

COMPARAÇÕES ENTRE OS MÉTODOS GMRES e LCD NA IMPLEMENTAÇÃO DO MÉTODO DE NEWTON INEXATO EM PROBLEMAS DE DIFERENÇAS FINITAS

Ellen C. Valentim
Lorena M. Pessoa
Bruno Z. Melotti
Andréa M.P. Valli
Lucia Catabriga

Departamento de Informática (DI)
Universidade Federal do Espírito Santo (UFES)
Av. Fernando Ferrari, s/n, Goiabeiras, 29060-900, Vitória, ES
avalli,luciac@inf.ufes.br

Resumo. Neste trabalho é discutida a solução de problemas não-lineares pelo método de Newton inexato considerando a solução do sistema linear resultante em cada iteração pelo método dos resíduos mínimos generalizados (GMRES) e pelo método das direções conjugadas à esquerda (LCD). São desenvolvidos estudos comparativos da eficiência computacional dos dois métodos iterativos lineares citados no contexto do método de Newton inexato para um problema de transferência de calor com condutividade quadrática e um problema de convecção e difusão. São apresentados também duas estratégias de escolha do critério de parada a ser adotado em cada iteração não-linear.

Palavras chave: Método de Newton Inexato, Método do Resíduo Mínimo Generalizado (GMRES), Método das Direções Conjugadas à Esquerda (LCD).

1. Introdução

O uso de técnicas numéricas para a solução de problemas complexos de engenharia é hoje uma realidade, principalmente devido ao desenvolvimento de computadores de alta velocidade e de grande capacidade de armazenamento. A modernização dos computadores associada ao desenvolvimento de novas técnicas numéricas tem permitido, por exemplo, a solução de fenômenos complexos de mecânica dos fluidos envolvendo termos difusivos, convectivos e não-lineares (Fortuna, 2000). Em geral, os processos de solução recaem na necessidade de resolver sistemas de equações algébricas lineares e não-lineares. O método de Newton é uma excelente alternativa de solução para sistemas não-lineares pois possui ordem de convergência quadrática, requerendo, porém, a solução de um sistema linear em cada iteração, o que o torna computacionalmente caro (Kelley, 1995).

A escolha do método a ser adotado na solução do sistema linear é muito importante, uma vez que esse é o passo do algoritmo que requer maior esforço computacional. Para problemas de grande porte os métodos diretos são proibitivos, pois em geral as matrizes originadas dos processos de discretização são esparsas. Uma alternativa bastante utilizada são os métodos iterativos não-estacionários baseados nos subespaços de Krylov (Saad, 1996). O sucesso destes métodos não-estacionários está na implementação eficiente do produto matriz-vetor e na escolha de um preconditionador adequado.

Quando o método das diferenças finitas é utilizado para obter a equação não-linear, as matrizes envolvidas possuem uma esparsidade conhecida, apresentando apenas 5 diagonais contendo coeficientes não nulos. Em geral essa matriz é denominada matriz pentadiagonal (Saad, 1996). Na implementação dos métodos iterativos não-estacionários deve ser considerada uma forma de armazenamento que otimize ao máximo as principais operações envolvidas nos algoritmos, que são: produto matriz-vetor, produto interno de dois vetores e soma de um vetor com o múltiplo de outro vetor.

Recentemente, Yuan et al., 2003, introduziram um novo algoritmo para resolver sistemas lineares não-simétricos, o método das direções conjugadas à esquerda (LCD). Este método é baseado no conceito de vetores conjugados à esquerda e à direita para matrizes não-simétricas e não-singulares. Os experimentos iniciais realizados em uma implementação em MatLab mostrou que o LCD possui taxas atrativas de convergência quando comparado com outros algoritmos não-estacionários, tais como: Bi-CGSTAB, QMR e GMRES (Saad, 1996). Catabriga et al., 2004 e Silva et al., 2002, apresentaram alguns experimentos em formulações baseadas no método dos elementos finitos. Os testes realizados por Catabriga et al., 2004, demonstram que o método

LCD converge em menos iterações que os demais, porém necessita de um tempo maior para a convergência, pois cada iteração do método LCD é mais cara computacionalmente.

Neste trabalho são apresentados algoritmos de solução de problemas não-lineares discretizados por diferenças finitas utilizando os métodos LCD e GMRES para resolver os sistemas lineares necessários em cada iteração não-linear. Além disso, são consideradas duas formas de escolher a tolerância para o método iterativo definidas por Papadrakakis, 1993 e Kelley, 1995. Testes comparativos entre os métodos LCD e GMRES, juntamente com as escolhas das tolerâncias, são apresentados. O restante deste artigo é organizado como se segue. Na seção 2 são descritas as principais idéias do método de Newton inexato, incluindo duas formas para a escolha da tolerância dos métodos iterativos. Na seção seguinte são apresentados os métodos iterativos GMRES e LCD. Os resultados para dois problemas bidimensionais simples não-lineares são apresentados na seção 4 e logo após, as principais conclusões dos testes realizados.

2. O método de Newton inexato

Um sistema de equações algébricas não-lineares pode ser escrito como $F(x) = 0$, supondo que F tenha derivadas contínuas em \mathbb{R}^n . A variação da função F com relação a x é representada pela matriz Jacobiana J . O método de Newton determina a sequência de soluções aproximadas dadas por $x_{k+1} = x_k + s_k$, onde o vetor s_k é calculado a partir da solução do sistema linear, $J(x_k)s_k = -F(x_k)$.

