Resolução Verificada de Sistemas Lineares Intervalares Densos de Grande Porte em Arquiteturas *Multicore*

Cleber Roberto Milani, Luiz Gustavo Fernandes
Pontifícia Universidade Católica do Rio Grande do Sul
{cleber.milani, luiz.fernandes}@pucrs.br

Mariana Kolberg
Universidade Luterana do Brasil
mariana.kolberg@ulbra.br

Introdução

As Equações Diferenciais (EDs) e os Sistemas de Equações Lineares Algébricas (SELAs) são as ferramentas matemáticas mais utilizadas na modelagem de problemas e simulações científicas. Sabe-se, no entanto, que o fato de um algoritmo ter sido testado e estar funcionando não implica a exatidão do seu resultado [HAM 97]. Frequentemente, o computador produz resultados incorretos para um problema numérico, não devido a erros de programação ou ao uso de *hardware* não confiável, mas porque computadores são máquinas discretas e finitas que não conseguem tratar alguns dos aspectos contínuos e infinitos da matemática.

A implementação de algoritmos com verificação do resultado faz uso de Aritmética Intervalar para garantir o rigor matemático dos resultados. Assim, as operações têm como resultado um intervalo que contém o resultado exato. Esse intervalo recebe o nome de *enclosure* [KOL 08a]. De posse dessa informação, é possível optar por um algoritmo alternativo, repetir a computação utilizando maior precisão ou informar ao usuário quando o resultado não é válido.

Além de viabilizar a Computação Verificada [HAM 97], outro grande benefício da Aritmética Intervalar é permitir o tratamento de **sistemas lineares intervalares**, ou seja, sistemas em que os valores de A e $\vec{b}$ são intervalos ao invés de números pontuais. Tais sistemas têm adquirido cada vez mais importância científica pelo fato de permitirem a representação de dados oriundos de medições imprecisas ou modelagens com incertezas.

Uma das ferramentas de Computação Verificada mais populares é o C-XSC (*C for eXtended Scientific Computing*). Entretanto, bibliotecas puramente de Computação Verificada, embora permitam a resolução verificada e, em alguns casos, deem suporte aos sistemas intervalares, acabam por se transformar no gargalo da aplicação por requererem uma grande quantidade de operações adicionais.

Mesmo utilizando-se bibliotecas com resultados aproximados como LAPACK (*Linear Algebra PACKage*) ou sua versão paralela para memória distribuída ScaLAPACK (*SCAlable Linear Algebra PACKage*), a resolução de SELAs continua apresentando grande custo computacional quando se trata de sistemas de grande porte. Além disso, tais bibliotecas não oferecem suporte à resolução verificada de sistemas pontuais ou intervalares. Assim, no contexto da Computação Verificada, emprega-se, em geral, essas bibliotecas apenas para auxiliar na otimização de alguns trechos específicos dos algoritmos.

Computação Verificada em Multicores

Nos últimos anos, diversos trabalhos de Computação Verificada têm sido realizados aplicando técnicas de Computação Paralela em agregados de computadores (*clusters*). Tais trabalhos combinam a biblioteca MPI (*Message Passing Interface*) com o ScaLAPACK. Alguns deles, como [KOL 08b], operam sistemas lineares do tipo pontual enquanto outros, como [KOL 08a], resolvem sistemas lineares intervalares.

Porém, muitas modificações vêm ocorrendo na Computação de Alto Desempenho. Os processadores *multicore* vêm ganhando cada vez mais espaço e espera-se que, em poucos anos, o número de núcleos nesses exceda uma dezena. Nesse contexto, adaptar as ferramentas numéricas para exploração dos recursos dessa arquitetura se faz indispensável. Como exemplo tem-se o projeto PLASMA (*Parallel Linear Algebra for Scalable MultiCore Architectures*) apresentado em [BUT 08] como sucessor da ScaLAPACK para álgebra linear de alto desempenho em *multicore*. A PLASMA explora o paralelismo em nível de *threads* baseada em mecanismos mais eficientes para comunicação entre os processadores e sincronização dos recursos.

Considerando essa quebra de paradigma na Computação Paralela, o objetivo deste trabalho é desenvolver um *solver* (ferramenta computacional para resolução) de SELAs densos intervalares de grande porte com verificação automática dos resultados otimizado para execução em arquiteturas *multicore*. Tal ferramenta utiliza os mesmos princípios matemáticos e numéricos de [KOL 08a], porém tem como grande diferencial explorar diferentes estratégias de paralelização de modo a permitir sua execução em arquiteturas *multicore* e beneficiar-se de suas particularidades alcançando escalabilidade e alto desempenho nas mesmas.

Uma implementação sequencial foi desenvolvida para processadores *multicore* e seu custo computacional contabilizado. Para tal, dividiu-se a execução em 10 passos e verificou-se que a complexidade concentra-se basicamente em dois passos. Tais operações correspondem a operações de inversão de matriz e pré-condicionamento do sistema e consomem, respectivamente, 55% e 42% do tempo total de execução. Testes de exatidão foram executados. Os resultados dos testes realizados estão orientando as escolhas referentes à paralelização do *solver* acima mencionado.

Referências

[HAM 97] R. Hammer and D. Ratz and U. W. Kulisch and M. Hocks. C++ Toolbox for Verified Scientific Computing I: Basic Numerical Problems. 1997, Springer-Verlag New York. Secaucus, USA.

[KOL 08a] M. L. Kolberg and M. Dorn and G. Bohlender and L. G. Fernandes. Parallel Verified Linear System Solver for Uncertain Input Data. Proceedings of 20th International Symposium on Computer Architecture and High Performance Computing, pages 89–96, 2008.

[KOL 08b] M. L. Kolberg and L. G. Fernandes and D. M. Claudio. Dense Linear System: A Parallel Self-verified Solver. International Journal of Parallel Programming, volume 36, number 4, pages 412–425, 2008.

[BUT 08] A. Buttari and J. J. Dongarra and J. Kurzak and J. Langou and P. Luszczek and S. Tomov. The Impact of Multicore on Math Software. Applied Parallel Computing. State of the Art in Scientific Computing. Springer Lecture Notes in Computer Science, volume 4699, pages 1–10, 2008.