
This is a reprint from
Revista de la Academia Colombiana de Ciencias
Vol. 28 (106) (2004), 49-55

PRECONDICIONAMIENTO DE MÉTODOS ITERATIVOS

Por

Guillermo Cervantes C. & Carlos Mejía S.

PRECONDICIONAMIENTO DE MÉTODOS ITERATIVOS*

por

Guillermo Cervantes C.¹ & Carlos Mejía S.²

A la memoria de Jairo Charris C.

Resumen

Cervantes, Guillermo & Carlos Mejía S.: Precondicionamiento de métodos iterativos. Rev. Acad. Colomb. Cienc. **26** (106): 49–55, 2004. ISSN 0370-3908.

Se exponen nociones generales sobre métodos iterativos no estacionarios y se explica la importancia de agregarles precondicionamientos. Se presentan varios tipos de métodos iterativos y varias clases de precondicionadores incluyendo algunos definidos en términos de matrices por bloques. Para estos últimos, se presentan algunos resultados sugeridos en la literatura reciente que se lograron probar en detalle. Los resultados se comprueban con ejemplos ilustrativos.

Palabras clave: Precondicionamiento, métodos iterativos, matrices ralas, sistemas lineales.

Abstract

We consider general material related to nonstationary iterative methods and we explain the importance of implementing a preconditioning strategy.

Key words: Preconditioning, iterative methods, sparse matrices, linear systems.

¹Guillermo Cervantes C., Universidad del Norte, Barranquilla, Colombia. email: gcervant@uninorte.edu.co

²Carlos Mejía S., Universidad Nacional de Colombia, Sede de Medellín, Colombia, email: cemejia@unalmed.edu.co

*Este trabajo fue apoyado por la Dirección Nacional de Investigaciones de la Universidad Nacional a través del Proyecto de Investigación código DI00C1236.

AMS Mathematics Subject Classification 2000. 65F10, 65F50, 65F30, 15A06.

1. Introducción

En este trabajo se muestran maneras de acelerar métodos iterativos para la solución numérica del sistema $Ax = b$, con A matriz no singular de $\mathbb{R}^{n \times n}$ y $b \in \mathbb{R}^n$. La aceleración se consigue por preconditionamiento, el cual se describe en la sección 2. Para que estos métodos sean verdaderamente útiles, es conveniente que la matriz A sea *rala*, es decir, que la mayor parte de sus elementos sean cero. Es muy conveniente también que la matriz sea estructurada, por ejemplo, una matriz en la que los elementos no nulos están situados únicamente en unas cuantas diagonales, llamadas bandas. De esta manera, los métodos trabajan más rápido pues no es necesario almacenar la matriz en memoria.

El trabajo está organizado así: Como lo dijimos antes, la idea de preconditionamiento se explica en la sección 2. Con algún detalle, se consideran preconditionamientos basados en métodos iterativos estacionarios y en factorización LU incompleta. En la sección 3, sobre preconditionamiento para matrices definidas por bloques, se ofrecen resultados relacionados con preconditionamientos recientemente definidos para matrices especiales 2×2 por bloques.

2. Precondicionamiento

La idea del preconditionamiento es la reducción del número de iteraciones requerido para la convergencia, transformando el sistema original $Ax = b$ por un sistema $\tilde{A}x = \tilde{b}$, de tal forma que se satisfagan las siguientes propiedades:

- Resolver $\tilde{A}x = \tilde{b}$ no debe incrementar considerablemente el número de operaciones que se requieren para resolver $Ax = b$.
- $Ax = b$ y $\tilde{A}x = \tilde{b}$ tienen la misma solución, es decir, $\tilde{A}^{-1}\tilde{b} = A^{-1}b$.

