

SIMULAÇÃO DO ESCOAMENTO DE FLUIDOS INCOMPRESSÍVEIS COM ALGORITMOS DE GÁS EM REDE E COMPUTAÇÃO PARALELA

Ariane Jevaux Souza

Instituto Politécnico – IPRJ, Universidade do Estado do Rio de Janeiro – UERJ, CP 97.282, CEP 28601-970, Nova Friburgo – RJ, Brasil.
jariane@iprj.uerj.br

Antônio José da Silva Neto

Instituto Politécnico – IPRJ, Universidade do Estado do Rio de Janeiro – UERJ, CP 97.282, CEP 28601-970, Nova Friburgo – RJ, Brasil.
ajsneto@iprj.uerj.br

João Flávio Vieira de Vasconcellos

Instituto Politécnico – IPRJ, Universidade do Estado do Rio de Janeiro – UERJ, CP 97.282, CEP 28601-970, Nova Friburgo – RJ, Brasil.
jflavio@iprj.uerj.br

Luiz Orlando Emerick dos Santos

Universidade Federal de Santa Catarina – UFSC, Florianópolis – SC, Brasil
emerick@lmpt.ufsc.br

Resumo: O algoritmo de gás em rede é uma técnica usada em simulações numéricas de escoamento de fluidos. Esta técnica não se baseia em equações diferenciais e tem sido usada em simulação de fluxos complexos, como por exemplo em meios porosos, sem a necessidade de se resolver equações diferenciais para o fenômeno de interesse.

O algoritmo do gás em rede é simples de ser implementado computacionalmente, porém o esforço do cálculo computacional é elevado. Por esta razão nesta dissertação desenvolvemos uma implementação paralela para que problemas mais complexos sejam estudados.

Palavras chave: Escoamento de Fluido incompressível, algoritmos de gás em rede, computação paralela.

1. Introdução

Nos últimos 20 anos, os modelos de gás em rede tem sido utilizado na simulação de escoamentos de fluidos. Os modelos Lattice Gas Cellular Automata e Lattice Boltzmann são métodos numéricos, promissores para soluções de equações diferenciais parciais, que simulam as equações de Navier-Stokes descrevendo o escoamento de fluidos complexos.

O desenvolvimento do Lattice Gas Cellular Automata (LGCA) começou em 1986 com os artigos de Frisch, Hasslacher e Pomeu. Alguns anos depois o modelo Lattice Boltzmann (LBM) surgiu como um aprimoramento do LGCA que conduz a modelos microscópicos de acordo com a equação de Navier-Stokes.

Os modelos de gás em rede são autômatos celulares utilizados na simulação de dinâmica de fluidos. Nestes modelos um fluido é um conjunto de partículas que habita um espaço discreto (os sítios de uma rede), possuindo um conjunto finito de velocidades, saltando de um sítio a outro da rede, em intervalos de tempo discretos, passos de tempo, e colidindo entre si.

Os autômatos celulares são sistemas dinâmicos discretos que são descritos segundo as equações diferenciais parciais, as quais têm a capacidade de descrever sistemas dinâmicos. O significado de “discreto” é que o espaço, o tempo e as propriedades do autômato, tem somente um número finito de estados. A idéia principal não é tentar descrever um sistema complexo usando equações complexas, mas simular através das interações das células, segundo suas regras.

Assim, a utilização dos modelos de gás em rede nas simulações de dinâmica de fluidos é a tentativa de se reproduzir às características, na dinâmica microscópica, responsáveis pelo comportamento macroscópico que pode ser descrito pelas equações de Navier-Stokes.

Para a computação dos modelos de gás em rede na simulação de sistemas físicos primeiramente deve-se criar um sistema análogo ao que se deseja simular, na qual as variáveis são discretas e só podem assumir um número finito de valores. Este sistema, entretanto, deve possuir as características do sistema físico para ser utilizado nas simulações, e a partir do seu comportamento deduz-se o comportamento do sistema real.

Todos os meses surgem novos artigos com investigações destes modelos em fenômenos conhecidos e novas aplicações em diferentes campos.

2. Autômato Celular

Os modelos de gás em rede caracterizam-se como parte de uma classe mais ampla de modelos denominada autômato celular. Estes são idealizações matemáticas de sistemas físicos nos quais todas as grandezas físicas podem assumir somente um número finito de valores. As características e os elementos básicos do autômato celular são os fundamentos dos modelos de gás em rede.

A teoria do Autômato Celular (CA) como sistema de auto-reprodução foi concebido e primeiramente desenvolvido por John von Neumann durante 1950 e foi apresentado em von Neumann, 1966.

Na década de 80 do século passado, Stephen Wolfram reavivou o interesse em CA com uma série de publicações (Wolfram, 1983, 1984, 1986). Em 1986 Frisch, Hasslacher e Pomeau demonstraram que o CA sobre uma rede com simetria hexagonal conduz à equação de Navier Stokes no limite macroscópico (Dieter, 2000).

O modelo do autômato celular foi caracterizado (Wolfram, 1984) como arranjos de células simples e de mesma espécie na qual cada célula tem um número finito de estados discretos, e os estados são alterados simultaneamente no espaço de tempo discreto. As regras de alteração são determinísticas, uniformes no espaço e tempo, e dependem somente da população local da célula.

