

## 4 Métodos Numéricos

As equações diferenciais parciais e as equações diferenciais estocásticas, utilizadas na modelagem da combustão turbulenta pré-misturada, discutidas no Cap. 3, requerem o emprego de métodos numéricos para que possam ser aplicadas em simulações. Neste trabalho, as primeiras são resolvidas numericamente por um programa computacional chamado *Fluids3d*, escrito em linguagem FORTRAN 90, o qual foi desenvolvido pelo grupo de pesquisa em dinâmica dos fluidos computacional da faculdade de Engenharia Mecânica da Universidade Federal de Uberlândia (Campregher, 2005). Este programa, baseado no método dos volumes finitos, foi originalmente concebido com a finalidade de simular escoamentos de fluidos incompressíveis. No presente trabalho, modificações são realizadas para possibilitar simulações de escoamentos de fluidos compressíveis a baixo número de Mach. A primeira parte do capítulo apresenta estas modificações e discute os principais aspectos dos métodos numéricos inclusos no *Fluids3d*.

A segunda parte do capítulo discute em detalhe o método de simulação de Monte Carlo, utilizado para a resolução das equações diferenciais estocásticas. Este método é caracterizado por um conjunto de procedimentos que incluem a resolução numérica das equações diferenciais estocásticas, as quais governam as propriedades das partículas lagrangeanas, e o cálculo dos momentos estatísticos das propriedades das partículas em relação aos centros dos volumes de controle usados no método dos volumes finitos. As principais características do programa híbrido, resultante deste acoplamento entre o *Fluids3d* e o programa de partículas, desenvolvido no presente trabalho, são apresentadas na seqüência.

Na parte final do capítulo, as estratégias de paralelização do programa *Fluids3d* são discutidas brevemente, e a paralelização do programa de partículas, desenvolvida no âmbito deste trabalho, é apresentada em detalhe.

#### 4.1. Método dos Volumes Finitos

Neste trabalho, o método dos volumes finitos (FVM - *Finite Volume Method*) (Patankar, 1980) é utilizado para a simulação numérica das equações de transporte da quantidade de movimento linear e do campo escalar, Eqs. (3-57) e (3-60). Tal método consiste em dividir o domínio de cálculo em volumes de controle elementares, de modo que exista um volume elementar ao redor de cada ponto de uma malha computacional euleriana. As equações de transporte são integradas em cada volume de controle elementar e as variações das propriedades do escoamento entre os pontos da malha são supostos lineares. Assim, obtém-se equações discretizadas que determinam os valores destas propriedades para o conjunto de pontos da malha computacional.

As equações obtidas deste modo expressam a conservação das propriedades de interesse no escoamento, para cada volume de controle elementar, sendo que, neste trabalho, estas propriedades são a quantidade de movimento linear e o campo escalar. Isto implica que a solução final, determinada pela integral destas propriedades em todo domínio de cálculo, resulta na conservação global das propriedades, o que infere ao método uma representatividade física do problema a ser resolvido. Note-se que esta característica independe da dimensão de malha.

Em uma malha computacional composta por volumes finitos, existem dois arranjos principais para se armazenar as propriedades de interesse do escoamento: (a) arranjos desencontrados (*staggered grids*), nos quais as componentes do vetor velocidade e do campo escalar são armazenadas nas faces dos volumes de controle, e a pressão é armazenada nos centros dos volumes de controle; (b) arranjos co-localizados (*colocated grids*), nos quais todas as variáveis, sejam elas vetores ou escalares, são armazenadas nos centros dos volumes de controle. A Figura 4.1 ilustra o armazenamento das propriedades para os dois arranjos.

Os arranjos desencontrados possuem a vantagem de apresentar um forte acoplamento entre pressão e velocidade, uma vez que os gradientes da primeira coincidem com a posição de armazenamento da segunda. Entretanto, o uso deste arranjo aumenta significativamente a complexidade de construção do algoritmo. Sendo assim, o arranjo co-localizado é utilizado neste trabalho, tendo em vista a simplicidade no desenvolvimento do algoritmo de discretização e a praticidade na

implementação de programas paralelos, nos quais a divisão do domínio principal em subdomínios menores é empregada (Campregher, 2005).

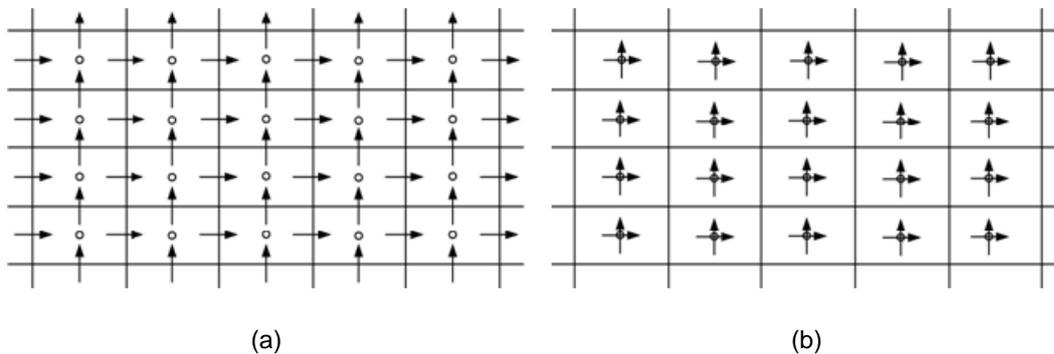


Figura 4.1 Ilustração esquemática dos arranjos das propriedades nas malhas, (a) arranjo desencontrado: as componentes do vetor velocidade são armazenadas nas faces dos volumes e a pressão é armazenada no centro, (b) arranjo co-localizado: todas as variáveis são armazenadas no centro do volume de controle. Os círculos são a pressão e as setas os componentes de velocidade (Campregher, 2005).

É importante ressaltar que, devido à simplicidade da geometria utilizada nas simulações, este trabalho emprega malhas estruturadas, as quais são caracterizadas por conjuntos de volumes de controle hexaédricos. A seção seguinte descreve o processo de discretização das equações de transporte, baseado em volumes de controle com esta topologia.

#### 4.1.1. Discretização das Equações de Transporte

A equação geral de transporte a ser discretizada tem a forma,

$$\frac{\partial \rho \Phi}{\partial t} + \nabla \cdot (\rho \mathbf{u} \Phi) = \nabla \cdot (D^\Phi \nabla \Phi) + b, \quad (4-1)$$

onde  $\Phi$  é a variável dependente, sendo, neste trabalho, os componentes do vetor velocidade,  $\mathbf{u}$ , e o campo escalar,  $\Phi$ ,  $D^\Phi$  é o coeficiente de difusão associado a  $\Phi$ , e  $b$  é o termo de geração de  $\Phi$ . O primeiro termo do lado esquerdo desta equação é o termo transiente e o segundo termo representa o transporte convectivo da variável dependente,  $\Phi$ . O primeiro termo do lado direito é o termo de transporte difusivo e o segundo termo é o termo fonte.

No método dos volumes finitos, as equações de transporte de quantidade de movimento linear e do campo escalar, Eqs. (3-57) e (3-60), são escritas na forma da equação geral de transporte, Eq.(4.1), e integradas com respeito ao tempo e ao espaço no volume de controle elementar  $P$ , mostrado na Figura 4.2.

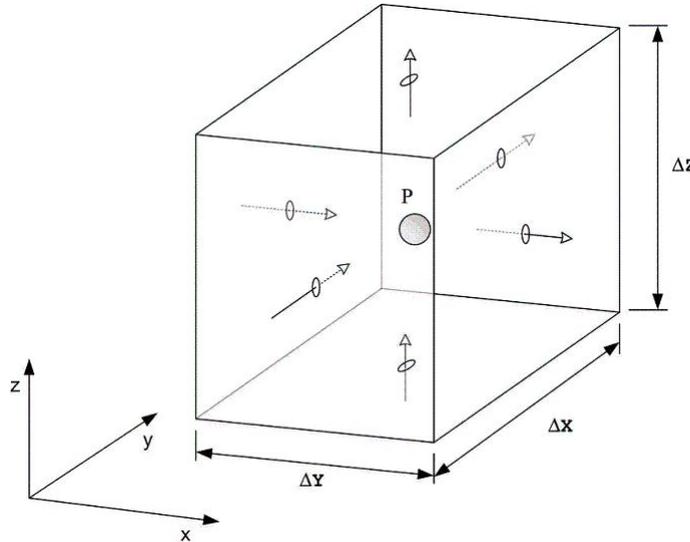


Figura 4.2 Volume de controle elementar, com topologia hexaédrica, utilizado na discretização das equações de transporte (Campregher, 2005).

