International Electronic Journal of Pure and Applied Mathematics

Volume 7 No. 2 2014, 53-62
ISSN: 1314-0744
url: http://www.e.ijpam.eu
doi: http://dx.doi.org/10.12732/iejpam.v7i2.1

DIVIDE-AND-CONQUER STRATEGY WITH CHOLESKY'S FACTORIZATION FOR INVERTING SYMMETRIC POSITIVE DEFINITE MATRICES

Awni M. Abu-Saman[1] §, Shady Sayed El-Okur[2]

Applied Mathematics
Alazhar University-Gaza (AUG)
Gaza, PALESTINE

Al-azhar University-Gaza(AUG),
Gaza, PALESTINE

Abstract: In this paper we study the solution of a system of equations **Ax=b** with singular and nearly singular, symmetric positive definite coefficient matrix **A**. Our algorithm based on, the Divide and Conquer strategy leading to the Divide-and-Conquer Algorithm (D&C algorithm) with, Cholesky's factorization algorithm. The Cholesky's factorization will be used to convert the matrix into a product of the form $\mathbf{LL^T}$, where **L** is a lower triangular matrix. The algorithm will be implemented on MATLAB and simulated as a user-subroutine. The user-subroutine is considering MATLAB features for reducing the round-off error especially for sensitive systems. Numerical examples will be given of a non- singular matrix and another for ill-conditioned matrix. The effect of round-off error will be analyzed. Results will be compared with previous ones, where LU factorization is used.

AMS Subject Classification: 15A09
Key Words: non-singular matrix, Ill-conditioned system, Cholesky's method, divides-and conquer algorithm, mathematical software

1. Introduction

The problem of matrix inversion is considered to be one of the basic problems widely

Received: February 21, 2013

§Correspondence author

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 7, No. 2 (2014)

encountered in science and engineering fields. It is usually an essential part of many solutions, e.g., as preliminary steps for optimization, single processing, electromagnetic systems, robotic control, statistics and physics.

Some effective direct solution algorithms exploiting displacement representation can be found in[2, 3]. Alternative iterative methods were proposed in [1]. The later methods non-trivially extend some preceding work for general input matrices [3] and can be most effective for well conditioned inputs.

Solving computational mathematical problems through using partial differential equations (PDE), integral equations, and boundary value problems in ordinary differential equations (ODE), normally produce system of linear equations.

Cholesky's method is a method for solving a linear system of equations $\mathbf{Ax=b}$ for special case when $\mathbf{A}$ is a symmetric positive definite matrix.[4, 5, 6, 10]. That is when $\mathbf{A^T} = \mathbf{A}$and all the eigenvalues of $\mathbf{A}$ are positive. Such systems occur in a very wide range of important applications usually associated with a 'minimum energy principle'. Cholesky's method relies on the following result. Any symmetric positive definite matrix $\mathbf{A}$ can be factorized into $\mathbf{A} = \mathbf{LL^T}$, where $\mathbf{L}$ is a lower triangular matrix. All computational results in this paper are performed in MATLAB.[9]. using different computational digits.

The relative error is computed using the formula:

$$\mathbf{res_{inv}} = \mathbf{max}[\| \mathbf{I\text{-}AA^{-1}} \|, \| \mathbf{I\text{-}A^{-1}A} \|]/ \| \mathbf{A} \|$$

In this paper the inverse of the matrix A will be computed by factorizing the matrix into a product of lower and upper triangular matrices, and then we apply the divide and conquer algorithm. The main aim of this work is to find the inverse of nearly singular positive definite matrices using Divide and Conquer Algorithm where the computational decimal digits will be increased to overcome the sensitivity of the system.

In Section 2 we present an overview of the Cholesky Factorization and Divide and Conquer Algorithms for finding the inverse of nearly singular matrices.

In Section 3, different examples of non-singular and nearly singular systems will be given and solved for increasing number of digits in the calculations; calculations will be simulated using our own subroutines in the mathematical code MATLAB. Concluding remarks and comments will be is in Section four.

