

**TESTES E ANÁLISE DE DESEMPENHO NUMÉRICO E
COMPUTACIONAL DE UM MÉTODO PARALELO DE
RESOLUÇÃO DE SISTEMAS DE EQUAÇÕES LINEARES
USANDO ENFOQUE DE ESTIMAÇÃO LINEAR ÓTIMA**

RELATÓRIO FINAL DE ATIVIDADES DE INICIAÇÃO CIENTÍFICA
(PIBIC/CNPq/INPE)

Leno Silva Rocha (ITA, Bolsista PIBIC/CNPq)
E-mail: lenosr2002@yahoo.com

Dr. Stephan Stephany (LAC/CTE/INPE, Orientador)
E-mail: stephan@lac.inpe.br

COLABORADOR
Dr. Atair Rios Neto (LAC/CTE/INPE, Co-orientador)

Julho de 2009

“Nothing dignifies labor so much as the saving of it.” J. Rodgers

“Divino Jesus, eu vos ofereço este Terço (...) pelo aumento e santificação das vocações.” Oferecimento do Terço

AGRADECIMENTOS

A Deus por sua bondade; à Virgem Maria pela intervenção constante em minha vida; ao CNPq-Conselho Nacional de Desenvolvimento Científico e Tecnológico pelo apoio financeiro; ao doutor Stephan Stephany pelo acolhimento e apoio, pela oportunidade de desenvolvimento de um trabalho rico tecnicamente e pelo exemplo de como trabalhar em equipe; ao doutor Atair Rios Neto por seus ensinamentos e pelo acompanhamento assíduo do trabalho; e a todas as inúmeras pessoas e instituições que contribuíram e contribuem para o funcionamento da engrenagem social, que possibilitou a publicação deste trabalho.

RESUMO

Estimação linear ótima estocástica de parâmetros é utilizada no método iterativo, para a solução paralela de sistemas algébricos de equações, denominado ATAIR (Alternative Treatment Algorithm for Iterative Resolution). Esta abordagem, aliada às técnicas de aproximação da hessiana e fatoração de Potter aprimorou o algoritmo de resolução de sistemas lineares cuja convergência é garantida. É mostrada uma gama de testes práticos e avaliações de desempenho numérico do método com sistemas de equações lineares com diversas peculiaridades, como densidade de entradas, condicionamento, aplicação, ordem, simetria e tipos de esparsidade. As motivações são o aperfeiçoamento do método que explora as possibilidades oferecidas por processamento paralelo, que tem característica “general purposes” e robustez frente a inicializações, para a solução eficiente de sistemas de larga escala.

ABSTRACT

Stochastic optimal linear estimation of parameters is used in the iterative method for the parallel solution of systems of linear algebraic equations, entitled ATAIR (Alternative Treatment Algorithm for Iterative Resolution). This approach, allied to the hessian approximation and Potter's factorization technics, improved the algorithm of resolution of linear algebraic equations, which convergence is guaranteed. It is shown several practical tests and numerical performance evaluations of the method with systems of linear equations with many peculiarities, as entries density, conditioning, applications, order, symmetry and kinds of sparsity. The motivations are the improvement of the method which explores the possibilities offered by parallel processing, wich has general purposes characteristic and robustness facing initializations, to the efficient solution of large scale systems.

SUMÁRIO

1 – INTRODUÇÃO	6
2 – DESCRIÇÃO DO PROBLEMA	6
3 – RESULTADOS	9
3.1 Metodologia	9
3.2 Robustez	10
3.3 Sanitização	10
3.4 Comparação entre métodos	11
4 – CONCLUSÕES E TRABALHOS FUTUROS	12
BIBLIOGRAFIA	12
APÊNDICE – ALGUNS SISTEMAS DE EQUAÇÕES USADOS	13

1.INTRODUÇÃO

A resolução numérica de sistemas de equações lineares é questão de extrema importância em engenharia e em diversas outras áreas do conhecimento. Avanços tecnológicos de processamento paralelo, têm estimulado o desenvolvimento de novos métodos iterativos para resolução de sistemas lineares, cenário no qual encontra-se este trabalho.

Por ter sido desenvolvido com base em estimação ótima linear estocástica [7], um tratamento minoritário na literatura de algoritmos iterativos de resolução de sistemas de equações lineares [1], o método sob estudo é caracterizado como “alternative treatment” e foi denominado ATAIR (Alternative Treatment Algorithm for Iterative Resolution). A sigla homenageia o principal responsável pelo método, que agora leva seu nome.