La matriz \tilde{A} y el vector \tilde{b} se consiguen por premultiplicación o posmultiplicación por una matriz M , llamada *precondicionador* que debe ser fácilmente invertible. Por medio del preconditionador se transforma el sistema $Ax = b$ en otro equivalente con condiciones espectrales más favorables y se reduce el número de iteraciones requeridas para la convergencia, sin incrementar significativamente la cantidad de cálculos por iteración. Cuando esto se hace, hay que poner en una balanza los costos y los beneficios; es decir, el costo de construir y aplicar un preconditionador versus la ganancia en rapidez de convergencia. Ciertos preconditionadores necesitan

una pequeña fase de construcción pero otros pueden necesitar un trabajo sustancial. En el segundo caso la ganancia puede estar en el uso repetido del mismo preconditionador en múltiples sistemas lineales.

2. 1. Métodos Iterativos Estacionarios. Dado el preconditionador M para el sistema $Ax = b$, obtenemos un sistema equivalente $M^{-1}Ax = M^{-1}b$. Si escribimos $A = M - N$, entonces $Ax = b$ se puede escribir como $Mx = Nx + b$.

Así que dada una aproximación x_{k-1} , se puede obtener una nueva aproximación x_k de la siguiente forma:

$$Mx_k = Nx_{k-1} + b$$

multiplicando por M^{-1} se obtiene

$$x_k = M^{-1}Nx_{k-1} + M^{-1}b$$

pero $M^{-1}N = I - M^{-1}A$, entonces

$$\begin{aligned} x_k &= (I - M^{-1}A)x_{k-1} + M^{-1}b \\ &= x_{k-1} + M^{-1}(b - Ax_{k-1}) \\ &= x_{k-1} + M^{-1}r_{k-1} \end{aligned}$$

Si $M = D = \text{diag}(A)$, la iteración se conoce como método de Jacobi. Se supone que los valores en la diagonal de A son diferentes de cero para que M^{-1} pueda estar definida.

Si la matriz A se escribe de la forma $A = D + L_A + U_A$, donde $D = \text{diag}(A)$, L_A es la parte de A estrictamente triangular inferior y U_A es la parte de A estrictamente triangular superior, podemos definir dos métodos iterativos adicionales:

1. $M = D + L_A$, en cuyo caso la iteración se llama método de Gauss-Seidel.
2. $M = \omega^{-1}D + L_A$, donde ω es un parámetro en el intervalo $(0, 2)$. En este caso la iteración se llama SOR (*successive overrelaxation*).

Notemos que si $\omega = 1$, el método SOR se reduce al método de Gauss-Seidel. Si $\omega > 1$, el método se llama de **sobrerrelajación**, y si $\omega < 1$, el método se llama de **relajación**.

Cuando A es real y simétrica o compleja hermitiana y se definen $M_1 = \omega^{-1}D + L_A$ y $M_2 = \omega^{-1}D + U_A$, el par de iteraciones

$$\begin{aligned} M_1x_{k-\frac{1}{2}} &= N_1x_{k-1} + \omega b \text{ con } N_1 = M_1 - A \\ M_2x_k &= N_2x_{k-\frac{1}{2}} + \omega b \text{ con } N_2 = M_2 - A \end{aligned}$$

son denominadas método SSOR o SOR simétrico. En este caso el preconditionador M puede ser escrito como $M = \frac{\omega}{2-\omega} (\omega^{-1}D + L_A) D^{-1} (\omega^{-1}D + U_A)$.

Para matrices *consistentemente ordenadas* y que satisfacen la *Propiedad A*, (ver [7]) se conoce un valor óptimo para el parámetro ω . Se calcula en términos del radio espectral de la matriz de iteración de Jacobi, siempre que dicha matriz tenga valores propios reales y su radio espectral sea menor que 1. Dicho valor óptimo es $\omega_{opt} = \frac{2}{1+\sqrt{1-\beta^2}}$, donde β ($\beta < 1$) es el radio espectral de la matriz de iteración de Jacobi. Un dato importante es que las matrices tridiagonales por bloques y en particular, las matrices tridiagonales, satisfacen las dos condiciones mencionadas arriba.