A característica mais importante é que eles provêm simples modelos para complexos sistemas. Ele exemplifica o fato de que um comportamento coletivo pode emergir da soma de simples componentes. Este fato faz o autômato celular uma aproximação muito interessante para modelar sistemas físicos.

Um exemplo interessante é o modelo FHP, proposto por Frisch, Hasslacher e Pomeau em 1986 (Frisch et al., 1986) o qual pode ser visto como uma dinâmica molecular completamente discreta que satisfaz a equação de Navier Stokes para fluidos incompressíveis (como será visto na seção 2.2).

Contudo, eles tiveram muito êxito para modelar situações complexas, fluxos em meios porosos, fluxos imiscíveis e instáveis, propagação de gotas de líquido, fluxos granulares, microemulsão, erosão e problemas de transporte são alguns exemplos relativos a dinâmica dos fluidos.

Outros estados físicos como, por exemplo, processos de reação-difusão, fenômeno de crescimento nucleação-agregação, processos de tráfico, são muito aplicados ao autômato celular.

Os elementos básicos do celular automata são:

- *A célula:* É um tipo de memória que guarda os estados. Em um caso mais simples cada célula pode ter dois estados 0 ou 1. Em sistemas mais complexos as células podem ter outros estados diferentes do binário.
- *A rede:* As células são arranjadas em uma rede de espaços. A rede mais simples é uma rede unidimensional que significa que todas as células são arranjadas em uma linha. Os CA's mais comuns são os bidimensionais e tridimensionais.
- *A população:* Até agora, as células arranjadas em uma rede representam estados estáticos. Para introduzir a dinâmica do sistema, nós temos que adicionar regras. O trabalho destas regras é definir o estado das células para o próximo passo de tempo. No automata celular a regra que define o estado da célula depende da população onde se encontra esta célula.

Podemos encontrar duas importantes populações:

- População von Neumann: quatro células, acima, abaixo, direitas, esquerdas são chamadas de população von Neumann desta célula. Somente as células que distam 1 ao longo das coordenadas são consideradas.
- População Moore: a célula é conectada a células que distam, no máximo, 1 em cada direção, incluindo a diagonal da célula.

Desde sua descoberta o CA têm se desenvolvido e utilizado em diferentes campos de aplicações.

As aplicações do CA em sistemas físicos foram o primeiro caso de sucesso. Em particular o caso denominado HPP, um marco importante na evolução dos modelos de CA. A dinâmica HPP foi inicialmente planejada como um modelo teórico para estudo das propriedades fundamentais de um gás com interação entre as partículas. Desde que conhecidos os fluxos de um sistema real de partículas completamente discreto e simplificado, a dinâmica molecular deve simular o fenômeno da dinâmica de fluidos adequadamente. Este é o princípio básico de aplicações em mecânica dos fluidos nos modelos do CA para simular fenômenos físicos reais.

A área biológica foi uma das primeiras disciplinas envolvidas na aplicação do CA. A principal proposta de von Neumann era criar um modelo computacional que descrevesse e simulasse a auto-reprodução no sentido biológico. Isto foi possível devido às pesquisas e aplicações em geral nos modelos de vida artificial e sistemas desenvolvidos para o estudo de propriedades dinâmicas de fenômenos biológicos.

3. Lattice Gas Cellular Automata

O método Lattice Gas Cellular Automata (LGCA) é um método relativamente novo desenvolvido para cálculos hidrodinâmicos. Designa uma classe de modelos cuja característica principal é a presença de partículas distribuídas sobre um espaço discreto, os nodos de uma rede regular, redes regulares de Bravais. Definições e propriedades de redes de Bravais podem ser encontradas em (Rothman e Zaleski, 1997), maiores especificações são encontradas em (Coxeter

,1977). As partículas, em cada sítio, interagem e se propagam em passos de tempo discretos do processo.

O primeiro LGCA (*Lattice Gas Cellular Automata*), espécie de autômato celular para simulação de escoamentos de fluidos e outros problemas físicos, foi proposto in 1973 por Hardy, Pomeu e Pazzis (Hardy et al. 1973, 1976) o HPP. Sua denominação é derivado das iniciais dos autores. Atualmente o HPP é principalmente de interesse histórico porque não conduz as equações de Navier-Stokes no limite macroscópico.

O HPP é um modelo de LGCA em duas dimensões sobre uma rede quadrada. Este modelo conserva a massa e o momento do fluxo mas não trata da simetria da rede. Sendo assim este modelo não obedece as equações hidrodinâmicas, equações de Navier-Stokes, no limite macroscópico. Por isso o HPP é de interesse apenas histórico.

Em 1986, Frisch, Hasslacher e Pomeau mostraram que o LGCA sobre uma rede com simetria hexagonal obedecia às equações de Navier-Stokes para fluidos incompressíveis no limite macroscópico. Este modelo é chamado de FHP de acordo com as iniciais de seus autores. Esta descoberta da simetria foi o início para um rápido desenvolvimento do método LGCA. O modelo FHP é representado na Fig. 1.

Para cada uma das seis direções de um sítio é permitida a presença de, no máximo, uma partícula, representada na Figura 1 através de uma seta. Outro ponto importante é possível presença de partículas em repouso nos sítios, representadas na figura por círculos. De acordo com o modelo pode-se ter ou não partículas em repouso, ou pode-se permitir a presença de mais de uma partícula em repouso por sítio.