A estratégia de resolução dos sistemas de equações não-lineares pelo método de Newton clássico requer a solução de um sistema linear em cada iteração k do método. Por este motivo, problemas de grande porte tornam-se dispendiosos do ponto de vista computacional. No entanto, uma maneira de torná-lo economicamente viável consiste em utilizar métodos iterativos na resolução dos sistemas lineares. Deste procedimento surge o que denotamos por método de Newton inexato, que é uma implementação do método de Newton na qual um solucionador linear iterativo é usado na resolução dos sistemas lineares. Neste trabalho, os métodos LCG e GMRES foram utilizados na resolução dos sistemas lineares originados a partir de sistemas não-lineares.

O cálculo das soluções lineares no início das iterações não-lineares não necessita de grande precisão. Entretanto, esta necessidade aumenta à medida em que as iterações não-lineares aproximam-se da solução final. Portanto, a idéia desta variação do método de Newton, também conhecida por Newton truncado, é minimizar o esforço empregado para a solução dos sistemas de equações lineares quando a direção de busca não-linear estiver longe da solução de equilíbrio (Elias, 2003). Assim, o esforço computacional no método de Newton inexato pode ser controlado por um parâmetro calculado, η_k , que representa a tolerância para o sistema linear dentro da iteração não-linear k .

No presente trabalho são utilizados dois métodos de determinação do critério de parada para os sistemas lineares; o primeiro, sugerido por Papadrakakis, 1993 e o segundo, por Kelley, 1995. A condição para o término das iterações não-lineares é baseada no tamanho do resíduo não-linear relativo, $\|F(x)\|/\|F(x_0)\|$, pois este representa um bom indicador para o tamanho do erro na iteração não-linear. O critério de parada adotado para as iterações não-lineares consiste em uma combinação linear entre o valor absoluto e o valor relativo do resíduo na iteração corrente, $\tau_r r_0 + \tau_a$, sendo $r_0 = \|F(x_0)\|$, ou seja, o resíduo da condição inicial x_0 . A seguir são apresentados os critérios de parada citados.

2.1. Critério de Papadrakakis

O cálculo da tolerância do sistema linear apresentado por Papadrakakis, 1993 sugere que:

$$\eta = \min\{\eta_{max}, (\frac{\|F(x)\|}{r_0})^t\}, \quad (1)$$

sendo η_{max} e $0 < t < 1$, parâmetros dependentes do problema a ser resolvido. Desta forma, o algoritmo do método de Newton inexato para a resolução de sistemas de equações não-lineares utilizando o critério de Papadrakakis é o descrito por:

Algoritmo 1 - Newton inexato - Papadrakakis

1. *Dados* $F, J, x_0, \tau_a, \tau_r, \eta_{max}, t$
2. $r_0 = \|F(x_0)\|$
3. *Enquanto* $\|F(x)\| > \tau_r r_0 + \tau_a$ *faça*
 - 3.1. $\eta = \min\{\eta_{max}, (\frac{\|F(x)\|}{r_0})^t\}$
 - 3.2. *Calcular* $J(x)$
 - 3.3. *Resolver* $J(x)s = -F(x)$ *com tolerância* η
 - 3.4. $x = x + s$
 - 3.5. *Avaliar* $F(x)$

2.2. Critério de Kelley

Segundo Kelley, 1995, a medida do grau de aproximação com que a iteração não-linear se aproxima da solução é dada por:

$$\eta_k^A = \gamma \frac{\|F(x_k)\|^2}{\|F(x_{k-1})\|^2}, \quad (2)$$

onde $\gamma \in (0, 1]$ é um parâmetro dependente do problema a ser resolvido. Se η_k^A for uniformemente limitado longe de 1, fazendo $\eta_k = \eta_k^A$ para $k > 0$ obtém-se convergência quadrática. Desta forma, a tolerância utilizada na solução do sistema linear é eficientemente explorada. Para determinar o valor da tolerância na primeira iteração e nas iterações subsequentes, mantendo o limite da escolha sempre longe de 1, tem-se:

$$\eta_k^B = \begin{cases} \eta_{max}, & k = 0 \\ \min(\eta_{max}, \eta_k^A), & k > 0 \end{cases}. \quad (3)$$

O parâmetro η_{max} é o limite superior da sequência $\eta_k, k = 1, 2, \dots$. Os valores sugeridos em Kelley, 1995, são $\gamma = 0.9$ e $\eta_{max} = 0.9999$. Entretanto, é possível que η_k^B seja pequeno demais para uma ou mais iterações enquanto a iteração não-linear ainda esteja longe da solução do problema. Para evitar que isto ocorra, Kelley, 1995, sugere que se η_{k-1} for suficientemente grande, o valor seguinte, η_k , não pode decrescer muito mais que um fator de η_{k-1} , e isto pode ser reescrito da seguinte forma:

$$\eta_k^C = \begin{cases} \eta_{max}, & k = 0 \\ \min(\eta_{max}, \eta_k^A), & k > 0 \text{ e } \gamma \eta_{k-1}^2 < 0.1, \\ \min(\eta_{max}, \max(\eta_k^A, \gamma \eta_{k-1}^2)) & k > 0 \text{ e } \gamma \eta_{k-1}^2 > 0.1 \end{cases}, \quad (4)$$

sendo 0.1 uma constante arbitrária. O algoritmo de Newton inexato utilizando o critério definido por Kelley, 1995, é descrito a seguir.

