

Un método simplificado de Newton para calcular la raíz cuadrada de una matriz real simétrica definida positiva

Alfredo Mendoza Mexía · Oscar Rubén Gómez Aldama

Recibido: Julio 2009, Aceptado: Noviembre 2009
©Universitat Politècnica de Catalunya, Barcelona, España 2010

Resumen Uno de los métodos numéricos estables más utilizado para calcular la raíz cuadrada de una matriz es el Método de Newton (MN) el cual tiene la desventaja de que, en cada iteración, es necesario resolver una ecuación matricial de Sylvester para matrices densas resultando computacionalmente costoso. Se han hecho simplificaciones al MN, resultando un par de iteraciones simplificadas. Higham [4] demostró, a través de la teoría de perturbaciones, que éstas son numéricamente inestables. En este artículo se desarrolla una iteración alternativa simplificada del Método de Newton (IASMN) para calcular la raíz cuadrada de una matriz real simétrica definida positiva, la cual es atractiva por ser convergente, computacionalmente económica y, para propósitos prácticos, es numéricamente estable.

A SIMPLIFIED NEWTON'S METHOD TO CALCULATE THE SQUARE ROOT OF A POSITIVE DEFINITE REAL MATRIX

Summary One of the numerically stable methods to calculate the square root of a matrix is the Newton's Method, but it has the disadvantage that in each iteration it is necessary to solve a Sylvester's matrix equation for full matrix, which is computationally expensive.

Alfredo Mendoza Mexía
Centro de Investigación Científica y de Educación Superior de Ensenada (CICESE)
División Ciencias de la Tierra, Dpto. de Geofísica Aplicada
Baja California, México
e-mail: amendoza@cicese.mx

Oscar Rubén Gómez Aldama
Dpto. de Física Matemáticas y Ciencias e Ingenierías
Universidad de Sonora, URS, Navojoa Sonora, México
e-mail: orgomez@navojoa.uson.mx

Modifications to the method have been made, resulting a pair of simplified iterations, but it has been proved, through the perturbations theory, that these are numerically unstable. In this article, a simplified alternative iteration of Newton's Method for the calculation of the square root of a positive definite real matrix is presented, which is attractive for being convergent, computationally economic and, for practical purposes, numerically stable.

1. Introducción

Un método usual para calcular la raíz cuadrada de una matriz es el Método de Newton (MN) [13, 20], además de este método existen muchos otros, por ejemplo: Higham [21] analizó el comportamiento de un conjunto de métodos para calcular la raíz cuadrada de matrices en general, contemplando también el caso hermitiano. Para el cálculo de raíces cuadradas de matrices en general, Meini [1] desarrolló el método de reducción cíclica y, más recientemente, Long [13] desarrolló el algoritmo de Newton con incorporación de búsqueda lineal exacta.

El problema de calcular la raíz cuadrada de una matriz se puede encontrar a menudo en problemas de la ciencia y la ingeniería [17], por ejemplo: en la modelación estocástica de acuíferos, en la construcción de funciones de matrices, en la solución de ecuaciones diferenciales, en control automático, en robótica, entre otros. La teoría de existencia de la raíz cuadrada de una matriz y del número de raíces está bien documentada en [18, 19]. En este artículo se desarrolla una iteración alternativa simplificada del Método de Newton (IASMN) para calcular la raíz cuadrada de una matriz

real simétrica definida positiva que es computacionalmente económica, convergente y numéricamente estable.

2. El Método de Newton

Dada una matriz $B \in C^{n \times n}$, se dice que $X \in C^{n \times n}$ es una raíz cuadrada de B si cumple con la siguiente ecuación matricial cuadrática

$$F(X) \equiv X^2 - B = 0 \quad (1)$$

uno de los métodos numéricos utilizados para resolver (1) es el MN. En general, para resolver (1) donde F es una función definida de $F : C^{n \times n} \rightarrow C^{n \times n}$, el MN propone una solución inicial X_0 e iterativamente se va aproximando en forma cuadrática a la solución verdadera por medio del siguiente proceso iterativo [6, 7]

$$X_{k+1} = X_k - F'(X_k)^{-1}F(X_k) \quad (2)$$

donde F' es la derivada de Fréchet de F .