Este primeiro algoritmo, publicado em 2000 [2], evoluiu para uma versão chamada PSIQUE (Parallel Solver with Iterative Quest Under Estimation). Como principais vantagens, ATAIR-PSIQUE apresenta: natureza altamente paralela, convergência matemática garantida, desempenho numérico competitivo [2], forte robustez frente a inicializações, característica “general purposes” e grande simetria, o que torna ainda mais simples sua implementação. Além disso, há versões do método tanto para sistemas determinados quanto para indeterminados.

Com respeito à relação entre as filosofias “science push x demand pull” de desenvolvimento científico-tecnológico, este trabalho é intrinsecamente mais próximo da primeira vertente. Porém, devido à importância do problema de resolução de sistemas lineares, há aderência ao segundo pólo.

O vínculo com o demand pull também pode ser percebido como a via de oferta de algoritmos que exploram a capacidade computacional facilmente encontrada no mercado atual. Sistemas com mais de 200 núcleos processadores já são acessíveis [10]. Assim, o desenvolvimento de métodos iterativos está atado, de certa forma, a essa realidade.

2.DESCRICÃO DO PROBLEMA

Dado o sistema linear de equações algébricas:

$$Ax = b, \tag{1}$$

com b real $n \times 1$ e A real $n \times n$ de posto cheio, encontrar uma solução numérica, com erro máximo tolerável da ordem de ε .

Pela abordagem de estimação linear estocástica, o sistema pode ser visto da seguinte maneira:

$$b = Ax + \varepsilon \tag{2}$$

Dessa forma, b é uma medida linearmente relacionada a uma quantidade desconhecida x na presença de ruído aditivo aleatório ε , de componentes independentes e não-enviesadas [7]. O objetivo é encontrar uma estimativa \hat{x} da incognita, que minimize o funcional ponderado pela matriz de pesos R^{-1} real $n \times n$:

$$f(x) = (1/2)[Ax - b]^T R^{-1}[Ax - b] \quad (3)$$

A matriz R^{-1} deve ser positiva definida, afim de garantir que a função custo seja estritamente convexa. A composição dessa matriz será deduzida adiante. Pela teoria de estimação, caso o sistema fosse de equações redundantes, R estaria relacionada a uma matriz de covariâncias [7] em um processo de mínimos quadrados.

Em cada iteração, as componentes incógnitas x_k ($k = 1, \dots, n$) serão vistas como:

$$x_k = \bar{x}_k + \bar{\eta}_k \quad (4)$$

onde $\bar{\eta}_k$ é o erro que representa a qualidade da aproximação \bar{x}_k , e será usado o esquema:

$$\rho(b - A\bar{x}) = A(x - \bar{x}) + \varepsilon \quad (5)$$

com ρ ajustando o resíduo, de modo que o erro $(x - \bar{x})$ seja convenientemente reduzido.

Para explorar paralelismo escreve-se a equação (5) em cada componente, como equações redundantes em x_j :

$$\rho(b_i - \sum_{t=1}^n a_{it}\bar{x}_t) = a_{ij}(x_j - \bar{x}_j) + \bar{v}_i^j \quad (6)$$

$$\bar{v}_i^j = \sum_{\substack{k=1 \\ k \neq j}}^n a_{ik}\bar{\eta}_k + \varepsilon_i, (i = 1, \dots, n) \quad (7)$$

Tratando a equação (6) como um problema de estimação de parâmetros, os erros \bar{v}_i^j podem ser vistos como aleatórios de média nula com distribuição normal, independentes, cuja dispersão é da forma:

$$E[(\bar{v}_i^j)^2] = E\left[\left(\sum_{\substack{k=1 \\ k \neq j}}^n a_{ik}\bar{\eta}_k\right)\left(\sum_{\substack{k=1 \\ k \neq j}}^n a_{ik}\bar{\eta}_k\right) + 2\varepsilon_i\left(\sum_{\substack{k=1 \\ k \neq j}}^n a_{ik}\bar{\eta}_k\right) + \varepsilon_i^2\right] \quad (8)$$

onde $E[.]$ é o operador valor esperado.