Assim, a equação geral de transporte escreve-se,

$$\left[ \frac{3(\rho\Phi_P)^{n+1} - 4(\rho\Phi_P)^n + (\rho\Phi_P)^{n-1}}{2\Delta t} \right] \Delta x \Delta y \Delta z +$$

$$\left\{ (\rho_e u_e \Phi_e - \rho_w u_w \Phi_w) - \left[ \left( D_e^\Phi \frac{\partial \Phi}{\partial x} \right)_e - \left( D_w^\Phi \frac{\partial \Phi}{\partial x} \right)_w \right] \right\}^{\Delta y \Delta z} +$$

$$\left\{ (\rho_n u_n \Phi_n - \rho_s u_s \Phi_s) - \left[ \left( D_n^\Phi \frac{\partial \Phi}{\partial y} \right)_n - \left( D_s^\Phi \frac{\partial \Phi}{\partial y} \right)_s \right] \right\}^{\Delta x \Delta z} +$$

$$\left\{ (\rho_t u_t \Phi_t - \rho_b u_b \Phi_b) - \left[ \left( D_t^\Phi \frac{\partial \Phi}{\partial z} \right)_t - \left( D_b^\Phi \frac{\partial \Phi}{\partial z} \right)_b \right] \right\}^{\Delta x \Delta y} =$$

$$b \Delta x \Delta y \Delta z , \tag{4-2}$$

onde os índices  $e$  (east),  $w$  (west),  $n$  (north),  $s$  (south),  $t$  (top) e  $b$  (bottom) denotam as faces do volume de controle  $P$ , as quais são compartilhadas com os volumes vizinhos. As faces  $(e, w)$ ,  $(n, s)$ ,  $(t, b)$  são perpendiculares as direções  $x, y$  e  $z$ , respectivamente. Os nós centrais dos volumes de controle vizinhos

recebem a mesma nomenclatura destas faces compartilhadas, porém com índices maiúsculos.

O termo transiente é discretizado pelo esquema *three-time level* (Muzaferija e Peric, 1997), o qual fornece uma aproximação temporal de segunda ordem, com precisão superior a dos esquemas *Euler* de primeira ordem. Esta discretização é equivalente a ajustar uma parábola pelos instantes de tempo  $t^{n+1}$ ,  $t^n$  e  $t^{n-1}$ , separados entre si por um intervalo de tempo  $\Delta t$ . Maiores detalhes sobre este procedimento são encontrados em Campregher (2005). A implementação deste esquema é simples, entretanto, é necessário armazenar-se um conjunto de dados extra, relativo às propriedades nos instantes de tempo  $n$  e  $n - 1$ . Cabe notar, também, que se trata de um esquema implícito, uma vez que o valor da propriedade no instante de tempo  $t^{n+1}$  é desconhecido.

As aproximações numéricas para os valores do transporte de  $\Phi$ , em cada uma das faces  $i$  do volume de controle elementar  $P$ , pelos fluxos  $f_i = f_{c,i} + f_{d,i} = \rho_i u_i \Phi_i - D_i^\Phi (\partial \Phi_i / \partial x_i)$ , desempenham um papel importante na consistência e estabilidade da simulação. O termo  $f_{c,i} = \rho_i u_i \Phi_i$  é o fluxo convectivo e o termo  $f_{d,i} = D_i^\Phi (\partial \Phi_i / \partial x_i)$  representa a parcela difusiva do fluxo da propriedade que atravessa a fronteira  $i$  do volume elementar, na direção do vetor normal a face.

O cálculo destes fluxos é realizado a partir das propriedades armazenadas nos nós centrais dos volumes de controle, visto que a discretização espacial segue um arranjo co-localizado. Como consequência, é necessário o emprego de funções de interpolação entre os pontos nodais da malha computacional, de modo que seja possível obter os fluxos nas faces dos volumes.

Neste trabalho, uma estratégia conhecida como correção atrasada (Ferziger e Peric, 2002) é empregada, a qual interpola o fluxo convectivo em uma face  $i$  do volume de controle, no instante de tempo  $n + 1$ , da seguinte forma,

$$(f_{c,i})^{n+1} = (f_{c,i}^L)^{n+1} + Y(f_{c,i}^H - f_{c,i}^L)^n, \quad (4-3)$$

onde os índices  $L$  e  $H$  denotam termos de baixa e alta ordem, respectivamente, e o coeficiente  $Y$  permite uma combinação entre estes termos.

Note-se que para  $Y$  igual a zero, o esquema é puramente de baixa ordem, para  $Y$  igual a um, tem-se um esquema puramente de alta ordem, e para valores intermediários obtém-se uma combinação de esquemas de baixa e alta ordem.

Sendo assim, esta estratégia permite aproveitar-se da estabilidade dos esquemas de baixa ordem e da maior precisão dos esquemas de alta ordem. Neste trabalho, os esquemas de baixa e alta ordem disponíveis são o *upwind* e o esquema de diferenças centradas (CDS – *Central Difference Scheme*), respectivamente. Nas interpolações das derivadas de primeira ordem o esquema CDS é utilizado.

Cabe ressaltar que no trabalho de Campregher (2005) foi demonstrado que o esquema CDS implementado no código computacional que serviu como base para este trabalho é, efetivamente, de segunda ordem de precisão. A generalização para o presente caso, onde a densidade é variável, ainda não foi realizada.

Com estas definições, a equação geral de transporte pode ser escrita em sua forma final,

$$A_P \Phi_P = A_E \Phi_E + A_W \Phi_W + A_N \Phi_N + A_S \Phi_S + A_T \Phi_T + A_B \Phi_B + B , \quad (4-4a)$$

com,

$$A_I = \max(-flux_i, 0) + diff_i , \quad I = E, N, T \quad i = e, n, t , \quad (4-4b)$$

$$A_I = \max(flux_i, 0) + diff_i , \quad I = W, S, B \quad i = w, s, b , \quad (4-4c)$$

$$flux_i = \rho_i u_i S_i , \quad i = e, w, n, s, t, b , \quad (4-4d)$$

$$diff_i = D_i^\Phi S_i / \delta_i , \quad i = e, w, n, s, t, b , \quad (4-4e)$$

$$A_P = \sum_I A_I + B_P \Delta V , \quad I = E, W, N, S, T, B , \quad (4-4f)$$

$$B_C = b \Delta V , \quad (4-4g)$$

onde,  $S_i$  é a área da seção transversal da malha normal a direção  $i$ ,  $\Delta V = \Delta x \Delta y \Delta z$  é o volume de controle,  $\delta_i$  é a distância entre os pontos nodais adjacentes na direção normal a face  $i$ , e  $B_P$  e  $B_C$  são as contribuições do termo fonte, sendo o primeiro relativo aos termos acoplados a variável  $\Phi_P$ . Como mencionado, os índices maiúsculos representam os nós centrais dos volumes e os minúsculos as faces dos volumes de controle.

Convém ressaltar que, no caso da equação de quantidade de movimento linear, Eq. (3-57), o gradiente de pressão e a segunda parte do tensor das tensões viscosas estão inclusos no termo fonte,  $b$ . Como visto no início do capítulo, o programa *Fluids3d* foi originalmente concebido para simulações de escoamentos de fluidos incompressíveis. No presente trabalho, uma modificação na formulação é realizada, incluindo-se também, no termo fonte, o traço do tensor das tensões viscosas. No caso da equação de transporte do campo escalar, Eq. (3-60), o termo fonte  $b$  corresponde ao termo de reação química média. Este termo é proveniente do referencial lagrangeano, obtido mediante o uso do método de simulações de Monte Carlo, o qual será discutido na seção 4.2.

Neste contexto, a Eq. (4.4a) representa a versão discretizada das Eqs. (3-57) e (3-60), a qual é aplicada a todos os volumes de controle elementares do domínio computacional, originando um sistema de equações lineares, cuja solução fornece os valores atualizados da variável dependente,  $\Phi$ . Cabe lembrar que a variável dependente,  $\Phi$ , é representada pelos componentes do vetor velocidade,  $\mathbf{u}$ , e pelo campo escalar,  $\phi$ .

#### 4.1.2. Acoplamento Pressão-Velocidade

O acoplamento pressão-velocidade, para escoamentos a baixo número de Mach, é realizado de modo que o campo de pressão resultante satisfaça a equação de conservação de massa, Eq. (3-56). No presente trabalho, o método SIMPLEC, proposto por van Doormal e Raithby (1984), é utilizado para este fim.

O algoritmo SIMPLEC pode ser obtido reescrevendo-se a Eq.(4-4a) para o transporte das componentes do vetor velocidade estimada,  $\mathbf{u}^*$ , e separando-se o gradiente de pressão do termo fonte  $B$ ,

$$A_P \mathbf{u}_P^* - \sum_I A_I \mathbf{u}_I^* = B^n - \nabla p_P^* \Delta V, \quad (4-5)$$

onde  $p$  é a pressão, o índice  $I$  representa todos os vizinhos do nó central  $P$  e o sobrescrito  $n$  denota a variável conhecida no instante de tempo  $n$ . O subscrito,  $P$ , no gradiente de pressão, indica que este foi calculado com relação ao nó central  $P$ . O índice,  $*$ , serve para indicar uma estimativa da propriedade, visto que ainda não se sabe o valor do campo de pressão atualizado, no instante de tempo  $n + 1$ , que

satisfaça a conservação de massa. No entanto, se o campo correto de velocidade for utilizado, a Eq. (4-5) assume a forma,

$$A_P \mathbf{u}_P - \sum_I A_I \mathbf{u}_I = B^n - \nabla p_P \Delta V . \quad (4-6)$$

Subtraindo-se a Eq. (4-5) da (4-6), escreve-se,

$$A_P \mathbf{u}_P^c - \sum_I A_I \mathbf{u}_I^c = -\nabla p_P^c \Delta V , \quad (4-7)$$

onde  $\mathbf{u}^c = \mathbf{u} - \mathbf{u}^*$  e  $p^c = p - p^*$  são as correções que necessitam a ser aplicadas aos campos estimados de velocidade e pressão.