Para calcular la raíz cuadrada de B , el MN se define como

$$\begin{aligned} & \text{Dado } X_0 \\ & X_k H_k + H_k X_k = B - X_k^2 \\ & X_{k+1} = X_k + H_k \\ & \text{Para } k = 0, 1, 2, \dots \end{aligned} \quad (3)$$

siendo X_0 una matriz que suele coincidir con la matriz B o con la matriz identidad I_n . Las iteraciones (3) convergen en forma cuadrática a una raíz cuadrada de B siempre que $\|X - X_0\|$ sea lo suficientemente pequeño y la transformación lineal $F'(X)$ sea no singular [6].

3. El Método de Newton simplificado

Una simplificación a las iteraciones (3) es suponer que el producto $H_k X_k$ conmuta [4], es decir,

$$H_k X_k = X_k H_k \quad (4)$$

sustituyendo (4) en (3) se obtiene un par de iteraciones simplificadas del MN [8, 9, 10, 11, 12, 15]

$$X_{k+1} = \frac{1}{2} (X_k + X_k^{-1} B) \quad (5)$$

$$Y_{k+1} = \frac{1}{2} (Y_k + B Y_k^{-1}) \quad (6)$$

las iteraciones (5) y (6) en caso de converger, lo hacen en forma cuadrática hacia una raíz cuadrada de B . Higham [4] demostró que (5) es numéricamente inestable para números de condición de B mayores que nueve, haciéndolo extensivo para (6).

4. Iteración alternativa simplificada del Método de Newton (IASMN) para matrices reales simétricas definidas positivas

Sea $A \in R^{n \times n}$ una matriz real simétrica definida positiva, es decir, satisface que $A = A^T$ y $q^T A q > 0$ para todo vector $q \neq 0$ en R^n , además todos los valores propios de A son positivos [5].

La matriz real simétrica definida positiva $X \in R^{n \times n}$ es una raíz cuadrada de A si satisface la siguiente ecuación cuadrática matricial

$$F(X) = X^2 - A = 0 \quad (7)$$

por simetría de X se tiene que

$$X = X^T \quad (8)$$

$$X(X) = X^T(X) \quad (9)$$

sustituyendo (9) en (7)

$$F(X) = X^T X - A = 0 \quad (10)$$

construyamos las iteraciones del MN a partir de (10)

$$\lim_{\|H\| \rightarrow 0} [F(X+H) - F(X)] = X^T H + H^T X \quad (11)$$

de la siguiente expresión

$$\lim_{H \rightarrow 0} \frac{\|F(X+H) - F(X) - F'(X)H\|}{\|H\|} = 0$$

donde F' representa la derivada de Fréchet [14] de F , se tiene que

$$F'(X)H = F(X+H) - F(X) \quad (12)$$

la perturbación H_k se define por

$$H_k = X_{k+1} - X_k \quad (13)$$

combinando (13) con (2), (10), (11) y (12) se llega a las iteraciones del MN

$$\begin{aligned} \text{Dado } X_0 &= \alpha I_n \quad \alpha > 0^4 \\ X_k^T H_k + H_k^T X_k &= A - X_k^T X_k \\ X_{k+1} &= X_k + H_k \\ \text{Para } k &= 0, 1, 2, \dots \end{aligned} \quad (14)$$

En el Anexo A se demuestra que las iteraciones (14) convergen en forma cuadrática a la raíz cuadrada de A , además, en este mismo anexo se obtiene la IASMN junto con su factor de escala α_k , para calcular la raíz cuadrada X de una matriz A real simétrica definida positiva. La IASMN con factor de escala tiene la siguiente expresión.

