

# **Matrices de Nekrasov y alta precisión relativa**



**Héctor Orera Hernández**

Trabajo de fin de Máster en Modelización e  
Investigación Matemática, Estadística y Computación  
Universidad de Zaragoza

Director del trabajo: Juan Manuel Peña  
Ferrández



# Introducción

El estudio y control del error aparece como un problema fundamental al desarrollar métodos numéricos. En particular, al utilizar la representación en coma flotante se presenta el estudio del error de redondeo. Un análisis clásico del mismo depende del condicionamiento del problema. No obstante, para algunos problemas se ha logrado dar un enfoque distinto desarrollando algoritmos con alta precisión relativa. Por ejemplo, para realizar cálculos con clases de matrices estructuradas. En estos algoritmos se parte de parametrizaciones de las matrices que permiten asegurar la alta precisión relativa independientemente del condicionamiento de las mismas. Hasta el momento, los ejemplos de clases de matrices encontrados que presentan esta ventaja son o están relacionados con subclases de P-matrices. Recordemos que las P-matrices son matrices cuadradas con todos los menores principales positivos. Dentro de esta clase, destacan las M-matrices no singulares por sus numerosas aplicaciones. En esta memoria, presentaremos dichas matrices, describiremos una parametrización que permite obtener alta precisión relativa y veremos cómo utilizarla. Después, introduciremos las denominadas matrices de Nekrasov. Estas matrices están íntimamente relacionadas con las M-matrices diagonalmente dominantes, y aparecen en diversas aplicaciones, por ejemplo, al buscar cotas de error en el problema de complementariedad lineal. Para las matrices de Nekrasov presentaremos una parametrización adecuada que nos permitirá obtener un método de alta precisión relativa para hallar la inversa y resolver ciertos sistemas lineales de ecuaciones. También, construiremos una matriz de escalado para las matrices de Nekrasov que las lleva a forma estrictamente diagonalmente dominante. A partir de ahí, deduciremos cotas para la norma de la inversa de una matriz de Nekrasov, problema con importantes aplicaciones potenciales.

La memoria se estructura en siete capítulos. En el primer capítulo introducimos los conceptos básicos que necesitamos para plantear un estudio del error, definimos alta precisión relativa y damos una condición suficiente para asegurarla.

En el segundo capítulo presentamos las diversas clases de matrices estructuradas relacionadas con las de Nekrasov, que engloban los problemas estudiados posteriormente: las P-matrices, las Z-matrices, las M-matrices, las matrices diagonalmente dominantes y las H-matrices.

En el tercer capítulo nos centramos en las M-matrices no singulares. Las M-matrices para las que vamos a lograr algoritmos con alta precisión relativa cumplen además la condición de dominancia diagonal. En este caso, la parametrización adecuada para trabajar con ellas vendrá dada por los elementos extradiagonales de la matriz así como la suma de los elementos de cada fila de la misma. Con estos parámetros, buscaremos obtener con alta precisión relativa lo que se llama una descomposición reveladora del rango. Estas descomposiciones permiten obtener los valores singulares con alta precisión relativa. En el caso de las M-matrices diagonalmente dominantes, las descomposiciones reveladoras del rango serán ciertas factorizaciones LDU, obtenidas utilizando la eliminación Gaussiana de forma apropiada, es decir, empleando adecuadas estrategias de pivotaje simétrico. Por ello recordaremos también dicho método de

eliminación y expondremos cómo adaptarlo para obtener una factorización LDU mediante un algoritmo libre de restas (y así con alta precisión relativa), pudiendo emplear para ello dos técnicas distintas de pivotaje simétrico. Al final del capítulo incluimos ejemplos numéricos para comparar las técnicas de pivotaje simétrico descritas.

Los demás capítulos se centran en el estudio de las matrices de Nekrasov. En el cuarto capítulo enunciamos propiedades elementales de las mismas, y para la subclase compuesta por las Z-matrices de Nekrasov con elementos diagonales positivos, proponemos una parametrización a partir de la cual se obtienen algoritmos con alta precisión relativa para el cálculo de inversas y para el cálculo de sistemas de ecuaciones lineales con términos independientes no negativos.

El quinto capítulo es el dedicado a desarrollar dichos métodos con alta precisión relativa, y en él se incluye también el pseudocódigo necesario para implementarlos. La clave consiste en relacionar las Z-matrices de Nekrasov con diagonal positiva con las M-matrices diagonalmente dominantes empleando una matriz de escalado, y utilizar técnicas conocidas para estas matrices descritas en el tercer capítulo aunque sin necesitar estrategias de pivotaje. Además, en diversos casos, también resulta necesario utilizar la estructura de ceros de nuestra matriz para poder resolver dichos problemas. Tras presentar los algoritmos necesarios, en el sexto capítulo se incluyen experimentos numéricos para comparar los resultados obtenidos al utilizar dichos algoritmos con alta precisión relativa con los empleados habitualmente.

El último capítulo trata un problema diferente. Se presentan dos matrices de escalado que llevan a una matriz de Nekrasov a forma estrictamente diagonalmente dominante. Con dichas matrices de escalado obtenemos cotas de la norma de la inversa de nuestra matriz original. Éste es un problema con importantes aplicaciones y que está siendo estudiado en la actualidad, por lo que para ver la bondad de las cotas introducidas en la memoria recogemos las cotas más utilizadas y concluimos el capítulo realizando una comparativa con otros métodos publicados recientemente.

# Índice general

<b>Introducción</b>	<b>III</b>
<b>1. Error y cálculos con alta precisión relativa</b>	<b>1</b>
1.1. Conceptos básicos . . . . .	1
1.2. Representación en coma flotante . . . . .	2
1.3. Condicionamiento y alta precisión relativa . . . . .	3
<b>2. Clases de matrices relacionadas</b>	<b>7</b>
2.1. P-matrices . . . . .	7
2.2. Z-matrices, M-matrices y dominancia diagonal . . . . .	8
2.3. H-matrices . . . . .	10
<b>3. Eliminación Gaussiana y M-matrices diagonalmente dominantes</b>	<b>11</b>
3.1. Eliminación Gaussiana, pivotaje y método de Gauss-Jordan . . . . .	11
3.2. Descomposiciones reveladoras del rango, parametrización y pivotaje . . . . .	12
3.3. Experimentación numérica . . . . .	15
<b>4. Matrices de Nekrasov</b>	<b>19</b>
4.1. Definición y parametrización . . . . .	19
4.2. Escalado para dominancia diagonal . . . . .	20
4.3. HRA para matrices de Nekrasov . . . . .	21
<b>5. Inversas precisas de Z-matrices de Nekrasov</b>	<b>23</b>
5.1. Inversas precisas: caso particular . . . . .	23
5.2. Inversas precisas: caso general . . . . .	25
5.3. Algoritmos . . . . .	28
<b>6. Experimentación numérica para inversas y sistemas</b>	<b>31</b>
6.1. Cálculo de inversas . . . . .	31
6.2. Resolución de sistemas lineales . . . . .	32
<b>7. Escalado para dominancia diagonal estricta</b>	<b>37</b>
7.1. Matrices de escalado . . . . .	37
7.2. Cota para la norma de la inversa . . . . .	40
7.3. Cotas alternativas para la norma de la inversa . . . . .	41
7.4. Comparación de cotas . . . . .	43
<b>Referencias</b>	<b>47</b>



# Capítulo 1

## Error y cálculos con alta precisión relativa

Un problema fundamental del álgebra lineal numérica consiste en encontrar métodos eficientes (estables y de bajo coste computacional) para la resolución de sistemas de ecuaciones lineales con ordenador, es decir, resolver  $Ax = b$  con  $A$  matriz cuadrada no singular. Un problema muy relacionado con el anterior es el de encontrar métodos eficientes para hallar la inversa de una matriz no singular puesto que, si tenemos el sistema  $Ax = b$ , entonces  $x = A^{-1}b$ . Si  $A$  es una matriz  $n \times n$  el método más usado en general para la resolución de sistemas de ecuaciones lineales es la eliminación Gaussiana (véase sección 3.1), que tiene un coste computacional del orden de  $n^3$  operaciones elementales. El método de Gauss-Jordan es el más usado en general para hallar la inversa de una matriz  $n \times n$ . Si la matriz tiene una estructura especial, puede haber métodos con menos coste computacional para resolver los problemas anteriores. Como el ordenador usa una aritmética de punto flotante (véase sección 1.2), ésto ocasiona que exista una diferencia entre nuestro cálculo y la solución exacta, a la que llamamos error. Para controlar estos errores en la eliminación Gaussiana se suelen usar técnicas de pivotaje, como, por ejemplo, el pivotaje parcial.

El primer apartado de este capítulo lo dedicamos a introducir algunos conceptos básicos relacionados con los errores. El siguiente apartado lo dedicamos a la representación en coma flotante. Finalmente, la última sección trata los conceptos de condicionamiento y alta precisión relativa.

### 1.1. Conceptos básicos

El estudio del error aparece como un problema fundamental para desarrollar buenos algoritmos. Existen tres causas fundamentales de error. La primera es el redondeo consecuencia de trabajar en una aritmética de precisión finita. Los errores de redondeo no son aleatorios, y aunque a veces puedan ser beneficiosos, como al aplicar el método de potencias partiendo de un vector inicial elegido desafortunadamente, hay que tener presente que un algoritmo deficiente puede magnificar estos errores y dar lugar a una solución numérica inútil. La segunda es la incertidumbre que podamos tener en los datos de cualquier problema en la práctica, bien sea debida a errores de medición o estimación, a errores de almacenamiento de los datos o a errores de cálculos previos si estos datos son solución de un problema anterior. La tercera es la discretización que puede tener que llevarse a cabo al plantear la resolución práctica del problema. En cualquier caso, nos encontramos una primera cuestión: ¿Cómo se cuantifica el error? Supongamos que queremos calcular un valor  $x$  escalar.

**Definición 1.1.** El error absoluto cometido al hallar  $\hat{x}$  es  $E_{abs}(\hat{x}) = |x - \hat{x}|$ .

Esta primera definición no tiene en cuenta la magnitud de la cantidad a calcular, por lo que puede no ser muy informativa. Por tanto, se introduce el error relativo:

**Definición 1.2.** El error relativo cometido al hallar  $\hat{x}$ , definido cuando  $x \neq 0$ , es  $E_{rel}(\hat{x}) = \frac{|x - \hat{x}|}{|x|}$ .

El concepto de error relativo está relacionado con el número de cifras significativas correctas que obtenemos, por lo que será el que atraiga nuestro interés. En el caso vectorial se puede extender la misma definición de esta forma:

**Definición 1.3.** El error relativo cometido al calcular el vector  $\hat{x}$ , definido cuando  $x \neq 0$ , es  $E_{rel}(\hat{x}) = \frac{\|x - \hat{x}\|}{\|x\|}$ .

No obstante, puede que se obvие el error cometido en las componentes de menor magnitud del vector, por lo que también es interesante definir :

**Definición 1.4.** El error relativo componente a componente del vector  $\hat{x}$ , definido cuando  $x_i \neq 0$ , es  $\max_i \frac{|x_i - \hat{x}_i|}{|x_i|}$ .

Como no conocemos con exactitud el error que cometemos, la forma de proceder consiste en dar cotas de este error, al que se denomina *forward* (o progresivo), que aseguren que nuestros cálculos son buenos.

Otro planteamiento posible y que ha resultado muy útil para acotar el error forward (como mencionamos en la sección 1.3) es considerar para qué valores iniciales del problema nuestra solución numérica sería la solución exacta. Tomando como ejemplo  $y = f(x)$ , una función continua real de variable real, e  $\hat{y}$  una aproximación numérica a  $f$  en un punto  $x$  dado, consideramos el conjunto de valores  $x + \Delta x$  para el que sería la solución exacta, y tomamos el menor  $|\Delta x|$ , al que llamamos *error backward* (o regresivo). Si para todo  $x$ , el valor  $|\Delta x|$  es pequeño (en el contexto del problema que tratemos) diremos que el método es *estable backward*. El estudio de la estabilidad backward juega un papel importante en el diseño de un buen algoritmo. En un problema concreto, podemos definir el factor de crecimiento, que es una medida del incremento de la magnitud de los datos con los que se trabaja. Si estos crecen demasiado podría darse un problema de *overflow*, lo que significa que una cantidad calculada ha superado el máximo del conjunto de números representables. Clásicamente, en álgebra lineal numérica, tener una cota adecuada del factor de crecimiento nos permite a su vez acotar el error backward. Así también evitaremos este tipo de problemas en el desarrollo del algoritmo. De igual forma, se dice que un método es *estable forward* si la magnitud del error forward de sus soluciones es similar a la del error backward asociado a un método estable backward.

## 1.2. Representación en coma flotante

Antes de introducir el concepto de alta precisión relativa, debemos especificar en qué contexto estamos trabajando. Hemos mencionado como una causa de error el trabajar utilizando una aritmética de precisión finita. Sea  $F$  un subconjunto de los números reales ( $F \subset \mathbb{R}$ ). Diremos que  $F$  es un sistema de numeración en coma flotante si sus elementos presentan la siguiente forma:

$$y = \pm m \times \beta^{e-t}.$$

El significando (también llamado mantisa),  $m$ , es un número entero que cumple  $0 \leq m \leq \beta^{t-1}$ . El sistema  $F$  queda caracterizado por los siguientes números enteros:



- la base  $\beta$ ,
- la precisión  $t$ ,
- y el rango de exponentes  $e_{\min} \leq e \leq e_{\max}$ .

En nuestro caso, la base será 2. Los números representables dependerán del número de bits empleado para almacenar el significando y el exponente. El número de bits utilizados para el significando determinará la precisión, y el número de bits usados para el exponente delimitará el rango de números representables. Lo común es emplear el estándar del IEEE para aritmética en coma flotante. En él se definen dos formatos de números en coma flotante muy utilizados: precisión simple (de 32 bits) y precisión doble (de 64 bits). El primero destina 8 bits al exponente y 23 al significando. El segundo 11 bits al exponente y 52 al significando. En ambos casos el primer bit corresponde al signo.

Si queremos realizar un análisis del error que cometemos al aplicar un algoritmo, existe un valor asociado a  $F$  fundamental: la unidad de redondeo,  $u$ . La unidad de redondeo es el máximo error relativo que se puede cometer al aproximar un número dentro del rango de números representables. En los siguientes capítulos consideraremos el modelo estándar de aritmética en coma flotante. Sean  $x, y \in F$ :

$$fl(x \odot y) = (x \odot y)(1 + \delta), \quad |\delta| < u, \quad \odot = +, -, *, /.$$

donde  $fl(\cdot)$  con un argumento representa el valor calculado de esa expresión. El modelo dice que el valor que se calcula es “tan bueno” como el valor exacto redondeado. A veces puede ser más conveniente utilizar la siguiente variación del modelo. De nuevo, sean  $x, y \in F$ :

$$fl(x \odot y) = \frac{x \odot y}{1 + \delta}, \quad |\delta| < u, \quad \odot = +, -, *, /.$$

### 1.3. Condicionamiento y alta precisión relativa

Además del interés propio que suscita el estudio del error backward (si tenemos estabilidad backward, la solución calculada es la solución de un problema ligeramente perturbado), éste puede servir para dar una estimación del error que definíamos originalmente, el error forward. La relación entre ambos errores está gobernada por el condicionamiento del problema, que mide la sensibilidad de la solución a las perturbaciones en los datos.

Como ejemplo concreto de condicionamiento, podemos considerar el problema de la resolución de un sistema lineal de ecuaciones  $Ax = b$ , con  $A$  matriz cuadrada no singular (puede consultarse, por ejemplo, en la sección 2.2 de [8]).

**Definición 1.5.** El número de condición de una matriz no singular  $A$  es  $\kappa(A) = \|A\| \|A^{-1}\|$ , donde  $\|\cdot\|$  denota una norma matricial.

Los casos más frecuentes corresponden a  $\kappa_{\infty}(A)$  (que usa  $\|\cdot\|_{\infty}$ ) y  $\kappa_2(A)$  (que utiliza  $\|\cdot\|_2$ ). El número de condición depende solamente de la matriz  $A$ , y cuando es muy grande, nos impide dar una cota satisfactoria del error forward de la solución del sistema.

En general, cuando en un problema tenemos definido el error forward, el error backward y el número de condición correspondientes, se busca probar la relación siguiente (véase la sección 1.6 de [15]):

$$\text{error forward} \lesssim \text{número de condición} \times \text{error backward}$$

ya que normalmente es mucho más fácil acotar el error backward que el error forward.

Aunque la solución numérica que obtengamos tenga un error backward pequeño, éste puede ser amplificado por un factor de hasta el tamaño del número de condición, dando lugar a una solución numérica con un error forward excesivo. Así, el condicionamiento se puede presentar como un impedimento intrínseco a la hora de dar una cota del error satisfactoria, en contraste con el error backward, que depende del método utilizado. En la práctica, si nuestro problema lleva asociado una matriz mal condicionada, es de interés buscar algún camino alternativo.

Un ejemplo que también justifica el buscar un planteamiento distinto es el cálculo de valores singulares de una matriz. Si buscamos acotar el error cometido al calcular el vector de valores singulares en norma, aunque veamos que los valores singulares grandes tendrán un error relativo pequeño, muchas veces no podremos asegurar lo mismo para los más próximos a cero (puede verse en [9]). Y estos son los que queremos conocer de forma precisa.

Para obtener resultados con varias cifras significativas correctas, buscaremos que el error de nuestro algoritmo cumpla esta relación:

$$\text{error forward relativo} \leq Ku, \quad \text{para alguna constante } K.$$

Entonces, diremos que los cálculos se han realizado con alta precisión relativa (HRA, de *high relative accuracy*). ¿Es posible lograr la HRA para cualquier problema? Desgraciadamente, la respuesta, en general, es negativa. Como primer ejemplo de cálculo que no puede realizarse con HRA tenemos la evaluación de la expresión  $x + y + z$  (véase [9]). También podemos encontrar un ejemplo entre las clases de matrices con una estructura sencilla, que es la evaluación de determinantes de las matrices de Toeplitz. Una matriz de Toeplitz presenta la siguiente forma:

$$B = \begin{pmatrix} a_0 & a_1 & \cdots & a_{n-2} & a_{n-1} \\ a_{-1} & a_0 & \ddots & & a_{n-2} \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ a_{-n+2} & & \ddots & \ddots & a_1 \\ a_{-n+1} & a_{-n+2} & \cdots & a_{-1} & a_0 \end{pmatrix}$$

Las matrices de Toeplitz se caracterizan porque en cada diagonal aparece siempre el mismo elemento. Aunque están parametrizadas con  $2n - 1$  parámetros, para un  $n$  arbitrariamente grande, no se puede asegurar la HRA. No obstante, para otras matrices con determinada estructura veremos cómo lograrla.

