Métodos de Decomposição de Domínio na Solução de Equações Diferenciais Parciais

André Luis Martinotto[1]
Delcino Picinin Júnior[2]
Rogério Luis Rizzi[3]
Ricardo Vargas Dorneles[4]
Tiarajú Asmuz Diverio[5]

Resumo

Esta artigo apresenta um estudo sobre métodos de decomposição de domínio na solução das equações diferencias parciais (EDPs) da hidronâmica no modelo HIDRA, que é modelo computacional paralelo multifísica com balanceamento dinâmico de carga desenvolvido no Grupo de Matemática da Computação e Processamento de Alto Desempenho (GMCPAD) para a simulação do escoamento e do transporte de substâncias em corpos de águas. Esse modelo pode ser aplicado sobre qualquer domínio físico (RIZZI, 2002) (DORNELES, 2003).

1 Introdução

Muitos fenômenos físicos podem ser modelados matematicamente através de EDPs, que são definidas em um espaço contínuo e infinito de pontos. Para resolver numericamente essas EDPs é necessário o uso de algum método de discretização, como, por exemplo, os métodos de diferenças finitas e de volumes finitos. Esses métodos transformam o domínio contínuo em um domínio discreto, com um número finito de pontos (células computacionais).

O processo de discretização resulta, geralmente, em sistemas de equações lineares que necessitam ser resolvidos a cada passo de tempo. Numericamente a acurácia da solução das EDPs depende, além do método de aproximação das EDPs utilizado, do número pontos utilizados na discretização. Quanto maior o número de pontos maior a acurácia obtida. Em contrapartida, maior será a ordem dos sistemas de equações a serem resolvidos (RIZZI, 2002). Sendo assim, em simulações numéricas realísticas de grande escala espaço-temporal, e que

[1] almartin@inf.ufrgs.br Bolsista CNPq
[2] picinin@inf.ufrgs.br
[3] rizzi@inf.ufrgs.br
[4] cadinho@inf.ufrgs.br
[5] Apoio: CNPq, FAPERGS e LabTeC UFRGS/Dell

exigem uma alta acurácia numérica, o tempo computacional necessário para simular tais fenômenos pode ser muito elevado. Uma alternativa, cada vez mais freqüente, para resolver este problema é o uso de computação de alto desempenho, em particular o uso de *clusters* de PCs.

Duas das mais importantes abordagens para a solução de EDPs em paralelo são: a decomposição de dados e a decomposição de domínio. Na primeira abordagem, a decomposição de dados, gera-se um único sistema de equações para todo o domínio que é resolvido através de um método numérico paralelizado, como por exemplo, o gradiente conjugado ou o GMRES.

A segunda abordagem consiste na utilização de um método de decomposição de domínio. Esses métodos são baseados no particionamento domínio computacional em subdomínios, de modo que a solução global do problema é obtida pela combinação apropriada das soluções obtidas em cada um dos subdomínios. Uma vez que os subdomínios podem ser tratados independentemente, tais métodos são atrativos para ambientes de memória distribuída (CHAN; MATHEW, 1994).

Neste trabalho foram desenvolvidas implementações, utilizando a abordagem de decomposição de domínio, para a solução paralela das EDPs que modelam o fenômeno da hidrodinâmica no modelo HIDRA. Os sistemas de equações resultantes da discretização, em diferenças finitas, das EDPs são de grande porte, lineares e esparsos e simétricos e definidos-positivos (SDP). Para a solução desses sistemas emprega-se o método do gradiente conjugado (GC). O paralelismo foi explorado através do uso do paradigma de troca de mensagens. Para o desenvolvimento das implementações utilizou-se o padrão MPI (*Message Passing Interface*).

2 Métodos de Decomposição de Domínio

Métodos de decomposição de domínio (MDDs) são baseados no particionamento do domínio em subdomínios e a solução global é obtida combinando as soluções locais. Desta forma, o problema original pode ser tratado como uma série de subproblemas de tamanho reduzido que são resolvidos localmente. Uma vez que esses subdomínios podem ser tratados independentemente, tais métodos são atrativos para ambientes paralelos de memória distribuída. Alguns dos principais atrativos para o uso dos MDDs são: necessitam de pouca comunicação, a qual, em geral, fica restrita nas fronteiras dos subdomínios; possuem versatilidade para trabalhar com distintos modelos de EDPs que são definidas em diferentes sub-regiões do domínio global; e podem ser utilizados para a construção de pré-condicionadores para métodos iterativos (SMITH; BJORSTAD; GROPP, 1996).

Existe uma série de métodos de decomposição de domínio, sendo que a maioria dos autores os classificam em dois grandes grupos: métodos de Schwarz, onde os subdomínios possuem uma região de sobreposição, e métodos de Schur, que não apresentam sobreposição dos domínios. Métodos de decomposição de domínios sem sobreposição não serão tratados neste trabalho (para maiores informações sobre esses métodos consulte (SMITH; BJORSTAD; GROPP, 1996) e (CHAN; MATHEW, 1994)).