2. 2. Factorización Incompleta. Una clase importante de preconditionadores se obtiene de la factorización LU , que sabemos trabaja para matrices arbitrarias. Sin embargo, existen algunas variantes que explotan la estructura especial que pueda tener la matriz: simétrica, definida positiva, ralas (sparse), etc. Para el caso de las matrices con un patrón de dispersión o ralas, se tiene un procedimiento denominado factorización incompleta, es decir: Si A es una matriz rala, los factores L y U usualmente no tienen el mismo patrón de dispersión de la matriz A , y como la idea es conservar este patrón, entonces se descartan los elementos diferentes de cero en aquellas posiciones donde la matriz A tenía un cero. Tales posiciones se denominan **fill-elements** (posiciones rellenas). Así se obtendría una factorización aproximada: $A \approx LU$.

La idea de generar factorizaciones aproximadas ha sido estudiada por muchos investigadores. El primero que lo hizo fue R. Varga en 1960 [9]. Posteriormente J. Meijerink y H. Van der Vorst en 1977 [4] la hicieron popular cuando la usaron para generar preconditionadores para el gradiente conjugado y otros métodos iterativos.

Uno de los más importantes preconditionadores de esta clase es el que se obtiene por la factorización incompleta de Cholesky. Si A es una matriz rala, simétrica y definida positiva, se puede calcular una matriz H que tenga el mismo patrón de dispersión que la triangular inferior de A y que de algún modo esté cercana al factor de Cholesky G de A , así el preconditionador toma la forma $M = HH^T$.

Recordemos que para dos matrices B y C , decir que $B \geq C$ significa que para todos los correspondientes elementos, se cumple $b_{ij} \geq c_{ij}$. La factorización incompleta

vía LU está garantizada si A es invertible y diagonalmente dominante o si A es una M-matriz, es decir, si A es una matriz con elementos no positivos fuera de la diagonal principal, invertible y $A^{-1} \geq 0$ (Ver [8], pag. 210).

El siguiente teorema, probado en [4], establece la existencia y unicidad de la factorización incompleta LU para M-matrices.

Teorema 1. Si $A = (a_{ij})_{n \times n}$ es una M-matriz, entonces para cada subconjunto P de $\{(i,j): j \neq i, i, j=1,2,\dots,n\}$ existe una matriz triangular inferior $L = (l_{ij})_{n \times n}$ con unos en la diagonal y una matriz triangular superior $U = (u_{ij})_{n \times n}$ tal que $A = LU - R$, donde $l_{ij} = 0$ si $(i,j) \in P$, $u_{ij} = 0$ si $(i,j) \in P$ y $r_{ij} = 0$ si $(i,j) \notin P$. Los factores L y U son únicos y la descomposición $A = LU - R$ es tal que LU es no singular, $(LU)^{-1} \geq 0$ y $LU \geq A$.

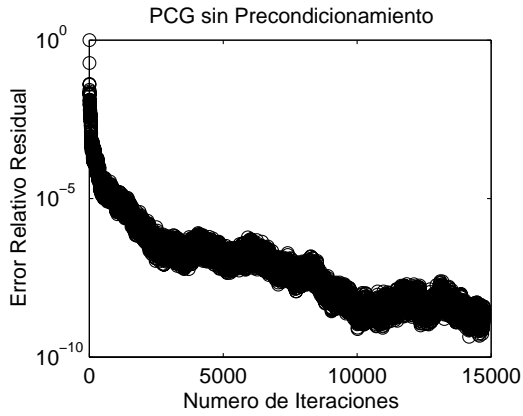
A la luz del teorema anterior, si escogemos $M = LU$, donde L es triangular inferior y U es triangular superior, el sistema $Mx = y$ (**instrucción del algoritmo del gradiente conjugado preconditionado**) se resuelve por la vía usual, es decir, por sustitución regresiva y progresiva. La factorización incompleta que se obtiene descartando todos los **fill-elements** en el proceso LU se conoce como **Factorización Incompleta de Nivel Cero** y se denota **ILU(0)**.