Comenzamos la búsqueda de la alta precisión relativa identificando las causas de la pérdida de la misma. El principal fenómeno que provoca este problema es la cancelación debido a restas de cantidades aproximadas durante el desarrollo de un algoritmo. Una resta de dos cantidades del mismo tamaño puede magnificar errores previos y provocar que los resultados obtenidos no sean válidos. Para entender mejor este fenómeno, vamos a plantear la operación (en aritmética exacta en este caso)  $\hat{x} = \hat{a} - \hat{b}$ , donde  $\hat{a} = a(1 + \Delta a)$ ,  $\hat{b} = b(1 + \Delta b)$  y  $\Delta a$  y  $\Delta b$  son los errores relativos en los datos que intervienen en la operación. Veamos qué podemos decir del error relativo que cometemos al calcular  $\hat{x}$  como aproximación de  $x = a - b$  (siempre que  $x \neq 0$ ):

$$\left| \frac{x - \hat{x}}{x} \right| = \left| \frac{-a\Delta a + b\Delta b}{a - b} \right| \leq \max(|\Delta a|, |\Delta b|) \frac{|a| + |b|}{|a - b|}.$$

Vemos que la cota para el error relativo de  $\hat{x}$  es grande cuando  $|a - b| \ll |a| + |b|$ , o equivalentemente, cuando se produce mucha cancelación al realizar la operación.

Aunque no toda resta tiene que provocar este efecto. Por ejemplo, podemos restar dos datos iniciales que se conozcan de forma precisa sin que se produzca una cancelación perniciosa. En cualquier caso, es un fenómeno que tenemos que tener presente al construir un método con HRA. Existe una condición suficiente para asegurar la alta precisión relativa de un algoritmo (véase [10]). Es la condición *no inaccurate cancellation* (NIC): las operaciones realizadas en el algoritmo son sumas de números del mismo signo, multiplicaciones, divisiones y restas de datos iniciales (entendiendo como resta la diferencia entre dos cantidades del mismo signo). Es decir, están prohibidas las restas (salvo de datos iniciales). Muchos de los algoritmos que vamos a presentar son algoritmos libres de restas (o SF, de *subtraction free*). Un algoritmo SF cumple en particular la condición NIC, y, por tanto, mediante su aplicación obtendremos resultados con HRA.

En esta sección hemos introducido conceptos fundamentales en el análisis del error. No obstante, a la hora de desarrollar un algoritmo hay que tener en cuenta más factores. Si las medidas utilizadas para evitar la propagación de errores acarrear un coste computacional excesivo, las consideraciones previas no podrán ponerse en práctica. Como vamos a describir cálculos matriciales, expresaremos el coste computacional en función del tamaño de la matriz  $n \times n$  en estudio. Normalmente, los algoritmos para resolver los problemas algebraicos enunciados al principio de esta sección se consideran eficientes si realizan  $\mathcal{O}(n^3)$  operaciones elementales. En el siguiente capítulo presentaremos la clase de las P-matrices, que engloba casi todas las matrices para las que hasta ahora se han encontrado algoritmos con HRA, e introduciremos una importante subclase de la misma íntimamente relacionada con las matrices de Nekrasov, las cuales estudiaremos en capítulos posteriores.



# Capítulo 2

## Clases de matrices relacionadas

Para ciertas clases de matrices, se pueden realizar muchos cálculos con alta precisión relativa independientemente del condicionamiento. Una justificación para este hecho es que estas matrices tienen detrás una estructura especial y traen asociados unos parámetros naturales, que son los que se emplean en los algoritmos para lograr la alta precisión relativa (HRA). Casi todas las clases de matrices estructuradas para las que se han encontrado hasta ahora algoritmos con HRA están muy relacionadas con subclases de P-matrices (véase [9]). Este hecho está probablemente relacionado con el de que la condición suficiente NIC para HRA propuesta en el capítulo anterior depende de una cuestión de signos.

En este capítulo comenzamos por tanto introduciendo el concepto de P-matriz en la primera sección. Incluimos algunas caracterizaciones y presentamos su aplicación a los problemas de complementariedad lineal. En la siguiente sección se presentan y relacionan los conceptos de Z-matriz, M-matriz y dominancia diagonal y en la última se introducen las H-matrices, clase que engloba a las matrices de Nekrasov (objeto de nuestro estudio en los capítulos siguientes).

### 2.1. P-matrices

Comencemos definiendo la importante clase de las P-matrices con la que estarán relacionadas todas las clases de matrices usadas en esta memoria.

**Definición 2.1.** Una matriz  $A = (a_{ij})_{1 \leq i, j \leq n}$  es una P-matriz si todos sus menores principales son positivos.

Recordemos que los menores principales de una matriz son aquellos que se forman eligiendo filas y columnas con el mismo índice. Vamos a introducir una notación que usaremos posteriormente. Definimos  $Q_{k,n}$  como el conjunto de sucesiones estrictamente crecientes de  $k$  números naturales menores o iguales que  $n$ . Sean  $\alpha = (\alpha_1, \dots, \alpha_k)$ ,  $\beta = (\beta_1, \dots, \beta_k)$  dos sucesiones de  $Q_{k,n}$ . Entonces  $A[\alpha|\beta]$  denota a la submatriz  $k \times k$  de  $A$  conteniendo las filas  $\alpha_1, \dots, \alpha_k$  y columnas  $\beta_1, \dots, \beta_k$ . Si  $\alpha = \beta$  la submatriz  $A[\alpha|\alpha]$  se denomina principal y también se representa de forma abreviada como  $A[\alpha]$ .

La Definición 2.1 es la definición más común de P-matriz. No obstante, existen muchas caracterizaciones. En el siguiente resultado (página 120 de [17]) se presentan las siguientes:

**Teorema 2.2.** Sea  $A = (a_{ij})_{1 \leq i, j \leq n}$ . Las siguientes condiciones son equivalentes:

- i)  $A$  es P-matriz.

- ii) Para todo  $x \in \mathbb{R}^n$  no nulo existe  $k \in \{1, \dots, n\}$  tal que  $x_k(Ax)_k > 0$ .
- iii) Para todo  $x \in \mathbb{R}^n$  no nulo existe una matriz  $D$  diagonal positiva tal que  $x^T(DA)x > 0$ .
- iv) Para todo  $x \in \mathbb{R}^n$  no nulo existe una matriz  $D$  diagonal no negativa tal que  $x^T(DA)x > 0$ .
- v) Todo valor propio real de cualquier submatriz principal de  $A$  es positivo.

Además de estas caracterizaciones, podemos encontrar otras que relacionan a las P-matrices directamente con sus aplicaciones. Como ejemplo de aplicación en programación lineal, tenemos el problema de complementariedad lineal (LCP):

$$\begin{aligned} \text{Dados } r \in \mathbb{R}^n \text{ y } M \in \mathbb{R}^{n \times n}, \text{ encontrar (o deducir que no existe) } z \in \mathbb{R}^n \text{ tal que} \\ w = r + Mz, \text{ con } w \geq 0, z \geq 0, z^T w = 0. \end{aligned} \quad (2.1)$$

Pues bien, la existencia y unicidad de solución del problema de complementariedad lineal caracteriza a una P-matriz (página 274 de [4]):

**Teorema 2.3.**  $M = (m_{ij})_{1 \leq i, j \leq n}$  es una P-matriz si y solo si el problema de complementariedad lineal (2.1) tiene solución única para todo  $r \in \mathbb{R}^n$ .

Antes de pasar a la siguiente sección, cabe destacar que una importante aplicación de las matrices de Nekrasov, a las que dedicaremos gran parte de la memoria, es la obtención de cotas de error en el problema de complementariedad lineal.

## 2.2. Z-matrices, M-matrices y dominancia diagonal

Las M-matrices constituyen una subclase de P-matrices presente en numerosas aplicaciones. Se encuentran, por ejemplo, en teoría de probabilidad, en el estudio de cadenas de Markov, en análisis numérico, al buscar cotas de valores propios, o al establecer criterios de convergencia de métodos iterativos para la resolución de grandes sistemas lineales de ecuaciones con matriz asociada hueca (o *sparse*, que es una matriz en la que predominan los elementos nulos). Entre estas aplicaciones, cabe destacar el papel que juegan las M-matrices en el campo de la economía. Su aparición en diversos modelos desembocó en el estudio de las mismas por parte de los economistas. Como ejemplo fundamental tenemos el modelo input-output o modelo de Leontief, denominado así por su precursor Wassily Leontief, premio nobel de economía en 1973. La novedad del trabajo de Leontief radica precisamente en emplear el álgebra lineal para describir una economía en la que diversos sectores producen y consumen bienes, y estudiar cómo sus diversas partes encajaban e interactuaban.

Las M-matrices para las que vamos a lograr algoritmos con HRA cumplen la condición de dominancia diagonal. Vamos a comenzar definiendo los correspondientes conceptos básicos:

**Definición 2.4.** Una matriz  $A = (a_{ij})_{1 \leq i, j \leq n}$  se dice Z-matriz si  $a_{ij} \leq 0 \quad \forall (i, j)$  tal que  $i \neq j$ .

Es decir, es una matriz cuyos elementos extradiagonales son no positivos.

**Definición 2.5.** Una Z-matriz  $A = (a_{ij})_{1 \leq i, j \leq n}$  se dice M-matriz si puede representarse de la forma:  $A = sI - B$ , con  $B \geq 0$  y  $s \geq \rho(B)$  (donde  $\rho(B)$  es el radio espectral de  $B$ ). Si se cumple  $s > \rho(B)$ , la matriz es una M-matriz no singular.

**Nota 2.6.** Observemos que las entradas diagonales de una M-matriz son no negativas (véase página 159 de [23]).

El siguiente resultado, que corresponde al Lema 4.1 del capítulo 6 de [4], da una relación entre las M-matrices y las M-matrices no singulares que nos será de utilidad posteriormente.

**Lema 2.7.** Sea  $A = (a_{ij})_{1 \leq i, j \leq n}$  una Z-matriz. Entonces  $A$  es M-matriz si y solo si  $A + \varepsilon I$  es una M-matriz no singular para todos los escalares  $\varepsilon > 0$ .

A continuación presentamos el tercer concepto importante de esta sección.

**Definición 2.8.** Una matriz  $A = (a_{ij})_{1 \leq i, j \leq n}$  se dice matriz diagonalmente dominante por filas (d.d.) si cumple:

$$|a_{ii}| \geq \sum_{i \neq j} |a_{ij}|, \quad i = 1, \dots, n.$$

Si  $A^T$  es d.d.,  $A$  se dice matriz diagonalmente dominante por columnas. Si la desigualdad es estricta para todas las filas de  $A$  (resp. de  $A^T$ ), la matriz es estrictamente diagonalmente dominante por filas y la denotamos SDD (resp. estrictamente diagonalmente dominante por columnas). Es bien conocido que una matriz SDD es no singular (Teorema de Levy-Desplanques, véase 5.6.17 de [16]). Para argumentarlo, se pueden usar, por ejemplo, discos de Gershgorin para demostrar que 0 no puede ser valor propio.

Antes hemos mencionado la variedad de aplicaciones de las M-matrices. Una curiosidad acerca de las M-matrices no singulares guarda relación con esta diversidad de aplicaciones, y es la gran cantidad de caracterizaciones que poseen. En el capítulo 6 del libro [4] aparecen 50 caracterizaciones. A continuación presentamos varias debido a su importancia. En particular, introduciremos las caracterizaciones  $D_{16}$ ,  $A_1$ ,  $E_{17}$ ,  $N_{38}$ ,  $M_{35}$ ,  $E_{18}$ ,  $C_{10}$  que corresponde a *ii*) – *viii*). Recordemos previamente, que un menor principal de una matriz  $A$  se llama director si usa filas y columnas iniciales, es decir, si es de la forma  $\det A[1, \dots, k]$  para algún  $k \in \{1, \dots, n\}$ .

**Teorema 2.9.** Sea  $A = (a_{ij})_{1 \leq i, j \leq n}$  una Z-matriz. Entonces, las siguientes condiciones son equivalentes:

- i*)  $A$  es una M-matriz no singular.
- ii*) Todos los valores propios reales de  $A$  son positivos.
- iii*) Todos los menores principales de  $A$  son positivos.
- iv*) Todos los menores principales directores de  $A$  son positivos.
- v*)  $A$  es invertible, y  $A^{-1}$  es no negativa ( $A^{-1} \geq 0$ ).
- vi*)  $A$  tiene diagonal positiva y existe una matriz  $D$  diagonal tal que  $AD$  es una matriz de diagonal estrictamente dominante.
- vii*)  $A = LU$ , donde  $L$  es una matriz triangular inferior,  $U$  es una matriz triangular superior y todos los elementos diagonales de ambas matrices son positivos.
- viii*)  $A$  es no singular, y  $A + D$  es no singular para cualquier matriz  $D$  diagonal positiva.

Vamos a presentar ahora las relaciones entre las definiciones 2.4, 2.5 y 2.8 de esta sección. Observemos que, utilizando la implicación  $vi) \Rightarrow i)$  con  $D = I$  (matriz identidad) en el Teorema 2.9 deducimos el siguiente resultado:

**Proposición 2.10.** *Si  $A = (a_{ij})_{1 \leq i, j \leq n}$  es una Z-matriz SDD con entradas diagonales positivas, entonces  $A$  es una M-matriz no singular.*

Terminamos esta sección con la caracterización que relaciona las definiciones 2.4, 2.5 y 2.8. Observemos que el recíproco es inmediato y que la implicación directa se puede obtener de la Proposición 2.10 y el Lema 2.7. De hecho, si  $A$  es una Z-matriz d.d. con entradas diagonales no negativas, entonces  $A + \varepsilon I$  es una Z-matriz SDD con entradas diagonales positivas para cualquier escalar  $\varepsilon > 0$ . Así, otra vez por la Proposición 2.10,  $A + \varepsilon I$  es una M-matriz no singular para cualquier  $\varepsilon > 0$ , y, por el Lema 2.7,  $A$  es una M-matriz.

**Teorema 2.11.** *Sea  $A = (a_{ij})_{1 \leq i, j \leq n}$  una matriz d.d. con entradas diagonales no negativas. Entonces  $A$  es Z-matriz si y solo si  $A$  es M-matriz.*

### 2.3. H-matrices

En la sección anterior hemos introducido las M-matrices. En muchos problemas teóricos y prácticos (véase [4]), aparecen otros tipos de matrices, que a pesar de no ser M-matrices, guardan cierta relación con éstas, la cual nos puede servir de guía para lograr algoritmos con HRA. Para ilustrar cómo puede aparecer esta relación, a continuación vamos a definir el concepto de H-matriz, una clase de matrices que engloba a las M-matrices. Para dar la definición de una forma clara, conviene primero introducir la noción de matriz de comparación:

**Definición 2.12.** La matriz de comparación  $M(A) = (m_{ij})_{1 \leq i, j \leq n}$  de una matriz compleja  $A = (a_{ij})_{1 \leq i, j \leq n}$  se define de la siguiente forma:

$$m_{ij} = \begin{cases} |a_{ij}| & \text{si } j = i, \\ -|a_{ij}| & \text{si } j \neq i. \end{cases}$$

**Definición 2.13.** Una matriz  $A = (a_{ij})_{1 \leq i, j \leq n}$  compleja se dice H-matriz si su matriz de comparación es una M-matriz no singular.

Observemos que una matriz de comparación  $M(A)$  es una Z-matriz con diagonal no negativa cuyas entradas tienen los mismos valores absolutos que los de las entradas correspondientes de  $A$ . Por tanto, la estructura de signos de la matriz de comparación es la de una Z-matriz con diagonal no negativa. Una H-matriz es M-matriz si y solo si su matriz de comparación coincide con ella misma. En el caso de las M-matrices no singulares hemos visto numerosas caracterizaciones. Para las H-matrices, existe una caracterización (véase p. 124 de [17]) que también la relaciona con las matrices estrictamente diagonalmente dominantes (SDD).

**Teorema 2.14.** *Sea  $A = (a_{ij})_{1 \leq i, j \leq n}$ . Entonces,  $A$  es H-matriz si y solo si existe una matriz diagonal  $D$  tal que  $AD$  es una matriz estrictamente diagonalmente dominante por filas.*

A partir del capítulo cuatro consideraremos una subclase de las H-matrices llamadas matrices de Nekrasov.



## Capítulo 3

# Eliminación Gaussiana y M-matrices diagonalmente dominantes

En el capítulo anterior hemos introducido la clase formada por las M-matrices, y hemos expuesto la relación de la misma con el concepto de positividad. Ya era conocido hace tiempo que la dominancia diagonal tenía ventajas en el estudio clásico del error de la resolución de sistemas (véase [24]). En este capítulo, se muestra que también tienen ventajas de cara a la obtención de algoritmos con alta precisión relativa (HRA). En particular, vamos a ver cómo el algoritmo de eliminación Gaussiana, empleado de forma adecuada, es clave para lograr resolver varios problemas del álgebra lineal numérica logrando HRA para las M-matrices d.d. (diagonalmente dominantes). La forma de lograrlo será obtener una factorización LDU asegurando la HRA.

Por tanto, en la primera sección se recuerda el algoritmo de eliminación Gaussiana, los pivotajes parcial y total, así como el método de Gauss-Jordan para calcular inversas. Después, se describen dos técnicas de pivotaje para M-matrices diagonalmente dominantes que, empleadas en el proceso de eliminación Gaussiana partiendo de una parametrización adecuada de la M-matriz d.d., dan lugar a una factorización LDU calculada con HRA. Por último, se concluye el capítulo con un apartado dedicado a la experimentación numérica que ilustra las diferentes consecuencias de aplicar las dos técnicas de pivotaje descritas.

### 3.1. Eliminación Gaussiana, pivotaje y método de Gauss-Jordan

Dada  $A = (a_{ij})_{1 \leq i, j \leq n}$  matriz no singular, la eliminación Gaussiana es un procedimiento de eliminación matricial empleado para hacer ceros debajo de su diagonal. Consiste en una sucesión de  $n - 1$  pasos que dan lugar a una sucesión de matrices de la forma:

$$A = A^{(1)} \rightarrow \tilde{A}^{(1)} \rightarrow A^{(2)} \rightarrow \tilde{A}^{(2)} \rightarrow \dots \rightarrow A^{(n)} = \tilde{A}^{(n)} = DU,$$

donde  $A^{(k)}$  tiene ceros por debajo de la diagonal en las primeras  $k - 1$  columnas y  $DU$  es triangular superior. Habiendo calculado  $A^{(k)}$ , reordenamos sus filas y/o columnas para obtener  $\tilde{A}^{(k)}$  mediante una estrategia de pivotaje. Una estrategia de pivotaje en el proceso de eliminación Gaussiana consiste en una reordenación de las filas y/o columnas de  $A$  en cada paso para seleccionar cuál será el elemento pivote que emplearemos para hacer ceros en la siguiente iteración.

En el esquema, su aplicación se produce en el paso de  $A^{(k)}$  a  $\tilde{A}^{(k)}$ . Dos estrategias muy utilizadas son el pivotaje parcial (reordenación solamente de filas, consiste en buscar un elemento de mayor módulo en la columna  $A^{(k)}[k, \dots, n|k]$  en la que haremos ceros en el siguiente paso) y el pivotaje total o completo (reordenación de filas y columnas, se busca un elemento pivote de módulo máximo en toda la submatriz  $A^{(k)}[k, \dots, n]$ ). Sea cual sea la estrategia elegida, necesitamos que el elemento pivote,  $\tilde{a}_{kk}^{(k)}$ , sea no nulo. Cuando en una estrategia de pivotaje utilizamos siempre el mismo índice de filas y columnas decimos que usamos un pivotaje simétrico.

