A FAST ALGORITHM FOR SOLVING BANDED TOEPLITZ SYSTEMS

SKANDER BELHAJ ∗†, MARWA DRIDI †‡, AND AHMED SALAM ‡

Abstract. A fast algorithm for solving systems of linear equations with banded Toeplitz matrices is presented. This new approach is based on extending the given matrix with several rows at the top and several columns at the right and to assign zeros and some nonzero constants in each of these rows and columns in such a way that the augmented matrix has a lower triangular Toeplitz structure. Stability of the algorithm is discussed and its performance is showed by numerical experiments.

Key words. linear system, banded matrices, Toeplitz matrices, lower triangular Toeplitz matrices

AMS subject classifications. 15A15, 15A09, 15A23

1. Introduction. Toeplitz systems have been around for a long time and are encountered in various application fields. An interesting class of Toeplitz matrices consists of matrices having a banded Toeplitz form. A \( n \times n \) matrix \( T \) is said to be banded Toeplitz if

\[
T = \begin{pmatrix}
t_0 & t_{-1} & \cdots & t_{-m_r} \\
t_{1} & t_0 & t_{-1} & \ddots \\
\vdots & \ddots & \ddots & \ddots \\
t_{m_c} & t_{1} & t_{0} & t_{-1} & \cdots \\
& \ddots & \ddots & \ddots & \ddots \\
& & t_{m_c} & \cdots & t_{1} & t_0
\end{pmatrix}
\] (1.1)

where \( t_m \neq 0 \) and \( t_{-m} \neq 0 \).

Systems of linear equations with banded Toeplitz matrices arise in a variety of applications in mathematics and engineering, see [17, 11, 5] and the references therein. In the literature there are a number of fast direct methods for large linear systems with banded Toeplitz matrices [8, 3, 4, 7, 15, 16, 14, 6]. The most popular approach is the cyclic reduction invented by Bini and Meini [3, 4] which seems the fastest and probably stable for the symmetric positive definite case or in the unsymmetric case with a suitable diagonal dominance. However, the cyclic reduction method fails sometimes to give accurate results in the unsymmetric case. Malyshev and Sadkane [15] proposed recently an alternative approach based on spectral factorization [12, 18, 13] and Woodbury’s formula [2] which can give promising results in the symmetric and unsymmetric cases and when assuming that the band is not too large, i.e., \( m = \max (m_c, m_r) \ll n \).

Although the algorithm described in [15] seems to be the method of choice, it presents some instability. In fact, when using a large band \( m = \max (m_c, m_r) \) of \( T \),

---

*MANOUBA UNIVERSITY, ISAMM, CAMPUS UNIVERSITAIRE DE LA MANOUBA, 2010 TUNIS, TUNISIA (SKANDER.BELHAJ@LAMSIN.RNU.TN).
†UNIVERSITY OF TUNIS EL MANAR, ENIT-LAMSIN, BP 37, 1002, TUNIS, TUNISIA.
‡UNIVERSITY OF LILLE NORD DE FRANCE, ULCO, LMPA, BP 699, 62228 CALAIS, CEDEX, FRANCE.
the method proposed in [15] sometimes fails or requires a huge computational time to give a result. Therefore, we introduce a new method for solving systems of linear equations with banded Toeplitz matrices to resolve this problem.

Such approach is based on extending the given matrix with several rows at the top and several columns at the right and to assign zeros and some nonzero constants in each of these rows and columns in such a way that the augmented matrix has a lower triangular Toeplitz structure. The computational cost of our algorithm is about $O(n \log n) + O(m^2) + O((n - m_r) \log(n - m_r)) + O((m_r + m_e + 1)n)$.

The paper is organized as follows: section 2 gives some definitions and theoretical results. An efficient algorithm for solving systems of linear equations with banded Toeplitz matrices is proposed in section 3. We studied the error analysis in section 4 and in section 5 some numerical examples are given to put in evidence the potential advantages of our method with respect to the other well known methods, in terms of numerical stability and in terms of computational cost.

2. Definition and basic concept. In this section, we shall present some properties for a Toeplitz matrix:

**Definition 2.1.** $T_n = [t_{ij}]_{i,j=0}^{n-1}$ is a Toeplitz matrix if $t_{ij} = t_{i+k,j+k}$ for all positive $k$ (finite), that is, if all the entries of $T_n$ are invariant in their shifts in the diagonal direction, so that the matrix $T_n$ is completely defined by its first row and its first column.