En todos los ejemplos presentados en este trabajo, se utilizan las rutinas que trae MATLAB 6.1 (versión instruccional) para los métodos iterativos *Gradiente Conjugado (CG)* y *Minimum Residual (MINRES)*. El primero se usa para matrices simétricas definidas positivas y el segundo para matrices generales. Además, en los ejemplos se trabaja con un vector de *unos* como solución exacta conocida y con un vector nulo como primera aproximación.

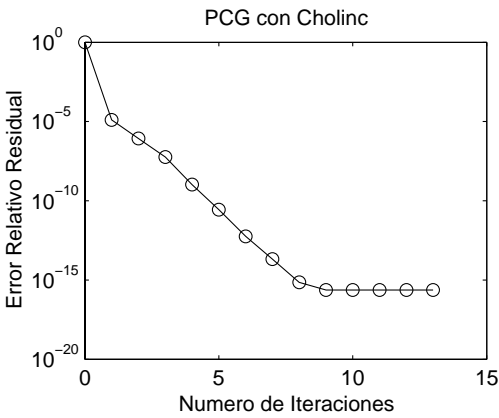
Ejemplo 1. Utilizamos los preconditionamientos Jacobi, SSOR y Factorización Incompleta de Cholesky en un sistema con matriz simétrica definida positiva tomada de la colección *Matrix Market* [6]. La matriz elegida para este ejemplo se llama *bcstk11*. Esta matriz aparece en aplicaciones de ingeniería estructural, tiene 1473 filas y un número de condición con un orden de magnitud de 10^8 .

Los experimentos que hicimos los presentamos en las siguientes figuras y en una tabla de resumen.

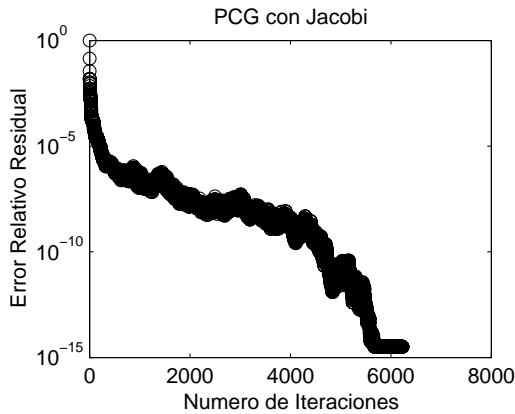
El primer resultado es el correspondiente a ausencia de preconditionamiento.



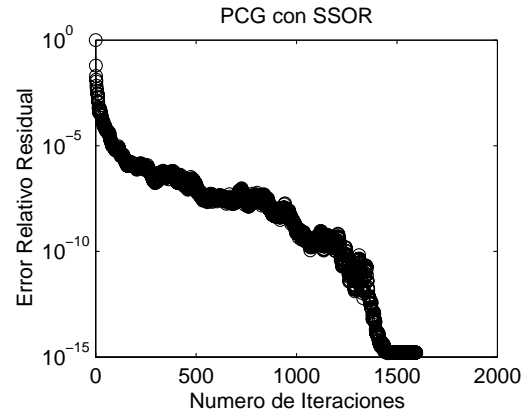
El segundo resultado corresponde a precondicionamiento con factorización incompleta de Cholesky.



En tercer lugar, aparecen los resultados de utilizar el método de Jacobi para precondicionamiento del CG.



Por último, estos son los resultados de aplicar el precondicionamiento SSOR a CG.



En la tabla siguiente resumimos estos experimentos.