Aplicando la permutación adecuada según la estrategia de pivotaje que elijamos, llegamos a  $\tilde{A}^{(k)}$ . El elemento  $\tilde{a}_{kk}^{(k)}$  será el pivote elegido por la estrategia de pivotaje, y se empleará para hacer ceros en la columna  $k$ . Para ello, restaremos múltiplos de la fila  $k$  a las filas de debajo, obteniendo así la matriz  $A^{(k+1)} = (a_{ij}^{(k+1)})_{1 \leq i, j \leq n}$

$$a_{ij}^{(k+1)} = \begin{cases} \tilde{a}_{ij}^{(k)}, & \text{si } 1 \leq i \leq k, \\ \tilde{a}_{ij}^{(k)} - \frac{\tilde{a}_{ik}^{(k)}}{\tilde{a}_{kk}^{(k)}} \tilde{a}_{kj}^{(k)}, & \text{si } k < i \leq n. \end{cases}$$

Si nuestro objetivo es calcular la inversa de una matriz no singular, se puede emplear el bien conocido método de Gauss-Jordan. El método de Gauss-Jordan sin pivotaje da lugar a una sucesión de matrices de la forma:

$$A = A^{(1)} \rightarrow A^{(2)} \rightarrow \dots \rightarrow A^{(n)} = DU \rightarrow A^{(n+1)} \rightarrow \dots \rightarrow A^{(2n-1)} = D \rightarrow A^{(2n)} = I,$$

donde en los primeros  $n - 1$  pasos hemos empleado eliminación Gaussiana sin pivotaje para hacer ceros por debajo de la diagonal de la matriz, y en los  $k - 1$  pasos siguientes hemos utilizado de nuevo eliminación Gaussiana sin pivotaje para hacer ceros por encima de la diagonal de la matriz. El último paso consiste en reescalar la matriz para lograr la matriz identidad. Si aplicamos las mismas operaciones elementales a la matriz identidad (es decir,  $B = B^{(1)} = I$ ), al obtener la matriz  $A^{(2n)} = I$  lograremos también la matriz  $B^{(2n)} = A^{-1}$ . Análogamente a la eliminación Gaussiana, el método de Gauss-Jordan también se puede aplicar con estrategias de pivotaje.

## 3.2. Descomposiciones reveladoras del rango, parametrización y pivotaje

Volviendo a nuestro problema de aplicar algoritmos de forma precisa, recordemos que muchas veces la clave consiste en buscar distintas factorizaciones o parametrizaciones del problema. Para hallar los valores singulares de una matriz, se emplea la llamada descomposición reveladora del rango (o *rank revealing decomposition*, RRD). Esta consiste en una descomposición de la matriz de la forma  $A = XDY^T$ , donde  $X, Y$  son matrices bien condicionadas y  $D$  es una matriz diagonal. En [10] se presenta un algoritmo que realiza  $\mathcal{O}(n^3)$  operaciones elementales para obtener con HRA los valores singulares de una matriz  $n \times n$  a partir de su RRD.

En el caso de las M-matrices, se considera como RRD la descomposición LDU obtenida tras una adecuada estrategia de pivotaje, en la que  $L$  es una matriz triangular inferior y  $U$  una matriz triangular superior. Además, los elementos diagonales de  $D$  son positivos, y tanto los de  $L$  como los de  $U$  son todos unos. Podemos obtener esta descomposición con HRA logrando unas matrices  $L$  y  $U$  bien condicionadas, por lo que habremos calculado una RRD

que podremos emplear para calcular los valores singulares de la matriz de forma precisa. Para calcular la descomposición emplearemos la eliminación Gaussiana con una adecuada estrategia de pivotaje.

En esta descripción de la eliminación Gaussiana no hemos tenido en cuenta la estructura de la matriz. Para obtener una factorización LDU de una M-matriz diagonalmente dominante (d.d.), con  $L$  y  $U$  bien condicionadas, es necesario realizar cambios en el planteamiento descrito. Por un lado, al trabajar directamente con los elementos de la M-matriz, el algoritmo de eliminación Gaussiana puede dar lugar a errores por cancelaciones debido a las restas que se llevan a cabo. Para evitar este fenómeno, en vez de trabajar directamente con los elementos de la matriz se utiliza una parametrización de la misma. Para las M-matrices diagonalmente dominantes, unos **parámetros adecuados** son las sumas de los elementos de cada fila y sus elementos extradiagonales. Estos parámetros pueden tener una interpretación física natural [2].

Además, si elegimos sin cuidado la estrategia de pivotaje, podemos perder la estructura de M-matriz en el desarrollo del algoritmo. Con el fin de evitar este problema, se utilizan las llamadas estrategias de pivotaje simétrico. La idea consiste en realizar en cada paso la misma permutación tanto de filas como de columnas. Así, teniendo en cuenta que estas permutaciones simultáneas de filas y de columnas preservan la propiedad de ser M-matriz y que por [12] el complemento de Schur de M-matrices también preserva la propiedad, concluimos que todas las submatrices  $\tilde{A}^{(k)}[k, \dots, n]$  con  $k \in \{1, \dots, n-1\}$  serán M-matrices. Obtendremos una factorización de la forma  $PAP^T = LDU$  con  $P$  una matriz de permutación. A continuación introducimos dos estrategias de pivotaje simétrico que pueden servir para obtener una RRD de la forma ya descrita.

La primera estrategia de pivotaje simétrico se denomina pivotaje simétrico total, y consiste en elegir un elemento de módulo máximo en la diagonal. En el caso de las M-matrices diagonalmente dominantes, esta estrategia coincide con pivotaje total. En [11] se presenta un algoritmo que emplea esta estrategia para lograr una descomposición LDU de una M-matriz d.d. Dados los elementos extradiagonales  $a_{ij}$ , con  $i \neq j$ , y el vector de sumas de filas  $s$ , con  $s_i = \sum_{j=1}^n a_{ij}$ , el Algoritmo 1 da la factorización LDU de una M-matriz d.d. usando pivotaje simétrico total, sin hacer uso de restas (SF). Las salidas del algoritmo son la matriz  $P$ , la matriz  $L$  y la matriz  $DU$  (estas dos últimas almacenadas en  $A$ ) de la factorización  $PAP^T = LDU$  mediante pivotaje simétrico total. Si necesitamos factorizar una M-matriz diagonalmente dominante por columnas, bastaría con aplicar el algoritmo a  $A^T$ . En ese caso, tendríamos como parámetros los elementos extradiagonales de  $A$  así como la suma de los elementos de cada columna, que se corresponderían con las sumas de las entradas de las filas de  $A^T$ .

La segunda técnica de pivotaje simétrico que presentamos para hallar una factorización LDU de una M-matriz diagonalmente dominante con  $L$  y  $U$  bien condicionadas se encuentra descrita de forma detallada en [25]. Aquí introduciremos esta técnica de pivotaje y el algoritmo para obtener una factorización LDU de una M-matriz diagonalmente dominante por columnas.

La estrategia de pivotaje simétrico se denomina *maximal absolute diagonal dominance* (m.a.d.d.) y se basa en elegir como pivote en el paso  $k$  ( $k \in \{1, \dots, n-1\}$ ) una fila  $i_k \geq k$  que cumpla:

$$|a_{i_k i_k}^{(k)}| - \sum_{j \geq k, j \neq i_k} |a_{i_k j}^{(k)}| = \max_{k \leq i \leq n} \{ |a_{ii}^{(k)}| - \sum_{j \geq k, j \neq i} |a_{ij}^{(k)}| \}.$$

Por el Teorema 2 de [1] una M-matriz  $A$  siempre tiene un elemento diagonal  $a_{ii}$  que verifica  $|a_{ii}| > \sum_{j \neq i} |a_{ij}|$ . Por tanto, el pivote que elijamos cumplirá  $a_{i_k i_k} \neq 0$  (será un pivote válido).

**Algoritmo 1** Eliminación Gaussiana para M-matrices d.d. utilizando pivotaje simétrico total

---

**Entradas:**  $A = (a_{ij})(i \neq j), s$  ▷  $s$  es el vector de sumas de las filas de  $A$   
 $P = I_n$  ▷ la matriz de permutación  
**for**  $k = 1 : n - 1$   
  **for**  $i = k : n$   
     $a_{ii} = s_i - \sum_{j \geq k, j \neq i}^n a_{ij}$   
  **end for**  
  Buscar  $t$  tal que  $a_{tt} = \max_{i \geq k} \{a_{ii}\}$   
  Elegir  $P_k$  matriz de permutación que intercambia la fila  $t$  y la fila  $k$ .  
  Actualizar  $P = P_k P, A = P_k A P_k^T, s = P_k s$   
  **for**  $i = k + 1 : n$   
     $a_{ik} = a_{ik} / a_{kk}$   
     $s_i = s_i - a_{ik} s_k$   
    **for**  $j = k + 1 : n$   
      **if**  $i \neq j$   
         $a_{ij} = a_{ij} - a_{ik} a_{kj}$   
      **end if**  
    **end for**  
  **end for**  
**end for**

---

Dados los elementos extradiagonales  $a_{ij}$ , con  $i \neq j$ , y el vector de sumas de columnas  $c$ , con  $c_i = \sum_{j=1}^n a_{ij}$ , el Algoritmo 2 es también SF y da la factorización LDU de una M-matriz diagonalmente dominante por columnas empleando la estrategia de pivotaje m.a.d.d. Las salidas del algoritmo son la matriz  $P$ , la matriz  $L$  y la matriz  $DU$  (estas dos últimas almacenadas en  $A$ ) de la factorización  $PAP^T = LDU$  mediante pivotaje m.a.d.d.

Hemos presentado dos estrategias para obtener una factorización LDU que sirve como RRD de una M-matriz diagonalmente dominante. No obstante, el siguiente teorema muestra una importante diferencia entre ambas (Proposition 3.2 de [25]):

**Teorema 3.1.** *Sea  $A = (a_{ij})_{1 \leq i, j \leq n}$  una M-matriz diagonalmente dominante por filas o columnas y sea  $P$  una matriz de permutación asociada a aplicar la estrategia de pivotaje m.a.d.d. de  $A$  o  $A^T$ , respectivamente. Entonces  $PAP^T = LDU$ , donde  $L$  es una matriz triangular inferior de diagonal dominante por columnas y  $U$  es una matriz triangular superior de diagonal dominante por filas.*

Si utilizamos pivotaje simétrico total con una M-matriz d.d. por columnas obtendremos una factorización LDU en la que la matriz  $L$  es d.d. por columnas. No obstante, no podremos asegurar que  $U$  sea d.d. por filas, sino solo que el elemento diagonal es mayor en módulo que los restantes de su fila. En la siguiente sección se muestran ejemplos en los cuales la matriz  $U$  obtenida empleando pivotaje simétrico total no es d.d. por filas, y su número de condición es considerablemente mayor que el de la  $U$  obtenida empleando pivotaje m.a.d.d. Para una M-matriz d.d. por filas, considerando su matriz traspuesta se deduce que solamente tendremos asegurada la dominancia diagonal por filas de  $U$  en el caso de estar utilizando pivotaje simétrico total. La dominancia diagonal de las matrices  $L$  y  $U$  implica que ambas están muy bien condicionadas (Proposition 2.1 de [25]).

**Algoritmo 2** Eliminación Gaussiana para M-matrices d.d. utilizando pivotaje m.a.d.d.

---

**Input:**  $A = (a_{ij})(i \neq j), c$  ▷  $c$  es el vector de sumas de las columnas de  $A$   
 $P = I_n$  ▷ la matriz de permutación

**for**  $i = 1 : n$   
 $s_i = \sum_{j=1, j \neq i}^n a_{ji}$   
 $a_{ii} = c_i - s_i$   
 $p_i = \sum_{j=1}^n a_{ij}$   
**end for**

**for**  $k = 1 : n - 1$   
 Buscar  $t$  tal que  $p_t = \max_{i \geq k} \{p_i\}$   
 Elegir  $P_k$  matriz de permutación que intercambia la fila  $t$  y la fila  $k$ .  
 Actualizar  $P = P_k P, A = P_k A P_k^T, c = P_k c, p = P_k p$   
**for**  $i = k + 1 : n$   
 $a_{ik} = a_{ik} / a_{kk}$   
 $c_i = c_i - a_{ki} c_k / a_{kk}$   
 $p_i = p_i - a_{ik} p_k$   
**for**  $j = k + 1 : n$   
**if**  $i \neq j$   
 $a_{ij} = a_{ij} - a_{ik} a_{kj}$   
**end if**  
**end for**  
**end for**

**for**  $j = k + 1 : n$   
 $s_j = \sum_{i \geq k+1, i \neq j}^n a_{ij}$   
 $a_{jj} = c_j - s_j$   
**end for**

**end for**

---

**Teorema 3.2.** Sea  $T = (t_{ij})_{1 \leq i, j \leq n}$  una matriz triangular diagonalmente dominante por filas (respectivamente columnas) cuyos elementos diagonales son todo unos. Entonces  $\kappa_\infty(T) \leq n^2$  (respectivamente  $\kappa_\infty(T) \leq 2n$ ).

De nuevo, tenemos que tener en cuenta la eficiencia del algoritmo que emplee una de estas estrategias de pivotaje. La implementación de cualquiera de las dos estrategias supone un aumento de  $\mathcal{O}(n^3)$  operaciones elementales sobre el coste computacional del algoritmo de eliminación Gaussiana. No obstante, la estrategia de pivotaje m.a.d.d. se puede implementar para esta clase de matrices de modo que podamos obtener una factorización LDU con un coste computacional aún menor y conseguir  $L$  y  $U$  matrices diagonalmente dominantes. En [3], se presenta cómo se realiza la implementación de la estrategia de forma que añade  $\mathcal{O}(n^2)$  operaciones elementales al coste computacional de la eliminación Gaussiana.

### 3.3. Experimentación numérica

En esta sección vamos a mostrar resultados obtenidos empleando una implementación en MATLAB de los algoritmos presentados en este capítulo. Para ello, vamos a comenzar definiendo una familia de matrices que nos servirá de batería de ejemplos. Dado  $n \in \mathbb{N}$ , definimos la matriz  $A_n$  de la siguiente manera:

$$A_n = \begin{pmatrix} 2n & -2n & 0 & \cdots & \cdots & 0 \\ 0 & 2n+1 & -2 & \cdots & -2 & -3 \\ \vdots & -2n & 2n & 0 & \cdots & 0 \\ \vdots & \vdots & 0 & \ddots & & \vdots \\ \vdots & \vdots & \vdots & & 2n & 0 \\ 0 & -2n & 0 & \cdots & 0 & 2n \end{pmatrix}.$$

En la Tabla 3.1 vemos el condicionamiento en norma infinito de la matriz  $L$  obtenida empleando eliminación Gaussiana sin pivotaje, así como los dos pivotajes específicos para  $M$ -matrices presentados en este capítulo.

n	sin piv.	total	madd
5	7.2511	7.2511	2.89
10	18.463	18.463	3.4225
15	31.1346	31.1346	3.61
20	44.7616	44.7616	3.7056
25	59.102	59.102	3.7636
30	74.0121	74.0121	3.8025
35	89.3968	89.3968	3.8304
40	105.1883	105.1883	3.8514
45	121.3356	121.3356	3.8678
50	137.7993	137.7993	3.8809

Tabla 3.1: Condicionamiento en norma infinito de  $L$

En este caso, emplear pivotaje total da el mismo resultado que no emplear una estrategia de pivotaje, lo que ocasiona que el condicionamiento empeore a medida que crece  $n$ . No obstante, utilizando el pivotaje m.a.d.d. obtenemos siempre un condicionamiento muy bueno.

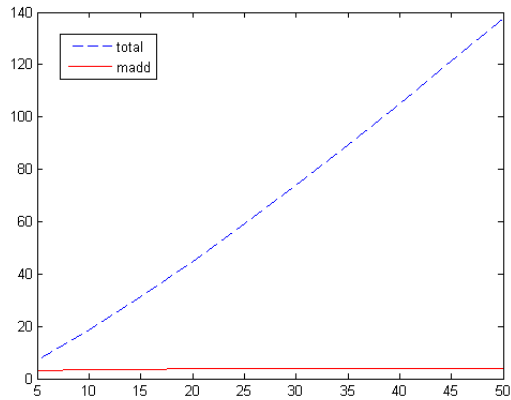
A continuación vamos a estudiar también el comportamiento de la matriz  $U$  (cabe esperar que sea muy bueno en todos los casos).

n	sin piv.	total	madd
5	5.6652	2.9988	4
10	6.8759	4.4115	4
15	7.6249	5.262	4
20	8.1686	5.8677	4
25	8.5957	6.3371	4
30	8.9475	6.7197	4
35	9.2465	7.0424	4
40	9.5066	7.3212	4
45	9.7367	7.5667	4
50	9.9431	7.7858	4

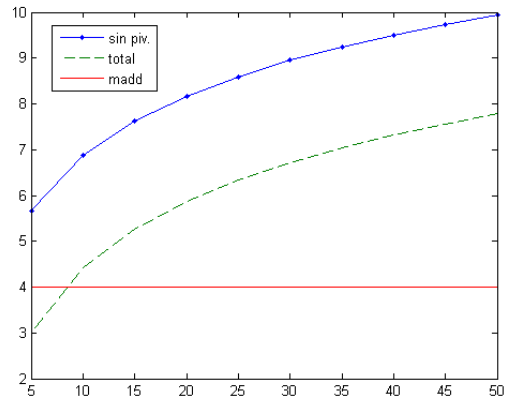
Tabla 3.2: Condicionamiento en norma infinito de  $U$

En la Tabla 3.2 vemos que efectivamente el resultado obtenido es muy bueno en cualquier caso. La diferencia radica en la matriz  $L$ , como se observa en la Tabla 3.1. No obstante,

empleando el pivotaje m.a.d.d. el condicionamiento de la matriz  $U$  también se mantiene prácticamente constante al aumentar  $n$ . En las siguientes gráficas se muestra el comportamiento del condicionamiento de  $L$  y  $U$  según la estrategia de pivotaje empleada:



$\kappa_\infty(L)$



$\kappa_\infty(U)$





# Capítulo 4

## Matrices de Nekrasov

En este capítulo introducimos el concepto de matriz de Nekrasov, que jugará un importante papel en el resto de la memoria. Se trata de una clase de matrices que generaliza la de las estrictamente diagonalmente dominantes. Al parecer, la condición que caracteriza a las matrices de Nekrasov fue introducida por el matemático ruso Pavel Nekrasov en el año 1892 en [22].

Comenzamos el capítulo definiendo el concepto de matriz de Nekrasov, proponiendo una parametrización adecuada para HRA y recopilando propiedades conocidas de estas matrices. En la sección 4.2 encontramos una matriz diagonal sencilla que transforma una Z-matriz de Nekrasov con diagonal positiva en una M-matriz diagonalmente dominante. También, en la última sección, se explica cómo obtener la parametrización adecuada (vista en el capítulo 3) de esta matriz transformada con HRA a partir de la parametrización que hemos introducido en este capítulo para una matriz de Nekrasov.