Algoritmo 2 - Newton inexato - Kelley

1. *Dados* $F, J, x_0, \tau_a, \tau_r, \eta_{max}, \gamma$
2. $r_0 = \|F(x_0)\|$
3. $iter = 0$
4. *Enquanto* $\|F(x)\| > \tau_r r_0 + \tau_a$ *faça*
 - 4.1. *Se* ($iter = 1$) *então*

$$\eta_{iter} = \eta_{max}$$
Senão Se ($\gamma \eta_{iter-1}^2 < 0.1$) *então*

$$\eta_{iter} = \min(\eta_{max}, \gamma \frac{\|F(x_{iter})\|^2}{\|F(x_{iter-1})\|^2})$$
Senão

$$\eta_{iter} = \min(\eta_{max}, \max(\gamma \frac{\|F(x_{iter})\|^2}{\|F(x_{iter-1})\|^2}, \gamma \eta_{iter-1}^2))$$
 - 4.2. *Calcular* $J(x_{iter})$
 - 4.3. *Resolver* $J(x_{iter})s = -F(x_{iter})$ *com tolerância* η_{iter}
 - 4.4. $x_{iter+1} = x_{iter} + s$
 - 4.5. *Avaliar* $F(x_{iter+1})$
 - 4.6. $iter = iter + 1$

3. Solução do sistema linear

Neste trabalho serão comparados, em termos de eficiência computacional, dois métodos iterativos para solução de sistemas de equações lineares, esparsos e não-simétricos: o método do resíduo mínimo generalizado (GMRES) e o método das direções conjugadas à esquerda (LCD), apresentados a seguir.

3.1. Método Iterativo GMRES

O método iterativo do resíduo mínimo generalizado (GMRES), desenvolvido por Saad and Schultz, 1986, tem como principal objetivo minimizar a norma do resíduo do sistema. Considerando uma solução inicial x_0 , uma solução aproximada é obtida através de $x_0 + z$, sendo z um membro do subespaço de Krylov $K_k = \text{span}\{r_0, Ar, A^2r_0, \dots, A^{k-1}r_0\}$, onde k é a dimensão do subespaço e $r_0 = b - Ax_0$. O método GMRES procura encontrar um z tal que a norma do resíduo seja mínima, isto é, $x_0 + z$ é solução do sistema se $\|b - A(x_0 + z)\|$ é mínima.

O método GMRES possui como principal característica a construção de uma base ortogonal no subespaço de Krylov obtida através do processo de Gram-Schmidt Modificado sucedido por um processo de rotações no

plano, denominado algoritmo QR (Saad and Schultz, 1986). O algoritmo do GMRES pode ser impraticável computacionalmente, pois a cada iteração os produtos matriz-vetor precisam ser armazenados, o que pode ocasionar problemas de falta de memória quando for necessário um grande número de iterações para resolver o sistema linear. Uma alternativa para contornar esse problema é utilizar um algoritmo com restart limitando a dimensão k do subespaço introduzido por Shakib et al., 1989. O Algoritmo 3 apresenta os principais comandos dessa implementação.

Algoritmo 3 - GMRES(k)

1. Dados A , b , l_{max} , k e ϵ_{tol}
2. $\epsilon = \epsilon_{tol} \|b\|$
3. $x = 0$
4. Para $l = 1, \dots, l_{max}$ faça
 - 4.1. $u_1 = b - Ax$
 - 4.2. $\bar{e}_1 = \|u_1\|$
 - 4.3. $u_1 = \frac{u_1}{\|u_1\|}$
 - 4.4. Para $i = 1, \dots, k$ faça
 - $u_{i+1} = Au_i$
 - Para $j = 1, \dots, i$ faça
 - $\beta_{i+1,j} = u_{i+1}^T u_j$
 - $u_{i+1} = u_{i+1} - \beta_{i+1,j} u_j$
 - $\bar{h}^{(i)} = \{\beta_{i+1,1}, \dots, \beta_{i+1,i}, \|u_{i+1}\|\}^T$
 - $u_{i+1} = \frac{u_{i+1}}{\|u_{i+1}\|}$
 - Para $j = 1, \dots, i-1$ faça
 - $$\begin{bmatrix} \bar{h}_j^{(i)} \\ \bar{h}_{j+1}^{(i)} \end{bmatrix} = \begin{bmatrix} c_j & s_j \\ -s_j & c_j \end{bmatrix} \begin{bmatrix} \bar{h}_j^{(i)} \\ \bar{h}_{j+1}^{(i)} \end{bmatrix}$$
 - $r = \sqrt{(\bar{h}_i^{(i)})^2 + (\bar{h}_{i+1}^{(i)})^2}$
 - $c_i = \frac{\bar{h}_i^{(i)}}{r}$
 - $s_i = \frac{\bar{h}_{i+1}^{(i)}}{r}$
 - $\bar{h}_i^{(i)} = r$
 - $\bar{h}_{i+1}^{(i)} = 0$
 - $\bar{e}_{i+1} = -s_i \bar{e}_i$
 - $\bar{e}_i = c_i \bar{e}_i$
 - Se $\bar{e}_{i+1} \leq \epsilon$ finalizar o loop i , x é solução aproximada.
 - 4.5. Encontrar y tal que:

$$\begin{bmatrix} \bar{h}_1^{(1)} & \dots & \bar{h}_1^{(i-1)} & \bar{h}_1^{(i)} \\ 0 & \ddots & \vdots & \vdots \\ \vdots & \ddots & \bar{h}_{i-1}^{(i-1)} & \bar{h}_{i-1}^{(i)} \\ 0 & \dots & 0 & \bar{h}_i^{(i)} \end{bmatrix} \begin{bmatrix} y_1 \\ \vdots \\ y_{i-1} \\ y_i \end{bmatrix} = \begin{bmatrix} \bar{e}_1 \\ \vdots \\ \bar{e}_{i-1} \\ \bar{e}_i \end{bmatrix}$$
 - 4.6. $x = x + \sum_{j=1}^i y_j u_j$
 - 4.7. Se $\bar{e}_{i+1} \leq \epsilon$ finalizar o loop l , x é solução aproximada.

onde l_{max} é o número máximo de iterações, ϵ_{tol} é a tolerância considerada e k é a dimensão do subespaço de Krylov considerado para compor a solução.

