QToNorthBoundary,U_Medio,
&
& Temp_Media, &
& AreaR_N,AreaR_S,Volume,F,U,V,ViscE,Viscosidade,TurbK,Epsilon,Ro, Temperatura,Cp,G
ama, &
& AIntensity,Pizao)
40 call COEFNS(Nr,Nz,Nrlp,Nzlp,R,Dr,Visc_Entrada,Ro_Entrada,Z,Dz,TalW,U,Uch,V,Vch,Vo
lDzAp, &
VolDrAp,ViscE,Viscosidade,Ro)
write(*,1010) Niter_Main
call PRESSAO(Nr,Nz,Nrlp,Nzlp,R,Dr,Dz,ATdma,BTdma,CTdma,DTdma, &
XTdma,BeTdma,GaTdma,P,Uch,Vch,VolDzAp,VolDrAp,AuxMtx,Ro)
call NEWVEL(Nr,Nz,Nrlp,Nzlp,flaguv,Dr,TalW,P,U,Uch,V,Vch,VolDzAp,VolDrAp, &
AuxMtx,TurbK,Ro,Viscosidade)
Niter_Main=Niter_Main+1
if(flaguv.or.Flag_TFinal)then
Gama=ViscE/0.7
Gama_Entrada=Visc_Entrada/0.7
call ENERGY(Nr,Nrlp,Nrlm,Nz,Nzlp,Nzlm,Nm,Flag_TFinal, &
& Coef_Mi,Coef_Zeta,W,AlphaMenos,AlphaMais,M_Simetrico, &
& Dr,Dr_N,Dr_S,AreaZ,TWall_W,TWall_E,Ro_Entrada,Gama_Entrada, &
& Dz,Dz_E,Dz_W,TWall_N,Temp_Media,Temp_Externa,U_Medio,FluxoCalorCond, &
& QToNorthBoundary,CoefHeatTransfer, &
& U,V,AreaR_N,AreaR_S,Volume,Temperatura,Ro,Cp,Gama,Viscosidade, &
& TurbK,Epsilon, &
& AIntensity,Pizao)
call TURBUL(Nr,Nrlp,Nz,Nzlp,R,Dr,VTurbK,VEpsilon,Visc_Entrada,Ro_Entrada,Z,Dz,TalW
,U,V,AuxMtx, &
Viscosidade,ViscE,TurbK,Epsilon,Ro,Gk,Can,Cas,Cae,Caw)
go to 40
else
call FINAL(Nr,Nz,Nrlp,Nzlp,U,V,Temperatura,Temp_Media,Temp_Externa,TWall_N,Z,C
oefHeatTransfer, &
P,VTurbK,VEpsilon,TurbK,Epsilon,TalW,FluxoCalorCond)
end if
1010 format(' N_Main=',i4)
end
!
SUBROUTINE COEFNS(Nr,Nz,Nrlp,Nzlp,R,Dr,Visc_Entrada,Ro_Entrada,Z,Dz,TalW,U,Uch,V,
Vch,VolDzAp, &
VolDrAp,ViscE,Viscosidade,Ro)
! *** calcula Uch,Vch,VolDzAp e VolDrAp ***
external Alfa,Beta
dimension R(0:Nrlp),Dr(0:Nrlp),Visc_Entrada(Nr),Ro_Entrada(Nr)
dimension Z(0:Nzlp),Dz(0:Nzlp),TalW(Nz)
dimension U(0:Nrlp,Nzlp),Uch(Nr,Nzlp),V(Nrlp,0:Nzlp),Vch(Nrlp,Nz), &
& VolDzAp(Nr,2:Nz),VolDrAp(2:Nr,Nz),ViscE(Nr,Nz),Viscosidade(Nr,Nz),Ro(Nr,Nz)
common/snewv/rlxuv,epsu,epsv,epsps,niter
do i=1,Nr
deltrn=R(i+1)-R(i)
deltrs=R(i)-R(i-1)
rintn=R(i)+Dr(i)*0.5
rints=R(i)-Dr(i)*0.5
sigmaz=R(i)*Dr(i)
do j=2,Nz
! *** face norte ***
if(i.eq.Nr)then
Visc=Viscosidade(Nr,j)
Ro_Interf=Prop_Interp(Ro(Nr,j-1),Dz(j-1),Ro(Nr,j),Dz(j))
else
Viscnw=Prop_Interp(ViscE(i,j-1),Dr(i),ViscE(i+1,j-1),Dr(i+1))
Viscne=Prop_Interp(ViscE(i,j),Dr(i),ViscE(i+1,j),Dr(i+1))
Visc=Prop_Interp(Viscnw,Dz(j-1),Viscne,Dz(j))
Ro_NW=Prop_Interp(Ro(i,j-1),Dr(i),Ro(i+1,j-1),Dr(i+1))
Ro_NE=Prop_Interp(Ro(i,j),Dr(i),Ro(i+1,j),Dr(i+1))
Ro_Interf=Prop_Interp(Ro_NW,Dz(j-1),Ro_NE,Dz(j))
end if
vn=(V(i+1,j-1)+V(i+1,j))*0.5
Peclet=Ro_Interf*vn*deltrn/Visc
an=rintn*(Dz(j-1)+Dz(j))*0.5*(-Ro_Interf*vn*(0.5-Alfa(Peclet))+Beta &
(Peclet)*Visc/deltrn)
! *** face sul ***
if(i.eq.1)then
Visc=Prop_Interp(ViscE(1,j-1),Dz(j-1),ViscE(1,j),Dz(j))
Ro_Interf=Prop_Interp(Ro(1,j-1),Dz(j-1),Ro(1,j),Dz(j))
else

```

```

        Viscsw=Prop_Interp(ViscE(i-1,j-1),Dr(i-1),ViscE(i,j-1),Dr(i))
        Viscse=Prop_Interp(ViscE(i-1,j),Dr(i-1),ViscE(i,j),Dr(i))
        Visc=Prop_Interp(Viscsw,Dz(j-1),Viscse,Dz(j))
        Ro_SW=Prop_Interp(Ro(i-1,j-1),Dr(i-1),Ro(i,j-1),Dr(i))
        Ro_SE=Prop_Interp(Ro(i-1,j),Dr(i-1),Ro(i,j),Dr(i))
        Ro_Interf=Prop_Interp(Ro_SW,Dz(j-1),Ro_SE,Dz(j))
    end if
    vs=(V(i,j-1)+V(i,j))*0.5
    Peclet=Ro_Interf*vs*deltrs/Visc
    as=rints*(Dz(j-1)+Dz(j))*0.5*(Ro_Interf*vs*(0.5+Alfa(Peclet))+Beta &
        (Peclet)*Visc/deltrs)
! *** face este ***
    Visc=ViscE(i,j)
    Ro_Interf=Ro(i,j)
    ue=(U(i,j)+U(i,j+1))*0.5
    Peclet=Ro_Interf*ue*Dz(j)/Visc
    ae=sigmaz*(-Ro_Interf*ue*(0.5-Alfa(Peclet))+Beta(Peclet)*Visc/Dz(j))
! *** face oeste ***
    Visc=ViscE(i,j-1)
    Ro_Interf=Ro(i,j-1)
    uw=(U(i,j),J(i,j-1))*0.5
    Peclet=Ro_Interf*uw*Dz(j-1)/Visc
    aw=sigmaz*(Ro_Interf*uw*(0.5+Alfa(Peclet))+Beta(Peclet)*Visc/Dz(j-1))
! *** calculo de Uch e VoldzAp ***
    Ap=(an+as+ae+aw)*(1.+1./rlxuv)
    Uch(i,j)=(an*U(i+1,j)+as*U(i-1,j)+ae*U(i,j+1)+aw*U(i,j-1)+ &
        (an+as+ae+aw)*U(i,j)/rlxuv)/Ap
    if(i.eq.Nr) then
        sp=0.5*(TalW(j)+TalW(j-1))*(R(Nr)+0.5*Dr(Nr))*(Z(j)-Z(j-1)) &
            /U(Nr,j)
        Ap0=an+as+ae+aw+sp
        Ap=Ap0*(1.+1./rlxuv)
        Uch(Nr,j)=(an*U(Nr1p,j)+as*U(Nr-1,j)+ae*U(Nr,j+1)+aw*U(Nr,j-1)+ &
            Ap0*U(Nr,j)/rlxuv)/Ap
    end if
    VoldzAp(i,j)=sigmaz/Ap
end do !j
end do !i
! *** pontos no contorno ***
do i=1,Nr
    Uch(i,1)=U(i,1)
    Uch(i,Nz1p)=U(i,Nz1p)
end do
! *** equacao do movimento para v ***
do i=2,Nr
    rmean=R(i)-Dr(i)*0.5
    Drmean=R(i)-R(i-1)
    sigmaz=rmean*Drmean
do j=1,Nz
! *** face norte ***
    Visc=ViscE(i,j)
    Ro_Interf=Ro(i,j)
    vn=(V(i,j)+V(i+1,j))*0.5
    Peclet=Ro_Interf*vn*Dr(i)/Visc
    an=R(i)*Dz(j)*(-Ro_Interf*vn*(0.5-Alfa(Peclet))+Beta(Peclet)*Visc/Dr(i))
! *** face sul ***
    Visc=ViscE(i-1,j)
    Ro_Interf=Ro(i-1,j)
    vs=(V(i,j)+V(i-1,j))*0.5
    Peclet=Ro_Interf*vs*Dr(i-1)/Visc
    as=R(i-1)*Dz(j)*(Ro_Interf*vs*(0.5+Alfa(Peclet))+Beta(Peclet)*Visc/Dr(i- &
        1))
! *** face este
    if(j.eq.Nz) then
        Visc=Prop_Interp(ViscE(i-1,Nz),Dr(i-1),ViscE(i,Nz),Dr(i))
        Ro_Interf=Prop_Interp(Ro(i-1,Nz),Dr(i-1),Ro(i,Nz),Dr(i))
    else
        Viscne=Prop_Interp(ViscE(i,j),Dz(j),ViscE(i,j+1),Dz(j+1))
        Viscse=Prop_Interp(ViscE(i-1,j),Dz(j),ViscE(i-1,j+1),Dz(j+1))
        Visc=Prop_Interp(Viscse,Dr(i-1),Viscne,Dr(i))
        Ro_NE=Prop_Interp(Ro(i,j),Dz(j),Ro(i,j+1),Dz(j+1))
        Ro_SE=Prop_Interp(Ro(i-1,j),Dz(j),Ro(i-1,j+1),Dz(j+1))
        Ro_Interf=Prop_Interp(Ro_SE,Dr(i-1),Ro_NE,Dr(i))
    end if
    ue=(U(i,j+1)+U(i-1,j+1))*0.5
    Peclet=Ro_Interf*ue*(Z(j+1)-Z(j))/Visc
    ae=sigmaz*(-Ro_Interf*ue*(0.5-Alfa(Peclet))+Beta(Peclet)*Visc/(Z(j+1)- &
        Z(j)))

```

```

! *** face oeste
      if(j.eq.1) then
        Visc=Prop_Interp(Visc_Entrada(i-1),Dr(i-1),Visc_Entrada(i),Dr(i))
        Ro_Interf=Prop_Interp(Ro_Entrada(i-1),Dr(i-1),Ro_Entrada(i),Dr(i))
      else
        Viscnw=Prop_Interp(ViscE(i,j-1),Dz(j-1),ViscE(i,j),Dz(j))
        Viscsw=Prop_Interp(ViscE(i-1,j-1),Dz(j-1),ViscE(i-1,j),Dz(j))
        Visc=Prop_Interp(Viscsw,Dr(i-1),Viscnw,Dr(i))
        Ro_NW=Prop_Interp(Ro(i,j-1),Dz(j-1),Ro(i,j),Dz(j))
        Ro_SW=Prop_Interp(Ro(i-1,j-1),Dz(j-1),Ro(i-1,j),Dz(j))
        Ro_Interf=Prop_Interp(Ro_SW,Dr(i-1),Ro_NW,Dr(i))
      end if
      uw=(U(i,j)+U(i-1,j))*0.5
      Peclet=Ro_Interf*uw*(Z(j)-Z(j-1))/Visc
      aw=sigmaz*(Ro_Interf*uw*(0.5+Alfa(Peclet))+Beta(Peclet)*Visc/(Z(j)- &
        Z(j-1)))
! *** calculo de Vch e VolDrAp ***
      Ap=(an+as+ae+aw)*(1.+1./rlxuv)
      Vch(i,j)=(an*V(i+1,j)+as*V(i-1,j)+ae*V(i,j+1)+aw*V(i,j-1)+ &
        (an+as+ae+aw)*V(i,j)/rlxuv)/Ap
      VolDrAp(i,j)=rmean*Dz(j)/Ap
    end do !j
  end do !i
! *** pontos no contorno ***
  do j=1,Nz
    Vch(Nrlp,j)=V(Nrlp,j)
    Vch(1,j)=V(1,j)
  end do
  return
end

!
  SUBROUTINE FINAL(Nr,Nz,Nrlp,Nzlp,U,V, Temperatura,Temp_Media,Temp_Externa,TWall_N,
Z,CoefHeatTransfer, &
P,VTurbK,VEpsilon,TurbK,Epsilon,TalW,FluxoCalorCond)
! *** processa resultados finais ***
!i
  dimension VTurbK(Nr),VEpsilon(Nr)
!j
  dimension Temp_Media(Nz),Temp_Externa(Nz),TWall_N(Nz),Z(0:Nzlp),CoefHeatTransfer(Nz)
, &
  TalW(Nz),FluxoCalorCond(Nz)
!i,j
  dimension U(0:Nrlp,Nzlp),V(Nrlp,0:Nzlp),Temperatura(Nr,Nz),P(Nr,Nz),TurbK(0:Nrlp,0:N
zlp),Epsilon(0:Nrlp,0:Nzlp)
  common/simpr2/ipressi,ipressj
  write(*,1000)
  write(6,1020)
  do i=1,Nr
    write(6,1010) (U(i,j),j=1,Nz,ipressj)
  end do
  write(6,1030)
  do i=1,Nr
    write(6,1010) (V(i,j),j=1,Nz,ipressj)
  end do
  write(6,1040)
  do i=1,Nr
    write(6,1010) (Temperatura(i,j),j=1,Nz,ipressj)
  end do
  write(6,1050)
  write(6,1010) (Temp_Media(j),j=1,Nz,ipressj)
  write(6,*) 'Temp_Media(50)= ',Temp_Media(50)
  write(6,1060)
  write(6,1010) (Temp_Externa(j),j=1,Nz,ipressj)
  write(6,1070)
  write(6,1010) (TWall_N(j),j=1,Nz,ipressj)
  write(6,1080)
  write(6,1010) (Z(j),j=1,Nz,ipressj)
  write(6,1090)
  write(6,1010) (CoefHeatTransfer(j),j=1,Nz,ipressj)
  write(6,*) ' FluxoCalorCond'
  write(6,1005) (FluxoCalorCond(j),j=1,Nz)
  write(9,1010) (Temp_Externa(j),j=1,Nz)
!
  write(11,*)'*** Temperatura ***'
  do i=1,Nr
    write(11,1010) (Temperatura(i,j),j=1,Nz)
  end do
  write(11,*)'*** U ***'

```

```

do i=1,Nr
  write(11,1010) (U(i,j),j=1,Nz1p)
end do
write(11,*) '*** V ***'
do i=1,Nr1p
  write(11,1010) (V(i,j),j=1,Nz)
end do
write(11,*) '*** P ***'
do i=1,Nr1p
  write(11,1010) (P(i,j),j=1,Nz)
end do
write(11,*) '*** VTurbK ***'
write(11,1010) (VTurbK(i),i=1,Nr)
write(11,*) '*** VEpsilon ***'
write(11,1010) (VEpsilon(i),i=1,Nr)
write(11,*) '*** TurbK ***'
do i=1,Nr
  write(11,1010) (TurbK(i,j),j=1,Nz)
end do
write(11,*) '*** Epsilon ***'
do i=1,Nr
  write(11,1010) (Epsilon(i,j),j=1,Nz)
end do
write(11,*) '*** TalW ***'
write(11,1010) (TalW(j),j=1,Nz)
!
return
1000 format(' *****',/, ' rodou com sucesso',/, &
' *****')
1005 format(50(e10.4,', '))
1010 format(10(e10.4,', '))
1020 format(' Matriz U')
1030 format(/, ' Matriz V')
1040 format(/, ' Temperatura')
1050 format(/, ' Temp_Media')
1060 format(/, ' Temp_Externa')
1070 format(/, ' TWall_N')
1080 format(/, ' Z')
1090 format(/, ' CoefHeatTransfer')
end
!
SUBROUTINE INICIO(Nr,Nz,Nr1p,Nz1p,Nr1m,Nz1m,Nm,AlphaMenos,AlphaMais,Coef_Mi,W, &
& R,Dr,Dr_S,Dr_N,AreaZ,VTurbK,VEpsilon,Visc_Entrada,Ro_Entrada,Gama_Entrada, &
& TWall_W,TWall_E, &
& Z,Dz,Dz_W,Dz_E,TalW,TWall_S,TWall_N,CoefHeatTransfer,QToNorthBoundary,U_Medio,
&
& Temp_Media, &
& AreaR_N,AreaR_S,Volume,P,U,V,ViscE,Viscosidade,TurbK,Epsilon,Ro,Temperatura,Cp,G
ama, &
& AIntensity,Pizao)
dimension AlphaMenos(Nm),AlphaMais(Nm),Coef_Mi(Nm),W(Nm)
dimension R(0:Nr1p),Dr(0:Nr1p),Dr_S(2:Nr),Dr_N(Nr1m),AreaZ(Nr),VTurbK(Nr),VEpsilon(Nr
),Visc_Entrada(Nr), &
& Ro_Entrada(Nr),Gama_Entrada(Nr),TWall_W(Nr),TWall_E(Nr)
dimension Z(0:Nz1p),Dz(0:Nz1p),Dz_W(2:Nz),Dz_E(Nz1m),TalW(Nz),TWall_S(Nz),TWall_N(Nz
), &
& CoefHeatTransfer(Nz),QToNorthBoundary(Nz),U_Medio(Nz),Temp_Media(Nz)
dimension AreaR_N(Nr,Nz),AreaR_S(Nr,Nz),Volume(Nr,Nz),P(Nr,Nz),U(0:Nr1p,Nz1p), &
& V(Nr1p,0:Nz1p),ViscE(Nr,Nz),Viscosidade(Nr,Nz),TurbK(0:Nr1p,0:Nz1p),Epsilon(0:Nr1p,
0:Nz1p), &
& Ro(Nr,Nz),Temperatura(Nr,Nz),Cp(Nr,Nz),Gama(Nr,Nz)
dimension AIntensity(Nr,Nz,2:Nm),Pizao(Nr,Nz,2:Nm)
character*1 SkipLine
logical flagimp
Integer Upwind_NS,Upwind_EW,Upwind_Virtual
common/fdebug/flagimp
common/simpr/niter2,npress,dnpress
common/sturb/RlxK,rlxeps,epsk,epseps
common/sturb2/cmi,te,tk
common/snewv/rlxuv,epsu,epsv,epsj,nite:
common/simpr2/1press1,1pressj
common/cmm1/AMol(5),Tc(5),Pc(5),Zc(5),AMi(5),CpA(5),CpB(5),CpC(5),CpD(5),Y(5)
common/cmm2/AK_A(5),AK_B(5),AK_C(5),AK_D(5),PesoMolecularMistura,Dr_Total
common/tue1/Pi,Sigma
common/tue2/Refraction,Extinction,Absorption,ReflectWall,EmissWall
common/tue3/Relax_Phi,Epsilon_Energy,Epsilon_Temp,MaxIterEnergy,MaxIterTemp,Relax_Temp
common/tue4/Diamond,Diamond_NS,Diamond_EW,Upwind_NS,Upwind_Virtual,Upwind_EW

```

```

data cmi,te,tk/0.09,9.,0.4/
data Pi,Sigma/3.141592654,5.6696e-8/
data Refraction,Relax_Phi,Extinction,Absorption/1.0,0.6,0.25,0.25/
data Diamond,Diamond_NS,Diamond_EW/0.5,0.5,0.5/
data Epsilon_Energy,Epsilon_Temp/0.001,0.001/
data ReflectWall,EmissWall,Relax_Temp/0.1,0.9,1./
data MaxIterEnergy,MaxIterTemp/2,2/
data Upwind_NS,Upwind_Virtual,Upwind_EW/1,1,1/
flagimp=.true.
read(5,9999) SkipLine
read(5,*) RlxK,rlxeps,rlxuv,epsk,epseps
write(6,*) 'RlxK,rlxeps,rlxuv,epsk,epseps ', &
& RlxK,rlxeps,rlxuv,epsk,epseps
read(5,9999) SkipLine
read(5,*) niter,epsp,epsu,epsv
write(6,*) 'niter,epsp,epsu,epsv ',niter,epsp,epsu,epsv
read(5,9999) SkipLine
read(5,*) niter2,npress,dnpress,ipressi,ipressj
write(6,*) 'niter2,npress,dnpress,ipressi,ipressj ', &
& niter2,npress,dnpress,ipressi,ipressj
read(5,9999) SkipLine
read(5,*) (Dr(i),i=1,Nr)
write(6,*) 'Dr ',Dr
Dr_Total=0.
do i=1,Nr
  Dr_Total=Dr_Total+Dr(i)
end do
write(6,*) 'Dr_Total ',Dr_Total
read(5,9999) SkipLine
read(5,*) (Dz(j),j=1,Nz)
write(6,*) 'Dz ',Dz
Dz_Total=0.
do j=1,Nz
  Dz_Total=Dz_Total+Dz(j)
end do
Dr(0)=Dr(1)
Dr(Nr1p)=Dr(Nr)
Dz(0)=Dz(1)
Dz(Nz1p)=Dz(Nz)
! ***
cotas ***
R(1)=0.5*Dr(1)
R(0)=-R(1)
do i=2,Nr1p
  R(i)=R(i-1)+(Dr(i-1)+Dr(i))*0.5
end do
Z(1)=0.5*Dz(1)
Z(0)=-Z(1)
do j=2,Nz1p
  Z(j)=Z(j-1)+(Dz(j-1)+Dz(j))*0.5
end do
do i=2,Nr
  Dr_S(i)=0.5*(Dr(i-1)+Dr(i))
end do
do i=1,Nr-1
  Dr_N(i)=Dr_S(i+1)
end do
do j=2,Nz
  Dz_W(j)=0.5*(Dz(j-1)+Dz(j))
end do
do j=1,Nz-1
  Dz_E(j)=Dz_W(j+1)
end do
do i=1,Nr
  AreaZ(i)=2.*Pi*R(i)*Dr(i)
  Aux1=2.*Pi*(R(i)+0.5*Dr(i))
  Aux2=2.*Pi*(R(i)-0.5*Dr(i))
  do j=1,Nz
    AreaR_N(i,j)=Aux1*Dz(j)
    AreaF_S(i,j)=Aux2*Dz(j)
    Volume(i,j)=AreaZ(i)*Dz(j)
  end do
end do
do m=2,Nm
  do j=1,Nr
    do j=1,Nz
      Pizao(i,j,m)=(AreaR_N(i,j)-AreaF_S(i,j))/W(m)
    end do
  end do
end do

```

```

end do
! *** Alphammenos
  AlphaMenos(1)=0.
  do m=1,Nm
    if (m.eq.Nm) then
      AlphaMais(m)=0.
    else
      AlphaMais(m)=AlphaMenos(m)+W(m)*Coef_Mi(m)
      AlphaMenos(m+1)=AlphaMais(m)
    end if
  end do
! *** Condição inicial
  read(10,*) (TWall_N(j),j=1,Nz)
  TWall_W=1223.15
  read(7,9999) SkipLine
  do i=1,Nr
    read(7,*) (Temperatura(i,j),j=1,Nz)
  end do
  TWall_S(:)=Temperatura(1,:)
  TWall_E(:)=Temperatura(:,Nz)
! Ro_Entrada,Gama_Entrada e demais propriedades de transporte
  do i=1,Nr
    Temp=TWall_W(i)
    call Propriedades(Temp,ViscMistura,CondTermicaMistura,CpMistura,Densidade)
    Ro_Entrada(i)=Densidade
  end do
  do i=1,Nr
    do j=1,Nz
      Temp=Temperatura(i,j)
      call propriedades(Temp,ViscMistura,CondTermicaMistura,CpMistura,Densidade)
      Viscosidade(i,j)=ViscMistura
      Ro(i,j)=Densidade
      Cp(i,j)=CpMistura
    end do
  end do
! *** Temperatura média
  do j=1,Nz
    Soma_Numerador=0.
    Soma_Denominador=0.
    do i=1,Nr
      Soma_Numerador=Soma_Numerador+Ro(i,j)*Volume(i,j)*Cp(i,j)*Temperatura(i,j)
      Soma_Denominador=Soma_Denominador+Ro(i,j)*Volume(i,j)*Cp(i,j)
    end do
    Temp_Media(j)=Soma_Numerador/Soma_Denominador
  end do
! *** Velocidade média
  do j=1,Nz
    Soma_Numerador=0.
    Soma_Denominador=0.
    do i=1,Nr
      Soma_Numerador=Soma_Numerador+Ro(i,j)*Volume(i,j)*U(i,j)
      Soma_Denominador=Soma_Denominador+Ro(i,j)*Volume(i,j)
    end do
    U_Medio(j)=Soma_Numerador/Soma_Denominador
  end do
  do j=1,Nz
    call propriedades(Temp_Media(j),ViscMistura,CondTermicaMistura,CpMistura,Densidad
e)
    Reynolds=Densidade*U_Medio(j)*(2.*Dr_Total)/ViscMistura
    Prandtl=CpMistura*ViscMistura/CondTermicaMistura
    CoefHeatTransfer(j)=0.023*CondTermicaMistura/(2.*Dr_Total)*Reynolds**0.8*Prandtl*
*0.3
  end do
  QToSoutBoundary=0.
  QToNorthBoundary=0.
  do m=1,Nm
    AIntensity(1,:,m)=Refraction**2*Sigma*Temperatura(1,:)**4/Pi
  end do
! *** tentativa do campo de velocidades ***
  read(7,9999) SkipLine
  do i=1,Nr
    read(7,*) (U(i,j),j=1,Nzlp)
  end do
  read(7,9999) SkipLine
  do i=1,Nr1p
    read(7,*) (V(i,j),j=1,Nz)
  end do
! *** tentativa do campo de pressoes ***

```