$$\begin{aligned} \text{Dado } X_0 &= I_n \\ \alpha_k &= \frac{\sqrt{\text{traza}(A)}}{\|X_k\|_F} \quad \alpha_k > 0 \\ X_{k+1} &= \frac{1}{2} \left(\alpha_k X_k + (\alpha_k X_k)^T \setminus A \right) \\ k &= 0, 1, 2, \dots \end{aligned} \quad (15)$$

5. Experimentos Numéricos

A través de varios experimentos numéricos con matrices reales simétricas definidas positivas, se probó la exactitud y la eficacia computacional de la IASMN con factor de escala para calcular la raíz cuadrada de una matriz, con respecto a los siguientes métodos: método de Denman-Beavers [2] (DB), método basado en la descomposición polar [3] de la matriz A (PD), algoritmo de reducción cíclica [1] (CR), algoritmo de búsqueda lineal exacta [13] (ELS) y la función (sqrtm) de MATLAB. Para cada experimento realizado se reporta: el método utilizado, el número de iteraciones, el error relativo residual, el tiempo utilizado del CPU para el cálculo, el tamaño n y el número de condición $k_2(A)$ de la matriz A .

Experimento 1 (Matriz de Poisson). En este experimento se utilizó la función gallery de MATLAB, es una matriz tridiagonal por bloques con diagonales constantes, proviene de la discretización a cinco puntos del problema de Dirichlet homogéneo. Es una matriz bien condicionada, los resultados obtenidos se muestran en la Tabla 1. De acuerdo con los residuales todos los métodos proporcionan muy buena exactitud en el resultado. La IASMN es la más eficaz en lo que respecta al tiempo de ejecución del CPU.

Experimento 2 (Matriz de Lehmer). En este experimento se utilizó la función gallery de MATLAB, los resultados obtenidos se muestran en la Tabla 2. De acuerdo con los residuales, la mayoría de los métodos propor-

Tabla 1. Matriz de Poisson, $n = 2025$, $k_2(A) = 856,916$

Método	Iteraciones	Residual	CPU (seg.)
IASMN	8	3.92 E-014	131.35
DB	7	7.93 E-014	213.62
PD	7	7.93 E-014	146.53
CR	6	7.95 E-014	467.52
ELS	8	8.74 E-012	3396.44
sqrtm	-	5.29 E-014	147.35

cionan muy buena exactitud en el resultado. La exactitud de ELS puede considerarse medianamente aceptable. La IASMN es la más eficaz en lo que respecta al tiempo de ejecución del CPU.

Tabla 2. Matriz de Lehmer, $n = 2025$, $k_2(A) = 4,429E + 006$

Método	Iteraciones	Residual	CPU (seg.)
IASMN	8	5.60 E-014	131.25
DB	9	2.17 E-012	273.60
PD	10	1.83 E-015	166.06
CR	8	4.74 E-013	628.09
ELS	8	5.24 E-006	3385.27
sqrtm	-	2.50 E-014	151.28

Experimento 3 (Matriz de Minij). En este experimento se utilizó la función gallery de MATLAB, los resultados obtenidos se muestran en la Tabla 3. De acuerdo con sus residuales, los métodos IASMN, DB, PD y sqrtm proporcionan muy buena exactitud en el resultado. La exactitud de CR y ELS puede considerarse como aceptable y medianamente aceptable, respectivamente. La IASMN es la más eficaz en lo que respecta al tiempo de ejecución del CPU.