O motivo de se limitar o número de partículas nos sítios é a utilização de variáveis booleanas para representar a presença ou ausência de uma partícula, e com isso reduzir a um mínimo o consumo de memória e tempo computacional nas simulações (Santos, 2000).

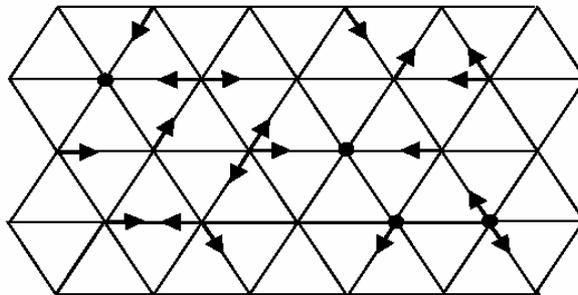


Figura 1 – Modelo FHP. As setas representam as partículas em movimento e os círculos representam as partículas em repouso.

As evoluções dinâmicas do modelo são dadas em 2 etapas. Na primeira etapa, chamada de colisão, as partículas são reordenadas segundo as regras que simulam as colisões características de cada modelo, no qual o número de partículas e a quantidade de movimento são conservados. Após esta etapa segue-se a propagação onde as partículas saltam de um sítio a outro na direção da sua velocidade, ou seja, se propagam proporcionando o desenvolvimento do modelo.

Devido ao trabalho de (Frisch et al., 1986,1987) que demonstraram que a dinâmica dos modelos de gás em rede obedece macroscopicamente às equações de Navier-Stokes em seu limite para escoamentos incompressíveis, este tem sido utilizado para simular problemas de escoamento de fluidos. Desde então, algumas atualizações têm sido feitas para simular fluidos multifásicos (Rothmann e Keller, 1988, Chen et al., 1991), transição de fases (Appert et al., 1990), e o desenvolvimento de um outro método para simulação de dinâmica de fluidos: o método Lattice Boltzmann (McNamara et al., 1988, Higera et al., 1989). O método Lattice Boltzmann é descrito a seguir.

4. Lattice Boltzmann

O modelo Lattice Boltzmann (LBM), é um modelo numérico para simulações de dinâmica de fluidos foi introduzido por McNamara e Zanetti em 1988.

Este modelo originou-se das equações dinâmicas microscópicas do Lattice Gas Automata (LGCA) (Higera et al., 1989, d’Humières et al., 1986). Portanto é baseado em uma rede regular discreta, mantendo as etapas de colisão e propagação onde, ao invés de variáveis booleanas, trabalha com variáveis reais, as quais são interpretadas como a probabilidade de encontrar uma partícula na direção de um dado sítio da rede num dado tempo.

As simulações de Lattice Boltzmann são uma alternativa para a dinâmica de fluidos clássica, pois este simula o comportamento da função distribuição de uma partícula, ao invés de resolver as equações hidrodinâmicas para conservação do campo. As equações originais de Boltzmann descrevem o comportamento da função distribuição de uma partícula $f(r, v, t)$, onde f representa a probabilidade de encontrar uma partícula de fluido no ponto r no tempo t , movendo-se com velocidade v . Se esta função é conhecida, então os valores locais de densidade, momento e temperatura podem ser encontrados avaliando os momentos de f (Bruce e David, 2000).

O LGCA é basicamente definido pela rede e regras de colisão, enquanto um LBM é composto de três principais

componentes:

- A rede;
- A distribuição de equilíbrio;
- A equação dinâmica.

A evolução dinâmica do LBM é caracterizado pelas etapas de colisão e propagação do modelo.

Para o estudo do LBM em três dimensões, que é o modelo que utilizamos para o estudo neste trabalho, D3Q19, onde D3 indica o problema tridimensional e Q19 o número de direções para a propagação de cada ponto do sítio que pode ser visto na Fig. 2.

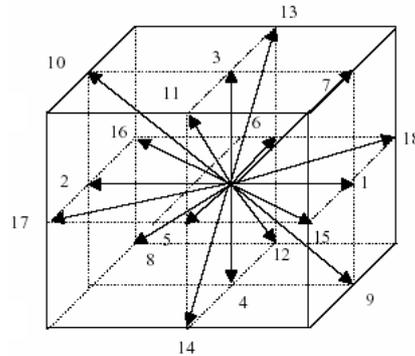


Figura 2 – Modelo D3Q19 velocidades.

Como neste trabalho é feita uma implementação em computação paralela para o método LBM, descrevemos o algoritmo utilizado na implementação paralela do método.

5. Implementação

Na implementação computacional do LBM, o sistema que se deseja simular é obtido através de um arquivo no qual contém a geometria do sistema em estudo juntamente com suas variáveis discretas que indicam onde temos o fluxo do escoamento e as condições de contorno do problema. Assim temos as características do sistema real que queremos simular.

O arquivo de geometria do meio contém o tamanho do meio (n_x , n_y , n_z) e, uma seqüência de zeros e uns, indicando sólido ou vazio, respectivamente. Ao ser lido todo o arquivo temos o número de pontos a serem preenchidos de fluido, ou seja, a quantidade de partículas que contém a malha do fluido.