3.2. Método iterativo LCD

O método iterativo das direções conjugadas à esquerda (LCD), recentemente apresentado por Yuan et al., 2003, tem como objetivo encontrar a solução do sistema em um conjunto ortogonal de direções de busca cujos vetores tem a característica de serem conjugados à esquerda da matriz A . Os vetores p_1, p_2, \dots, p_n são chamados direções conjugadas à esquerda de uma matriz real não-singular $A_{n \times n}$ se $p_i^T A p_j = 0$ para $i < j$ e $p_i^T A p_j \neq 0$ para $i = j$. É possível provar que os vetores conjugados à esquerda de A são linearmente independentes. Dessa forma, considerando uma aproximação inicial x_0 , a solução do sistema $Ax = b$ pode ser representada por $x = x_0 + \sum_{i=1}^n \alpha_i p_i$, onde os α_i 's são determinados unicamente por p_1, p_2, \dots, p_n e x_0 .

O processo de obtenção das direções conjugadas à esquerda é apresentado em Yuan et al., 2003. O Algoritmo 4, inicialmente apresentado por Catabriga et al., 2004, representa o método LCD numa estrutura com restart

análoga ao algoritmo GMRES(k) descrito pelo Algoritmo 3, limitando a dimensão k do conjunto de direções de busca da solução e reinicializando o processo iterativo a partir de uma determinada iteração. Essa implementação com restart evita que todos os produtos matriz-vetor e matriz-transposta-vetor sejam efetuados e que todas as direções conjugadas à esquerda sejam armazenadas.

Algoritmo 4 - LCD(k)

1. Dados x_0, A, b, l_{max}, k e ϵ_{tol}
2. $r_0 = b - Ax_0$
3. $\epsilon = \epsilon_{tol} \|r\|$
4. Escolha p_1 tal que $p_1^T A p_1 \neq 0$
5. Para $l = 1, \dots, l_{max}$ faça
 - 5.1. Para $i = 1, \dots, k$ faça
 - 5.1.1. $q_i = A^T p_i$
 - $\alpha_i = \frac{p_i^T r_{i-1}}{q_i^T p_i}$
 - $x_i = x_{i-1} + \alpha_i p_i$
 - $r_i = r_{i-1} - \alpha_i A p_i$
 - 5.1.2. Se $\|r_i\| < \epsilon$ então finalize o loop l , x_i solução aproximada.
 - 5.1.3. $p_{i+1} = r_i$
 - Para $j = 1, \dots, i$ faça
 - $\beta_j = -\frac{q_j^T p_{i+1}}{q_j^T p_j}$
 - $p_{i+1} = p_{i+1} + \beta_j p_j$
 - 5.2. Escolha o novo p_1 tal que $p_1^T A p_1 \neq 0$

onde l_{max} é o número máximo de iterações, ϵ_{tol} é a tolerância considerada e k é número de vetores conjugados à esquerda considerados para o restart. É importante observar que o custo computacional de cada iteração do método LCD é maior que o custo computacional do método GMRES, já que além do produto matriz-vetor envolve um produto matriz-transposta-vetor.

4. Experimentos numéricos

Nesta seção é comparado o desempenho do método GMRES com o método LCD utilizados no algoritmo de Newton inexato na solução de dois problemas não-lineares. Para cada método são empregadas as técnicas de tolerância fixa e de tolerância variável, com os critérios de Papadrakakis, 1993 e de Kelley, 1995. O primeiro experimento apresenta a solução por diferenças finitas centrais de uma problema de transferência de calor, cuja condutividade varia quadraticamente com a temperatura. O segundo experimento apresenta a solução, também por diferenças finitas centrais, de um problema de convecção e difusão não-linear. Nos dois problemas as tolerâncias utilizadas no método de Newton inexato são $\tau_a = 10^{-9}$ e $\tau_r = 10^{-12}$. Os parâmetros para o critério de Papadrakakis são $\eta_{max} = 0.999$ e $t = 0.5$, e para o critério de Kelley são $\eta_{max} = 0.999$ e $\gamma = 0.9$. A tolerância fixa considerada em todos os testes foi de 10^{-5} .

4.1. Transferência de calor com condutividade quadrática

O problema de transferência de calor utilizado para realizar experimentos satisfaz a equação:

$$-\nabla \cdot (k(u) \nabla u) = 0 \quad \text{em } (0, 1) \times (0, 1) \quad (5)$$

onde $k(u) = 0.001(1 + 0.01u + 0.0002u^2)$ e condições de contorno iguais a $u(x, 0) = u(1, y) = 10$ e $u(x, 1) = u(0, y) = 100$. As Fig. 1(a) e 1(b) mostram, respectivamente, a solução numérica obtida para o problema de transferência de calor para os métodos GMRES(10) e LCD(10), considerando o domínio discretizado por uma malha com 64 divisões iguais nos eixos x e y , e o critério Papadrakakis. Os dois métodos apresentam a solução esperada e são virtualmente iguais. A Tab. 1 apresenta o número total de iterações dos métodos GMRES e LCD e o tempo de CPU para a convergência do método de Newton inexato considerando uma malha quadrada com 512 divisões iguais nos eixos x e y , para diferentes números de vetores para o restart.