Considerando variâncias da ordem de tolerância a ser atingida, segue a simplificação:

$$E[(\bar{v}_i^j)^2] = \bar{\beta}_i^2 \sum_{\substack{k=1 \\ k \neq j}}^n a_{ik}^2 \varepsilon_i^2 = \bar{r}_i \quad (9)$$

onde $\bar{\beta}_i$ é um fator de ajuste e homogeneização. Um modo natural e conservador de realizar esse ajuste é considerar um critério consistente tipo maximum likelihood statistical [7] (máxima verossimilhança), com α a analisar:

$$\beta_i = \max \left\{ \max \left\{ \left(b_i - \sum_{t=1}^n a_{it}\bar{x}_t \right)^2, i = 1, \dots, n \right\}, 1 \right\}, \bar{\beta} \square \alpha\beta \quad (10)$$

Desse modo, as equações (6) podem ser processadas em paralelo usando um estimador de Gauss-Markov sem informação a priori [9]:

$$\hat{x}_j = \bar{x}_j + \alpha [A_j^T \bar{R}^{-1} A_j]^{-1} A_j^T \bar{R}^{-1} (b - A\bar{x}) \quad (11)$$

onde A_j é a j -ésima coluna de A ; e

$$\bar{R} = \text{diag}[\bar{r}_i] = \alpha^2 \beta^2 R \quad (12)$$

Notando-se que o estimador da equação (11) permite o cancelamento do fator $\alpha^2 \beta^2$ pode-se retomar o problema da equação (3), adotando-se um processo iterativo básico para otimização de $f(x)$: o método de Newton modificado [6]:

$$\hat{x} = \bar{x} - \alpha S \nabla f(\bar{x}) \quad (13)$$

Substituindo-se a expressão (13) no funcional (3), segue:

$$f(x - \alpha S g) = [Ax - \alpha ASg - b]^T R^{-1} [Ax - \alpha ASg - b] \quad (14)$$

Se R^{-1} for simétrica, então:

$$\bar{g} = \nabla f(\bar{x}) = A^T R^{-1} (A\bar{x} - b) \quad (15)$$

o que simplifica a expressão da derivada do funcional:

$$\frac{d}{d\alpha} f(x - \alpha S g) = -g^T S^T A^T R^{-1} (Ax - \alpha ASg - b) \quad (16)$$

Pela equação (15) na (16) igualada a zero, encontra-se o valor ótimo do regulador α do passo de busca [6]:

$$\alpha = (\bar{g}^T S \bar{g}) / (\bar{g}^T S A^T R^{-1} A S \bar{g}) \quad (17)$$

No ATAIR, usou-se S como matriz diagonal para aproximação da hessiana, com intuito de manter alto grau de paralelismo. Por motivo de instabilidade numérica, optou-se posteriormente, por utilizar o algoritmo DAVIDON-FLETCHER-POWELL (DFP) para a aproximação de uma matriz cheia da hessiana [6].

Apesar de existirem algoritmos mais avançados para a aproximação da hessiana, o DFP foi priorizado por não exigir grande aumento de cálculos em relação à primeira versão do ATAIR. Entretanto, com essa abordagem há um aumento da quantidade de passos não paralelos no algoritmo.

Essa estratégia de resolução do sistema de equações (1) corresponde ao PSIQUE, que pode ser sintetizada nos passos de 1 a 8 a seguir, considerando-se a convenção:

$x_{iter:k}$ $\equiv x_k$: valor do vetor x na k -ésima iteração.

x_k : k -ésima componente do vetor x .

Passos do algoritmo:

1- Calcular a matriz diagonal R da eq. (12)

- 2- Calcular a matriz diagonal $S = \text{diag}[(A_j^T R^1 A_j)^{-1} : j = 1, \dots, n]$
- 3- Encontrar a direção $d = - \underset{\text{iter.:}k}{S} \bar{g}$ usando \bar{g} da eq.(15)
- 4- Calcular α pela eq. (17)
- 5- Atualizar o valor de \hat{x} pela eq. (13)
- 6- Calcular a melhoria $p = \alpha d$
- 7- Definir $q = \underset{\text{iter.:}k}{\bar{g}} - \underset{\text{iter.:}k+1}{\bar{g}}$ e calcular $\underset{\text{iter.:}k+1}{S} = \underset{\text{iter.:}k}{S} + \frac{p_k p_k^T}{p_k^T q_k} - \frac{\underset{\text{iter.:}k}{S} q_k q_k^T \underset{\text{iter.:}k}{S}}{q_k^T \underset{\text{iter.:}k}{S} q_k}$
- 8- Atualizar a ordem k da iteração e:
 - 1- Se $k \leq n$ retornar ao passo 3
 - 2- Caso contrário, retornar ao passo 5 e ignorar o passo 7, mantendo S constante com seu último valor e fazer α constante igual a um.