```

read(7,9999) SkipLine
do i=1,Nr1p
  read(7,*) (P(i,j),j=1,Nz)
end do
! *** velocidades virtuais calculadas pelas condicoes de contorno ***
! *** para u - cascas norte/sul ***
do j=2,Nz
  U(Nr1p,j)=-U(Nr,j)
  U(0,j)=U(1,j)
end do
! *** para v - cascas este/oeste ***
do i=2,Nr
  V(i,Nz1p)=-V(i,Nz)
  V(i,0)=-V(i,1)
end do
read(7,9999) SkipLine
read(7,*) (VTurbK(i),i=1,Nr)
read(7,9999) SkipLine
read(7,*) (VEpsilon(i),i=1,Nr)
read(7,9999) SkipLine
do i=1,Nr
  read(7,*) (TurbK(i,j),j=1,Nz)
end do
read(7,9999) SkipLine
do i=1,Nr
  read(7,*) (Epsilon(i,j),j=1,Nz)
end do
read(7,9999) SkipLine
read(7,*) (TalW(j),j=1,Nz)
do i=1,Nr
  Visc_Entrada(i)=Viscosidade(i,j)+0.09*Ro(i,j)*VTurbK(i)**2/VEpsilon(i)
end do
do i=1,Nr
  do j=1,Nz
    ViscE(i,j)=Viscosidade(i,j)+0.09*Ro(i,j)*TurbK(i,j)**2/Epsilon(i,j)
  end do
end do
return
9999 format(a1)
end
!
SUBROUTINE keps1(Nr,Nr1p,Nr1m,Nz,Nz1p,R,Dr,VTurbK,Visc_Entrada,Ro_Entrada,Z,Dz,Ta
1W,U,V,AuxMtx, &
  Viscosidade,ViscE,TurbK,Epsilon,Ro,Gk,Can,Cas,Cae,Caw)
  external Alfa,Beta
  dimension R(0:Nr1p),Dr(0:Nr1p),VTurbK(Nr),Visc_Entrada(Nr),Ro_Entrada(Nr)
  dimension Z(0:Nz1p),Dz(0:Nz1p),TalW(Nz)
  dimension U(0:Nr1p,Nz1p),V(Nr1p,0:Nz1p),AuxMtx(0:Nr1p,0:Nz1p),Viscosidade(Nr,Nz), &
  & ViscE(Nr,Nz),TurbK(0:Nr1p,0:Nz1p),Epsilon(0:Nr1p,0:Nz1p),Ro(Nr,Nz),Gk(Nr,Nz), &
  & Can(Nr,Nz),Cas(Nr,Nz),Cae(Nr,Nz),Caw(Nr,Nz)
  logical flagimp
  common/fdebug/flagimp
  common/sturb/RlxK,rlxeps,epsk,epseps
  common/sturb2/cmi,te,tk
  common/snewv/rlxuv,epsu,epsv,epsn,nter
! *** calculo dos coeficientes ***
do i=1,Nr
  deltrn=R(i+1)-R(i)
  deltrs=R(i)-R(i-1)
  rintn=R(i)+Dr(i)*0.5
  rintS=R(i)-Dr(i)*0.5
  areaz=R(i)*Dr(i)
  do j=1,Nz
    deltze=Z(j+1)-Z(j)
    deltzW=Z(j)-Z(j-1)
! *** face norte ***
    if(i.eq.Nr) then
      Visc=Viscosidade(Nr,j)
      Ro_Interf=Ro(Nr,j)
    else
      Visc=Prop_Interp(ViscE(1,j),Dr(i),ViscE(1+1,j),Dr(1+1))
      Ro_Interf=Prop_Interp(Ro(1,j),Dr(1),Ro(1+1,j),Dr(1+1))
    end if
    sigma=Visc
    Peclet=Ro_Interf*V(1+1,j)*deltrn/sigma
    Can(i,j)=rintn*Dz(j)*(-Ro_Interf*V(i+1,j)*(0.5-Alfa(Peclet))+ &
    Beta(Peclet)*sigma/deltrn)
! *** face sul ***

```



```

if(i.eq.1)then
  Visc=ViscE(1,j)
  Ro_Interf=Ro(1,j)
else
  Visc=Prop_Interp(ViscE(i-1,j),Dr(i-1),ViscE(i,j),Dr(i))
  Ro_Interf=Prop_Interp(Ro(i-1,j),Dr(i-1),Ro(i,j),Dr(i))
end if
sigma=Visc
Peclet=Ro_Interf*V(i,j)*deltrs/sigma
Cas(i,j)=rints*Dz(j)*(Ro_Interf*V(i,j)*(0.5+Alfa(Peclet))+ &
  Beta(Peclet)*sigma/deltrs)
! *** face este ***
if(j.eq.Nz) then
  Visc=ViscE(i,Nz)
  Ro_Interf=Ro(i,Nz)
else
  Visc=Prop_Interp(ViscE(i,j),Dz(j),ViscE(i,j+1),Dz(j+1))
  Ro_Interf=Prop_Interp(Ro(i,j),Dz(j),Ro(i,j+1),Dz(j+1))
end if
sigma=Visc
Peclet=Ro_Interf*U(i,j+1)*deltz/sigma
Cae(i,j)=areaz*(-Ro_Interf*U(i,j+1)*(0.5+Alfa(Peclet))+ &
  Beta(Peclet)*sigma/deltz)
! *** face oeste ***
if(j.eq.1) then
  Visc=Visc_Entrada(i)
  Ro_Interf=Ro_Entrada(i)
else
  Visc=Prop_Interp(ViscE(i,j-1),Dz(j-1),ViscE(i,j),Dz(j))
  Ro_Interf=Prop_Interp(Ro(i,j-1),Dz(j-1),Ro(i,j),Dz(j))
end if
sigma=Visc
Peclet=Ro_Interf*U(i,j)*deltzw/sigma
Caw(i,j)=areaz*(Ro_Interf*U(i,j)*(0.5+Alfa(Peclet))+ &
  Beta(Peclet)*sigma/deltzw)
duDz=(U(i,j+1)-U(i,j))/Dz(j)
dvDr=(V(i+1,j)-V(i,j))/Dr(i)
if(i.eq.Nr) then
  un=0.
else
  un=0.5*((U(i+1,j)+U(i+1,j+1)-U(i,j)-U(i,j+1))*Dr(i)/(Dr(i+1)+ &
  Dr(i))+U(i,j)+U(i,j+1))
end if
if(i.eq.1) then
  us=0.5*(U(1,j)+U(1,j+1))
else
  us=0.5*((U(i,j)+U(i,j+1)-U(i-1,j)-U(i-1,j+1))*Dr(i-1)/ &
  (Dr(i-1)+Dr(i))+U(i-1,j)+U(i-1,j+1))
end if
duDr=(un-us)/Dr(i)
if(j.eq.Nz) then
  ve=0.5*(V(i,j)+V(i+1,j))
else
  ve=0.5*((V(i,j+1)+V(i+1,j+1)-V(i,j)-V(i+1,j))*Dz(j)/ &
  (Dz(j)+Dz(j+1))+V(i,j)+V(i+1,j))
end if
if(j.eq.1) then
  vw=0.
else
  vw=0.5*((V(i,j)+V(i+1,j)-V(i,j-1)-V(i+1,j-1))*Dz(j-1)/ &
  (Dz(j-1)+Dz(j))+V(i,j-1)+V(i+1,j-1))
end if
dvDz=(ve-vw)/Dz(j)
Gk(i,j)=(ViscE(i,j)-Viscosidade(i,j))*(2.*(duDz**2+dvDr**2+((V(i+1,j) &
  +V(1,j))*0.5/r(i))**2)+(duDr+dvDz)**2)
end do
end do
do laux=1,niter
  do j=1,Nr
    do j=1,Nz
      AuxMtx(i,j)=TurbK(1,j)
    end do
  end do
! *** colocacao dos valores dos ptos. virtuais pelas condicoes de contorno
! *** cascas norte/sul ***
  do j=1,Nz
    TurbK(Nr1p,j)=TurbK(Nr,j)
    TurbK(0,j)=TurbK(1,j)
  end do
end do

```



```

end do
! *** casca este/oeste ***
do i=1,Nr
  TurbK(i,Nzlp)=TurbK(i,Nz)
  TurbK(i,0)=2.*VTurbK(i)-TurbK(i,1)
end do
! *** calculo do novo valor ***
do j=1,Nz
  us=0.5*(U(Nr,j)+U(Nr,j+1)-U(Nrlm,j)-U(Nrlm,j+1))*Dr(Nrlm)/ &
    (Dr(Nrlm)+Dr(Nr))+U(Nrlm,j)+U(Nrlm,j+1)
  Ap0=cmi*Ro(i,j)**2*TurbK(Nr,j)*us*r(Nr)*Dz(j)/TalW(j)
  Ap0=Ap0+Can(Nr,j)+Cas(Nr,j)+Cae(Nr,j)+Caw(Nr,j)
  Ap=Ap0*(1.+1./RlxK)
  cb=TalW(j)*us*r(Nr)*Dz(j)+Ap0*TurbK(Nr,j)/RlxK
  TurbK(Nr,j)=(cb+Can(Nr,j)*TurbK(Nrlp,j)+Cas(Nr,j)*TurbK(Nrlm,j) &
    +Cae(Nr,j)*TurbK(Nr,j+1)+Caw(Nr,j)*TurbK(Nr,j-1))/Ap
end do
do i=Nrlm,1,-1
  do j=1,Nz
    delv=R(i)*Dr(i)*Dz(j)
    p0=Can(i,j)+Cas(i,j)+Cae(i,j)+Caw(i,j)
    Ap=Ap0*(1.+1./RlxK)
    cb=(Gk(i,j)-Ro(i,j)*Epsilon(i,j))*delv+Ap0*TurbK(i,j)/RlxK
    TurbK(i,j)=(cb+Can(i,j)*TurbK(i+1,j)+Cas(i,j)*TurbK(i-1,j) &
      +Cae(i,j)*TurbK(i,j+1)+Caw(i,j)*TurbK(i,j-1))/Ap
  end do
end do
! *** verifica convergencia ***
do i=1,Nr
  do j=1,Nz
    if(dabs(AuxMtx(i,j)-TurbK(i,j))/AuxMtx(i,j)).gt.epsk) go to 60
  end do
end do
! *** convergiu ***
if(flagimp) print 1010, iaux
return
60 end do
print 1000
1000 format('                                K - numero max. de itera
coes')
1010 format('                                K :',i3,' iteracoes')
return
end
!
SUBROUTINE keps2(Nr,Nrlp,Nrlm,Nz,Nzlp,R,Dr,VEpsilon,Visc_Entrada,Ro_Entrada,Z,Dz,U
,V,AuxMtx,Viscosidade, &
& ViscE,TurbK,Epsilon,Ro,Gk,Can,Cas,Cae,Caw)
external Alfa,Beta
dimension R(0:Nrlp),Dr(0:Nrlp),VEpsilon(Nr),Visc_Entrada(Nr),Ro_Entrada(Nr)
dimension Z(0:Nzlp),Dz(0:Nzlp)
dimension U(0:Nrlp,Nzlp),V(Nrlp,0:Nzlp),AuxMtx(0:Nrlp,0:Nzlp),Viscosidade(Nr,Nz), &
& ViscE(Nr,Nz),TurbK(0:Nrlp,0:Nzlp),Epsilon(0:Nrlp,0:Nzlp),Ro(Nr,Nz),Gk(Nr,Nz), &
& Can(Nr,Nz),Cas(Nr,Nz),Cae(Nr,Nz),Caw(Nr,Nz)
logical flagimp
common/fdebug/flagimp
common/sturb/RlxK,rlxeps,epsk,epseps
common/sturb2/cmi,te,tk
common/snewv/rlxuv,epsu,epsv,epsn,iter
! *** calculo dos coeficientes ***
do i=1,Nrlm
  deltrn=R(i+1)-R(i)
  deltrs=R(i)-R(i-1)
  rintn=R(i)+Dr(i)*0.5
  rints=R(i)-Dr(i)*0.5
  areaz=R(i)*Dr(i)
  do j=1,Nz
    deltze=Z(j+1)-Z(j)
    deltzw=Z(j)-Z(j-1)
! *** face norte ***
    Visc=Prop_Interp(ViscE(1,j),Dr(i),ViscE(1+1,j),Dr(1+1))
    Ro_Interf=Prop_Interp(Ro(i,j),Dr(i),Ro(1+1,j),Dr(1+1))
    sigma=Visc/1.21
    Peclet=Ro_Interf*V(i+1,j)*deltrn/sigma
    Can(i,j)=rintn*Dz(j)*(Ro_Interf*V(i+1,j)*(0.5-Alfa(Peclet))+ &
      Beta(Peclet)*sigma/deltrn)
! *** face sul ***
    if(i.eq.1)then
      Visc=ViscE(1,j)

```

```

      Ro_Interf=Ro(1,j)
    else
      Visc=Prop_Interp(ViscE(i-1,j),Dr(i-1),ViscE(i,j),Dr(i))
      Ro_Interf=Prop_Interp(Ro(i-1,j),Dr(i-1),Ro(i,j),Dr(i))
    end if
    sigma=Visc/1.21
    Peclet=Ro_Interf*V(i,j)*deltres/sigma
    Cas(i,j)=rints*Dz(j)*(Ro_Interf*V(i,j)*(0.5+Alfa(Peclet))+ &
      Beta(Peclet)*sigma/deltres)
! *** face este ***
    if(j.eq.Nz) then
      Visc=ViscE(i,Nz)
      Ro_Interf=Ro(i,Nz)
    else
      Visc=Prop_Interp(ViscE(i,j),Dz(j),ViscE(i,j+1),Dz(j+1))
      Ro_Interf=Prop_Interp(Ro(i,j),Dz(j),Ro(i,j+1),Dz(j+1))
    end if
    sigma=Visc/1.21
    Peclet=Ro_Interf*U(i,j+1)*deltze/sigma
    Cae(i,j)=areaz*(-Ro_Interf*U(i,j+1)*(0.5-Alfa(Peclet))+ &
      Beta(Peclet)*sigma/deltze)
! *** face oeste ***
    if(j.eq.1) then
      Visc=Visc_Entrada(i)
      Ro_Interf=Ro_Entrada(i)
    else
      Visc=Prop_Interp(ViscE(i,j-1),Dz(j-1),ViscE(i,j),Dz(j))
      Ro_Interf=Prop_Interp(Ro(i,j-1),Dz(j-1),Ro(i,j),Dz(j))
    end if
    sigma=Visc/1.21
    Peclet=Ro_Interf*U(i,j)*deltzw/sigma
    Caw(i,j)=areaz*(Ro_Interf*U(i,j)*(0.5+Alfa(Peclet))+ &
      Beta(Peclet)*sigma/deltzw)
!
    duDz=(U(i,j+1)-U(i,j))/Dz(j)
    dvDr=(V(i+1,j)-V(i,j))/Dr(i)
!
    un=0.5*((U(i+1,j)+U(i+1,j+1)-U(i,j)-U(i,j+1))*Dr(i)/(Dr(i+1)+ &
      Dr(i))+U(i,j)+U(i,j+1))
    if(i.eq.1) then
      us=0.5*(U(1,j)+U(1,j+1))
    else
      us=0.5*((U(i,j)+U(i,j+1)-U(i-1,j)-U(i-1,j+1))*Dr(i-1)/ &
      (Dr(i-1)+Dr(i))+U(i-1,j)+U(i-1,j+1))
    end if
    duDr=(un-us)/Dr(i)
!
    if(j.eq.Nz) then
      ve=0.5*(V(i,j)+V(i+1,j))
    else
      ve=0.5*((V(i,j+1)+V(i+1,j+1)-V(i,j)-V(i+1,j))*Dz(j)/ &
      (Dz(j)+Dz(j+1))+V(i,j)+V(i+1,j))
    end if
    if(j.eq.1) then
      vw=0.
    else
      vw=0.5*((V(i,j)+V(i+1,j)-V(i,j-1)-V(i+1,j-1))*Dz(j-1)/ &
      (Dz(j-1)+Dz(j))+V(i,j-1)+V(i+1,j-1))
    end if
    dvDz=(ve-vw)/Dz(j)
    Gk(1,j)=(ViscE(1,j)-Viscosidade(i,j))*(2.*(duDz**2+dvDr**2+(V(i+1,j) &
      +V(i,j))*0.5/r(i))**2)+(duDr+dvDz)**2)
  end do
end do
do laux=1,niter
  do i=1,Nr
    do j=1,Nz
      AuxMtx(1,j)=Epsilon(i,j)
    end do
  end do
! *** colocacao dos valores dos ptos. virtuais pelas condicoes de contorno
! *** cascas norte/sul ***
  do j=1,Nz
    Epsilon(Nr,j)=cmu**0.75*TurbK(Nr,j)**1.5*2./(tk*Dr(Nr,j))
    Epsilon(0,j)=Epsilon(1,j)
  end do
! *** casca este/oeste ***
  do i=1,Nr

```

```

        Epsilon(i,Nz1p)=Epsilon(i,Nz)
        Epsilon(i,0)=2.*VEpsilon(i)-Epsilon(i,1)
    end do
! *** calculo do novo valor ***
    do i=Nrlm,1,-1
        do j=1,Nz
            delv=R(i)*Dr(i)*Dz(j)
            Ap0=Can(i,j)+Cas(i,j)+Cae(i,j)+Caw(i,j)+ &
                (1.92*Ro(i,j)*Epsilon(i,j)/TurbK(i,j))*delv
            Ap=Ap0*(1.+1./rlxeps)
            cb=1.44*Gk(i,j)*Epsilon(i,j)/TurbK(i,j)
            cb=cb*delv+Ap0*Epsilon(i,j)/rlxeps
            Epsilon(i,j)=(cb+Can(i,j)*Epsilon(i+1,j)+Cas(i,j)*Epsilon(i-1,j) &
                +Cae(i,j)*Epsilon(i,j+1)+Caw(i,j)*Epsilon(i,j-1))/Ap
        end do
    end do
! *** verifica convergencia ***
    do i=1,Nr
        do j=1,Nz
            if(dabs((AuxMtx(i,j)-Epsilon(i,j))/AuxMtx(i,j)).gt.epseps) go to 60
        end do
    end do
! *** convergiu ***
    if(flagimp) print 1010, iaux
    return
+ 60 end do
    print 1000
    1000 format('
e iteracoes')
    1010 format('
)
    return
end
!
SUBROUTINE NEWVEL(Nr,Nz,Nrlp,Nz1p,flaguv,Dr,TalW,P,U,Uch,V,Vch,VolDzAp,VolDrAp, &
    AuxMtx,TurbK,Ro,Viscosidade)
! *** calcula novos valores das velocidades e verifica convergencia final ***
dimension Dr(0:Nrlp)
dimension TalW(Nz)
dimension P(Nr,Nz),U(0:Nrlp,Nz1p),Uch(Nr,Nz1p),V(Nrlp,0:Nz1p),Vch(Nrlp,Nz), &
    & VolDzAp(Nr,2:Nz),VolDrAp(2:Nr,Nz),AuxMtx(0:Nrlp,0:Nz1p),TurbK(0:Nrlp,0:Nz1p), &
    & Ro(Nr,Nz),Viscosidade(Nr,Nz)
logical flaguv
common/sturb2/cmi,te,tk
common/snewv/rlxuv,epsu,epsv,epsp,niter
! *** correcao das velocidades ***
flaguv=.false.
do i=1,Nr
    do j=1,Nz1p
        AuxMtx(i,j)=U(i,j)
    end do
end do
! *** correcao de u nos pontos internos ***
do i=1,Nr
    do j=2,Nz
        if(i.eq.Nr) VolDzAp(i,j)=0.
        U(i,j)=Uch(i,j)-VolDzAp(i,j)*(P(i,j)-P(i,j-1))
    end do
end do
! *** correcao de u na casca este
do i=1,Nr
    U(i,Nz1p)=U(i,Nz)
end do
! *** verificacao da convergencia de u ***
do i=1,Nr
    do j=1,Nz1p
        if(dabs(AuxMtx(i,j)-U(i,j)).gt.epsu) then
            flaguv=.true.
            go to 150
        end if
    end do
end do
do i=1,Nrlp
    do j=1,Nz
        AuxMtx(i,j)=V(i,j)
    end do
end do
! *** correcao de v nos pontos internos ***

```

Epsilon - numero maximo d

Epsilon:',i3,' iteracoes'

```

150 do i=2,Nr
    do j=1,Nz
        V(i,j)=Vch(i,j)-VolDrAp(i,j)*(P(i,j)-P(i-1,j))
    end do
end do
! *** verificacao da convergencia de v ***
if(flaguv) go to 180
do i=1,Nr1p
    do j=1,Nz
        if(dabs(AuxMtx(i,j)-V(i,j)).gt.epsv) then
            flaguv=.true.
            return
        end if
    end do
end do
!
! *** velocidades virtuais calculadas pelas condicoes de contorno ***
! *** para u - cascas norte/sul ***
180 do j=2,Nz
    U(Nr1p,j)=-U(Nr,j)
    U(0,j)=U(1,j)
end do
! *** para v - cascas este/oeste ***
do i=2,Nr
    V(i,Nz1p)=-V(i,Nz)
    V(i,0)=-V(i,1)
end do
do j=1,Nz
    up=(U(Nr,j)+U(Nr,j+1))*0.5
    ymais=cmi**0.25*dsqrt(TurbK(Nr,j))*Ro(Nr,j)*Dr(Nr)/(2.*Viscosidade(Nr,j))
    TalW(j)=cmi**0.25*dsqrt(TurbK(Nr,j))*up*Ro(Nr,j)*tk/dlog(te*ymais)
    if(ymais.lt.11.5) TalW(j)=Viscosidade(Nr,j)*up*2./Dr(Nr)
end do
return
end
!
SUBROUTINE PRESSAO(Nr,Nz,Nr1p,Nz1p,R,Dr,Dz,ATdma,BTdma,CTdma,DTdma, &
XTdma,BeTdma,GaTdma,P,Uch,Vch,VoldZAp,VoldrAp,AuxMtx,Ro)
! *** converge o campo de pressoes para dado campo de velocidades ***
dimension R(0:Nr1p),Dr(0:Nr1p)
dimension Dz(0:Nz1p),ATdma(Nz),BTdma(Nz),CTdma(Nz),DTdma(Nz), &
& XTdma(Nz),BeTdma(Nz),GaTdma(Nz)
dimension P(Nr,Nz),Uch(Nr,Nz1p),Vch(Nr1p,Nz),VoldZAp(Nr,2:Nz), &
& VoldrAp(2:Nr,Nz),AuxMtx(0:Nr1p,0:Nz1p),Ro(Nr,Nz)
logical flagimp
common/fdebug/flagimp
common/snewv/rlxuv,epsu,epsv,epsn,niter
do iaux=1,niter
    do i=1,Nr
        do j=1,Nz
            AuxMtx(i,j)=P(i,j)
        end do
    end do
! *** corrige pressoes linha a linha ***
do i=1,Nr
    rintn=R(i)+Dr(i)*0.5
    rintS=R(i)-Dr(i)*0.5
    sigmaz=R(i)*Dr(i)
    do j=1,Nz
        if(i.eq.1.and.j.eq.Nz) then
            ATdma(j)=1.
            BTdma(j)=0.
            CTdma(j)=0.
            DTdma(j)=0.
            go to 10
        end if
! *** face norte ***
auxn=Ro(i,j)*rintn*Dz(j)
if(i.eq.Nr) then
    an=0.
    anpn=0.
else
    an=auxn*VoldrAp(i+1,j)
    anpn=an*P(i+1,j)
end if
! *** face sul ***
auxs=Ro(i,j)*rintS*Dz(j)
if(i.eq.1) then

```

```

        as=0.
        asps=0.
    else
        as=auxs*VolDrAp(i,j)
        asps=as*P(i-1,j)
    end if
! *** face este *
    auxe=Ro(i,j)*sigmaz
    if(j.eq.Nz) then
        ae=0.
    else
        ae=auxe*VolDzAp(i,j+1)
    end if
! *** face oeste ***
    auxw=auxe
    if(j.eq.1) then
        aw=0.
    else
        aw=auxw*VolDzAp(i,j)
    end if
! *** correcao ***
    b=-auxn*Vch(i+1,j)+auxs*Vch(i,j)-auxe*Uch(i,j+1)+auxw*Uch(i,j)
    ATdma(j)=an+as+ae+aw
    BTdma(j)=ae
    CTdma(j)=aw
    DTdma(j)=b+anpn+asps
10 end do
end do
call Tdma(Nz,Nzlm,ATdma,BTdma,CTdma,DTdma,XTdma,BeTdma,GaTdma)
do j=1,Nz
    P(i,j)=XTdma(j)
end do
! *** corrige campo de pressoes coluna a coluna ***
do j=1,Nz
    do i=1,Nr
        if(j.eq.Nz.and.i.eq.1) then
            ATdma(i)=1.
            BTdma(i)=0.
            CTdma(i)=0.
            DTdma(i)=0.
            go to 30
        end if
! *** face norte ***
        auxn=Ro(i,j)*(R(i)+Dr(i)*0.5)*Dz(j)
        if(i.eq.Nr) then
            an=0.
        else
            an=auxn*VolDrAp(i+1,j)
        end if
! *** face sul ***
        auxs=Ro(i,j)*(R(i)-Dr(i)*0.5)*Dz(j)
        if(i.eq.1) then
            as=0.
        else
            as=auxs*VolDrAp(i,j)
        end if
! *** face este *
        auxe=Ro(i,j)*r(i)*Dr(i)
        if(j.eq.Nz) then
            ae=0.
            aepe=0.
        else
            ae=auxe*VolDzAp(i,j+1)
            aepe=ae*P(i,j+1)
        end if
! *** face oeste ***
        auxw=auxe
        if(j.eq.1) then
            aw=0.
            awpw=0.
        else
            aw=auxw*VolDzAp(i,j)
            awpw=aw*P(i,j-1)
        end if
! *** correcao ***
        b=-auxn*Vch(i+1,j)+auxs*Vch(i,j)-auxe*Uch(i,j+1)+auxw*Uch(i,j)
        ATdma(i)=an+as+ae+aw
        BTdma(i)=an

```