A correção das velocidades vizinhas ao nó central  $P$ , definida no segundo termo do lado esquerdo da Eq. (4-7), é substituída por uma média ponderada dos valores das correções nos termos vizinhos,

$$\mathbf{u}_P^c = \frac{\sum A_I \mathbf{u}_I^c}{\sum A_I} , \quad (4-8)$$

que, quando inserida na Eq. (4-7), e após rearranjo dos termos, leva a,

$$\mathbf{u}_P = \mathbf{u}_P^* - d_P^u \nabla p_P^c , \quad (4-9)$$

onde,

$$d_P^u = \frac{\Delta V}{A_P - \sum A_I} . \quad (4-10)$$

A Eq. (4-9) permite efetuar as correções no campo de velocidade, uma vez conhecido o campo de correção de pressão,  $p^c$ , cuja equação discretizada é obtida aplicando-se o princípio de conservação de massa às componentes  $\mathbf{u}_P$ ,

$$A_P p_P^c = A_E p_E^c + A_W p_W^c + A_N p_N^c + A_S p_S^c + A_T p_T^c + A_B p_B^c + B , \quad (4-11a)$$

com,

$$A_P = \sum_I A_I , \quad (4-11b)$$

$$A_I = \frac{\rho d_I^u S_i}{\delta_i} , \quad I = E, W, N, S, T, B, \quad i = e, w, n, s, t, b , \quad (4-11c)$$

$$B = \partial\rho/\partial t + \nabla \cdot \mathbf{u}^* , \quad (4-11d)$$

onde  $\delta_i$  é a distância, no sentido de  $i$ , entre o nó central do volume elementar  $P$  e o nó central do volume vizinho  $I$  e  $B$  representa a conservação de massa.

#### 4.1.2.1. Interpolação de Rhie-Chow

A discretização das equações de transporte é análoga nos casos dos arranjos deslocados e co-localizados. Entretanto, no arranjo co-localizado, a velocidade necessária nas faces deve ser interpolada a partir dos centros dos volumes de controle. Neste trabalho, a função de interpolação usada para este fim é conhecida como interpolação de Rhie-Chow (Rhie e Chow, 1983).

O algoritmo é concebido partindo-se da forma rearranjada da Eq. (4-5),

$$A_P \mathbf{u}_P^* + \nabla p_P^* \Delta V_P = B_P^n + \left[ \sum_I A_I \mathbf{u}_I^* \right]_P . \quad (4-12)$$

Aplicando a mesma equação para o ponto nodal,  $E$ , escreve-se,

$$A_P \mathbf{u}_E^* + \nabla p_E^* \Delta V_E = B_E^n + \left[ \sum_I A_I \mathbf{u}_I^* \right]_E , \quad (4-13)$$

e para uma face  $e$  entre os pontos nodais  $P$  e  $E$ , tem-se,

$$A_P \mathbf{u}_e^* + \nabla p_e^* \Delta V_e = B_e^n + \left[ \sum_I A_I \mathbf{u}_I^* \right]_e . \quad (4-14)$$

Considerando-se que o lado esquerdo da Eq. (4-14) pode ser aproximado, na interface  $e$  do volume de controle, por uma interpolação linear a partir das Eqs. (4-12) e (4-13), tem-se,

$$A_P \mathbf{u}_e^* + \nabla p_e^* \Delta V_e = \overleftarrow{\left[ \sum_I A_I \mathbf{u}_I^* \right]_e} = \overleftarrow{A_P} \mathbf{u}_e^* + \overleftarrow{\nabla p_e^*} \Delta V_e , \quad (4-15)$$

onde a interpolação é representada pelas setas superiores.

Rearranjando a Eq. (4-15) e assumindo que  $A_e \approx \overleftarrow{A_e}$ , escreve-se,

$$\mathbf{u}_e = \overleftarrow{\mathbf{u}}_e^* - \overleftarrow{d}_e^{\mathbf{u}} (\overleftarrow{\nabla p}_e^* - \nabla p_e^*) , \quad (4-16)$$

com,

$$\overleftarrow{\mathbf{u}}_e^* = \varrho \mathbf{u}_P^* + (1 - \varrho) \mathbf{u}_E^* , \quad (4-17)$$

$$\overleftarrow{\nabla p}_e^* = \varrho \nabla p_P^* + (1 - \varrho) \nabla p_E^* , \quad (4-18)$$

$$\nabla p_e^* = \frac{(p_E - p_P)}{\delta x_e} , \quad (4-19)$$

$$\nabla p_P^* = \frac{(p_E - p_W)}{\delta x_e + \delta x_w} , \quad (4-20)$$

$$\nabla p_E^* = \frac{(p_{EE} - p_P)}{\delta x_{ee} + \delta x_e} , \quad (4-21)$$

$$\overleftarrow{d}_e^{\mathbf{u}} = \varrho d_P^{\mathbf{u}} + (1 - \varrho) d_E^{\mathbf{u}} , \quad (4-22)$$

onde  $\varrho$  é um coeficiente de interpolação, o qual é ponderado pela distância dos pontos nodais aos centros dos volumes de controle.

A Eq. (4-16) é inserida na equação de conservação de massa, analogamente ao procedimento realizado para Eq. (4-9). É possível notar que o campo de velocidade nas faces dos volumes de controle depende, agora, dos valores das pressões nos pontos nodais adjacentes, de modo análogo aos arranjos deslocados. Assim, é possível realizar o acoplamento entre pressão e velocidade, mediante o emprego do algoritmo SIMPLEX, detalhado na seção anterior.

Nota-se que o sistema de equações algébricas a ser resolvido, Eqs. (4-4a) e (4-11a), constitui um sistema que permite determinar, a cada instante de tempo, os valores de velocidade, pressão e campo escalar, uma vez conhecidas as condições de contorno e o valor da densidade. Em problemas de combustão em escoamentos subsônicos, a densidade do fluido é determinada pela evolução termoquímica da mistura, como visto no Cap. 3.

Neste sentido, as mudanças realizadas no programa *Fluids3d*, em relação à sua formulação matemática, têm o intuito de possibilitar simulações de casos que apresentam variações da densidade do fluido, para escoamentos a baixo número de Mach. Uma vez realizadas estas modificações, o presente trabalho se atém ao desenvolvimento de um programa de partículas, baseado no método de Monte Carlo, o qual é acoplado ao *Fluids3d*, caracterizando um programa híbrido que permite a simulação de escoamentos com presença de reações químicas.

## 4.2. Método de Monte Carlo

A principal dificuldade associada à resolução das equações de transporte da PDF conjunta da velocidade e campo escalar e da PDF do campo escalar, Eqs. (3-38) e (3-44), mediante ao uso de métodos clássicos de discretização, tais como volumes finitos ou diferenças finitas, é o elevado custo computacional associado ao grande número de variáveis necessárias para se representar os processos estocásticos nos espaços amostrais de velocidade e campo escalar. Considerando-se, por exemplo, um caso de combustão em regime estacionário, descrito por  $\chi$  dimensões geométricas e  $\sigma$  variáveis escalares, a dimensão do espaço amostral da PDF conjunta da velocidade e campo escalar,  $P_{\mathbf{u},\phi}(\mathbf{V}, \boldsymbol{\psi}; \mathbf{x})$ , é  $2\chi + \sigma$  (Nooren, 1998). Quando de simulações com cinética química detalhada, o campo escalar pode ser composto por dezenas de espécies químicas, resultando em uma alta dimensionalidade, proibitiva no âmbito dos métodos clássicos de discretização. Em tais métodos, a demanda de poder computacional aumenta exponencialmente com o número de variáveis independentes envolvidas no problema (Pope, 1985).

De maneira oposta, no método de Monte Carlo o custo computacional cresce linearmente com a quantidade de variáveis independentes envolvidas. Desta forma, simulações que envolvem um grande número de escalares, tais como aquelas que consideram os mecanismos da cinética química detalhada, podem ser viabilizadas. Por este motivo, observa-se que este método é usualmente utilizado no contexto dos modelos baseados em PDF transportada (Oran e Boris, 2001).

Neste trabalho, o método de Monte Carlo utilizado pode ser caracterizado por aspectos relativos à: (a) representação do escoamento mediante uma amostra de partículas, (b) resolução numérica das equações diferenciais estocásticas, as quais são estatisticamente equivalentes à Eq. (3-44), e que descrevem a evolução das propriedades das partículas no escoamento, (c) prescrição de condições iniciais e condições de contorno das propriedades das partículas, e (d) estimativa dos momentos estatísticos das propriedades das partículas, em relação aos centros dos volumes de controle da malha computacional euleriana.

