Proceeding Series of the Brazilian Society of Applied and Computational Mathematics, Vol. 5, N. 1, 2017.

Trabalho apresentado no CNMAC, Gramado - RS, 2016.

Proceeding Series of the Brazilian Society of Computational and Applied Mathematics

Método de Precondicionamento CPR em Simulações de Reservatório de Petróleo

João Paulo K. Zanardi[1]
Faculdade de Engenharia, UERJ, Rio de Janeiro, RJ
Luiz Mariano de Carvalho[2]
Instituto de Matemática e Estatística, UERJ, Rio de Janeiro, RJ
Paulo Goldfeld[3]
Instituto de Matemática, UFRJ, Rio de Janeiro, RJ
Michael Souza[4]
Departamento de Estatística e Matemática Aplicada, UFC, Fortaleza, CE

Resumo. Este trabalho apresenta a implementação, em MATLAB, de um esquema de dois estágios do tipo CPR (do inglês *Constrained Pressure Residual*) [5], para a resolução de sistemas lineares de grande porte oriundos de simulações de extração de reservatórios de petróleo. Vamos descrever o método CPR, seus dois estágios e apresentar resultados para matrizes de problemas reais, comparando-os com resultados utilizando precondicionadores clássicos.

Palavras-chave. Precondicionadores, CPR, Sistemas Lineares.

1 Introdução

Nos problemas de simulação de reservatórios temos que repetidamente resolver sistemas de equações lineares com matrizes Jacobianas resultantes da linearização (em geral pelo método de Newton) das equações de balanço e volume. Estes sistemas lineares apresentam diferentes tipos de variáveis como, por exemplo, pressão e saturação, e resolvê-los numericamente de maneira rápida e robusta ainda é um grande desafio na área computacional.

Tradicionalmente a maioria dos solvers atuais utilizam métodos de fatoração incompleta dos fatores LU (ILU) [2] aplicada ao sistema completo ou aplicam o método CPR. O CPR é uma técnica de precondicionamento de dois níveis que privilegia a solução da pressão, reconhecendo a sua natureza elítica e, portanto, a dificuldade de convergir as frequências mais baixas do erro. Particularmente quando combinado com o multigrid, o algoritmo tem recebido muita atenção na literatura e vem sendo aplicado em diversos

[1]jpzanardi@gmail.com
[2]luizmc@gmail.com
[3]goldfeld@ufrj.br
[4]souza.michael@gmail.com

simuladores de nova geração.
Apresentaremos um trabalho incial de estudo e implementação de precondicionador. Na seção 2 apresentaremos a parte teórica do CPR e na seção 3 apresentaremos resultados do CPR, comparando o número de iterações gastas para resolver o sistema com o número de iterações do ILU(0) e ILU(1).

2 O Precondicionador CPR

2.1 Precondicionadores Multi Estágios

Um precondicionador multi estágio é um precondicionador que trata equações e variáveis de acordo com sua natureza e característica do erro a ser tratado. O objetivo é reservar métodos computacionalmente custosos para equações e variáveis que tem erros com alta ordem de grandeza. Isso ocorre muitas vezes em problemas com muitos graus de liberdade, onde a maioria das variáveis apresenta dependência local, enquanto apenas um conjunto pequeno de equações são afetados fortemente de maneira global. Vamos descrever agora a metodologia de um precondicionador multi estágio.
Para resolver o sistema linear

$$Au = f, \tag{1}$$

um preconcionador com n estágios pode ser escrito como

$$u^n - u^0 = \sum_{i=1}^{n} \delta u^i, \tag{2}$$

onde δu_i é a correção da solução u no estágio i, que pode ser escrita como

$$\delta u^i = M_i^{-1} u^{i-1}. \tag{3}$$

Nesta equação, M_i^{-1} é o precondiciondor do i-ésimo estágio, e r^{i-1} é o resíduo ao fim da iteração anterior. A correção total pode então ser vista como:

$$\sum_{i=1}^{n} \delta u^i = M_1^{-1} r^0 + \sum_{i=2}^{n} M_i^{-1} [I - AM_{i-1}^{-1}] r^{i-2} \tag{4}$$

onde

$$r^0 = f - Au^0 \tag{5}$$

e

$$r^i = [I - AM_i^{-1}] r^{i-1} \tag{6}$$

denotam o resíduo na iteração 0 e i, respectivamente. De forma geral um precondicionador pode ser escrito como

$$M_{1,2,\ldots,n}^{-1} = M_n^{-1}[I - AM_{n-1}^{-1}] \ldots [I - AM_1^{-1}] + \ldots + M_2^{-1}[I - AM_1^{-1}] + M_1^{-1} \tag{7}$$