```

        CTdma(i)=as
        DTdma(i)=b+aepe+awpw
30    end do
        call Tdma(Nr,Nrlm,ATdma,BTdma,CTdma,DTdma,XTdma,BeTdma,GaTdma)
        do i=1,Nr
            P(i,j)=XTdma(i)
        end do
    end do
! *** verifica convergencia ***
    do i=1,Nr
        do j=1,Nz
            if(dabs(AuxMtx(i,j)-P(i,j)).gt.epsp) go to 70
        end do
    end do
    if(flagimp) print 1010,iaux
    return
70    end do
    print 1000
1000 format('                Pressao - numero maximo de iteracoes')
1010 format('                Pressao:',i3,' iteracoes')
    stop
    end
!
    Subroutine Tdma(N,N1M,A,B,C,D,X,BETA,GAMA)
    Dimension A(N),B(N),C(N),D(N),X(N),Beta(N),Gama(N)
    Beta(1)=A(1)
    Gama(1)=D(1)/Beta(1)
    do i=2,N
        Beta(i)=A(i)-C(i)*B(i-1)/Beta(i-1)
        Gama(i)=(C(i)*Gama(i-1)+D(i))/Beta(i)
    end do
    X(N)=Gama(N)
    do k=1,N1M
        i=N-k
        X(i)=B(i)*X(i+1)/Beta(i)+Gama(i)
    end do
    return
    end
!
    SUBROUTINE TURBUL(Nr,Nrlp,Nz,Nzlp,R,Dr,VTurbK,VEpsilon,Visc_Entrada,Ro_Entrada,Z,D
z,TalW,U,V,AuxMtx, &
    Viscosidade,ViscE,TurbK,Epsilon,Ro,Gk,Can,Cas,Cae,Caw)
    dimension R(0:Nrlp),Dr(0:Nrlp),VTurbK(Nr),VEpsilon(Nr),Visc_Entrada(Nr),Ro_Entrada(Nr
)
    dimension Z(0:Nzlp),Dz(0:Nzlp),TalW(Nz)
    dimension U(0:Nrlp,Nzlp),V(Nrlp,0:Nzlp),AuxMtx(0:Nrlp,0:Nzlp),Viscosidade(Nr,Nz), &
    & ViscE(Nr,Nz),TurbK(0:Nrlp,0:Nzlp),Epsilon(0:Nrlp,0:Nzlp),Ro(Nr,Nz),Gk(Nr,Nz), &
    & Can(Nr,Nz),Cas(Nr,Nz),Cae(Nr,Nz),Caw(Nr,Nz)
    common/sturb/RlxK,rlxeps,epsk,epseps
    common/sturb2/cmi,te,tk
    common/snewv/rlxuv,epsu,epsv,eps, niter
    call keps1(Nr,Nrlp,Nrlm,Nz,Nzlp,R,Dr,VTurbK,Visc_Entrada,Ro_Entrada,Z,Dz,TalW,U,V
,AuxMtx, &
    Viscosidade,ViscE,TurbK,Epsilon,Ro,Gk,Can,Cas,Cae,Caw)
    call keps2(Nr,Nrlp,Nrlm,Nz,Nzlp,R,Dr,VEpsilon,Visc_Entrada,Ro_Entrada,Z,Dz,U,V,Aux
Mtx,Viscosidade, &
    & ViscE,TurbK,Epsilon,Ro,Gk,Can,Cas,Cae,Caw)
    do i=1,Nr
        do j=1,Nz
            Visct=0.09*Ro(1,j)*TurbK(1,j)**2/Epsilon(i,j)
            ViscE(1,j)=Viscosidade(i,j)+Visct
        end do
    end do
    return
    end
!
    subroutine ENERGY(Nr,Nrlp,Nrlm,Nz,Nzlp,Nzlm,Nm,Flaq_TFinal, &
    & Coef_Mi,Coef_Zeta,W,AlphaMenos,AlphaMais,M_Simetrico, &
    & Dr,Dr_N,Dr_S,AreaZ,TWall_W,TWall_E,Ro_Entrada, &
    & Dz,Dz_E,Dz_W,TWall_N,Temp_Media,Temp_Externa,U_Medio,FluxoCalorCond, &
    & QToNorthBoundary,CoefHeatTransfer, &
    & U,V,AreaR_N,AreaR_S,Volume, Temperatura, Ro, Cp, Gama, Viscosidade, &
    & TurbK, Epsilon, &
    & AIntensity, Pizao)
    external Alfa,Beta
    keal Coef_Mi(Nm),Coef_Zeta(Nm),W(Nm),AlphaMenos(Nm),AlphaMais(Nm)
    integer M_Simetrico(Nm)
    Dimension AreaZ(Nr)

```

```

Real Dr(Nr),Dr_N(Nr1m),Dr_S(2:Nr),IFromWestBoundary(Nr), &
IFromEastBoundary(Nr),INorth(0:Nr),ISouth(Nr1p),IEast(Nr),IWest(Nr), &
TWall_W(Nr),TWall_E(Nr),Phi_2(Nr),Phi_3(Nr),Phi_4(Nr),Teta_2(Nr),Teta_3(Nr),Teta_4(N
r), &
Ro_Entrada(Nr),Gama_Entrada(Nr)
Real Dz(Nz),Dz_E(Nz1m),Dz_W(2:Nz),IFromSouthBoundary(Nz),IFromNorthBoundary(Nz), &
QToNorthBoundary(Nz),QToSouthBoundary(Nz),TWall_N(Nz),CoefHeatTransfer(Nz), &
Temp_Media(Nz),Temp_Externa(Nz),U_Medio(Nz),FluxoCalorCond(Nz)
Real IToSouthBoundary(Nz,2:Nm),IToNorthBoundary(Nz,2:Nm)
Real U(0:Nr1p,Nz1p),V(Nr1p,0:Nz1p),AreaR_N(Nr,Nz),AreaR_S(Nr,Nz),Volume(Nr,Nz), &
Source(Nr,Nz),RadMenosMeio(Nr,Nz),Sp(Nr,Nz),Sc(Nr,Nz),DivQr(Nr,Nz), &
Phi(Nr,Nz),NewPhi(Nr,Nz),Temperatura(Nr,Nz),Ro(Nr,Nz),Cp(Nr,Nz),Gama(Nr,Nz), &
Viscosidade(Nr,Nz),TurbK(0:Nr1p,0:Nz1p),Epsilon(0:Nr1p,0:Nz1p;
Real AIntensity(Nr,Nz,2:Nm),Pizao(Nr,Nz,2:Nm)
Integer Upwind_NS,Upwind_EW,Upwind_Virtual
Logical Flag_Temp,Flag_Energy,Flag_TFinal
common/cmm1/AMol(5),Tc(5),Pc(5),Zc(5),AMi(5),CpA(5),CpB(5),CpC(5),CpD(5),Y(5)
common/cmm2/AK_A(5),AK_B(5),AK_C(5),AK_D(5),PesoMolecularMistura,Dr_Total
common/tue1/Pi,Sigma
common/tue2/Refraction,Extinction,Absorption,ReflectWall,EmissWall
common/tue3/Relax_Phi,Epsilon_Energy,Epsilon_Temp,MaxIterEnergy,MaxIterTemp,Relax_Temp

common/tue4/Diamond,Diamond_NS,Diamond_EW,Upwind_NS,Upwind_Virtual,Upwind_EW
Niter_Temp=1
Flag_Temp=.true.
do while(Flag_Temp)
! 1. Condições de contorno e termos fontes
IFromNorthBoundary(:)=Refraction**2*Sigma*TWall_N(:)**4/Pi
IFromWestBoundary(:)=Refraction**2*Sigma*TWall_W(:)**4/Pi
IFromEastBoundary(:)=Refraction**2*Sigma*Temperatura(:,Nz)**4/Pi
Source(:,:)=Refraction**2*Sigma*Temperatura(:,:)**4/Pi
! 2. Cálculo das intensidades de radiação
! 2.1. Primeira direção (Mi=-1; Eta=Zeta=0)
! Cálculo dos primeiros RadMenosMeio(i,j)
m=1
do j=1,Nz
do i=1,Nr
Phi_4(i)=Extinction*Volume(i,j)
Teta_4(i)=Extinction*Source(i,j)*Volume(i,j)
end do
INorth(Nr)=IFromNorthBoundary(j)
if (Upwind_NS.eq.1) then !upwind
do i=Nr,1,-1
Phi_1=-Coef_Mi(1)*AreaR_S(i,j)
Teta_1=-Coef_Mi(1)*AreaR_N(i,j)*INorth(i)
RadMenosMeio(i,j)=(Teta_1+Teta_4(i))/(Phi_1+Phi_4(i))
INorth(i-1)=AIIntensity(i,j,1)
end do
else !diamond
do i=Nr,1,-1
Phi_1=-Coef_Mi(1)*AreaR_S(i,j)/(1.-Diamond_NS)
Teta_1=-Coef_Mi(1)*INorth(i)*(AreaR_N(i,j)+Diamond_NS/(1.-Diamond_NS)*Ar
ear_S(i,j))
RadMenosMeio(i,j)=(Teta_1+Teta_4(i))/(Phi_1+Phi_4(i))
INorth(i-1)=(AIIntensity(i,j,1)-Diamond_NS*INorth(i))/(1.-Diamond_NS)
end do
end if
end do !(j)
! 2.2. Demais direções
do m=2,Nm
! 2.2.i. Classificação conforme os cosenos diretores da direção
if (Coef_Mi(m).gt.0.) then
if (Coef_Zeta(m).gt.0.) then
! a) Mz>0, Zeta>0
IWest(:)=IFromWestBoundary(:)
do j=1,Nz
if (Upwind_EW.eq.1) then !upwind
do i=1,Nr
Phi_3(i)=Coef_Zeta(m)*AreaZ(i)
Teta_3(i)=Coef_Zeta(m)*AreaZ(i)*IWest(i)
end do
else !diamond
do i=1,Nr
Phi_3(i)=Coef_Zeta(m)*AreaZ(i)/Diamond_EW
Teta_3(i)=Coef_Zeta(m)*AreaZ(i)*IWest(i)/Diamond_EW
end do
end if
if (Upwind_Virtual.eq.1) then !upwind

```

```

do i=1,Nr
  Phi_2(i)=-Pizao(i,j,m)*AlphaMais(m)
  Teta_2(i)=-Pizao(i,j,m)*AlphaMenos(m)*RadMenosMeio(i,j)
end do
else !diamond
do i=1,Nr
  Phi_2(i)=-Pizao(i,j,m)*AlphaMais(m)/Diamond
  Teta_2(i)=-Pizao(i,j,m)*RadMenosMeio(i,j)*((1.-Diamond)/Diamond*AlphaMais
(m)+AlphaMenos(m))
end do
end if
do i=1,Nr
  Phi_4(i)=Extinction*Volume(i,j)
  Teta_4(i)=Extinction*Source(i,j)*Volume(i,j)
end do
if (Upwind_NS.eq.1) then !upwind
  ISouth(1)=AIntensity(1,j,M_Simetrico(m))
do i=1,Nr
  Phi_1=Coef_Mi(m)*AreaR_N(i,j)
  Teta_1=Coef_Mi(m)*AreaR_S(i,j)*ISouth(i)
  AIntensity(i,j,m)=(Teta_1+Teta_2(i)+Teta_3(i)+Teta_4(i))/ &
    (Phi_1+Phi_2(i)+Phi_3(i)+Phi_4(i))
  ISouth(i+1)=AIntensity(i,j,m)
end do
  IToNorthBoundary(j,m)=ISouth(Nr1p)
  IToSouthBoundary(j,m)=0.
else !diamond
  INorth_Aux=AIntensity(1,j,M_Simetrico(m))+(AIntensity(2,j,M_Simetrico(m))-A
Intensity(1,j,M_Simetrico(m)))*Dr(1)/(Dr(1)+Dr(2))
  ISouth(1)=(AIntensity(1,j,M_Simetrico(m))-Diamond_NS*INorth_Aux)/(1.-Diamon
d_NS)
do i=1,Nr
  Phi_1=Coef_Mi(m)*AreaR_N(i,j)/Diamond_NS
  Teta_1=Coef_Mi(m)*ISouth(i)*((1.-Diamond_NS)/Diamond_NS*AreaR_N(i,j)+Are
aR_S(i,j))
  AIntensity(i,j,m)=(Teta_1+Teta_2(i)+Teta_3(i)+Teta_4(i))/ &
    (Phi_1+Phi_2(i)+Phi_3(i)+Phi_4(i))
  ISouth(i+1)=(AIntensity(i,j,m)-(1.-Diamond_NS)*ISouth(i))/Diamond_NS
end do
  IToNorthBoundary(j,m)=ISouth(Nr1p)
  IToSouthBoundary(j,m)=0.
end if
if (Upwind_EW.eq.1) then !upwind
do i=1,Nr
  IWest(i)=AIntensity(i,j,m)
end do
else !diamond
do i=1,Nr
  IWest(i)=(AIntensity(i,j,m)-(1.-Diamond_EW)*IWest(i))/Diamond_EW
end do
end if
end do !j
else !Zeta<0
! b) M1>0, Zeta<0
  IEast(:)=IFromEastBoundary(:)
do j=Nz,1,-1
  if (Upwind_EW.eq.1) then !upwind
do i=1,Nr
  Phi_3(i)=-Coef_Zeta(m)*AreaZ(i)
  Teta_3(i)=-Coef_Zeta(m)*AreaZ(i)*IEast(i)
end do
else !diamond
do i=1,Nr
  Phi_3(i)=-Coef_Zeta(m)*AreaZ(i)/(1.-Diamond_EW)
  Teta_3(i)=-Coef_Zeta(m)*AreaZ(i)*IEast(i)/(1.-Diamond_EW)
end do
end if
if (Upwind_Virtual.eq.1) then !upwind
do i=1,Nr
  Phi_2(i)=-Pizao(i,j,m)*AlphaMais(m)
  Teta_2(i)=-Pizao(i,j,m)*AlphaMenos(m)*RadMenosMeio(i,j)
end do
else !diamond
do i=1,Nr
  Phi_2(i)=-Pizao(i,j,m)*AlphaMais(m)/Diamond
  Teta_2(i)=-Pizao(i,j,m)*RadMenosMeio(i,j)*((1.-Diamond)/Diamond*AlphaMais
(m)+AlphaMenos(m))
end do

```



```

end if
do i=1,Nr
  Phi_4(i)=Extinction*Volume(i,j)
  Teta_4(i)=Extinction*Source(i,j)*Volume(i,j)
end do
if (Upwind_NS.eq.1) then !upwind
ISouth(1)=AIntensity(1,j,M_Simetrico(m))
do i=1,Nr
  Phi_1=Coef_Mi(m)*AreaR_N(i,j)
  Teta_1=Coef_Mi(m)*AreaR_S(i,j)*ISouth(i)
  AIntensity(i,j,m)=(Teta_1+Teta_2(i)+Teta_3(i)+Teta_4(i))/ &
    (Phi_1+Phi_2(i)+Phi_3(i)+Phi_4(i))
  ISouth(i+1)=AIntensity(i,j,m)
end do
IToNorthBoundary(j,m)=ISouth(Nr1p)
IToSouthBoundary(j,m)=0.
else !diamond
INorth_Aux=AIntensity(1,j,M_Simetrico(m))+(AIntensity(2,j,M_Simetrico(m))-A
Intensity(1,j,M_Simetrico(m)))*Dr(1)/(Dr(1)+Dr(2))
ISouth(1)=(AIntensity(1,j,M_Simetrico(m))-Diamond_NS*INorth_Aux)/(1.-Diamon
d_3)
do i=1,Nr
  Phi_1=Coef_Mi(m)*AreaR_N(i,j)/Diamond_NS
  Teta_1=Coef_Mi(m)*ISouth(i)*((1.-Diamond_NS)/Diamond_NS*AreaR_N(i,j)+Are
aR_S(i,j))
  AIntensity(i,j,m)=(Teta_1+Teta_2(i)+Teta_3(i)+Teta_4(i))/ &
    (Phi_1+Phi_2(i)+Phi_3(i)+Phi_4(i))
  ISouth(i+1)=(AIntensity(i,j,m)-(1.-Diamond_NS)*ISouth(i))/Diamond_NS
end do
IToNorthBoundary(j,m)=ISouth(Nr1p)
IToSouthBoundary(j,m)=0.
end if
if (Upwind_EW.eq.1) then !upwind
do i=1,Nr
  IEast(i)=AIntensity(i,j,m)
end do
else !diamond
do i=1,Nr
  IEast(i)=(AIntensity(i,j,m)-Diamond_EW*IEast(i))/(1.-Diamond_EW)
end do
end if
end do !j
end if !Zeta
else !Mi<0
if (Coef_Zeta(m).gt.0.) then
! c) Mi<0, Zeta>0
IWest(:)=IFromWestBoundary(:)
do j=1,Nz
if (Upwind_EW.eq.1) then !upwind
do i=1,Nr
  Phi_3(i)=Coef_Zeta(m)*AreaZ(i)
  Teta_3(i)=Coef_Zeta(m)*AreaZ(i)*IWest(i)
end do
else !diamond
do i=1,Nr
  Phi_3(i)=Coef_Zeta(m)*AreaZ(i)/Diamond_EW
  Teta_3(i)=Coef_Zeta(m)*AreaZ(i)*IWest(i)/Diamond_EW
end do
end if
if (Upwind_Virtual.eq.1) then !upwind
do i=1,Nr
  Phi_2(1)=-Pizao(i,j,m)*AlphaMais(m)
  Teta_2(i)=-Pizao(i,j,m)*AlphaMenos(m)*RadMenosMeio(1,j)
end do
else !diamond
do i=1,Nr
  Phi_2(i)=-Pizao(i,j,m)*AlphaMais(m)/Diamond
  Teta_2(i)=-Pizao(i,j,m)*RadMenosMeio(i,j)*((1.-Diamond)/Diamond*AlphaMais
(m)+AlphaMenos(m))
end do
end if
do i=1,Nr
  Phi_4(i)=Extinction*Volume(i,j)
  Teta_4(i)=Extinction*Source(1,j)*Volume(1,j)
end do
INorth(Nr)=IFromNorthBoundary(j)
if (Upwind_NS.eq.1) then !upwind
do i=Nr,1,-1

```

```

Phi_1=-Coef_Mi(m)*AreaR_S(i,j)
Teta_1=-Coef_Mi(m)*AreaR_N(i,j)*INorth(i)
AIntensity(i,j,m)=(Teta_1+Teta_2(i)+Teta_3(i)+Teta_4(i))/ &
(Phi_1+Phi_2(i)+Phi_3(i)+Phi_4(i))
INorth(i-1)=AIntensity(i,j,m)
end do
IToNorthBoundary(j,m)=0.
IToSouthBoundary(j,m)=INorth(0)
else !diamond
do i=Nr,1,-1
Phi_1=-Coef_Mi(m)*AreaR_S(i,j)/(1.-Diamond_NS)
Teta_1=-Coef_Mi(m)*INorth(i)*(AreaR_N(i,j)+Diamond_NS/(1.-Diamond_NS)*Ar
eaR_S(i,j))
AIntensity(i,j,m)=(Teta_1+Teta_2(i)+Teta_3(i)+Teta_4(i))/ &
(Phi_1+Phi_2(i)+Phi_3(i)+Phi_4(i))
INorth(i-1)=(AIntensity(i,j,m)-Diamond_NS*INorth(i))/(1.-Diamond_NS)
end do
IToNorthBoundary(j,m)=0.
IToSouthBoundary(j,m)=INorth(0)
end if
if (Upwind_EW.eq..) then !upwind
do i=1,Nr
IWest(i)=AIntensity(i,j,m)
end do
else !diamond
do i=1,Nr
IWest(i)=(AIntensity(i,j,m)-(1.-Diamond_EW)*IWest(i))/Diamond_EW
end do
end if
end do !j
else !Zeta<0
! d) Mi<0, Zeta<0
IEast(:)=IFromEastBoundary(:)
do j=Nz,1,-1
if (Upwind_EW.eq.1) then !upwind
do i=1,Nr
Phi_3(i)=-Coef_Zeta(m)*AreaZ(i)
Teta_3(i)=-Coef_Zeta(m)*AreaZ(i)*IEast(i)
end do
else !diamond
do i=1,Nr
Phi_3(i)=-Coef_Zeta(m)*AreaZ(i)/(1.-Diamond_EW)
Teta_3(i)=-Coef_Zeta(m)*AreaZ(i)*IEast(i)/(1.-Diamond_EW)
end do
end if
if (Upwind_Virtual.eq.1) then !upwind
do i=1,Nr
Phi_2(i)=-Pizao(i,j,m)*AlphaMais(m)
Teta_2(i)=-Pizao(i,j,m)*AlphaMenos(m)*RadMenosMeio(i,j)
end do
else !diamond
do i=1,Nr
Phi_2(i)=-Pizao(i,j,m)*AlphaMais(m)/Diamond
Teta_2(i)=-Pizao(i,j,m)*RadMenosMeio(i,j)*((1.-Diamond)/Diamond*AlphaMais
(m)+AlphaMenos(m))
end do
end if
do i=1,Nr
Phi_4(i)=Extinction*Volume(i,j)
Teta_4(i)=Extinction*Source(i,j)*Volume(i,j)
end do
INorth(Nr)=IFromNorthBoundary(j)
if (Upwind_NS.eq.1) then !upwind
do i=Nr,1,-1
Phi_1=-Coef_Mi(m)*AreaR_S(i,j)
Teta_1=-Coef_Mi(m)*AreaR_N(i,j)*INorth(i)
AIntensity(i,j,m)=(Teta_1+Teta_2(i)+Teta_3(i)+Teta_4(i))/ &
(Phi_1+Phi_2(i)+Phi_3(i)+Phi_4(i))
INorth(i-1)=AIntensity(i,j,m)
end do
IToNorthBoundary(j,m)=0.
IToSouthBoundary(j,m)=INorth(0)
else !diamond
do i=Nr,1,-1
Phi_1=-Coef_Mi(m)*AreaR_S(i,j)/(1.-Diamond_NS)
Teta_1=-Coef_Mi(m)*INorth(i)*(AreaR_N(i,j)+Diamond_NS/(1.-Diamond_NS)*Ar
eaR_S(i,j))
AIntensity(i,j,m)=(Teta_1+Teta_2(i)+Teta_3(i)+Teta_4(i))/ &

```

```

      (Phi_1+Phi_2(i)+Phi_3(i)+Phi_4(i))
      INorth(i-1)=(AIntensity(i,j,m)-Diamond_NS*INorth(i))/(1.-Diamond_NS)
    end do
    IToNorthBoundary(j,m)=0.
    IToSouthBoundary(j,m)=INorth(0)
  end if
  if (Upwind_EW.eq.1) then !upwind
    do i=1,Nr
      IEast(i)=AIntensity(i,j,m)
    end do
  else !diamond
    do i=1,Nr
      IEast(i)=(AIntensity(i,j,m)-Diamond_EW*IEast(i))/(1.-Diamond_EW)
    end do
  end if
end do !j
end if !Zeta
end if !Mi
! 2.2.2. Cálculo de RadMenosMeio(i,j) para a próxima direção
if (Upwind_Virtual.eq.1) then !upwind
  do j=1,Nz
    do i=1,Nr
      RadMenosMeio(i,j)=AIntensity(i,j,m)
    end do
  end do
else
  do j=1,Nz
    do i=1,Nr
      RadMenosMeio(i,j)=(AIntensity(i,j,m)-(1.-Diamond)*RadMenosMeio(i,j))/Diamond
    end do
  end do
end if
end do ! fim do ciclo das direções "m"
! 3. Cálculo dos fluxos de radiação para cada volume elementar
do j=1,Nz
  do i=1,Nr
    G=0
    do m=2,Nm
      G=G+W(m)*AIntensity(i,j,m)
    end do
    DivQr(i,j)=Absorption*(4.*Sigma*Temperatura(i,j)**4-G)
  end do
end do
! 4. Cálculo de QToNorthBoundary(j) e QToSouthBoundary(j)
do j=1,Nz
  QToNorthBoundary(j)=0.
  QToSouthBoundary(j)=0.
  do m=2,Nm
    QToNorthBoundary(j)=QToNorthBoundary(j)+dabs(W(m)*Coef_Mi(m)*IToNorthBoundary(
j,m))
    QToSouthBoundary(j)=QToSouthBoundary(j)+dabs(W(m)*Coef_Mi(m)*IToSouthBoundary(
j,m))
  end do
end do
Niter_Energy=1
Flag_Energy=.true.
Phi(:,:)=Temperatura(:,:)
do while (Flag_Energy)
! *** North Boundary
  i=Nr
  do j=1,Nz
! ** face Norte
    An=0.
    Phi_N=0.
! ** face Sul
    Ro_Interf=Prop_Interp(Ro(Nr-1,j),Dr(Nr-1),Ro(Nr,j),Dr(Nr))
    Gama_Interf=Prop_Interp(Gama(Nr-1,j),Dr(Nr-1),Gama(Nr,j),Dr(Nr))
    Convection=Ro_Interf*V(Nr,j)*AreaR_S(Nr,j)
    Diffusion=Gama_Interf*AreaR_S(Nr,j)/Dr_S(Nr)
    Peclet=Ro_Interf*V(Nr,j)*Dr_S(Nr)/Gama_Interf
    As=Convection*(0.5+Alfa(Peclet))+Diffusion*Beta(Peclet)
    Phi_S=Phi(Nr-1,j)
! ** face Este
    if (j.eq.Nz) then
      Ro_Interf=Ro(i,j)
      Gama_Interf=Gama(i,j)
      DeltaZ_E=Dz(j)
      Convection=Ro_Interf*U(i,j+1)*AreaZ(i)

```

```

Diffusion=Gama_Interf*AreaZ(i)/DeltaZ_E
Peclet=Ro_Interf*U(i,j+1)*DeltaZ_E/Gama_Interf
Ae=-Convection*(0.5-Alfa(Peclet))+Diffusion*Beta(Peclet)
Phi_E=Phi(i,j) !Derivada zero
else
  Ro_Interf=Prop_Interp(Ro(Nr,j),Dz(j),Ro(Nr,j+1),Dz(j+1))
  Gama_Interf=Prop_Interp(Gama(Nr,j),Dz(j),Gama(Nr,j+1),Dz(j+1))
  Convection=Ro_Interf*U(Nr,j+1)*AreaZ(Nr)
  Diffusion=Gama_Interf*AreaZ(Nr)/Dz_E(j)
  Peclet=Ro_Interf*U(Nr,j+1)*DeltaZ_E(j)/Gama_Interf
  Ae=-Convection*(0.5-Alfa(Peclet))+Diffusion*Beta(Peclet)
  Phi_E=Phi(Nr,j+1)
end if
! ** face Oeste
if (j.eq.1) then
  Ro_Interf=Ro_Entrada(i)
  Gama_Interf=Gama_Entrada(i)
  DeltaZ_W=Dz(1)
  Convection=Ro_Interf*U(i,j)*AreaZ(i)
  Diffusion=Gama_Interf*AreaZ(i)/DeltaZ_W
  Peclet=Ro_Interf*U(i,j)*DeltaZ_W/Gama_Interf
  Aw=Convection*(0.5+Alfa(Peclet))+Diffusion*Beta(Peclet)
  Phi_W=2.*TWall_W(i)-Phi(i,j) !Valor estabelecido na interface
else
  Ro_Interf=Prop_Interp(Ro(Nr,j-1),Dz(j-1),Ro(Nr,j),Dz(j))
  Gama_Interf=Prop_Interp(Gama(Nr,j-1),Dz(j-1),Gama(Nr,j),Dz(j))
  Convection=Ro_Interf*U(Nr,j+1)*AreaZ(Nr)
  Diffusion=Gama_Interf*AreaZ(Nr)/Dz_W(j)
  Peclet=Ro(Nr,j)*U(Nr,j+1)*DeltaZ_W(j)/Gama_Interf
  Aw=Convection*(0.5+Alfa(Peclet))+Diffusion*Beta(Peclet)
  Phi_W=Phi(Nr,j-1)
end if
! ** termo fonte
Sp(Nr,j)=0.
Sc(Nr,j)=(-DivQr(Nr,j)+CoefHeatTransfer(j)*AreaR_N(Nr,j)*(TWall_N(j)-Temp_Media(j)))/Volume(Nr,j)/Cp(Nr,j)
! ** novo valor de Phi
Aux1=An*Phi_N+As*Phi_S+Ae*Phi_E+Aw*Phi_W+Sc(Nr,j)*Volume(Nr,j)
Aux2=(An+As+Ae+Aw-Sp(Nr,j))*Volume(Nr,j)
NewPhi(Nr,j)=Relax_Phi*Aux1/Aux2+(1.-Relax_Phi)*Phi(Nr,j)
end do
! *** South Boundary
i=1
do j=1,Nz
! ** face Norte
  Ro_Interf=Prop_Interp(Ro(1,j),Dr(1),Ro(1+1,j),Dr(1+1))
  Gama_Interf=Prop_Interp(Gama(1,j),Dr(1),Gama(1+1,j),Dr(1+1))
  Convection=Ro_Interf*V(1+1,j)*AreaR_N(1,j)
  Diffusion=Gama_Interf*AreaR_N(1,j)/Dr_N(1)
  Peclet=Ro_Interf*V(1+1,Nz)*Dr_N(i)/Gama_Interf
  An=-Convection*(0.5-Alfa(Peclet))+Diffusion*Beta(Peclet)
  Phi_N=Phi(1+1,j)
! ** face Sul
  Ro_Interf=Ro(i,j)
  Gama_Interf=Gama(i,j)
  DeltaR_S=Dr(1)
  Diffusion=Gama_Interf*AreaR_S(i,j)/DeltaR_S
  As=Diffusion
  Phi_S=Phi(i,j) !Derivada zero
! ** face Este
  if (j.eq.Nz) then
    Ro_Interf=Ro(i,j)
    Gama_Interf=Gama(i,j)
    DeltaZ_E=Dz(j)
    Convection=Ro_Interf*U(i,j+1)*AreaZ(i)
    Diffusion=Gama_Interf*AreaZ(i)/DeltaZ_E
    Peclet=Ro_Interf*U(i,j+1)*DeltaZ_E/Gama_Interf
    Ae=-Convection*(0.5-Alfa(Peclet))+Diffusion*Beta(Peclet)
    Phi_E=Phi(i,j) !Derivada zero
  else
    Ro_Interf=Prop_Interp(Ro(1,j),Dz(j),Ro(1,j+1),Dz(j+1))
    Gama_Interf=Prop_Interp(Gama(1,j),Dz(j),Gama(1,j+1),Dz(j+1))
    Convection=Ro_Interf*U(1,j+1)*AreaZ(1)
    Diffusion=Gama_Interf*AreaZ(1)/Dz_E(j)
    Peclet=Ro_Interf*U(1,j+1)*DeltaZ_E(j)/Gama_Interf
    Ae=-Convection*(0.5-Alfa(Peclet))
    Ae=Ae+Diffusion*Beta(Peclet)
    Phi_E=Phi(1,j+1)
  end if
end do

```