Considerando o paradigma de memória distribuída, a paralelização do modelo LBM consiste em dividir a rede em pedaços menores. Estes pedaços são igualmente distribuídos aos processadores para equilibrar a carga.

Como vimos à dinâmica do modelo de LBM só requer conhecimento do vizinho mais próximo. Por conseguinte, define-se o local regular como um local com todos seus vizinhos no mesmo processador, e um local de limite como um local com pelo menos um de seus vizinhos em um outro processador. Isto significa que, para o local regular, a dinâmica não é alterada pela decomposição de domínio. Este não é o caso para os locais que formam o limite. O conhecimento completo da rede para estes locais implica em um pouco de comunicação.

Portanto na implementação paralela, o algoritmo tem passos adicionais. A divisão da geometria da rede os processadores e o envio das informações do local limite.

Assim temos o algoritmo paralelo com os seguintes passos:

- (i) Ler arquivo de geometria do meio;
- (ii) Dividir a geometria da rede entre os processadores;**
- (iii) Inicializar as velocidades da rede;
- (iv) Calcular a distribuição de equilíbrio da rede;
- (v) Enquanto quantidade de passos de tempo desejada
 - (vi) Enviar as informações das fronteiras;**
 - (vii) Calcular massa específica e a velocidade do fluido em cada sítio;
 - (viii) Baseado no valor da densidade, calcular o movimento das partículas em cada sítio;
 - (ix) Obter a nova distribuição de equilíbrio e completar a etapa de colisão;
 - (x) Se passo igual a intervalo
 - (xi) Gravar arquivos de saída
 - (xii) Senão
 - (xiii) Efetuar etapa de propagação de cada sítio da rede
 - (xiv) Voltar ao passo (v do algoritmo).

Para o desenvolvimento computacional paralelo, temos que dividir nosso domínio entre os processadores que

participam da computação. Isto está representado no algoritmo acima pelo passo (xvi).

O processador 1 envia informações de seu local limite para o processador 2 que, respectivamente, recebe as informações, armazena na posição correspondente, e envia ao processador 1 as informações do seu local limite. Desta forma, as informações sobre os vizinhos são atualizadas pela fase de comunicação executada em cada etapa, e cada sub-matriz pode ser considerada completamente independente.

A implementação do algoritmo descrito acima foi testada em um sistemas multicomputador formado por 8 microcomputadores dedicados da Universidade do Estado do Rio de Janeiro – Instituto Politécnico.

Os microcomputadores que compõem este cluster, Metatutor, possuem processador AMD 750 Mhz, 256MB de memória, espaço em disco de 20G. Biblioteca MPI versão 7.02. Sua rede de conexão está composta de uma chave (switch) e algumas ligações de Fast Ethernet.

Na implementação paralela, como previamente mencionado, um domínio tem que ser decomposto em subdomínios. Escolher uma decomposição ou outra depende essencialmente da relação computacional durante o tempo de comunicação. Realmente, uma decomposição em faixas muito pequenas implicaria em muitas comunicações com respeito à computação. Por outro lado, para o mesmo volume de comunicação, uma decomposição em caixas em lugar de faixas soma algum tempo de latência¹ porque há 4 vizinhos e por conseguinte são necessárias 4 comunicações em vez de 2. Conseqüentemente, dependendo da aplicação e sua geometria, tem que ajustar o método da decomposição. Note que uma decomposição em faixas ou caixas é relacionada diretamente à implementação por uma ordem multidimensional. Conseqüentemente, um gasto extra (overhead), para a quantidade inicial de trabalho é somado. É composto de operações extras, como as comunicações. Por conseguinte, o ganho (speedup) não será igual ao número de processadores.

6. Resultados

Os resultados obtidos neste trabalho para as simulações usando o método LBM foram gerados para o escoamento entre placas planas e paralelas. Este foi escolhido para testar nossa implementação em virtude de ser um problema simples e de conhecida e direta solução analítica para a verificação dos resultados obtidos.

Da solução da equação de Navier-Stokes obtêm-se um perfil de velocidade parabólico para fluxos entre placas planas e paralelas em escoamento laminar.

6.1 Validação

Para a validação do resultado encontrado utilizamos um dos casos simulados para a comparação dos resultados analíticos e computacionais.

A solução analítica do problema das placas planas e paralelas na Região Plenamente Desenvolvida (RPD) do escoamento é dado por (Fox McDonalds,1995).

Os resultados obtidos foram comparados para o escoamento em um canal formado por placas planas e paralelas.

Uma comparação entre o resultado analítico e o resultado numérico para $\frac{u}{u_{max}}$ é apresentada na Fig. 3.