Tabla 3. Matriz de Minij, $n = 2025$, $k_2(A) = 6,651E + 006$

Método	Iteraciones	Residual	CPU (seg.)
IASMN	8	5.88 E-013	131.42
DB	15	1.13 E-012	469.30
PD	15	2.33 E-015	254.01
CR	14	4.73 E-008	1079.08
ELS	8	3.20 E-005	3400.28
sqrtm	-	4.01 E-015	146.62

Experimento 4 (Matriz de Moler, alpha=1). En este experimento se utilizó la función gallery de MATLAB, los resultados obtenidos se muestran en la Tabla 4. De acuerdo con sus residuales, los métodos IASMN, DB, PD y sqrtm proporcionan muy buena exactitud en el resultado. La exactitud de CR y ELS puede considerarse como aceptable y medianamente aceptable, respectivamente.

vamente. La IASMN es la más eficaz en lo que respecta al tiempo de ejecución del CPU.

Tabla 4. Matriz de Moler, $n = 2025$, $k_2(A) = 6,651E + 006$

Método	Iteraciones	Residual	CPU (seg.)
IASMN	8	5.88 E-013	132.05
DB	15	1.14 E-012	475.36
PD	15	2.34 E-015	252.41
CR	15	4.73 E-008	1084.52
ELS	8	3.20 E-005	3377.28
sqrtm	-	4.01 E-015	147.05

Experimento 5 (Matriz de Lehmer). En este experimento se utilizó la función gallery de MATLAB, por el tamaño de la matriz se considera pequeña, los resultados obtenidos se muestran en la Tabla 5. De acuerdo con los residuales, la mayoría de los métodos proporcionan muy buena exactitud en el resultado. La exactitud de ELS puede considerarse suficientemente aceptable. Los métodos IASMN y PD son los más eficaces en lo que respecta al tiempo de ejecución del CPU.

Tabla 5. Matriz de Lehmer, $n = 100$, $k_2(A) = 1,027E + 004$

Método	Iteraciones	Residual	CPU (seg.)
IASMN	7	2.38 E-015	0.047
DB	8	3.85 E-015	0.075
PD	8	5.67 E-016	0.047
CR	7	4.60 E-015	0.105
ELS	10	4.07 E-011	1.081
sqrtm	-	1.06 E-015	0.087

6. Conclusiones

Con base en los resultados experimentales obtenidos en esta investigación, se comprobó que la IASMN con factor de escala desarrollada en este artículo, es atractiva por la exactitud de su estimación de la raíz cuadrada de una matriz real simétrica definida positiva bien condicionada esparza o densa sin importar su tamaño y, por su eficacia en el tiempo de ejecución del CPU. Además, de su fácil codificación, evaluación y por sus excelentes propiedades teóricas de convergencia y estabilidad numérica.

Agradecimientos

Quiero expresar mi más sincero agradecimiento y gratitud al Consejo Nacional de Ciencia y Tecnología (CONACYT), al Centro de Investigación Científica y

de Educación Superior de Ensenada (CICESE) y a la Universidad de Sonora (UNISON), por el apoyo recibido durante mi estancia en el CICESE y durante la elaboración de este trabajo.