3. RESULTADOS E ANÁLISES

Com intuito de ampliar a gama de casos de teste também com relação à avaliação do desempenho prático do método, foram utilizadas diferentes máquinas, sistemas operacionais, linguagens e softwares. Foram desenvolvidos programas com base nos algoritmos ATAIR e PSIQUE nos seguintes ambientes computacionais:

Tabela 1 – Ambientes computacionais.

Processamento	Memória RAM	Frequência	Precisão	Sistema Operacional	Linguagem
Single Core	1 Gb	1.8 Ghz	32 bits	Windows Vista™	MatLab®
Dual Core	786 Mb	2.8 Ghz	32 bits	Linux Ubuntu 8.04	Fortran 90
Single Core	1 Gb	2.2 Ghz	64 bits	Linux Suse 10.2	Fortran 90
Dual Core	4 Gb	2x 1.0 Ghz	64 bits	Linux Ubuntu 8.04	Fortran 90

3.1-Metodologia

Os sistemas utilizados foram obtidos por coleções de matrizes de teste. Para matrizes de ordem maior que 10 recorre-se ao uso de arquivos computacionais e programas de manipulação em massa de caracteres, já que é inviável tratar manualmente matrizes dessas ordens. No caso de matrizes esparsas usa-se a estratégia de armazenamento apenas das entradas não nulas das matrizes A e dos vetores b , como o formato CCS (Compressed Column Storage) usado, por exemplo, na coleção Harwell-Boeing (HB) [5]. Usa-se ainda, rotinas específicas para operações com as matrizes armazenadas nos formatos especiais, como o CCS.

A inclusão dos sistemas utilizados seria bastante válida, contudo o volume de informação envolvido se torna excessivamente grande, motivo pelo qual serão mostrados no apêndice apenas os sistemas de ordem não muito elevada. Contudo, estão disponíveis na tabela abaixo os dados mais relevantes sobre os sistemas, para a percepção de pontos

relevantes do trabalho como: característica “general purpose” dos métodos ATAIR e PSIQUE; abrangência da abordagem probatória. Além disso, a referência para possibilitar a reprodução e comparação dos testes também está indicada.

O número de condicionamento (coluna Cond.) de matriz adotado é o de Turing, que usa norma dos valores singulares [4], garantindo invariância de escala [1]. Para sistemas de ordem maior que 3, exceto nas coleções especiais, vale a regra:

$$b = A \cdot eye, \text{ em que } eye = (1, 1, \dots, 1)^T.$$

Para a simetria (coluna Sim.), usa-se a mesma convenção da coleção HB: S para matriz simétrica e U para assimétrica. Dados sobre padrão de esparsidade são informados nas coleções usadas.

Para indicação da densidade das matrizes (coluna D), utiliza-se o valor aproximado da porcentagem de entradas não nulas dentre o total de entradas. Dentre as matrizes esparsas, aquelas que apresentam padrão de esparsidade têm tal característica especificada na coluna P3.

Tabela2 – Sistemas usados para teste e suas principais características.

Origem	Matriz A	Ordem	Cond.	Sim.	D%	P3
[2]	“(i) case”	3	3	S	100	-
[2]	“(ii) case”	3	1441	S	100	-
[4]	Ex. 3.1	3	4573	U	100	-
[4]	Ex. 3.2	4	10	S	100	-
[4]	Ex. 3.5	4	2984	S	100	-
[4]	Ex. 3.4	6	3	U	27	Regular
[4]	Ex. 5.27	10	2587	U	8	Semi Regular
HB	RUA32	32	1378	U	1.3	-
HB	Utm300	300	1.5e6	U	3.5	-

3.2-Robustez

Em relação à escolha da aproximação da solução inicial no PSIQUE, observou-se grande robustez. Para cada um dos casos da tabela 1, de até quarta ordem, mais de cinco soluções iniciais de diferentes módulos e direções foram adotadas. Pretende-se buscar um método analítico para futuras análises de robustez. Caso não seja encontrada tal abordagem analítica, buscar-se-á sistematizar a metodologia de alteração dos vetores iniciais por meio de simulações de Monte-Carlo.

3.3-Sanitização

Grande parte dos métodos iterativos que dependem de um parâmetro de aceleração, como o alfa na equação (17), podem ter taxas de convergência muito baixa [1]. É comum que o parâmetro ótimo se encontre em uma região de declividade acentuada, de modo que pequenos erros de aproximação numérica no parâmetro podem prejudicar

significativamente a convergência do método. A mesma argumentação se aplica em relação ao cálculo da Hessiana [6].