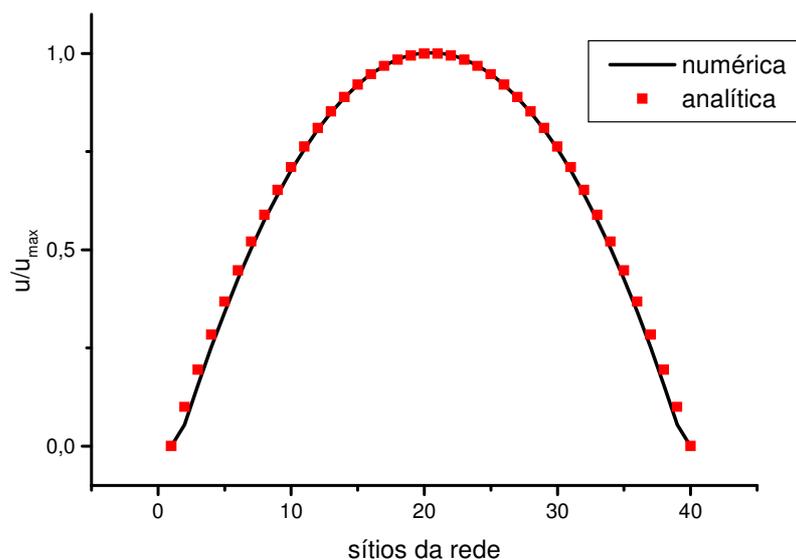


Figura 3 – Resultado analítico e resultado numérico.

¹ Tempo perdido com a comunicação entre os diversos dispositivos.

Após a validação aqui apresentada tem-se então confiança no algoritmo implementado e parte-se para a análise de situações mais complexas e na implementação em processamento paralelo.

Alguns casos de escoamento de fluidos laminares entre placas planas e paralelas foram testados, mas apresentaremos, para resumir, um deles.

6.2 Escoamento em um canal formado por duas placas planas e paralelas

Nesta seção descreveremos o escoamento entre duas placas planas e paralelas onde a diferença de pressão exercida sobre o fluido ao longo do escoamento é tomada como idêntica àquela utilizada por (Santos, 2000). O número de Reynolds deste escoamento tem valor igual a 79,33.

O comportamento computacional do escoamento do fluido no fim do escoamento (ao longo do domínio do cálculo), é apresentado na Fig. 4.

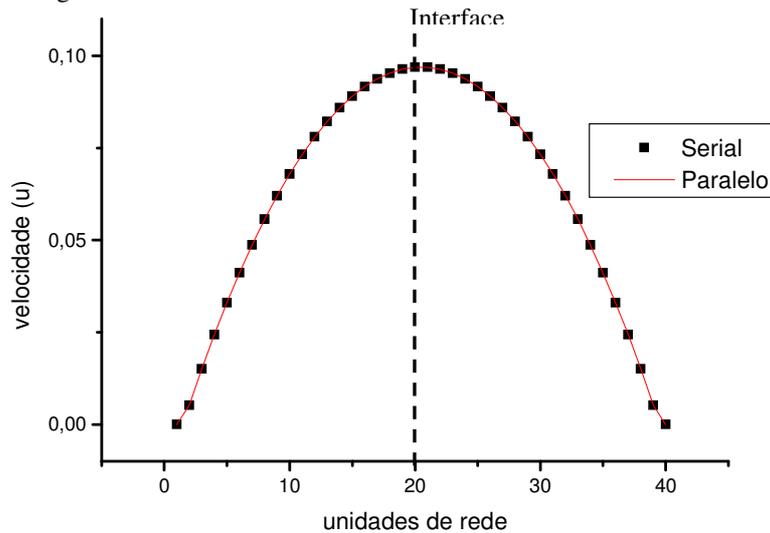


Figura 4 – Resultados computacionais para o escoamento em um canal formado por duas placas planas e paralelas no final do escoamento.

6.3 Escoamento em dois canais formado por três placas planas e paralelas

Nesta seção descreveremos o escoamento formado por três paredes fixas planas e paralelas, formando, portanto, dois canais. A pressão exercida sobre o fluido é idêntica àquela usada na seção 6.2. O número de Reynolds deste escoamento tem valor igual a 20,95.

O comportamento computacional do escoamento do fluido no fim do escoamento (ao longo do domínio do cálculo), é apresentado na Fig. 5.

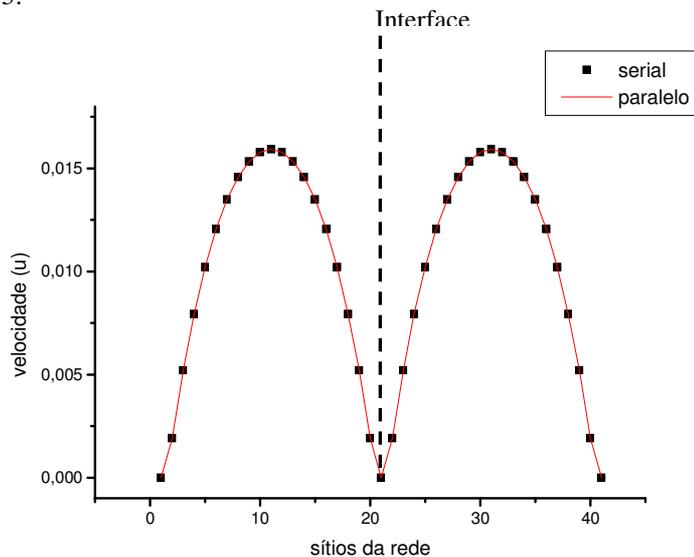


Figura 5 – Resultados computacionais para o escoamento em dois canais

formados por três placas planas e paralelas no final do escoamento.