Neste exemplo o método LCD obteve desempenho melhor do que o método GMRES, tanto para o critério de tolerância fixa quanto para os critérios de tolerância variável. Além disto, o aumento de vetores para o restart melhora o desempenho do GMRES e piora o desempenho do LCD, mas ainda este apresenta um desempenho melhor que o obtido para o método GMRES. Neste testes o tempo de CPU, necessário para a convergência do método LCD, foi consideravelmente menor que no método GMRES, sendo o menor tempo obtido quando foi utilizado 5 vetores para o restart e o critério de Papadrakakis. A Fig. 2 mostra o comportamento do resíduo não-linear do método de Newton inexato para o problema de transferência de calor considerando tolerância fixa,

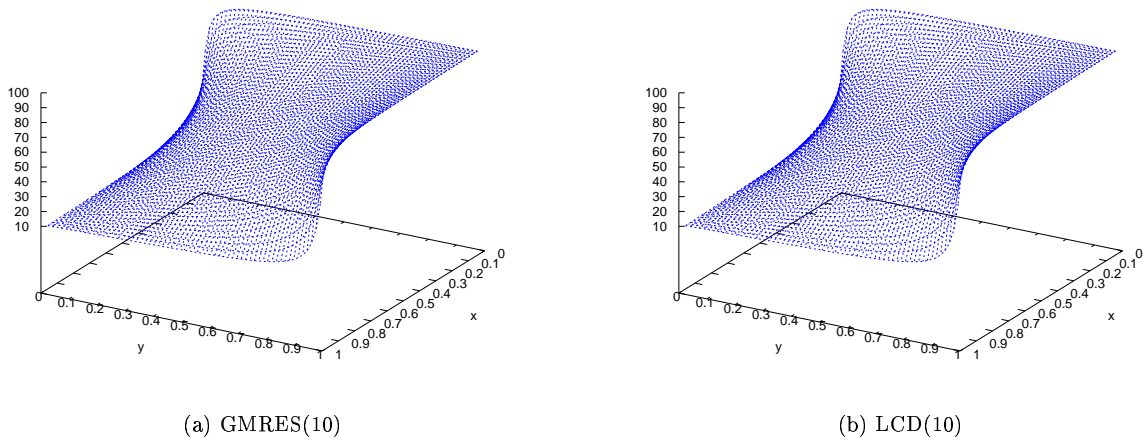


Figura 1: Problema de transferência de calor - Solução aproximada considerando uma malha com 64 divisões iguais nos eixos x e y , 10 vetores para o restart e o critério Papadrakakis.

Tolerância (5 vetores)	GMRES(5)		LCD(5)	
	Iterações	Tempo(min)	Iterações	Tempo(min)
Fixa (10^{-5})	42415	456.42	2519	32.22
Papadrakakis	30526	307.38	1803	24.61
Kelley	46245	488.90	2322	31.50
Tolerância (10 vetores)	GMRES(10)		LCD(10)	
	Iterações	Tempo(min)	Iterações	Tempo(min)
Fixa (10^{-5})	22364	411.98	2548	68.31
Papadrakakis	15608	300.25	1775	47.65
Kelley	21752	419.31	1761	47.13
Tolerância (20 vetores)	GMRES(20)		LCD(20)	
	Iterações	Tempo(min)	Iterações	Tempo(min)
Fixa (10^{-5})	11619	338.93	2645	88.23
Papadrakakis	7511	231.24	1783	63.10
Kelley	6752	203.94	1774	60.47
Tolerância (40 vetores)	GMRES(40)		LCD(40)	
	Iterações	Tempo(min)	Iterações	Tempo(min)
Fixa (10^{-5})	5745	318.16	2443	157.95
Papadrakakis	3429	189.84	1878	118.72
Kelley	3891	211.86	1728	107.79

Tabela 1: Problema de transferência de calor - Desempenho dos algoritmos GMRES e LCD no método de Newton inexato considerando uma malha com 512 divisões iguais nos eixos x e y .

o critério Papadrakis e critério Kelley, para os métodos GMRES(10) e LCD(10) utilizando uma malha com 512 divisões iguais nos eixos x e y . Observa-se na Fig. 2 que o resíduo não-linear cai mais rapidamente quando é utilizado o critério de tolerância fixa. No entanto, os menores tempo de CPU, para os métodos GMRES e LCD, aconteceram quando foi utilizado algum dos critérios de tolerância variável.

As técnicas de tolerância variável em geral apresentam melhores desempenhos do que a técnica de tolerância fixa e possuem uma influência significativa no número final de iterações lineares obtidas. Na maioria dos casos estudados neste exemplo, o critério de Kelley mostrou-se mais eficiente em reduzir o tempo de CPU quando foi utilizado o método LCD, apesar do menor tempo de CPU acontecer quando foi utilizado 5 vetores e o critério de Papadrakakis. Já no método GMRES, o critério de Papadrakakis teve um desempenho melhor na maioria dos casos, com o menor tempo obtido quando foi utilizado 40 vetores. As Figuras 3(a) e 3(b) mostram, respectivamente, o número de iterações lineares em cada iteração não-linear usando os métodos GMRES(10) e LCD(10), considerando tolerância fixa e os critérios de Papadrakakis e Kelley. Pode-se observar que o número de iterações não-lineares quando foi utilizado tolerância fixa é muito menor do que nos casos de tolerância variável.

No entanto, o número de iterações lineares na maioria dos passos não-lineares é superior quando foi utilizado tolerância fixa. O critério de Kelley no método GMRES exigiu um número muito grande de iterações no último passo não-linear, o que resultou em um tempo de CPU maior. Apesar de possuir este mesmo comportamento no método LCD, este critério conseguiu um desempenho melhor.