Matriz bcsstk11 de Matrix Market			
Método	Iteraciones	Error	Tiempo(sg)
No Precond.	28312	4.4E-12	167
Cholinc(6)	13	8.1E-12	1.13
Jacobi	6236	1.3E-11	38
SSOR $\omega = 1.8$	1695	8.6E-12	511

3. Precondicionadores por Bloques

En las aplicaciones, frecuentemente las matrices deben considerarse por bloques. De hecho, las estrategias anteriores para producir precondicionadores por factorización incompleta y por los métodos iterativos clásicos se pueden adaptar para cuando la matriz A se particiona por bloques.

Otro tipo de precondicionadores por bloques se presenta en casos muy específicos. Por ejemplo, la matriz proveniente de la discretización de la ecuación de Poisson en un cuadrado unitario con n^2 puntos de malla tiene la siguiente estructura

$$\text{Trid por bloques}(-I, T, -I),$$

donde $T = \text{Trid}(-1, 4, -1)$. En este caso, un precondicionador natural es: $M = \text{Diag por bloques}(T)$.

Otra fuente de precondicionadores con estructura de bloque se origina cuando la matriz del sistema tiene una forma denominada **KKT** (Karush, Kuhnn, Tucker). Es decir, matrices *no singulares* que tienen la siguiente estructura:

$$\begin{pmatrix} A & B^T \\ C & 0 \end{pmatrix}, \text{ donde } A \in \mathbb{R}^{n \times n} \text{ y } B, C \in \mathbb{R}^{m \times n}, \quad (1)$$

con $n \geq m$. En muchas aplicaciones A es simétrica y $B = C$, en cuyo caso (1) sería simétrica; en todo caso, si A es o no simétrica, la matriz (1) es generalmente indefinida, es decir, las partes reales de sus valores propios pueden ser positivas o negativas. Matrices con esta estructura se presentan frecuentemente en aplicaciones: problemas de optimización, problemas de fluidos, etc. Murphy, Golub y Wathen muestran en [5] un preconditionador bastante eficiente para sistemas lineales cuya matriz de coeficientes tiene la forma (1). Dicho preconditionador tiene la siguiente estructura cuando la matriz A es invertible:

$$P = \begin{pmatrix} A & 0 \\ 0 & CA^{-1}B^T \end{pmatrix} \quad (2)$$

Este preconditionador fue generalizado posteriormente por Ipsen en [2] para matrices *no singulares* de la forma

$$\begin{pmatrix} A & B^T \\ C & D \end{pmatrix}, \quad (3)$$

donde $A \in \mathbb{R}^{n \times n}$, B y $C \in \mathbb{R}^{m \times n}$. y $D \in \mathbb{R}^{m \times m}$ con $n \geq m$. Dicho preconditionador también se ha utilizado en la solución de sistemas lineales provenientes de la discretización de las ecuaciones linealizadas de Navier-Stokes [3] con muy buenos resultados.

A continuación se detallan varias proposiciones en las cuales al preconditionar sistemas lineales cuyas matrices de coeficientes tengan la forma (1) o (3), usando preconditionadores con la forma (2), se obtienen matrices preconditionadas con a lo más cuatro, tres o dos valores propios distintos. La primera de tales proposiciones aparece demostrada en [5], las restantes aparecen referenciadas pero no probadas en [5] y [2]. Las pruebas detalladas de tales proposiciones pueden consultarse en [1].

El preconditionador (2) es sugerido mediante una descomposición de la matriz (1) usando el complemento de Schur, el cual se define a continuación:

Definición 2. Supongamos que la matriz $A \in \mathbb{R}^{n \times n}$ se puede escribir de la forma $A = \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix}$, donde A_{11} es $r \times r$. Si se supone que A_{11} es no singular, la matriz $S = A_{22} - A_{21}A_{11}^{-1}A_{12}$ se denomina el **Complemento de Schur** de A_{11} en A .

La primera proposición que presentamos afirma que el preconditionador a izquierda (2) genera una matriz preconditionada con polinomio minimal de grado a lo más 4.