```

end if
! ** face Oeste
if (j.eq.1) then
  Ro_Interf=Ro_Entrada(i)
  Gama_Interf=Gama_Entrada(i)
  DeltaZ_W=Dz(1)
  Convection=Ro_Interf*U(i,j)*AreaZ(i)
  Diffusion=Gama_Interf*AreaZ(i)/DeltaZ_W
  Peclet=Ro_Interf*U(i,j)*DeltaZ_W/Gama_Interf
  Aw=Convection*(0.5+Alfa(Peclet))+Diffusion*Beta(Peclet)
  Phi_W=2.*TWall_W(i)-Phi(i,j) !Valor estabelecido na interface
else
  Ro_Interf=Prop_Interp(Ro(1,j-1),Dz(j-1),Ro(1,j),Dz(j))
  Gama_Interf=Prop_Interp(Gama(1,j-1),Dz(j-1),Gama(1,j),Dz(j))
  Convection=Ro_Interf*U(1,j)*AreaZ(1)
  Diffusion=Gama_Interf*AreaZ(1)/Dz_W(j)
  Peclet=Ro_Interf*U(1,j)*Dz_W(j)/Gama(1,j)
  Aw=Convection*(0.5+Alfa(Peclet))+Diffusion*Beta(Peclet)
  Phi_W=Phi(1,j-1)
end if
! ** termo fonte
Sp(1,j)=0.
Sc(1,j)=-DivQr(1,j)/Cp(1,j)
! ** novo valor de Phi
Aux1=An*Phi_N+As*Phi_S+Ae*Phi_E+Aw*Phi_W+Sc(1,j)*Volume(1,j)
Aux2=(An+As+Ae+Aw-Sp(1,j)*Volume(1,j))
NewPhi(1,j)=Relax_Phi*Aux1/Aux2+(1.-Relax_Phi)*Phi(1,j)
end do
! *** East Boundary

j=Nz
do i=2,Nr-1
! ** face Norte
  Ro_Interf=Prop_Interp(Ro(i,Nz),Dr(i),Ro(i+1,Nz),Dr(i+1))
  Gama_Interf=Prop_Interp(Gama(i,Nz),Dr(i),Gama(i+1,Nz),Dr(i+1))
  Convection=Ro_Interf*V(i+1,Nz)*AreaR_N(i,Nz)
  Diffusion=Gama_Interf*AreaR_N(i,Nz)/Dr_N(i)
  Peclet=Ro_Interf*V(i+1,Nz)*Dr_N(i)/Gama_Interf
  An=-Convection*(0.5-Alfa(Peclet))+Diffusion*Beta(Peclet)
  Phi_N=Phi(i+1,Nz)
! ** face Sul
  Ro_Interf=Prop_Interp(Ro(i-1,Nz),Dr(i-1),Ro(i,Nz),Dr(i))
  Gama_Interf=Prop_Interp(Gama(i-1,Nz),Dr(i-1),Gama(i,Nz),Dr(i))
  Convection=Ro_Interf*V(i,Nz)*AreaR_S(i,Nz)
  Diffusion=Gama_Interf*AreaR_S(i,Nz)/Dr_S(i)
  Peclet=Ro_Interf*V(i,Nz)*Dr_S(i)/Gama_Interf
  As=Convection*(0.5+Alfa(Peclet))+Diffusion*Beta(Peclet)
  Phi_S=Phi(i-1,Nz)
! ** face Este
  Ro_Interf=Ro(i,j)
  Gama_Interf=Gama(i,j)
  DeltaZ_E=Dz(j)
  Convection=Ro_Interf*U(i,j+1)*AreaZ(i)
  Diffusion=Gama_Interf*AreaZ(i)/DeltaZ_E
  Peclet=Ro_Interf*U(i,j+1)*DeltaZ_E/Gama_Interf
  Ae=-Convection*(0.5-Alfa(Peclet))+Diffusion*Beta(Peclet)
  Phi_E=Phi(i,j) !Derivada zero
! ** face Oeste
  Ro_Interf=Prop_Interp(Ro(i,Nz-1),Dz(Nz-1),Ro(i,Nz),Dz(Nz))
  Gama_Interf=Prop_Interp(Gama(i,Nz-1),Dz(Nz-1),Gama(i,Nz),Dz(Nz))
  Convection=Ro_Interf*U(i,Nz)*AreaZ(i)
  Diffusion=Gama_Interf*AreaZ(i)/Dz_W(Nz)
  Peclet=Ro_Interf*U(i,Nz)*Dz_W(Nz)/Gama_Interf
  Aw=Convection*(0.5+Alfa(Peclet))+Diffusion*Beta(Peclet)
  Phi_W=Phi(1,Nz-1)
! ** termo fonte
Sp(i,Nz)=0.
Sc(i,Nz)=-DivQr(1,Nz)/Cp(1,Nz)
! ** novo valor de Phi
Aux1=An*Phi_N+As*Phi_S+Ae*Phi_E+Aw*Phi_W+Sc(1,Nz)*Volume(i,Nz)
Aux2=(An+As+Ae+Aw-Sp(1,Nz)*Volume(1,Nz))
NewPhi(1,Nz)=Relax_Phi*Aux1/Aux2+(1.-Relax_Phi)*Phi(i,Nz)
end do
! *** West Boundary
j=1
do i=2,Nr-1
! ** face Norte
  Ro_Interf=Prop_Interp(Ro(i,1),Dr(i),Ro(i+1,1),Dr(i+1))

```

```

Gama_Interf=Prop_Interp(Gama(i,1),Dr(i),Gama(i+1,1),Dr(i+1))
Convection=Ro_Interf*V(I-1,1)*AreaR_N(i,1)
Diffusion=Gama_Interf*AreaR_N(i,1)/Dr_N(i)
Peclet=Ro_Interf*V(i+1,1)*Dr_N(i)/Gama_Interf
An=-Convection*(0.5-Alfa(Peclet))+Diffusion*Beta(Peclet)
Phi_N=Phi(i+1,1)
! ** face Sul
Ro_Interf=Prop_Interp(Ro(i-1,1),Dr(i-1),Ro(i,1),Dr(i))
Gama_Interf=Prop_Interp(Gama(i-1,1),Dr(i-1),Gama(i,1),Dr(i))
Convection=Ro_Interf*V(i,1)*AreaR_S(i,1)
Diffusion=Gama_Interf*AreaR_S(i,1)/Dr_S(i)
Peclet=Ro_Interf*V(i,1)*Dr_S(i)/Gama_Interf
As=Convection*(0.5+Alfa(Peclet))+Diffusion*Beta(Peclet)
Phi_S=Phi(i-1,1)
! ** face Este
Ro_Interf=Prop_Interp(Ro(i,1),Dz(1),Ro(i,1+1),Dz(1+1))
Gama_Interf=Prop_Interp(Gama(i,1),Dz(1),Gama(i,1+1),Dz(1+1))
Convection=Ro_Interf*U(i,1+1)*AreaZ(i)
Diffusion=Gama_Interf*AreaZ(i)/Dz_E(1)
Peclet=Ro_Interf*U(i,1+1)*Dz_E(1)/Gama_Interf
Ae=-Convection*(0.5-Alfa(Peclet))+Diffusion*Beta(Peclet)
Phi_E=Phi(i,1+1)
! ** face Oeste
Ro_Interf=Ro_Entrada(i)
Gama_Interf=Gama_Entrada(i)
DeltaZ_W=Dz(1)
Convection=Ro_Interf*U(i,j)*AreaZ(i)
Diffusion=Gama_Interf*AreaZ(i)/DeltaZ_W
Peclet=Ro_Interf*U(i,j)*DeltaZ_W/Gama_Interf
Aw=Convection*(0.5+Alfa(Peclet))+Diffusion*Beta(Peclet)
Phi_W=2.*TWall_W(i)-Phi(i,j) !Valor estabelecido na interface
! ** termo fonte
Sp(i,1)=0.
Sc(i,1)=-DivQr(i,1)/Cp(i,1)
! ** novo valor de Phi
Aux1=An*Phi_N+As*Phi_S+Ae*Phi_E+Aw*Phi_W+Sc(i,1)*Volume(i,1)
Aux2=(An+As+Ae+Aw-Sp(i,1))*Volume(i,1)
NewPhi(i,1)=Relax_Phi*Aux1/Aux2+(1.-Relax_Phi)*Phi(i,1)
end do
! *** Internal Points
do j=2,Nz-1
do i=2,Nr-1
! ** face Norte
Ro_Interf=Prop_Interp(Ro(i,j),Dr(i),Ro(i+1,j),Dr(i+1))
Gama_Interf=Prop_Interp(Gama(i,j),Dr(i),Gama(i+1,j),Dr(i+1))
Convection=Ro_Interf*V(i+1,j)*AreaR_N(i,j)
Diffusion=Gama_Interf*AreaR_N(i,j)/Dr_N(i)
Peclet=Ro_Interf*V(i+1,j)*Dr_N(i)/Gama_Interf
An=-Convection*(0.5-Alfa(Peclet))+Diffusion*Beta(Peclet)
Phi_N=Phi(i+1,j)
! ** face Sul
Ro_Interf=Prop_Interp(Ro(i-1,j),Dr(i-1),Ro(i,j),Dr(i))
Gama_Interf=Prop_Interp(Gama(i-1,j),Dr(i-1),Gama(i,j),Dr(i))
Convection=Ro_Interf*V(i,j)*AreaR_S(i,j)
Diffusion=Gama_Interf*AreaR_S(i,j)/Dr_S(i)
Peclet=Ro_Interf*V(i,j)*Dr_S(i)/Gama_Interf
As=Convection*(0.5+Alfa(Peclet))+Diffusion*Beta(Peclet)
Phi_S=Phi(i-1,j)
! ** face Este
Ro_Interf=Prop_Interp(Ro(i,j),Dz(j),Ro(i,j+1),Dz(j+1))
Gama_Interf=Prop_Interp(Gama(i,j),Dz(j),Gama(i,j+1),Dz(j+1))
Convection=Ro_Interf*U(i,j+1)*AreaZ(i)
Diffusion=Gama_Interf*AreaZ(i)/Dz_E(j)
Peclet=Ro_Interf*U(i,j+1)*Dz_E(j)/Gama_Interf
Ae=-Convection*(0.5-Alfa(Peclet))+Diffusion*Beta(Peclet)
Phi_E=Phi(i,j+1)
! ** face Oeste
Ro_Interf=Prop_Interp(Ro(i,j-1),Dz(j-1),Ro(i,j),Dz(j))
Gama_Interf=Prop_Interp(Gama(i,j-1),Dz(j-1),Gama(i,j),Dz(j))
Convection=Ro_Interf*U(i,j)*AreaZ(i)
Diffusion=Gama_Interf*AreaZ(i)/Dz_W(j)
Peclet=Ro_Interf*U(i,j)*Dz_W(j)/Gama_Interf
Aw=Convection*(0.5+Alfa(Peclet))+Diffusion*Beta(Peclet)
Phi_W=Phi(i,j-1)
! ** termo fonte
Sp(i,j)=0.
Sc(i,j)=-DivQr(i,j)/Cp(i,j)
! ** novo valor de Phi

```

```

Aux1=An*Phi_N+As*Phi_S+Ae*Phi_E+Aw*Phi_W+Sc(i,j)*Volume(i,j)
Aux2=(An+As+Ae+Aw-Sp(i,j)*Volume(i,j))
NewPhi(i,j)=Relax_Phi*Aux1/Aux2+(1.-Relax_Phi)*Phi(i,j)
end do
end do
! *** Verifica convergência da temperatura
if(Niter_Energy.lt.MaxIterEnergy) then
  Niter_Energy=Niter_Energy+1
  Flag_Energy=.false.
  do j=1,nz
    do i=1,nr
      if(dabs((NewPhi(i,j)-Phi(i,j))/Phi(i,j)) &
        .gt.Epsilon_Energy) Flag_Energy=.true.
    end do
  end do
  if(.not.Flag_Energy) write(*,1000) Niter_Energy !convergiu
else
  Flag_Energy=.false.
  write(*,1010) !no. maximo de iteracoes
end if
Phi(:,,:)=NewPhi(:,,:)
TWall_E(:)=NewPhi(:,Nz)
end do !(do while)
! Verifica convergência final
if(Niter_Temp.lt.MaxIterTemp) then
  Niter_Temp=Niter_Temp+1
  Flag_Temp=.false.
  do j=1,Nz
    do i=1,Nr
      if(dabs((NewPhi(i,j)-Temperatura(i,j))/ &
        Temperatura(i,j)).gt.Epsilon_Temp) Flag_Temp=.true.
    end do
  end do
  if(.not.Flag_Temp) then
    write(*,1020) Niter_Temp !convergiu
    Flag_TFinal=.false.
  end if
else
  Flag_TFinal=.true.
  Flag_Temp=.false.
  write(*,1030) !numero maximo de iteracoes
end if
Temperatura(:,:)=(1.-Relax_Temp)*Temperatura(:,:)+Relax_Temp*NewPhi(:,,:)
do i=1,Nr
  do j=1,Nz
    Temp=Temperatura(i,j)
    call propriedades(Temp,ViscMistura,CondTermicaMistura,CpMistura,Densidade)
    Viscosidade(i,j)=ViscMistura
    Ro(i,j)=Densidade
    Cp(i,j)=CpMistura
  end do
end do
do i=1,Nr
  do j=1,Nz
    Gama(i,j)=(Viscosidade(i,j)+0.09*Ro(i,j)*TurbK(i,j)**2/Epsilon(i,j))/0.7
  end do
end do
! *** Temperatura media
do j=1,Nz
  Soma_Numerador=0.
  Soma_Denominador=0.
  do i=1,Nr
    Soma_Numerador=Soma_Numerador+Ro(i,j)*Volume(i,j)*Cp(i,j)*Temperatura(i,j)
    Soma_Denominador=Soma_Denominador+Ro(i,j)*Volume(i,j)*Cp(i,j)
  end do
  Temp_Media(j)=Soma_Numerador/Soma_Denominador
end do
! *** Velocidade media
do j=1,Nz
  Soma_Numerador=0.
  Soma_Denominador=0.
  do i=1,Nr
    Soma_Numerador=Soma_Numerador+Ro(i,j)*Volume(i,j)*U(i,j)
    Soma_Denominador=Soma_Denominador+Ro(i,j)*Volume(i,j)
  end do
  U_Medio(j)=Soma_Numerador/Soma_Denominador
end do
do j=1,Nz

```



```

call propriedades (Temp_Media (j), ViscMistura, CondTermicaMistura, CpMistura, Densidad
e)
Reynolds=Densidade*U_Medio(j)*(2.*Dr_Total)/ViscMistura
Prandtl=CpMistura*ViscMistura/CondTermicaMistura
CoefHeatTransfer(j)=0.023*CondTermicaMistura/(2.*Dr_Total)*Reynolds**0.8*Prandtl*
*0.3
end do
end do !(while)
! *** Temperatura externa do tubo
do j=1,Nz
  Q_Pelricula=CoefHeatTransfer(j)*(TWall_N(j)-Temp_Media(j))
  Q_Emissao=Sigma*TWall_N(j)**4
  FluxoCalorCond(j)=QToNorthBoundary(j)-Q_Pelricula-Q_Emissao
  Temp_Externa(j)=TWall_N(j)-FluxoCalorCond(j)*0.004735.
end do
1000 format('          Phi:',i3,' iteracoes')
1010 format('          Phi: numero maximo de iteracoes')
1020 format('          Temperatura:',i3,' iteracoes')
1030 format('          Temperatura: numero maximo de iteracoes')
end
!
SUBROUTINE PROPRIEADADES (Temp,ViscMistura,CondTermicaMistura,CpMistura,Densidade)
! *** Subrotina que atualiza propriedades função da temperatura
!
! 1...CO2
! 2...H2O
! 3...N2
! 4...O2
! 5...SO2
!
dimension Tr(5),Eps(5),FpZero(5),ANi(5),Phi(5,5),CondTermica(5),CalorEspecifico(5)
common/cmm1/AMol(5),Tc(5),Pc(5),Zc(5),AMi(5),CpA(5),CpB(5),CpC(5),CpD(5),Y(5)
common/cmm2/AK_A(5),AK_B(5),AK_C(5),AK_D(5),PesoMolecularMistura,Dr_Total
! Viscosidade
Tr(:)=Temp/Tc(:)
Eps(:)=0.176*(Tc(:)/(AMol(:)**3*Pc(:)**4))**(1./6.)
FpZero(1)=1.
FpZero(2)=1.+30.55*(0.292-Zc(2))**1.72*dabs(0.96+0.1*(Tr(2)-0.7))
FpZero(3)=1.
FpZero(4)=1.
FpZero(5)=1.+30.55*(0.292-Zc(5))**1.72
ANi(:)=(0.807*Tr(:)**0.618-0.357*dexp(-0.449*Tr(:))+0.34*dexp(-4.058*Tr(:)+0.018))/E
ps(:)
do i=1,5
  do j=1,5
    if(j.eq.i) then
      Phi(i,j)=1.
    else
      Phi(i,j)=(1.+(ANi(i)/ANi(j))**0.5*(AMol(j)/AMol(i))**0.25)**2/(8.*(1.+AMol(
i)/AMol(j)))**0.5
    end if
  end do
end do
ViscMistura=0.
do i=1,5
  Aux=0.
  do j=1,5
    Aux=Aux+Y(j)*Phi(i,j)
  end do
  ViscMistura=ViscMistura+Y(i)*ANi(i)/Aux
end do
ViscMistura=ViscMistura*1.e-7 ![1.e-6*poise] para [Ns/m2]
! Condutividade térmica [W/(m K)]
CondTermica(:)=AK_A(:)+AK_B(:)*Temp+AK_C(:)*Temp**2+AK_D(:)*Temp**3
do i=1,5
  do j=1,5
    if(j.eq.i) then
      Phi(i,j)=1.
    else
      Phi(i,j)=(1.+(CondTermica(i)/CondTermica(j))**0.5*(AMol(j)/AMol(i))**0.25)*
*2/(8.*(1.+AMol(i)/AMol(j)))**0.5
    end if
  end do
end do
end do
CondTermicaMistura=0.
do i=1,5
  Aux=0.
  do j=1,5

```



```

        Aux=Aux+Y(j)*Phi(i,j)
    end do
    CondTermicaMistura=CondTermicaMistura+Y(i)*CondTermica(i)/Aux
end do
! Calor especifico a pressão constante
CalorEspecifico(:)=CpA(:)+CpB(:)*Temp+CpC(:)*Temp**2+CpD(:)*Temp**3
CpMistura=0.
do i=1,5
    CpMistura=CpMistura+Y(i)*CalorEspecifico(i)
end do
CpMistura=CpMistura*1000./PesoMolecularMistura ![kJ/(kmol K)] para [J/(kg K)]
! Densidade
Densidade=PesoMolecularMistura/(0.082057*Temp) ![kg/m3]
end
!
!-----

```

```

FUNCTION Alfa(Peclet)
    Aux=Peclet*Peclet
    Alfa=Aux/(10.+2.*Aux)
    if(Peclet.lt.0.) Alfa=-1.*Alfa
    return
end
!
!-----

```

```

FUNCTION Beta(Peclet)
    Aux=Peclet*Peclet
    Beta=(1.+0.005*Aux)/(1.+0.05*Aux)
    return
end
!
!-----

```

```

FUNCTION PROP_INTERP(Prop_Menos,Dist_Menos,Prop_Mais,Dist_Mais)
    F_Menos=Dist_Menos/(Dist_Menos+Dist_Mais)
    Prop_Interp=1./(F_Menos/Prop_Menos+(1.-F_Menos)/Prop_Mais)
end
!
!-----

```

```

*****
*** Arquivo 2: Arquivo "tese03_geral_in.dat" ***
***      requerido por "tese03_c.f90"      ***
*****
*** rlxk,rlxeps,rlxuv,epsk,epseps ***
1.,1.,4.,0.10,0.10
*** niter,eps,epsu,epsv ***
100,0.1e+1,5.e-3,5.e-3
*** niter2,npress,dnpress,ipressi,ipressj ***
0,1,20,1,1
*** Dz(20) ***
.178E-01, .214E-01, .256E-01, .308E-01, .369E-01, .443E-01, .532E-01, .638E-01, .766E-
01, .919E-01,
.766E-01, .638E-01, .531E-01, .442E-01, .369E-01, .307E-01, .256E-01, .213E-01, .177E-
01, .177E-01,
*** Dz(50) ***
.270E-01, .324E-01, .388E-01, .466E-01, .559E-01, .671E-01, .805E-01, .966E-01, .116E+
00, .139E+00,
.139E+00, .139E+00, .139E+00, .139E+00, .139E+00, .139E+00, .139E+00, .139E+00, .139E+
00, .139E+00,
.139E+00, .139E+00, .139E+00, .139E+00, .139E+00, .139E+00, .139E+00, .139E+00, .139E+
00, .139E+00,
.116E+00, .965E-01, .804E-01, .670E-01, .558E-01, .465E-01, .387E-01, .322E-01, .269E-
01, .269E-01,
*****
*** Arquivo 3: Arquivo "campos_iniciais_first.dat" ***
*** (pode substituir "campos_iniciais_in.dat) ***
***      requerido por "tese03_c.f90"      ***
*****
*** Temperatura
.122E+04, .122E+04, .122E+04, .121E+04, .120E+04, .120E+04, .119E+04, .118E+04, .117E+
04, .116E+04,
.115E+04, .114E+04, .113E+04, .112E+04, .111E+04, .110E+04, .109E+04, .107E+04, .105E+
04, .104E+04,
.990E+03, .989E+03, .991E+03, .993E+03, .994E+03, .994E+03, .994E+03, .994E+03, .994E+
03, .994E+03,
.994E+03, .994E+03, .994E+03, .994E+03, .994E+03, .994E+03, .994E+03, .994E+03, .994E+
03, .994E+03,
.991E+03, .988E+03, .986E+03, .984E+03, .983E+03, .983E+03, .984E+03, .986E+03, .991E+

```