Toeplitz matrix of size $n$ is completely specified by $2n - 1$ parameters, thus requiring less storage space than ordinary dense matrices. Moreover, many computations with Toeplitz matrices can be performed faster; this is the case, for instance, for the sum and the product by a scalar. Less trivial examples are given by the following results:

**Proposition 2.2 ([1]).** The multiplication of a Toeplitz matrix of size $n$ by a vector can be reduced to multiplication of two polynomials of degree at most $2n$ and performed with a computational cost of $O(n \log n)$.

**Lemma 2.3 (Some properties of triangular Toeplitz matrices).**

(i) Let $T_1$ and $T_2$ be two lower (upper) triangular Toeplitz matrices. Then $T_1T_2$ is also a lower (upper) triangular Toeplitz matrix, and $T_1T_2 = T_2T_1$.

(ii) If $T$ is a nonsingular lower (upper) triangular Toeplitz matrix, then $T^{-1}$ is also a lower (upper) triangular Toeplitz matrix. Therefore, to obtain $T^{-1}$, we only require to compute the entries of its first column (row).

(iii) Inverting a banded triangular Toeplitz matrix costs $O(kn)$, where $k$ is the band size.

**Algorithm 2.4 (Banded triangular Toeplitz matrices inversion).**

**Input:** $t$ the first column of a band triangular Toeplitz matrix $t = (t_1, \ldots, t_k)$

**Output:** $x$ the first column of the matrix inverse

```
x(1) = 1/t(1)
For i = 2 : k
x(i) = t(i : -1 : 2) * x(1 : i - 1)
x(i) = -x(i)/t(1)
End
For i = k + 1 : n
x(i) = t(k : -1 : 2) * x(i + 1 : k : i - 1)
x(i) = -x(i)/t(1)
End
```

**Remark 2.5.** There are two types of fast direct algorithms for solving Toeplitz
systems: Levinson-type algorithm which requires $O(n^2)$ flops, and Schur-type algorithm which requires $O(n \log^2 n)$ or $O(n \log^3 n)$.

3. Extension in a banded triangular Toeplitz matrix. In the following, we give an algorithm which compute the solution of a linear system of $n$ equations with a banded Toeplitz matrix.

**Proposition 3.1.** Let $T$ is a banded Toeplitz matrix defined as in (1.1). Thus, $T$ can be extended in a banded lower triangular matrix Toeplitz $M$ of size $(n+p) \times (n+p)$ which the first column of $M$ is given by $r = (t_{-m_r}, \ldots, t_{-1}, t_0, t_1, \ldots, t_{m_c}, 0, \ldots, 0)^T$ with $n-(m_c+1)$

More precisely,

$$M = \begin{pmatrix}
  t_{-m_r} & & & & \\
  \vdots & \ddots & & & \\
  t_1 & \cdots & t_{-m_r} & & \\
  t_0 & t_{-1} & \cdots & t_{-m_r} & \\
  \vdots & \ddots & \ddots & \ddots & \\
  t_{m_c} & t_1 & \cdots & t_{-1} & t_{-m_r} \\
  \vdots & \ddots & \ddots & \ddots & \ddots \\
  t_{m_c} & \cdots & t_1 & t_0 & t_{-m_r} \\
  t_{m_c} & \cdots & t_{-1} & t_0 & t_{-m_r}
\end{pmatrix}
$$

$$= \begin{pmatrix}
  L & 0 & \cdots & 0 \\
  \vdots & \ddots & \ddots & \ddots \\
  T & 0 & L
\end{pmatrix}, \text{ where } L = \begin{pmatrix}
  t_{-m_r} & \vdots & \vdots \\
  \vdots & \ddots & \ddots \\
  t_{-1} & \cdots & t_{-m_r}
\end{pmatrix}.
$$

In order to simplify, we note that $p = m_r$, and $q = m_c$.