Proposición 3. Si la matriz $A = \begin{pmatrix} A & B^T \\ C & 0 \end{pmatrix}$, donde $A \in \mathbb{R}^{n \times n}$ es invertible y $B, C \in \mathbb{R}^{m \times n}$ con $n \geq m$, se preconditiona con (2), entonces la matriz preconditionada $T = P^{-1}A$ satisface $T(T - I)(T^2 - T - I) = 0$.

La segunda proposición afirma algo similar del polinomio minimal de la matriz preconditionada cuando el preconditionamiento se toma a derecha o a izquierda.

Proposición 4. Si $T = AP^{-1}$ o si $T = P_1^{-1}AP_2^{-1}$ con $P_1P_2 = P$ y A como en la proposición anterior, entonces también se satisface que

$$T(T - I)(T^2 - T - I) = 0.$$

La proposición que se presenta a continuación es referenciada sin demostración en [5]. Indica que métodos iterativos basados en subespacios de Krylov, como *min-res*, convergen en a lo más 4 iteraciones, suponiendo que se puede trabajar con *matemática exacta*.

Proposición 5. Para cualquier vector r , el subespacio de Krylov

$$\text{gen}\{r, Tr, T^2r, T^3r, \dots\}$$

con T como en la proposición anterior, es a lo más de dimensión 3 si T es no singular (o 4 si T es singular).

Demostración. Sin pérdida de generalidad supondremos que T es singular. Como

$$Q(t) = t(t - 1) \left(t - \frac{1 + \sqrt{5}}{2} \right) \left(t - \frac{1 - \sqrt{5}}{2} \right) \quad (4)$$

es un polinomio mónico que anula a T , entonces el polinomio minimal de T tiene a lo más grado 4.

Usando el Teorema 1.1.1 de [1], obtenemos que la dimensión de

$$\text{gen}\{r, Tr, T^2r, T^3r, \dots\} \quad (5)$$

es a lo sumo 4. Si T es no singular, el polinomio (4) tendría a lo sumo grado 3 y por tanto el subespacio en (5) tiene a lo más dimensión 3. \square

En la siguiente proposición se presentan variaciones del preconditionador (2), las cuales producen matrices preconditionadas con uno o dos valores propios distintos. Tal proposición es presentada en [5] pero sin demostración.

Proposición 6. Si se escoge $P = \begin{pmatrix} A & B^T \\ 0 & CA^{-1}B^T \end{pmatrix}$ en la proposición 3 el sistema preconditionado que resulta tiene exactamente dos valores propios 1 y -1. Si $CA^{-1}B^T$ es reemplazado por $-CA^{-1}B^T$, entonces la matriz preconditionada tiene a 1 como único valor propio.

El preconditionador trabajado en las proposiciones anteriores puede ser extendido a matrices no singulares de la forma $\mathcal{A} = \begin{pmatrix} A & B^T \\ C & D \end{pmatrix}$, donde la matriz A es invertible. La matriz \mathcal{A} no es necesariamente definida positiva y puede ser compleja, pero aquí trabajamos únicamente el caso real.

La proposición que sigue está propuesta sin demostración en [2] y muestra una extensión del preconditionador (2) desarrollado en la proposición 3.

Proposición 7. Sea $\mathcal{A} = \begin{pmatrix} A & B^T \\ C & D \end{pmatrix}$, donde $A \in \mathbb{R}^{n \times n}$ es invertible, B y $C \in \mathbb{R}^{m \times n}$, $D \in \mathbb{R}^{m \times m}$, $n \geq m$, $C \neq 0$ y $P = \begin{pmatrix} A & B^T \\ 0 & S \end{pmatrix}$, donde $S = D - CA^{-1}B^T$ es el Complemento de Schur de A . Entonces AP^{-1} y $P^{-1}\mathcal{A}$ tienen como polinomio minimal a $Q(t) = (t-1)^2$.

La siguiente proposición, también de [2], es una generalización de la proposición 4 propuesta arriba.