```

.151E+02, .151E+02, .151E+02, .152E+02, .152E+02, .152E+02, .152E+02, .151E+02, .151E+
02, .151E+02,
.150E+02, .150E+02, .149E+02, .148E+02, .148E+02, .147E+02, .146E+02, .146E+02, .145E+
02, .145E+02,
.144E+02, .143E+02, .143E+02, .142E+02, .141E+02, .141E+02, .140E+02, .140E+02, .139E+
02, .138E+02,
.138E+02, .137E+02, .137E+02, .136E+02, .136E+02, .135E+02, .135E+02, .134E+02, .134E+
02, .134E+02,
.133E+02, .133E+02, .133E+02, .133E+02, .133E+02, .132E+02, .132E+02, .132E+02, .132E+
02, .132E+02,
.132E+02,
.151E+02, .151E+02, .150E+02, .149E+02, .148E+02, .147E+02, .146E+02, .144E+02, .143E+
02, .141E+02,
.139E+02, .137E+02, .136E+02, .134E+02, .133E+02, .131E+02, .130E+02, .129E+02, .128E+
02, .127E+02,
.126E+02, .125E+02, .124E+02, .123E+02, .122E+02, .121E+02, .121E+02, .120E+02, .119E+
02, .119E+02,
.118E+02, .118E+02, .117E+02, .117E+02, .116E+02, .116E+02, .115E+02, .115E+02, .115E+
02, .114E+02,
.114E+02, .114E+02, .113E+02, .113E+02, .113E+02, .113E+02, .113E+02, .113E+02, .113E+
02, .113E+02,
.113E+02,
*** Velocidade V
.000E+00, .000E+00, .000E+00, .000E+00, .000E+00, .000E+00, .000E+00, .000E+00, .000E+
00, .000E+00,
.000E+00, .000E+00, .000E+00, .000E+00, .000E+00, .000E+00, .000E+00, .000E+00, .000E+
00, .000E+00,
.000E+00, .000E+00, .000E+00, .000E+00, .000E+00, .000E+00, .000E+00, .000E+00, .000E+
00, .000E+00,
.000E+00, .000E+00, .000E+00, .000E+00, .000E+00, .000E+00, .000E+00, .000E+00, .000E+
00, .000E+00,
.000E+00, .000E+00, .000E+00, .000E+00, .000E+00, .000E+00, .000E+00, .000E+00, .000E+
00, .000E+00,
.165E-01, .158E-01, .102E-01, .622E-02, .380E-02, .229E-02, .128E-02, .602E-03, .149E-
03, -.148E-03,
-.203E-03, -.241E-03, -.268E-03, -.285E-03, -.294E-03, -.299E-03, -.302E-03, -.308E-03, -.317E-
03, -.332E-03,
-.351E-03, -.373E-03, -.395E-03, -.413E-03, -.426E-03, -.431E-03, -.429E-03, -.421E-03, -.410E-
03, -.398E-03,
-.387E-03, -.379E-03, -.374E-03, -.372E-03, -.370E-03, -.368E-03, -.362E-03, -.350E-03, -.331E-
03, -.318E-03,
-.366E-03, -.540E-03, -.900E-03, -.151E-02, -.240E-02, -.360E-02, -.506E-02, -.664E-02, -.549E-
02, .762E-01,
.346E-02, .513E-02, .397E-02, .245E-02, .128E-02, .442E-03, -.183E-03, -.653E-03, -.100E-
02, -.126E-02,
-.137E-02, -.143E-02, -.146E-02, -.147E-02, -.146E-02, -.143E-02, -.140E-02, -.137E-02, -.134E-
02, -.131E-02,
-.130E-02, -.130E-02, -.129E-02, -.129E-02, -.128E-02, -.126E-02, -.124E-02, -.120E-02, -.116E-
02, -.111E-02,
-.107E-02, -.104E-02, -.101E-02, -.990E-03, -.967E-03, -.936E-03, -.884E-03, -.797E-03, -.650E-
03, -.416E-03,
-.949E-04, .277E-03, .682E-03, .111E-02, .159E-02, .224E-02, .341E-02, .621E-02, .140E-
01, .393E-01,
.767E-03, .159E-02, .134E-02, .627E-03, -.113E-03, -.755E-03, -.129E-02, -.174E-02, -.210E-
02, -.238E-02,
-.253E-02, -.260E-02, -.264E-02, -.264E-02, -.262E-02, -.257E-02, -.251E-02, -.244E-02, -.237E-
02, -.232E-02,
-.228E-02, -.225E-02, -.223E-02, -.222E-02, -.220E-02, -.218E-02, -.214E-02, -.209E-02, -.203E-
02, -.196E-02,
-.190E-02, -.183E-02, -.178E-02, -.173E-02, -.168E-02, -.162E-02, -.152E-02, -.137E-02, -.112E-
02, -.698E-03,
-.940E-04, .670E-03, .160E-02, .271E-02, .406E-02, .581E-02, .833E-02, .123E-01, .187E-
01, .292E-01,
.680E-04, .168E-03, -.577E-04, -.570E-03, -.120E-02, -.184E-02, -.244E-02, -.297E-02, -.339E-
02, -.372E-02,
-.390E-02, -.398E-02, -.402E-02, -.402E-02, -.398E-02, -.391E-02, -.382E-02, -.371E-02, -.360E-
02, -.350E-02,
-.342E-02, -.336E-02, -.332E-02, -.329E-02, -.327E-02, -.324E-02, -.321E-02, -.316E-02, -.310E-
02, -.302E-02,
-.293E-02, -.284E-02, -.275E-02, -.266E-02, -.257E-02, -.246E-02, -.232E-02, -.210E-02, -.173E-
02, -.113E-02,
-.261E-03, .852E-03, .222E-02, .389E-02, .591E-02, .840E-02, .115E-01, .154E-01, .200E-
01, .254E-01,
-.162E-03, -.484E-03, -.909E-03, -.149E-02, -.219E-02, -.295E-02, -.369E-02, -.437E-02, -.493E-
02, -.534E-02,
-.554E-02, -.564E-02, -.568E-02, -.567E-02, -.562E-02, -.553E-02, -.541E-02, -.527E-02, -.511E-
02, -.496E-02,
-.483E-02, -.471E-02, -.463E-02, -.457E-02, -.453E-02, -.451E-02, -.448E-02, -.445E-02, -.440E-

```

02,-.433E-02,
 -.423E-02,-.412E-02,-.399E-02,-.385E-02,-.370E-02,-.353E-02,-.331E-02,-.299E-02,-.250E-
 02,-.170E-02,
 -.568E-03,.874E-03,.264E-02,.474E-02,.723E-02,.101E-01,.134E-01,.168E-01,.203E-
 01,.237E-01,
 -.274E-03,-.865E-03,-.153E-02,-.231E-02,-.320E-02,-.417E-02,-.514E-02,-.604E-02,-.676E-
 02,-.728E-02,
 -.753E-02,-.764E-02,-.768E-02,-.767E-02,-.761E-02,-.751E-02,-.737E-02,-.719E-02,-.700E-
 02,-.679E-02,
 -.659E-02,-.641E-02,-.626E-02,-.616E-02,-.608E-02,-.604E-02,-.602E-02,-.600E-02,-.598E-
 02,-.593E-02,
 -.585E-02,-.573E-02,-.558E-02,-.539E-02,-.518E-02,-.492E-02,-.459E-02,-.415E-02,-.349E-
 02,-.245E-02,
 -.102E-02,.765E-03,.289E-02,.534E-02,.809E-02,.111E-01,.142E-01,.172E-01,.200E-
 01,.226E-01,
 -.364E-03,-.118E-02,-.211E-02,-.317E-02,-.435E-02,-.562E-02,-.690E-02,-.806E-02,-.899E-
 02,-.963E-02,
 -.993E-02,-.101E-01,-.101E-01,-.101E-01,-.100E-01,-.990E-02,-.974E-02,-.954E-02,-.931E-
 02,-.905E-02,
 -.879E-02,-.854E-02,-.832E-02,-.814E-02,-.802E-02,-.794E-02,-.790E-02,-.789E-02,-.788E-
 02,-.786E-02,
 -.781E-02,-.771E-02,-.755E-02,-.734E-02,-.706E-02,-.672E-02,-.627E-02,-.566E-02,-.479E-
 02,-.347E-02,
 -.169E-02,.481E-03,.298E-02,.573E-02,.863E-02,.116E-01,.144E-01,.171E-01,.195E-
 01,.217E-01,
 -.468E-03,-.155E-02,-.279E-02,-.421E-02,-.578E-02,-.745E-02,-.910E-02,-.106E-01,-.117E-
 01,-.125E-01,
 -.128E-01,-.130E-01,-.130E-01,-.130E-01,-.129E-01,-.128E-01,-.126E-01,-.124E-01,-.121E-
 01,-.118E-01,
 -.115E-01,-.112E-01,-.109E-01,-.106E-01,-.104E-01,-.103E-01,-.102E-01,-.102E-01,-.102E-
 01,-.102E-01,
 -.102E-01,-.101E-01,-.994E-02,-.972E-02,-.940E-02,-.899E-02,-.843E-02,-.766E-02,-.656E-
 02,-.491E-02,
 -.275E-02,-.161E-03,.275E-02,.582E-02,.889E-02,.118E-01,.145E-01,.169E-01,.191E-
 01,.211E-01,
 -.614E-03,-.205E-02,-.372E-02,-.563E-02,-.773E-02,-.991E-02,-.120E-01,-.138E-01,-.152E-
 01,-.160E-01,
 -.164E-01,-.165E-01,-.165E-01,-.164E-01,-.163E-01,-.162E-01,-.160E-01,-.158E-01,-.154E-
 01,-.151E-01,
 -.147E-01,-.143E-01,-.140E-01,-.137E-01,-.134E-01,-.132E-01,-.131E-01,-.130E-01,-.130E-
 01,-.130E-01,
 -.130E-01,-.129E-01,-.128E-01,-.125E-01,-.122E-01,-.117E-01,-.111E-01,-.102E-01,-.893E-
 02,-.699E-02,
 -.449E-02,-.150E-02,.180E-02,.523E-02,.858E-02,.117E-01,.144E-01,.168E-01,.188E-
 01,.208E-01,
 -.844E-03,-.285E-02,-.518E-02,-.781E-02,-.106E-01,-.135E-01,-.161E-01,-.182E-01,-.197E-
 01,-.204E-01,
 -.207E-01,-.207E-01,-.207E-01,-.206E-01,-.205E-01,-.203E-01,-.201E-01,-.198E-01,-.194E-
 01,-.190E-01,
 -.185E-01,-.181E-01,-.177E-01,-.173E-01,-.170E-01,-.168E-01,-.166E-01,-.165E-01,-.165E-
 01,-.164E-01,
 -.163E-01,-.162E-01,-.161E-01,-.158E-01,-.155E-01,-.150E-01,-.143E-01,-.134E-01,-.120E-
 01,-.986E-02,
 -.719E-02,-.400E-02,-.437E-03,.328E-02,.692E-02,.103E-01,.133E-01,.160E-01,.183E-
 01,.207E-01,
 -.114E-02,-.385E-02,-.698E-02,-.104E-01,-.140E-01,-.173E-01,-.202E-01,-.224E-01,-.238E-
 01,-.243E-01,
 -.243E-01,-.243E-01,-.242E-01,-.241E-01,-.239E-01,-.237E-01,-.234E-01,-.230E-01,-.226E-
 01,-.221E-01,
 -.217E-01,-.212E-01,-.207E-01,-.203E-01,-.200E-01,-.197E-01,-.195E-01,-.194E-01,-.193E-
 01,-.192E-01,
 -.191E-01,-.189E-01,-.187E-01,-.184E-01,-.180E-01,-.175E-01,-.168E-01,-.159E-01,-.145E-
 01,-.124E-01,
 -.983E-02,-.672E-02,-.320E-02,.566E-03,.436E-02,.802E-02,.114E-01,.146E-01,.175E-
 01,.205E-01,
 -.152E-02,-.514E-02,-.923E-02,-.136E-01,-.178E-01,-.214E-01,-.244E-01,-.264E-01,-.275E-
 01,-.276E-01,
 -.274E-01,-.273E-01,-.271E-01,-.269E-01,-.267E-01,-.264E-01,-.260E-01,-.256E-01,-.252E-
 01,-.247E-01,
 -.241E-01,-.236E-01,-.232E-01,-.228E-01,-.224E-01,-.221E-01,-.219E-01,-.217E-01,-.215E-
 01,-.213E-01,
 -.211E-01,-.209E-01,-.206E-01,-.203E-01,-.199E-01,-.194E-01,-.187E-01,-.179E-01,-.166E-
 01,-.145E-01,
 -.122E-01,-.930E-02,-.598E-02,-.232E-02,.153E-02,.542E-02,.925E-02,.130E-01,.166E-
 01,.205E-01,
 -.203E-02,-.684E-02,-.121E-01,-.173E-01,-.219E-01,-.256E-01,-.283E-01,-.300E-01,-.307E-
 01,-.304E-01,
 -.300E-01,-.298E-01,-.295E-01,-.293E-01,-.290E-01,-.286E-01,-.282E-01,-.277E-01,-.272E-


```

.696E+01, .635E+01, .610E+01, .594E+01, .575E+01, .553E+01, .528E+01, .499E+01, .465E+
01, .427E+01,
.384E+01, .340E+01, .297E+01, .255E+01, .213E+01, .173E+01, .135E+01, .987E+00, .647E+
00, .330E+00,
.379E-01, -.229E+00, -.471E+00, -.690E+00, -.887E+00, -.106E+01, -.122E+01, -.137E+01, -.150E+
01, -.162E+01,
-.173E+01, -.183E+01, -.193E+01, -.202E+01, -.210E+01, -.218E+01, -.225E+01, -.231E+01, -.236E+
01, -.240E+01,
-.240E+01, -.237E+01, -.230E+01, -.221E+01, -.211E+01, -.200E+01, -.189E+01, -.175E+01, -.152E+
01, .000E+00,
.651E+01, .631E+01, .613E+01, .595E+01, .576E+01, .554E+01, .528E+01, .499E+01, .465E+
01, .427E+01,
.384E+01, .340E+01, .297E+01, .254E+01, .213E+01, .173E+01, .135E+01, .985E+00, .645E+
00, .328E+00,
.362E-01, -.231E+00, -.473E+00, -.691E+00, -.888E+00, -.107E+01, -.123E+01, -.137E+01, -.150E+
01, -.162E+01,
-.173E+01, -.184E+01, -.193E+01, -.202E+01, -.210E+01, -.218E+01, -.225E+01, -.231E+01, -.236E+
01, -.240E+01,
-.240E+01, -.236E+01, -.229E+01, -.220E+01, -.209E+01, -.198E+01, -.185E+01, -.171E+01, -.150E+
01, -.119E+01,
.642E+01, .630E+01, .614E+01, .596E+01, .577E+01, .554E+01, .528E+01, .499E+01, .465E+
01, .427E+01,
.384E+01, .340E+01, .297E+01, .254E+01, .213E+01, .173E+01, .135E+01, .985E+00, .644E+
00, .327E+00,
.355E-01, -.231E+00, -.473E+00, -.692E+00, -.889E+00, -.107E+01, -.123E+01, -.137E+01, -.150E+
01, -.162E+01,
-.173E+01, -.184E+01, -.193E+01, -.202E+01, -.210E+01, -.218E+01, -.225E+01, -.231E+01, -.236E+
01, -.240E+01,
-.240E+01, -.237E+01, -.230E+01, -.220E+01, -.210E+01, -.198E+01, -.187E+01, -.175E+01, -.163E+
01, -.154E+01,
.639E+01, .629E+01, .614E+01, .597E+01, .577E+01, .555E+01, .529E+01, .499E+01, .465E+
01, .427E+01,
.384E+01, .340E+01, .297E+01, .254E+01, .213E+01, .173E+01, .135E+01, .984E+00, .643E+
00, .327E+00,
.348E-01, -.232E+00, -.474E+00, -.693E+00, -.890E+00, -.107E+01, -.123E+01, -.137E+01, -.150E+
01, -.162E+01,
-.173E+01, -.184E+01, -.193E+01, -.202E+01, -.210E+01, -.218E+01, -.225E+01, -.231E+01, -.236E+
01, -.240E+01,
-.240E+01, -.237E+01, -.230E+01, -.221E+01, -.210E+01, -.200E+01, -.190E+01, -.181E+01, -.174E+
01, -.172E+01,
.639E+01, .629E+01, .615E+01, .598E+01, .578E+01, .555E+01, .529E+01, .499E+01, .465E+
01, .427E+01,
.384E+01, .340E+01, .297E+01, .254E+01, .213E+01, .173E+01, .135E+01, .983E+00, .643E+
00, .326E+00,
.340E-01, -.233E+00, -.475E+00, -.694E+00, -.890E+00, -.107E+01, -.123E+01, -.137E+01, -.150E+
01, -.162E+01,
-.174E+01, -.184E+01, -.193E+01, -.202E+01, -.211E+01, -.218E+01, -.225E+01, -.231E+01, -.237E+
01, -.240E+01,
-.241E+01, -.237E+01, -.231E+01, -.222E+01, -.212E+01, -.202E+01, -.194E+01, -.187E+01, -.183E+
01, -.182E+01,
.640E+01, .629E+01, .616E+01, .599E+01, .579E+01, .556E+01, .530E+01, .500E+01, .466E+
01, .427E+01,
.384E+01, .340E+01, .297E+01, .254E+01, .213E+01, .173E+01, .134E+01, .982E+00, .641E+
00, .325E+00,
.329E-01, -.234E+00, -.476E+00, -.695E+00, -.891E+00, -.107E+01, -.123E+01, -.137E+01, -.150E+
01, -.162E+01,
-.174E+01, -.184E+01, -.193E+01, -.202E+01, -.211E+01, -.218E+01, -.225E+01, -.232E+01, -.237E+
01, -.241E+01,
-.242E+01, -.239E+01, -.232E+01, -.224E+01, -.215E+01, -.206E+01, -.199E+01, -.193E+01, -.189E+
01, -.188E+01,
.642E+01, .631E+01, .617E+01, .601E+01, .581E+01, .558E+01, .531E+01, .501E+01, .466E+
01, .427E+01,
.384E+01, .340E+01, .297E+01, .254E+01, .213E+01, .173E+01, .134E+01, .981E+00, .640E+
00, .323E+00,
.314E-01, -.235E+00, -.477E+00, -.696E+00, -.893E+00, -.107E+01, -.123E+01, -.137E+01, -.151E+
01, -.163E+01,
-.174E+01, -.184E+01, -.194E+01, -.202E+01, -.211E+01, -.218E+01, -.226E+01, -.232E+01, -.237E+
01, -.241E+01,
-.243E+01, -.240E+01, -.235E+01, -.227E+01, -.219E+01, -.211E+01, -.204E+01, -.199E+01, -.196E+
01, -.193E+01,
.644E+01, .634E+01, .620E+01, .603E+01, .583E+01, .560E+01, .533E+01, .502E+01, .467E+
01, .428E+01,
.384E+01, .340E+01, .297E+01, .254E+01, .212E+01, .172E+01, .134E+01, .979E+00, .638E+
00, .321E+00,
.296E-01, -.237E+00, -.479E+00, -.698E+00, -.894E+00, -.107E+01, -.123E+01, -.138E+01, -.151E+
01, -.163E+01,
-.174E+01, -.184E+01, -.194E+01, -.203E+01, -.211E+01, -.219E+01, -.226E+01, -.232E+01, -.238E+
01, -.242E+01,

```

```

-.244E+01,-.242E+01,-.238E+01,-.231E+01,-.224E+01,-.217E+01,-.211E+01,-.206E+01,-.202E+
01,-.198E+01,
.648E+01,.638E+01,.624E+01,.607E+01,.587E+01,.563E+01,.536E+01,.504E+01,.468E+
01,.428E+01,
.384E+01,.340E+01,.297E+01,.254E+01,.212E+01,.172E+01,.134E+01,.977E+00,.636E+
00,.319E+00,
.271E-01,-.240E+00,-.482E+00,-.700E+00,-.896E+00,-.107E+01,-.123E+01,-.138E+01,-.151E+
01,-.163E+01,
-.174E+01,-.184E+01,-.194E+01,-.203E+01,-.211E+01,-.219E+01,-.226E+01,-.233E+01,-.239E+
01,-.244E+01,
-.247E+01,-.246E+01,-.242E+01,-.237E+01,-.231E+01,-.225E+01,-.219E+01,-.214E+01,-.209E+
01,-.204E+01,
.655E+01,.644E+01,.631E+01,.614E+01,.593E+01,.568E+01,.540E+01,.507E+01,.470E+
01,.429E+01,
.385E+01,.340E+01,.296E+01,.254E+01,.212E+01,.172E+01,.134E+01,.974E+00,.633E+
00,.316E+00,
.239E-01,-.243E+00,-.485E+00,-.703E+00,-.899E+00,-.108E+01,-.124E+01,-.138E+01,-.151E+
01,-.163E+01,
-.174E+01,-.184E+01,-.194E+01,-.203E+01,-.211E+01,-.219E+01,-.227E+01,-.234E+01,-.240E+
01,-.246E+01,
-.250E+01,-.250E+01,-.249E+01,-.245E+01,-.240E+01,-.235E+01,-.229E+01,-.223E+01,-.217E+
01,-.210E+01,
.663E+01,.653E+01,.640E+01,.622E+01,.600E+01,.575E+01,.545E+01,.510E+01,.472E+
01,.430E+01,
.385E+01,.340E+01,.296E+01,.253E+01,.212E+01,.172E+01,.133E+01,.970E+00,.629E+
00,.312E+00,
.203E-01,-.246E+00,-.488E+00,-.706E+00,-.903E+00,-.108E+01,-.124E+01,-.138E+01,-.151E+
01,-.163E+01,
-.174E+01,-.185E+01,-.194E+01,-.203E+01,-.212E+01,-.220E+01,-.227E+01,-.234E+01,-.241E+
01,-.248E+01,
-.253E+01,-.255E+01,-.255E+01,-.253E+01,-.250E+01,-.245E+01,-.239E+01,-.233E+01,-.225E+
01,-.217E+01,
.673E+01,.663E+01,.649E+01,.631E+01,.608E+01,.581E+01,.549E+01,.513E+01,.474E+
01,.431E+01,
.385E+01,.340E+01,.296E+01,.253E+01,.212E+01,.171E+01,.133E+01,.967E+00,.626E+
00,.309E+00,
.171E-01,-.250E+00,-.491E+00,-.709E+00,-.906E+00,-.108E+01,-.124E+01,-.138E+01,-.152E+
01,-.164E+01,
-.175E+01,-.185E+01,-.194E+01,-.203E+01,-.212E+01,-.220E+01,-.228E+01,-.235E+01,-.242E+
01,-.249E+01,
-.256E+01,-.259E+01,-.260E+01,-.260E+01,-.257E+01,-.253E+01,-.248E+01,-.241E+01,-.233E+
01,-.224E+01,
.684E+01,.674E+01,.660E+01,.640E+01,.616E+01,.586E+01,.553E+01,.515E+01,.474E+
01,.431E+01,
.385E+01,.340E+01,.296E+01,.253E+01,.211E+01,.171E+01,.133E+01,.964E+00,.623E+
00,.306E+00,
.141E-01,-.253E+00,-.494E+00,-.712E+00,-.908E+00,-.108E+01,-.124E+01,-.139E+01,-.152E+
01,-.164E+01,
-.175E+01,-.185E+01,-.195E+01,-.204E+01,-.212E+01,-.220E+01,-.228E+01,-.236E+01,-.243E+
01,-.251E+01,
-.258E+01,-.262E+01,-.264E+01,-.265E+01,-.263E+01,-.260E+01,-.255E+01,-.248E+01,-.240E+
01,-.231E+01,
.697E+01,.687E+01,.671E+01,.650E+01,.623E+01,.591E+01,.556E+01,.517E+01,.475E+
01,.430E+01,
.384E+01,.340E+01,.296E+01,.253E+01,.211E+01,.171E+01,.133E+01,.962E+00,.621E+
00,.303E+00,
.113E-01,-.255E+00,-.497E+00,-.715E+00,-.911E+00,-.109E+01,-.125E+01,-.139E+01,-.152E+
01,-.164E+01,
-.175E+01,-.185E+01,-.195E+01,-.204E+01,-.212E+01,-.220E+01,-.228E+01,-.236E+01,-.244E+
01,-.252E+01,
-.259E+01,-.264E+01,-.267E+01,-.269E+01,-.268E+01,-.266E+01,-.261E+01,-.255E+01,-.247E+
01,-.238E+01,
.711E+01,.700E+01,.683E+01,.659E+01,.629E+01,.595E+01,.558E+01,.518E+01,.475E+
01,.430E+01,
.384E+01,.340E+01,.296E+01,.253E+01,.211E+01,.171E+01,.132E+01,.960E+00,.618E+
00,.301E+00,
.889E-02,-.258E+00,-.499E+00,-.717E+00,-.913E+00,-.109E+01,-.125E+01,-.139E+01,-.152E+
01,-.164E+01,
-.175E+01,-.185E+01,-.195E+01,-.204E+01,-.212E+01,-.221E+01,-.228E+01,-.236E+01,-.244E+
01,-.253E+01,
-.260E+01,-.266E+01,-.270E+01,-.272E+01,-.272E+01,-.270E+01,-.266E+01,-.260E+01,-.253E+
01,-.244E+01,
.727E+01,.715E+01,.695E+01,.666E+01,.633E+01,.597E+01,.559E+01,.518E+01,.475E+
01,.429E+01,
.384E+01,.339E+01,.295E+01,.253E+01,.211E+01,.171E+01,.132E+01,.958E+00,.616E+
00,.299E+00,
.673E-02,-.260E+00,-.502E+00,-.719E+00,-.915E+00,-.109E+01,-.125E+01,-.139E+01,-.152E+
01,-.164E+01,

```

```

-.175E+01,-.186E+01,-.195E+01,-.204E+01,-.213E+01,-.221E+01,-.229E+01,-.237E+01,-.245E+
01,-.254E+01,
-.261E+01,-.267E+01,-.271E+01,-.274E+01,-.275E+01,-.274E+01,-.271E+01,-.265E+01,-.259E+
01,-.251E+01,
.746E+01,.732E+01,.706E+01,.673E+01,.636E+01,.599E+01,.560E+01,.519E+01,.475E+
01,.429E+01,
.384E+01,.339E+01,.295E+01,.252E+01,.211E+01,.170E+01,.132E+01,.956E+00,.614E+
00,.297E+00,
.477E-02,-.262E+00,-.504E+00,-.721E+00,-.917E+00,-.109E+01,-.125E+01,-.139E+01,-.152E+
01,-.164E+01,
-.175E+01,-.186E+01,-.195E+01,-.204E+01,-.213E+01,-.221E+01,-.229E+01,-.237E+01,-.245E+
01,-.254E+01,
-.261E+01,-.267E+01,-.272E+01,-.275E+01,-.277E+01,-.276E+01,-.274E+01,-.269E+01,-.263E+
01,-.257E+01,
.770E+01,.751E+01,.716E+01,.677E+01,.638E+01,.600E+01,.560E+01,.519E+01,.475E+
01,.428E+01,
.383E+01,.339E+01,.295E+01,.252E+01,.210E+01,.170E+01,.132E+01,.954E+00,.613E+
00,.295E+00,
.283E-02,-.264E+00,-.506E+00,-.723E+00,-.919E+00,-.109E+01,-.125E+01,-.140E+01,-.153E+
01,-.165E+01,
-.176E+01,-.186E+01,-.195E+01,-.204E+01,-.213E+01,-.221E+01,-.229E+01,-.237E+01,-.245E+
01,-.254E+01,
-.262E+01,-.268E+01,-.273E+01,-.276E+01,-.278E+01,-.278E+01,-.276E+01,-.272E+01,-.266E+
01,-.260E+01,
.803E+01,.770E+01,.723E+01,.679E+01,.638E+01,.600E+01,.560E+01,.519E+01,.475E+
01,.427E+01,
.383E+01,.339E+01,.295E+01,.252E+01,.210E+01,.170E+01,.132E+01,.952E+00,.610E+
00,.293E+00,
.655E-03,-.266E+00,-.508E+00,-.725E+00,-.921E+00,-.110E+01,-.125E+01,-.140E+01,-.153E+
01,-.165E+01,
-.176E+01,-.186E+01,-.195E+01,-.204E+01,-.213E+01,-.221E+01,-.229E+01,-.237E+01,-.245E+
01,-.255E+01,
-.262E+01,-.268E+01,-.273E+01,-.276E+01,-.278E+01,-.279E+01,-.277E+01,-.273E+01,-.268E+
01,-.261E+01,
.851E+01,.789E+01,.727E+01,.678E+01,.638E+01,.599E+01,.560E+01,.518E+01,.474E+
01,.427E+01,
.383E+01,.339E+01,.295E+01,.252E+01,.210E+01,.170E+01,.131E+01,.951E+00,.609E+
00,.291E+00,
-.104E-02,-.268E+00,-.509E+00,-.727E+00,-.923E+00,-.110E+01,-.126E+01,-.140E+01,-.153E+
01,-.165E+01,
-.176E+01,-.186E+01,-.196E+01,-.205E+01,-.213E+01,-.221E+01,-.229E+01,-.237E+01,-.245E+
01,-.255E+01,
-.262E+01,-.268E+01,-.273E+01,-.277E+01,-.279E+01,-.279E+01,-.278E+01,-.274E+01,-.268E+
01,-.261E+01,
.635E+01,.610E+01,.594E+01,.575E+01,.553E+01,.528E+01,.499E+01,.465E+01,.427E+
01,.384E+01,
.340E+01,.297E+01,.255E+01,.213E+01,.173E+01,.135E+01,.987E+00,.647E+00,.330E+
00,.379E-01,
-.229E+00,-.471E+00,-.690E+00,-.887E+00,-.106E+01,-.122E+01,-.137E+01,-.150E+01,-.162E+
01,-.173E+01,
-.183E+01,-.193E+01,-.202E+01,-.210E+01,-.218E+01,-.225E+01,-.231E+01,-.236E+01,-.240E+
01,-.240E+01,
-.237E+01,-.230E+01,-.221E+01,-.211E+01,-.200E+01,-.189E+01,-.175E+01,-.152E+01,.000E+
00,-.274E+01,
*** VTurbK
.114E+01,.114E+01,.114E+01,.114E+01,.114E+01,.114E+01,.114E+01,.114E+01,.114E+
01,.114E+01,
.114E+01,.114E+01,.114E+01,.114E+01,.114E+01,.114E+01,.114E+01,.114E+01,.114E+
01,.114E+01,
*** VEpsilon
.431E+01,.431E+01,.431E+01,.431E+01,.431E+01,.431E+01,.431E+01,.431E+01,.431E+
01,.431E+01,
.431E+01,.431E+01,.431E+01,.431E+01,.431E+01,.431E+01,.431E+01,.431E+01,.431E+
01,.431E+01,
*** TurbK
.114E+01,.113E+01,.112E+01,.111E+01,.109E+01,.107E+01,.105E+01,.103E+01,.101E+
01,.977E+00,
.949E+00,.922E+00,.898E+00,.874E+00,.851E+00,.830E+00,.809E+00,.790E+00,.771E+
00,.753E+00,
.736E+00,.720E+00,.704E+00,.689E+00,.675E+00,.661E+00,.648E+00,.635E+00,.623E+
00,.611E+00,
.600E+00,.589E+00,.579E+00,.569E+00,.560E+00,.551E+00,.544E+00,.536E+00,.530E+
00,.524E+00,
.521E+00,.518E+00,.518E+00,.519E+00,.520E+00,.523E+00,.526E+00,.531E+00,.537E+
00,.547E+00,
.114E+01,.113E+01,.112E+01,.111E+01,.109E+01,.107E+01,.105E+01,.103E+01,.101E+
01,.977E+00,
.949E+00,.922E+00,.898E+00,.874E+00,.851E+00,.830E+00,.809E+00,.790E+00,.771E+

```



