

Software livre SisLLLin para solução de sistemas lineares grandes e esparsos e sua integração ao software de CFD FluidoSSS

Free Software SisLLLin for solution of large and sparse linear systems and their integration into the CFD software FluidoSSS

Lauro Cássio Martins de Paula

laurocassio21@gmail.com
PUC Goiás

Leandro Barra Santana de Souza

souza.leandro@gmail.com
UFMG

Leonardo Barra Santana de Souza

professorleobarra@gmail.com
com
UFG

Harry Edmar Schulz

harryeschulz@gmail.com
EESC-USP

Resumo

Desenvolveu-se neste trabalho um *software* de uso irrestrito e código aberto, em linguagem C++, para solução iterativa de sistemas lineares. Os métodos clássicos Jacobi e Gauss-Seidel foram implementados juntamente com BiCGStab e BiCGStab(2), em destaque na literatura contemporânea para solução de sistemas grandes e esparsos. Em sua primeira aplicação, o *software* SisLin foi integrado ao *software* FluidoSSS de dinâmica dos fluidos computacional, também em desenvolvimento nessa linha de pesquisa e de uso irrestrito, para simulação numérica de um escoamento bidimensional de fluido incompressível sob pressão. Mil sistemas lineares de 1.200 equações foram resolvidos para comparação da velocidade de convergência dos quatro métodos. Especificamente neste estudo inicial, constatou-se uma equivalência entre BiCGStab e BiCGStab(2) e a superioridade de ambos em relação aos métodos Jacobi e Gauss-Seidel. O *software* foi desenvolvido com interface gráfica e fácil entrada de dados, buscando-se aplicações didáticas nas diversas áreas da Ciência que envolvem a solução de sistemas lineares.

Palavras-chave: Software Livre. Sistemas Lineares. Dinâmica dos Fluidos Computacional.

Abstract

A free software with open source code in C++ language was developed for iterative solution of linear systems. The classic methods Jacobi and Gauss-Seidel were implemented, as well as the BiCGStab and BiCGStab(2), distinguished for solving large sparse linear systems. In its first application, the software, named SisLin, was integrated to the free CFD software FluidoSSS, also in development in this research work, for the numerical simulation of a two dimensional incompressible and pressurized fluid flow. A thousand linear systems of 1200 equations were solved for comparison of convergence speed of the four methods. Specifically in this initial study, it was found equivalence between BiCGStab and BiCGStab(2), and their superiority over Jacobi and Gauss-Seidel methods. The software was developed with graphic interface and easy data input, seeking didactic applications in various fields of Science that involve the solution of linear systems.

Keywords: Free software. Linear systems. CFD.

1 Introdução

As equações diferenciais de conservação da massa e da quantidade de movimento aplicadas a um sistema fluido, também conhecidas como as equações de Navier-Stokes, são resolvidas de forma numérica para que escoamentos de fluidos sejam representados matematicamente. Essa representação é chamada de simulação numérica de escoamentos, estudada na subárea da Mecânica dos Fluidos denominada Dinâmica dos Fluidos Computacional (DFC) ou *Computational Fluid Dynamics* (CFD), e auxilia pesquisadores e profissionais a compreender melhor, juntamente com trabalhos experimentais, os fenômenos físicos existentes na natureza.

Para o caso de escoamentos incompressíveis, que englobam líquidos (água, óleo, sangue etc.) ou gases sob certas condições restritivas (baixo número de Mach), métodos numéricos que utilizam acoplamento pressão-velocidade são

largamente empregados, tais como os métodos SIMPLE (Patankar e Spalding, 1972) e Passos Fracionados (Kim e Moin, 1985).

Para solução das equações de conservação através desses métodos, são gerados sistemas lineares cujas incógnitas são diferenciais de pressão (ou variações de pressão) em diversas regiões do escoamento estudado, denominadas células, nas quais se deseja calcular, a partir desses diferenciais, valores de pressão e velocidade em cada *instante* (cada passo no tempo). Quanto menor essas células e o passo no tempo, e quanto maior o número de passos ou período que se deseja simular, melhor a qualidade da simulação numérica, ou seja, melhor a representação do escoamento, que pode ser, por exemplo: o escoamento de ar em torno de um avião, um foguete, um edifício ou um submarino, para estudo de sua aerodinâmica; o escoamento bifásico de petróleo e água em tubulações; o escoamento de água em um canal de drenagem urbana; a inundação de uma região, causada pelo rompimento da barragem de um reservatório; um rio carregado de sedimentos, causando assoreamento em um reservatório de barragem; a dispersão de um poluente lançado no meio ambiente por uma indústria; ou o escoamento de sangue em um coração.

De acordo com Sousa (2003), a solução dos sistemas lineares para cálculo das variações de pressão representa, em muitos casos, mais de 80% do tempo de processamento do computador. Em simulações de alta qualidade, requer-se um tempo de horas, dias ou mesmo semanas de processamento computacional. Nesse sentido, métodos robustos e com rápida convergência são necessários para a solução dos sistemas lineares, que chegam facilmente a milhões de equações e incógnitas e devem ser resolvidos em cada um dos milhares ou milhões de passos no tempo.

Como os sistemas lineares encontrados na DFC são grandes e esparsos, métodos iterativos de solução são mais indicados do que métodos exatos, por usarem menos memória e reduzirem erros de arredondamento do computador (Franco, 2006). Métodos iterativos clássicos, como Jacobi, Gauss-Seidel e SOR, apesar de sua fácil implementação, podem apresentar convergência lenta ou mesmo não convergirem para grandes sistemas e, portanto, são inadequados para DFC (Versteeg e Malalasekera, 1995).