**Proposition 3.2.** Let $M$ be a lower triangular Toeplitz matrix as defined in Proposition 3.1. Thus, $M^{-1}$ is a lower triangular Toeplitz matrix defined by its first
column: \((v_1, v_2, \ldots, v_{n+p})^T\). In addition, \(M^{-1}\) can be partitioned as follows:

\[
M^{-1} = \begin{pmatrix}
    v_1 & v_2 & \cdots & v_1 \\
    \vdots & \ddots & \vdots & \vdots \\
    v_{n-p+1} & \cdots & v_1 \\
    \vdots & \cdots & \vdots & \ddots \\
    v_{n-1} & \cdots & \vdots & \ddots \\
    v_n & v_{n-1} & \cdots & v_{n-p+1} \\
    v_{n+1} & v_n & \cdots & v_{n-p+2} \\
    v_{n+2} & \ddots & \vdots & \vdots \\
    \vdots & \ddots & \ddots & \vdots \\
    v_{n+p} & \cdots & v_n & v_{n+1} \\
\end{pmatrix}
\]

\[
= \begin{pmatrix}
    A & B \\
    C & D \\
\end{pmatrix}
\]

where

\[
A = \begin{pmatrix}
    v_1 & v_2 & \cdots & v_1 \\
    \vdots & \ddots & \vdots & \vdots \\
    v_{n-p+1} & \cdots & v_1 \\
    \vdots & \cdots & \vdots & \ddots \\
    v_{n-1} & \cdots & \vdots & \ddots \\
    v_n & v_{n-1} & \cdots & v_{n-p+1} \\
\end{pmatrix}, \quad
B = \begin{pmatrix}
    v_1 \\
    \vdots \\
    v_{n-p} \\
    v_n \\
\end{pmatrix}
\]

\[
C = \begin{pmatrix}
    v_{n+1} & v_n & \cdots & v_{n-p+2} \\
    v_{n+2} & \ddots & \vdots & \vdots \\
    \vdots & \ddots & \ddots & \vdots \\
    v_{n+p} & \cdots & v_n & v_{n+1} \\
\end{pmatrix}, \quad
D = \begin{pmatrix}
    v_{n-p+1} & \cdots & v_2 & v_1 \\
    \vdots & \ddots & \vdots & \vdots \\
    v_{n-1} & \cdots & \vdots & \ddots \\
    v_n & v_{n-1} & \cdots & v_2 \end{pmatrix}
\]

are matrices of size \(n \times p\), \(n \times n\), \(p \times p\), \(p \times n\), respectively.

**Theorem 3.3.** Let \(T\) be a nonsingular banded Toeplitz matrix and \(M\) be its associated lower triangular matrix. Suppose that \(M^{-1}\) is partitioned as follows

\[
M^{-1} = \begin{pmatrix}
    A & B \\
    C & D \\
\end{pmatrix}
\]

where \(A, B, C,\) and \(D\) are matrices of size \(n \times p\), \(n \times n\), \(p \times p\), \(p \times n\), respectively. Then, the inverse of \(T\) is given explicitly by

\[
T^{-1} = B - AC^{-1}D.
\]
Based on Theorem 3.3, a solution of $Tx = f$, where $T$ is defined in (1.1) and $x$, $f \in \mathbb{R}^n$ can be computed by an efficient complexity. In fact, the idea is to extend the solution in another equivalent system of size $(n + k)$

$$M \begin{pmatrix} x \\ 0 \end{pmatrix} = \begin{pmatrix} a \\ f \end{pmatrix} \quad (3.3)$$

where $a$ is an unknown vector of length $p$. We also consider the same partition as in (3.1)

$$M^{-1} = \begin{pmatrix} A & B \\ C & D \end{pmatrix}$$

where $A, B, C,$ and $D$ are matrices of size $n \times p$, $n \times n$, $p \times p$, $p \times n$, respectively.

Thus, the vector $a$ solve a $p \times p$ system

$$Ca = -Df.$$

Then, we can compute $a$ in $O(p \log^2 p)$ flops. In addition, $(x^T \ 0)^T$ is a solution of a linear system whose matrix is lower triangular Toeplitz matrix which can be computed in $O((p + n) \log (p + n))$ flops. Finally, the product $Df$ can be computed in $O(pn)$ flops. The main operations of the proposed are summarized in the following:

**Algorithm 3.4.**

- **Step 0:** Recover $A$, $B$, $C$ and $D$ as defined in (3.1) via the inverse of a lower triangular Toeplitz matrix $M$.
- **Step 1:** Compute $b = Bf$.
- **Step 2:** Compute $d = Df$.
- **Step 3:** Find $a$ by solving $Ca = -d$.
- **Step 4:** Compute $a' = Aa$.
- **Step 5:** Compute the solution $x$ using $x = a' + b$.