2. Divide and Conquer Algorithm

Divide and conquer (D&C)is an important algorithm design paradigm. It works by recursively breaking down a problem into two or more sub-problems of the same type, until these become simple enough to be solved directly. The solutions to the sub-problems are then combined to give a solution to the original problem. [5, 10, 7].

Three Divide and Conquer approach summarized in the following steps:

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 7, No. 2 (2014)

1. Divide the problem into two or smaller sub-problem.
2. Conquer the sub-problem by solving them recursively.
3. Combine the solutions to the sub-problems into the solutions for the original problem.

Consider the nxn upper triangular matrix $\mathbf{L}$

$$\begin{bmatrix} l_{11} & 0 & 0 & 0 & 0 \\ l_{21} & l_{22} & 0 & 0 & 0 \\ l_{31} & l_{32} & l_{33} & 0 & 0 \\ \dots & \dots & \dots & \dots & 0 \\ l_{n1} & l_{n2} & l_{n3} & \dots & l_{nn} \end{bmatrix}$$

Partition $\mathbf{L}$ into $(n/2)x(n/2)$ blocks. The inverse of an upper or lower triangular matrix is itself upper or lower triangular.

If $\mathbf{L}$ is lower triangular matrix partitioned as

$$\mathbf{L} = \begin{bmatrix} \mathbf{L}_{11} & \\ \mathbf{L}_{21} & \mathbf{L}_{22} \end{bmatrix}$$

Then let $\mathbf{B}$ be a matrix of equal dimensions and partitioned as $\mathbf{L}$ as

$$\mathbf{B} = \begin{bmatrix} \mathbf{B}_{11} & \\ \mathbf{B}_{21} & \mathbf{B}_{22} \end{bmatrix}$$

The equation $\mathbf{LB} = \mathbf{I}$, If $\mathbf{L}$ is invertible and then $\mathbf{L}^{-1} = \mathbf{B}$

$\mathbf{L}_{11}\mathbf{B}_{11} = \mathbf{I}$ implies $\mathbf{B}_{11} = \mathbf{L}_{11}^{-1}$, and $\mathbf{L}_{22}\mathbf{B}_{22} = \mathbf{I}$

Implies $\mathbf{B}_{22} = \mathbf{L}_{22}^{-1}$,

$\mathbf{L}_{21}\mathbf{B}_{11} + \mathbf{L}_{22}\mathbf{B}_{21} = 0$, implies that $\mathbf{B}_{21} = -\mathbf{L}_{22}^{-1}\mathbf{L}_{21}\mathbf{L}_{11}^{-1}$

Therefore the inverse of the lower triangular matrix $\mathbf{L}$ is given by

$$\mathbf{L}^{-1} = \begin{bmatrix} \mathbf{L}_{11}^{-1} & \\ -\mathbf{L}_{22}^{-1}\mathbf{L}_{21}\mathbf{L}_{11}^{-1} & \mathbf{L}_{22}^{-1} \end{bmatrix}$$

Consider nxn upper triangular matrix $\mathbf{U}$

$$\begin{bmatrix} u_{11} & u_{12} & u_{13} & \dots & u_{1n} \\ 0 & u_{22} & u_{23} & \dots & u_{2n} \\ 0 & 0 & u_{33} & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & u_{nn} \end{bmatrix}$$

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 7, No. 2 (2014)

If $\mathbf{U}$ is upper triangular matrix partitioned as

$$\mathbf{U} = \begin{bmatrix} \mathbf{U}_{11} & \mathbf{U}_{12} \\ & \mathbf{U}_{22} \end{bmatrix}$$

Then let $\mathbf{B}$ be a matrix of equal dimensions and partitioned as $\mathbf{U}$ as

$$\mathbf{B} = \begin{bmatrix} \mathbf{B}_{11} & \mathbf{B}_{12} \\ & \mathbf{B}_{22} \end{bmatrix}$$