Portanto, a pesquisa, a implementação e a comparação dos métodos iterativos nos códigos computacionais compreendem tarefa importante na Dinâmica dos Fluidos Computacional e em outras áreas da Ciência que envolvam a solução de grandes sistemas de equações. Nesse sentido, este trabalho compreendeu o desenvolvimento de um *software* livre, nomeado SisLin, para uma solução iterativa de sistemas lineares e uma comparação inicial da eficiência dos métodos iterativos implementados, baseada na velocidade de solução dos sistemas gerados no programa computacional de DFC, FluidoSSS, apresentado em Souza *et al.* (2010a). Ambos os *softwares*, FluidoSSS e SisLin, foram desenvolvidos em trabalhos de iniciação científica em cursos de graduação, no Departamento de Hidráulica e Saneamento da Escola de Engenharia de São Carlos (Universidade de São Paulo) e na Escola de Engenharia Civil (Universidade Federal de Goiás).

2 Metodologia

O SisLin foi elaborado em linguagem C++ para solução de sistemas lineares através de quatro métodos iterativos: os métodos clássicos Jacobi e Gauss-Seidel, amplamente citados na literatura, e os métodos BiCGStab (*Bi Conjugate Gradient Stabilized*) e BiCGStab(2), em destaque na literatura contemporânea. Uma interface gráfica foi criada para facilitar seu uso e a transferência, a partir de um arquivo texto ou de outro *software*, dos dados do sistema a ser resolvido. A solução do sistema é apresentada também em arquivo texto.

2.1 Método BiGStab

O BiCGStab, método do Gradiente Biconjugado Estabilizado, destaca-se por sua robustez e eficiência. Foi desenvolvido por Vorst (1992), trabalho que foi, segundo o ISI (Institute for Scientific Information), o mais citado na área de Matemática durante a década de 1990. Também tem sido usado em trabalhos nacionais de destaque em DFC, como Alamy Filho (2006), da simulação de escoamento de rios envolvendo transporte de sedimentos.

O algoritmo do BiCGStab é apresentado na Figura 1. Algumas adaptações de nomenclatura foram feitas em relação ao algoritmo original, na busca de melhor entendimento do método. Na figura, letras minúsculas sem negrito representam escalares, letras minúsculas em negrito representam vetores expressos na forma matricial e as letras maiúsculas em negrito representam matrizes. Parênteses com vetores separados por vírgula representam multiplicação desses vetores.

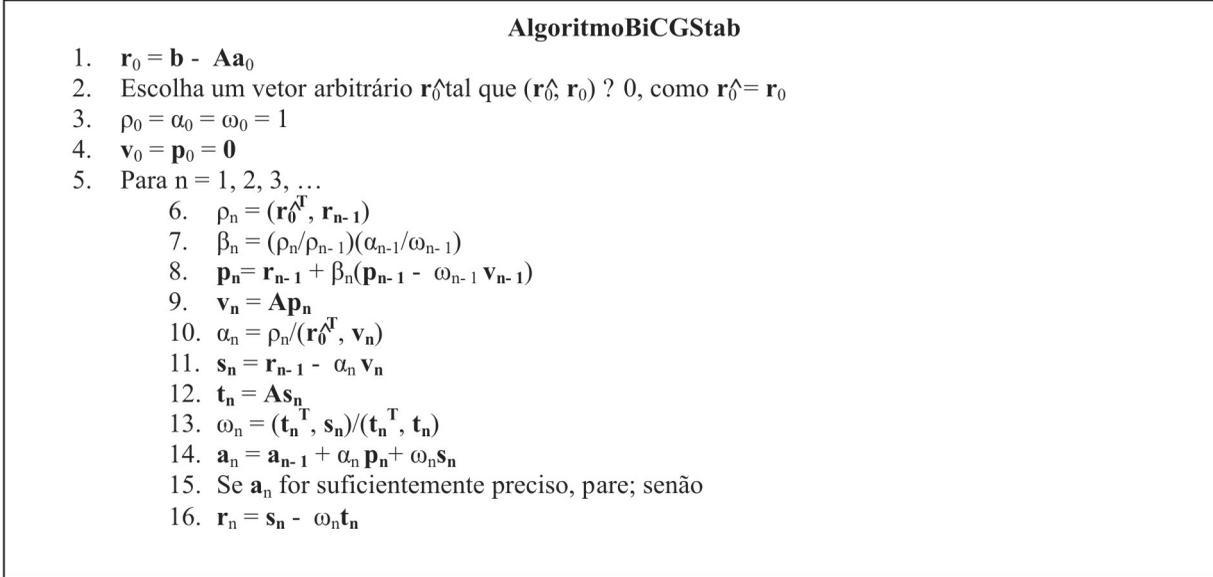


Figura 1: Algoritmo do método iterativo BiCGStab, adaptado de Vorst (1992).

2.2 Método BiCGStab(2)

Desenvolvido por Sleijpen e Vorst (1995), o BiCGStab(2) reúne, segundo seus criadores, as vantagens do BiCGStab e do GMRES (Generalized Minimal Residual), resultando em um método com garantia de convergência superior à do BiCGStab e adequado para solução de sistemas lineares gerados na solução das equações diferenciais de escoamentos de fluidos. O algoritmo do BiCGStab(2) é apresentado na Figura 2, seguindo a nomenclatura já descrita.