2.1 Particionamento do Domínio

O emprego de MDDs requer o particionamento do domínio computacional entre os processadores disponíveis. O particiomento do domínio deve ser feito de forma que a carga de trabalho de cada processador seja proporcional a sua capacidade de processamento. Além disso, em problemas que requerem a troca de informações nas fronteiras dos subdomínios, como no modelo de hidrodinâmica utilizado para os testes, o particionamento deve ser feito de modo a reduzir as fronteiras dos subdomínios e, conseqüentemente, a comunicação entre os processadores.

O particionamento de domínio pode ser modelado como um problema de particionamento de grafo, onde os vértices do grafo representam as células do domínio e as arestas representam os dados trocados entre os subdomínios. O total de comunicação entre os subdomínios é dado pelo total de arestas entre os subdomínios.

Considerando que um grafo representa o domínio do problema deve-se dividi-lo em k subgrafos, onde k é o número de processadores disponíveis, de modo que cada processador tenha um número semelhante de vértices (células) e que o número de arestas (comunicação) entre as partes seja o menor possível. O problema de dividir um grafo em k subgrafos com as propriedades de maximizar a computação e minimizar as comunicações entre eles é um problema NP-Difícil, existindo apenas heurísticas que encontram uma aproximação e não a solução ótima (DORNELES, 2003).

Diferentes bibliotecas implementam algoritmos de particionamento de grafos, entre essas se destacam o METIS, Chaco, JOSTLE. Para o desenvolvimento deste trabalho foi utilizado, no particionamento do domínio, o Pacote METIS 4.0 (KARYPIS; KUMAR, 1998).

Em geral, quando usado algum algoritmo de parcionamento de grafos esse resulta em subdomínios sem sobreposição. Para o uso de MDDs com sobreposição é necessário expandir os subdomínios de forma a garantir que todas as células existentes nas fronteiras artificiais, isto é criadas pelo particionamento, pertençam, também, ao interior dos subdomínios vizinhos (CAI, 2002). A necessidade desta sobreposição, bem como a especificação do tamanho mínimo da região de sobreposição, é decorrente do particular esquema numérico empregado no modelo discreto.

2.2 Métodos de Decomposição de Domínio com Sobreposição

Métodos de Schwarz baseiam-se no particionamento do domínio em um conjunto de subdomínios que se sobrepõem. A solução em cada subdomínio pode ser obtida através do uso de um método iterativo ou direto. O método de Schwarz consiste basicamente em resolver o problema original em cada subdomínio e trocar as condições de contorno nas fronteiras artificiais criadas pelo particionamento. A forma escolhida para a troca das condições de contorno entre os subdomínios vizinhos determina as variantes do método: versão multiplicativa ou aditiva (SILVA, 1997).

Na versão multiplicativa os subdomínios utilizam as condições de contorno calculadas nos subdomínios vizinhos na mesma iteração. Desta forma, essa versão é naturalmente seqüencial e, para obter paralelismo nessa versão, é necessário o uso de alguma técnica de coloração dos subdomínios. Com essa técnica cada subdomínio recebe uma cor de forma que subdomínios vizinhos não possuam cores idênticas. Subdomínios com mesma cor podem ser resolvidos em paralelo e, conseqüentemente, o número de passos seqüenciais pode ser minimizado. O problema de coloração dos subdomínios pode ser visto como um problema de coloração de grafos, onde vértices do grafo representam os subdomínios e as arestas os subdomínios vizinhos, isto é, subdomínios que possuem uma região de sobreposição (CAI; SAAD, 1993). Achar o número mínimo de cores de um grafo é um problema NP-difícil. Existem muitas heurísticas de baixo custo computacional para a coloração de grafos, na prática, segundo (CAI; SAAD, 1993), um algoritmo guloso com heurísticas oferece bons resultados.

Neste trabalho, utilizou-se um algoritmo guloso para a coloração de grafos. A seleção dos vértices para a coloração é feita em ordem decrescente do grau (número de vértices adjacentes) dos mesmos. A primeira cor que não pertence a nenhum dos vértices adjacentes será escolhida para colorir o vértice. Se os vértices adjacentes, já coloridos, usam todas as cores, o vértice será colorido com uma nova cor (WEST, 2001).

Na versão aditiva os subdomínios utilizam a condições de contorno nos subdomínios vizinhos na iteração anterior, tal que os subdomínios podem ser resolvidos independentemente. Desta forma, a versão aditiva apresenta um maior potencial de paralelismo (SILVA, 1997).

No que diz respeito à convergência, o método aditivo Schwarz, em vários casos, pode exigir até o dobro de iterações em relação ao método multiplicativo Schwarz. É importante ressaltar, ainda, que a convergência dos métodos de Schwarz é influenciada pelo tamanho da região de sobreposição. Quanto maior a região de sobreposição maior a velocidade de convergência. Em compensação, maior será o tamanhos dos subdomínios e, conseqüentemente, o custo computacional para o cálculo das soluções locais. Também é importante notar que o número de iterações necessárias para a convergência pode ser maior com o aumento do número de subdomínios (SMITH; BJORSTAD; GROPP, 1996).