Thus, we have the following result:

**PROPOSITION 3.5.** A nonsingular linear system of $n$ equations with a banded Toeplitz matrix can be solved with a computational cost of

$$O(n \log n) + O(p^2) + O((n - p) \log (n - p)) + O((p + q + 1)n).$$

**Proof.** It is clear to use $O((p + q + 1)n)$ ops in Step 1 and $O(n \log n)$ ops in Step 2. Recall that the matrix-vector product $Mv$, where $M$ is a general $n \times n$ Toeplitz matrix, can be computed by the fast Fourier transform [9] in $O(n \log_2 n)$ arithmetic operations with a small constant. Thus, we need $O(p^2)$ ops (see (2.5)) in Step 3, $O(n \log n)$ in Step 4 and $O(n)$ in the last step of Algorithm 3.4. \qed

**Remark 3.6.** When $n \gg m$, the total cost of Algorithm 3.4 is about $O(n \log n)$ ops. In this case, it is clear that our alternative is more expensive than the other well known methods [15, 3, 4]. Otherwise, Algorithm 3.4 is a good approach in terms of numerical stability and in terms of computational cost.

**4. Error analysis.** Here we present a brief analysis of the forward error

$$\|x - \hat{x}\|/\|x\|,$$

where $\hat{x}$ is the solution of $Tx = f$ computed by Algorithm 3.4. Before this, we recall some classical results.
**Definition 4.1.** Evolution of expression in floating point arithmetic is denoted \( fl(\cdot) \), and we assume that the basic arithmetic operation \( op = +, -, *, / \) satisfies

\[
fl(x \ op \ y) = (x \ op \ y)(1 + \delta), \quad |\delta| \leq u
\]  

(4.1)

\( u \) is the unit roundoff (or machine precision).

Let \( x \in \mathbb{R} \), \( \hat{x} = fl(x) = x(1 + \delta) \) where \( \hat{x} \) be an approximation to \( x \).

**Theorem 4.2.** Consider the inner product \( s_n = x^T y \), where \( x, y \in \mathbb{R}^n \)

\[
\hat{s}_n = fl(x^T y) = x^T (y + \Delta y) = (x + \Delta x)^T y, \quad |\Delta x| < \gamma_n |x|, |\Delta y| < \gamma_n |y|
\]  

(4.2)

\[
|x^T y - fl(x^T y)| \leq \gamma_n |x^T| |y|
\]  

(4.3)

where \( \gamma_n = \frac{nu}{1 - nu} \).

*Proof.* See [10]. \( \square \)

Let \( A \in \mathbb{R}^{m \times n}, x \in \mathbb{R}^n \), and \( y = Ax \), the vector \( y \) can be formed as inner products \( y_i = a_i^T x, i = 1 \ldots m \), where the \( a_i^T \) is the \( i \)-th column of \( A \)

\[
\hat{y}_i = (a_i + \Delta a_i)^T x, \quad |\Delta a_i| \leq \gamma_n |a_i|
\]  

(4.4)

Thus, we give the backward error

\[
\hat{y} = (A + \Delta A)^T x, \quad |\Delta A| \leq \gamma_n |A|
\]  

(4.5)

which implies the forward error bound

\[
|y - \hat{y}| \leq \gamma_n |A||x|
\]  

(4.6)

Then, we have

\[
\|y - \hat{y}\|_p \leq \gamma_n \|A\|_p\|x\|_p, \quad p = 1, \infty
\]  

(4.7)

and for the 2-norm using

\[
\|y - \hat{y}\|_2 \leq \gamma_n \|A\|_2\|x\|_2.
\]  

(4.8)

**Lemma 4.3.** If \( X_j + \Delta X_j \in \mathbb{R}^{m \times n} \) satisfies \( \|\Delta X_j\|_F \leq \delta_j\|X_j\|_2 \) for all \( j \), with a consistent norm \( \|\cdot\| \) which depends on \( A \) and \( \varepsilon \) such that \( \|A\| \leq \rho(A) + \varepsilon \). In addition, if \( \rho(A) < 1 \) then there is a consistent norm \( \|\cdot\| \) such that \( \|A\| < 1 \).

**Theorem 4.4.** Let be \( \rho(B) = \max\{ |\lambda| : \det(B - \lambda I) = 0 \} \) the spectral radius. Let \( A \in C^{n \times n} \) and \( \varepsilon > 0 \). There is a consistent norm \( \|\cdot\| \) which depends on \( A \) and \( \varepsilon \) such that \( \|A\| \leq \rho(A) + \varepsilon \). In addition, if \( \rho(A) < 1 \) then there is a consistent norm \( \|\cdot\| \) such that \( \|A\| < 1 \).