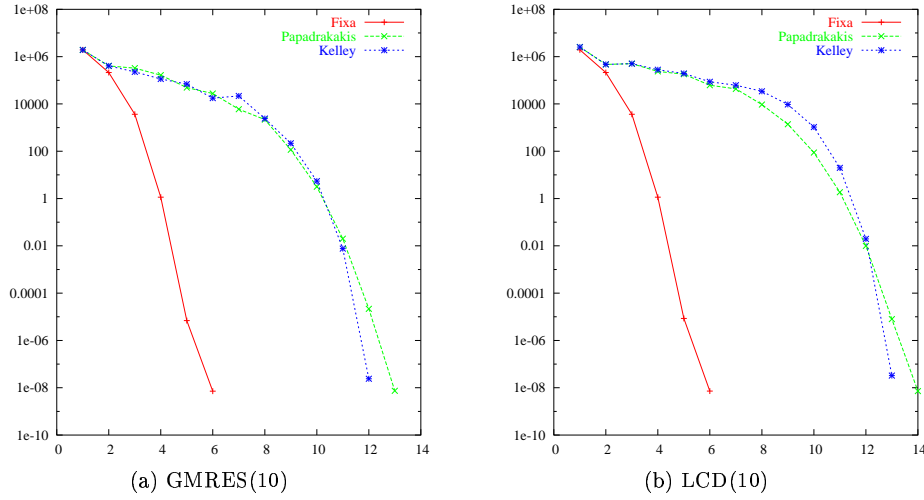


Figura 2: Problema de transferência de calor - Iterações não-lineares $\times \|F(u^k)\|$ - Malha com 512 divisões iguais nos eixos x e y e 10 vetores para o restart.

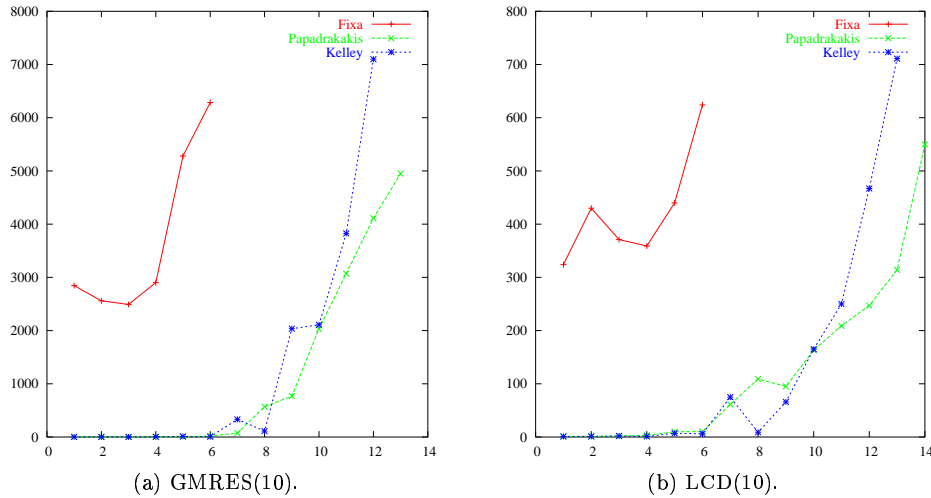


Figura 3: Problema de transferência de calor - Número de iterações lineares usando diferentes critério de tolerância - Malha com 512 divisões iguais nos eixos x e y e 10 vetores para o restart.

Em relação ao número de vetores utilizados, observa-se que o método LCD convergiu em um número menor de iterações quando passou-se de 5 para 10 vetores. No entanto, o tempo de CPU aumenta. No método GMRES, o tempo de CPU diminuiu quando aumentou-se o número de vetores. O melhor desempenho do método GMRES aconteceu quando foi utilizado 40 vetores e o critério de Papadarakakis.

4.2. Problema de convecção e difusão não-linear

O problema de convecção e difusão utilizado para realizar experimentos satisfaz a equação:

$$-\nabla^2 u + 20u \nabla \cdot u = g \quad \text{em } (0, 1) \times (0, 1), \quad (6)$$

sendo g tal que a solução exata é dada por $u(x, y) = 10xy(1-x)(1-y)e^{x^{4.5}}$ e com condições de contorno $u(x, 0) = u(1, y) = u(x, 1) = u(0, y) = 0$. As Fig. 4(a) e 4(b) mostram, respectivamente, a solução numérica

obtida para os métodos GMRES(10) e LCD(10), considerando o domínio discretizado por uma malha com 64 divisões iguais nos eixos x e y , e o critério Papadrakakis. Os dois métodos apresentam a solução esperada e são virtualmente iguais. A Tab. 2 apresenta o número total de iterações dos métodos GMRES e LCD e o tempo de CPU para a convergência do método de Newton inexato utilizando uma malha quadrada com 512 divisões iguais nos eixos x e y , para diferentes números de vetores para o restart.

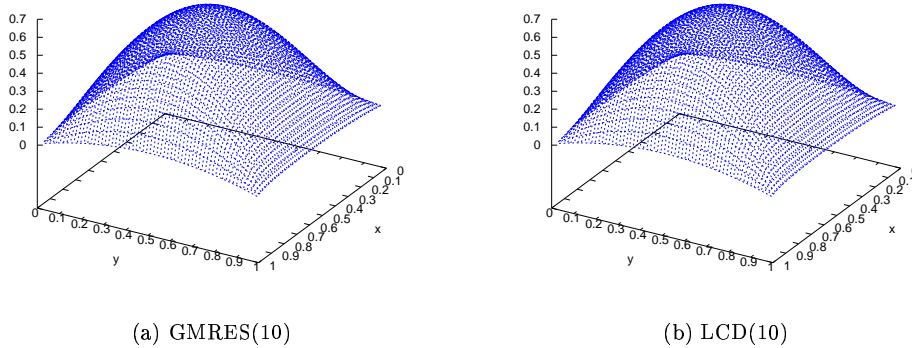


Figura 4: Problema de convecção e difusão - Solução aproximada considerando uma malha com 64 divisões iguais nos eixos x e y , 10 vetores para o restart e o critério Papadrakakis.