```

01, .977E+00,
.949E+00, .923E+00, .898E+00, .874E+00, .852E+00, .830E+00, .810E+00, .790E+00, .772E+
00, .754E+00,
.737E+00, .721E+00, .705E+00, .690E+00, .676E+00, .662E+00, .649E+00, .636E+00, .623E+
00, .611E+00,
.600E+00, .589E+00, .578E+00, .568E+00, .558E+00, .548E+00, .539E+00, .529E+00, .521E+
00, .512E+00,
.505E+00, .499E+00, .495E+00, .491E+00, .488E+00, .485E+00, .483E+00, .481E+00, .480E+
00, .479E+00,
.114E+01, .113E+01, .112E+01, .111E+01, .109E+01, .107E+01, .105E+01, .103E+01, .101E+
01, .977E+00,
.949E+00, .923E+00, .898E+00, .874E+00, .852E+00, .830E+00, .810E+00, .790E+00, .772E+
00, .754E+00,
.737E+00, .721E+00, .705E+00, .690E+00, .675E+00, .661E+00, .648E+00, .635E+00, .622E+
00, .610E+00,
.599E+00, .588E+00, .577E+00, .566E+00, .556E+00, .546E+00, .537E+00, .527E+00, .518E+
00, .510E+00,
.502E+00, .497E+00, .492E+00, .488E+00, .485E+00, .482E+00, .480E+00, .478E+00, .477E+
00, .475E+00,
.114E+01, .113E+01, .112E+01, .111E+01, .109E+01, .107E+01, .105E+01, .103E+01, .101E+
01, .977E+00,
.949E+00, .923E+00, .898E+00, .874E+00, .852E+00, .830E+00, .810E+00, .790E+00, .771E+
00, .753E+00,
.736E+00, .719E+00, .703E+00, .688E+00, .673E+00, .659E+00, .645E+00, .632E+00, .619E+
00, .606E+00,
.594E+00, .583E+00, .571E+00, .561E+00, .550E+00, .540E+00, .530E+00, .520E+00, .511E+
00, .502E+00,
.495E+00, .489E+00, .484E+00, .480E+00, .476E+00, .474E+00, .472E+00, .470E+00, .468E+
00, .467E+00,
.114E+01, .113E+01, .112E+01, .111E+01, .109E+01, .107E+01, .105E+01, .103E+01, .101E+
01, .977E+00,
.949E+00, .922E+00, .897E+00, .873E+00, .850E+00, .828E+00, .807E+00, .787E+00, .768E+
00, .749E+00,
.731E+00, .714E+00, .697E+00, .681E+00, .665E+00, .650E+00, .636E+00, .622E+00, .609E+
00, .596E+00,
.583E+00, .571E+00, .559E+00, .548E+00, .537E+00, .526E+00, .516E+00, .506E+00, .496E+
00, .487E+00,
.480E+00, .474E+00, .469E+00, .465E+00, .461E+00, .459E+00, .457E+00, .455E+00, .454E+
00, .452E+00,
.114E+01, .113E+01, .112E+01, .111E+01, .109E+01, .107E+01, .105E+01, .103E+01, .101E+
01, .976E+00,
.948E+00, .921E+00, .894E+00, .869E+00, .845E+00, .822E+00, .800E+00, .778E+00, .758E+
00, .738E+00,
.718E+00, .700E+00, .682E+00, .665E+00, .649E+00, .633E+00, .618E+00, .603E+00, .589E+
00, .575E+00,
.562E+00, .550E+00, .538E+00, .527E+00, .516E+00, .505E+00, .495E+00, .485E+00, .476E+
00, .467E+00,
.459E+00, .454E+00, .449E+00, .445E+00, .442E+00, .439E+00, .437E+00, .436E+00, .435E+
00, .434E+00,
.114E+01, .113E+01, .112E+01, .111E+01, .109E+01, .107E+01, .105E+01, .103E+01, .100E+
01, .973E+00,
.943E+00, .914E+00, .885E+00, .858E+00, .832E+00, .806E+00, .782E+00, .759E+00, .736E+
00, .715E+00,
.694E+00, .675E+00, .656E+00, .638E+00, .621E+00, .605E+00, .589E+00, .575E+00, .561E+
00, .548E+00,
.535E+00, .523E+00, .512E+00, .501E+00, .491E+00, .481E+00, .472E+00, .463E+00, .455E+
00, .447E+00,
.441E+00, .436E+00, .432E+00, .429E+00, .426E+00, .424E+00, .423E+00, .421E+00, .420E+
00, .419E+00,
.114E+01, .113E+01, .112E+01, .111E+01, .109E+01, .107E+01, .105E+01, .103E+01, .999E+
00, .964E+00,
.930E+00, .897E+00, .865E+00, .834E+00, .805E+00, .777E+00, .751E+00, .725E+00, .702E+
00, .679E+00,
.658E+00, .639E+00, .620E+00, .603E+00, .586E+00, .571E+00, .557E+00, .544E+00, .532E+
00, .520E+00,
.510E+00, .500E+00, .491E+00, .483E+00, .475E+00, .468E+00, .462E+00, .456E+00, .451E+
00, .446E+00,
.443E+00, .441E+00, .439E+00, .437E+00, .436E+00, .435E+00, .434E+00, .433E+00, .432E+
00, .432E+00,
.114E+01, .113E+01, .112E+01, .111E+01, .109E+01, .107E+01, .105E+01, .102E+01, .984E+
00, .943E+00,
.904E+00, .866E+00, .830E+00, .796E+00, .764E+00, .734E+00, .707E+00, .682E+00, .659E+
00, .638E+00,
.619E+00, .602E+00, .587E+00, .573E+00, .561E+00, .550E+00, .540E+00, .532E+00, .525E+
00, .519E+00,
.514E+00, .510E+00, .508E+00, .506E+00, .505E+00, .504E+00, .505E+00, .506E+00, .508E+
00, .511E+00,
.513E+00, .515E+00, .517E+00, .519E+00, .520E+00, .521E+00, .522E+00, .522E+00, .522E+

```

```

00, .522E+00,
.114E+01, .113E+01, .112E+01, .111E+01, .109E+01, .107E+01, .104E+01, .100E+01, .960E+
00, .910E+00,
.864E+00, .822E+00, .784E+00, .750E+00, .720E+00, .694E+00, .672E+00, .653E+00, .637E+
00, .625E+00,
.615E+00, .608E+00, .603E+00, .600E+00, .599E+00, .599E+00, .602E+00, .606E+00, .611E+
00, .617E+00,
.624E+00, .632E+00, .640E+00, .650E+00, .660E+00, .670E+00, .680E+00, .691E+00, .701E+
00, .712E+00,
.721E+00, .728E+00, .734E+00, .739E+00, .743E+00, .746E+00, .749E+00, .750E+00, .752E+
00, .752E+00,
.114E+01, .113E+01, .113E+01, .112E+01, .111E+01, .108E+01, .105E+01, .101E+01, .962E+
00, .911E+00,
.868E+00, .833E+00, .805E+00, .784E+00, .769E+00, .759E+00, .755E+00, .754E+00, .758E+
00, .765E+00,
.774E+00, .787E+00, .801E+00, .817E+00, .834E+00, .852E+00, .871E+00, .890E+00, .910E+
00, .929E+00,
.948E+00, .966E+00, .984E+00, .100E+01, .102E+01, .103E+01, .105E+01, .106E+01, .108E+
01, .109E+01,
.110E+01, .111E+01, .111E+01, .112E+01, .112E+01, .113E+01, .113E+01, .113E+01, .114E+
01, .114E+01,
.126E+01, .144E+01, .157E+01, .165E+01, .168E+01, .169E+01, .167E+01, .164E+01, .161E+
01, .157E+01,
.154E+01, .150E+01, .147E+01, .144E+01, .141E+01, .139E+01, .136E+01, .134E+01, .132E+
01, .130E+01,
.128E+01, .127E+01, .125E+01, .123E+01, .122E+01, .121E+01, .120E+01, .119E+01, .118E+
01, .117E+01,
.116E+01, .115E+01, .114E+01, .113E+01, .113E+01, .112E+01, .111E+01, .111E+01, .110E+
01, .110E+01,
.109E+01, .109E+01, .109E+01, .109E+01, .108E+01, .108E+01, .108E+01, .108E+01, .108E+
01, .108E+01,
*** Epsilon
.428E+01, .422E+01, .414E+01, .405E+01, .395E+01, .383E+01, .370E+01, .355E+01, .338E+
01, .320E+01,
.303E+01, .287E+01, .273E+01, .259E+01, .247E+01, .235E+01, .224E+01, .214E+01, .204E+
01, .196E+01,
.187E+01, .179E+01, .172E+01, .165E+01, .159E+01, .152E+01, .147E+01, .141E+01, .136E+
01, .131E+01,
.127E+01, .122E+01, .118E+01, .114E+01, .110E+01, .107E+01, .104E+01, .100E+01, .976E+
00, .948E+00,
.927E+00, .911E+00, .899E+00, .890E+00, .884E+00, .879E+00, .877E+00, .877E+00, .880E+
00, .891E+00,
.428E+01, .421E+01, .414E+01, .405E+01, .395E+01, .383E+01, .370E+01, .355E+01, .338E+
01, .320E+01,
.303E+01, .287E+01, .273E+01, .259E+01, .247E+01, .235E+01, .224E+01, .214E+01, .204E+
01, .196E+01,
.187E+01, .179E+01, .172E+01, .165E+01, .159E+01, .152E+01, .147E+01, .141E+01, .136E+
01, .131E+01,
.127E+01, .122E+01, .118E+01, .114E+01, .110E+01, .107E+01, .104E+01, .100E+01, .975E+
00, .948E+00,
.926E+00, .910E+00, .898E+00, .888E+00, .882E+00, .878E+00, .875E+00, .875E+00, .876E+
00, .881E+00,
.428E+01, .421E+01, .414E+01, .405E+01, .395E+01, .383E+01, .370E+01, .355E+01, .338E+
01, .320E+01,
.303E+01, .287E+01, .273E+01, .259E+01, .247E+01, .235E+01, .224E+01, .214E+01, .204E+
01, .196E+01,
.187E+01, .179E+01, .172E+01, .165E+01, .159E+01, .153E+01, .147E+01, .141E+01, .136E+
01, .131E+01,
.127E+01, .122E+01, .118E+01, .114E+01, .110E+01, .107E+01, .103E+01, .100E+01, .974E+
00, .946E+00,
.924E+00, .907E+00, .894E+00, .885E+00, .878E+00, .873E+00, .870E+00, .868E+00, .868E+
00, .869E+00,
.428E+01, .421E+01, .414E+01, .405E+01, .395E+01, .383E+01, .370E+01, .355E+01, .338E+
01, .320E+01,
.303E+01, .287E+01, .273E+01, .259E+01, .247E+01, .235E+01, .224E+01, .214E+01, .204E+
01, .196E+01,
.187E+01, .179E+01, .172E+01, .165E+01, .159E+01, .153E+01, .147E+01, .141E+01, .136E+
01, .131E+01,
.127E+01, .122E+01, .118E+01, .114E+01, .110E+01, .107E+01, .103E+01, .100E+01, .972E+
00, .944E+00,
.921E+00, .904E+00, .890E+00, .880E+00, .872E+00, .866E+00, .862E+00, .860E+00, .859E+
00, .858E+00,
.428E+01, .421E+01, .414E+01, .405E+01, .395E+01, .383E+01, .370E+01, .355E+01, .338E+
01, .320E+01,
.303E+01, .287E+01, .273E+01, .259E+01, .247E+01, .235E+01, .224E+01, .214E+01, .205E+
01, .196E+01,
.187E+01, .180E+01, .172E+01, .165E+01, .159E+01, .153E+01, .147E+01, .141E+01, .136E+
01, .131E+01,

```



```

.189E+01, .181E+01, .174E+01, .167E+01, .160E+01, .154E+01, .149E+01, .143E+01, .138E+
01, .133E+01,
.128E+01, .124E+01, .120E+01, .116E+01, .112E+01, .108E+01, .105E+01, .101E+01, .984E+
00, .954E+00,
.930E+00, .911E+00, .896E+00, .883E+00, .873E+00, .864E+00, .857E+00, .851E+00, .847E+
00, .842E+00,
.428E+01, .421E+01, .414E+01, .405E+01, .395E+01, .383E+01, .370E+01, .355E+01, .339E+
01, .321E+01,
.304E+01, .289E+01, .275E+01, .262E+01, .250E+01, .239E+01, .229E+01, .219E+01, .210E+
01, .201E+01,
.193E+01, .185E+01, .178E+01, .171E+01, .165E+01, .158E+01, .153E+01, .147E+01, .142E+
01, .137E+01,
.132E+01, .128E+01, .124E+01, .120E+01, .116E+01, .113E+01, .109E+01, .106E+01, .103E+
01, .101E+01,
.983E+00, .966E+00, .951E+00, .940E+00, .930E+00, .922E+00, .915E+00, .910E+00, .906E+
00, .901E+00,
.428E+01, .421E+01, .414E+01, .405E+01, .395E+01, .384E+01, .371E+01, .357E+01, .342E+
01, .326E+01,
.311E+01, .298E+01, .285E+01, .273E+01, .261E+01, .251E+01, .240E+01, .231E+01, .222E+
01, .213E+01,
.205E+01, .197E+01, .189E+01, .183E+01, .176E+01, .170E+01, .164E+01, .159E+01, .154E+
01, .149E+01,
.145E+01, .141E+01, .137E+01, .133E+01, .130E+01, .127E+01, .124E+01, .122E+01, .120E+
01, .117E+01,
.116E+01, .114E+01, .113E+01, .113E+01, .112E+01, .111E+01, .111E+01, .110E+01, .110E+
01, .109E+01,
.428E+01, .421E+01, .414E+01, .406E+01, .398E+01, .389E+01, .380E+01, .370E+01, .360E+
01, .349E+01,
.337E+01, .326E+01, .315E+01, .303E+01, .292E+01, .282E+01, .271E+01, .261E+01, .251E+
01, .242E+01,
.234E+01, .226E+01, .219E+01, .212E+01, .206E+01, .201E+01, .196E+01, .191E+01, .187E+
01, .184E+01,
.181E+01, .178E+01, .176E+01, .174E+01, .173E+01, .172E+01, .171E+01, .171E+01, .171E+
01, .171E+01,
.171E+01, .172E+01, .172E+01, .172E+01, .172E+01, .172E+01, .172E+01, .172E+01, .172E+
01, .171E+01,
.428E+01, .423E+01, .419E+01, .417E+01, .417E+01, .419E+01, .423E+01, .426E+01, .426E+
01, .423E+01,
.416E+01, .406E+01, .395E+01, .383E+01, .371E+01, .359E+01, .348E+01, .338E+01, .328E+
01, .320E+01,
.312E+01, .306E+01, .301E+01, .297E+01, .295E+01, .293E+01, .292E+01, .292E+01, .293E+
01, .295E+01,
.298E+01, .301E+01, .306E+01, .310E+01, .316E+01, .322E+01, .328E+01, .335E+01, .342E+
01, .349E+01,
.355E+01, .361E+01, .365E+01, .368E+01, .371E+01, .372E+01, .373E+01, .374E+01, .374E+
01, .373E+01,
.430E+01, .441E+01, .462E+01, .493E+01, .530E+01, .569E+01, .606E+01, .636E+01, .655E+
01, .661E+01,
.656E+01, .645E+01, .632E+01, .619E+01, .606E+01, .595E+01, .587E+01, .582E+01, .579E+
01, .579E+01,
.582E+01, .587E+01, .595E+01, .605E+01, .617E+01, .631E+01, .646E+01, .663E+01, .680E+
01, .699E+01,
.718E+01, .738E+01, .759E+01, .779E+01, .799E+01, .820E+01, .840E+01, .859E+01, .878E+
01, .896E+01,
.911E+01, .923E+01, .932E+01, .939E+01, .945E+01, .949E+01, .952E+01, .953E+01, .954E+
01, .954E+01,
.485E+01, .642E+01, .834E+01, .104E+02, .125E+02, .143E+02, .158E+02, .168E+02, .173E+
02, .174E+02,
.172E+02, .170E+02, .167E+02, .166E+02, .164E+02, .164E+02, .164E+02, .165E+02, .166E+
02, .168E+02,
.171E+02, .174E+02, .178E+02, .182E+02, .186E+02, .190E+02, .194E+02, .198E+02, .202E+
02, .206E+02,
.210E+02, .214E+02, .217E+02, .220E+02, .223E+02, .226E+02, .229E+02, .231E+02, .233E+
02, .235E+02,
.236E+02, .237E+02, .238E+02, .239E+02, .239E+02, .240E+02, .240E+02, .240E+02, .240E+
02, .241E+02,
.655E+02, .804E+02, .914E+02, .981E+02, .101E+03, .102E+03, .100E+03, .979E+02, .949E+
02, .916E+02,
.885E+02, .856E+02, .829E+02, .804E+02, .780E+02, .759E+02, .739E+02, .721E+02, .704E+
02, .688E+02,
.674E+02, .660E+02, .648E+02, .637E+02, .626E+02, .617E+02, .608E+02, .599E+02, .592E+
02, .585E+02,
.578E+02, .572E+02, .566E+02, .560E+02, .555E+02, .550E+02, .546E+02, .542E+02, .538E+
02, .534E+02,
.531E+02, .528E+02, .526E+02, .525E+02, .524E+02, .523E+02, .522E+02, .522E+02, .522E+
02, .522E+02,
*** TalW
.517E+00, .547E+00, .565E+00, .574E+00, .575E+00, .571E+00, .564E+00, .554E+00, .543E+

```

```
00, .530E+00,
.518E+00, .507E+00, .496E+00, .486E+00, .477E+00, .468E+00, .460E+00, .453E+00, .446E+
00, .439E+00,
.433E+00, .427E+00, .422E+00, .417E+00, .412E+00, .408E+00, .404E+00, .400E+00, .396E+
00, .393E+00,
.390E+00, .387E+00, .384E+00, .381E+00, .379E+00, .377E+00, .374E+00, .372E+00, .370E+
00, .369E+00,
.367E+00, .366E+00, .365E+00, .364E+00, .364E+00, .363E+00, .363E+00, .363E+00, .363E+
00, .362E+00,
```

```
*****
*** Arquivo 4: Arquivo "TempDuto_Externa.dat" ***
*** requerido por "tese03_c.f90" ***
*****
```

```
.8714E+03, .8661E+03, .8588E+03, .8475E+03, .8364E+03, .8198E+03, .8036E+03, .7873E+03
, .7748E+03, .7644E+03,
.7590E+03, .7569E+03, .7562E+03, .7581E+03, .7617E+03, .7645E+03, .7678E+03, .7706E+03
, .7741E+03, .7789E+03,
.7819E+03, .7863E+03, .7904E+03, .7925E+03, .7974E+03, .8001E+03, .8038E+03, .8077E+03
, .8101E+03, .8140E+03,
.8163E+03, .8207E+03, .8234E+03, .8270E+03, .8301E+03, .8343E+03, .8390E+03, .8440E+03
, .8507E+03, .8585E+03,
.8676E+03, .8780E+03, .8891E+03, .8988E+03, .9098E+03, .9188E+03, .9256E+03, .9295E+03
, .9284E+03, .9193E+03,
```

```
*****
*** Arquivo 5: Programa energy04.f90 ***
*** Escoamento do ar ***
*****
```

```
! *** Escoamento do ar
program main
parameter (Nr=5, Nrlp=6, Nrlm=4, Nz=50, Nzlp=51, Nzlm=49, Nm=13)
Real Mi (Nm), Zeta (Nm), W (Nm), AlphaMenos (Nm), AlphaMais (Nm)
Real R (Nr), Dr (Nr), Dr_N (Nrlm), Dr_S (2:Nr), AreaZ (Nr), IFromWestBoundary (Nr), &
IFromEastBoundary (Nr), INorth (0:Nr), ISouth (Nrlp), IEast (Nr), IWest (Nr), &
TWall_W (Nr), TWall_E (Nr), Phi_2 (Nr), Phi_3 (Nr), Phi_4 (Nr), Teta_2 (Nr), Teta_3 (Nr), &
Teta_4 (Nr), Ro_Entrada (Nr), Gama_Entrada (Nr), VTurbK (Nr), VEpsilon (Nr)
Real Z (Nz), Dz (Nz), Dz_E (Nzlm), Dz_W (2:Nz), IFromSouthBoundary (Nz), IFromNorthBoundary (Nz)
), &
QToNorthBoundary (Nz), QToSouthBoundary (Nz), TWall_S (Nz), TWall_N (Nz), Temp_Externa (Nz),
&
HeatTransferCoef_N (Nz), HeatTransferCoef_S (Nz), Temp_Media (Nz), FluxoCalorCond (Nz)
Real IToSouthBoundary (Nz, 2:Nm), IToNorthBoundary (Nz, 2:Nm)
Real AreaR_N (Nr, Nz), AreaR_S (Nr, Nz), Volume (Nr, Nz), Source (Nr, Nz), &
Temperatura (Nr, Nz), IMenosMeio (Nr, Nz), Sp (Nr, Nz), Sc (Nr, Nz), NablaQr (Nr, Nz), &
Phi (Nr, Nz), NewPhi (Nr, Nz), Ro (Nr, Nz), Cp (Nr, Nz), Gama (Nr, Nz), Viscosidade (Nr, Nz), &
TurbK (Nr, Nz), Epsilon (Nr, Nz), Gama_Laminar (Nr, Nz)
Real Intensity (Nr, Nz, 2:Nm), Pizao (Nr, Nz, 2:Nm)
Real MolMistura
Character*1 SkipLine
Integer Upwind_NS, Upwind_EW, Upwind_Virtual
logical Flag_Energy, Flag_Temp, Flag_NorthBoundary
common/cmm1/Mol (5), Tc (5), Pc (5), Zc (5), MiMi (5), CpA (5), CpB (5), CpC (5), CpD (5), Y (5)
common/cmm2/AK_A (5), AK_B (5), AK_C (5), AK_D (5), MolMistura
open (5, file='coroa_entrada.dat', form='formatted', mode='read')
open (6, file='energy04.dat', status='replace')
open (7, file='TempDuto_Interna.dat', status='replace')
open (8, file='TempDuto_Externa.dat', form='formatted', mode='read')
open (9, file='Fluxos.dat', status='replace')
open (10, file='TempInicial_in.dat', form='formatted', mode='read')
open (11, file='TempInicial_out.dat', status='replace')
data Y, MolMistura/0., 0., 0.79, 0.21, 0., 28.97/ !Ar
data Mol (1), Tc (1), Pc (1), Zc (1), MiMi (1), CpA (1), CpB (1), CpC (1), CpD (1) &
/44.01, 304.1, 73.8, 0.274, 0., 1.980E+01, 7.344E-02, -5.602E-05, 1.715E-08/
data Mol (2), Tc (2), Pc (2), Zc (2), MiMi (2), CpA (2), CpB (2), CpC (2), CpD (2) &
/18.015, 647.3, 221.2, 0.235, 1.8, 3.224E+01, 1.924E-03, 1.055E-05, -3.596E-09/
data Mol (3), Tc (3), Pc (3), Zc (3), MiMi (3), CpA (3), CpB (3), CpC (3), CpD (3) &
/28.013, 126.2, 33.9, 0.29, 0., 3.115E+01, -1.357E-02, 2.680E-05, -1.168E-08/
data Mol (4), Tc (4), Pc (4), Zc (4), MiMi (4), CpA (4), CpB (4), CpC (4), CpD (4) &
/31.999, 154.6, 50.4, 0.288, 0., 2.811E+01, -3.680E-06, 1.746E-05, -1.065E-08/
data Mol (5), Tc (5), Pc (5), Zc (5), MiMi (5), CpA (5), CpB (5), CpC (5), CpD (5) &
/64.063, 430.8, 78.8, 0.269, 1.6, 2.385E+01, 6.699E-02, -4.961E-05, 1.328E-08/
data AK_A (1), AK_B (1), AK_C (1), AK_D (1) /-7.215e-3, 8.015e-5, 5.477e-9, -1.053e-11/
data AK_A (2), AK_B (2), AK_C (2), AK_D (2) /7.341e-3, -1.013e-5, 1.801e-7, -9.1e-11/
data AK_A (3), AK_B (3), AK_C (3), AK_D (3) /3.919e-4, 9.816e-5, -5.067e-8, 1.504e-11/
data AK_A (4), AK_B (4), AK_C (4), AK_D (4) /-3.273e-4, 9.966e-5, -3.743e-8, 9.732e-12/
data AK_A (5), AK_B (5), AK_C (5), AK_D (5) /-8.086e-3, 6.344e-5, -1.382e-8, 2.303e-12/
data Pi, Sigma/3.141592654, 5.6696e-8/
```