6.4 Escoamento em três canais formado por quatro placas planas e paralelas

Nesta seção descreveremos o escoamento formado por quatro paredes fixas planas e paralelas, formando, portanto três canais. A pressão exercida sobre o fluido é idêntica àquela usada na seção 6.2. O número de Reynolds deste escoamento tem valor igual a 8,10.

O comportamento computacional do escoamento do fluido no fim do escoamento (ao longo do domínio do cálculo), é apresentado na Fig. 6.

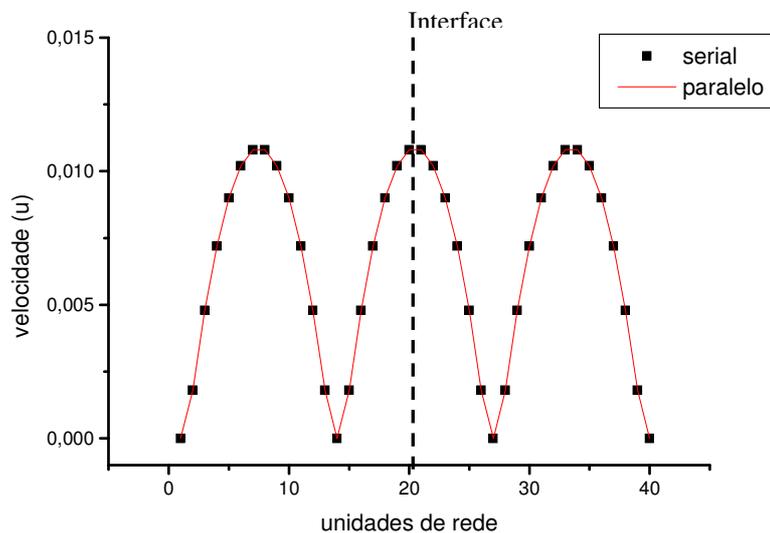


Figura 6 – Resultados computacionais para o escoamento em três canais formados por quatro placas planas e paralelas no final do escoamento.

A medida de desempenho computacional no processamento paralelo, ainda que a velocidade de computação do problema e as exigências de memória continuem sendo os fatores essenciais também na computação paralela, outros fatores afetam significativamente o desempenho neste tipo de computação: gastos de tempo com a comunicação e perdas de tempo por sincronização. Estes fatores podem influenciar consideravelmente a eficiência de um algoritmo paralelo, resultando em tempos de processamento longe do esperado, ainda mais quando se aumenta o número de processadores e as dimensões do problema.

Com isso as principais medidas, na computação paralela, que buscam avaliar seu desempenho são: ganho (speedup) e a eficiência. O ganho é um fator que compara o tempo total consumido por um algoritmo quando executado em uma máquina serial em relação ao tempo requerido pelo mesmo algoritmo quando executado em uma máquina paralela. A eficiência dá uma indicação da porcentagem do tempo total realmente dispendido na aplicação por cada um dos processadores.

Os resultados encontrados para o método Lattice Boltzmann no ambiente paralelo onde o domínio foi decomposto em 2 faixas, utilizando 2 processadores, apresentam speedup (ganho) e eficiência como mostra a tabela a seguir.

Tabela 1 – Resumo dos tempos computacionais dos casos testes, o ganho (speedup) e a eficiência nos ambientes serial e paralelo.

| Simulações | Processador(es) | Tempo médio ² (h:min:s) | Speedup | Eficiência | Desvio padrão |
|--------------|-----------------|---------------------------------------|---------|------------|---------------|
| (6.2) 2 PPPs | 1 | 00:10:18 | - | - | 0.001901 |
| (6.2) 2 PPPs | 2 | 00:06:04 | 1,68 | 0,84 | 0.009534 |
| (6.3) 3 PPPs | 1 | 00:09:33 | - | - | 0.000390 |
| (6.3) 3 PPPs | 2 | 00:05:51 | 1,69 | 0,85 | 0.003879 |
| (6.4) 4 PPPs | 1 | 00:10:18 | - | - | 0.000115 |
| (6.4) 4 PPPs | 2 | 00:06:04 | 1,68 | 0,84 | 0.009470 |

² Tempo médio em 3 rodadas do programa.

Podemos observar que ao dividirmos nosso domínio entre dois processadores deveríamos ter uma eficiência de 100%, mas isto se o nosso problema fosse completamente paralelizável, o que significa que nenhuma comunicação entre os processadores seria feita. Contudo no nosso caso, existe troca de informações, comunicação entre os processadores participantes do processo, assim os resultados encontrados para a eficiência do processamento paralelo no método Lattice Boltzmann obtêm um ganho de tempo computacional considerável.

7. Conclusão

Neste trabalho foram apresentados os modelos de gás em rede para simulações de escoamentos de fluidos incompressíveis a baixo número de Reynolds. Assim como seu processo de evolução na rede de partículas.

Também foram apresentadas as formas de implementação do modelo utilizadas no processamento seriais e paralelas, bem como a representação gráfica das geometrias (casos testes) que serviram de estudo.

Nas aplicações onde se deseja os campos de velocidade e o comportamento do escoamento, a aplicação do modelo Lattice Boltzmann mostrou-se adequada.

As características utilizadas nos modelos de gás em rede neste trabalho são características comuns a estudos de casos onde podemos comprovar a eficácia do método, com isso pode-se pensar na utilização do modelo como um laboratório onde se estuda a característica essencial de fenômenos simples e complexos.