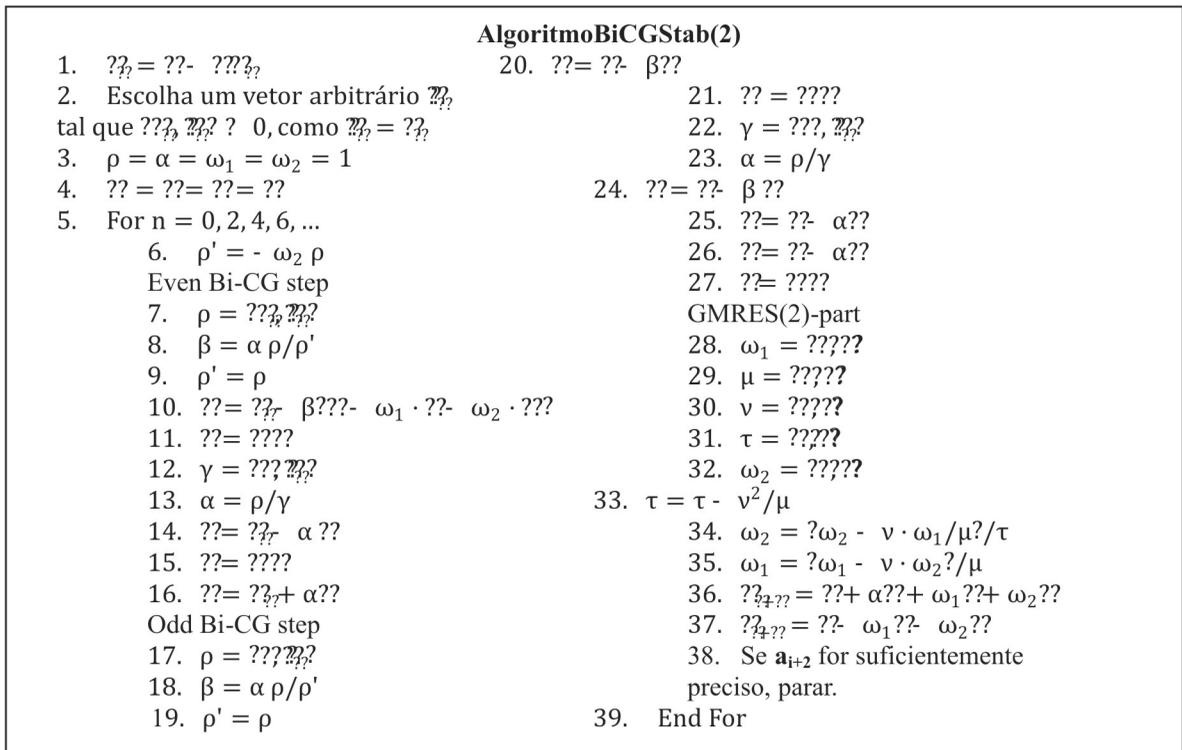


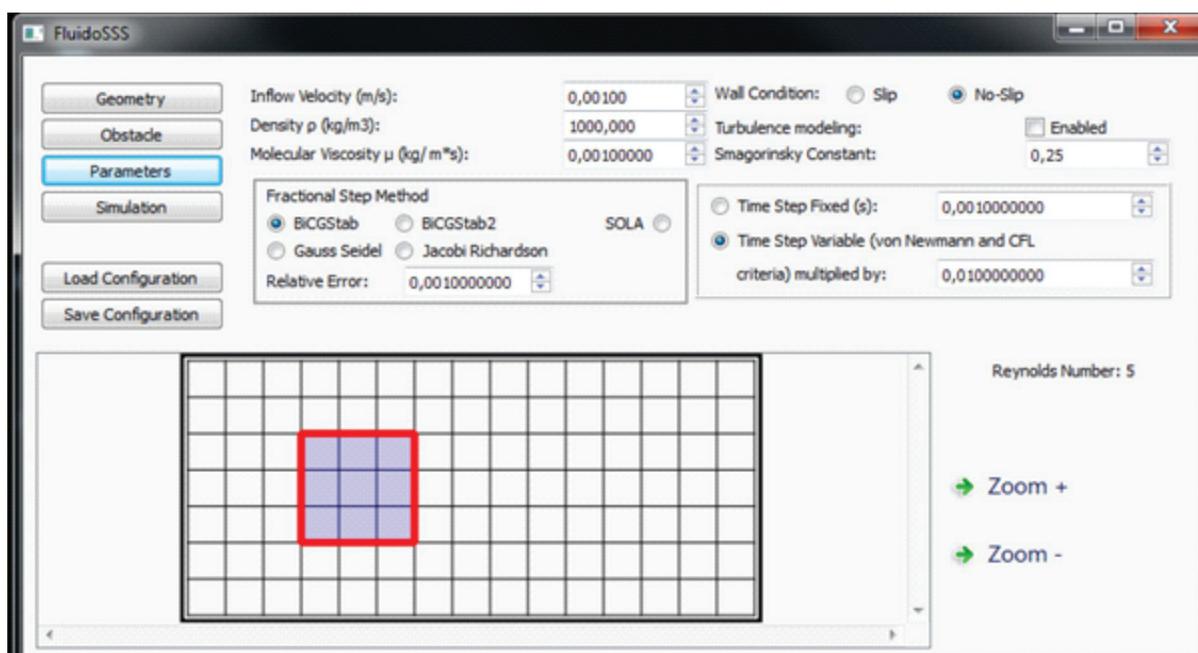
Figura 2: Algoritmo do método iterativo BiCGStab(2), adaptado de Sleijpen e Vorst (1995).