### 4.1. Definición y parametrización

Vamos a introducir las matrices de Nekrasov (véase [27]). Para ello necesitamos una notación previa: Sea  $A = (a_{ij})_{1 \leq i, j \leq n}$  una matriz compleja tal que  $a_{ii} \neq 0$ . Se define  $h_i(A)$  con  $i = 1, \dots, n$  de la siguiente forma:

$$h_i(A) = \begin{cases} \sum_{j=2}^n |a_{1j}|, & \text{si } i = 1, \\ \sum_{j=1}^{i-1} |a_{ij}| \frac{h_j(A)}{|a_{jj}|} + \sum_{j=i+1}^n |a_{ij}|, & \text{si } 2 \leq i \leq n, \\ \sum_{j=1}^{n-1} |a_{nj}| \frac{h_j(A)}{|a_{jj}|}, & \text{si } i = n. \end{cases} \quad (4.1)$$

**Definición 4.1.** Una matriz compleja  $A = (a_{ij})_{1 \leq i, j \leq n}$  se llama matriz de Nekrasov si cumple la condición  $|a_{ii}| > h_i(A)$  para  $i = 1, \dots, n$ .

Esta es una condición suficiente para que una matriz sea no singular [27], por lo que tendrá sentido plantear el cálculo de  $A^{-1}$ .

Algunas aplicaciones recientes de las matrices de Nekrasov pueden consultarse en los siguientes artículos: [6, 14, 19, 20, 26, 29, 30].

Existe una clase de matrices muy relacionada con las matrices de Nekrasov y que la generaliza, la clase compuesta por las matrices de Gudkov. Una matriz se dice de Gudkov si existe una permutación simultánea de filas y de columnas tal que al actuar sobre ella la lleva a forma de matriz de Nekrasov. Es decir, una matriz  $A$  es una matriz de Gudkov si existe una matriz de permutación  $P$  tal que  $PAP^T$  es una matriz de Nekrasov. En particular, una matriz de Gudkov también es no singular. No obstante, el método de alta precisión relativa que vamos a desarrollar para  $Z$ -matrices de Nekrasov con diagonal positiva no puede extenderse en general a las matrices de Gudkov con la misma estructura de signos. Esto es debido a que la definición de matriz de Nekrasov depende del orden de filas, y nuestra parametrización lleva intrínseca dicha dependencia. En general, al permutar una matriz de Nekrasov su parametrización variará. Para ver más propiedades de las matrices de Gudkov puede consultarse [27].

Para lograr alta precisión relativa al trabajar con  $Z$ -matrices de Nekrasov  $n \times n$  con diagonal positiva, emplearemos los siguientes  $n^2$  parámetros:

$$\begin{cases} a_{ij}, & i \neq j, \\ \Delta_j(A) := a_{jj} - h_j(A), & j = 1, \dots, n. \end{cases} \quad (4.2)$$

Observemos que, a partir de los  $n^2$  signos dados en (4.2), podemos caracterizar las  $Z$ -matrices de Nekrasov con diagonal positiva. De hecho,  $A$  cumple dicha propiedad si y solo si los  $n^2 - n$  primeros parámetros (los elementos extradiagonales,  $a_{ij}$  con  $i \neq j$ ) son no positivos y los  $n$  últimos parámetros ( $\Delta_j(A)$  con  $j = 1, \dots, n$ ) son positivos.

## 4.2. Escalado para dominancia diagonal

Las matrices de Nekrasov están íntimamente relacionadas con las matrices d.d. Nos vamos a aprovechar de esta relación para resolver nuestro problema del cálculo de la inversa con HRA. Es conocido que una matriz de Nekrasov  $A$  es una H-matriz (por el corolario 2 de [27]) y, por tanto, por el Teorema 2.14, existe una matriz  $D$  diagonal tal que  $AD$  es estrictamente diagonalmente dominante (SDD). Con objeto de tener una matriz diagonal  $S$  sencilla, nosotros nos conformaremos con que  $AS$  sea diagonalmente dominante. Esta matriz  $S$  es la siguiente:

$$S = \begin{pmatrix} \frac{h_1(A)}{|a_{11}|} & & & \\ & \frac{h_2(A)}{|a_{22}|} & & \\ & & \ddots & \\ & & & \frac{h_n(A)}{|a_{nn}|} \end{pmatrix}. \quad (4.3)$$

**Lema 4.2.** Sea  $A = (a_{ij})_{1 \leq i, j \leq n}$  una matriz de Nekrasov y  $S$  la matriz dada por (4.3). Entonces, la matriz  $AS$  es una matriz diagonalmente dominante por filas.

*Demostración.* Llamemos  $B := AS$  a nuestra matriz en estudio. Tenemos que los elementos de  $B = (b_{ij})_{1 \leq i, j \leq n}$  son:

$$b_{ij} = \begin{cases} a_{ij} \frac{h_j(A)}{|a_{jj}|}, & \text{si } i \neq j, \\ h_i(A), & \text{si } i = j. \end{cases}$$

Para ver que  $B$  es diagonalmente dominante, veamos que la propiedad de dominancia diagonal se cumple para la fila  $i$ -ésima, con  $i = 1, \dots, n$ :

$$h_i(A) = \sum_{j=1}^{i-1} |a_{ij}| \frac{h_j(A)}{|a_{jj}|} + \sum_{j=i+1}^n |a_{ij}| \geq \sum_{j=1}^{i-1} |a_{ij}| \frac{h_j(A)}{|a_{jj}|} + \sum_{j=i+1}^n |a_{ij}| \frac{h_j(A)}{|a_{jj}|},$$

puesto que  $h_j(A) < |a_{jj}|$  por ser  $A$  matriz de Nekrasov, y así queda demostrada la dominancia diagonal.  $\square$

La matriz  $S$  nos permite transformar una matriz de Nekrasov en una matriz diagonalmente dominante. Pero además, si la matriz tiene la estructura de signos de una  $Z$ -matriz con diagonal positiva, el escalado por  $S$  mantendrá dicha estructura, lo cual será clave para la resolución de nuestro problema. Esta afirmación se corresponde con el siguiente Corolario, consecuencia del Lema 4.2.

**Corolario 4.3.** *Sea  $A = (a_{ij})_{1 \leq i, j \leq n}$  una  $Z$ -matriz de Nekrasov con entradas diagonales positivas y  $S$  la matriz dada por (4.3). Entonces, la matriz  $AS$  es una  $Z$ -matriz diagonalmente dominante por filas con entradas diagonales no negativas.*

*Demostración.* Aplicando el Lema 4.2 a la matriz  $A$  obtenemos que  $B := AS$  es una matriz diagonalmente dominante, por lo que solamente queda estudiar los signos de dicha matriz. Notemos que  $\frac{h_i(A)}{a_{ii}} \geq 0$  para  $i = 1, \dots, n$ , y, por tanto,  $S \geq 0$ . Entonces, al hacer el producto  $B = AS$  se conserva la estructura de signos de  $A$ , y tenemos que los elementos de  $B = (b_{ij})_{1 \leq i, j \leq n}$  son de nuevo:

$$b_{ij} = \begin{cases} a_{ij} \frac{h_j(A)}{a_{jj}}, & \text{si } i \neq j, \\ h_i(A), & \text{si } i = j. \end{cases}$$

Como  $A$  es  $Z$ -matriz, los elementos extradiagonales de  $B$  son no positivos. Por definición,  $h_i(A) \geq 0$ , por lo que los elementos diagonales de  $B$  son no negativos y  $B$  es una  $Z$ -matriz de diagonal no negativa.  $\square$

Gracias a este corolario, vamos a poder apoyarnos en los resultados conocidos para  $M$ -matrices diagonalmente dominantes a la hora de afrontar la resolución de nuestro problema del cálculo de inversas utilizando la  $Z$ -matriz d.d.  $AS$  (y por el Teorema 2.11  $M$ -matriz d.d.). Como hemos mencionado previamente, la clave para aplicar algoritmos con HRA a estas matrices se encontraba en utilizar una parametrización adecuada de las mismas, que en este caso se correspondía con los elementos extradiagonales y la suma de los elementos de cada una de sus filas. Por tanto, buscaremos hallar estos parámetros de  $AS$  de una forma que nos asegure su obtención con alta precisión relativa, y así estaremos ya en condiciones de resolver nuestro problema. Además, no nos hará falta usar estrategias de pivotaje.

### 4.3. HRA para matrices de Nekrasov

Dada una  $Z$ -matriz de Nekrasov con diagonal positiva,  $A$ , nuestro objetivo es obtener la parametrización adecuada (vista en el capítulo 3) con HRA de la matriz escalada  $B := AS$ , donde  $S$  es la matriz dada por (4.3), a partir de los parámetros (4.2). En el siguiente resultado se muestra que es posible lograr los parámetros necesarios de  $B$  con HRA dando una demostración constructiva que permite obtener los mismos con un coste computacional del orden de  $n^2$  operaciones elementales.

**Teorema 4.4.** Sea  $A = (a_{ij})_{1 \leq i, j \leq n}$  una Z-matriz de Nekrasov con entradas diagonales positivas y  $S$  la matriz dada por (4.3). Entonces, podemos hallar las sumas de las entradas de cada fila y los elementos extradiagonales de  $AS$  a partir de los  $n^2$  parámetros dados por (4.2) mediante un algoritmo libre de restas (SF) que realiza  $\frac{3n(n-1)}{2}$  sumas,  $2n(n-1)$  productos y  $2n-1$  cocientes.

*Demostración.* Observemos que por (4.2),

$$a_{jj} = \Delta_j(A) + h_j(A), \quad j = 1, \dots, n. \quad (4.4)$$

Así, tras calcular con SF  $h_1(A)$  con la fórmula de (4.1) procedemos a calcular  $a_{11}$  con SF mediante (4.4) para  $j=1$ . A continuación, seguimos calculando  $h_2(A)$ ,  $a_{22}$ ,  $h_3(A)$ ,  $a_{33}$ ,  $\dots$ ,  $h_n(A)$ ,  $a_{nn}$  con SF mediante (4.1) y (4.2). Como el elemento extradiagonal  $(i, j)$ ,  $i \neq j$ , de  $AS$  es  $a_{ij} \frac{h_j(A)}{a_{jj}}$ , podemos calcularlo con SF. Finalmente, para cada  $i = 1, \dots, n$  la suma de los elementos de la fila  $i$ -ésima de  $AS$  es

$$\sum_{j=1}^{i-1} a_{ij} \frac{h_j(A)}{a_{jj}} + h_i(A) + \sum_{j=i+1}^n a_{ij} \frac{h_j(A)}{a_{jj}},$$

que al sustituir  $h_i(A)$  por su valor en (4.1) y tener en cuenta que  $A$  es Z-matriz toma el valor

$$\sum_{j=i+1}^n (-a_{ij}) \left( 1 - \frac{h_j(A)}{a_{jj}} \right) = \sum_{j=i+1}^n |a_{ij}| \frac{a_{jj} - h_j(A)}{a_{jj}} = \sum_{j=i+1}^n |a_{ij}| \frac{\Delta_j(A)}{a_{jj}}, \quad (4.5)$$

que de nuevo se puede calcular con SF.

Veamos cuántas operaciones elementales son necesarias para calcular los parámetros. Como todos los cálculos descritos son SF, se realizarán cero restas. El cálculo de los elementos diagonales  $a_{ii}$  con  $i = 1, \dots, n$  supone la realización de  $n$  sumas. Para cada  $h_i(A)$ , con  $i = 1, \dots, n$  necesitaremos realizar  $n-2$  sumas además de un número de productos y cocientes que depende del índice  $i$ . Notemos, eso sí, que el cálculo  $\frac{h_j(A)}{a_{jj}}$  se empleará tanto para calcular los  $h_i(A)$  con  $j < i \leq n$  así como para obtener los elementos extradiagonales de la columna  $j$ -ésima de  $AS$ , por lo que los calcularemos una vez y los emplearemos cuando sea necesario. Esto supone realizar  $n$  cocientes,  $\sum_{i=1}^{n-1} i = \frac{(n-1)n}{2}$  productos y  $n(n-2)$  sumas. Para obtener los elementos extradiagonales  $(i, j)$  con  $i \neq j$  de  $AS$ ,  $a_{ij} \frac{h_j(A)}{a_{jj}}$ , bastará con realizar un producto. Por tanto, se añade el realizar  $n(n-1)$  productos. Ahora solo queda calcular la suma de los elementos de cada fila de  $AS$ . Primero, calcularemos los cocientes  $\frac{\Delta_j(A)}{a_{jj}}$  para  $j = 2, \dots, n$ , lo que añade  $n-1$  cocientes al coste computacional. Finalmente, realizaremos  $\frac{n(n-1)}{2}$  sumas y  $\frac{n(n-1)}{2}$  productos para obtener el valor de los últimos  $n$  parámetros. En total, necesitamos  $\frac{3n(n-1)}{2}$  sumas,  $2n(n-1)$  productos y  $2n-1$  cocientes.  $\square$

En el siguiente capítulo veremos cómo utilizar el Teorema 4.4 para calcular la inversa de  $A$  con HRA y también cómo resolver con HRA sistemas de ecuaciones lineales de la forma  $Ax = b$ , siendo  $b$  un vector de componentes no negativas.

# Capítulo 5

## Inversas precisas de Z-matrices de Nekrasov

El objetivo de este capítulo es dar una metodología para resolver el problema de hallar la inversa con HRA de una clase concreta de matriz estructurada: una Z-matriz de Nekrasov con elementos diagonales positivos. Para lograrlo, vamos a seguir una estrategia que se apoya en las descritas para trabajar con M-matrices diagonalmente dominantes. Utilizando la parametrización introducida en el capítulo anterior logramos relacionarlas con las M-matrices diagonalmente dominantes y podemos aprovecharnos de las técnicas conocidas en este caso para lograr nuestro objetivo. Además, podremos resolver también con HRA el sistema lineal de ecuaciones  $Ax = b$ , con la condición de que ninguna componente del vector  $b$  sea negativa ( $b \geq 0$ ). Aseguraremos que trabajamos con HRA viendo que los algoritmos descritos satisfacen la condición NIC.

Comenzamos el capítulo demostrando varios resultados que serán fundamentales para el cálculo de la matriz inversa, logrando con ellos obtener la misma en un caso particular. El siguiente paso consiste en generalizar este resultado gracias al estudio de la estructura de ceros asociada a una matriz de Nekrasov. Por último, se incluye en pseudocódigo la implementación de las rutinas desarrolladas durante el capítulo.

### 5.1. Inversas precisas: caso particular

Comencemos viendo que dada una Z-matriz d.d. de la que conocemos sus sumas de filas y elementos extradiagonales, podemos obtener su inversa mediante Gauss-Jordan (véase la sección 3.1) sin usar restas y sin uso de estrategias de pivotaje, a diferencia de como se hacía en [11, 25] (véase sección 3.2).

**Proposición 5.1.** *Sea  $A = (a_{ij})_{1 \leq i, j \leq n}$  una Z-matriz d.d. no singular con entradas diagonales positivas. Si conocemos con HRA las sumas de los elementos de cada fila de  $A$  así como sus elementos extradiagonales, podemos calcular  $A^{-1}$  y la solución del sistema de ecuaciones lineales  $Ax = b$  con  $b \geq 0$  con un algoritmo libre de restas (SF, y así con HRA) de  $\mathcal{O}(n^3)$  operaciones elementales.*

*Demostración.* Como  $A$  es una Z-matriz d.d. con diagonal positiva cumple que  $A + D$  es SDD para cualquier matriz diagonal  $D > 0$ , por lo que  $A + D$  es no singular. Por la equivalencia de viii) con i) en el Teorema 2.9, deducimos que  $A$  es por tanto una M-matriz no singular. Veamos

que podemos calcular  $A^{-1}$  con HRA. Para ello usaremos el método de Gauss-Jordan sin pivotaje (véase sección 3.1). Construimos la matriz ampliada  $\tilde{M} := (A|I|s)$ , donde  $I$  es la matriz identidad y  $s$  es el vector de las sumas de las filas de  $A$ , es decir,  $s_i$  es la suma de los elementos de la fila  $i$ -ésima de  $A$ . Aplicaremos la eliminación Gaussiana de  $A$  realizando las operaciones por filas en toda la matriz ampliada  $\tilde{M}$ . El primer pivote es  $a_{11}$ , que se calcula sumando a  $s_1$  el valor absoluto de los elementos extradiagonales de la primera fila. Comenzamos haciendo ceros en la primera columna debajo de éste empleando múltiplos de la primera fila, y, excepto los elementos diagonales de  $A^{(2)}[2, \dots, n]$ , todo elemento de  $\tilde{M}$  se calcula con HRA. No obstante, estos elementos los calcularemos con HRA solo cuando necesitemos emplearlos como pivote (y el último,  $a_{nn}^{(n)}$ , cuando hayamos terminado de hacer ceros por debajo de la diagonal). Así, para la siguiente iteración, solo queda calcular el elemento  $a_{22}^{(2)}$ . Para lograrlo con HRA basta sumar  $s_2^{(2)}$  y los valores absolutos de los elementos extradiagonales de la segunda fila de  $A^{(2)}$ . Notemos que por la estructura de signos se corresponderá con sumar los opuestos de los elementos.

Para realizar el segundo paso, tenemos que  $A^{(2)}[2, \dots, n]$  vuelve a ser M-matriz por ser el complemento de Schur de una M-matriz (véase [12]). Por tanto, utilizando la misma estrategia que en el paso 1 sobre  $A^{(2)}[2, \dots, n]$  haremos ceros en la segunda columna. Repetimos hasta llegar a  $U = A^{(n)}$  con HRA, que será triangular superior, y la estructura de signos de  $\tilde{M}$  será la siguiente:

$$\tilde{M}^{(n)} = \left( \begin{array}{cccc|ccc|c} + & - & - & \dots & - & 1 & & & \\ & + & - & \dots & - & + & 1 & & \\ & & + & \ddots & \vdots & + & + & 1 & \\ & & & \ddots & - & \vdots & \vdots & \ddots & \ddots \\ & & & & + & + & + & \dots & + & 1 \end{array} \right)$$

$\underbrace{\hspace{15em}}_{U = A^{(n)}} \quad \underbrace{\hspace{10em}}_C \quad \underbrace{\hspace{2em}}_{s^{(n)}}$

En esta matriz, “+” quiere decir que el elemento correspondiente de  $\tilde{M}$  es  $\geq 0$ , y “-”, que es  $\leq 0$ . Los elementos diagonales de  $U$  son positivos ( $> 0$ ). A partir de ahora el vector  $s^{(n)}$  ya no es necesario, así que lo omitiremos al representar  $\tilde{M}^{(n)}$ .

Para llegar desde aquí hasta  $A^{-1}$  basta con repetir el proceso empleando como fila pivote la fila inferior para hacer ceros por encima de la diagonal de  $U$ :  $(U|C) \rightarrow (D|DA^{-1})$ , con  $D$  matriz diagonal.