Proposición 8. Si $P_1 = \begin{pmatrix} I & 0 \\ CA^{-1} & -I \end{pmatrix}$, $P_2 = \begin{pmatrix} A & B^T \\ 0 & S \end{pmatrix}$ y \mathcal{A} como en la proposición anterior, entonces $P_1^{-1}\mathcal{A}P_2^{-1} = \begin{pmatrix} I & 0 \\ 0 & -I \end{pmatrix}$. Además la matriz $P = P_1P_2 = \begin{pmatrix} A & B^T \\ C & D - 2S \end{pmatrix}$ es tal que $P^{-1}\mathcal{A}$ tiene como polinomio minimal a $Q(t) = (t-1)(t+1)$.

La proposición que se presenta a continuación es una extensión de la proposición 6 y aparece referenciada en [2] pero sin demostración.

Proposición 9. Si $\mathcal{A} = \begin{pmatrix} A & B^T \\ C & D \end{pmatrix}$, con A , B , C y D como en la proposición 7, entonces el preconditionador $P = \begin{pmatrix} A & 0 \\ 0 & -S \end{pmatrix}$, donde S es el complemento de Schur de A , es tal que la matriz preconditionada es $T = AP^{-1} = \begin{pmatrix} I & -B^TS^{-1} \\ CA^{-1} & -DS^{-1} \end{pmatrix}$. Si \mathcal{A} es de la forma KKT, es decir $D = 0$, entonces $T^2 - T = \begin{pmatrix} -B^TS^{-1}CA^{-1} & 0 \\ 0 & I \end{pmatrix}$. Puesto que $(T^2 - T)^2 = T^2 - T$, entonces el polinomio minimal de T es a lo más de grado 4.

Las proposiciones anteriores permiten concluir que para matrices de la forma

$$\mathcal{A} = \begin{pmatrix} A & B^T \\ C & 0 \end{pmatrix},$$

donde $A \in \mathbb{R}^{n \times n}$ y $B, C \in \mathbb{R}^{m \times n}$ con $n \geq m$, que por lo general son indefinidas, los preconditionadores

$$P = \begin{pmatrix} A & 0 \\ 0 & \pm CA^{-1}B^T \end{pmatrix} \text{ y } P = \begin{pmatrix} A & B^T \\ 0 & \pm CA^{-1}B^T \end{pmatrix}$$

logran que el polinomio minimal de $P^{-1}\mathcal{A}$ tenga a lo más 4 valores propios distintos, lo cual implica que, bajo aritmética exacta, métodos Krylov como el **minres** al ser aplicados al sistema lineal preconditionado converja en a lo sumo 4 iteraciones.

Conclusión similar se puede obtener para matrices de la forma

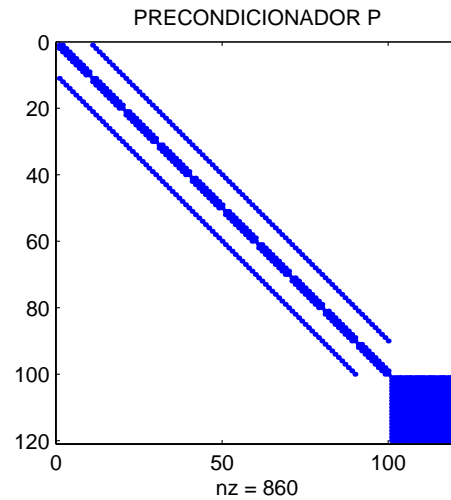
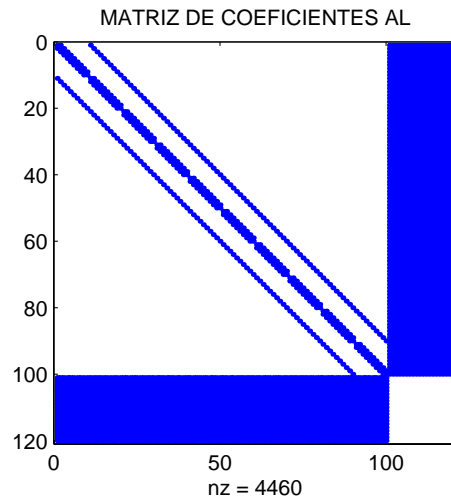
$$\mathcal{A} = \begin{pmatrix} A & B^T \\ C & D \end{pmatrix},$$