2.3 Software FluidoSSS

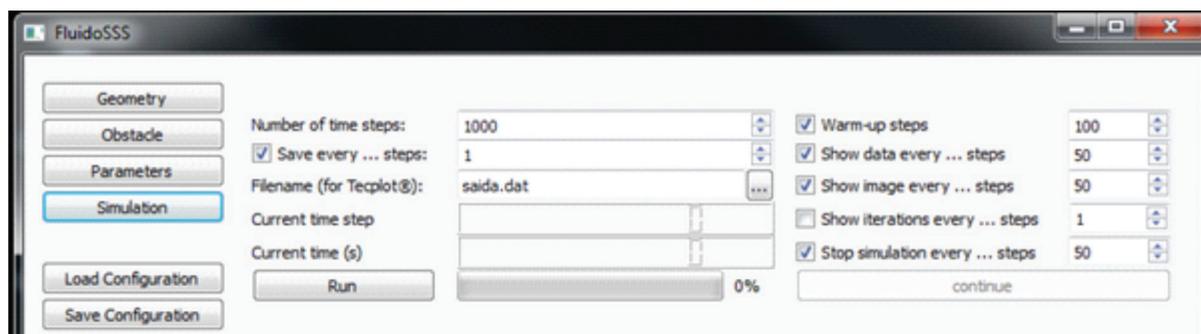
Criou-se o *software* acadêmico FluidoSSS, em linguagem C++, de uso livre e com fácil entrada de dados, para simulação numérica de escoamentos bidimensionais incompressíveis sob pressão, em regime laminar ou turbulento.

A discretização das equações de conservação da massa e da quantidade de movimento é feita com malha cartesiana deslocada e usando volumes finitos, visando o desenvolvimento continuado do código para malhas com refinamento adaptativo ou malhas não estruturadas para simulação de escoamentos sobre superfícies de geometrias complexas. As equações discretizadas são resolvidas através do método SOLA (Solution Algorithm), desenvolvido por Hirt *et al.* (1975), ou do método dos Passos Fracionados, de Kim e Moin (1985). O SOLA foi escolhido por sua relativa facilidade de implementação em programação paralela, objeto de pesquisa futura. Os Passos Fracionados necessitam, em seu algoritmo, da solução de sistemas lineares para cálculo da correção da pressão, uma etapa de cálculo presente nos métodos com acoplamento pressão-velocidade para simulação de escoamentos incompressíveis. Caso esses sistemas sejam resolvidos por métodos eficientes, como o BiCGStab e o BiCGStab(2), presentes no SisLin, o método dos Passos Fracionados supera amplamente o SOLA e se destaca entre os métodos na simulação de escoamentos de fluidos.

A seguir, na Figura 3, duas telas da interface gráfica do FluidoSSS, para especificação de parâmetros necessários à simulação numérica de escoamentos.



(a)



(b)

Figura 3: (a) Tela de entrada dos parâmetros do escoamento, do fluido e do método numérico. (b) Tela de acompanhamento da simulação e armazenamento de dados

2.4 Comparação dos métodos implementados no SisLin

Para verificação da eficiência e da superioridade do BiCGStab e do BiCGStab(2) em relação aos clássicos Jacobi e Gauss-Seidel, foram comparados os quatro métodos com base no tempo de CPU para solução dos sistemas, para uma determinada precisão dos resultados, simulando-se um escoamento bidimensional clássico da mecânica dos fluidos no programa FluidoSSS: o escoamento viscoso entre duas placas planas e paralelas. O número de Reynolds (Re, adimensional importante na Mecânica dos Fluidos) foi definido igual a 1, caracterizando o regime do escoamento como laminar, ou seja, sem flutuações de velocidade e, portanto, sem turbulência. A simulação partiu com velocidade zero em todo o campo computacional, e 1.000 passos no tempo foram simulados para obtenção do regime permanente. A matriz esparsa A do sistema gerado em cada passo tem a forma mostrada na Figura 4.

$$\begin{bmatrix}
 A & a & 0 & a & 0 & 0 & 0 & 0 & 0 \\
 a & A & a & 0 & a & 0 & 0 & 0 & 0 \\
 0 & a & A & a & 0 & a & 0 & 0 & 0 \\
 a & 0 & a & A & a & 0 & a & 0 & 0 \\
 0 & a & 0 & a & A & a & 0 & a & 0 \\
 & & & & \ddots & & & & \\
 0 & 0 & 0 & a & 0 & a & A & a & 0 \\
 0 & 0 & 0 & 0 & a & 0 & a & A & a \\
 0 & 0 & 0 & 0 & 0 & a & 0 & a & A
 \end{bmatrix}
 \begin{bmatrix}
 x_1 \\
 x_2 \\
 x_3 \\
 x_4 \\
 x_5 \\
 \vdots \\
 x_{n-2} \\
 x_{n-1} \\
 x_n
 \end{bmatrix}
 =
 \begin{bmatrix}
 b_1 \\
 b_2 \\
 b_3 \\
 b_4 \\
 b_5 \\
 \vdots \\
 b_{n-2} \\
 b_{n-1} \\
 b_n
 \end{bmatrix}$$

Figura 4: Forma da matriz esparsa dos coeficientes do sistema linear gerado em cada um dos 1.000 passos da simulação.