3 Testes e Resultados Obtidos

Os algoritmos desenvolvidos foram implementados utilizando a linguagem C com as bibliotecas de troca de mensagens MPICH 1.2.4, sobre o sistema operacional Linux. Os testes foram realizados no *cluster labtec*. Nos testes utilizou-se células com tamanho $\Delta x = \Delta y = 100$ metros, o que resulta sistemas com 46024 equações. Para a solução dos sistemas de equações locais (subdomínios) utilizou-se o método do GC com um erro de 10^{-6} e na solução global do método de Schwarz um erro de 10^{-4}.

No gráfico da figura refschwarz pode-se observar um comparativo entre o tempo de execução das variantes do método de Schwarz. Observa-se que o método aditivo apresenta um melhor desempenho em ambientes de *clusters*.

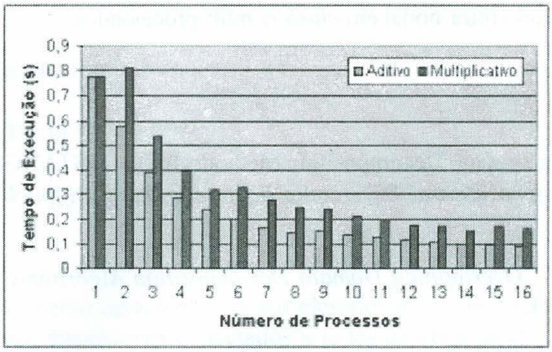

Figura 1: Tempo de Execução - Aditivo X Multiplicativo

Nota-se, ainda, que o desempenho do método multiplicativo é influenciado pelo número de cores. Na tabela 1 pode-se observar o número de cores utilizadas. Em relação a convergência, o método aditivo exige uma iteração a mais em relação ao método multiplicativo. No método multiplicativo a solução global é obtida em três iterações, enquanto no método aditivo são necessárias quatro iterações.

Tabela 1: Algoritmo de Coloração

Número de Subdomínios	Número de Cores
2 à 5	2
6 à 14	3
15 à 16	4

4 Conclusões

Este trabalho apresentou o uso de métodos de Schwarz na solução dos sistemas de equações gerados pelo modelo computacional paralelo para simulação da hidrodinâmica e transporte de substâncias em corpos de água, mais especificamente o modelo de hidrodinâmica.

Os testes mostram que os métodos de Schwarz, principalmente na versão aditiva, são uma boa opção para problemas modelados através de EDPs, uma vez que o domínio computacional pode ser particionado e os subdomínios tratados independentemente, restringindo a comunicação nas fronteiras.

Como trabalhos futuros pretende-se testar métodos de decomposição de domínio sem sobreposição e a adição de *threads* nas implementações desenvolvidas para uma melhor exploração do paralelismo intra-nodal em *clusters* multiprocessados.

Referências

CAI, X. Overlapping Domain Decomposition Methods. In: LANGTANGEN, H. P.; TVEITO, A. (Ed.). *Computational Partial Differential Equations Using Diffpack - Advanced Topics*. [S.l.: s.n.], 2002.

CAI, X.; SAAD, Y. *Overlapping Domain Decomposition Algorithms for General Sparse Matrices*. [S.l.], 1993.

CHAN, T. F.; MATHEW, T. Domain Decomposition Algorithms. *Acta Numerica*, p. 61–143, 1994.

DORNELES, R. V. *Particionamento de Domínio e Balanceamento de Carga no Modelo HIDRA*. Tese (Doutorado) — Instituto de Informática, UFRGS, Porto Alegre - RS, 2003.

KARYPIS, G.; KUMAR, V. *METIS: A Software Package for Partitioning Unstructured Graphs, Partitioning Meshes, and Computing Fill-reducing Orderings of Sparse Matrices*. 1998. Disponível em: http://www.cs.umn.edu/~karypis. Acesso em: maio de 2002.

RIZZI, R. L. *Modelo Computacional Paralelo para a Hidrodinâmica e para o Transporte de Massa Bidimensional e Tridimensional*. Tese (Doutorado) — Instituto de Informática, UFRGS, Porto Alegre, 2002.

SILVA, R. et al. Iterative Local Solvers for Distributed Krylov-Schwarz Methods Applied to Convection-Diffusion Problems. *Computer Methods for Applied Mechanics and Engineering*, v. 149, 1997.

SMITH, B.; BJORSTAD, P.; GROPP, W. *Domain Decomposition: Parallel Multilevel Methods for Elliptic Partial Differential Equations*. Cambridge: Cambridge University Pres, 1996.

WEST, D. B. *Introduction to Graph Theory*. [S.l.]: Published by Prentice Hall, 2001.