The equation $\mathbf{UB} = \mathbf{I}$, If $\mathbf{U}$ is invertible and then $\mathbf{U}^{-1} = \mathbf{B}$
$\mathbf{U}_{11}\mathbf{B}_{11} = \mathbf{I}$ implies $\mathbf{B}_{11} = \mathbf{U}_{11}^{-1}$, and $\mathbf{U}_{22}\mathbf{B}_{22} = \mathbf{I}$, Implies $\mathbf{B}_{22} = \mathbf{U}_{22}^{-1}$,
$\mathbf{U}_{11}\mathbf{B}_{12} + \mathbf{U}_{12}\mathbf{B}_{22} = 0$, implies that $\mathbf{B}_{12} = -\mathbf{U}_{11}^{-1}\mathbf{U}_{12}\mathbf{U}_{22}^{-1}$
Therefore the inverse of the upper triangular matrix $\mathbf{U}$ is given by

$$\mathbf{U}^{-1} = \begin{bmatrix} \mathbf{U}_{11}^{-1} & -\mathbf{U}_{11}^{-1}\mathbf{U}_{12}\mathbf{U}_{22}^{-1} \\ & \mathbf{U}_{22}^{-1} \end{bmatrix}$$

Algorithm of the Divide and Conquer user subroutine in MATLAB for computing $\mathbf{A}^{-1}$:

1. Consider a n x n square matrix,
2. Use the Cholesky algorithm to factorize the matrix into a product of upper and lower triangular matrices $\mathbf{A}=\mathbf{L}\mathbf{L}^T$, where $\mathbf{L}$ is Cholesky factor of $\mathbf{A}$ and $\mathbf{L}^T$ is the transpose of $\mathbf{L}$. let $\mathbf{L}^T = \mathbf{U}$
3. $\mathbf{A}^{-1} = (\mathbf{L}\mathbf{U})^{-1} = \mathbf{U}^{-1}\mathbf{L}^{-1}$

Computation of $\mathbf{U}^{-1}$

1. Divide $\mathbf{U}$ into four blocks $\mathbf{U}_{11}, \mathbf{U}_{12}, \mathbf{U}_{22}$ and a zero matrix
2. Recursively compute inverses of $\mathbf{U}_{11}$ and $\mathbf{U}_{22}$
3. Multiply $-\mathbf{U}_{11}^{-1}\mathbf{U}_{12}\mathbf{U}_{22}^{-1}$ and combine with $\mathbf{U}_{11}^{-1}$ and $\mathbf{U}_{22}^{-1}$ to get $\mathbf{U}^{-1}$
4. $\mathbf{U}^{-1} = \begin{bmatrix} \mathbf{U}_{11}^{-1} & -\mathbf{U}_{11}^{-1}\mathbf{U}_{12}\mathbf{U}_{22}^{-1} \\ 0 & \mathbf{U}_{22}^{-1} \end{bmatrix}$

Computation of $\mathbf{L}^{-1}$

1. Divide $\mathbf{L}$ into four blocks $\mathbf{L}_{11}, \mathbf{L}_{12}, \mathbf{L}_{22}$ and a zero matrix
2. Recursively compute inverses of $\mathbf{L}_{11}$ and $\mathbf{L}_{22}$.
3. Multiply $-\mathbf{L}_{11}^{-1}\mathbf{L}_{21}\mathbf{L}_{22}^{-1}$ and combine with $\mathbf{L}_{11}^{-1}$ and $\mathbf{L}_{22}^{-1}$ to get $\mathbf{L}^{-1}$
4. $\mathbf{L}^{-1} = \begin{bmatrix} \mathbf{L}_{11}^{-1} & 0 \\ -\mathbf{L}_{11}^{-1}\mathbf{L}_{21}\mathbf{L}_{22}^{-1} & \mathbf{L}_{22}^{-1} \end{bmatrix}$

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 7, No. 2 (2014)

3. Numerical Examples and Results

The matrix **A** is factorized by **Cholesky-factorization** into a product of upper and lower triangular matrices $\mathbf{A}=\mathbf{L}\mathbf{L}^T$, where **L** is Cholesky factor of **A** and $\mathbf{L}^T$ is the transpose of **L**. let $\mathbf{L}^T = \mathbf{U}$ are computed by Divide and Conquer MATLAB user sub routines for different number of computational digits.