Now we begin the forward error analysis. Suppose \( X = T^{-1}, E = C^{-1} \) where \( \hat{A}, \hat{E}, \hat{D}, \hat{B} \) computed by Algorithm 3.4 and

\[
-\hat{A}\hat{E}\hat{D} + \hat{B} = \Delta L + X, \quad |\Delta L| < \gamma_n |\hat{A}| |\hat{E}| |\hat{D}| + |\hat{B}|
\]
\[ \hat{x} = -(\hat{A} + \Delta A)\hat{y} + (\hat{B} + \Delta B)f, \quad |\Delta A| < \gamma_n \quad |\hat{A}| = |\Delta B| < \gamma_n \quad |\hat{B}| \]

\[ = -(\hat{A} + \Delta A)(\hat{E} + \Delta E)\hat{z} + (\hat{B} + \Delta B)f, \quad |\Delta E| < \gamma_n \quad |\hat{E}| \]

\[ = -(\hat{A} + \Delta A)(\hat{E} + \Delta E)(\hat{D} + \Delta D)f + (\hat{B} + \Delta B)f, \quad |\Delta D| < \gamma_n \quad |\hat{D}| \]

If \( |\hat{A}\hat{E}\hat{D}| = |\hat{A}| |\hat{E}| |\hat{D}| \) (where \(\hat{E}, \hat{D}, \hat{A}\) non-negative matrices)
\[ |x - \hat{x}| \leq ((1 + \gamma_n)^3 - 1) |\hat{A}\hat{E}\hat{D}| |f| + \gamma_n |\hat{B}| |f|. \]

Or,
\[ |\hat{A}\hat{E}\hat{D}| \leq |X| + |\Delta L| + |\hat{B}| \]
\[ \leq |X| + \gamma_n |\hat{A}\hat{E}\hat{D}| + (\gamma_n + 1) |\hat{B}| \]
\[ |\hat{A}\hat{E}\hat{D}| \leq \frac{1}{1 - \gamma_n} |X| + \gamma_n + 1 |\hat{B}|. \]

Thus,
\[ |x - \hat{x}| \leq \frac{((1 + \gamma_n)^3 - 1)}{1 - \gamma_n} |X| |f| + \left( \frac{((1 + \gamma_n)^3 - 1)(\gamma_n + 1)}{(1 - \gamma_n)} + \gamma_n \right) |\hat{B}| |f|. \]

Via the \(p\)-norm (when \(p = 1, \infty\)) and applying Theorem 4.4, we obtain the following:
\[ \frac{\|x - \hat{x}\|_p}{\|x\|_p} \leq \frac{((1 + \gamma_n)^3 - 1)}{1 - \gamma_n} \|T^{-1}\|_p \|T\|_p + \left( \frac{((1 + \gamma_n)^3 - 1)(\gamma_n + 1)}{(1 - \gamma_n)} + \gamma_n \right) \|T\|_p, \]
where
\[ |(1 + \gamma_n)^n - 1| < n\gamma_n \frac{1}{1 - \frac{n\gamma_n}{2}} = \eta_n. \quad (4.9) \]

(4.9) is justified since \( (1 + \gamma_n)^n - 1 \leq (1 + u)^n - 1 \) and \( (1 + u)^n \leq \exp(nu) \) for \( u \geq 0 \), then
\[ (1 + u)^n - 1 < nu + \frac{(nu)^2}{2} + \frac{(nu)^2}{2} + \frac{(nu)^2}{2} + \cdots = nu \frac{1}{1 - \frac{n\gamma_n}{2}}. \]

For more details, see [10].

Finally, the forward error satisfies the estimate
\[ \frac{\|x - \hat{x}\|_p}{\|x\|_p} \leq \mathcal{O}_1(\gamma_n) \times \|T^{-1}\|_p \|T\|_p + \mathcal{O}_2(\gamma_n) \times \|T\|_p, \]
where
\[ \mathcal{O}_1(\gamma_n) = \frac{(1 + \gamma_n)^3 - 1}{1 - \gamma_n} \quad \text{and} \quad \mathcal{O}_2(\gamma_n) = \frac{((1 + \gamma_n)^3 - 1)(\gamma_n + 1)}{(1 - \gamma_n)} + \gamma_n. \]

To illustrate error analysis results, we plot in Fig. 6.1 the \(\infty\)-norm of the errors between the exact solution \(x\) and our computed solution \(x_{OUR}\), the computed solution \(x_{CR}\) via the standard cyclic reduction and the computed solution \(x_{SMW}\) via Sherman-Morrison-Woodbury, respectively.