Estes aspectos são discutidos em detalhe nas quatro seções seguintes.

### 4.2.1. Propriedades das Partículas Estocásticas

No método de Monte Carlo, uma representação discreta do escoamento é obtida mediante o emprego de partículas estocásticas distribuídas no domínio de cálculo. Para uma amostra estatisticamente independente de partículas com massas idênticas, uma PDF da amostra pode ser estimada a partir das propriedades instantâneas de cada partícula. Na seção 4.2.4, será mostrado como a PDF e os momentos estatísticos das propriedades de interesse são estimados, mas antes é necessário definir quais as propriedades transportadas pelas partículas e como tais propriedades evoluem ao longo do tempo.

No presente trabalho, cada partícula é caracterizada por um vetor posição, em coordenadas cartesianas,  $\mathbf{x}$ , e por um vetor do campo escalar,  $\boldsymbol{\phi}$ , o qual, segundo as hipóteses simplificadoras adotadas neste trabalho, é representado pela variável de progresso da reação química,  $c$ . Para fins de discretização, cada partícula também carrega um vetor de números inteiros, o qual é utilizado no decorrer das simulações para facilitar a localização desta em relação aos centros dos volumes de controle da malha euleriana. Visto que as partículas não carregam informações sobre suas massas, estas são consideradas idênticas (Fox, 2003). A massa das partículas é determinada por  $m_p = M_f/N$ , sendo  $N$  o número de partículas e  $M_f$  a massa do fluido, a qual é obtida pela relação,  $M_f = \rho\mathcal{V}$ , onde  $\rho$  é a densidade e  $\mathcal{V}$  é o volume de fluido contido no problema a ser resolvido.

As posições das partículas no espaço físico,  $(x, y, z)$ , e o valor da variável de progresso,  $c$ , evoluem de acordo com as equações diferenciais estocásticas (SDEs) fornecidas pelas Eqs. (3-58) e (3-59), respectivamente. A seção seguinte aborda os métodos empregados na resolução destas equações.

### 4.2.2. Métodos de Resolução das SDEs

As equações diferenciais estocásticas (SDEs) resolvidas neste trabalho, Eqs. (3-58) e (3-59), podem ser representadas por uma equação diferencial estocástica geral, de acordo com a interpretação de Itô (Gardiner, 1990),

$$dX = A[X(t), t]dt + \sum_{j=1}^{N_p} B_j[X(t), t] dW_j(t), \quad X(0) = X_0 \quad t \in [t_0, t_{fin}] , \quad (4-23)$$

onde o primeiro termo do lado direito representa a parcela determinística e o segundo termo do lado direito é a parcela estocástica. O termo  $A[X(t), t]$  é uma função chamada de coeficiente de deriva e o termo  $B_j[X(t), t]$  é conhecido como coeficiente de difusão. A natureza estocástica do segundo termo é devida ao processo de Wiener,  $W_j$ , que descreve o movimento Browniano de cada partícula. Os incrementos do processo de Wiener,  $dW_j(t + dt) = W_j(t + dt) - W_j(t)$ , com  $j = 1, \dots, N_w$ , onde  $N_w$  é o número de processos de Wiener independentes que influenciam a evolução de  $X(t)$ , são variáveis aleatórias independentes do tipo Normal, com média zero e variância igual a  $dt$ , isto é,  $N(0, dt)$ .

Usualmente, a abordagem da simulação dos caminhos amostrais (*sample paths approach*) é empregada na resolução da Eq. (4-23). Tal abordagem consiste em uma aproximação numérica, mediante o emprego de tempos discretos, o que implica na necessidade da discretização do intervalo de tempo  $[t_0, t_{fin}]$  (Higham, 2001). No presente trabalho, esta discretização é uniforme no tempo,

$$\Delta t = \frac{t_{fin} - t_0}{NI} , \quad (4-24)$$

onde  $NI$  é o número de intervalos que dividem  $[t_0, t_{fin}]$ . O tempo discretizado pode ser expresso por,

$$t_n = t_0 + n\Delta t , \quad n = 1, \dots, NI . \quad (4-25)$$

De uma forma geral, os esquemas numéricos que resolvem as SDEs podem ser divididos em três categorias: (a) esquemas numéricos do tipo explícito, nos quais os coeficientes de deriva e de difusão são integrados de forma explícita, (b) esquemas numéricos do tipo semi-implícitos, em que o coeficiente de deriva é integrado implicitamente e o coeficiente de difusão é integrado de forma explícita, (c) esquemas numéricos implícitos, nos quais ambos os coeficientes de deriva e de difusão são integrados implicitamente. Com base nesta classificação, os esquemas numéricos de Euler-Maruyama explícito, semi-implícito e implícito, escrevem-se, respectivamente,

$$X_{n+1} = X_n + A[X_n, t_n]\Delta t + B[X_n, t_n]\Delta W_n , \quad (4-26)$$

$$X_{n+1} = X_n + A[X_{n+1}, t_{n+1}]\Delta t + B[X_n, t_n]\Delta W_n , \quad (4-27)$$

$$X_{n+1} = X_n + A[X_{n+1}, t_{n+1}]\Delta t + B[X_{n+1}, t_{n+1}]\Delta W_n , \quad (4-28)$$

e a variável aleatória  $\Delta W_n$  tem a forma  $\sqrt{\Delta t}N(0,1)$ . Cabe notar que estas equações estão escritas para um processo de Wiener unidimensional.

Todos os esquemas numéricos de resolução de SDEs do tipo Euler possuem convergência de ordem forte,  $\gamma_s$ , igual a 0,5 e convergência de ordem fraca,  $\gamma_w$ , igual a 1,0. Convém mencionar que um método numérico possui convergência de ordem forte igual a  $\gamma_s$  se existe uma constante  $K_s$  tal que,

$$E|X_n - X(t_n)| \leq K_s \Delta t^{\gamma_s} , \quad (4-29)$$

onde o termo  $E$  representa o valor médio,  $X_n$  é a aproximação de  $X(t_n)$  no tempo  $t_{fin}$ , depois de  $NI$  intervalos de tempo uniformes de valor  $\Delta t$ . A convergência de ordem forte mede a taxa na qual a média do erro de aproximação numérica decai quando o número de intervalos no tempo  $NI$  tende para o infinito, isto é, quando  $\Delta t \rightarrow 0$  (Higham, 2001).

Um método numérico possui convergência de ordem fraca,  $\gamma_w$ , se existe uma constante  $K_w$  tal que, para toda função,

$$|Ef(X_n) - Ef[X(t_n)]| \leq K_w \Delta t^{\gamma_w} , \quad (4-30)$$

onde a função  $f$  deve ser do tipo suave e polinomial. Usualmente, esta função  $f$  é escolhida da forma mais simples, como, por exemplo, uma função identidade  $f(X) = X$ . Neste caso, a convergência de ordem fraca mede a taxa de decaimento da média dos erros de aproximação numérica quando o número de intervalos no tempo  $NI$  tende para o infinito, isto é, quando  $\Delta t \rightarrow 0$  (Higham, 2001).

Neste trabalho, o método de Euler-Maruyama explícito é empregado para resolver a Eq. (3-58). Na Eq. (3-59) é empregada uma técnica de separação de operadores para a resolução da evolução no espaço amostral do campo escalar. Isto é, o termo de micro-mistura e o termo fonte de reação química são resolvidos separadamente. O primeiro é resolvido pelo método explícito, enquanto o segundo

é resolvido pelo método implícito, uma vez que a taxa de reação química, determinada pela expressão empírica de Arrhenius, Eq. (3-34), apresenta elevada não linearidade. Cabe mencionar que na Eq. (3-59) o coeficiente de difusão é igual a zero. Portanto, esta equação possui apenas o termo determinístico. Sendo assim, ambos os métodos semi-implícito e implícito são equivalentes neste caso.

### 4.2.3. Condições Iniciais e de Contorno

Para que a resolução das SDEs seja possível pelos métodos descritos na seção anterior, é necessária a prescrição de condições iniciais e de contorno para as propriedades das partículas, isto é, para as posições no espaço físico,  $\mathbf{x}$ , e para o campo escalar,  $\phi$ . Neste trabalho, as partículas são distribuídas inicialmente sobre uma malha euleriana uniforme, de maneira que cada volume de controle contém um número de partículas idêntico no instante de tempo  $t = 0$ . As posições destas partículas no espaço físico,  $\mathbf{x}$ , são aleatórias e uniformemente distribuídas. O campo escalar,  $\phi$ , também é prescrito de maneira uniforme. As simulações são iniciadas com gases frescos em todo o domínio de cálculo, ou seja, com a variável de progresso,  $c$ , igual a zero.