Referencias

1. Meini B. (2004) The matrix square root from a new functional perspective: Theoretical results and computational issues. *SIAM J. Matrix Anal. Appl.* 26:362-376
2. Denman E.D., Beavers A.N. (1976) The matrix sign function and computations in systems. *Appl. Math. Comput.* 2:63-94
3. Higham N.J. (1986) Computing the polar decomposition-with applications. *SIAM J. Sci. Statist. Comput.*, 71160-1174
4. Higham N.J. (1986) Newton's method for the matrix square root. *Math. Comp.* 46:537-549
5. Kolman B. (2006) *Algebra lineal*. Octava Edición, Pearson, México
6. Ortega J.M. (1972) *Numerical analysis: A second course*. Academic Press, New York
7. Smith W.A. (1988) *Análisis numérico*. Prentice-Hall, México
8. Björck Å., Hammarling S. (1983) A Schur method for the square root of a matrix. *Linear Algebra Appl.* 52:127-140
9. Golub G.H., Van Loan C.F. (1983) *Matrix computations*. Johns Hopkins Univ. Press. Baltimore, Maryland
10. Hoskins W.D., Walton D.J. (1978) A faster method of computing the square root of a matrix. *IEEE Trans. Automat. Control* AC-23:494-495
11. Hoskins W.D., Walton D.J. (1979) A faster, more stable method for computing the pth roots of positive definite matrices. *Linear Algebra Appl.* 26:139-163
12. Laasonen P. (1958) On the iterative solution of the matrix equation $AX^2 - I = 0$. *M.T.A.C.* 12:109-116
13. Long J.H., Hu X.Y., Zhang L. (2008) Newton's method with exact line search for the square root of a matrix. *J. Phys., Conf. Ser.* 96:012034
14. Selberherr S. (1984) *Analysis and simulation of semiconductor devices*. Wien, Springer-Verlag, New York
15. Schatzman M. (2002) *Numerical analysis: A mathematical introduction*. Clarendon Press/Oxford University Press, Oxford/New York
16. Meyer C.D. (2000) *Matrix analysis and applied linear algebra*. SIAM
17. He J.H. (2003) A new iteration method for solving algebraic equations. *Appl. Math. Comput.* 135:81-84
18. Horn R.A., Johnson C.R. (1991) *Topics in matrix analysis*. Cambridge University Press, Cambridge
19. Higham N.J. (1987) Computing real square roots of a real matrix. *Linear algebra and Appl.* 88:405-430
20. He J.H. (2000) Improvement of Newton iteration method. *International Journal of Nonlinear Sciences and Numerical Simulation* 1:239-240
21. Higham N.J. (1997). Stable iterations for the matrix square root. *Numerical Algorithms* 15:227-242
22. Henrici P., (1964) *Elements of Numerical Analysis*, Wiley. New York

Anexo A

Análisis de convergencia de las iteraciones (14)

Teorema 1. Sea $A \in R^{n \times n}$ una matriz real simétrica definida positiva, si $\|X - X_0\|$ es lo suficientemente pequeño y la transformación lineal $F'(X)$ es no singular entonces las iteraciones $\{X_k\}$ proporcionadas por (14) convergen a X cuando $k \rightarrow \infty$.

Demostración:

de acuerdo con (13) las iteraciones (14) se representan por

$$X_k^T (X_{k+1} - X_k) + (X_{k+1} - X_k)^T X_k = A - X_k^T X_k \quad (A1)$$

desarrollando (A1) y simplificando

$$X_k^T X_{k+1} + X_{k+1}^T X_k = A + X_k^T X_k = X_k^T X_k + A \quad (A2)$$

para $k = 0$, la solución inicial en (A2) es $X_0 = \alpha I_n$ con $\alpha > 0$, entonces

$$X_0^T X_1 + X_1^T X_0 = X_0^T X_0 + A \quad (A3)$$

donde X_0 y $(X_0^T X_0 + A)$ son matrices reales simétricas definidas positivas, entonces los productos $X_0^T X_1$ y $X_1^T X_0$ también representan matrices reales definidas positivas, por consiguiente X_1 y X_1^T son también matrices reales definidas positivas.

Por las características de A , existe una matriz Q no singular tal que

$$Q^{-1} A Q = \Sigma = \text{diag}(\lambda_1, \dots, \lambda_n) \quad (A4)$$

donde $\lambda_1, \dots, \lambda_n$ son los valores propios de la matriz A . También se tiene que

$$Q^{-1} X_1 Q = D_1 = \text{diag}(d_1^k, \dots, d_n^k) \quad (A5)$$

$$Q^{-1} X_1^T Q = D_1^T = D_1 \quad (A6)$$

$$Q^{-1} X_0 Q = D_0 = D_0^T \quad (A7)$$

diagonalizando y simplificando (A3) se tiene

$$D_1 = \frac{1}{2} (D_0 + D_0^{-T} \Sigma) \quad (A8)$$

como puede verse D_1 es una matriz diagonal, utilizando el proceso inverso veremos que

$$\begin{aligned} X_1 &= Q D_1 Q^{-1} = \frac{1}{2} \left(Q D_0 Q^{-1} + (Q D_0 Q^{-1})^{-T} Q \Sigma Q^{-1} \right) \\ &= \frac{1}{2} (X_0 + X_0^{-T} A) \end{aligned}$$

es fácil comprobar que X_1 , además de ser una matriz real definida positiva, también es simétrica.