Em particular, um precondicionador de dois estágios pode ser escrito como

$$M_{1,2}^{-1} = M_2^{-1}[I - AM_1^{-1}] + M_1^{-1}. \tag{8}$$

DOI: 10.5540/03.2017.005.01.0188

2.2 Método CPR

Os sistemas lineares oriundos simulações de reservatório de petróleo são um alvo natural para estratégias de precondicionamento multi estágios, devido a natureza elítica da pressão. Neste contexto, um dos mais conhecidos métodos é o CPR, um preconcionador de dois estágios que pode ser descrito para uma determinada matriz A como

$$M_{CPR}^{-1} = M^{-1}[I - AC(W^T AC)^{-1} W^T] + C(W^T AC)^{-1} W^T A^{-1}. \tag{9}$$

Aqui $C(W^T AC)^{-1} W^T$ e M^{-1} são identificados com M_1^{-1} e M_2^{-1} na equação (8). Na equação (9) C é uma matriz bloco diagonal de ordem $(n_{eq} \cdot n_c) \times n_c$, onde n_{eq} e n_c denotam o número de equações e o número de células de A, respectivamente. Considerando a pressão como a primeira variável em cada célula, podemos expressar C como

$$\text{C} = \begin{bmatrix} e_p & & & \\ & e_p & & \\ & & \ddots & \\ & & & e_p \end{bmatrix},$$

$$\text{onde} \quad \text{e}_p = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

Já W^T é uma matriz bloco diagonal $n_c \times (n_{eq}.n_c)$ que pode ser escrita na forma

$$W^T = C^T \cdot DIAG^{-1}(A), \tag{10}$$

neste contexto $DIAG^{-1}(A)$ representa a matriz formada pela inversa dos blocos diagonais de A. Note que W^T é apenas C^T multiplicado por um fatores escalares. Tendo essas matrizes, podemos restringer a matriz incial A à matriz das pressões

$$A_p = W^T AC. \tag{11}$$

Então, no CPR, diremos que o primeiro estágio é a solução do sistema das pressões $(W^T AC)^{-1}$, enquanto o segundo estágio é feito através do precondicionador M^{-1}. Vale ressaltar que este precondicionador será aplicado ao sistema todo.

A aplicação do precondicionador CPR de dois estágios ao sistema completo pode ser descrito nos cinco passos abaixo:

1. Restrinja o resíduo do sistema global ao resíduo das pressões $r_p = W^T r$.

2. Resolva o sistema $x_p = (W^T AC)^{-1} r_p$ com algum solver linear e expanda a solução do sistema das pressões para o sistema completo $C \cdot x_p$ (primeiro estágio).

3. Corrija o resíduo do sistema completo usando a solução do primeiro estágio $r_{corrigido} = r - A(C \cdot x_p)$.

DOI: 10.5540/03.2017.005.01.0188

4. Resolva o sistema completo com o residuo corrigido usando o precondicionador do segundo estágio $M^{-1}r_{corrigido} = M^{-1}(r - A(C \cdot x_p))$.

5. Combine as soluções de ambos os estágios

$$x = M^{-1}(r - A(C \cdot x_p)) + C \cdot x_p. \tag{12}$$

Tradicionalmente as escolhas para a solução do primeiro e segundo estágio são multigrid e ILU, respectivamente.

3 Resultados

O código do CPR foi implementado em MATLAB seguindo [3], a única alteração feita foi a utilização do *backslash* para a resolução do primeiro estágio, uma vez que o MATLAB não possui codificação nativa. O solver linear utilizado foi o GMRES [4], usando precondicionamento pela esquerda, $restart = 30$, a tolerância usada foi 10^{-4}. Os testes foram realizados num notebook com processador i7, frequência de 800 MHz e sistema operacional Ubuntu 12.04. Os problemas testados foram o tradicional SPE01, e cinco matrizes oriundas de problemas reais *black oil*, em todos os casos foram extraídos só a parte do reservatório da matriz. Os dados das matrizes podem ser encontrados na Tabela (1), o resultado comparativo entre o CPR, ILU(0) e ILU(1) podem ser vistos na Tabela (2) e na Figura (1). A Figura (1) mostra com clareza a eficiência do CPR em reduzir o número de iterações do CPR.