En este procedimiento, en el paso  $k$  se emplea como pivote  $u_{n-k, n-k}^{(k)}$ , que es siempre mayor que 0 (puesto que en los pasos de esta eliminación no se ven afectados). Al ser los extradiagonales no positivos y los pivotes positivos, no se han realizado restas al calcular  $DA^{-1}$ . Solamente queda realizar el producto  $D^{-1}DA^{-1}$  para obtener  $A^{-1}$ . Por tanto, hemos llevado a cabo todo el proceso sin llevar a cabo restas (condición SF).  $\square$

El siguiente resultado es consecuencia del Lema 4.3 y la Proposición 5.1.

**Corolario 5.2.** *Sea  $A = (a_{ij})_{1 \leq i, j \leq n}$  una Z-matriz de Nekrasov con entradas diagonales positivas que cumpla  $h_i(A) \neq 0$  para  $i = 1, \dots, n$  y sea  $S$  la matriz diagonal dada por (4.3). Si conocemos con HRA las sumas de los elementos de cada fila y los elementos extradiagonales de  $AS$  con HRA entonces podemos calcular  $A^{-1}$  y la solución del sistema de ecuaciones lineales  $Ax = b$  con  $b \geq 0$  con un algoritmo libre de restas (y así con HRA) de  $\mathcal{O}(n^3)$  operaciones elementales.*

*Demostración.* Comencemos recordando que  $A$  es una matriz no singular por ser matriz de Nekrasov. Además, por el Lema 4.3 la matriz  $B := AS$  será una  $Z$ -matriz d.d., y, por la hipótesis de no anulación de los parámetros  $h_i(A)$ ,  $S$  será no singular, y, por tanto,  $B$  también será no singular. Así, estamos en condiciones de aplicar la Proposición 5.1 a  $B$  y calcular la matriz  $B^{-1}$  con HRA. Con esta matriz, basta realizar el cálculo  $A^{-1} = SB^{-1}$  para obtener la inversa de  $A$ . La resolución del sistema de ecuaciones lineales se obtiene de forma análoga realizando el producto  $x = SB^{-1}b = A^{-1}b$ .  $\square$

El siguiente resultado consecuencia del Teorema 4.4 y el Corolario 5.2 nos asegura, a partir de los parámetros (4.2), el cálculo con HRA de la inversa y de la resolución de ciertos sistemas lineales cuando tenemos una  $Z$ -matriz de Nekrasov con diagonal positiva y una condición adicional. Posteriormente veremos que podemos prescindir de esta condición añadida.

**Corolario 5.3.** *Sea  $A = (a_{ij})_{1 \leq i, j \leq n}$  una  $Z$ -matriz de Nekrasov con entradas diagonales positivas que cumpla  $h_i(A) \neq 0$  para  $i = 1, \dots, n$ . Si conocemos (4.2) con HRA entonces podemos calcular  $A^{-1}$  y la solución del sistema de ecuaciones lineales  $Ax = b$  con  $b \geq 0$  con un algoritmo libre de restas (y así con HRA) de  $\mathcal{O}(n^3)$  operaciones elementales.*

*Demostración.* Sea  $S$  la matriz diagonal dada por (4.3). Por el Teorema 4.4 podemos calcular con HRA los elementos extradiagonales de  $B := AS$  así como la suma de los elementos de cada una de sus filas. Así, estamos en condiciones de aplicar el Corolario 5.2 y obtener el resultado buscado empleando un algoritmo libre de restas (SF).  $\square$

## 5.2. Inversas precisas: caso general

Hemos conseguido resolver con HRA el problema descrito al comienzo de la sección anterior con la condición adicional de que  $h_i(A) \neq 0$  para  $i = 1, \dots, n$ . El problema que se presentaba si no imponíamos esta condición es que el producto  $AS$ , con  $S$  la matriz dada en 4.3, da lugar a una matriz singular si  $h_i(A) = 0$  para algún  $i = 1, \dots, n$ . No obstante, podemos suprimir esta imposición y el resultado seguirá siendo cierto. Para demostrarlo, conviene comenzar estudiando la estructura de ceros intrínseca a una matriz de Nekrasov que satisface  $h_i(A) = 0$  para algún  $i = 1, \dots, n$ .

**Lema 5.4.** *Sea  $A = (a_{ij})_{1 \leq i, j \leq n}$  una matriz de Nekrasov, y sea  $J = \{i_1, \dots, i_k\} \subseteq \{1, \dots, n\}$  el conjunto ordenado de menor a mayor de todos los índices tales que  $h_{i_j}(A) = 0$ . Entonces para cada  $j = 1, \dots, k$  la fila de índice  $i_j$  tiene al menos  $n - j$  ceros entre sus elementos extradiagonales.*

*Demostración.* Comenzamos estudiando la fila de índice  $i_1$ :

$$h_{i_1}(A) = \sum_{k=1}^{i_1-1} |a_{i_1 k}| \frac{h_k(A)}{|a_{kk}|} + \sum_{k=i_1+1}^n |a_{i_1 k}| = 0. \quad (5.1)$$

Como  $h_k(A) \neq 0$  para  $k < i_1$  debido a que  $k \notin J$ , tiene que ocurrir por (5.1) que  $a_{i_1 k} = 0$  siempre que  $k \neq i_1$ . Es decir, todos los elementos extradiagonales de la fila son nulos. En el caso de la fila de índice  $i_j$ , con  $i_j \in J$ :

$$\sum_{k=1}^{i_j-1} |a_{i_j k}| \frac{h_k(A)}{|a_{kk}|} + \sum_{k=i_j+1}^n |a_{i_j k}| = \sum_{k=1, k \notin J}^{i_j-1} |a_{i_j k}| \frac{h_k(A)}{|a_{kk}|} + \sum_{k=i_j+1}^n |a_{i_j k}| = 0.$$

En este caso, se tiene que  $a_{i,jk} = 0$  siempre que  $k \notin \{i_1, \dots, i_j\}$ . De aquí se deduce que en la fila de índice  $i_j$  hay al menos  $n - j$  ceros situados en las columnas de índice  $k$ , con  $k \notin \{i_1, \dots, i_j\}$ .  $\square$

Observemos que en el Lema 5.4 también se demuestra que la fila de índice  $i_1$  tiene exactamente  $n - 1$  ceros. Este lema juega un papel fundamental en el resultado principal que buscamos demostrar. En su demostración denotaremos por  $I^c$  al conjunto complementario en  $N := \{1, \dots, n\}$  a un conjunto de índices  $I$  contenido en  $N$ .

**Teorema 5.5.** *Sea  $A = (a_{ij})_{1 \leq i, j \leq n}$  una Z-matriz de Nekrasov con entradas diagonales positivas. Si conocemos (4.2) con HRA entonces podemos calcular con HRA y  $\mathcal{O}(n^3)$  operaciones elementales  $A^{-1}$  y la solución del sistema de ecuaciones lineales  $Ax = b$  con  $b \geq 0$  mediante un algoritmo libre de restas (SF).*

*Demostración.* Comenzamos calculando  $h_1(A), a_{11}, \dots, h_n(A), a_{nn}$ . Definimos  $I \subseteq N$  como el conjunto de índices para los que se verifica  $h_i(A) \neq 0$ . Si  $I = N$  basta con aplicar el Corolario 5.3. En caso contrario comenzamos definiendo  $S$  como la matriz diagonal dada por (4.3). Formamos la submatriz  $\hat{A} := A[I]$  y  $B := (AS)[I]$ . Esta segunda matriz es diagonalmente dominante (d.d.) debido a que  $AS$  ya lo era y hemos construido la submatriz utilizando los mismos índices de filas y columnas. Así,  $B$  es una Z-matriz d.d. con entradas diagonales no negativas, y, por el Teorema 2.11, es una M-matriz. Por la Proposición 5.1 podemos calcular su inversa mediante un algoritmo libre de restas si conocemos una parametrización adecuada con HRA. En este caso los parámetros que necesitamos obtener son los elementos extradiagonales y la suma de los elementos de cada fila. Los primeros son de la forma  $a_{ij} \frac{h_j(A)}{a_{jj}}$ . Las sumas de los elementos de cada fila presentan la forma:

$$\sum_{j \in I, j \neq i} a_{ij} \frac{h_j(A)}{a_{jj}} + h_i(A) = \sum_{j=1}^{i-1} a_{ij} \frac{h_j(A)}{a_{jj}} + h_i(A) + \sum_{j=i+1}^n a_{ij} \frac{h_j(A)}{a_{jj}} = \sum_{j=i+1}^n |a_{ij}| \frac{\Delta_j(A)}{a_{jj}}.$$

La primera igualdad es debida a que los sumandos que incluimos son 0, ya que  $h_j(A) = 0$  cuando  $j \in I^c$ . La segunda se obtiene al sustituir  $h_i(A)$  en (4.1) teniendo en cuenta (4.2) y que  $A$  es Z-matriz. En ambos casos, tenemos la expresión obtenida en el Teorema 4.4, por lo que podemos lograr con alta precisión relativa los parámetros necesarios para poder aplicar la Proposición 5.1 y hallar la inversa de  $B = (AS)[I]$ , M-matriz d.d., y a partir de la misma obtener  $\hat{A}^{-1} = S[I]B^{-1}$ . Recordemos que por la equivalencia  $i \Leftrightarrow v$  del Teorema 2.9  $B^{-1}$  es no negativa, y así,  $\hat{A}^{-1}$  también es no negativa.

A partir de la submatriz  $\hat{A}^{-1}$  podemos construir  $A^{-1}$ . Para ello vamos a realizar  $n - |I|$  pasos, que consistirán en añadir la fila y columna de mayor índice de  $I^c$  que aún no hayamos incluido en  $\hat{A}$  (en el lugar que corresponda) y obtener la inversa de esta nueva submatriz:

$$\hat{A} = \hat{A}^{(1)} \rightarrow \hat{A}^{(2)} \rightarrow \dots \rightarrow \hat{A}^{(n-|I|+1)} = A. \quad (5.2)$$

Como en cada paso añadiremos una fila y columna, y obtendremos la inversa de la matriz que hemos construido, al añadir la última fila y columna y continuar el proceso obtendremos la inversa de la matriz  $A$ . Para llevar a cabo el primer paso elegimos el mayor índice  $k \in I^c$  (es decir, el mayor  $k$  tal que  $h_k(A) = 0$ ). Entonces, formamos la matriz  $(n - |I| + 1) \times (n - |I| + 1)$   $\hat{A}^{(2)}$  añadiendo la fila y columna  $k$ -ésima de  $A$  a  $\hat{A}$  (en el lugar correspondiente). Para obtener la inversa de esta nueva submatriz a partir de  $C = \hat{A}^{-1}$  nos apoyamos en el Lema 5.4, por el cual la nueva fila añadida a  $\hat{A}$  tiene al menos  $|I|$  ceros, los cuales aparecen como elementos



extradiagonales en la fila correspondiente de  $\hat{A}^{(2)}$ . Por tanto, en esa fila el único elemento no nulo es el de la diagonal,  $a_{kk}$ . Los elementos de  $C^{(2)} = (\hat{A}^{(2)})^{-1}$  son los siguientes:

$$c_{ij}^{(2)} = \begin{cases} c_{ij}, & i, j \in I, \\ \frac{1}{a_{kk}}, & i = j = k, \\ 0, & i = k, j \in I, \\ c_{ik}^{(2)}, & i \in I, j = k. \end{cases}$$

Comprobemos que estos son los elementos de  $C^{(2)}$  y deduzcamos una expresión para los que faltan por determinar. Tiene que cumplirse  $\hat{A}^{(2)}C^{(2)} = I_{|I|+1}$ . Si  $i, j \in I$ , el elemento  $(i, j)$  del producto matricial es cero o uno cuando corresponde ya que la inversa de  $\hat{A}$  es  $C$ , y la operación que se realiza es

$$\sum_{s \in I} a_{is}c_{sj} + a_{ik} \cdot 0 = \begin{cases} 0, & i \neq j, \\ 1, & i = j. \end{cases}$$

Si  $i = k, j \in I$ , se tiene

$$\sum_{s \in I} a_{ks}c_{sj} + a_{kk}c_{kj} = \sum_{s \in I} 0 \cdot c_{sj} + a_{kk} \cdot 0 = 0.$$

Si  $i = j = k$ , obtenemos

$$\sum_{s \in I} a_{ks}c_{sk} + a_{kk}c_{kk} = \sum_{s \in I} 0 \cdot c_{sk} + \frac{a_{kk}}{a_{kk}} = 1.$$

Queda el caso  $i \in I, j = k$ , que determina los valores que nos faltan.

$$\sum_{s \in I} a_{is}c_{sk}^{(2)} + \frac{a_{ik}}{a_{kk}} = 0, \quad i \in I.$$

Considerando el vector  $c = \left( c_{ik}^{(2)} \right)_{i \in I}$  que queremos calcular podemos reformular el sistema anterior en forma matricial:

$$\hat{A}c = - (a_{ik})_{i \in I} \left( \frac{1}{a_{kk}} \right)$$

y, como conocemos la matriz no negativa  $\hat{A}^{-1}$  con alta precisión relativa, y el término de la derecha es también no negativo, obtenemos  $c$  con alta precisión relativa. Para ello basta con hacer el siguiente producto:

$$c = C (a_{ik})_{i \in I} \left( \frac{-1}{a_{kk}} \right) = \hat{A}^{-1} (a_{ik})_{i \in I} \left( \frac{-1}{a_{kk}} \right).$$

Una vez calculados estos elementos ya tenemos  $C^{(2)}$ . Si se tenía que  $\hat{A}^{(2)} = A$  ya habremos obtenido  $A^{-1}$ . En caso contrario, procedemos de forma análoga. Es decir, seleccionamos el mayor índice  $k \in I^c$  de fila y columna que aún no hayamos incluido en nuestra submatriz, y formamos  $\hat{A}^{(3)}$  añadiendo a  $\hat{A}^{(2)}$  la fila y columna de índice  $k$  en el lugar correspondiente. La fila añadida tiene al menos  $|I| + 1$  ceros en la matriz original, los cuales aparecen como elementos extradiagonales en la submatriz construida. Para usar una notación más compacta, definimos  $I^{(2)}$  como el conjunto de índices ordenado de filas y columnas de  $A$  que aparecían

en  $\hat{A}^{(2)}$ . Buscamos hallar  $C^{(3)} = (\hat{A}^{(3)})^{-1}$ . Razonando de forma análoga al anterior paso, los elementos de  $C^{(3)}$  serán:

$$c_{ij}^{(3)} = \begin{cases} c_{ij}^{(2)}, & i, j \in I^{(2)}, \\ \frac{1}{a_{kk}}, & i = j = k, \\ 0, & i = k, j \in I^{(2)}, \\ c_{ik}^{(3)}, & i \in I^{(2)}, j = k. \end{cases}$$

De nuevo, conocemos todos los elementos menos los últimos, que podemos calcular de forma análoga. Si  $c = (c_{ik}^{(3)})_{i \in I^{(2)}}$ , se tiene:

$$c = C^{(2)} (a_{ik})_{i \in I^{(2)}} \frac{-1}{a_{kk}} = (\hat{A}^{(2)})^{-1} (a_{ik})_{i \in I^{(2)}} \begin{pmatrix} -1 \\ a_{kk} \end{pmatrix}.$$

En general, tras realizar esta operación  $p - 1$  veces podemos obtener  $A^{-1}$  y terminar el procedimiento, o tener que reiterarlo añadiendo la fila y columna de mayor índice  $k \in I^c$  tal que no haya sido añadida en ningún paso anterior. Una vez más, la fila añadida tiene al menos  $|I| + p - 1$  ceros en la matriz original, los cuales aparecen como elementos extradiagonales en la fila correspondiente de la submatriz construida. También definimos  $I^{(p)}$  como el conjunto de índices ordenado de filas y columnas de  $A$  que aparecían en  $\hat{A}^{(p)}$ . Entonces realizamos el producto  $c = C^{(p)} (a_{ik})_{i \in I^{(p)}} \begin{pmatrix} -1 \\ a_{kk} \end{pmatrix}$  para obtener los elementos de la matriz inversa  $C^{(p+1)} = (\hat{A}^{(p+1)})^{-1}$  a calcular y construimos la misma. Sus elementos serán de la forma:

$$c_{ij}^{(p+1)} = \begin{cases} c_{ij}^{(p)}, & i, j \in I^{(p)}, \\ \frac{1}{a_{kk}}, & i = j = k, \\ 0, & i = k, j \in I^{(p)}, \\ c, & i \in I^{(p)}, j = k. \end{cases}$$

□

La demostración del teorema da una idea de cómo se puede llevar a cabo una implementación de un algoritmo para calcular la inversa de una Z-matriz de Nekrasov de elementos diagonales positivos. En cualquier caso, no es necesario ir construyendo la sucesión de matrices  $\hat{A}$ ; basta con ir realizando los productos matriciales de la forma  $c = C^{(p)} (a_{ik})_{i \in I^{(p)}} \begin{pmatrix} -1 \\ a_{kk} \end{pmatrix}$  y construir la matriz  $C^{(p+1)}$  a partir de la matriz  $C^{(p)}$  y el vector  $c$ .

### 5.3. Algoritmos

A continuación se recogen todos los algoritmos necesarios para resolver nuestro problema con HRA. El primer algoritmo que presentamos, el Algoritmo 3, lleva nuestra Z-matriz de Nekrasov inicial a la forma necesaria para calcular su inversa con HRA. En el caso de que  $h_i(A) \neq 0$  para  $i = 1, \dots, n$  el procedimiento que se realiza se corresponde con el descrito en el Teorema 4.4. Si alguno de estos parámetros se anula, se trabaja con la submatriz descrita en la demostración del Teorema 5.5. Las salidas del algoritmo son la matriz  $A$ , en la que se almacenan los parámetros de la matriz  $AS$  en la submatriz  $A(I, I)$ , el conjunto de índices  $I$  y la matriz  $S$  cuando el conjunto de índices está compuesto por más de un índice.

Una vez obtenida la parametrización de la M-matriz d.d.  $AS$ , nuestro objetivo es calcular su inversa con HRA. Para ello, emplearemos el Algoritmo 4 (este método se corresponde con

**Algoritmo 3** nektoddgen - Preparar Z-matriz de Nekrasov para obtener su inversa con HRA

---

**Input:**  $A = (a_{ij})(i \neq j), \Delta$   $\triangleright \Delta$  es el vector de los parámetros  $\Delta_j(A)$

**for**  $i = 1 : n$

$h_i = \sum_{j=1}^{i-1} a_{ij}k_j + \sum_{j=i+1}^n a_{ij}$

$a_{ii} = \Delta_i + h_i$

$k_i = h_i/a_{ii}$

**end for**

Construir  $I$ , lista de índices ordenados tales que  $h_i \neq 0$ .

**if**  $|I| > 1$

**for**  $i = I$

$a_{ii} = \sum_{j=i+1}^n a_{ij}\Delta_j/a_{jj}$

**for**  $j = I \setminus \{i\}$

$a_{ij} = a_{ij}k_j$

**end for**

**end for**

Construir  $S$ , matriz diagonal  $I \times I$  con los elementos  $k_j, j \in I$ .