De nuevo en (A2) hacemos $k = 1$ y, utilizando el mismo procedimiento para $k = 0$ se encuentra que D_2 también será diagonal y, que X_2 , es una matriz real simétrica definida positiva, y así sucesivamente, por lo tanto, generalizando se tiene

$$D_{k+1} = \frac{1}{2} (D_k + D_k^{-T} \Sigma) \quad (A9)$$

la ecuación (A9) representa esencialmente n iteraciones escalares desacopladas de Newton para las raíces cuadradas $\sqrt{\lambda_i}$, $1 \leq i \leq n$, o sea

$$d_i^{k+1} = \frac{1}{2} \left(d_i^k + \frac{\lambda_i}{d_i^k} \right), \quad 1 \leq i \leq n \quad (A10)$$

utilizando la siguiente iteración escalar [22]

$$z_{k+1} = \frac{1}{2} \left(z_k + \frac{a}{z_k} \right) \quad (A11)$$

$$z_{k+1} \pm \sqrt{a} = \frac{(z_k \pm \sqrt{a})^2}{2z_k} \quad (A12)$$

$$\frac{z_{k+1} - \sqrt{a}}{z_{k+1} + \sqrt{a}} = \left(\frac{z_0 - \sqrt{a}}{z_0 + \sqrt{a}} \right)^{2^{k+1}} \equiv \gamma^{2^{k+1}} \quad (A13)$$

para z_0 y a positivos se tiene que $|\gamma| < 1$, entonces:

$$\lim_{k \rightarrow \infty} z_k = \sqrt{a} \quad (A14)$$

debido a que los valores propios λ_i y los valores de la solución inicial $d_i^{(0)} = \alpha > 0$ tienen la misma forma de a y de z_0 respectivamente, entonces se cumple que,

$$\lim_{k \rightarrow \infty} D_k = \sqrt{\Sigma} = \text{diag} \left(\lambda_i^{\frac{1}{2}} \right) \quad (A15)$$

por lo tanto,

$$\lim_{k \rightarrow \infty} Q D_k Q^{-1} = \lim_{k \rightarrow \infty} X_k = Q \sqrt{\Sigma} Q^{-1} = \sqrt{A} = X. \quad (A16)$$

La IASMN, se obtiene a partir de (A9) de la siguiente manera

$$Q D_{k+1} Q^{-1} = \frac{1}{2} (Q D_k Q^{-1} + Q D_k^{-T} Q^{-1} Q \Sigma Q^{-1}) \quad (A17)$$

$$X_{k+1} = \frac{1}{2} (X_k + X_k^{-T} A) \quad (A18)$$

utilizando en (A18) la eliminación gaussiana (\setminus) en lugar de la inversa por ser más eficaz en cuanto al tiempo de ejecución y exactitud numérica, se llega a la IASMN para calcular la raíz cuadrada X de una matriz A real simétrica definida positiva, la cual tiene la siguiente expresión.

$$\begin{aligned} \text{Dado } X_0 &= \alpha I_n, \quad \alpha > 0 \\ X_{k+1} &= \frac{1}{2} (X_k + X_k^T \setminus A) \quad (A19) \\ \text{Para } k &= 0, 1, 2, \dots \end{aligned}$$

Corolario. Sea $A \in R^{n \times n}$ una matriz real simétrica definida positiva, si la solución inicial en (A19) es de la forma $X_0 = \alpha I_n$ con $\alpha > 0$, entonces todas las iteraciones $\{X_k\}$ proporcionadas por (A19) son reales simétricas definidas positivas.