Um total de 1.000 sistemas lineares, um para cada passo no tempo da simulação, foram criados e resolvidos, cada um contendo 1.200 equações e 1.200 incógnitas. Nota-se que essa quantidade de sistemas, nessa grandeza, geraria resultados simulados muito aquém da qualidade exigida na DFC, caso esses escoamentos apresentassem números de Reynolds mais elevados. Seria o caso de escoamentos turbulentos, comumente encontrados, por exemplo, com Reynolds na ordem de 10⁶. Contudo, como se trata de uma comparação inicial dos métodos, em um trabalho prioritariamente focado na criação de um novo *software* para solução de sistemas lineares, limita-se à comparação dos métodos com 1.000 sistemas lineares de 1.200 equações, compatíveis com escoamentos laminares de Reynolds iguais à unidade. A simulação foi realizada em um computador com processador Intel Core i7 980A 3,33GHz e 12GB de memória RAM.

3 Resultados e Discussão

A interface gráfica da primeira versão do SisLin é ilustrada na Figura 5, com a tela para entrada de dados após escolha de um dos métodos de solução.

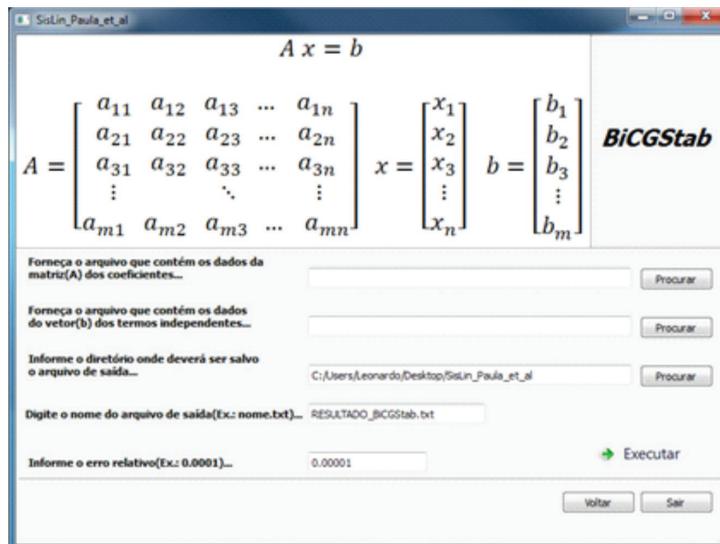
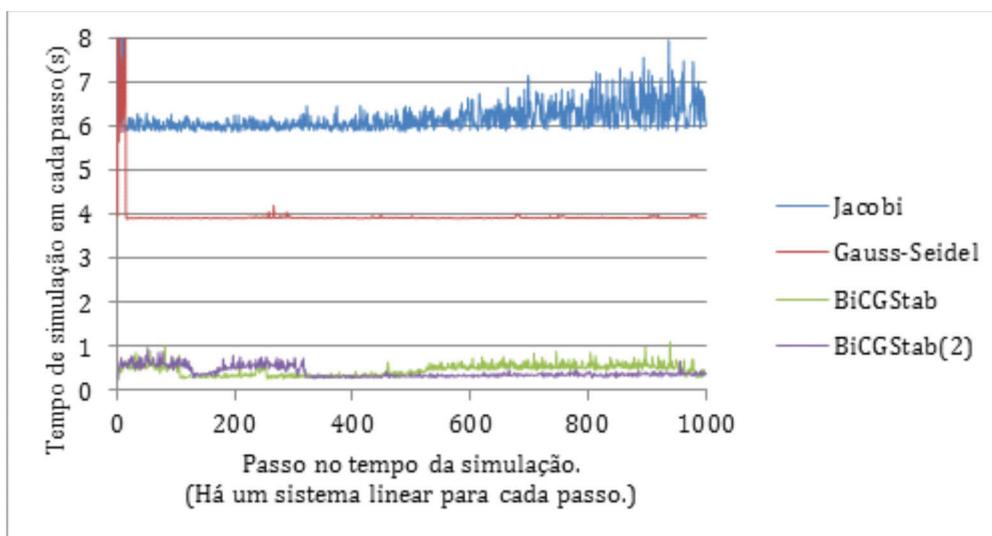
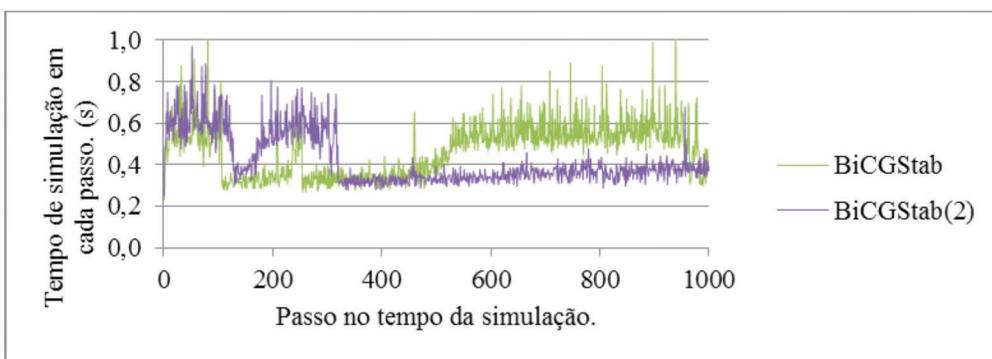


Figura 5: Tela para entrada de dados do sistema linear a ser resolvido. Neste caso, ilustra-se a tela do BiCGStab.