Uma maneira de contornar esses problemas é usar a fatoração de raiz quadrada de Potter [3], tanto no cálculo do parâmetro de aceleração quanto na aproximação da Hessiana. Em relação ao desenvolvimento apresentado, ocorrem as seguintes alterações nas equações (11), (13) e (17):

$$v_j^T = A_j^T R^{-1/2}, s_j = (v_j^T v_j)^{-1}, p_j = s_j v_j^T R^{-1/2}$$

$$\hat{x}_j = \bar{x}_j + \alpha p_j (b - A\bar{x}) \quad (18)$$

$$V = [v_1 : v_2 : \dots : v_n], s = \text{diag}[s_j : j = 1, 2, \dots, n], p = s V^T R_e^{-1/2}$$

$$\hat{x} = \bar{x} + \alpha p r \quad (19)$$

$$g(\bar{x}) = -V^T R_e^{-1/2} r, g_s(\bar{x}) = s g(\bar{x}), g_V(\bar{x}) = V g_s(\bar{x})$$

$$\alpha = (g_s^T(\bar{x}) g(\bar{x})) / (g_V^T(\bar{x}) g_V(\bar{x})) \quad (20)$$

Foram implementados algoritmos em F90 e Matlab aplicando as técnicas de fatoração supracitadas, chamadas de sanitização, e observou-se melhoria geral nos métodos ATAIR-PSIQUE. A tabela abaixo mostra a melhoria para os casos de matrizes de ordem mais alta:

Tabela 3 – Comparação entre os métodos com e sem sanitização.

		PSIQUE Sem sanitização	PSIQUE Com Sanitização
RUA_32	iterações	32	32
	precisão	2e-4	4e-14
Utm300	iterações	849	300
	precisão	e-5	e-13

Como o desempenho do método melhora com as sanitizações, então elas foram incorporadas aos algoritmos de resolução de sistemas e doravante toda citação ATAIR-PSIQUE não incluirá resultados não sanitizados.

3.4-Comparações entre métodos

O método Jacobi [8], por ser amplamente difundido e ter natureza altamente paralela foi escolhido como base de comparação. Como convenção, a inicialização é o vetor nulo, exceto nas coleções especiais ou quando houver indicação explícita. Os principais resultados podem ser observados na tabela a seguir:

Tabela 4 – Resultados práticos dos diferentes métodos.

	Ordem	ATAIR		PSIQUE		Jacobi	
		Nº de iter.	Precisão	Nº iter.	Precisão	Nº iter.	Precisão
“(i) case”	3	14	e-5	3	e-15	3	e-5
“(ii) case”	3	2898	e-5	3	e-7	2000	e-5
Ex. 3.1	3	223574	e-5	3	e-8	-	Diverge
Ex. 3.2	4	65	e-5	4	e-14	-	Diverge
Ex. 3.5	4	648516	e-5	4	e-5	1997	e-5
Ex. 3.4	6	13	e-5	6	e-13	3	e-5
Ex. 5.27	10	4471719	e-5	11	e-8	3	e-5
RUA32	32	500000	e-5	32	e-14	-	Diverge
Utm300	300	Inviável	-	300	e-13	Inviável	-

4.CONCLUSÕES E TRABALHOS FUTUROS

Nota-se que o uso da aproximação da hessiana faz o algoritmo convergir em N iterações, sendo N a ordem do sistema. Além disso, a precisão alcançada, em média, melhora significativamente, sendo que em nenhum dos casos testados a precisão do algoritmo que não usa a aproximação da hessiana foi melhor que o PSIQUE.

As principais avaliações com sistemas de ordem até 300 foram realizadas em máquinas sequenciais. Pretende-se dar continuidade a este trabalho, usando matrizes de ordens mais altas (maiores que 300), tanto da coleção de matrizes HB, quanto de outras coleções, para um estudo mais abrangente do método, o que incluirá a análise de desempenho e temporização do programa sequencial em F90 (FORTRAN 90) para que seja feita a implementação de programas paralelos usando a biblioteca de comunicação por troca de mensagens MPI. Os testes serão realizados em um cluster do LAC/CTE/INPE.