```

data Refraction,Extinction,Absorption/1.0,0.001,0.001/
data Diamond,Diamond_NS,Diamond_EW/0.5,0.5,0.5/
data ReflectWall,EmissWall/0.2,0.8/
data U,Beta_Zero/24.565,1./
data RSul,Diâmetro_Equivalente/0.854,0.04/
data MaxIterNorthBoundary,Epsilon_NorthBoundary/50,0.001/
data Upwind_NS,Upwind_Virtual,Upwind_EW/1,1,1/
data Mi(1),Zeta(1),W(1)/-1.,0.,0./
data Mi(2),Zeta(2),W(2)/-0.9082483,-0.2958759,1.047197551/
data Mi(3),Zeta(3),W(3)/-0.9082483,0.2958759,1.047197551/
data Mi(4),Zeta(4),W(4)/-0.2958759,-0.9082483,1.047197551/
data Mi(5),Zeta(5),W(5)/-0.2958759,-0.2958759,1.047197551/
data Mi(6),Zeta(6),W(6)/-0.2958759,0.2958759,1.047197551/
data Mi(7),Zeta(7),W(7)/-0.2958759,0.9082483,1.047197551/
data Mi(8),Zeta(8),W(8)/0.2958759,-0.9082483,1.047197551/
data Mi(9),Zeta(9),W(9)/0.2958759,-0.2958759,1.047197551/
data Mi(10),Zeta(10),W(10)/0.2958759,0.2958759,1.047197551/
data Mi(11),Zeta(11),W(11)/0.2958759,0.9082483,1.047197551/
data Mi(12),Zeta(12),W(12)/0.9082483,-0.2958759,1.047197551/
data Mi(13),Zeta(13),W(13)/0.9082483,0.2958759,1.047197551/
read(5,9999) SkipLine
read(5,*) MaxIterEnergy,MaxIterTemp,Relax_Phi,Epsilon_Energy,Epsilon_Temp
write(6,*) 'MaxIterEnergy,MaxIterTemp,Relax_Phi,Epsilon_Energy,Epsilon_Temp'
write(6,*) MaxIterEnergy,MaxIterTemp,Relax_Phi,Epsilon_Energy,Epsilon_Temp
read(5,9999) SkipLine
read(5,*) (Dr(i),i=1,Nr)
Dr_Total=0.
do i=1,Nr
  Dr_Total=Dr_Total+Dr(i)
end do
write(6,*) 'Dr_Total ',Dr_Total
read(5,9999) SkipLine
read(5,*) (Dz(j),j=1,Nz)
Dz_Total=0.
do j=1,Nz
  Dz_Total=Dz_Total+Dz(j)
end do
write(6,*) 'Dz_Total ',Dz_Total
R(1)=0.5*Dr(1)+RSul
do i=2,Nr
  R(i)=R(i-1)+(Dr(i-1)+Dr(i))*0.5
end do
Z(1)=0.5*Dz(1)
do j=2,Nz
  Z(j)=Z(j-1)+(Dz(j-1)+Dz(j))*0.5
end do
do i=2,Nr
  Dr_S(i)=0.5*(Dr(i-1)+Dr(i))
end do
do i=1,Nr-1
  Dr_N(i)=Dr_S(i+1)
end do
do j=2,Nz
  Dz_W(j)=0.5*(Dz(j-1)+Dz(j))
end do
do j=1,Nz-1
  Dz_E(j)=Dz_W(j+1)
end do
do i=1,Nr
  AreaZ(i)=2.*Pi*R(i)*Dr(i)
  Aux1=2.*Pi*(R(i)+0.5*Dr(i))
  Aux2=2.*Pi*(R(i)-0.5*Dr(i))
  do j=1,Nz
    AreaR_N(i,j)=Aux1*Dz(j)
    AreaR_S(i,j)=Aux2*Dz(j)
    Volume(i,j)=AreaZ(i)*Dz(j)
  end do
end do
do m=2,Nm
  do i=1,Nr
    do j=1,Nz
      Pizao(i,j,m)=(AreaR_N(i,j)-AreaR_S(i,j))/W(m)
    end do
  end do
end do
! *** Alphamenos
AlphaMenos(1)=0
do m=1,Nm

```

```

    if (m.eq.Nm) then
        AlphaMais (m)=0.
    else
        AlphaMais (m)=AlphaMenos (m)+W (m) *Mi (m)
        AlphaMenos (m+1)=AlphaMais (m)
    end if
end do
! *** Condição inicial
read(8,*) (TWall_S(j),j=1,Nz)
TWall_W=293.15
read(10,*) SkipLine
read(10,*) (TWall_N(j),j=1,Nz)
read(10,*) SkipLine
read(10,*) (TWall_E(i),j=1,Nr)
read(10,*) SkipLine
do i=1,Nr
    read(10,*) (Temperatura(i,j),j=1,Nz)
end do
do i=1,Nr
    Temp=TWall_W(i)
    call Propriedades(Temp,ViscMistura,CondTermicaMistura,CpMistura,Densidade)
    Ro_Entrada(i)=Densidade
end do
do i=1,Nr
    do j=1,Nz
        Temp=Temperatura(i,j)
        call propriedades(Temp,ViscMistura,CondTermicaMistura,CpMistura,Densidade)
        Ro(i,j)=Densidade
        Cp(i,j)=CpMistura
        Viscosidade(i,j)=ViscMistura
        Gama_Laminar(i,j)=CondTermicaMistura/CpMistura
    end do
end do
! *** Temperatura média
do j=1,Nz
    Soma_Numerador=0.
    Soma_Denominador=0.
    do i=1,Nr
        Soma_Numerador=Soma_Numerador+Ro(i,j)*Volume(i,j)*Cp(i,j)*Temperatura(i,j)
        Soma_Denominador=Soma_Denominador+Ro(i,j)*Volume(i,j)*Cp(i,j)
    end do
    Temp_Media(j)=Soma_Numerador/Soma_Denominador
end do
do j=1,Nz
    call propriedades(Temp_Media(j),ViscMistura,CondTermicaMistura,CpMistura,Densidad
e)
    ViscMistura_M=ViscMistura
    Densidade_M=Densidade
    CpMistura_M=CpMistura
    Reynolds=Densidade*U*Diametro_Equivalente/ViscMistura
    Prandtl=CpMistura*ViscMistura/CondTermicaMistura
    HeatTransferCoef_S(j)=0.023*CondTermicaMistura/Diametro_Equivalente*Reynolds**0.8
    * &
        Prandtl**0.4*1.01
    call propriedades(TWall_N(j),ViscMistura,CondTermicaMistura,CpMistura,Densidade)
    HeatTransferCoef_N(j)=0.023*Densidade_M*U*CpMistura_M*Reynolds**(-0.2)* &
        Prandtl**(-2./3.)*(ViscMistura_M/ViscMistura)**0.14
end do
VTurbK_Valor=0.005*U**2
VEpsilon_Valor=84.53
VTurbK(:)=VTurbK_Valor
VEpsilon(:)=VEpsilon_Valor
TurbK(:,:)=VTurbK_Valor
Epsilon(:,:)=VEpsilon_Valor
do i=1,Nr
    Gama_Entrada(i)=(Viscosidade(i,j)+0.09*Ro(i,j)*VTurbK(i)**2/VEpsilon(i))/0.7
end do
do i=1,Nr
    do j=1,Nz
        Gama(i,j)=(Viscosidade(i,j)+0.09*Ro(i,j)*TurbK(i,j)**2/Epsilon(i,j))/0.7
    end do
end do
QToSoutBoundary=0.
QToNorthBoundary=0.
do m=1,Nm
    Intensity(1,:,m)=Refraction**2*Sigma*Temperatura(1,:)**4/Pi
end do
Flag_Temp=:true.

```

```

Niter_Temp=1
print*, ' Processando... Aguarde...'
do while (Flag_Temp)
! 1. Condições de contorno e termos fontes
  IFromSouthBoundary(:)=(EmissWall*Refraction**2*Sigma*TWall_S(:)**4 &
    +ReflectWall*QToSouthBoundary(:))/Pi
  IFromNorthBoundary(:)=(EmissWall*Refraction**2*Sigma*TWall_N(:)**4 &
    +ReflectWall*QToNorthBoundary(:))/Pi
  IFromWestBoundary(:)=Refraction**2*Sigma*TWall_W(:)**4/Pi
  IFromEastBoundary(:)=Refraction**2*Sigma*Temperatura(:,Nz)**4/Pi
  Source(:,:)=Refraction**2*Sigma*Temperatura(:,:)**4/Pi
! 2. Cálculo das intensidades de radiação
! 2.1. Primeira direção (Mi=-1; Eta=Zeta=0)
!   Cálculo dos primeiros IMenosMeio(i,j)
  m=1
  do j=1,Nz
    do i=1,Nr
      Phi_4(i)=Extinction*Volume(i,j)
      Teta_4(i)=Extinction*Source(i,j)*Volume(i,j)
    end do
    INorth(Nr)=IFromNorthBoundary(j)
    if (Upwind_NS.eq.1) then !upwind
      do i=Nr,1,-1
        Phi_1=-Mi(m)*AreaR_S(i,j)
        Teta_1=-Mi(m)*AreaR_N(i,j)*INorth(i)
        IMenosMeio(i,j)=(Teta_1+Teta_4(i))/(Phi_1+Phi_4(i))
        INorth(i-1)=Intensity(i,j,m)
      end do
    else !diamond
      do i=Nr,1,-1
        Phi_1=-Mi(m)*AreaR_S(i,j)/(1.-Diamond_NS)
        Teta_1=-Mi(m)*INorth(i)*(AreaR_N(i,j)+Diamond_NS/(1.-Diamond_NS)*AreaR_S
(i,j))
        IMenosMeio(i,j)=(Teta_1+Teta_4(i))/(Phi_1+Phi_4(i))
        INorth(i-1)=(Intensity(i,j,m)-Diamond_NS*INorth(i))/(1.-Diamond_NS)
      end do
    end if
  end do !(j)
! 2.2. Demais direções
  do m=2,Nm
! 2.2.1. Classificação conforme os cosenos diretores da direção
    if (Mi(m).gt.0.) then
      if (zeta(m).gt.0.) then
! a) Mi>0, Zeta>0
        IWest(:)=IFromWestBoundary(:)
        do j=1,Nz
          if (Upwind_EW.eq.1) then !upwind
            do i=1,Nr
              Phi_3(i)=Zeta(m)*AreaZ(i)
              Teta_3(i)=Zeta(m)*AreaZ(i)*IWest(i)
            end do
          else !diamond
            do i=1,Nr
              Phi_3(i)=Zeta(m)*AreaZ(i)/Diamond_EW
              Teta_3(i)=Zeta(m)*AreaZ(i)*IWest(i)/Diamond_EW
            end do
          end if
          if (Upwind_Virtual.eq.1) then !upwind
            do i=1,Nr
              Phi_2(i)=-Pizao(i,j,m)*AlphaMais(m)
              Teta_2(i)=-Pizao(i,j,m)*AlphaMenos(m)*IMenosMeio(i,j)
            end do
          else !diamond
            do i=1,Nr
              Phi_2(i)=-Pizao(i,j,m)*AlphaMais(m)/Diamond
              Teta_2(i)=-Pizao(i,j,m)*IMenosMeio(i,j)*((1.-Diamond)/Diamond*AlphaMais(m)
)+AlphaMenos(m))
            end do
          end if
        end do
      end if
    end if
  end do
  do i=1,Nr
    Phi_4(i)=Extinction*Volume(i,j)
    Teta_4(i)=Extinction*Source(i,j)*Volume(i,j)
  end do
  ISouth(1)=IFromSouthBoundary(j)
  if (Upwind_NS.eq.1) then !upwind
    do i=1,Nr
      Phi_1=Mi(m)*AreaR_N(i,j)
      Teta_1=Mi(m)*AreaR_S(i,j)*ISouth(i)
    end do
  end if
end do
end while

```

```

Intensity(i,j,m)=(Teta_1+Teta_2(i)+Teta_3(i)+Teta_4(i))/ &
    (Phi_1+Phi_2(i)+Phi_3(i)+Phi_4(i))
ISouth(i+1)=Intensity(i,j,m)
end do
IToNorthBoundary(j,m)=ISouth(Nr1p)
IToSouthBoundary(j,m)=0.
else !diamond
do i=1,Nr
Phi_1=Mi(m)*AreaR_N(i,j)/Diamond_NS
Teta_1=Mi(m)*ISouth(i)*((1.-Diamond_NS)/Diamond_NS*AreaR_N(i,j)+AreaR_S(
i,j))
Intensity(i,j,m)=(Teta_1+Teta_2(i)+Teta_3(i)+Teta_4(i))/ &
    (Phi_1+Phi_2(i)+Phi_3(i)+Phi_4(i))
ISouth(i+1)=(Intensity(i,j,m)-(1.-Diamond_NS)*ISouth(i))/Diamond_NS
end do
IToNorthBoundary(j,m)=ISouth(Nr1p)
IToSouthBoundary(j,m)=0.
end if
if (Upwind_EW.eq.1) then !upwind
do i=1,Nr
IWest(i)=Intensity(i,j,m)
end do
else !diamond
do i=1,Nr
IWest(i)=(Intensity(i,j,m)-(1.-Diamond_EW)*IWest(i))/Diamond_EW
end do
end if
end do !j

else !Zeta<0
! b) Mi>0, Zeta<0
IEast(:)=IFromEastBoundary(:)
do j=Nz,1,-1
if (Upwind_EW.eq.1) then !upwind
do i=1,Nr
Phi_3(i)=-Zeta(m)*AreaZ(i)
Teta_3(i)=-Zeta(m)*AreaZ(i)*IEast(i)
end do
else !diamond
do i=1,Nr
Phi_3(i)=-Zeta(m)*AreaZ(i)/(1.-Diamond_EW)
Teta_3(i)=-Zeta(m)*AreaZ(i)*IEast(i)/(1.-Diamond_EW)
end do
end if
if (Upwind_Virtual.eq.1) then !upwind
do i=1,Nr
Phi_2(i)=-Pizao(i,j,m)*AlphaMais(m)
Teta_2(i)=-Pizao(i,j,m)*AlphaMenos(m)*IMenosMeio(i,j)
end do
else !diamond
do i=1,Nr
Phi_2(i)=-Pizao(i,j,m)*AlphaMais(m)/Diamond
Teta_2(i)=-Pizao(i,j,m)*IMenosMeio(i,j)*((1.-Diamond)/Diamond*AlphaMais(m)
)+AlphaMenos(m)
end do
end if
do i=1,Nr
Phi_4(i)=Extinction*Volume(i,j)
Teta_4(i)=Extinction*Source(i,j)*Volume(i,j)
end do
ISouth(1)=IFromSouthBoundary(j)
if (Upwind_NS.eq.1) then !upwind
do i=1,Nr
Phi_1=Mi(m)*AreaR_N(i,j)
Teta_1=Mi(m)*AreaR_S(i,j)*ISouth(i)
Intensity(i,j,m)=(Teta_1+Teta_2(i)+Teta_3(i)+Teta_4(i))/ &
    (Phi_1+Phi_2(i)+Phi_3(i)+Phi_4(i))
ISouth(i+1)=Intensity(i,j,m)
end do
IToNorthBoundary(j,m)=ISouth(Nr1p)
IToSouthBoundary(j,m)=0.
else !diamond
do i=1,Nr
Phi_1=Mi(m)*AreaR_N(i,j)/Diamond_NS
Teta_1=Mi(m)*ISouth(i)*((1.-Diamond_NS)/Diamond_NS*AreaR_N(i,j)+AreaR_S(
1,j))
Intensity(i,j,m)=(Teta_1+Teta_2(i)+Teta_3(i)+Teta_4(i))/ &

```



```

        (Phi_1+Phi_2(i)+Phi_3(i)+Phi_4(i))
        ISouth(i+1)=(Intensity(i,j,m)-(1.-Diamond_NS)*ISouth(i))/Diamond_NS
    end do
    IToNorthBoundary(j,m)=ISouth(Nr1p)
    IToSouthBoundary(j,m)=0.
end if
if (Upwind_EW.eq.1) then !upwind
    do i=1,Nr
        IEast(i)=Intensity(i,j,m)
    end do
else !diamond
    do i=1,Nr
        IEast(i)=(Intensity(i,j,m)-Diamond_EW*IEast(i))/(1.-Diamond_EW)
    end do
end do !j
end if !Zeta
else !Mi<0
    if (zeta(m).gt.0.) then
! c) Mi<0, Zeta>0
        IWest(:)=IFromWestBoundary(:)
    do j=1,NZ
        if (Upwind_EW.eq.1) then !upwind
            do i=1,Nr
                Phi_3(i)=Zeta(m)*AreaZ(i)
                Teta_3(i)=Zeta(m)*AreaZ(i)*IWest(i)
            end do
        else !diamond
            do i=1,Nr
                Phi_3(i)=Zeta(m)*AreaZ(i)/Diamond_EW
                Teta_3(i)=Zeta(m)*AreaZ(i)*IWest(i)/Diamond_EW
            end do
        end if
        if (Upwind_Virtual.eq.1) then !upwind
            do i=1,Nr
                Phi_2(i)=-Pizao(i,j,m)*AlphaMais(m)
                Teta_2(i)=-Pizao(i,j,m)*AlphaMenos(m)*IMenosMeio(i,j)
            end do
        else !diamond
            do i=1,Nr
                Phi_2(i)=-Pizao(i,j,m)*AlphaMais(m)/Diamond
                Teta_2(i)=-Pizao(i,j,m)*IMenosMeio(i,j)*((1.-Diamond)/Diamond*AlphaMais(m)
)+AlphaMenos(m)
            end do
        end if
    do i=1,Nr
        Phi_4(i)=Extinction*Volume(i,j)
        Teta_4(i)=Extinction*Source(i,j)*Volume(i,j)
    end do
    INorth(Nr)=IFromNorthBoundary(j)
    if (Upwind_NS.eq.1) then !upwind
        do i=Nr,1,-1
            Phi_1=-Mi(m)*AreaR_S(i,j)
            Teta_1=-Mi(m)*AreaR_N(i,j)*INorth(i)
            Intensity(i,j,m)=(Teta_1+Teta_2(i)+Teta_3(i)+Teta_4(i))/ &
                (Phi_1+Phi_2(i)+Phi_3(i)+Phi_4(i))
            INorth(i-1)=Intensity(i,j,m)
        end do
        IToNorthBoundary(j,m)=0.
        IToSouthBoundary(j,m)=INorth(0)
    else !diamond
        do i=Nr,1,-1
            Phi_1=-Mi(m)*AreaR_S(i,j)/(1.-Diamond_NS)
            Teta_1=-Mi(m)*INorth(i)*(AreaR_N(i,j)+Diamond_NS/(1.-Diamond_NS)*AreaR_S
(i,j))
            Intensity(i,j,m)=(Teta_1+Teta_2(i)+Teta_3(i)+Teta_4(i))/ &
                (Phi_1+Phi_2(i)+Phi_3(i)+Phi_4(i))
            INorth(i-1)=(Intensity(i,j,m)-Diamond_NS*INorth(i))/(1.-Diamond_NS)
        end do
        IToNorthBoundary(j,m)=0.
        IToSouthBoundary(j,m)=INorth(0)
    end if
    if (Upwind_EW.eq.1) then !upwind
        do i=1,Nr
            IWest(i)=Intensity(i,j,m)
        end do
    else !diamond
        do i=1,Nr

```

```

        IWest(i)=(Intensity(i,j,m)-(1.-Diamond_EW)*IWest(i))/Diamond_EW
    end do
end if
end do !j
else !Zeta<0
! d) Mi<0, Zeta<0
    IEast(:)=IFromEastBoundary(:)
    do j=Nz,1,-1
        if (Upwind_EW.eq.1) then !upwind
            do i=1,Nr
                Phi_3(i)=-Zeta(m)*AreaZ(i)
                Teta_3(i)=-Zeta(m)*AreaZ(i)*IEast(i)
            end do
        else !diamond
            do i=1,Nr
                Phi_3(i)=-Zeta(m)*AreaZ(i)/(1.-Diamond_EW)
                Teta_3(i)=-Zeta(m)*AreaZ(i)*IEast(i)/(1.-Diamond_EW)
            end do
        end if
        if (Upwind_Virtual.eq.1) then !upwind
            do i=1,Nr
                Phi_2(i)=-Pizao(i,j,m)*AlphaMais(m)
                Teta_2(i)=-Pizao(i,j,m)*AlphaMenos(m)*IMenosMeio(i,j)
            end do
        else !diamond
            do i=1,Nr
                Phi_2(i)=-Pizao(i,j,m)*AlphaMais(m)/Diamond
                Teta_2(i)=-Pizao(i,j,m)*IMenosMeio(i,j)*((1.-Diamond)/Diamond*AlphaMais(m)
)+AlphaMenos(m))
            end do
        end if
        do i=1,Nr
            Phi_4(i)=Extinction*Volume(i,j)
            Teta_4(i)=Extinction*Source(i,j)*Volume(i,j)
        end do
        INorth(Nr)=IFromNorthBoundary(j)
        if (Upwind_NS.eq.1) then !upwind
            do i=Nr,1,-1
                Phi_1=-Mi(m)*AreaR_S(i,j)
                Teta_1=-Mi(m)*AreaR_N(i,j)*INorth(i)
                Intensity(i,j,m)=(Teta_1+Teta_2(i)+Teta_3(i)+Teta_4(i))/ &
                    (Phi_1+Phi_2(i)+Phi_3(i)+Phi_4(i))
                INorth(i-1)=Intensity(i,j,m)
            end do
            IToNorthBoundary(j,m)=0.
            IToSouthBoundary(j,m)=INorth(0)
        else !diamond
            do i=Nr,1,-1
                Phi_1=-Mi(m)*AreaR_S(i,j)/(1.-Diamond_NS)
                Teta_1=-Mi(m)*INorth(i)*(AreaR_N(i,j)+Diamond_NS/(1.-Diamond_NS)*AreaR_S
(i,j))
                Intensity(i,j,m)=(Teta_1+Teta_2(i)+Teta_3(i)+Teta_4(i))/ &
                    (Phi_1+Phi_2(i)+Phi_3(i)+Phi_4(i))
                INorth(i-1)=(Intensity(i,j,m)-Diamond_NS*INorth(i))/(1.-Diamond_NS)
            end do
            IToNorthBoundary(j,m)=0.
            IToSouthBoundary(j,m)=INorth(0)
        end if
        if (Upwind_EW.eq.1) then !upwind
            do i=1,Nr
                IEast(i)=Intensity(i,j,m)
            end do
        else !diamond
            do i=1,Nr
                IEast(i)=(Intensity(i,j,m)-Diamond_EW*IEast(i))/(1.-Diamond_EW)
            end do
        end if
    end do !j
end if !Zeta
end if !Mi
! 2.2.2. Cálculo de IMenosMeio(i,j) para a próxima direção
if (Upwind_Virtual.eq.1) then !upwind
    do j=1,Nz
        do i=1,Nr
            IMenosMeio(i,j)=Intensity(i,j,m)
        end do
    end do
else

```



```

do j=1,Nz
do i=1,Nr
  IMenosMeio(i,j)=(Intensity(i,j,m)-(1.-Diamond)*IMenosMeio(i,j))/Diamond
end do
end do
end if
end do ! fim do ciclo das direções "m"
! 3. Cálculo dos fluxos de radiação para cada volume elementar
do j=1,Nz
do i=1,Nr
  G=0
  do m=2,Nm
    G=G+W(m)*Intensity(i,j,m)
  end do
  NablaQr(i,j)=Absorption*(4.*Sigma*Temperatura(i,j)**4-G)
end do
end do
! 4. Cálculo de QToNorthBoundary(j) e QToSouthBoundary(j)
do j=1,Nz
  QToNorthBoundary(j)=0.
  QToSouthBoundary(j)=0.
  do m=2,Nm
    QToNorthBoundary(j)=QToNorthBoundary(j)+abs(W(m)*Mi(m)*IToNorthBoundary(j,m))
    QToSouthBoundary(j)=QToSouthBoundary(j)+abs(W(m)*Mi(m)*IToSouthBoundary(j,m))
  end do
end do
Niter_Energy=1
Flag_Energy=.true.
Phi(:,:)=Temperatura(:,:)
do while (Flag_Energy)
! *** North Boundary
  i=Nr
  do j=1,Nz
! ** face Norte
    An=0.
    Phi_N=0.
! ** face Sul
    Gama_Interf=Prop_Interp(Gama(Nr-1,j),Dr(Nr-1),Gama(Nr,j),Dr(Nr))
    Diffusion=Gama_Interf*AreaR_S(Nr,j)/Dr_S(Nr)
    As=Diffusion*Beta_Zero
    Phi_S=Phi(Nr-1,j)
! ** face Este
    if (j.eq.Nz) then
      Ae=0.
      Phi_E=0. !Sem condicao de contorno
    else
      Ro_Interf=Prop_Interp(Ro(Nr,j),Dz(j),Ro(Nr,j+1),Dz(j+1))
      Gama_Interf=Prop_Interp(Gama(Nr,j),Dz(j),Gama(Nr,j+1),Dz(j+1))
      Convection=Ro_Interf*U*AreaZ(Nr)
      Diffusion=Gama_Interf*AreaZ(Nr)/Dz_E(j)
      Peclet=Ro_Interf*U*Dz_E(j)/Gama_Interf
      Ae=-Convection*(0.5+Alfa(Peclet))+Diffusion*Beta(Peclet)
      Phi_E=Phi(Nr,j+1)
    end if
! ** face Oeste
    Convection=Ro(i,j)*U*AreaZ(i)
    if (j.eq.1) then
      Ro_Interf=Ro_Entrada(i)
      Gama_Interf=Gama_Entrada(i)
      DeltaZ_W=Dz(1)
      Convection=Ro_Interf*U*AreaZ(i)
      Diffusion=Gama_Interf*AreaZ(i)/DeltaZ_W
      Peclet=Ro_Interf*U*DeltaZ_W/Gama_Interf
      Aw=Convection*(0.5+Alfa(Peclet))+Diffusion*Beta(Peclet)
      Phi_W=2.*TWall_W(i)-Phi(i,j) !Valor estabelecido na interface
    else
      Ro_Interf=Prop_Interp(Ro(Nr,j-1),Dz(j-1),Ro(Nr,j),Dz(j))
      Gama_Interf=Prop_Interp(Gama(Nr,j-1),Dz(j-1),Gama(Nr,j),Dz(j))
      Convection=Ro_Interf*U*AreaZ(Nr)
      Diffusion=Gama_Interf*AreaZ(Nr)/Dz_W(j)
      Peclet=Ro(Nr,j)*U*Dz_W(j)/Gama_Interf
      Aw=Convection*(0.5+Alfa(Peclet))+Diffusion*Beta(Peclet)
      Phi_W=Phi(Nr,j-1)
    end if
! ** termo fonte
    Sp(i,j)=0.
    Sc(i,j)=(-NablaQr(i,j)+HeatTransferCoef_N(j)*AreaR_N(i,j)*(TWall_N(j)-Temp_Media(j)))/Volume(i,j)/Cp(i,j)

```