As condições de contorno das propriedades das partículas são impostas de acordo com a geometria do domínio de cálculo euleriano, mostrada na Figura 4.3. No presente trabalho, três tipos de condições de contorno podem ser encontradas,

- (a) Condição de entrada: a partícula que cruza a fronteira de entrada sofre uma correção na sua posição no espaço físico e na sua composição do campo escalar. A posição corrigida,  $\mathbf{x}$ , é sorteada aleatoriamente dentro dos volumes de controle de fronteira. O novo valor do campo escalar,  $\phi$ , é atribuído de acordo com as condições de entrada do problema em questão. Neste trabalho, a partícula recebe o valor  $c = 0$ , se sua posição corresponde a entrada de gases frescos, e  $c = 1$  se esta posição equivale a entrada de gases queimados,
- (b) Condição de saída: a partícula que cruza a fronteira de saída é eliminada da amostra e substituída por outra, posicionada aleatoriamente dentro dos volumes de controle de fronteira de entrada, de maneira análoga ao procedimento adotado para as condições de entrada citado acima. Os

novos valores do campo escalar,  $\phi$ , também são impostos como no procedimento das condições de entrada. Isto garante que o número total de partículas ao longo das simulações permanece constante,

- (c) Condição de parede: a partícula que cruza uma fronteira sólida sofre uma correção na sua posição, sendo refletida para o interior do domínio de cálculo. Os valores do campo escalar,  $\phi$ , permanecem inalterados.

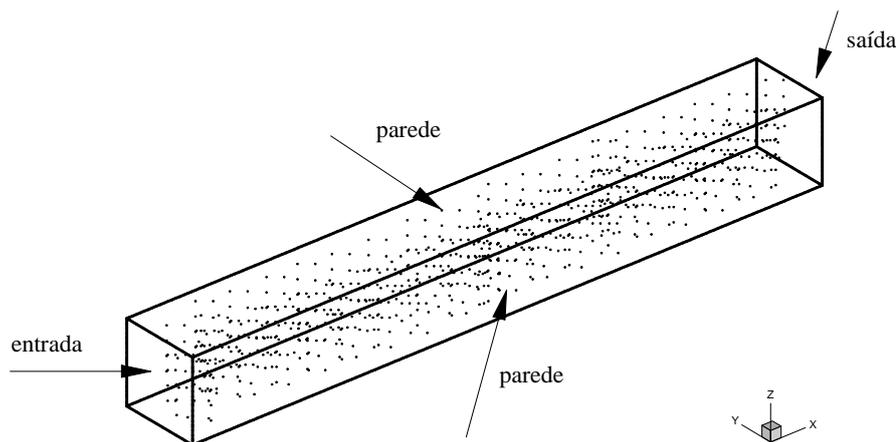


Figura 4.3 Representação da configuração geométrica utilizada com as respectivas condições de contorno atribuídas para as partículas.

É importante mencionar que no caso em que o domínio de cálculo euleriano é dividido em subdomínios menores, quando da paralelização do problema na direção principal do escoamento, a qual é paralela ao eixo  $x$  na Figura 4.3, as partículas que excedem as fronteiras de um dado subdomínio no espaço físico são enviadas para o subdomínio vizinho, mantendo suas coordenadas globais,  $\mathbf{x}$ , e os valores do campo escalar,  $\phi$ , inalterados. Este procedimento será discutido na seção 4.4, a qual trata da paralelização do programa computacional.

#### 4.2.4. Cálculo dos Momentos Estatísticos

Uma vez resolvidas as SDEs, mediante as condições iniciais e de contorno prescritas, a PDF e os momentos estatísticos do campo escalar são estimados. O método NGP (*Nearest Grid Point*), mostrado na seção 2.4, é utilizado para este fim. Neste método, somente as partículas localizadas dentro de cada volume de

controle da malha euleriana são utilizadas nos cálculos dos momentos estatísticos associados ao centro deste volume. Além disto, cada partícula contribui para a estimativa das estatísticas com um peso idêntico, independentemente de sua distância em relação ao centro do volume de controle. A PDF do campo escalar corresponde a um histograma estimado pelos valores instantâneos do campo escalar,  $\phi$ . Os momentos estatísticos de 1ª e 2ª ordens do campo escalar são determinados pela média amostral e pela variância da amostra, respectivamente.

Cabe mencionar que o método NGP foi adotado neste trabalho com base nos resultados de Peirano et al. (2006), os quais mostram que este método fornece o mesmo nível de precisão estatística do método CIC, visto na seção 2.4, com a vantagem de apresentar uma maior simplicidade na implementação do algoritmo. A Figura 4.4 ilustra de maneira esquemática o método NGP.

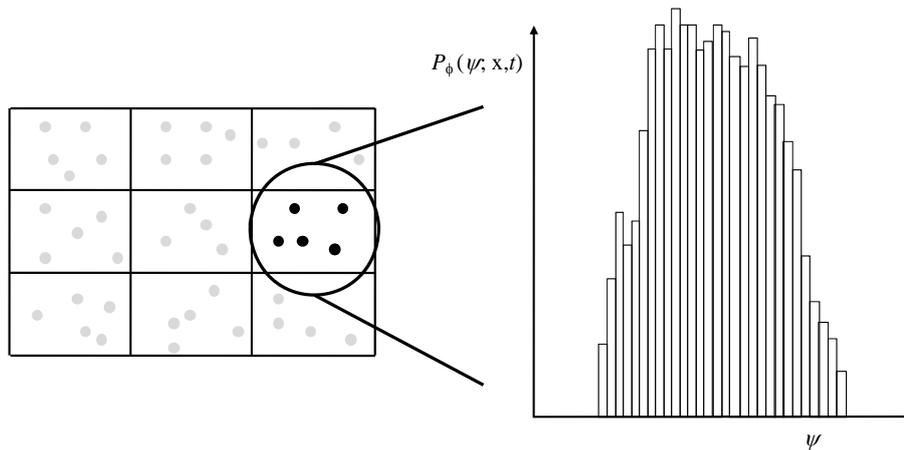


Figura 4.4 Ilustração esquemática dos volumes de controle da malha euleriana contendo as partículas utilizadas na estimativa da PDF do campo escalar, a qual é associada ao centro do volume de controle.

Devido ao número finito de partículas,  $N_p$ , presentes dentro de cada volume de controle, a média amostral do campo escalar corresponde a uma aproximação do valor esperado do campo escalar, sendo o erro estatístico definido por,

$$\varepsilon_{\phi N_p} = \langle \phi \rangle_{N_p} - \mathbf{E}\{\phi\}. \tag{4-31}$$

onde  $\langle \phi \rangle_{N_p}$  é a média amostral e  $\mathbf{E}\{\phi\}$  é o valor esperado do campo escalar.

A taxa de convergência estatística é medida pelo desvio padrão de  $\varepsilon_{\phi N_p}$  e aumenta na proporção de  $\sqrt{N_p}$  (Xu e Pope, 1999). Isto implica que o número de

partículas usado nas simulações é um parâmetro importante, o qual determina a precisão das estimativas dos momentos estatísticos do campo escalar. Percebe-se que cerca de 100 partículas, inicialmente distribuídas por volume de controle, é um número viável de ser utilizado na prática, tendo em vista a limitação imposta pela quantidade de memória computacional disponível. Nooren (1998) apresenta uma análise de sensibilidade do número de partículas empregadas em um modelo híbrido, na qual se utilizam amostras que variam de 100 a 300 partículas inicialmente distribuídas por volume de controle. Este autor conclui que, para este intervalo de número de partículas, não há diferença significativa na precisão da estimativa dos momentos estatísticos.

No presente trabalho, a média e a variância do campo escalar, associadas ao centro de cada volume de controle da malha computacional euleriana,  $V_c$ , são estimadas, respectivamente, por,

$$\bar{\Phi}_{V_c} = \frac{1}{N_p} \sum_{i=1}^{N_p} \Phi_i, \quad (4-32)$$

$$\Phi_{V_c}^2 = \frac{1}{N_p} \sum_{i=1}^{N_p} (\Phi_i - \bar{\Phi}_{V_c})^2, \quad (4-33)$$

onde  $N_p$  é o número de partículas dentro do volume de controle,  $V_c$ , e o campo escalar,  $\Phi$ , é representado pela variável de progresso da reação química,  $c$ .