O gráfico comparativo da velocidade de processamento da simulação do canal com os quatro métodos é apresentado nas Figuras 6 e 7, com o tempo da solução de cada sistema e com o tempo acumulado na solução dos 1.000 sistemas, respectivamente.



(a)

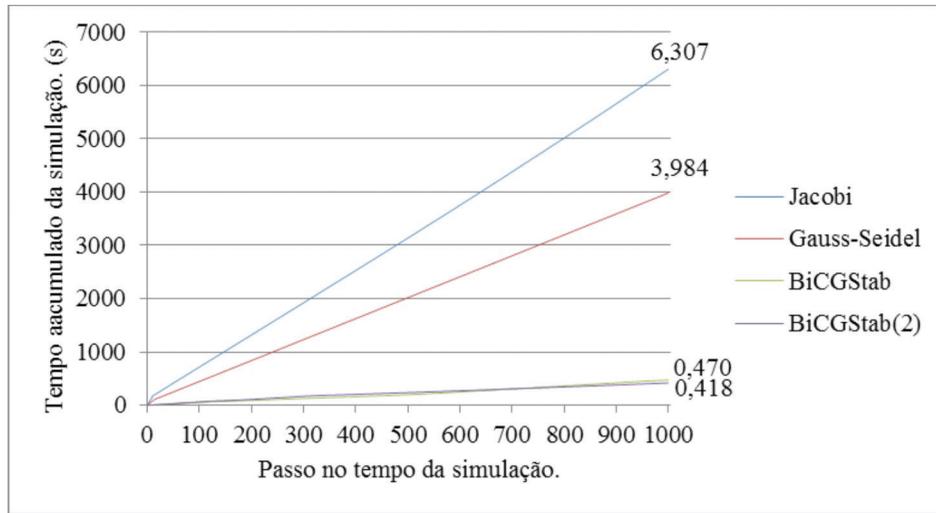


(b)

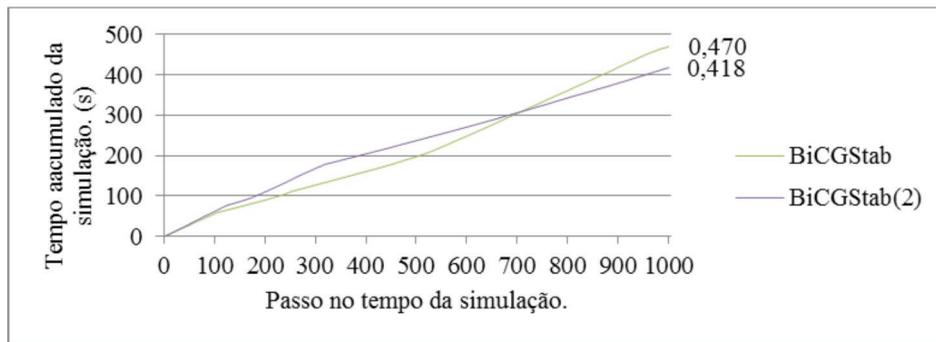
Figura 6: (a) Comparação da velocidade de cálculo de cada sistema linear, um para cada passo no tempo. (b) Detalhe do gráfico (a), com foco no desempenho dos métodos BiCGStab e BiCGStab(2).

Nota-se, na Figura 6, a superioridade dos métodos BiCGStab e BiCGStab(2) na solução dos sistemas tratados. Nos primeiros passos da simulação, os sistemas lineares foram resolvidos por Jacobi e Gauss-Seidel com picos de tempo que chegaram, respectivamente, a 31s e 16s, valores de tempo inclusive não mostrados na figura, ao passo que o BiCGStab ultrapassou a casa de 1s levemente em apenas duas ocasiões (1,027s e 1,108s); e o BiCGStab(2), mais estável, em nenhum momento. Com foco apenas nos métodos mais rápidos, observa-se que o BiCGStab soluciona os primeiros sistemas, em média, mais rapidamente, sendo ultrapassado pelo BiCGStab(2) após 300 sistemas, aproximadamente. Já nos últimos sistemas, o BiCGStab volta a ser mais rápido em média.

O tempo total despendido por cada método na solução dos 1.000 sistemas é mostrado na Figura 7. Os métodos BiCGStab e BiCGStab(2) realizaram as 1.000 simulações com tempo aproximadamente 14 vezes menor do que o Jacobi e 9 vezes menor do que o Gauss-Seidel.



(a)



(b)

Figura 7: (a) Comparação da velocidade de cálculo de solução dos sistemas lineares, com base no tempo acumulado. (b) Detalhe do gráfico (a), com foco no desempenho dos métodos BiCGStab e BiCGStab(2).

Na Figura 8, ilustra-se o resultado da simulação numérica do escoamento laminar entre suas placas planas e paralelas, com $Re = 1$, usada para comparação dos métodos de solução dos sistemas lineares. Com escoamento uniforme na seção de entrada, observa-se o desenvolvimento da camada limite da esquerda para a direita, resultado da condição de não escorregamento (velocidade igual a zero, em azul) junto às paredes superior e inferior. As cores em tons de vermelho no centro do escoamento correspondem às velocidades mais elevadas, nas regiões distantes das paredes.