Análisis de estabilidad numérica de la IASMN

Por conveniencia en la IASMN se utilizará la inversa en lugar de la eliminación gaussiana, es decir,

$$X_{k+1} = \frac{1}{2} \left(X_k + (X_k^T)^{-1} A \right) \quad (\text{A20})$$

si \hat{X}_k representa la k -ésima iteración perturbada con la matriz de perturbaciones Δ_k entonces [4]

$$\hat{X}_k \approx X_k$$

$$\Delta_k = \hat{X}_k - X_k \quad (\text{A21})$$

perturbaremos (A20) con (A21), para analizar la propagación de estas perturbaciones en la $(k+1)$ -ésima iteración perturbada representada por la matriz \hat{X}_{k+1} . Considerando que no se introducen nuevos errores de redondeo en el cálculo de \hat{X}_{k+1} se tiene

$$\begin{aligned} \hat{X}_{k+1} &= \frac{1}{2} \left(\hat{X}_k + (\hat{X}_k^T)^{-1} A \right) = \\ &= \frac{1}{2} \left(X_k + \Delta_k + [(X_k + \Delta_k)^T]^{-1} A \right) \end{aligned} \quad (\text{A22})$$

utilizando una aproximación de primer orden en series de Taylor para $(X_k^T + \Delta_k^T)^{-1}$ y considerando que $\|\Delta_k^T\|$ es una cantidad muy pequeña [17]

$$\begin{aligned} (X_k^T + \Delta_k^T)^{-1} &= (X_k^T)^{-1} - (X_k^T)^{-1} \Delta_k^T (X_k^T)^{-1} + \\ &+ O\left(\|\Delta_k^T\|^2\right) \end{aligned} \quad (\text{A23})$$

sustituyendo (A23) en (A22) y desarrollando

$$\begin{aligned} \hat{X}_{k+1} &= \\ &= \frac{1}{2} \left(X_k + \Delta_k + (X_k^T)^{-1} A - (X_k^T)^{-1} \Delta_k^T (X_k^T)^{-1} A \right) + \\ &+ O\left(\|\Delta_k^T\|^2\right) \end{aligned} \quad (\text{A24})$$

restando (A20) a (A24)

$$\Delta_{k+1} = \frac{1}{2} \left(\Delta_k - (X_k^T)^{-1} \Delta_k^T (X_k^T)^{-1} A \right) + O\left(\|\Delta_k^T\|^2\right) \quad (\text{A25})$$

utilizando la notación (A4) para diagonalizar (A25)

$$\tilde{\Delta}_{k+1} = \frac{1}{2} \left(\tilde{\Delta}_k - (D_k^T)^{-1} \tilde{\Delta}_k^T (D_k^T)^{-1} \Sigma \right) + O\left(\|\tilde{\Delta}_k^T\|^2\right) \quad (\text{A26})$$

$$\begin{aligned} D_k^T &= D_k = \text{diag}(d_i^k) \\ \tilde{\Delta}_k &= \delta_{ij}^k \\ \tilde{\Delta}_k^T &= \delta_{ji}^k \end{aligned} \quad (\text{A27})$$

expresando a (A26) como n iteraciones desacopladas

$$\delta_{ij}^{k+1} = \frac{1}{2} \left(\delta_{ij}^k - \frac{\delta_{ji}^k}{d_j^k d_i^k} \lambda_i \right) + O\left(\|\tilde{\Delta}_k^T\|^2\right) \quad 1 \leq i, j \leq n. \quad (\text{A28})$$