Example 1. Consider the non-singular 8x8 symmetric positive definite matrix.

$$A = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\ 1 & 3 & 6 & 10 & 15 & 21 & 28 & 36 \\ 1 & 4 & 10 & 20 & 35 & 56 & 84 & 120 \\ 1 & 5 & 15 & 35 & 70 & 126 & 210 & 330 \\ 1 & 6 & 21 & 56 & 126 & 252 & 462 & 792 \\ 1 & 7 & 28 & 84 & 210 & 462 & 924 & 1716 \\ 1 & 8 & 36 & 120 & 330 & 792 & 1716 & 3432 \end{bmatrix}$$

First we find the Cholesky factorization of the matrix **A** by Matlab user-subroutine.

Where **L** is called the Cholesky factor of **A**, and $\mathbf{L}^T$ it is the transpose of **L**,then the matrix A can be written as $A = L\,L^T$.

$$A = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 2 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 3 & 3 & 1 & 0 & 0 & 0 & 0 \\ 1 & 4 & 6 & 4 & 1 & 0 & 0 & 0 \\ 1 & 5 & 10 & 10 & 5 & 1 & 0 & 0 \\ 1 & 6 & 15 & 20 & 15 & 6 & 1 & 0 \\ 1 & 7 & 21 & 35 & 35 & 21 & 7 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ 0 & 0 & 1 & 3 & 6 & 10 & 15 & 21 \\ 0 & 0 & 0 & 1 & 4 & 10 & 20 & 35 \\ 0 & 0 & 0 & 0 & 1 & 5 & 15 & 35 \\ 0 & 0 & 0 & 0 & 0 & 1 & 6 & 21 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 7 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} = LL^T$$

The inverse of the upper triangular matrix ($\mathbf{L}^T$)is computed by divide and conquer method, let $\mathbf{U} = \mathbf{L}^T$

$$U^{-1} = \begin{bmatrix} 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 0 & 1 & -2 & 3 & -4 & 5 & -6 & 7 \\ 0 & 0 & 1 & -3 & 6 & -10 & 15 & -21 \\ 0 & 0 & 0 & 1 & -4 & 10 & -20 & 35 \\ 0 & 0 & 0 & 0 & 1 & -5 & 15 & -35 \\ 0 & 0 & 0 & 0 & 0 & 1 & -6 & 21 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -7 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 7, No. 2 (2014)

The inverse of the lower triangular matrix $\mathbf{L}^{-1}$ is also computed by Divide and conquer method

$$L^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -2 & 1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 3 & -3 & 1 & 0 & 0 & 0 & 0 \\ 1 & -4 & 6 & -4 & 1 & 0 & 0 & 0 \\ -1 & 5 & -10 & 10 & -5 & 1 & 0 & 0 \\ 1 & -6 & 15 & -20 & 15 & -6 & 1 & 0 \\ -1 & 7 & -21 & 35 & -35 & 21 & -7 & 1 \end{bmatrix}$$

Since $A = LL^T = LU$, where $L^T = U$, then product both side by $(LU)^{-1}$ we have $(LU)^{-1}A = (LU)^{-1}(LU)$
Hence $(U^{-1}L^{-1})A = I$, there for $A^{-1} = U^{-1}L^{-1}$. Then

$$A^{-1} = \begin{bmatrix} 8 & -28 & 56 & -70 & 56 & -28 & 8 & -1 \\ -28 & 140 & -322 & 434 & -364 & 188 & -55 & 7 \\ 65 & -322 & 812 & -1162 & 1016 & -541 & 162 & -21 \\ -70 & 434 & -1162 & 1742 & -1579 & 865 & -265 & 35 \\ 56 & -364 & 1016 & -1579 & 1476 & -830 & 260 & -35 \\ -28 & 188 & -541 & 865 & -830 & 478 & -153 & 21 \\ 8 & -55 & 162 & -265 & 260 & -153 & 50 & -7 \\ -1 & 7 & -21 & 35 & -35 & 21 & -7 & 1 \end{bmatrix}$$