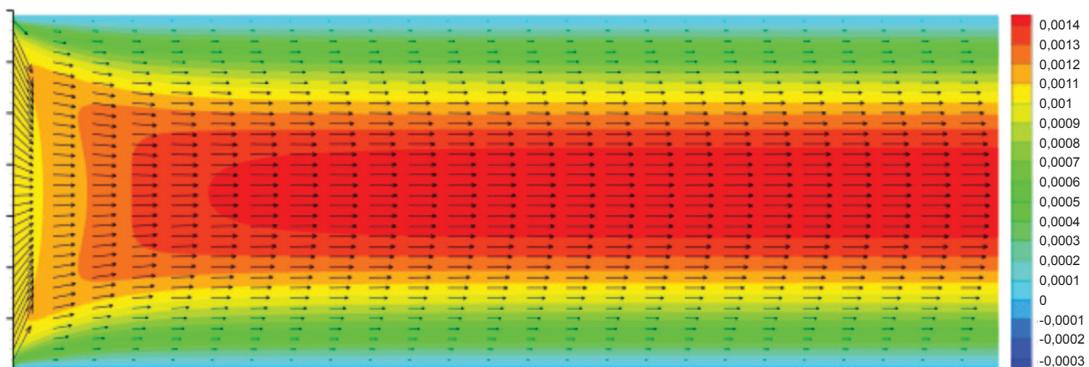


Figura 8: Desenvolvimento da camada limite laminar entre placas planas paralelas. A escala corresponde à componente longitudinal da velocidade (da esquerda para a direita na figura)

Outras simulações de escoamentos confinados (sob pressão) foram realizadas com o FluidoSSS, usando os métodos BiCGStab e BiCGStab(2). Comparadas às simulações equivalentes ao método SOLA, realizadas por Souza *et al.* (2010b), a diminuição do tempo de processamento numérico foi considerável em todos os casos, variando de acordo com o refinamento da malha computacional. Exemplos dessas simulações são o escoamento laminar com vários obstáculos e o escoamento turbulento sobre um obstáculo, ilustrados nas Figuras 9 e 10.

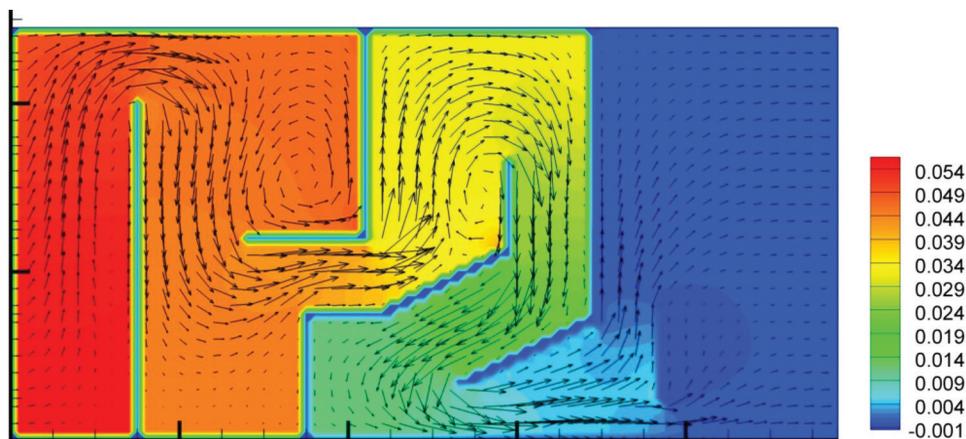


Figura 9: Escoamento laminar com vários obstáculos, com $Re = 1$. A escala corresponde à pressão no escoamento.

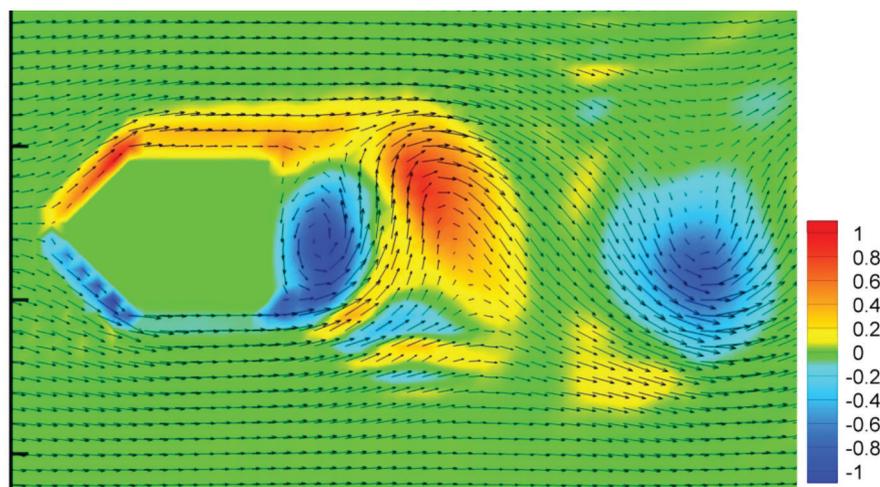


Figura 10: Escoamento turbulento sobre um obstáculo, com $Re = 1000$. A escala corresponde à variável vorticidade.

Para simulações de escoamentos reais, nas quais os sistemas lineares gerados são muito maiores do que os gerados neste trabalho e o processamento computacional pode durar dias, as diferenças de velocidade de solução dos métodos se ampliam significativamente. Portanto, a implementação de métodos como o BiCGStab e o BiCGStab(2) se torna mais importante e, muitas vezes, indispensável para que as simulações sejam realizadas com qualidade e em tempo não proibitivo.