**else if**  $|I| == 1$

$a_{II} = 1/a_{II}$

**else**

$a_{nn} = 1/a_{nn}$

$I = [n]$

**end if**

---

el descrito en la demostración de la Proposición 5.1). Así obtenemos la inversa de la submatriz de  $AS$  correspondiente a los índices de filas tales que  $h_i \neq 0$ , es decir, las filas de índices  $I$ .

A continuación buscamos construir la matriz inversa de  $AS$  a partir de esta submatriz. Como entradas necesitaremos la matriz  $A$  en la que se ha sustituido  $A(I, I)$  por  $A^{-1}(I, I)$  e  $I$  (basta con multiplicar antes a izquierda por  $S(I)$  al output de `inversadd`). Este procedimiento es el que se muestra en el Algoritmo 5.

Con las rutinas que hemos descrito, podemos dar una expresión compacta del método que hemos desarrollado para calcular la matriz inversa con HRA de una Z-matriz de Nekrasov con elementos diagonales positivos partiendo de la parametrización propuesta:

**Algoritmo 4** inversadd - Calcular inversa de una M-matriz d.d.

---

**Input:**  $A = (a_{ij})(i \neq j), s$   $\triangleright s$  es el vector de sumas de elementos de cada fila de  $A$   
 $P = I_n$

**for**  $k = 1 : n - 1$

$a_{kk} = \sum_{j=k+1}^n a_{kj}$

**for**  $i = k + 1 : n$

$piv = a_{ik}/a_{kk}$

$s_i = s_i - pivs_k$

$a_{ik} = 0$

$A(i, :) = A(i, :) - pivA(k, :)$

$P(i, :) = P(i, :) - pivP(k, :)$

**end for**

**end for**

$a_{nn} = s_n$

**for**  $k = n : -1 : 2$

**for**  $i = k - 1 : -1 : 1$

$piv = a_{ik}/a_{kk}$

$P(i, :) = P(i, :) - pivP(k, :)$

**end for**

**end for**

**for**  $i = 1 : n$

$P(i, :) = P(i, :) * 1/a_{ii}$

**end for**

---

**Algoritmo 5** buildnekinv - construir inversa

---

**Input:**  $A, I$   $\triangleright$  en  $A(I, I)$  está almacenada la inversa de la submatriz original  
 Construir el conjunto de índices  $J$ , compuesto por los índices de  $I^c$  ordenados de mayor a menor.

**for**  $i = J$

$a_{ii} = 1/a_{ii}$

$A(I, i) = -A(I, I)(A(I, i) ./ a_{ii})$   $\triangleright ./$  denota que la operación es elemento a elemento

$I = I \cup \{i\}$  ordenado

**end for**

---

**Algoritmo 6** Calcular inversa de una Z-matriz de Nekrasov con elementos diagonales positivos

---

**Input:**  $A = (a_{ij})(i \neq j), \Delta$   $\triangleright \Delta$  es el vector de los parámetros  $\Delta_j(A)$   
 $[A, I, S] = nektoddgen(A = (a_{ij})(i \neq j), \Delta)$

**if**  $|I| > 1$

$A(I, I) = S * inversadd(A(I, I))$

**end if**

$A^{-1} = buildnekinv(A)$

---

## Capítulo 6

# Experimentación numérica para inversas y sistemas

En este capítulo vamos a comparar los resultados de una implementación en MATLAB de los algoritmos descritos en el capítulo anterior con el de las rutinas ya implementadas en dicho software. En particular, se van a estudiar dos problemas.

El primero es el cálculo de la inversa de una Z-matriz de Nekrasov, problema al que se dedica la primera sección de este capítulo. La segunda sección tratará de la resolución de sistemas lineales de ecuaciones cuya matriz de coeficientes pertenezca a esta clase. Aunque la HRA solo se garantiza cuando  $b \geq 0$ , veremos en los experimentos numéricos que nuestro método siempre da soluciones precisas para cualesquiera términos independientes. En ambos apartados, se comparan los resultados con los obtenidos empleando las rutinas implementadas en MATLAB para resolver dichos problemas. Los errores mostrados se han obtenido considerando como solución exacta la obtenida empleando cálculo simbólico de MATLAB.

### 6.1. Cálculo de inversas

El primer problema que hemos abordado es el cómputo de la matriz inversa. Vamos a comparar los resultados obtenidos al calcular una matriz inversa siguiendo el procedimiento con HRA que hemos descrito y utilizando la orden `inv(A)` implementada en MATLAB en diversos ejemplos de Z-matrices de Nekrasov con elementos diagonales positivos.

A continuación vamos a fijarnos en el comportamiento del error en casos concretos. Comenzamos estudiando 10 casos de matrices con entradas generadas aleatoriamente, en los que la primera fila es muy dominante. Los resultados obtenidos se incluyen en la Tabla 6.1. En este caso, el algoritmo que se utiliza es en la mayoría de los casos el descrito en la sección 5.1. Aunque se observa una mejoría en los resultados obtenidos, los ejemplos generados no son muy mal condicionados. Una forma de obtener matrices peor condicionadas dentro de esta clase será imponer que alguna de las primeras filas cumpla  $h_i = 0$ , para que los elementos de la columna  $i$ -ésima sean arbitrariamente grandes, y la matriz esté lejos de ser diagonalmente dominante. Además, así pondremos a prueba el método desarrollado en el Teorema 5.5. Cabe destacar que en el caso de `inv` de MATLAB, se presenta el máximo error relativo de las componentes no nulas (puesto que el error relativo no está definido para una entrada nula). En el algoritmo propuesto las entradas nulas de la matriz inversa se fijan a 0, por lo que no se introduce ningún error en las mismas. Los resultados se muestran en la Tabla 6.2.

nº condición	inv	HRA
3.6714e+04	1.5308e-12	3.6475e-15
5.4599e+04	8.3304e-13	6.6024e-15
5.8606e+04	3.2871e-13	6.6759e-15
3.8833e+04	7.651e-13	3.5972e-15
4.2645e+04	9.3474e-13	5.5312e-15
8.8803e+04	1.1321e-14	1.5372e-15
8.2928e+04	1.533e-12	6.9976e-15
4.9831e+04	3.9542e-13	6.3254e-15
9.9321e+04	7.6187e-13	1.4831e-15
6.4873e+04	4.8123e-14	2.0905e-15

Tabla 6.1: Máximo error relativo cometido al calcular la inversa.

nº condición	inv	HRA
8.6088e+09	1.7700e-06	1.1907e-15
1.0732e+10	1.1087e+07	8.2809e-16
2.4392e+10	4.0486e-07	5.6175e-16
4.1714e+10	1.8978e+00	8.2015e-16
3.0772e+10	6.2672e-07	5.6761e-16
2.1360e+10	7.9297e-07	1.2172e-15
1.8158e+10	1.3845e+13	6.9970e-16
6.2268e+10	2.9948e-01	1.0738e-15
1.2966e+11	1.8669e-06	1.0067e-15
2.2980e+07	6.2546e-08	7.2895e-16

Tabla 6.2: Máximo error relativo cometido al calcular la inversa con  $h_i = 0$  para algún  $i$ .

## 6.2. Resolución de sistemas lineales

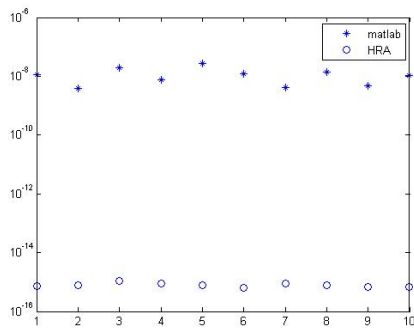
Con el cálculo de la inversa, podemos también resolver sistemas de ecuaciones lineales de la forma  $Ax = b$ . A continuación, vamos a comparar el resultado que obtendríamos calculando  $A^{-1}$  con HRA y realizando el producto  $A^{-1}b$  con la rutina implementada en MATLAB para la resolución de sistemas de ecuaciones lineales. Vamos a estudiar tanto el efecto de variar la matriz de coeficientes, como el de cambiar simplemente el término independiente manteniendo la matriz. Comenzamos generando una matriz  $30 \times 30$  que cumple la condición  $h_i = 0$  para algunos valores de  $i \in N$ . Su número de condición es de  $2,1334e + 09$ . Con esta matriz, resolvemos 10 sistemas de ecuaciones lineales en los que  $b$  tienen todas las componentes no negativas, y 10 sistemas en los que  $b$  presenta tanto componentes positivas como negativas.

A \ b	HRA
1.1721e-08	7.4861e-16
3.8531e-09	8.0188e-16
1.9166e-08	1.0915e-15
7.6172e-09	9.0263e-16
2.7821e-08	8.0516e-16
1.2423e-08	6.2222e-16
4.1265e-09	8.7997e-16
1.4059e-08	7.5829e-16
4.5942e-09	6.8648e-16
1.0882e-08	6.6262e-16

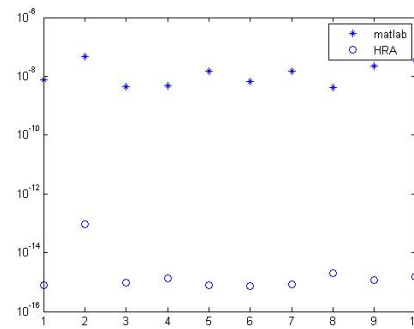
Tabla 6.3: Máximo error relativo cuando  $b \geq 0$

A \ b	HRA
7.6517e-09	7.6183e-16
4.7859e-08	9.2650e-14
4.4467e-09	9.7674e-16
4.8492e-09	1.3250e-15
1.4603e-08	7.7579e-16
6.5952e-09	7.3791e-16
1.5348e-08	8.2503e-16
4.0365e-09	2.0006e-15
2.2514e-08	1.1452e-15
3.4846e-08	1.5119e-15

Tabla 6.4: Máximo error relativo con  $b$  cualquiera



$b \geq 0$



$b$  cualquiera

Aunque el método de HRA es superior en ambos casos, a simple vista no se observa un cambio al imponer la condición  $b \geq 0$ . No obstante, si representamos gráficamente estos resultados, podemos observar que cuando no se exige la condición  $b \geq 0$  hemos obtenido un error mayor en un caso al utilizar el método con HRA para calcular  $A^{-1}b$ . El caso  $b \geq 0$  se corresponde con la Tabla 6.3, y el caso en el que  $b$  puede presentar componentes de ambos signos se recoge en la Tabla 6.4. También se incluye una gráfica en la que se compara el error obtenido en ambos casos.

Ahora vamos a cambiar la forma de proceder. Vamos a fijar un término independiente y cambiaremos la matriz de coeficientes. Primero, comenzamos con un vector positivo:

$b = (1, 1, \dots, 1, 1)^T$ . Para dicho vector generamos 10 matrices  $20 \times 20$  y resolvemos el sistema. Los resultados aparecen reflejados en la Tabla 6.5. También, procedemos de forma análoga, pero esta vez con el vector  $b = (b_i)_{1 \leq i \leq n}$  tal que  $b_i = (-1)^{i+1}$ . Los resultados se corresponden con la Tabla 6.6.

nº condición	A \ b	HRA
2.3319e+10	7.8736e-07	5.1521e-16
7.5491e+10	1.9843e-06	6.7465e-16
2.7086e+09	1.427e-14	6.2988e-16
1.0107e+10	9.7238e-15	6.0401e-16
5.8361e+10	4.7639e-07	4.7223e-16
9.6114e+09	5.5802e-14	5.318e-16
1.4918e+10	2.2768e-08	3.9299e-16
2.2116e+10	1.6766e-06	9.8126e-16
1.5419e+10	3.4e-14	9.2892e-16
1.4407e+11	1.6289e-06	4.5371e-16

Tabla 6.5: Máximo error relativo con  $b = (1, \dots, 1)$

nº condición	A \ b	HRA
1.1381e+10	1.0735e-06	1.9298e-15
2.5795e+10	2.0074e-14	7.6335e-16
2.4822e+11	3.5413e-12	1.8627e-14
4.1003e+10	1.8994e-06	3.837e-16
3.9208e+10	1.4595e-14	1.4025e-15
1.3225e+10	1.4073e-14	1.8282e-15
1.5134e+18	17.8615	6.1444e-16
4.9096e+11	1.3033e-06	8.2058e-16
9.5209e+09	6.1324e-14	1.5654e-14
3.6935e+10	1.9329e-13	3.6418e-15

Tabla 6.6: Máximo error relativo con  $b_i = (-1)^{i+1}$

Para concluir la sección de experimentación numérica, vamos a estudiar el comportamiento de todas las componentes calculadas para un caso particular. El vector de término independiente será  $b = (1, 1, \dots, 1, 1)^T$ , y la matriz  $A$  es la que hemos empleado anteriormente. La Tabla 6.7 muestra el resultado obtenido.



i	A \ b	HRA
1	5.3216e-09	3.9245e-17
2	4.2614e-09	1.3142e-17
3	2.6467e-14	1.3924e-16
4	3.2771e-09	4.6722e-16
5	2.6950e-09	4.1587e-16
6	3.3957e-09	1.0395e-15
7	3.4720e-09	2.5815e-17
8	3.5137e-09	2.3178e-16
9	3.2675e-09	8.6618e-16
10	2.3871e-13	2.3388e-16
11	3.4203e-09	3.2448e-16
12	3.2456e-09	2.2012e-16
13	3.4487e-09	1.4569e-16
14	3.4492e-09	1.0272e-15
15	3.3465e-09	2.1040e-16
16	3.2992e-09	2.0365e-16
17	4.7102e-13	3.5679e-17
18	3.4396e-09	2.1671e-16
19	3.2488e-09	4.4854e-17
20	6.8877e-13	1.1473e-16
21	3.2574e-09	3.5262e-16
22	4.9279e-17	8.8495e-17
23	3.3460e-09	3.1197e-16
24	3.3794e-09	4.4262e-16
25	3.3693e-09	6.5713e-16
26	4.2728e-18	4.2728e-18
27	7.6104e-18	7.6104e-18
28	3.2768e-09	8.7506e-16
29	1.0341e-16	1.0341e-16
30	2.4571e-17	2.4571e-17

Tabla 6.7: error relativo componente a componente con  $b = e$



# Capítulo 7

## Escalado para dominancia diagonal estricta

En un capítulo anterior hemos desarrollado un método para calcular la inversa de una  $Z$ -matriz de Nekrasov con elementos diagonales positivos empleando un escalado ( $AS$ , donde  $S$  es una matriz diagonal) que daba lugar a una matriz diagonalmente dominante. No obstante, las matrices de Nekrasov son  $H$ -matrices, por lo que podríamos buscar matrices de escalado que las lleven a forma estrictamente diagonalmente dominante. En [7, 13] se dan matrices de escalado que logran este fin imponiendo condiciones similares a la que hemos exigido inicialmente al calcular la inversa ( $h_i(A) \neq 0$  para todo  $i = 1, \dots, n$ , o que en cada fila, excepto en la última, exista un elemento no nulo a derecha de la diagonal). Como aplicación de la matriz de escalado, se obtendrán cotas superiores para la norma de la inversa de una matriz de Nekrasov. Estas cotas se podrán aplicar al estudio del condicionamiento de dichas matrices y también a la obtención de cotas para el error del problema de complementariedad lineal comentado en la sección 2.1.

En el primer apartado de este capítulo vamos a presentar unas matrices de escalado que no requieren la imposición de ninguna condición adicional y llevan una matriz de Nekrasov a forma estrictamente diagonalmente dominante. A continuación, se deduce una cota para la norma de la inversa de una matriz de esta clase a partir de dichas matrices de escalado. La cota es válida para cualquier matriz de Nekrasov y no requiere una parametrización de la matriz. En la tercera sección se presentan otras cotas para el mismo problema que se han publicado recientemente, y se concluye el capítulo haciendo una comparación de todas las cotas presentadas.

### 7.1. Matrices de escalado

Comenzamos este apartado obteniendo una matriz de escalado  $S$  que transforma cualquier matriz de Nekrasov en una matriz SDD. Después obtendremos otra matriz de escalado que aprovecha la existencia de elementos no nulos en  $A$  para reducir el coste computacional de la obtención de  $S$ .

**Teorema 7.1.** Sea  $A = (a_{ij})_{1 \leq i, j \leq n}$  una matriz de Nekrasov. Entonces la matriz

$$S = \begin{pmatrix} \frac{h_1(A) + \varepsilon_1}{|a_{11}|} & & \\ & \ddots & \\ & & \frac{h_n(A) + \varepsilon_n}{|a_{nn}|} \end{pmatrix},$$

$$\text{con } \begin{cases} \varepsilon_1 > 0, \\ 0 < \varepsilon_i \leq |a_{ii}| - h_i, \quad \varepsilon_i > \sum_{j=1}^{i-1} \frac{|a_{ij}| \varepsilon_j}{|a_{jj}|} \text{ para } i = 2, \dots, n. \end{cases}$$

es una matriz diagonal positiva tal que  $AS$  es estrictamente diagonalmente dominante.

*Demostración.* Comenzamos viendo que existen  $(\varepsilon_1, \dots, \varepsilon_n)$  tales que satisfacen el enunciado del teorema anterior. Tomamos  $\varepsilon_i = |a_{ii}| - h_i(A)$  para  $i = 2, \dots, n$ . Elegimos  $\varepsilon_1$  tal que  $\varepsilon_2 > \frac{|a_{21}| \varepsilon_1}{|a_{11}|}$ . Comprobamos si  $\varepsilon_3 > \sum_{j=1}^2 \frac{|a_{3j}| \varepsilon_j}{|a_{jj}|}$ . Si se verifica la desigualdad continuamos. En caso contrario, sustituimos  $\varepsilon_1$  y  $\varepsilon_2$  por  $\frac{\varepsilon_1}{\hat{M}_3}$  y  $\frac{\varepsilon_2}{\hat{M}_3}$  respectivamente, donde  $\hat{M}_3$  es un número positivo lo suficientemente grande, de forma que se verifique la desigualdad. Notemos que al dividir tanto el valor de  $\varepsilon_1$  como el de  $\varepsilon_2$  por el mismo número se seguirá satisfaciendo la desigualdad que relaciona a  $\varepsilon_1$  y  $\varepsilon_2$ . En el caso de  $\varepsilon_i$ , comprobamos si  $\varepsilon_i > \sum_{j=1}^{i-1} \frac{|a_{ij}| \varepsilon_j}{|a_{jj}|}$ . Si se cumple, continuamos. Si no, sustituimos el valor de  $\varepsilon_j$  por el de  $\frac{\varepsilon_j}{\hat{M}_i}$  para  $j = 1, \dots, i-1$  de forma que se verifique la desigualdad y, al igual que antes, se sigan cumpliendo las desigualdades que habíamos comprobado previamente. Por tanto, reiteramos el proceso hasta comprobar la desigualdad  $\varepsilon_n > \sum_{j=1}^{n-1} \frac{|a_{nj}| \varepsilon_j}{|a_{jj}|}$ . Si se satisface, tendremos ya los valores de  $(\varepsilon_1, \dots, \varepsilon_n)$  buscados. Si no, basta con sustituir el valor actual de  $\varepsilon_j$  por el de  $\frac{\varepsilon_j}{\hat{M}_n}$  para  $j = 1, \dots, n-1$ , eligiendo  $\hat{M}_n$  de forma que los nuevos valores de  $(\varepsilon_1, \dots, \varepsilon_n)$  verifiquen la desigualdad.