A taxa de reação química média, associada ao centro do volume de controle,  $V_c$ , é estimada por,

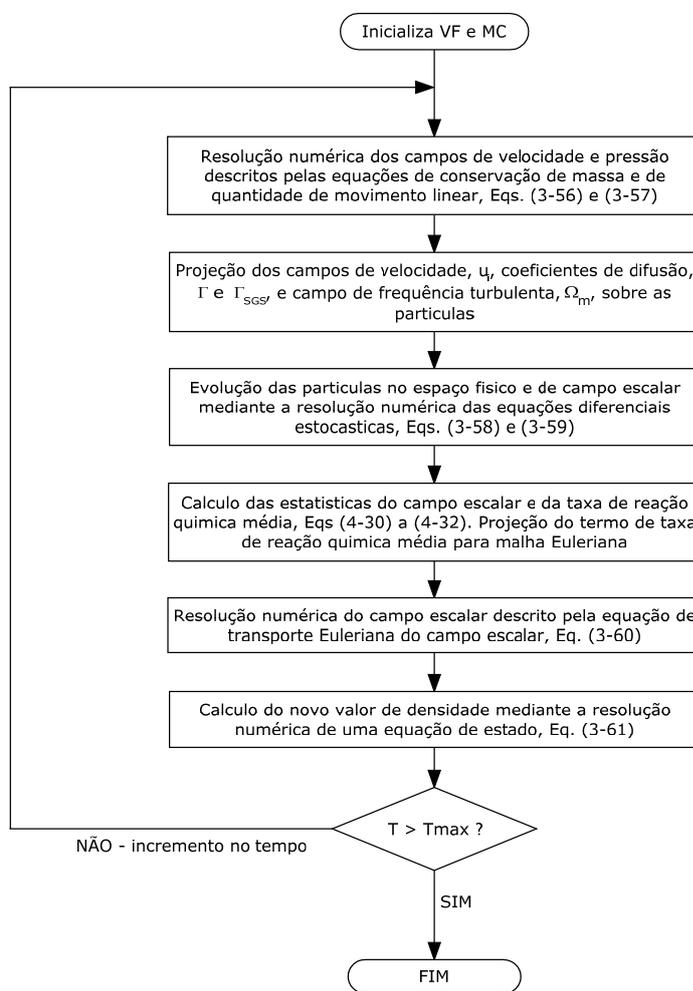
$$\bar{S}(\Phi)_{V_c} = \frac{1}{N_p} \sum_{i=1}^{N_p} S_i(\Phi), \quad (4-34)$$

onde a taxa de reação química instantânea de cada partícula,  $S_i(\Phi)$ , é calculada pela relação empírica de Arrhenius, Eq. (3-34).

Cabe mencionar que estas operações são realizadas a cada passo de tempo das simulações, de modo que os campos escalares médios, a variância do campo escalar e a taxa de reação química média são transmitidas para a malha euleriana, caracterizando a existência de um acoplamento dinâmico entre as partículas e a malha euleriana. A seção subsequente mostra em detalhe como é realizado o acoplamento entre os métodos dos volumes finitos e de Monte Carlo.

### 4.3. Dinâmica do Modelo Híbrido

Como mostrado previamente, o método dos volumes finitos e de simulação Monte Carlo, utilizados neste trabalho, são baseados em referenciais distintos, sendo o primeiro empregado em malhas eulerianas e o segundo baseado em partículas lagrangeanas. A utilização destes dois referenciais caracteriza uma classe de modelagem conhecida como híbrida, na qual o procedimento de troca de informações entre malha e partículas representa um aspecto essencial na solução numérica. Sendo assim, a presente seção apresenta o mecanismo de acoplamento entre os dois métodos, ilustrado de maneira esquemática na Figura 4.5.



PUC-Rio - Certificação Digital Nº 0510816/CA

Figura 4.5 Representação esquemática da dinâmica entre a formulação resolvida na malha euleriana, pelo método dos volumes finitos, e a formulação da PDF do campo escalar, resolvida sobre partículas estocásticas via método de Monte Carlo.

A interação entre os métodos nos dois referenciais acontece de forma dinâmica, uma vez que existe uma troca contínua de informações entre a malha euleriana e as partículas estocásticas no decorrer da simulação. Primeiramente, os campos de velocidade e os campos de pressão são inicializados e as partículas são distribuídas uniformemente sobre a malha computacional, com suas respectivas propriedades escalares iniciais. O acoplamento tem início com a resolução do campo do escoamento na malha euleriana. Para um dado passo de tempo, os campos de velocidade e pressão são resolvidos mediante a utilização conjunta das equações de transporte de massa e de quantidade de movimento linear, Eqs. (3-56) e (3-57), empregando-se os métodos numéricos apresentados nas seções 4.2.1 e 4.2.2, respectivamente.

Em seguida, as partículas são localizadas no escoamento e suas posições são vinculadas aos centros de cada volume de controle da malha euleriana que as contém. Desta maneira, o campo de velocidade,  $\tilde{u}_i$ , é projetado sobre as partículas de modo que aquelas que se encontram dentro de um mesmo volume de controle recebem valores idênticos de velocidade. Os coeficientes de difusão,  $\Gamma$  e  $\Gamma_{SGS}$ , e o campo de frequências turbulentas,  $\Omega_m$ , também são projetados sobre as partículas de maneira análoga, permitindo que estas evoluam no espaço físico e do campo escalar, de acordo com as Eqs. (3-58) e (3-59), que são resolvidas numericamente pelas Eqs. (4-26) e (4-28), respectivamente. Cabe lembrar que o campo escalar é representado pela variável de progresso da reação química,  $c$ .

Após a evolução das partículas no espaço físico e do campo escalar, o termo de taxa de reação química filtrada é determinado de acordo com a Eq. (4-34) e projetado de volta para malha euleriana, onde a equação de transporte do campo escalar, Eq. (3-60), pode ser resolvida pelo método numérico mostrado na seção 4.2.1. Cabe observar que os  $m$ -ésimos momentos estatísticos do campo escalar também podem ser obtidos pelo método de Monte Carlo e projetados para a malha euleriana. No entanto, a solução obtida pela Eq. (3-60) é mais estável, uma vez que esta não apresenta ruídos gerados, por exemplo, pelos erros estatísticos associados ao uso de um número finito de partículas estocásticas (James et al. 2007).

Com o campo escalar resolvido, o novo valor da densidade é calculado pela Eq. (3-61). Este valor é utilizado nos cálculos do próximo passo de tempo e a sequência de operações é repetida até que o tempo desejado de simulação seja alcançado.

#### 4.4. Paralelização do Programa Computacional

No método LES, as soluções numéricas das equações governantes do movimento dos fluidos requerem o emprego de um grande número de volumes de controle computacionais para que a maior parte do espectro de energia associada ao movimento turbulento seja resolvida de maneira explícita. Na abordagem lagrangeana dos modelos de PDF transportada, um número elevado de partículas é empregado para uma avaliação adequada das propriedades estatísticas da combustão. A junção destas duas abordagens, no âmbito de um modelo híbrido, resulta na exigência de uma alta capacidade de processamento de dados e requer uma considerável disponibilidade de memória computacional.

Usualmente, esta questão é solucionada mediante a divisão do domínio de cálculo original em subdomínios menores. Cada subdomínio é atribuído a uma unidade de processamento de dados (CPU – *Central Processing Unity*) diferente, de forma que estas trabalhem em conjunto para resolver o problema global. Este é o conceito principal usado na paralelização do presente programa computacional.

A forma como o problema contido no domínio original é subdividido entre os diversos subdomínios pode influenciar de maneira decisiva o desempenho do programa paralelo. Na grande maioria dos casos, os problemas são resolvidos simultaneamente em todos os processadores, mas não de forma independente. No caso de problemas de dinâmica dos fluidos computacional, para obter-se a solução em um determinado subdomínio, as informações de fronteira armazenadas nos subdomínios vizinhos são necessárias. A troca de informações, conhecida como troca de mensagens, é uma tarefa de fundamental importância no processamento paralelo, sendo que a implementação da arquitetura de troca de mensagens entre subdomínios demanda grande esforço no desenvolvimento do programa paralelo (Campregher, 2005).

O desempenho de um programa paralelo pode ser medido pela aceleração (*speedup*),  $Sp$ , definida pela razão entre o tempo de processamento em um único processador e o tempo de processamento em  $np$  processadores, resolvendo o mesmo problema,

$$Sp = \frac{t_s}{t_{np}} , \quad (4-35)$$

onde  $t_s$  é o tempo de execução do programa serial e  $t_{np}$  é o tempo de execução do programa paralelo. A aceleração ideal é obtida quando  $Sp = np$ , ou seja, quando a velocidade de processamento cresce linearmente com o aumento do número de processadores. Cabe mencionar que tal situação ideal é obtida somente nos casos em que não há etapas seriais no algoritmo.

A capacidade dos programas de aumentarem a aceleração na medida em que mais processadores são adicionados é chamada de escalabilidade. A eficiência,  $E_p$ , de um programa paralelo é uma estimativa de quanta aceleração é obtida à medida que novos processadores são adicionados,

$$E_p = \frac{Sp}{np} , \quad (4-36)$$

onde os valores de  $E_p$  variam tipicamente entre 0 e 1.

Neste trabalho, a paralelização do programa computacional é fundamentada na decomposição do domínio principal em subdomínios menores. Cabe mencionar que esta estratégia é largamente utilizada em sistemas de memórias distribuídas, isto é, naqueles em que cada CPU possui uma área de memória própria. O padrão MPI-2 (*Message Passage Interface*) é empregado na arquitetura das trocas de mensagens, mediante as instruções de programação MPICH2 (Gropp et al., 1999a, 1999b, Chergui et al., 2006). A comunicação entre os subdomínios pode acontecer das seguintes formas,

- (a) Global: todos os processadores podem se comunicar entre si,
- (b) Dinâmica: as estruturas de mensagens a serem trocadas podem variar ao longo do tempo,
- (c) Sincronizada ou não sincronizada: na primeira, as operações de trocas de mensagens entre processadores são executadas de maneira coordenada entre as partes envolvidas, onde cada uma das partes não realiza sua operação sem a contrapartida da outra. Na segunda, um processador pode receber ou enviar informações sem que haja a cooperação do outro.