4 Conclusões

Desenvolveu-se neste trabalho um *software* em linguagem C++, nomeado SisLin, para solução de sistemas lineares através de quatro métodos iterativos: Jacobi, Gauss-Seidel, BiCGStab e BiCGStab(2). Sua integração ao *software* FluidoSSS permitiu uma comparação inicial da velocidade desses métodos na solução de sistemas lineares grandes e esparsos. Especificamente para os sistemas avaliados, constatou-se uma equivalência entre BiCGStab e BiCGStab(2) e a superioridade de ambos em relação aos métodos clássicos de Jacobi e Gauss-Seidel.

O software SisLin é livre, bem como o FluidoSSS, e suas aplicações estendem-se aos diversos problemas matemáticos que envolvem sistemas lineares. Visando sua ampla e irrestrita utilização, suas interfaces gráficas encontram-se em fase final de teste e aprimoramento.

Trabalhos seguintes nessa linha de pesquisa envolverão comparações mais completas entre os métodos de solução de sistemas lineares, adaptação do código computacional para processamento paralelo e simulação de escoamentos tridimensionais.

Referências

- ALAMYFILHO, J. E. *Modelação numérica de processos de sedimentação em escoamentos turbulentos e análise de ressuspensão em canais*. 2006. 233f. Tese (Engenharia Civil)- Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2006.
- FRANCO, N. B. *Cálculo numérico*. São Paulo: Pearson Prentice Hall, 2006.
- HIRT, C. W.; NICHOLS, B. D.; ROMERO, N. C. SOLA: a numerical solution algorithm for transient fluid flows. *Los Alamos National Laboratory Report LA-5852*, Los Alamos, 1975.
- KIM, J.; MOIN, P. Application of a fractional-step method to incompressible Navier-Stokes Equations. *Journal of Computational Physics*, v. 59, p. 308-323, 1985.
- PATANKAR, S. V.; SPALDING, D. B. A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. *International Journal of Heat and Mass Transfer*, v. 15, n. 10, p. 1787-1806, 1972.
- SLEIJPEN, G. L. G.; VORST, H. A. Hybrid bi-conjugate gradient methods for CFD problems. *Computational Fluid Dynamics Review*, [s.l.], p. 457-476, 1995.
- SOUSA, J. F. *Simulação de grandes escalas de escoamentos em um hidrociclone*. 2003. Tese (Engenharia Civil)- Departamento de Engenharia Mecânica, Universidade Federal de Uberlândia, Uberlândia, 2003.
- SOUZA, Leandro B. S.; SCHULZ, H. E.; SOUZA, Leonardo B. S. Criação do software FluidoSSS para simulação numérica de escoamentos incompressíveis. *Revista de Iniciação Científica*, n. 9, p. 51-54, 2010a.
- SOUZA, Leandro B. S.; SOUZA, Leonardo B. S.; SCHULZ, Harry E.. Software livre de cfd em computação paralela para simulação 2d de escoamentos turbulentos confinados. In: SIMPÓSIO INTERNACIONAL DE INICIAÇÃO CIENTÍFICA DA USP- CIÊNCIAS EXATAS E ENGENHARIAS, 18., 2010, São Paulo. *Anais...* São Paulo: Universidade de São Paulo, 2010b. CD-ROM.
- VERSTEEG, H. K.; MALALASEKERA, W. *An introduction to computational fluid dynamics: the finite volume method*. London: Longman Scientific & Technical, 1995.
- VORST, H. A. Bi-CGStab: a fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems. *Journal of Scientific and Statistical Computing*, v. 13, n. 2, p. 631-644, 1992.

Sobre os autores

Lauro Cássio Martins de Paula

Graduação em Bacharelado em Ciência da Computação, com ênfase em Matemática Computacional, pela Pontifícia Universidade Católica de Goiás (2012) e Mestrado em Ciência da Computação pela Universidade Federal de Goiás - UFG (2014). Atualmente está cursando Doutorado em Ciência da Computação pela Universidade Federal de Mato Grosso do Sul em associação com a UFG.

Leandro Barra Santana de Souza

Graduado em Ciência da Computação e Bacharelado em Administração de Empresas. É especialista em Educação, Comunicação e Tecnologias em Interfaces Digitais. Atualmente é mestrando em Ciências da Computação na área de redes, Universidade Federal de Minas Gerais.

Leonardo Barra Santana de Souza

Graduado em Engenharia Civil pela Universidade Federal de Uberlândia, Mestrado em Engenharia Civil (Hidráulica e Saneamento) e Doutorado em Engenharia Civil (Hidráulica e Saneamento) pela Universidade de São Paulo. É Professor Adjunto IV da Escola de Engenharia Civil da Universidade Federal de Goiás (UFG). Diretor do Centro Integrado de Aprendizagem em Rede (CIAR - órgão suplementar da Reitoria, de administração e apoio à Educação a Distância) e Coordenador na UFG do Programa Universidade Aberta do Brasil, da CAPES.

Harry Edmar Schulz

Graduado (1982) em Engenharia Civil na Fundação Universidade Regional de Blumenau, mestrado (1985) e doutorado (1990) em Engenharia Hidráulica e Saneamento na Universidade de São Paulo. Realizou dois projetos de pesquisa no exterior em nível pós-doutorado (1992-1993 e 1998-1999) no Institut für Hydromechanik da Universität Karlsruhe, Alemanha (FAPESP) e mais um projeto (2007-2008) no Saint Anthony Falls Laboratory, University of Minnesota, USA (CAPES). Diretor do Centro de Recursos Hídricos e Ecologia Aplicada/SHS/EESC/ USP (1999-2001). Chefe do Departamento de Hidráulica e Saneamento/EESC/USP (2001- 2005). Professor titular da Universidade de São Paulo.