Tolerância (5 vetores)	GMRES(5)		LCD(5)	
	Iterações	Tempo(min)	Iterações	Tempo(min)
Fixa (10^{-5})	35834	372.68	8835	113.88
Papadrakakis	7063	73.48	4214	57.97
Kelley	6214	64.91	12614	172.07
Tolerância (10 vetores)	GMRES(10)		LCD(10)	
	Iterações	Tempo(min)	Iterações	Tempo(min)
Fixa (10^{-5})	17903	318.07	4907	132.84
Papadrakakis	3144	55.86	1754	46.85
Kelley	3444	64.31	3243	87.29
Tolerância (20 vetores)	GMRES(20)		LCD(20)	
	Iterações	Tempo(min)	Iterações	Tempo(min)
Fixa (10^{-5})	9098	276.92	12809	463.17
Papadrakakis	1721	50.90	1414	49.85
Kelley	2450	72.15	2132	76.46
Tolerância (40 vetores)	GMRES(40)		LCD(40)	
	Iterações	Tempo(min)	Iterações	Tempo(min)
Fixa (10^{-5})	4495	245.62	2962	178.59
Papadrakakis	1319	61.67	1481	90.64
Kelley	1650	85.15	1678	106.14

Tabela 2: Problema de convecção e difusão - Desempenho dos algoritmos GMRES e LCD no método de Newton inexato considerando uma malha com 512 divisões iguais nos eixos x e y .

Neste segundo exemplo observa-se algumas mudanças no comportamento dos dois métodos, tanto em relação aos critérios utilizados quanto em relação ao número de vetores para o restart. Para o método LCD, o critério de Papadrakakis apresentou melhor desempenho que o de Kelley na redução total do número de iterações lineares. O melhor desempenho aconteceu com o critério de Papadrakakis e 10 vetores para o restart. Para o método GMRES, o critério de Papadrakakis também representou o melhor desempenho, fornecendo o menor tempo quando foi utilizado 20 vetores para o restart. Neste exemplo, o desempenho do método LCD foi um pouco melhor que o método GMRES. No entanto, como esta eficiência não foi significativa, pode-se dizer que os dois métodos são competitivos. As Fig. 5(a) e 5(b) mostram o comportamento do resíduo não-linear do método de Newton inexato e as Fig. 6(a) e 6(b) mostram o número de iterações lineares em cada iteração não-linear, usando os métodos GMRES(10) e LCD(10). Neste exemplo, o critério de Papadrakakis apresentou o melhor

desempenho nos dois métodos, apesar de ter decaimento mais lento para o resíduo não-linear. Isto foi possível devido ao baixo número de iterações lineares em cada passo não-linear.

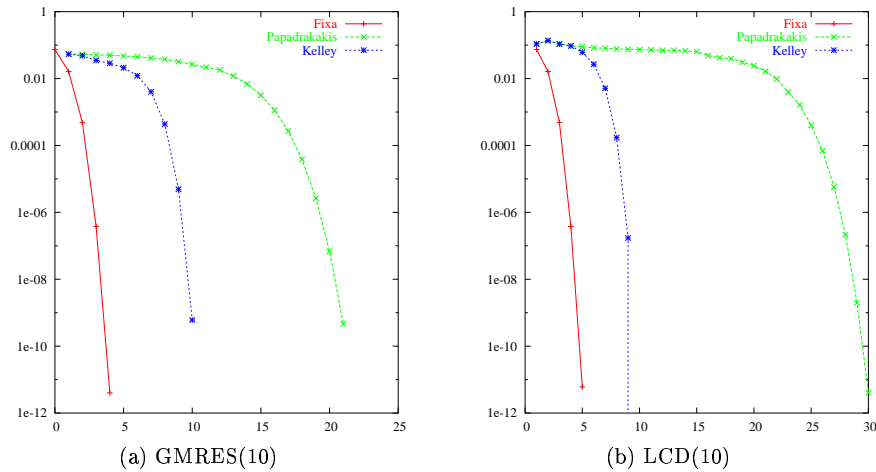


Figura 5: Problema de convecção e difusão - Iterações não-lineares $\times \|F(u^k)\|$ - Malha com 512 divisões iguais nos eixos x e y e 10 vetores para o restart.

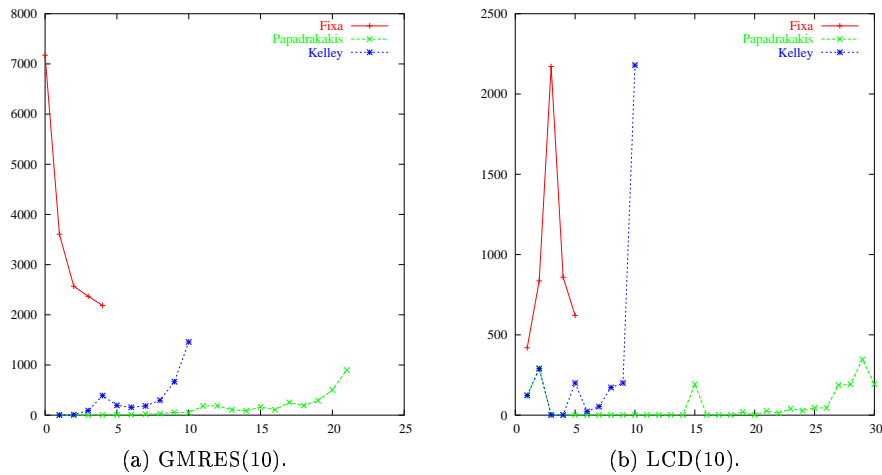


Figura 6: Problema de convecção e difusão - Número de iterações lineares usando diferentes critério de tolerância - Malha com 512 divisões iguais nos eixos x e y e 10 vetores para o restart.