Si la matriz A está bien condicionada y el número de iteraciones k tiende a infinito entonces la matriz de perturbaciones Δ_k conserva su simetría, es decir, $\Delta_k^T = \Delta_k$ y como consecuencia

$$\delta_{ji}^k = \delta_{ij}^k \quad (\text{A29})$$

rescribiendo (A28)

$$\delta_{ij}^{k+1} = \frac{1}{2} \left(\delta_{ij}^k - \frac{\delta_{ij}^k}{d_j^k d_i^k} \lambda_i \right) + O\left(\|\tilde{\Delta}_k\|^2\right) \quad (\text{A30})$$

factorizando

$$\delta_{ij}^{k+1} = \frac{1}{2} \delta_{ij}^k \left(1 - \frac{\lambda_i}{d_j^k d_i^k} \right) + O\left(\|\tilde{\Delta}_k\|^2\right) \quad (\text{A31})$$

si $k \rightarrow \infty$ entonces se tiene que $d_j^k = \lambda_j^{1/2} + \varepsilon_j^k$, $d_i^k = \lambda_i^{1/2} + \varepsilon_i^k$, con $\varepsilon_i^k, \varepsilon_j^k \rightarrow 0$ rescribiendo (A31)

$$\delta_{ij}^{k+1} = \frac{1}{2} \delta_{ij}^k \left(1 - \frac{\lambda_i^{1/2}}{\lambda_j^{1/2}} \right) + O\left(\|\Delta_k\|^2\right) \quad 1 \leq i, j \leq n \quad (\text{A32})$$

para asegurar la estabilidad numérica de (A20), y por ende la de la IASMN se debe de cumplir que

$$\frac{1}{2} \left| 1 - \frac{\lambda_i^{1/2}}{\lambda_j^{1/2}} \right| \leq 1 \quad (\text{A33})$$

el número de condición [4] de A es $k_2(A) = \|A\|_2 \|A^{-1}\|_2 = \frac{\lambda_j}{\lambda_i}$ $1 \leq i, j \leq n$, por lo tanto, $\frac{\lambda_i}{\lambda_j} = k_2(A)^{-1}$ rescribiendo (A33)

$$\frac{1}{2} \left| 1 - k_2(A)^{-\frac{1}{2}} \right| \leq 1 \quad (\text{A34})$$

resolviendo (A34) para $k_2(A)$ se obtienen dos resultados $k_2(A) \geq 1$ (A35)

$$k_2(A) \geq \frac{1}{9} \quad (\text{A36})$$

Dado que la matriz A es real simétrica definida positiva, el mínimo valor para el número de condición que se espera es la unidad, por lo tanto, el resultado correcto para este caso es (A35). Con esto se demuestra que la IASMN es, para propósitos prácticos, numéricamente estable para números de condición mayores que la unidad, siempre y cuando este número de condición no represente un mal condicionamiento.

Factor de la escala IASMN

Con la finalidad de disminuir el tiempo de convergencia de la IASMN, cada una de las iteraciones X_k será multiplicada por un factor de escala $\alpha_k > 0$, de tal manera que se minimice

$$\|A - \alpha_k^2 X_k^T X_k\|_F^2 \quad \text{para } k = 0, 1, 2, \dots \quad (\text{A37})$$

el valor de α_k que minimiza (A37) está dado por

$$\alpha_k = \frac{\sqrt{\text{traza}(A)}}{\|X_k\|_F} \quad \alpha_k > 0 \quad \text{para } k = 0, 1, 2, \dots \quad (\text{A38})$$

el costo computacional de calcular el factor de escala α_k puede considerarse despreciable.

Entonces la IASMN con factor de escala tiene la siguiente expresión

$$\begin{aligned} \text{Dado } X_0 &= I_n \\ \alpha_k &= \frac{\sqrt{\text{traza}(A)}}{\|X_k\|_F} \quad \alpha_k > 0 \\ X_{k+1} &= \frac{1}{2} \left(\alpha_k X_k + (\alpha_k X_k)^T \setminus A \right) \\ k &= 0, 1, 2, \dots \end{aligned} \quad (\text{A39})$$

debido a que la IASMN es convergente, entonces el

$$\lim_{k \rightarrow \infty} \alpha_k = 1$$