donde $A \in \mathbb{R}^{n \times n}$, B y $C \in \mathbb{R}^{m \times n}$ y $D \in \mathbb{R}^{m \times m}$ con $n \geq m$, con preconditionadores

$$P = \begin{pmatrix} A & B^T \\ 0 & \pm S \end{pmatrix}, P = \begin{pmatrix} A & B^T \\ C & D - 2S \end{pmatrix}$$

donde $S = D - CA^{-1}B^T$.

Ejemplo 2. Se diseñaron sistemas lineales con matrices de la forma (1) de tamaños diferentes. En todos los casos, los experimentos se hicieron de manera que la solución exacta sea un vector con todos sus elementos iguales a 1 y siempre se tomó el vector nulo como aproximación inicial. Como criterio de parada se utilizó una tolerancia de $1E-8$.



La tabla siguiente resume los experimentos que se hicieron.

Dimensiones		Sin precond.		MGW	
A	AL	# Iter.	Error	# Iter.	Error
100	120	130	2.8E-5	2	2.6E-10
400	450	295	8.8E-5	2	1.2E-7
900	1000	326	1.3E-4	2	3.8E-6
2025	4000	2582	1.4E-4	10	2.6E-3
2500	3000	865	1.4E-4	4	2.0E-4
3600	4000	629	1.0E-4	6	1.5E-5

El caso en el que se presentaron 10 iteraciones tuvo final anormal. El mensaje es que dos iterados consecutivos no se distinguen, de acuerdo con la tolerancia dada. Nótese que corresponde a una dimensión pequeña de A , en comparación con la de AL : La dimensión 2025 es menos del 51% de la dimensión 4000.

El otro caso con más de 4 iteraciones es el último, en el que se reportan 6 pero tuvo final normal.

En estos dos casos debe pensarse también que con matrices de tamaño 4000×4000 , no se puede pretender

que los cálculos se hagan exactamente. El inevitable redondeo es un factor que puede explicar el alejamiento moderado de lo que dice la teoría para estos casos.

Referencias

- [1] **G. Cervantes**, *Introducción a los preconditionadores*, Tesis de Maestría, Universidad Nacional de Colombia, Medellín, 2003.
- [2] **I. Ipsen**, *A note on preconditioning nonsymmetric matrices*, SIAM J. Sci. Comput., **23** (3) (2001), 1050–1051.
- [3] **D. Kay, D. Loghin & A. Wathen**, *A preconditioner for steady-state Navier–Stokes equations*, SIAM J. Sci. Comput., **24** (1) (2002), 237–256.
- [4] **J. A. Meijerink & H. A. van der Vorst** *An iterative solution method for linear systems of which the coefficient matrix is a symmetric m -matrix*, Math. Comput. **31** (137) (1977), 148–162.
- [5] **M. Murphy, G. Golub & A. Wathen**, *A note on preconditioning for indefinite linear systems*. SIAM J. Sci. Comput., **21** (6) (2000), 1969–1972.
- [6] **National Institute of Standards and Technology**, *The matrix market*. <http://math.nist.gov/MatrixMarket>.
- [7] **J. M. Ortega**, *Numerical Analysis, a Second Course*. SIAM, 1990.
- [8] **G. W. Stewart**, *Afternotes on Numerical Analysis*. SIAM, 1996.
- [9] **R. Varga**, *Factorization and normalized iteratives methods*. IN **R. Laner** (ed.), *Boundary Problems in Differential Equations*, University of Wisconsin, Madison, 1960, 12–142.