Example 2. Consider the Nearly-Singular 6x6 symmetric positive definite matrix.

$$\mathbf{A} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 2 & 3 & 4 & 5 & 6 \\ 1 & 3 & 6 & 10 & 15 & 21 \\ 1 & 4 & 10 & 20 & 35 & 56 \\ 1 & 5 & 15 & 35 & 70 & 126 \\ 1 & 6 & 21 & 56 & 126 & 251.00000000001 \end{bmatrix}$$

First we factorize the matrix $\mathbf{A}$ by Cholesky-factorization $A = L\,L^T$. Where $\mathbf{L}$ is called the Cholesky factor of A, and $\mathbf{L}^T$ it is the transpose of $\mathbf{L}$, the inverses of $\mathbf{L}$ and $\mathbf{L}^T$ are computed by Divide and Conquer MATLAB user sub routine for different number of Computational digits. The above matrix is a nearly-singular matrix with $Cond(A) = 8.4920e+015$ and determinant $= 1.0079e-011$. The (D&C algorithm) is implemented using singe precision calculations. The results obtained

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 7, No. 2 (2014)

are shown in Table 3.1 and Figure 1, it's clear that computational results of the (D&C) user subroutine in this example are of less residual relative error than of LU factorization for some digits. When considering more calculation digits the subroutine with LU factorization gives less residual relative error than of the subroutine Cholesky. Results also show that increasing the number of the computational digits significantly reduces the relative residual error ($\mathbf{res}_{inv}$) in two cases.

Cholesky-factorization		*LU-factorization*
Digits Number	res_{inv}	res_{inv}
10 digits	4.8296e-005	7.9663e-005
13 digits	2.3439e-005	8.6889e-005
16 digits	2.9486e-012	1.8811e-011
17 digits	3.0246e-013	8.7485e-013
18 digits	7.1465e-014	4.3818e-014
19 digits	5.0584e-015	6.9197e-015
20 digits	2.3125e-016	9.4608e-016
21 digits	3.4162e-017	6.3241e-017
22 digits	8.4832e-018	7.7341e-018
23 digits	1.0668e-018	7.4899e-019
24 digits	1.1212e-018	7.2925e-020

Table 3.1: Computations of the residual relative error for different computational digits by Divide and Conquer subroutine for the Nearly-Singular 6x6 symmetric positive definite matrix.

4. Concluding Remarks

We first recalled built-in MATLAB functions for the inversion of the symmetric positive definite matrices and then presented our own alternative MATLAB user-subroutine based on Divide-and-Conquer strategy with Cholesky factorization, There are several challenges to overcome to fully exploit the available capabilities of our algorithm. First, the number of digits considered in the calculations are increased to reduce the round-off error. Second, the matrices are scaled before the factorization. Third, the number of operations is reduced. The numerical calculations show that for well conditioned matrices the Divide-and-Conquer scheme results in obtaining almost same accuracy results. When the matrix is ill-conditioned, our calculations show the advantages of our MATLAB user-subroutines in the manipulation of sensitive systems. Results of our User-subroutine, based on Cholesky factorization, show the advantages of the algorithm over previous calculations with

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 7, No. 2 (2014)