It is clear that the numerical results coincide with the theoretical results.
5. Examples and numerical tests. Algorithm 3.4 presented in this paper has been implemented in Matlab (R2011) and run on an Intel(R) core(TM)i3 CPU M380 laptop with a 2.53 GHz processor and 4 GO of RAM. We have implemented the standard cyclic reduction from [3] and the algorithm via Sherman-Morrison-Woodbury from [15].

All numerical examples of varied difficulty are recovered from [15] with a right hand side \( f \) chosen so that the exact solution has all its components equal to 1. Thus, the \( \infty \)-norm of the error between the exact solution \( x \) and computed solution \( x_C \) equals \( \max_{1 \leq i \leq n} |(x_C)_i - x| \).

Tables 5.1-5.4 show the performance of Algorithm 3.4 when \( T \) is an \( n \times n \) banded Toeplitz matrix. Tables indicate the \( \infty \)-norm of the error between the exact solution \( x \) and the computed solution \( x_C \), and execution (CPU) time in seconds of our method, the algorithm via Sherman-Morrison-Woodbury and the cyclic reduction approach.

**Example 1:** The banded Toeplitz matrix \( T \) has the coefficients \( t_1 = 1000, t_0 = 1, t_{-1} = 0.001, t_{-2} = 1000 \). The solution \( x_{OUR} \) computed by Algorithm 3.4 and the solution \( x_{SMW} \) computed by formula Sherman-Morrison-Woodbury possess the accuracy \( \|x-x_{OUR}\|_\infty = 2.21 \times 10^{-12} \) and \( \|x-x_{SMW}\|_\infty = 1.9 \times 10^{-14} \), respectively. Whereas, the cyclic reduction fails to solve this example because \( T \) is far from being diagonally dominant and it must solve very ill-conditioned systems during the execution.

**Example 2:** The banded Toeplitz matrix has \( t_0 = 0.5 \) on the main diagonal and 1 elsewhere within the band. The order of \( T \) is fixed at \( n = 2^{20} \), and \( m_r = m \) and \( m_c = m/2 \), where \( m \) varies. The results given in Table 6.1 show that when \( m \) increases, Algorithm 3.4 performs much better than the cyclic reduction algorithm and the algorithm via Sherman-Morrison-Woodbury in terms of numerical efficiency and computational time. Besides, the accuracy of the cyclic reduction method deteriorates and the method via Sherman-Morrison Woodbury does not work.

**Example 3:** In the following example, we choose \( t_0 = 1.0001 \) on the main diagonal and 1 elsewhere within the band. The order of \( T \) is fixed at \( n = 2^{20} \) and we chose \( q = m \) and \( p = m/2 \), where \( m \) varies. The results are given in Table 6.2. We see that as \( m \) increases, Algorithm 3.4 performs much better than the method via Sherman-Morrison Woodbury in terms of computational time. Moreover, the cyclic reduction method fails.

**Example 4:** Let \( t_0 = 1 + 10^{-14} \) on the main diagonal and 1 the coefficients of Banded Toeplitz matrix, elsewhere within the band. The order of \( T \) is fixed at \( n = 2^{20} \) and we chose \( p = q = m \), and \( n = 2^{20} \). The results given in Table 6.3 show that even if \( m << n \), Algorithm 3.4 performs better than the cyclic reduction algorithm and the algorithm via Sherman-Morrison-Woodbury in terms of numerical efficiency and computational time.

**Example 5:** The \( T \) matrix is a banded Toeplitz matrix, has \( t_0 = 1 \) on the main diagonal and 2 elsewhere within the band. The order of \( T \) is fixed at \( n = 2^{20} \), and we choose \( p = m \) and \( q = m \), where \( m \) varies. Table 6.4 strengthens the efficiency of our approach with respect to the cyclic reduction algorithm and the algorithm via Sherman-Morrison-Woodbury in terms of accuracy and computational time.

**Example 6:** The matrix \( T \) is banded Toeplitz with coefficients \( t_3 = 1, t_2 = 3, t_1 = 2, \),
$t_0 = 3/5, t_{-1} = 4$. Table 6.5 strengthens the efficiency of our approach with respect to the cyclic reduction algorithm and the algorithm via Sherman-Morrison-Woodbury in terms of accuracy and computational time.