```

! ** novo valor de Phi
  Aux1=An*Phi_N+As*Phi_S+Ae*Phi_E+Aw*Phi_W+Sc(i,j)*Volume(i,j)
  Aux2=(An+As+Ae+Aw-Sp(i,j)*Volume(i,j))
  NewPhi(i,j)=Relax_Phi*Aux1/Aux2+(1.-Relax_Phi)*Phi(i,j)
end do
! *** South Boundary
  i=1
  do j=1,Nz
! ** face Norte
    Gama_Interf=Prop_Interp(Gama(1,j),Dr(1),Gama(1+1,j),Dr(1+1))
    Diffusion=Gama_Interf*AreaR_N(1,j)/Dr_N(1)
    An=Diffusion*Beta_Zero
    Phi_N=Phi(1+1,j)
! ** face Sul
    As=0.
    Phi_S=0.
! ** face Este
    if (j.eq.Nz) then
      Ae=0.
      Phi_E=0. !Sem condicao de contorno
    else
      Ro_Interf=Prop_Interp(Ro(1,j),Dz(j),Ro(1,j+1),Dz(j+1))
      Gama_Interf=Prop_Interp(Gama(1,j),Dz(j),Gama(1,j+1),Dz(j+1))
      Convection=Ro_Interf*U*AreaZ(1)
      Diffusion=Gama_Interf*AreaZ(1)/Dz_E(j)
      Peclet=Ro_Interf*U*Dz_E(j)/Gama_Interf
      Ae=-Convection*(0.5-Alfa(Peclet))
      Ae=Ae+Diffusion*Beta(Peclet)
      Phi_E=Phi(1,j+1)
    end if
! ** face Oeste
    if (j.eq.1) then
      Ro_Interf=Ro_Entrada(i)
      Gama_Interf=Gama_Entrada(i)
      DeltaZ_W=Dz(1)
      Convection=Ro_Interf*U*AreaZ(i)
      Diffusion=Gama_Interf*AreaZ(i)/DeltaZ_W
      Peclet=Ro_Interf*U*DeltaZ_W/Gama_Interf
      Aw=Convection*(0.5+Alfa(Peclet))+Diffusion*Beta(Peclet)
      Phi_W=2.*TWall_W(i)-Phi(i,j) !Valor estabelecido na interface
    else
      Ro_Interf=Prop_Interp(Ro(1,j-1),Dz(j-1),Ro(1,j),Dz(j))
      Gama_Interf=Prop_Interp(Gama(1,j-1),Dz(j-1),Gama(1,j),Dz(j))
      Convection=Ro_Interf*U*AreaZ(1)
      Diffusion=Gama_Interf*AreaZ(1)/Dz_W(j)
      Peclet=Ro_Interf*U*Dz_W(j)/Gama(1,j)
      Aw=Convection*(0.5+Alfa(Peclet))+Diffusion*Beta(Peclet)
      Phi_W=Phi(1,j-1)
    end if
! ** termo fonte
    Sp(i,j)=0.
    Sc(i,j)=(-NablaQr(i,j)+HeatTransferCoef_S(j)*AreaR_S(i,j)*(TWall_S(j)-Temp_Media(j)))/Volume(i,j)/Cp(i,j)
! ** novo valor de Phi
    Aux1=An*Phi_N+As*Phi_S+Ae*Phi_E+Aw*Phi_W+Sc(i,j)*Volume(i,j)
    Aux2=(An+As+Ae+Aw-Sp(i,j)*Volume(i,j))
    NewPhi(i,j)=Relax_Phi*Aux1/Aux2+(1.-Relax_Phi)*Phi(i,j)
  end do
! *** East Boundary
  j=Nz
  do i=2,Nr-1
! ** face Norte
    Gama_Interf=Prop_Interp(Gama(i,Nz),Dr(i),Gama(i+1,Nz),Dr(i+1))
    Diffusion=Gama_Interf*AreaR_N(i,Nz)/Dr_N(i)
    An=Diffusion*Beta_Zero
    Phi_N=Phi(i+1,Nz)
! ** face Sul
    Gama_Interf=Prop_Interp(Gama(i-1,Nz),Dr(i-1),Gama(i,Nz),Dr(i))
    Diffusion=Gama_Interf*AreaR_S(i,Nz)/Dr_S(i)
    As=Diffusion*Beta_Zero
    Phi_S=Phi(i-1,Nz)
! ** face Este
    Ae=0.
    Phi_E=0. !Sem condicao de contorno
! ** face Oeste
    Ro_Interf=Prop_Interp(Ro(i,Nz-1),Dz(Nz-1),Ro(i,Nz),Dz(Nz))
    Gama_Interf=Prop_Interp(Gama(i,Nz-1),Dz(Nz-1),Gama(i,Nz),Dz(Nz))
    Convection=Ro_Interf*U*AreaZ(1)

```

```

Diffusion=Gama_Interf*AreaZ(i)/Dz_W(Nz)
Peclet=Ro_Interf*U*Dz_W(Nz)/Gama_Interf
Aw=Convection*(0.5+Alfa(Peclet))+Diffusion*Beta(Peclet)
Phi_W=Phi(i,Nz-1)
! ** termo fonte
Sp(i,j)=0.
Sc(i,j)=-NablaQr(i,j)/Cp(i,j)
! ** novo valor de Phi
Aux1=An*Phi_N+As*Phi_S+Ae*Phi_E+Aw*Phi_W+Sc(i,j)*Volume(i,j)
Aux2=(An+As+Ae+Aw-Sp(i,j)*Volume(i,j))
NewPhi(i,j)=Relax_Phi*Aux1/Aux2+(1.-Relax_Phi)*Phi(i,j)
end do
! *** West Boundary
j=1
do i=2,Nr-1
! ** face Norte
Gama_Interf=Prop_Interp(Gama(i,1),Dr(i),Gama(i+1,1),Dr(i+1))
Diffusion=Gama_Interf*AreaR_N(i,1)/Dr_N(i)
An=Diffusion*Beta_Zero
Phi_N=Phi(i+1,1)
! ** face Sul
Gama_Interf=Prop_Interp(Gama(i-1,1),Dr(i-1),Gama(i,1),Dr(i))
Diffusion=Gama_Interf*AreaR_S(i,1)/Dr_S(i)
As=Diffusion*Beta_Zero
Phi_S=Phi(i-1,1)
! ** face Este
Ro_Interf=Prop_Interp(Ro(i,1),Dz(1),Ro(i,1+1),Dz(1+1))
Gama_Interf=Prop_Interp(Gama(i,1),Dz(1),Gama(i,1+1),Dz(1+1))
Convection=Ro_Interf*U*AreaZ(i)
Diffusion=Gama_Interf*AreaZ(i)/Dz_E(1)
Peclet=Ro_Interf*U*Dz_E(1)/Gama_Interf
Ae=-Convection*(0.5-Alfa(Peclet))+Diffusion*Beta(Peclet)
Phi_E=Phi(i,1+1)
! ** face Oeste
Ro_Interf=Ro_Entrada(i)
Gama_Interf=Gama_Entrada(i)
DeltaZ_W=Dz(1)
Convection=Ro_Interf*U*AreaZ(i)
Diffusion=Gama_Interf*AreaZ(i)/DeltaZ_W
Peclet=Ro_Interf*U*DeltaZ_W/Gama_Interf
Aw=Convection*(0.5+Alfa(Peclet))+Diffusion*Beta(Peclet)
Phi_W=2.*TWall_W(i)-Phi(i,j) !Valor estabelecido na interface
! ** termo fonte
Sp(i,j)=0.
Sc(i,j)=-NablaQr(i,j)/Cp(i,j)
! ** novo valor de Phi
Aux1=An*Phi_N+As*Phi_S+Ae*Phi_E+Aw*Phi_W+Sc(i,j)*Volume(i,j)
Aux2=(An+As+Ae+Aw-Sp(i,j)*Volume(i,j))
NewPhi(i,j)=Relax_Phi*Aux1/Aux2+(1.-Relax_Phi)*Phi(i,j)
end do
! *** Internal Points
do j=2,Nz-1
do i=2,Nr-1
! ** face Norte
Gama_Interf=Prop_Interp(Gama(i,j),Dr(i),Gama(i+1,j),Dr(i+1))
Diffusion=Gama_Interf*AreaR_N(i,j)/Dr_N(i)
An=Diffusion*Beta_Zero
Phi_N=Phi(i+1,j)
! ** face Sul
Gama_Interf=Prop_Interp(Gama(i-1,j),Dr(i-1),Gama(i,j),Dr(i))
Diffusion=Gama_Interf*AreaR_S(i,j)/Dr_S(i)
As=Diffusion*Beta_Zero
Phi_S=Phi(i-1,j)
! ** face Este
Ro_Interf=Prop_Interp(Ro(i,j),Dz(j),Ro(i,j+1),Dz(j+1))
Gama_Interf=Prop_Interp(Gama(i,j),Dz(j),Gama(i,j+1),Dz(j+1))
Convection=Ro_Interf*U*AreaZ(i)
Diffusion=Gama_Interf*AreaZ(i)/Dz_E(j)
Peclet=Ro_Interf*U*Dz_E(j)/Gama_Interf
Ae=-Convection*(0.5-Alfa(Peclet))+Diffusion*Beta(Peclet)
Phi_E=Phi(i,j+1)
! ** face Oeste
Ro_Interf=Prop_Interp(Ro(i,j-1),Dz(j-1),Ro(i,j),Dz(j))
Gama_Interf=Prop_Interp(Gama(i,j-1),Dz(j-1),Gama(i,j),Dz(j))
Convection=Ro_Interf*U*AreaZ(i)
Diffusion=Gama_Interf*AreaZ(i)/Dz_W(j)
Peclet=Ro_Interf*U*Dz_W(j)/Gama_Interf
Aw=Convection*(0.5+Alfa(Peclet))+Diffusion*Beta(Peclet)

```

```

    Phi_W=Phi(i,j-1)
! ** termo fonte
    Sp(i,j)=0.
    Sc(i,j)=-NablaQr(i,j)/Cp(i,j)
! ** novo valor de Phi
    Aux1=An*Phi_N+As*Phi_S+Ae*Phi_E+Aw*Phi_W+Sc(i,j)*Volume(i,j)
    Aux2=(An+As+Ae+Aw-Sp(i,j))*Volume(i,j)
    NewPhi(i,j)=Relax_Phi*Aux1/Aux2+(1.-Relax_Phi)*Phi(i,j)
end do
end do
! *** Verifica convergência da temperatura
if(Niter_Energy.lt.MaxIterEnergy) then
    Niter_Energy=Niter_Energy+1
    Flag_Energy=.false.
    do j=1,nz
        do i=1,nr
            if(abs((NewPhi(i,j)-Phi(i,j))/Phi(i,j)) &
                .gt.Epsilon_Energy) Flag_Energy=.true.
        end do
    end do
    if(.not.Flag_Energy) write(*,1000) Niter_Energy !convergiu
else
    Flag_Energy=.false.
    write(*,*) 'Convergencia intermediaria: numero maximo permitido'
end if
Phi(:,:)=NewPhi(:,:)
TWall_E(:)=NewPhi(:,Nz)
end do !(do while)
! Cálculo das temperaturas na parede isolada (North Boundary)
do j=1,Nz
    Flag_NorthBoundary=.true.
    IterNorthBoundary=1
    do while (Flag_NorthBoundary)
        F=HeatTransferCoef_N(j)*(TWall_N(j)-Temperatura(Nr,j))-EmissWall* &
            (QToNorthBoundary(j)-Refraction**2*Sigma*TWall_N(j)**4)
        FLinha=HeatTransferCoef_N(j)+4.*EmissWall*Refraction**2 &
            *Sigma*TWall_N(j)**3
        NewTWall_N=TWall_N(j)-F/FLinha
        if(abs((NewTWall_N-TWall_N(j))/TWall_N(j)).gt.Epsilon_NorthBoundary) then
            TWall_N(j)=NewTWall_N
        else
            Flag_NorthBoundary=.false.
        end if
        if(IterNorthBoundary.lt.MaxIterNorthBoundary) then
            IterNorthBoundary=IterNorthBoundary+1
        else
            Flag_NorthBoundary=.false.
        end if
    end do
end do
! Verifica convergência
if(Niter_Temp.lt.MaxIterTemp) then
    Niter_Temp=Niter_Temp+1
    Flag_Temp=.false.
    do j=1,Nz
        do i=1,Nr
            if(abs((NewPhi(i,j)-Temperatura(i,j))/ &
                Temperatura(i,j)).gt.Epsilon_Temp) Flag_Temp=.true.
        end do
    end do
    if(.not.Flag_Temp) write(*,1010) Niter_Temp !convergiu
else
    Flag_Temp=.false.
end if
--Temperatura=NewPhi
do i=1,Nr
    do j=1,Nz
        Temp=Temperatura(i,j)
        call propriedades(Temp,ViscMistura,CondTermicaMistura,CpMistura,Densidade)
        Ro(i,j)=Densidade
        Cp(i,j)=CpMistura
        Viscosidade(i,j)=ViscMistura
    end do
end do
do i=1,Nr
    do j=1,Nz
        Gama(i,j)=(Viscosidade(i,j)+0.09*Ro(i,j)*TurbK(i,j)**2/Epsilon(i,j))/0.7
    end do
end do

```

```

end do
! *** Temperatura média
do j=1,Nz
  Soma_Numerador=0.
  Soma_Denominador=0.
  do i=1,Nr
    Soma_Numerador=Soma_Numerador+Ro(i,j)*Volume(i,j)*Cp(i,j)*Temperatura(i,j)
    Soma_Denominador=Soma_Denominador+Ro(i,j)*Volume(i,j)*Cp(i,j)
  end do
  Temp_Media(j)=Soma_Numerador/Soma_Denominador
end do
do j=1,Nz
  call propriedades(Temp_Media(j),ViscMistura,CondTermicaMistura,CpMistura,Densidad
e)
  ViscMistura_M=ViscMistura
  Densidade_M=Densidade
  CpMistura_M=CpMistura
  Reynolds=Densidade*U*Diametro_Equivalente/ViscMistura
  Prandtl=CpMistura*ViscMistura/CondTermicaMistura
  HeatTransferCoef_S(j)=0.023*CondTermicaMistura/Diametro_Equivalente*Reynolds**0.8
* &
  Prandtl**0.4*1.01
  call propriedades(TWall_N(j),ViscMistura,CondTermicaMistura,CpMistura,Densidade)
  HeatTransferCoef_N(j)=0.023*Densidade_M*U*CpMistura_M*Reynolds**(-0.2)* &
  Prandtl**(-2./3.)*(ViscMistura_M/ViscMistura)**0.14
end do
end do !(while)
! *** Temperatura externa do tubo interno
do j=1,Nz
  Q_Pelicula=HeatTransferCoef_S(j)*(TWall_S(j)-Temp_Media(j))
  Q_Emissao=Sigma*TWall_S(j)**4
  FluxoCalorCond(j)=QToSouthBoundary(j)*(ReflectWall-1.)+Q_Pelicula+Q_Emissao
  Temp_Externa(j)=TWall_S(j)+FluxoCalorCond(j)*0.004/35.
end do
write(7,1040) (Temp_Externa(j),j=1,Nz)
write(6,*) ' *** Temperaturas '
write(6,*) ' Temp_Externa'
write(6,1030) (Temp_Externa(j),j=1,Nz,1)
write(6,*) ' TWall_S'
write(6,1040) (TWall_S(j),j=1,Nz,1)
write(6,1030) ((Temperatura(i,j),j=1,Nz,1),i=1,Nr)
write(6,*) ' TWall_N'
write(6,1040) (TWall_N(j),j=1,Nz,1)
!
write(6,*) ' TWall_S'
write(6,1030) (TWall_S(j),j=1,Nz,1)
write(6,*) ' Temp_Media'
write(6,1040) (Temp_Media(j),j=1,Nz,1)
write(6,*) 'Temp_Media(50)= ',Temp_Media(50)
write(6,*) ' TWall_N'
write(6,1030) (TWall_N(j),j=1,Nz,1)
write(6,*) ' Z'
write(6,1030) (Z(j),j=1,Nz,1)
write(6,*) ' HeatTransferCoef_N'
write(6,1030) (HeatTransferCoef_N(j),j=1,Nz,1)
write(6,*) ' HeatTransferCoef_S'
write(6,1030) (HeatTransferCoef_S(j),j=1,Nz,1)
write(9,1040) (FluxoCalorCond(j),j=1,Nz)
write(11,*) ' TWall_N'
write(11,1030) (TWall_N(j),j=1,Nz)
write(11,*) ' TWall_E'
write(11,1030) (TWall_E(i),j=1,Nr)
write(11,*) ' Temperatura'
do i=1,Nr
  write(11,1040) (Temperatura(i,j),j=1,Nz)
end do
1000 format(' Convergencia intermediaria em ',I3,' iteracoes')
1010 format(' *****',/, &
' Convergencia final em ',i3,'-1 iteracoes',/, &
' *****')
1030 format(10(e10.4,','))
1040 format(50(e10.4,','))
9999 format(a1)
end
FUNCTION Alfa(Peclet)
  Aux=Peclet*Peclet
  Alfa=Aux/(10.+2.*Aux)
  if(Peclet.lt.0.) Alfa=-1.*Alfa

```

```

    return
end
FUNCTION Beta(Peclet)
    Aux=Peclet*Peclet
    Beta=(1.+0.005*Aux)/(1.+0.05*Aux)
    return
end
subroutine propriedades(Temp,ViscMistura,CondTermicaMistura,CpMistura,Densidade)
! *** Subrotina que atualiza propriedades função da temperatura
!
! 1...CO2
! 2...H2O
! 3...N2
! 4...O2
! 5...SO2
dimension Tr(5),Eps(5),FpZero(5),Ni(5),Phi(5,5),CondTermica(5),CalorEspecifico(5)
Real MolMistura
common/cmml/Mol(5),Tc(5),Pc(5),Zc(5),MiMi(5),CpA(5),CpB(5),CpC(5),CpD(5),Y(5)
common/cmml2/AK_A(5),AK_B(5),AK_C(5),AK_D(5),MolMistura
! Viscosidade
Tr(:)=Temp/Tc(:)
Eps(:)=0.176*(Tc(:)/(Mol(:)**3*Pc(:)**4))**(1./6.)
FpZero(1)=1.
FpZero(2)=1.+30.55*(0.292-Zc(2))**1.72*abs(0.96+0.1*(Tr(2)-0.7))
FpZero(3)=1.
FpZero(4)=1.
FpZero(5)=1.+30.55*(0.292-Zc(5))**1.72
Ni(:)=(0.807*Tr(:)**0.618-0.357*exp(-0.449*Tr(:))+0.34*exp(-4.058*Tr(:)+0.018))/Eps(
:)
do i=1,5
do j=1,5
if(j.eq.i) then
Phi(i,j)=1.
else
Phi(i,j)=(1.+(Ni(i)/Ni(j))**0.5*(Mol(j)/Mol(i))**0.25)**2/(8.*(1.+Mol(i)/Mo
l(j)))**0.5
end if
end do
end do
ViscMistura=0.
do i=1,5
Aux=0.
do j=1,5
Aux=Aux+Y(j)*Phi(i,j)
end do
ViscMistura=ViscMistura+Y(i)*Ni(i)/Aux
end do
ViscMistura=ViscMistura*1.e-7 ![1.e-6*poise] para [Ns/m2]
! Condutividade térmica [W/(m K)]
CondTermica(:)=AK_A(:)+AK_B(:)*Temp+AK_C(:)*Temp**2+AK_D(:)*Temp**3
do i=1,5
do j=1,5
if(j.eq.i) then
Phi(i,j)=1.
else
Phi(i,j)=(1.+(CondTermica(i)/CondTermica(j))**0.5*(Mol(j)/Mol(i))**0.25)**2
/(8.*(1.+Mol(i)/Mol(j)))**0.5
end if
end do
end do
CondTermicaMistura=0.
do i=1,5
Aux=0.
do j=1,5
Aux=Aux+Y(j)*Phi(i,j)
end do
CondTermicaMistura=CondTermicaMistura+Y(i)*CondTermica(i)/Aux
end do
! Calor específico a pressão constante
CalorEspecifico(:)=CpA(:)+CpB(:)*Temp+CpC(:)*Temp**2+CpD(:)*Temp**3
CpMistura=0.
do i=1,5
CpMistura=CpMistura+Y(i)*CalorEspecifico(i)
end do
CpMistura=CpMistura*1000./MolMistura ![kJ/(kmol K)] para [J/(kg K)]
! Densidade
Densidade=MolMistura/(0.062057*Temp) ![kg/m3]
end

```



```

FUNCTION PROP_INTERP(Prop_Menos,Dist_Menos,Prop_Mais,Dist_Mais)
  F_Menos=Dist_Menos/(Dist_Menos+Dist_Mais)
!   F_Mais=1.-F_Menos
  Prop_Interp=1./(F_Menos/Prop_Menos+(1.-F_Menos)/Prop_Mais)
end

```

```

*****
*** Arquivo 6: Arquivo "coroa_entrada.dat" ***
***      requerido por "energy04.f90"      ***
*****

```

```

*** MaxIterEnergy,MaxIterTemp,Relax_Phi,Epsilon_Energy,Epsilon_Temp
2,1000,1.,0.001,0.001
*** Dr ( 5) ***
.382E-02, .459E-02, .459E-02, .382E-02, .318E-02,
*** Dz ( 50) ***
.270E-01, .324E-01, .388E-01, .466E-01, .559E-01, .671E-01, .805E-01, .966E-01, .116E+
00, .139E+00,
.139E+00, .139E+00, .139E+00, .139E+00, .139E+00, .139E+00, .139E+00, .139E+00, .139E+
00, .139E+00,
.139E+00, .139E+00, .139E+00, .139E+00, .139E+00, .139E+00, .139E+00, .139E+00, .139E+
00, .139E+00,
.139E+00, .139E+00, .139E+00, .139E+00, .139E+00, .139E+00, .139E+00, .139E+00, .139E+
00, .139E+00,
.116E+00, .965E-01, .804E-01, .670E-01, .558E-01, .465E-01, .387E-01, .322E-01, .269E-
01, .269E-01,

```

```

*****
*** Arquivo 7: Arquivo "TempInicial.in.dat" ***
***      requerido por "energy04.f90"      ***
*****

```

```

TWall_N
.4650E+03, .4890E+03, .4990E+03, .4980E+03, .4960E+03, .4900E+03, .4860E+03, .4810E+03
, .4840E+03, .4860E+03,
.4960E+03, .5030E+03, .5150E+03, .5250E+03, .5380E+03, .5480E+03, .5600E+03, .5700E+03
, .5820E+03, .5910E+03,
.6020E+03, .6120E+03, .6220E+03, .6300E+03, .6410E+03, .6480E+03, .6580E+03, .6650E+03
, .6730E+03, .6800E+03,
.6880E+03, .6950E+03, .7020E+03, .7090E+03, .7160E+03, .7230E+03, .7310E+03, .7380E+03
, .7480E+03, .7570E+03,
.7680E+03, .7780E+03, .7900E+03, .7990E+03, .8090E+03, .8160E+03, .8210E+03, .8210E+03
, .8140E+03, .7920E+03,
TWall_E
.0000E+00, .0000E+00, .0000E+00, .0000E+00, .0000E+00,
Temperatura
.2944E+03, .3012E+03, .3103E+03, .3151E+03, .3238E+03, .3292E+03, .3390E+03, .3458E+03
, .3577E+03, .3672E+03,
.3802E+03, .3895E+03, .4022E+03, .4116E+03, .4236E+03, .4327E+03, .4441E+03, .4527E+03
, .4640E+03, .4723E+03,
.4833E+03, .4914E+03, .5020E+03, .5096E+03, .5199E+03, .5272E+03, .5370E+03, .5441E+03
, .5535E+03, .5603E+03,
.5693E+03, .5759E+03, .5846E+03, .5911E+03, .5994E+03, .6057E+03, .6140E+03, .6201E+03
, .6285E+03, .6345E+03,
.6420E+03, .6460E+03, .6520E+03, .6546E+03, .6595E+03, .6610E+03, .6649E+03, .6654E+03
, .6685E+03, .6683E+03,
.2922E+03, .2976E+03, .3007E+03, .3084E+03, .3131E+03, .3222E+03, .3284E+03, .3394E+03
, .3478E+03, .3614E+03,
.3709E+03, .3840E+03, .3935E+03, .4062E+03, .4155E+03, .4273E+03, .4362E+03, .4476E+03
, .4562E+03, .4674E+03,
.4757E+03, .4866E+03, .4946E+03, .5051E+03, .5127E+03, .5228E+03, .5300E+03, .5397E+03
, .5467E+03, .5560E+03,
.5629E+03, .5718E+03, .5784E+03, .5870E+03, .5934E+03, .6018E+03, .6081E+03, .6164E+03
, .6225E+03, .6309E+03,
.6358E+03, .6424E+03, .6455E+03, .6508E+03, .6526E+03, .6569E+03, .6578E+03, .6612E+03
, .6612E+03, .6642E+03,
.2941E+03, .2939E+03, .2992E+03, .3020E+03, .3098E+03, .3149E+03, .3248E+03, .3321E+03
, .3444E+03, .3544E+03,
.3679E+03, .3776E+03, .3907E+03, .4003E+03, .4125E+03, .4218E+03, .4334E+03, .4422E+03
, .4536E+03, .4622E+03,
.4733E+03, .4815E+03, .4923E+03, .5001E+03, .5105E+03, .5180E+03, .5280E+03, .5352E+03
, .5447E+03, .5517E+03,
.5609E+03, .5677E+03, .5765E+03, .5831E+03, .5916E+03, .5980E+03, .6064E+03, .6126E+03
, .6210E+03, .6270E+03,
.6344E+03, .6383E+03, .6442E+03, .6466E+03, .6513E+03, .6526E+03, .6564E+03, .6567E+03
, .6598E+03, .6596E+03,
.2922E+03, .2960E+03, .2972E+03, .3034E+03, .3069E+03, .3155E+03, .3214E+03, .3324E+03
, .3409E+03, .3548E+03,
.3646E+03, .3780E+03, .3877E+03, .4005E+03, .4101E+03, .4219E+03, .4310E+03, .4426E+03
, .4513E+03, .4627E+03,

```

Apêndice.f90

.4711E+03, .4822E+03, .4903E+03, .5009E+03, .5086E+03, .5188E+03, .5262E+03, .5360E+03
, .5431E+03, .5525E+03,
.5594E+03, .5684E+03, .5752E+03, .5838E+03, .5904E+03, .5988E+03, .6052E+03, .6136E+03
, .6197E+03, .6282E+03,
.6331E+03, .6397E+03, .6429E+03, .6481E+03, .6500E+03, .6543E+03, .6551E+03, .6585E+03
, .6584E+03, .6612E+03,
.2942E+03, .2956E+03, .3009E+03, .3032E+03, .3103E+03, .3149E+03, .3244E+03, .3314E+03
, .3437E+03, .3537E+03,
.3673E+03, .3770E+03, .3902E+03, .4000E+03, .4124E+03, .4217E+03, .4334E+03, .4423E+03
, .4539E+03, .4625E+03,
.4738E+03, .4821E+03, .4930E+03, .5009E+03, .5114E+03, .5189E+03, .5289E+03, .5362E+03
, .5458E+03, .5528E+03,
.5621E+03, .5689E+03, .5778E+03, .5844E+03, .5930E+03, .5994E+03, .6079E+03, .6141E+03
, .6226E+03, .6287E+03,
.6363E+03, .6403E+03, .6463E+03, .6488E+03, .6537E+03, .6551E+03, .6590E+03, .6593E+03
, .6622E+03, .6615E+03,