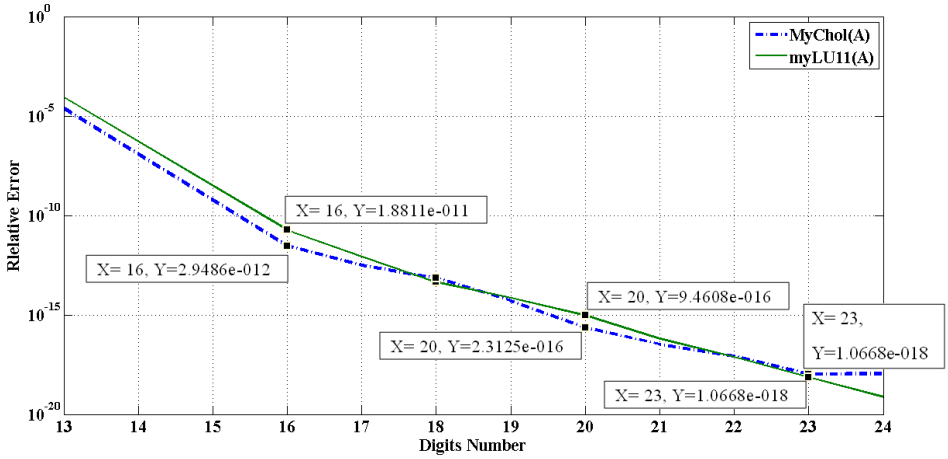

Figure 1: Residual relative error for different digits by Divide and Conquer subroutine MyChol (A) and myLU11 (A).

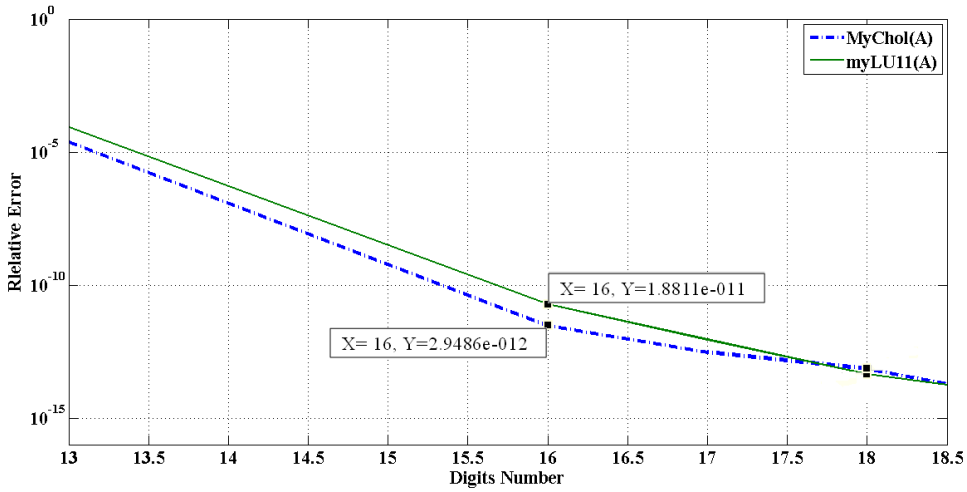

Figure 2: A part of Figure 1 to explain the difference between the two results.

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 7, No. 2 (2014)