5.1. Ill-conditioned matrices. Note that we replace the banded Toeplitz matrix $T$ with the triangular Toeplitz matrix $M$ in the solution process. This has positive and negative effects, as we will show. Among the advantage of our method, whatever matrix (not limited to a symmetric matrix or a diagonal dominant matrix) can be used and also the spectral factorization method is not needed.

5.1.1. Ill-conditioned $M$. It could happen that even for a well-conditioned matrix $T$, $M$ would be ill-conditioned or singular. This is illustrated by the following example: consider the case when $T$ is a tridiagonal, symmetric positive definite, strong diagonally dominant, with $a > 0$ on the main diagonal and $\varepsilon$, $0 < \varepsilon << a$ on its first upper and lower diagonal. The matrix $M$ is of order $n + 1$, has $\varepsilon = 1$ on its diagonal, $a = 10$ on its first subdiagonal, $\varepsilon = 1$ on its second subdiagonal and 0 elsewhere. The condition number of $T$ and the embedding triangular matrix $M$ for different sizes $n$, are given in Table 6.6.

5.1.2. Ill-conditioned $T$. In this case the extension of the problem to a triangular Toeplitz matrix $M$ has a regularizing effect. For instance, we consider a tridiagonal matrix $T$ with $a > 0$ on the main diagonal and $b$, $0 < a << b$ on its first upper and lower diagonals. We choose $a = 10^{-30}$ and $b = 1$. $f$ is also chosen so that the exact solution is the vector of all ones. The condition number of $T$ and the embedding triangular matrix $M$, for different sizes $n$, are given in Table 6.7.

6. Concluding remarks. In this paper, we proposed a new method for solving systems of linear equations with banded Toeplitz matrices. Numerical examples show that the proposed algorithm is one of the good alternatives in terms of efficiency and computational time when assuming that the band is moderately large. Unfortunately, the new algorithm might suffer from some numerical instabilities. In fact, even though the original banded Toeplitz matrix is well conditioned, the embedding lower triangular Toeplitz matrix might be very ill-conditioned (For instance, for a banded Toeplitz matrix $T$ with coefficients $t_1 = -1$, $t_0 = -2$, $t_{-1} = 1$). Moreover, properties like positive definiteness or diagonal dominance of the original banded system are sometimes lost in the new triangular system (For instance, for a tridiagonal, symmetric positive definite and strongly diagonally dominant matrix $T$, with $a > 0$ on the main diagonal and $\varepsilon$, $0 < \varepsilon << a$ on its first upper and lower diagonals).

REFERENCES

Fig. 6.1. $\log_{10}(Error_A)$ (head), $\log_{10}(Error_{A_{CR}})$ (middle) and $\log_{10}(Error_{A_{SMW}})$ (down) for a banded Toeplitz matrix of size $20^2$ and $m_r = m_c = 32$
A Fast algorithm for solving banded Toeplitz systems