La matriz diagonal  $S$  es positiva puesto que  $h_i(A) \geq 0$  y  $\varepsilon_i > 0$  por hipótesis. El elemento  $(i, j)$  de la matriz  $AS$  viene dado por la expresión  $a_{ij} \frac{h_j(A) + \varepsilon_j}{|a_{jj}|}$ . Para comprobar que  $AS$  es estrictamente diagonal dominante se comienza comprobando que se cumple la condición de dominancia diagonal estricta en la fila  $n$ -ésima:

$$\sum_{j=1}^{n-1} |a_{nj}| \frac{h_j(A) + \varepsilon_j}{|a_{jj}|} = \underbrace{\sum_{j=1}^{n-1} |a_{nj}| \frac{h_j(A)}{|a_{jj}|}}_{h_n(A)} + \sum_{j=1}^{n-1} |a_{nj}| \frac{\varepsilon_j}{|a_{jj}|} < h_n(A) + \varepsilon_n = |(AS)(n, n)|$$

Del mismo modo, se comprueba la fila  $n-1$ :

$$\begin{aligned} \sum_{j=1}^{n-2} |a_{n-1,j}| \frac{h_j(A) + \varepsilon_j}{|a_{jj}|} + |a_{n-1,n}| \underbrace{\frac{h_n(A) + \varepsilon_n}{|a_{nn}|}}_{\leq 1} &\leq h_{n-1}(A) + \sum_{j=1}^{n-2} |a_{n-1,j}| \frac{\varepsilon_j}{|a_{jj}|} < h_{n-1}(A) + \varepsilon_{n-1} = \\ &= |(AS)(n-1, n-1)| \end{aligned}$$

La primera desigualdad es debida a la hipótesis  $\varepsilon_n \leq |a_{nn}| - h_n(A)$ , por la que se tiene  $\frac{h_n(A) + \varepsilon_n}{|a_{nn}|} \leq 1$ . En general, para la fila  $i$ -ésima:

$$\sum_{j=1}^{i-1} |a_{ij}| \frac{h_j(A) + \varepsilon_j}{|a_{jj}|} + \sum_{j=i+1}^n |a_{ij}| \frac{h_j(A) + \varepsilon_j}{|a_{jj}|} \leq h_i(A) + \sum_{j=1}^{i-1} |a_{ij}| \frac{\varepsilon_j}{|a_{jj}|} < h_i(A) + \varepsilon_i$$

y, cuando  $i = 1$ :

$$\sum_{j=2}^n |a_{1j}| \frac{h_j(A) + \varepsilon_j}{|a_{jj}|} \leq h_1(A) < h_1(A) + \varepsilon_1.$$

En el estudio de la fila  $i$ -ésima se utiliza la hipótesis  $\varepsilon_j \leq |a_{jj}| - h_j(A)$  para  $j = i + 1, \dots, n$ , así como que  $\varepsilon_i > \sum_{j=1}^{i-1} \frac{|a_{ij}| \varepsilon_j}{|a_{jj}|}$ . Cuando  $i = 1$ , esta última hipótesis se reduce a que  $\varepsilon_1$  sea estrictamente positivo.  $\square$

En el Teorema 7.1 damos una matriz de escalado  $S$  que sirve para cualquier matriz de Nekrasov. No obstante, teniendo en cuenta su estructura de ceros, podemos dar una matriz que en muchos de los casos será más sencilla y su producto dará lugar también a una matriz SDD. La clave consiste en buscar la primera fila  $k (\leq n)$  para la cual no exista ningún elemento no nulo a la derecha del elemento diagonal. Entonces, podremos imponer  $\varepsilon_j = 0$  para  $j = 1, \dots, k - 1$  y el producto  $AS$  será también SDD con alguna pequeña matización sobre la elección de los demás  $\varepsilon_j$ ,  $j = k, \dots, n$ . Si  $k = 1$ , el resultado se corresponde con el Teorema anterior.

**Teorema 7.2.** Sea  $A = (a_{ij})_{1 \leq i, j \leq n}$  una matriz de Nekrasov, y sea  $k \in \{1, \dots, n\}$  el primer índice tal que no existe  $a_{kj} \neq 0$  con  $j > k$ . Entonces, la matriz

$$S = \begin{pmatrix} \frac{h_1(A) + \varepsilon_1}{|a_{11}|} & & & \\ & \ddots & & \\ & & \ddots & \\ & & & \frac{h_n(A) + \varepsilon_n}{|a_{nn}|} \end{pmatrix}, \quad (7.1)$$

$$\text{con } \begin{cases} \varepsilon_i = 0, & i = 1, \dots, k - 1, \\ 0 < \varepsilon_i < |a_{ii}| - h_i, & \varepsilon_i > \sum_{j=k}^{i-1} \frac{|a_{ij}| \varepsilon_j}{|a_{jj}|} \text{ para } i = k, \dots, n. \end{cases}$$

es una matriz diagonal positiva tal que  $AS$  es estrictamente diagonalmente dominante.

*Demostración.* La demostración de este resultado es similar a la del Teorema 7.1. Comenzamos razonando que existen  $(\varepsilon_1, \dots, \varepsilon_n)$  tales que satisfacen el enunciado del resultado. Dado que  $A$  es una matriz de Nekrasov, se tiene que  $|a_{ii}| > h_i(A)$  para  $i = 1, \dots, n$ , lo cual sumado a la forma constructiva dada en la demostración del Teorema anterior asegura que podemos encontrar tales  $(\varepsilon_1, \dots, \varepsilon_n)$ . Notemos además que siempre se cumplirá que  $\varepsilon_n \neq 0$ , dado que  $k$  tomará como mucho el valor  $n$ . Veamos que en cualquier caso la matriz  $AS$  es también SDD.

Consideramos primero el caso de la fila  $i$ -ésima, cuando  $i < k$ :

$$\sum_{j=1}^{i-1} |a_{ij}| \frac{h_j(A) + \varepsilon_j}{|a_{jj}|} + \sum_{j=i+1}^n |a_{ij}| \frac{h_j(A) + \varepsilon_j}{|a_{jj}|} < h_i(A) + \sum_{j=1}^{i-1} |a_{ij}| \frac{\varepsilon_j}{|a_{jj}|} = h_i(A).$$

En este caso, se tiene la relación de dominancia diagonal estricta debido a que el Lema 5.4 nos asegura que  $h_i(A) \neq 0$  y a la cota superior estricta que satisface por hipótesis  $\varepsilon_j$  cuando  $j > i$ . Veamos ahora que también se tiene dominancia diagonal estricta en la fila  $k$ -ésima:

$$\sum_{j=1}^{k-1} |a_{kj}| \frac{h_j(A) + \varepsilon_j}{|a_{jj}|} + \sum_{j=k+1}^n |a_{kj}| \frac{h_j(A) + \varepsilon_j}{|a_{jj}|} = \sum_{j=1}^{k-1} |a_{kj}| \frac{h_j(A)}{|a_{jj}|} = h_k(A) < h_k(A) + \varepsilon_k.$$

En esta fila, dado que  $a_{kj} = 0$  siempre que  $j > k$ , se tiene que necesitamos elegir un  $\varepsilon_k > 0$  para lograr dominancia estricta. Además, la elección de  $\varepsilon_k > 0$  implicará que necesitamos que

$\varepsilon_j > 0$  siempre que  $j > k$ . Por último, comprobemos que también se da la relación de forma estricta para la fila  $i$ -ésima cuando  $i > k$ :

$$\sum_{j=1}^{i-1} |a_{ij}| \frac{h_j(A) + \varepsilon_j}{|a_{jj}|} + \sum_{j=i+1}^n |a_{ij}| \frac{h_j(A) + \varepsilon_j}{|a_{jj}|} \leq h_i(A) + \sum_{j=k}^{i-1} |a_{ij}| \frac{\varepsilon_j}{|a_{jj}|} < h_i(A) + \varepsilon_i.$$

□

En particular, si para todo  $i = 1, \dots, n-1$  existe al menos un elemento  $a_{ij} \neq 0$  para algún índice  $j > i$ , en la matriz  $S$  podremos anular todos los  $\varepsilon_i$  excepto  $\varepsilon_n$ . Este caso particular es el que aparece ya recogido en [13].

En cualquier caso, el tener una matriz de escalado que nos permita llevar una matriz de Nekrasov a forma estrictamente diagonalmente dominante nos da una forma rápida de dar una cota de la norma de su inversa, sin tener que proceder a calcular dicha inversa como en capítulos anteriores. Este procedimiento se mostrará en el siguiente apartado.

## 7.2. Cota para la norma de la inversa

Comenzamos esta sección recordando un resultado clásico presentado por Varah en [28] para acotar la norma de la inversa de una matriz SDD. Este resultado se utiliza también con frecuencia para obtener cotas de la norma de las inversas de matrices pertenecientes a clases relacionadas con las SDD. Se corresponde con el Teorema 1 de [28]:

**Teorema 7.3.** *Sea  $A = (a_{ij})_{1 \leq i, j \leq n}$  una matriz estrictamente diagonalmente dominante por filas. Entonces*

$$\|A^{-1}\|_{\infty} < \frac{1}{\min_i (|a_{ii}| - \sum_{j \neq i} |a_{ij}|)}. \quad (7.2)$$

En el Teorema 7.3 se aprecia que la bondad de la cota que obtengamos al llevar nuestra matriz de Nekrasov a forma SDD depende de la “holgura” que logremos en cada fila al realizar el escalado  $AS$ . A continuación, enunciamos la cota de la inversa de una matriz de Nekrasov. Una vez definida la matriz  $S$  según el Teorema 7.1 o el Teorema 7.2 podemos conseguir una cota u otra de la norma infinito de la inversa de una matriz de Nekrasov. Posteriormente usaremos la que utiliza el Teorema 7.2 por tener menor coste computacional.

**Teorema 7.4.** *Sea  $A = (a_{ij})$  una matriz de Nekrasov. Entonces*

$$\|A^{-1}\|_{\infty} < \frac{\max_{i \in N} \left( \frac{h_i(A) + \varepsilon_i}{|a_{ii}|} \right)}{\min_{i \in N} (\varepsilon_i - w_i + p_i)}, \quad (7.3)$$

donde  $(\varepsilon_1, \dots, \varepsilon_n)$  vienen dados por el Teorema 7.1 o el Teorema 7.2,  $w_i = \sum_{j=1}^{i-1} |a_{ij}| \frac{\varepsilon_j}{|a_{jj}|}$  y  $p_i = \sum_{j=i+1}^n |a_{ij}| \frac{|a_{jj}| - h_j(A) - \varepsilon_j}{|a_{jj}|}$ .

*Demostración.* Observemos que:

$$\|A^{-1}\|_{\infty} = \|S(S^{-1}A^{-1})\|_{\infty} = \|S(AS)^{-1}\|_{\infty} \leq \|S\|_{\infty} \|(AS)^{-1}\|_{\infty}.$$

En esta expresión, podemos obtener la norma infinito de  $S$  fácilmente ya que es una matriz diagonal, y aplicar el Teorema 7.3 a  $\|(AS)^{-1}\|_{\infty}$ , que por el Teorema 7.1 o 7.2 (según la matriz

$S$  que utilizemos) es SDD. Entonces, queda solamente estudiar cómo de dominante es la fila  $i$ -ésima de la matriz  $AS$ :

$$\begin{aligned} h_i(A) + \varepsilon_i - \sum_{j \neq i} |a_{ij}| \frac{h_j(A) + \varepsilon_j}{|a_{jj}|} &= \varepsilon_i - \sum_{j=1}^{i-1} |a_{ij}| \frac{\varepsilon_j}{|a_{jj}|} + \sum_{j=i+1}^n |a_{ij}| \frac{|a_{jj}| - h_j(A) - \varepsilon_j}{|a_{jj}|} \\ &= \varepsilon_i - w_i + p_i, \end{aligned}$$

donde simplemente hemos utilizado la definición de  $h_i(A)$ , (4.1).  $\square$

Como la matriz  $S$  cumple que  $\|S\|_\infty \leq 1$ , podemos dar la siguiente simplificación de la fórmula anterior para la cota de la inversa de una matriz de Nekrasov apoyándonos en el Teorema 7.4:

**Corolario 7.5.** *Sea  $A = (a_{ij})$  una matriz de Nekrasov. Entonces*

$$\|A^{-1}\|_\infty < \frac{1}{\min_{i \in N} (\varepsilon_i - w_i + p_i)},$$

donde  $(\varepsilon_1, \dots, \varepsilon_n)$  vienen dados por el Teorema 7.1 o el Teorema 7.2,  $w_i = \sum_{j=1}^{i-1} |a_{ij}| \frac{\varepsilon_j}{|a_{jj}|}$  y  $p_i = \sum_{j=i+1}^n |a_{ij}| \frac{|a_{jj}| - h_j(A) - \varepsilon_j}{|a_{jj}|}$ .

A continuación vamos a presentar el Algoritmo 7, que permite conseguir todos los parámetros necesarios para obtener la matriz  $S$  del Teorema 7.2 y las sumas de filas de la matriz  $AS$ , necesarias para dar la cota 7.3 que hemos definido anteriormente, con los  $\varepsilon_i$   $i = 1, \dots, n$  como en el Teorema 7.2. Si repasamos la forma de obtener la matriz  $S$  en este caso observamos que hay cierta libertad a la hora de elegir los valores de  $(\varepsilon_1, \dots, \varepsilon_n)$ . En el siguiente algoritmo, el valor al que se inicializa  $\varepsilon_i$  será 0 o  $t\Delta_i(A)$ , siendo  $t$  un parámetro que podemos elegir en el intervalo  $(0, 1)$ . También podríamos dar una mayor libertad eligiendo un  $t_i$  para cada  $\varepsilon_i$  no nulo. No obstante, el criterio para elegir cada uno no es claro, por lo que dado que muchas veces se tendrá que  $\varepsilon_i = 0$  para  $i = 1, \dots, n-1$ , podemos utilizar esta elección simplificada. Eso sí, la bondad de la cota obtenida depende de la buena elección del valor de  $t \in (0, 1)$ . Partiendo de la parametrización de la matriz  $M(A)$  (dada en (4.2)) calculamos los valores de  $h_i(A)$ ,  $|a_{ii}|$ ,  $\varepsilon_i$  y  $s_i$  para  $i = 1, \dots, n$ . En  $s_i$  se obtienen los valores necesarios para aplicar el Teorema 7.4 y obtener nuestra cota de  $\|A^{-1}\|_\infty$ .

### 7.3. Cotas alternativas para la norma de la inversa

En la sección anterior hemos dado un procedimiento para obtener una cota de la norma de la inversa de una matriz de Nekrasov basándonos en utilizar una matriz de escalado. En diversos artículos recientes, se ha abordado el mismo problema desde otro punto de vista, que se apoya también en el resultado de Varah (Teorema 7.3). Por tanto, a continuación vamos a introducir la base que se utiliza en los mismos para obtener esas cotas para así después proceder a realizar una comparación de ambas técnicas.

El punto de partida de esas cotas se basa en expresar la matriz  $A$  como  $A = D - L - U$ , donde  $D$  es la diagonal de  $A$ ,  $L$  su parte triangular inferior (sin la diagonal) y  $U$  su parte triangular superior (también sin la diagonal). Es decir, las matrices  $L = (l_{ij})_{1 \leq i, j \leq n}$ ,  $D = (d_{ij})_{1 \leq i, j \leq n}$ ,  $U = (u_{ij})_{1 \leq i, j \leq n}$  en este caso son:

$$l_{ij} = \begin{cases} a_{ij} & \text{si } i > j \\ 0 & \text{si } i \leq j \end{cases}, \quad d_{ij} = \begin{cases} a_{ij} & \text{si } i = j \\ 0 & \text{si } i \neq j \end{cases}, \quad u_{ij} = \begin{cases} a_{ij} & \text{si } i < j \\ 0 & \text{si } i \geq j \end{cases}.$$

**Algoritmo 7** nektoSDD - Obtención de la cota 7.3

---

**Input:**  $A = (a_{ij})(i \neq j), \Delta, t$   $\triangleright \Delta$  es el vector de los parámetros  $\Delta_j(M(A))$

**for**  $i = 1 : n$

$h_i = \sum_{j=1}^{i-1} |a_{ij}| k_j$

$r = \sum_{j=i+1}^n |a_{ij}|$

**if**  $r == 0, J == 0$   $\triangleright$  Detección de la primera fila para la que  $\varepsilon_i > 0$

$J=i;$

**end if**

$h_i = h_i + r$

$a_{ii} = \Delta_i + h_i$

$k_i = h_i / |a_{ii}|$

**end for**

$\varepsilon_K = t \Delta_K$

$w_1 = \dots = w_K = 0$   $\triangleright$  Cuando  $i \leq K$ , se tiene que  $w_i = 0$

**for**  $i = K + 1 : n$

$\varepsilon_i = t \Delta_i$

$p_j = \varepsilon_j / |a_{jj}|$

$w_i = \sum_{j=K}^{i-1} |a_{ij}| p_j$

**if**  $w_i - \varepsilon_i > 0$

$M = 1/2w_i$

**for**  $j = K : i - 1$

$\varepsilon_j = \varepsilon_j \varepsilon_i M$

$w_j = w_j \varepsilon_i M$

**end for**

$w_i = \varepsilon_i / 2$

**end if**

**end for**

**for**  $i=n:-1:2$

$s_i = \varepsilon_i - w_i + \sum_{j=i+1}^n |a_{ij}| f_j$

$f_i = (\Delta_i - \varepsilon_i) / |a_{ii}|$

**end for**

$s_1 = \varepsilon_1 - w_1 + \sum_{j=2}^n |a_{1j}| f_j$

Cálculo de la cota utilizando la fórmula (7.3).  $\triangleright$  Utilizando que  $s_i = \varepsilon_i - w_i + p_i$

---

Con esta descomposición, se tiene el siguiente Lema:

**Lema 7.6.** *Dada una matriz  $A = (a_{ij})_{1 \leq i, j \leq n}$  compleja tal que  $a_{ii} \neq 0$  para  $i = 1, \dots, n$ , entonces*

$$h_i(A) = |a_{ii}| \left( (|D| - |L|)^{-1} |U| e \right)_i \quad (7.4)$$

donde  $e$  es el vector de todo unos.

Apoyándose en el Lema 7.6, en [27] (Remark 3.3) se consigue una caracterización de matriz de Nekrasov:

**Teorema 7.7.** *Una matriz compleja  $A = (a_{ij})_{1 \leq i, j \leq n}$  es de Nekrasov si y solo si*

$$(|D| - |L|)^{-1} |U| e < e,$$

es decir, si y solo si  $I_n - (|D| - |L|)^{-1} |U|$  es una matriz SDD, donde  $I_n$  es la matriz identidad  $n \times n$ .