Os resultados das simulações envolvendo a computação serial e paralela foram apresentadas, e é apresentada, também, uma comparação entre estes resultados. Os resultados obtidos com o paralelismo do método mostraram que podemos utilizá-lo de forma eficaz para estas simulações com ganho de tempo computacional considerável.

Apesar dos bons resultados obtidos com a utilização da computação paralela, ainda há o que se fazer, pois não obtivemos resultados com um número maior de processadores, e interessante seria descobrir qual o limite do número de processadores onde encontramos ganho de tempo computacional.

8. Referências

- Andrews G.R., 1991. "Concurrent Programming – Principles and Practice", Menlo Park, Addison-Wesley.
- Andrews G.R., 1991. "Foundations of multithreaded parallel and distributed programming", Menlo Park, Addison-Wesley.
- Appert, C. e Zaleski, 1990. "A lattice gas liquid-gas transition". *Physical Review Letters* 64, 1-4.
- Bandini S., G. Mauri, R. Serra, 2001. "Cellular Automata: From a theoretical parallel computational model to its application to complex systems", *Parallel Computing*, 27.
- Bueno A. D. , 2002. "Introdução ao Processamento Paralelo e ao Uso de Clusters de Workstations em Sistemas GNU/LINUX", Laboratório de Meios Porosos e Propriedades Termofísicas – LMPT, UFSC.
- Bruce J. P. e David R. R., 2000. "Lattice Boltzmann Algorithm for Simulating Thermal Flow Incompressible Fluids". *Journal of Computational Physics* 161, 1-20.
- Cercignani, C., 1990. "Mathematical methods in kinetic theory – 2nd ed". Plenum Press.
- Chapman, S.; Cowling, T. G. , 1952. "The Mathematical Theory of Non-uniform Gases". Cambridge University Press.
- Chen S., Doolen G. D., Eggert, K., Grunau, D. e Loh, E. Y. (1991). "Local lattice-gas model for immiscible fluids". *Physical Review A* 43, 7053-7056.
- Chopard Bastien, Alexandre Masselot, 1999. "Cellular Automata and lattice Boltzmann methods: a new approach to computational fluid dynamics and particle transport", Computer Science Department, 24, Switzerland.
- Coxeter, H. S. M. , 1977. "Regular Polytypes". Dover.
- D. Talia, P. Slood, 1999. "Cellular Automata: Promise and Prospect in Computational Science", FGCS, 16.
- Dazhi Yu, Renwei Mei, Li-Shi Luo, Wei Shyy, 2003. "Viscous flow computations with the method of Lattice Boltzmann equation". *Progress in Aerospace Sciences* 39, p329-367.
- DeWITT, D. J., GRAY, J., 1992. "Parallel Database Systems: The Future of High Performance Database Systems". *Communications of the ACM*, v. 35, n. 6.
- Dellar P.J. , 2003. "Incompressible limits of lattice Boltzmann equations using multiple relaxation times". *Journal de Computation Physics* 190, p351-370.
- Desplat J. C, I. Pagonabarraga, P. Blandon , 2002. "Ludwig: A parallel Lattice-Boltzmann code for complex fluids".
- Dieter A. Wolf-Gladrow, 2000. "Lattice-Gas Cellular Automata and Lattice Boltzmann Models, An Introduction", *Lecture Notes in Mathematics*.
- d'Humières D., Lallemand P., e Frisch U., 1986. "Lattice gas models for 3D hydrodynamics *Europhys*". *Lett.* 2, 291-297.
- Dupuis, A. , 2002. "From a lattice Boltzmann model to a parallel and reusable implementation of a virtual river". Tese de doutorado, Cologny, Genève.
- Fox McDonald, 1995. "Introdução à Mecânica dos Fluidos", 4ª edição, Guanabara Koogan.
- Frisch, U., d'Humières, D., Hasslacher, B., Lallemand, P., Pomeau, Y. e Rivet, J., 1987. "Lattice Gas Hydrodynamics in Two and Three Dimensions", *Complex Systems* 1, 649- 707.
- Frisch, U., Hasslacher B. e Pomeau Y., 1986. "Lattice-Gas Automata for the Navier-Stokes Equation", *Physical Review*

Letters, v 56, 1505-1508.