BIBLIOGRAFIA

- [1] Assad, Y. (2003). Iterative Methods for Sparse Linear Systems. 2nd edition, SIAM.
- [2] Rios Neto, A. e Rios Neto, W. (2000). An Optimal Linear Estimation Approach to the Parallel Solution of Linear Algebraic Systems of Equations. SBA, Controle & Automação, 11(1), 61-67.
- [3] Pinto, R.L.U.F e Rios Neto, A. (1990). An Optimal Linear Estimation Approach to Solve Systems of Linear Algebraic Equations. Journal of Computacional and Applied Mathematics, 33, 261-268.
- [4] Gregory, R. T., e Karney, D. L. (1962). A Collection of Matrices for Testing Computational Algorithms. Wiley-Interscience.

[5] Duff, I. S. et al. (1992). User' Guide for the Harwell-Boeing Sparse Matrix Collection. Release I.

[6] Luenberger, D. G. (1984). Linear and Nonlinear Programming. Addison-Wesley, Reading, MA.

[7] Gelb, A. (1974). Applied Optimal Estimation. The MIT Press.

[8] Mehmood, R. e Crowcroft, J. (2005). Parallel iterative solution method for large sparse linear equation systems. Cambridge, UCAM-CL-TR-650, ISSN 1476-2986.

[9] Liebelt, P.B. (1967). An introduction to Optimal Estimation. Addison-Wesley, Reading, MA.

[10] Soluções TESLA™ HPC. Disponível em: <<http://www.nvidia.com>>. Acessado em 29 de Junho de 2009.

APÊNDICE – ALGUNS SISTEMAS DE EQUAÇÕES USADOS

$$\text{"(i)case": } A = \begin{bmatrix} 10 & 2 & 1 \\ 1 & 5 & 1 \\ 2 & 3 & 10 \end{bmatrix}; b = \begin{bmatrix} 1 \\ 2 \\ -3 \end{bmatrix}.$$

$$\text{"(ii)case": } A = \begin{bmatrix} 6 & 13 & -17 \\ 13 & 29 & -38 \\ -17 & -38 & 50 \end{bmatrix}; b = \begin{bmatrix} 1 \\ 2 \\ -3 \end{bmatrix}.$$

$$\text{Ex. 3.1: } A = \begin{bmatrix} 33 & 16 & 72 \\ -24 & -10 & -57 \\ -8 & -4 & -17 \end{bmatrix}; b = \begin{bmatrix} -359 \\ 281 \\ 85 \end{bmatrix}.$$

$$\text{Ex. 3.2: } A = \begin{bmatrix} 1 & -2 & 3 & 1 \\ -2 & 1 & -2 & -1 \\ 3 & -2 & 1 & 5 \\ 1 & -1 & 5 & 3 \end{bmatrix}; b = \begin{bmatrix} 3 \\ -4 \\ 7 \\ 8 \end{bmatrix}.$$

$$\text{Ex. 3.4: } A = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & -1 \\ -1 & 1 & 1 & 0 & 0 & 1 \\ 1 & -1 & 1 & 1 & 0 & -1 \\ -1 & 1 & -1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 1 & -1 \end{bmatrix}$$

$$\text{Ex. 3.4: } A = \begin{bmatrix} 5 & 7 & 6 & 5 \\ 7 & 10 & 8 & 7 \\ 6 & 8 & 10 & 9 \\ 5 & 7 & 9 & 10 \end{bmatrix}; b = \begin{bmatrix} 23 \\ 32 \\ 33 \\ 31 \end{bmatrix}$$

$$\text{Ex. 5.27: } A = \begin{bmatrix} 1 & 1 & 1 & -2 & 1 & -1 & 2 & -2 & 4 & -3 \\ -1 & 2 & 3 & -4 & 2 & -2 & 4 & -4 & 8 & -6 \\ -1 & 0 & 5 & -5 & 3 & -3 & 6 & -6 & 12 & -9 \\ -1 & 0 & 3 & -4 & 4 & -4 & 8 & -8 & 16 & -12 \\ -1 & 0 & 3 & -6 & 5 & -4 & 10 & -10 & 20 & -15 \\ -1 & 0 & 3 & -6 & 2 & -2 & 12 & -12 & 24 & -18 \\ -1 & 0 & 3 & -6 & 2 & -5 & 15 & -13 & 28 & -21 \\ -1 & 0 & 3 & -6 & 2 & -5 & 12 & -11 & 32 & -24 \\ -1 & 0 & 3 & -6 & 2 & -5 & 12 & -14 & 37 & -26 \\ -1 & 0 & 3 & -6 & 2 & -5 & 12 & -14 & 36 & -25 \end{bmatrix}$$