Tabela 1: Matrizes Testadas

Matriz	**n**	**nnz**	**BS**
SPE01	1200	19200	4
Caso 1	18364	293824	4
Caso 2	30028	480448	4
Caso 3	141512	2264192	4
Caso 4	38232	936512	4
Caso 5	38232	936512	4

4 Conclusões

O CPR se mostrou apto em reduzir drasticamente o número de iterações dos problemas apresentados, vale ressaltar que as iterações do CPR são computacionalmente mais custosas que as dos precondicionadores normais, mas a diferença viabiliza seu uso. Os testes em MATLAB são suficientes para averiguação da eficiência do método mas não para performance, por isso pretende-se implementar a mesma versão em PETSc [1], essa

DOI: 10.5540/03.2017.005.01.0188

Tabela 2: Resultados Obtidos

Matriz	ILU(0)	ILU(1)	CPR
SPE01	17	7	2
Caso 1	27	23	3
Caso 2	63	48	4
Caso 3	28	11	2
Caso 4	85	49	2
Caso 5	58	30	4

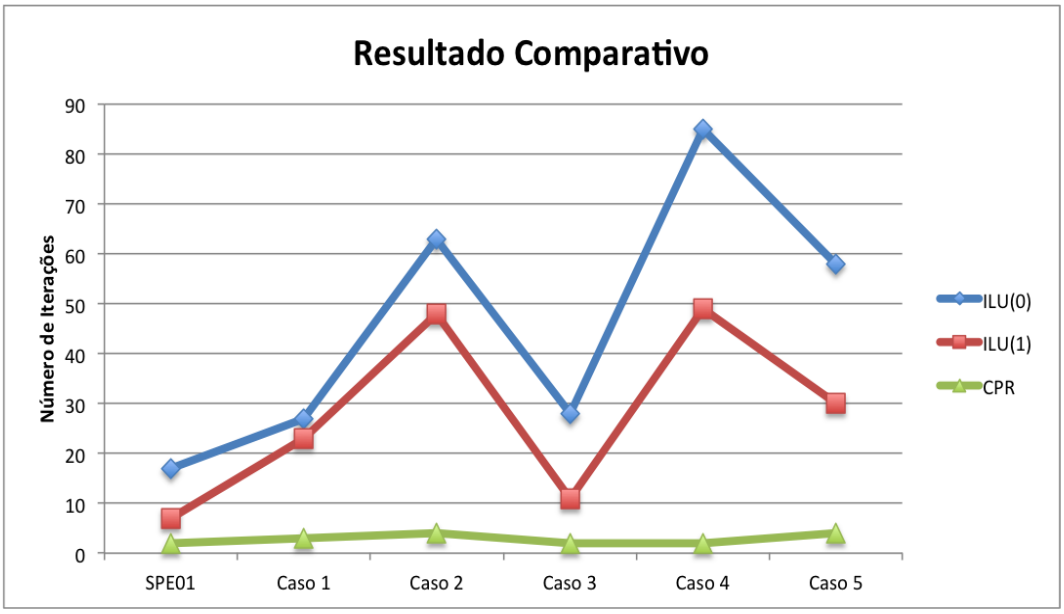

Figura 1: Comparação entre o número de iterações obtidos com os precondicionadores testados.

fiel ao artigo utilizando multigrid no primeiro estágio. Com isso, poderemos avaliar a performance do precondicionador, além disso fazer testes em paralelos.

Agradecimentos

Agradeço ao meu orientador Luiz Mariano de Carvalho Paes por todo apoio à minha pesquisa e a FAPERJ pelo fomento dado à pesquisa.

Referências

[1] S. Balay et al.Petsc users manual revision 3.6. In *Technical report, Argonne National Laboratory (ANL)*, 2015.

[2] T. Chan and V. A. Hank. *Approximate and incomplete factorizations*, Springer, Holanda, 1997.

[3] H. Cao, H. A. Tchelepi, J. Wallis and H. Yardumian. Parallel scalable unstructured CPR-type linear solver for reservoir simulation. In *SPE Annual Technical Conference and Exhibition*, 2005.

[4] Y. Saad and M. H. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. In *SIAM Journal on scientific and statistical computing*, 1986.

[5] J. Wallis, R. P. Kendall and T. E. Little. Constrained residual acceleration of conjugate residual methods. In *SPE Reservoir Simulation Symposium*, 1985.