No âmbito da estratégia de decomposição de domínios, os procedimentos de paralelização empregados nos referenciais eulerianos e lagrangeanos possuem características próprias, as quais são discutidas separadamente nas duas seções a seguir.

#### 4.4.1. Referencial Euleriano

Cabe lembrar que a paralelização do programa no referencial euleriano foi realizada pelo grupo de transferência de calor e massa e dinâmica dos fluidos do Departamento de Engenharia Mecânica da Universidade Federal de Uberlândia. Referências completas sobre este trabalho podem ser encontradas em Campregher et al. (2004) e Campregher (2005).

O domínio original no referencial euleriano é particionado em subdomínios menores, de forma que a carga de trabalho pode ser dividida uniformemente entre os processadores, mediante a escolha adequada da dimensão de cada subdomínio. Todos os processadores executam as mesmas instruções do programa, porém, sobre a base de dados associada a cada subdomínio. As condições de contorno são fornecidas pelos processadores vizinhos mediante um processo de troca de mensagens. As topologias das interfaces de comunicação entre os subdomínios podem variar de unidimensionais a tridimensionais. Neste trabalho, a primeira opção é empregada, na qual as trocas de mensagens são realizadas em uma única direção, como ilustra a Figura 4.6. Cabe notar que a direção de particionamento adotada é a direção principal do escoamento.

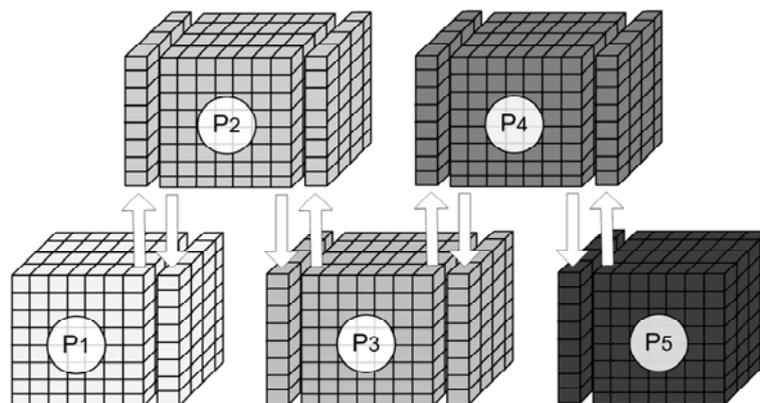


Figura 4.6 Ilustração das interfaces de comunicação entre cinco processadores vizinhos usando uma topologia unidimensional de divisão (Campregher, 2005).

Observa-se que é necessário um estêncil de apenas um volume de controle vizinho nas trocas de mensagens, visto que a discretização dos termos espaciais é de segunda ordem. Nas interfaces entre os subdomínios, a sobreposição de apenas um plano é realizada. Na Figura 4.7 é possível notar que as soluções armazenadas

nos volumes internos 2 e  $n - 1$  são transferidas para os volumes de controle  $n$  e 1, respectivamente, agindo como condições de contorno dos subdomínios  $\Omega_1$  e  $\Omega_2$ .

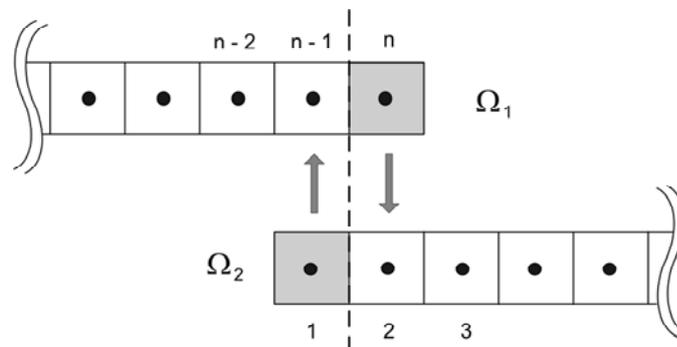


Figura 4.7 Ilustração das sobreposições necessárias para aproximações espaciais de segunda ordem. As condições de contorno são obtidas mediante a troca de mensagens entre os subdomínios  $\Omega_1$  e  $\Omega_2$  (Campregher, 2005).

Decomposições de domínios em topologias de dimensões maiores que a unitária fornecem, em geral, uma melhor escalabilidade do algoritmo. No entanto, o particionamento unidimensional foi adotado neste trabalho, tendo em vista as características geométricas do domínio computacional utilizado e por uma questão de simplicidade no acoplamento com o programa lagrangeano.

#### 4.4.2. Referencial Lagrangeano

A paralelização do programa no referencial lagrangeano é realizada com base na estratégia de decomposição de domínios, já empregada no referencial euleriano. Como mencionado, a topologia de divisão adotada é unidimensional, sendo que as partições são feitas ao longo da direção principal do escoamento. Desta forma, as partículas que evoluem no escoamento e atravessam as fronteiras de comunicação entre os subdomínios no espaço físico são enviadas para o processador vizinho. As partículas que excedem as fronteiras sólidas, impostas pelas paredes da geometria utilizada neste trabalho, mostrada na Figura 4.3, são refletidas para o interior dos subdomínios.

Nota-se que a escolha desta estratégia ocorre de maneira natural, uma vez que as posições das partículas no espaço devem ser associadas àquelas dos centros dos volumes de controle, quando da troca de informações entre partículas e malha,

característica do modelo híbrido. Além disto, esta estratégia permite um razoável equilíbrio na distribuição da parcela de trabalho em cada processador, pelo fato da malha euleriana ser fixa e visto que é esperado que o número médio de partículas se mantenha razoavelmente constante em cada processador durante a simulação.

Neste contexto, as principais diferenças nos procedimentos de paralelização entre os referenciais euleriano e lagrangeano estão relacionadas às estruturas de dados trocadas (Kaludercic, 2004). Enquanto no primeiro, as estruturas de dados trocadas são fixas, referentes aos planos de volumes de controle que se sobrepõem nas fronteiras dos subdomínios, no segundo, estas estruturas variam com o tempo. Ou seja, o número de partículas enviadas ou recebidas entre os processadores não é constante no decorrer da simulação. Esta característica exige a elaboração de uma complexa arquitetura de troca de mensagens a ser executada a cada passo de tempo.

Em geral, uma arquitetura de troca de mensagens pode ser definida a partir dos seguintes procedimentos (Rembold et al., 2008),

- (a) As partículas devem ser transportadas mediante as equações governantes de seus movimentos até que um passo de tempo seja percorrido,
- (b) Um algoritmo deve detectar as partículas que cruzam as fronteiras de comunicação entre os subdomínios e precisam ser transmitidas para os processadores vizinhos. A estrutura de dados destas partículas deve ser armazenada em áreas de memória, chamadas de zonas intermediárias (*send buffers*),
- (c) As zonas intermediárias devem ser enviadas aos processadores vizinhos,
- (d) Os processadores vizinhos devem receber a estrutura de dados das partículas até que as zonas intermediárias sejam totalmente esvaziadas.

Na prática, a maior dificuldade na implementação desta estratégia decorre do número variável de partículas a serem trocadas entre os processadores a cada passo de tempo. Isto implica que a estrutura de dados, ou seja, os vetores que armazenam as propriedades das partículas em cada processador, precisam ser redimensionados e reorganizados a cada troca de mensagens.

Neste trabalho, as propriedades das partículas, isto é, suas posições em coordenadas cartesianas,  $\mathbf{x}$ , e os valores do campo escalar,  $\phi$ , são armazenadas na estrutura de dados chamada de *partícula*. A cada elemento desta estrutura é associado um número inteiro, armazenado em vetores locais a cada processador,

chamados de *presença*, os quais assinalam 1 caso a partícula esteja presente no processador e 0 caso ela não esteja. Desta maneira, cada processador tem a informação do número de partículas que ele abriga e a posição destas na estrutura de dados *partícula*.

A Tabela 4.1 mostra a construção da estrutura de dados *partícula* e do vetor *presença*, em linguagem de programação FORTRAN 90. Nota-se que a partícula é caracterizada por sua posição no espaço, nas três direções cartesianas, *posix*, *posiy* e *posiz*, e pelo valor do escalar, *vprog*. A variável “partícula” consiste de um vetor que engloba esta estrutura de dados.

```

type :: part_struct
real(double) :: posix
real(double) :: posiy
real(double) :: posiz
real(double) :: vprog
end type part_struct

type(part_struct), allocatable, dimension(:) :: partícula

integer, allocatable, dimension(:) :: presença

```

Tabela 4.1 Construção da estrutura de dados *partícula* e declaração da variável *presença* em linguagem de programação FORTRAN 90.