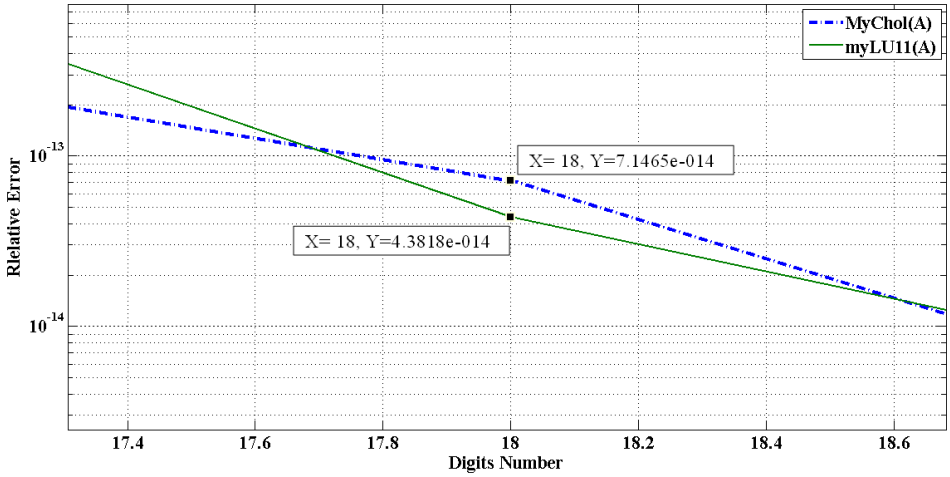

Figure 3: A part of Figure 1 to explain the difference between the two results.

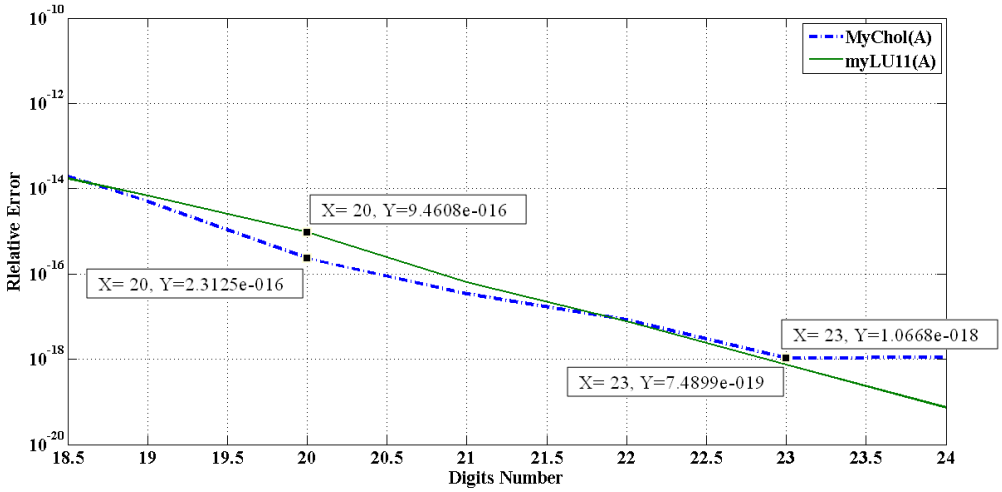

Figure 4: A part of Figure 1 to explain the difference between the two results.

International Electronic Journal of Pure and Applied Mathematics – IEJPAM, Volume 7, No. 2 (2014)

LU factorization.

References

[1] D. Bini, V. Pan, *Polynomial and Matrix Computations, Volume 1: Fundamental Algorithms*, Birkhäaser, Boston, 1994.

[2] V.Y. Pan, A. Zheng, X. Huang, O. Dias, Newton's iteration for inversion of Cauchy-like and other Structure matrices, *Journal of Complexity*, **13** (1997), 108-124.

[3] T.Sderstrm, G. Stewart, On the numerical properties of an iterative method for computing the Moore-Penrose generalized inverse, *SIAM Journal of Numerical Analysis*, **11** (1974), 61-74.

[4] *Book-Mathematics-Numerical Mathematics and Scientific Computation*, Volume 2 and 3, AkeBjorck, 1999.

[5] D.C. Lay, *Linear Algebra and its Applications*, 2-nd Edition, Addison-Wesley, 1997.

[6] D.S. Watkins, *Fundamentals of Matrix Computations*, 2-nd Edition, John Wiley & Sons, 2002.

[7] Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, *Computer Algorithms*, W.H. Freeman and Company, 1998.

[8] G.W. Collins, II, *Fundamental Numerical Methods and Data Analysis*, 2003.

[9] *MATLAB User's Guide*, Version 7.9.0, The Math Works (R2009b).

[10] Nicholas J. Higham, *Accuracy and Stability of Numerical Algorithms*, 2-nd Edition, SIAM, 2002.