- Guinot V., 2002. "Modelling using stochastic, finite state cellular automata: rule inference from continuum models". *Applied Mathematical Modelling* 26, p 701-714.
- Hardy, J.; de Pazzis O. e Pomeau Y., 1976. "Molecular dynamics of a classical lattice gas: Transport properties and time correlation functions". *Phys. Rev. A*, v.13, p.1949-1961.
- Hardy, J., Pomeau, Y. e de Pazzis O., 1973. "Time Evolution of a Two Dimensional Model System. I. Invariant States and Time Correlation Functions", *J. Math. Phys.* 14, 1746- 1759.
- Higera, F. e Jimenez, J., 1989. "Boltzmann approach to lattice gas simulations". *Europhys. Lett.* 9, 663-668.
- Hercker C., D. Roytenberg, J. R. Sack, Z. Wang, 1999. "System development for parallel cellular automata and its applications", *FGCS*, 16.
- Waterman M. S., 1984. "Some applications of information theory to cellular automata", *Physica*, 10D, Amsterdam, p 45-51.
- Wolfram, S., 1983. "Statistical Mechanics of Cellular Automata". *Reviews of Modern Physics*, v.55, p.601-644.
- Wolfram, S., 1984. "Cellular automata as models of complexity". *Nature*, v.311, p.419.
- Wolfram, S., 1986. "Cellular Automaton Fluids: Basic Theory". *J. Stat. Phys.*, v.45, p.471-526.
- Lallemand Pierre, Li-Shi Luo, 2003. "Lattice Boltzmann method for moving boundaries". *Journal de Computacional Physics* 184, p 406-421.
- Lind D. A., 1984. "Applications of ergodic theory and sofic systems to cellular automata", *Physica D* 10, 36-44.
- Lindgren K., Nordhal M., 1988. "Complexity measures and cellular automata", *Complex Systems* 2 (4), 409-440.
- Love P.J., M. Nekovee, P.V. Coveney, J. Chin, N. González-Segredo, J.M.R. Martin, 2003. "Simulations of amphiphilic fluids using mesoscale lattice-Boltzmann and lattice-gas methods". *Computer Physics Communication* 153, p 340-358.
- McNamara G. G. e Zanetti G., 1988. "Use of the Boltzmann Equation to Simulate Lattice-Gas Automata". *Physical Review Letters* 61, 2332-2335.
- Meyer J., 1994. "Message Passing Interface for Microsoft Windows 3.1", Tese de Mestrado, University of Nebraska.
- Message Passing Interface Forum. MPI: "A Message-passing Interface Standard." Technical Report CS-94-230, Computer Science Dept. University of Tennessee, Knoxville, TN, 1994.
- Milnor J., 1988. "On the entropy geometry of cellular automata", *Complex Systems* 2 (3), 257-385.
- Peng Y., Shu C., Chew Y.T., 2003. "A 3D incompressible thermal lattice Boltzmann model and application to simulate natural convection in a cubic cavity". *Journal of Computacional Physics* 193, 260-274.
- Philippi, P. C., 1998. "Modelos de gás em rede I". Notas de aula do curso oferecido pelo Programa de Pós-graduação em Engenharia Mecânica - Universidade de Santa Catarina, Florianópolis.
- Qian, Y. H.; d'Humières D. e Lallemand P. "Lattice BGK Models for Navier-Stokes Equation *Europhys.* Lett., v.17, p.479-484, 1992.
- Ricarte, I.L. Marques, 1999. "Organização de Computadores", Tese de Doutorado, Unicamp, São Paulo.
- Rothmann, D. H. e Keller, J. M., 1988. "Immiscible Cellular-Automaton Fluids", *J.Stat. Phys.* 52, 1119-1127.
- Rothman, D. H. e Zaleski, S., 1997. "Lattice-gas cellular automata: simple models of complex hydrodynamics". Cambridge University Press.
- Santos, L.O.E., 2000. "Desenvolvimento de modelos de Gás em Rede para escoamentos Monofásicos e Bifásicos". Tese de Doutorado, UFSC.
- Sardiña I. Milán, 2000. "Solução em Paralelo de Problemas Parabólicos de Grande Porte: Desenvolvimento de Software e Hardware", IPRJ - UERJ, Brasil.
- Spezzano G., D. Talia, 1999. "Programming cellular automata algorithms on parallel computers". *Future Generation Computer Systems* 16, p203-216.
- von Neumann, J., 1966. "Theory of Self-reproducing Automata", University of Illinois Press, Urbana.

SIMULATION OF INCOMPRESSIBLE FLUID FLOW WITH NET GAS ALGORITHMS AND PARALLEL COMPUTING

Ariane Jevaux Souza

Instituto Politécnico – IPRJ, Universidade do Estado do Rio de Janeiro – UERJ, CP 97.282, CEP 28601-970,
Nova Friburgo – RJ, Brasil.
jariane@iprj.uerj.br

Antônio José da Silva Neto

Instituto Politécnico – IPRJ, Universidade do Estado do Rio de Janeiro – UERJ, CP 97.282, CEP 28601-970,
Nova Friburgo – RJ, Brasil.
ajsneto@iprj.uerj.br

João Flávio Vieira de Vasconcellos

Instituto Politécnico – IPRJ, Universidade do Estado do Rio de Janeiro – UERJ, CP 97.282, CEP 28601-970,
Nova Friburgo – RJ, Brasil.
jflavio@iprj.uerj.br

Luiz Orlando Emerick dos Santos

Universidade Federal de Santa Catarina – UFSC, Florianópolis – SC, Brasil
emerick@Impt.ufsc.br

Abstract: The algorithm of net gas is a technique that has been used in the numerical simulation of flow of fluids. This technique is not based on differential equations. It has been used in the simulation of complex flows, for example in porous media, without the need to solve differential equations for the modelling of the phenomenon of interest.

The computational implementation of the net gas algorithm is simple, but the computational cost is high. For this reason, starting from the serial implementation developed by the team of Prof. Philippi in Universidade Federal de Santa Catarina (UFSC) we present in this work a parallel computing implementation.

Key words: incompressible fluid flow, net gas flow algorithm, parallel computing.