Vemos que la caracterización de que una matriz sea de Nekrasov viene dada por la dominancia diagonal estricta de la matriz  $I_n - (|D| - |L|)^{-1}|U|$ . La base de las cotas que se han obtenido en el artículo [5] y en artículos en los que se intentan mejorar éstas (véase [18],[21]) se basan en utilizar adecuadamente el Teorema 7.3 en una matriz obtenida a partir de la expresión  $I_n - (|D| - |L|)^{-1}|U|$ . Las cotas obtenidas de esta forma emplean los siguientes parámetros:

$$z_i(A) = \begin{cases} 1, & i = 1, \\ \sum_{j=1}^{i-1} \frac{|a_{ij}|}{|a_{jj}|} z_j(A) + 1, & i = 2, \dots, n. \end{cases} \quad (7.5)$$

Con dichos parámetros, estamos en condiciones de introducir las siguientes cotas, presentadas en [5]:

**Teorema 7.8.** *Sea  $A = (a_{ij})_{1 \leq i, j \leq n}$  una matriz de Nekrasov. Entonces,*

$$\|A^{-1}\|_{\infty} \leq \frac{\max_{i \in N} \frac{z_i(A)}{|a_{ii}|}}{1 - \max_{i \in N} \frac{h_i(A)}{|a_{ii}|}}, \quad (7.6)$$

y

$$\|A^{-1}\|_{\infty} \leq \frac{\max_{i \in N} z_i(A)}{\min_{i \in N} (|a_{ii}| - h_i(A))}. \quad (7.7)$$

## 7.4. Comparación de cotas

Como hemos mencionado anteriormente, estas cotas del apartado anterior se han obtenido mediante un planteamiento distinto al que hemos seguido al comienzo del capítulo. Por tanto, es de interés comparar la bondad de ambos tipos de cota. Comencemos estudiando el coste computacional que acarrea el cálculo de cada una de las cotas. Obtener la cotas (7.6) o (7.7) supone realizar  $\frac{3}{2}n(n-1)$  sumas y  $n(n-1)$  productos. Por otro lado, están las cotas dadas en esta memoria. Si echamos un vistazo al Algoritmo 7, vemos que hay unas condiciones *if* que aumentan el coste en algún ejemplo. Por tanto, aquí consideraremos el peor caso, en el que siempre se realicen todos los cálculos dentro de estas condiciones. Además, hay que tener en cuenta que el coste del algoritmo depende del índice  $K$  definido en el mismo. En el contexto del Teorema 7.2, este índice representa la primera fila para la que hay que tomar un  $\varepsilon_j$  no nulo. El Teorema 7.1 puede verse como un caso particular a la hora de contar las operaciones realizadas, si tomamos siempre  $K = 1$ .

A continuación vamos a mostrar el coste computacional dependiendo tanto de  $n$  como del índice  $K$ . No obstante, los casos particulares  $K = 1$  y  $K = n$  son los que se presentan en muchos ejemplos, así que mostraremos también el coste en estos casos particulares:

Operaciones	general	$K = n$	$K = 1$
sumas/restas	$\frac{3n^2+n+2}{2} + \frac{(n-K-1)(n-K)}{2}$	$\frac{3n^2+n+2}{2}$	$2n^2 - n + 2$
multiplicaciones	$\frac{7n^2+9n+4}{2} + \frac{5K^2-10Kn-11K}{2}$	$n(n-1)$	$\frac{7n^2-n-2}{2}$
divisiones	$2n - 1 + 2(n - K)$	$2n - 1$	$4n - 3$

Tabla 7.1: Coste computacional de la cota (7.3)

Con este objetivo, vamos a contar todas las operaciones llevadas a cabo y comparar el término  $T$  de mayor grado del coste de las cotas:

	(7.6), (7.7)	$K = n$	$K = 1$
$T$	$\frac{5}{2}n^2$	$\frac{5}{2}n^2$	$\frac{11}{2}n^2$

Tabla 7.2: Término de mayor grado del coste computacional de las cotas de  $\|A^{-1}\|_\infty$ 

En esta tabla vemos que el utilizar la matriz dada en el Teorema 7.1 puede acarrear en el peor de los casos un poco más del doble de operaciones. En cualquier caso, el algoritmo para obtener la cota es de orden cuadrático.

Una vez comprobado que el coste computacional es similar, nuestro interés radica en comparar la bondad de la acotación obtenida en cada caso. Para comenzar tal comparación, vamos a utilizar la batería de ejemplos utilizada en [5] y en los artículos posteriores:

$$A_1 = \begin{pmatrix} -7 & 1 & -0,2 & 2 \\ 7 & 88 & 2 & -3 \\ 2 & 0,5 & 13 & -2 \\ 0,5 & 3 & 1 & 6 \end{pmatrix}, \quad A_2 = \begin{pmatrix} 8 & 1 & -0,2 & 3,3 \\ 7 & 13 & 2 & -3 \\ -1,3 & 6,7 & 13 & -2 \\ 0,5 & 3 & 1 & 6 \end{pmatrix},$$

$$A_3 = \begin{pmatrix} 21 & -9,1 & -4,2 & -2,1 \\ -0,7 & 9,1 & -4,2 & -2,1 \\ -0,7 & -0,7 & 4,9 & -2,1 \\ -0,7 & -0,7 & -0,7 & 2,8 \end{pmatrix}, \quad A_4 = \begin{pmatrix} 5 & 1 & 0,2 & 2 \\ 1 & 21 & 1 & -3 \\ 2 & 0,5 & 6,4 & -2 \\ 0,5 & -1 & 1 & 9 \end{pmatrix},$$

$$A_5 = \begin{pmatrix} 6 & -3 & -2 \\ -1 & 11 & -8 \\ -7 & -3 & 10 \end{pmatrix}, \quad A_6 = \begin{pmatrix} 8 & -0,5 & -0,5 & -0,5 \\ -9 & 16 & -5 & -5 \\ -6 & -4 & 15 & -3 \\ -4,9 & -0,9 & -0,9 & 6 \end{pmatrix}.$$

En la siguiente tabla comparamos los resultados obtenidos empleando las cotas (7.6),(7.7), la cota (7.4) obtenida en este capítulo y, cuando la matriz es SDD, la cota de Varah (Teorema 7.3). En la fórmula (7.4), utilizamos los parámetros  $\varepsilon_i$   $i = 1, \dots, n$  de la forma del Teorema 7.2. Para esta batería de ejemplos, elegir una matriz como la del Teorema 7.2 supone tener todos los parámetros epsilon nulos salvo  $\varepsilon_n$ . Las celdas coloreadas contienen la mejor cota obtenida para cada matriz.

Matriz	Norma (exacta)	Varah	(7.6)	(7.7)	(7.3)
$A_1$	0.1921	0.6667	0.3805	0.5263	0.3521
$A_2$	0.2390	1	0.8848	0.6885	0.9015
$A_3$	0.8759	1.4286	1.8076	0.9676	1.3563
$A_4$	0.2707	0.5556	0.6200	0.7937	0.4484
$A_5$	1.1519	-	1.4909	2.4848	1.1658
$A_6$	0.4474	-	1.1557	0.5702	1.0850

Tabla 7.3: Cota superior de  $\|A^{-1}\|_\infty$ , (7.3) usa el Teorema 7.2

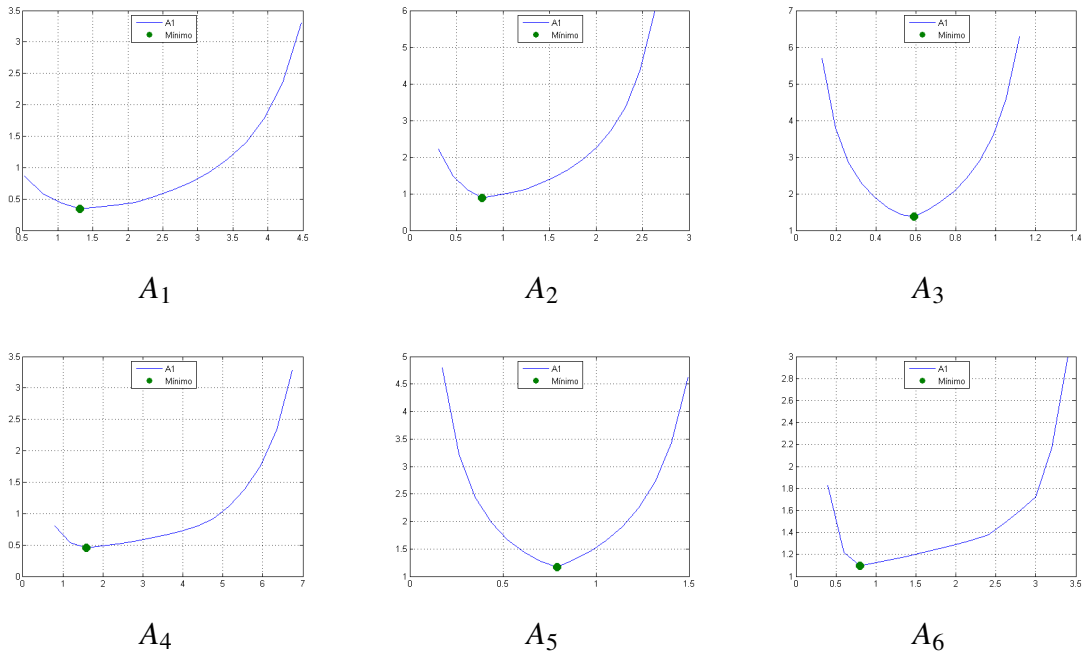
En la Tabla 7.3, mirando la columna (7.3) vemos la cota obtenida imponiendo  $\varepsilon_i = 0$  en todas las filas excepto en la última. El resultado logrado mejora las cotas obtenidas en el otro artículo en varios casos, pero no siempre es mejor. A continuación vamos a estudiar las cotas obtenidas empleando de nuevo el Teorema 7.2, inicializando los valores de  $\varepsilon_i$  como 0 o

$\Delta_i(A)/2$ , según corresponda, y compararemos las cotas con las obtenidas empleando las fórmulas (7.6) y (7.7). Las matrices sobre las que utilizaremos las cotas son matrices  $20 \times 20$ . Los resultados se incluyen en la Tabla 7.4. En ella, las celdas coloreadas también contienen la mejor cota obtenida para esa matriz.

Norma (exacta)	(7.6)	(7.7)	(7.3)
6.7500e+03	6.2740e+06	7.0302e+06	4.6267e+06
4.9663e+03	7.5983e+05	3.1374e+06	1.9948e+06
4.9565e+03	9.6626e+05	1.8445e+07	1.9506e+07
6.5044e+03	1.4000e+06	5.2040e+06	3.1536e+06
5.2081e+03	4.0631e+06	6.7523e+06	2.7879e+05
5.9789e+03	2.9510e+05	5.3717e+05	5.0178e+06
6.0290e+03	1.3081e+06	2.0881e+06	1.2653e+06
4.1255e+03	3.7089e+06	1.7445e+07	1.5422e+05
4.9070e+03	1.8392e+06	3.7896e+06	1.6405e+06
5.2453e+03	3.0964e+06	2.9300e+06	1.6907e+07

Tabla 7.4: Cota superior de  $\|A^{-1}\|_\infty$ , (7.3) usa el Teorema 7.2

En la mitad de los casos, la cota obtenida mediante escalado es mejor que las demás. Y cabe intentar mejorar la misma estudiando la forma óptima de inicializar los parámetros. El problema reside en discernir la forma óptima de los mismos. Por ejemplo, si volvemos a las matrices de la batería de ejemplos, podemos representar la cota obtenida según el valor de  $\epsilon_n \in (0, \Delta_n(A))$ . En las siguientes gráficas representamos el valor de la cota obtenida según los valores iniciales de  $\epsilon_n$ . El valor mínimo aparece resaltado, y es el que se ha incluido en la Tabla 7.3.



Hasta ahora, hemos elegido los parámetros  $\epsilon_i$   $i = 1, \dots, n$  siguiendo el Teorema 7.2. Las razones para esta elección son el menor coste computacional, y la facilidad para elegir unos valores iniciales para  $\epsilon_i$   $i = 1, \dots, n$ . No obstante, también es de interés ver qué ocurre empleando la otra elección. En la Tabla 7.5 comparamos los resultados obtenidos sobre la batería de ejemplos utilizando una matriz  $S$  de la forma dada en el Teorema 7.1.

Matriz	Norma (exacta)	Varah	(7.6)	(7.7)	(7.3)
$A_1$	0.1921	0.6667	0.3805	0.5263	0.2354
$A_2$	0.2390	1	0.8848	0.6885	0.5260
$A_3$	0.8759	1.4286	1.8076	0.9676	0.9273
$A_4$	0.2707	0.5556	0.6200	0.7937	0.3168
$A_5$	1.1519	-	1.4909	2.4848	1.1588
$A_6$	0.4474	-	1.1557	0.5702	0.4527

Tabla 7.5: Cota superior de  $\|A^{-1}\|_{\infty}$ , (7.3) usa el Teorema 7.1

A primera vista, parece que esta elección es superior a las demás: podemos lograr valores muy próximos al valor exacto de la norma, siendo las estimaciones mejores que las dadas por cualquier otra fórmula en todos los casos. No obstante, la cota presenta un problema subyacente: la inicialización de los parámetros  $\varepsilon$ . Dados con el valor adecuado, la cota obtenida es muy buena. No obstante, una mala elección de los mismos da lugar a cotas desastrosas. Por ilustrar este hecho con un ejemplo sencillo fijémonos en  $A_5$ . Utilizando el mismo criterio empleado en la Tabla 7.4, es decir, escoger como valor inicial de  $\varepsilon_i = \Delta_i/2$  para  $i = 1, \dots, n$  da lugar a la cota:  $2,1392e + 15$ . Para rellenar los datos de la columna AS(2) de la Tabla anterior lo que hemos hecho ha sido probar distintos datos de los parámetros y quedarnos con la mejor cota.

# Referencias

- [1] A. A. Ahac and D. D. Olesky. A stable method for the LU factorization of M-matrices. *SIAM. J. Alg. Disc. Math.*, 7: 368–378, 1986.
- [2] A. S. Alfa, J. Xue, and Q. Ye. Entrywise perturbation theory for diagonally dominant M-matrices with applications. *Numerische Mathematik*, 90(3): 401–414, 2002.
- [3] A. Barreras and J. M. Peña. Accurate and efficient LDU decompositions of diagonally dominant M-matrices. *Electron. J. Linear Algebra*, 24: 152–167, 2012/13.
- [4] A. Berman and R. J. Plemmons. *Nonnegative matrices in the mathematical sciences*. Academic Press [Harcourt Brace Jovanovich, Publishers], New York-London, 1979.
- [5] L. Cvetković, P.-F. Dai, K. Doroslovački, and Y.-T. Li. Infinity norm bounds for the inverse of Nekrasov matrices. *Appl. Math. Comput.*, 219(10): 5020–5024, 2013.
- [6] L. Cvetković, V. Kostić, and M. Nedović. Generalizations of Nekrasov matrices and applications. *Open Math.*, 13(5): 96–105, 2015.
- [7] L. Cvetković, T. Szulc, and M. Nedović. Scaling technique for partition-Nekrasov matrices. *Appl. Math. Comput.*, 271: 201–208, 2015.
- [8] J. Demmel. *Applied Numerical Linear Algebra*. SIAM, 1997.
- [9] J. Demmel, I. Dumitriu, O. Holtz, and P. Koev. Accurate and efficient expression evaluation and linear algebra. *Acta Numer.*, 17: 87–145, 2008.
- [10] J. Demmel, M. Gu, S. Eisenstat, I. Slapničar, K. Veselić, and Z. Drmač. Computing the singular value decomposition with high relative accuracy. *Linear Algebra Appl.*, 299: 21–80, 1999.
- [11] J. Demmel and P. Koev. Accurate SVDs of weakly diagonally dominant M-matrices. *Numer. Math.*, 98: 99–104, 2004.
- [12] K. Fan. Note on M-matrices. *Q. J. Math.*, 11: 43–49, 1960.
- [13] M. García-Esnaola and J. M. Peña. Error bounds for linear complementarity problems of Nekrasov matrices. *Numer. Algorithms*, 67(3): 655–667, 2014.
- [14] M. García-Esnaola and J. M. Peña. B-nekrasov matrices and error bounds for linear complementarity problems. *Numerical Algorithms*, 72(2): 435–445, 2016.
- [15] N. Higham. *Accuracy and Stability of Numerical Algorithms*. SIAM, 2002.
- [16] C. R. Johnson and R. A. Horn. *Matrix Analysis*. Cambridge University Press, 1990.

- [17] C. R. Johnson and R. A. Horn. *Topics in Matrix Analysis*. Cambridge University Press, 1991.
- [18] L. Y. Kolotilina. On bounding inverses to Nekrasov matrices in the infinity norm. *Journal of Mathematical Sciences*, 199(4): 432–437, 2014.
- [19] L. Y. Kolotilina. Some characterizations of Nekrasov and S-Nekrasov matrices. *J. Math. Sci. (N.Y.)*, 207(5): 767–775, 2015.
- [20] C. Li, Q. Liu, L. Gao, and Y. Li. Subdirect sums of Nekrasov matrices. *Linear and Multilinear Algebra*, 64(2): 208–218, 2016.
- [21] C. Li, H. Pei, A. Gao, and Y. Li. Improvements on the infinity norm bound for the inverse of Nekrasov matrices. *Numerical Algorithms*, 71(3): 613–630, 2016.
- [22] M. Mehmke and P. Nekrasov. Solutions of linear systems of equations by means of successive approximations. *Mat. Sb.*, 16: 437–459, 1892.
- [23] H. Minc. *Nonnegative matrices*. Wiley-Interscience Series in Discrete Mathematics and Optimization. A Wiley-Interscience Publication. John Wiley & Sons, Inc., New York, 1988.
- [24] J. M. Peña. Pivoting strategies leading to diagonal dominance by rows. *Numer. Math.*, 81(2): 293–304, 1998.
- [25] J. M. Peña. LDU decompositions with L and U well conditioned. *Electron. Trans. Numer. Anal.*, 18: 198–208, 2004.
- [26] L. L. Shi, Q. Xu, Zhong; Lu, and W. W. Zhou. New iterative criteria for generalized Nekrasov matrices (Chinese). *J. Numer. Methods Comput. Appl.*, 34(2): 117–122, 2013.
- [27] T. Szulc. Some remarks on a theorem of Gudkov. *Linear Algebra Appl.*, 225: 221–235, 1995.
- [28] J. M. Varah. A lower bound for the smallest singular value of a matrix. *Linear Algebra and Appl.*, 11: 3–5, 1975.
- [29] Y. Y. Wang, Z. Xu, and Q. Lu. Practical criteria for generalized Nekrasov matrices (chinese). *J. Numer. Methods Comput. Appl.*, 35(3): 171–180, 2014.
- [30] Y. Y. Wang, Z. Xu, and Q. Lu. Iterative criteria for generalized Nekrasov matrices (Chinese). *Numer. Math. J. Chinese Univ.*, 37(1): 19–30, 2015.