5. Conclusões

Neste artigo foi apresentado um estudo comparativo entre os métodos LCD e GMRES quando utilizados na resolução dos sistemas lineares presentes em cada iteração não-linear do método de Newton inexato e na discretização por diferenças finitas centrais de dois problemas: transferência de calor com condutividade quadrática e convecção e difusão não-linear. São comparados também a eficiência de dois critérios de parada para os sistemas lineares: o critério sugerido por Kelley e o critério sugerido por Papadarakakis. O método LCD teve um desempenho melhor nos dois exemplos estudados. No primeiro caso, o método LCD resolveu o problema em 24.61 (min) e o método GMRES em 189.84 (min) quando são comparados os melhores resultados obtidos nos dois métodos. No segundo caso, os métodos foram comparativos em termos de eficiência computacional, com 46.85 (min) para o método LCD e 50.90 (min) para o método GMRES. Independente do método e do número de vetores para o restart utilizados, observa-se que as técnicas de tolerância variável reduzem tanto o número de iterações lineares na resolução dos problemas implementados quanto o tempo de execução, sendo as melhores soluções obtidas utilizando o critério de Papadarakakis. Além disto, a implementação do algoritmo LCD utilizando a técnica do restart é uma opção bastante eficiente na redução do número de produtos matriz-vetor

e matriz-transporta-vetor, assim como na redução do número de direções conjugadas à esquerda que necessitam ser armazenadas. Mais testes e uma nova implementação do método LCD que não necessita de dois produtos matriz-vetor, apresentado por Dai and Yuan, 2004, serão estudados e sua eficiência comparada com implementações anteriores.

6. Agradecimentos

Os autores agradecem o apoio do CNPq e da UFES, que proporcionaram a oportunidade de participação dos três primeiros no programa PIBIC/PIVIC. Este trabalho foi realizado também com o apoio parcial da CAPES, dentro da cooperação internacional CAPES-Universidade do Texas, Austin, EUA.

7. Referências

- Catabriga, L., Coutinho, A. L. G. A., and Franca, L. P., 2004, Evaluating the LCD algorithm for solving linear systems of equations arising from implicit SUPG formulation of compressible flows, “International Journal for Numerical Methods in Engineering”, Vol. 60, pp. 1513–1534.
- Dai, Y. and Yuan, J. Y., 2004, Study on semi-conjugate direction methods for non-symmetric systems, “International Journal for Numerical Methods in Engineering”, Vol. 60, pp. 1383–1399.
- Elias, R. N., 2003, Métodos Tipo-Newton Inexatos para a Solução de Problemas Não-Lineares Resultantes da Formulação SUPG/PSPG das Equações de Navier-Stokes Incompressíveis em Regime Permanente, COPPE/UFRJ - <http://www.nacad.ufrj.br/rnelias>, Dissertação de Mestrado.
- Fortuna, A. O., 2000, “Técnicas Computacionais para Mecânica dos Fluidos - Conceitos Básicos e Aplicações”, Ed. USP, S.Paulo, Brazil.
- Kelley, C., 1995, “Iterative Methods for Linear and Nonlinear Equations”, Siam, Philadelphia, US.
- Papadrakakis, M., 1993, “Solving Large-Scale Problems in Mechanics: The Development and Application of Computational Solution Procedures”, John Wiley & sons, London, England.
- Saad, Y., 1996, “Iterative Methods for Sparse Linear Systems”, PWS Publishing, Boston, US.
- Saad, Y. and Schultz, M. H., 1986, GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems, “SIAM J. Sci. Stat. Comp.”, Vol. 7, pp. 865–869.
- Shakib, F., Hughes, T. J. R., and Johan, Z., 1989, A multi-element group preconditioned GMRES algorithm for nonsymmetric systems arising in finite element analysis, “Comput. Methods Appl. Mech. and Engrg.”, Vol. 65, pp. 415–456.
- Silva, R. S., Raupp, F. M., and Almeida, R. C., 2002, A numerical methodology to solve thermal pollution problems, “9th Brazilian Congress of Thermal Engineering and Sciences - CD-ROM”, pp. 1–12, Caxambu, Minas Gerais, Brazil.
- Yuan, J., Golub, G., Plemmons, R., and Cecílio, W. A. G., 2003, Semiconjugate direction methods for real positive definite systems, BIT (<http://www-sccm.stanford.edu/urap/pub-tech.html/SCCM-02-02>), (accepted).

COMPARISON BETWEEN GMRES AND LCD METHODS IN THE IMPLEMENTATION OF AN INEXACT NEWTON METHOD

Ellen C. Valentim

Lorena M. Pessoa

Bruno Z. Melotti

Andréa M.P. Valli

Lucia Catabriga

Departamento de Informática (DI)

Universidade Federal do Espírito Santo (UFES)

Av. Fernando Ferrari, s/n, Goiabeiras, 29060-900, Vitória, ES

`avalli, luciac@inf.ufes.br`

Abstract. *This work studies the solution of nonlinear systems using an inexact Newton method where the approximate solution of the resultant linear system at each iteration is obtained by the generalized minimum residual method (GMRES) or by the left conjugated direction method (LCD). We use a spatial discretization based on central finite difference formulation of the heat equation and the convection-diffusion equation. Our discussion considers comparison studies between the computational efficiency of the two linear solvers and some issues related to the choice of the forcing term in the inexact Newton method.*

Keywords. *Inexact Newton Method, Generalized Minimum Residual Method (GMRES), Left Conjugated Direction Method (LCD).*