Com estas definições, a arquitetura de troca de mensagens, desenvolvida no presente trabalho, pode ser exemplificada de maneira esquemática, como mostra a Figura 4.8. Em um número arbitrário de três processadores, *P1*, *P2* e *P3*, os quais abrigam subdomínios com dimensões idênticas e contendo o mesmo número de volumes de controle, dez partículas são distribuídas uniformemente no instante de tempo  $t = 0$  em cada subdomínio.

Ao iniciar a simulação, o campo do escoamento é resolvido e as partículas evoluem no espaço físico e do campo escalar de forma sincronizada, percorrendo um passo de tempo. O algoritmo detecta as partículas que cruzam as fronteiras dos subdomínios, em todos os processadores, e os seguintes procedimentos de envio são realizados,

- (a) Estas partículas são armazenadas nas zonas intermediárias. Os endereços de destino, isto é, o processador para o qual cada partícula deve ser enviada, também é armazenado. Além disto, é armazenado um número inteiro igual a 1, associado a cada partícula, o qual funciona como um

contador das partículas a serem enviadas. Note-se que as partículas a serem enviadas são armazenadas nas zonas intermediárias uma-a-uma, em um procedimento realizado mediante o emprego do comando *mpi\_send* (Chergui et al., 2006). Este comando caracteriza uma operação de envio bloqueante, ou seja, o programa pode prosseguir somente após a conclusão da formação de todas as zonas intermediárias, em todos os processadores,

- (b) As posições nos vetores de presença, correspondentes as posições das partículas enviadas, são zeradas.

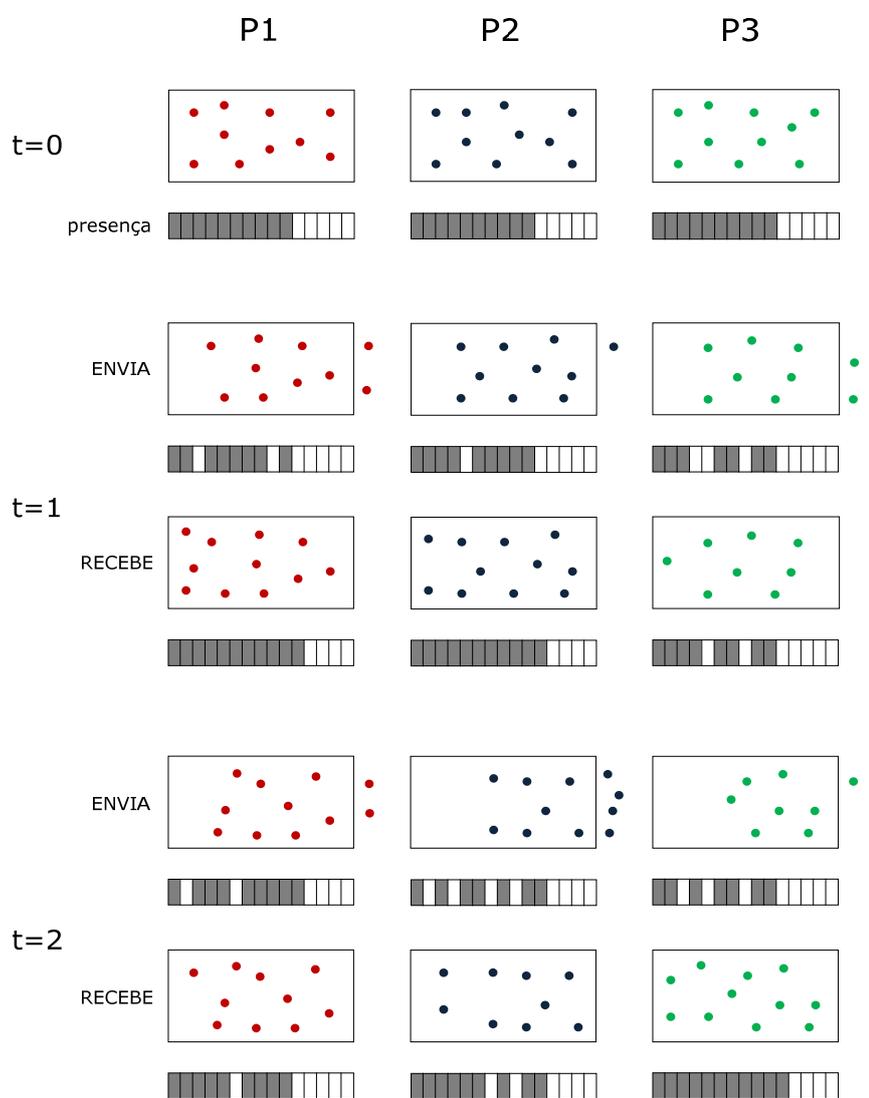


Figura 4.8 Ilustração esquemática da arquitetura de troca de mensagens do programa lagrangeano entre três processadores associados a subdomínios de dimensões idênticas para instantes de tempo  $t = 0, 1, 2$ .

A Figura 4.8 mostra que, para o instante de tempo  $t = 1$ , duas partículas são enviadas do processador  $P1$  para o  $P2$ , uma partícula é enviada de  $P2$  para  $P3$  e três partículas são enviadas de  $P3$  para  $P1$ . Notam-se as posições abertas nos vetores *presença* de cada processador, as quais são correspondentes as posições vazias nas estruturas de dados *partícula*, deixadas pelas partículas enviadas.

Uma vez que o processo de envio é concluído em todos os processadores, os procedimentos de recebimento são executados da seguinte forma,

- (a) Os números inteiros que funcionam como contadores, associados a cada partícula que foi enviada, são recebidos primeiramente. Desta forma, cada processador tem a informação exata sobre o número de partículas a serem recebidas,
- (b) Enquanto existem partículas a receber, o algoritmo percorre o vetor de presença de cada processador, do início para o fim. Se o elemento do vetor *presença* está preenchido com o número inteiro igual a 1, ou seja, a posição correspondente na estrutura de dados *partícula* está ocupada, o algoritmo pula para o próximo elemento e continua a percorrer o vetor. Se o elemento do vetor *presença* está preenchido com o número inteiro igual a 0, isto é, a posição correspondente na estrutura *partícula* está vazia, as propriedades de uma partícula são recebidas nesta posição, mediante o uso do comando *mpi\_recv*, o vetor de presença é preenchido e o algoritmo prossegue para o próximo elemento. Este procedimento é realizado até que todas as partículas sejam recebidas, em todos os processadores, e as zonas intermediárias sejam totalmente esvaziadas.

Na Figura 4.8, percebe-se que no instante de tempo  $t = 1$ , três partículas são recebidas pelo processador  $P1$ , duas partículas são recebidas por  $P2$  e uma partícula é recebida por  $P3$ . Nota-se que os espaços vazios nos vetores *presença*, correspondentes aos espaços vazios nas estruturas de dados *partícula*, os quais foram deixados pelas partículas enviadas, são preenchidos em cada processador.

É importante observar que as estruturas de dados *partícula* e os vetores *presença* são dimensionados de modo a disponibilizarem mais espaço do que aquele ocupado pelo número efetivamente criado de partículas. O *loop* de cálculo no algoritmo tem a dimensão da estrutura de dados *partícula*, no entanto, as instruções do programa são executadas somente no caso do vetor *presença* ser

igual a 1, isto é, caso exista uma partícula ocupando aquela posição na estrutura de dados.

No procedimento de recebimento, as posições vazias dos vetores, deixadas pelas partículas enviadas, são preenchidas primeiramente e, caso o número de partículas a receber seja maior do que o número de espaços abertos pelas partículas enviadas em cada processador, os espaços de sobra são ocupados na seqüência. Cabe mencionar, também, que caso o número de partículas a receber seja menor do que o número de partículas enviadas, em cada processador, existe uma descontiguidade no vetor *presença* e na estrutura de dados *partícula*, a qual é caracterizada por espaços vazios entre os elementos dos vetores. Na Figura 4.8, tal descontiguidade pode ser percebida no vetor *presença* do processador *P3*, no instante de tempo  $t = 1$ , e nos vetores *presença* dos processadores *P1* e *P2*, no instante de tempo  $t = 2$ .

Convém ressaltar que os procedimentos descritos acima são necessários devido ao número flutuante de partículas presentes em cada processador ao longo da simulação. Nota-se que no instante de tempo  $t = 0$ , todos os processadores contém dez partículas. No instante  $t = 1$ , os processadores *P1*, *P2* e *P3* contém onze, onze e oito partículas, e no instante  $t = 2$ , tais processadores contém dez, onze e nove partículas, respectivamente. Este exemplo mostra que, eventualmente, no decorrer da simulação, o desequilíbrio entre o número de partículas presentes em cada processador pode ser significativo, o que justifica os espaços extras nas estruturas de dados *partícula* e nos vetores *presença*. No entanto, é desejável que o número médio de partículas seja aproximadamente igual entre os processadores ao longo do tempo. Desta maneira, um razoável equilíbrio na distribuição da carga de trabalho média de cada processador pode ser obtido.