Table 6.1
Numerical result for Example 2

<table>
<thead>
<tr>
<th>( m )</th>
<th>( |x - x_{OUR}|_\infty )</th>
<th>Time (s)</th>
<th>( |x - x_{SMW}|_\infty )</th>
<th>Time(_{SMW}) (s)</th>
<th>( |x - x_{CR}|_\infty )</th>
<th>Time(_{CR}) (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>( 4.60 \times 10^{-14} )</td>
<td>0.53</td>
<td>( 1.82 \times 10^{-14} )</td>
<td>0.81</td>
<td>( 3.75 \times 10^{-14} )</td>
<td>0.22</td>
</tr>
<tr>
<td>8</td>
<td>( 8.50 \times 10^{-14} )</td>
<td>0.53</td>
<td>( 1.84 \times 10^{-14} )</td>
<td>0.90</td>
<td>( 2.40 \times 10^{-14} )</td>
<td>0.21</td>
</tr>
<tr>
<td>16</td>
<td>( 3.53 \times 10^{-10} )</td>
<td>0.8</td>
<td>( 1.97 \times 10^{-10} )</td>
<td>1.80</td>
<td>( 2.94 \times 10^{-10} )</td>
<td>0.34</td>
</tr>
<tr>
<td>32</td>
<td>( 6.30 \times 10^{-14} )</td>
<td>0.9</td>
<td>( 2.31 \times 10^{-14} )</td>
<td>2.26</td>
<td>( 4.37 \times 10^{-14} )</td>
<td>0.36</td>
</tr>
<tr>
<td>64</td>
<td>( 1.18 \times 10^{-10} )</td>
<td>0.98</td>
<td>does not work</td>
<td>1.88 \times 10^{-9}</td>
<td>does not work</td>
<td>0.52</td>
</tr>
<tr>
<td>128</td>
<td>( 8.18 \times 10^{-10} )</td>
<td>1.02</td>
<td>does not work</td>
<td>7.89 \times 10^{-9}</td>
<td>does not work</td>
<td>0.64</td>
</tr>
<tr>
<td>256</td>
<td>( 8.18 \times 10^{-10} )</td>
<td>1.15</td>
<td>does not work</td>
<td>1.35 \times 10^{-7}</td>
<td>does not work</td>
<td>1.23</td>
</tr>
<tr>
<td>512</td>
<td>( 9.83 \times 10^{-8} )</td>
<td>1.27</td>
<td>does not work</td>
<td>2.24 \times 10^{-7}</td>
<td>does not work</td>
<td>8.83</td>
</tr>
<tr>
<td>1024</td>
<td>( 6.18 \times 10^{-8} )</td>
<td>1.58</td>
<td>does not work</td>
<td>2.85 \times 10^{-6}</td>
<td>does not work</td>
<td>35.41</td>
</tr>
</tbody>
</table>

Table 6.2
Numerical result for Example 3.

<table>
<thead>
<tr>
<th>( m )</th>
<th>( |x - x_{OUR}|_\infty )</th>
<th>Time (s)</th>
<th>( |x - x_{SMW}|_\infty )</th>
<th>Time(_{SMW}) (s)</th>
<th>( |x - x_{CR}|_\infty )</th>
<th>Time(_{CR}) (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>32</td>
<td>( 2.64 \times 10^{-7} )</td>
<td>0.54</td>
<td>( 3.77 \times 10^{-12} )</td>
<td>1.18</td>
<td>Failure</td>
<td>0.62</td>
</tr>
<tr>
<td>64</td>
<td>( 1.59 \times 10^{-10} )</td>
<td>0.54</td>
<td>( 7.48 \times 10^{-12} )</td>
<td>1.2</td>
<td>Failure</td>
<td>0.75</td>
</tr>
<tr>
<td>128</td>
<td>( 3.35 \times 10^{-10} )</td>
<td>0.86</td>
<td>( 2.01 \times 10^{-11} )</td>
<td>1.4</td>
<td>Failure</td>
<td>1.81</td>
</tr>
<tr>
<td>256</td>
<td>( 8.73 \times 10^{-17} )</td>
<td>0.57</td>
<td>( 4.68 \times 10^{-8} )</td>
<td>7.344</td>
<td>Failure</td>
<td>2.76</td>
</tr>
<tr>
<td>512</td>
<td>( 2.72 \times 10^{-16} )</td>
<td>0.69</td>
<td>( 7.82 \times 10^{-7} )</td>
<td>38.18</td>
<td>Failure</td>
<td>2.34</td>
</tr>
<tr>
<td>1024</td>
<td>( 1.04 \times 10^{-8} )</td>
<td>0.82</td>
<td>( 8.30 \times 10^{-8} )</td>
<td>315.789</td>
<td>Failure</td>
<td>8.2614</td>
</tr>
</tbody>
</table>

Table 6.3
Numerical result for Example 4.

<table>
<thead>
<tr>
<th>( m )</th>
<th>( |x - x_{OUR}|_\infty )</th>
<th>Time (s)</th>
<th>( |x - x_{SMW}|_\infty )</th>
<th>Time(_{SMW}) (s)</th>
<th>( |x - x_{CR}|_\infty )</th>
<th>Time(_{CR}) (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>4</td>
<td>( 4.27 \times 10^{-3} )</td>
<td>0.73</td>
<td>( 6.3 \times 10^{-3} )</td>
<td>0.62</td>
<td>( 3.5 \times 10^{-20} )</td>
<td>0.52</td>
</tr>
<tr>
<td>8</td>
<td>( 2.7145 \times 10^{-11} )</td>
<td>0.86</td>
<td>( 5.2 \times 10^{-4} )</td>
<td>0.7</td>
<td>( 3 \times 10^{-11} )</td>
<td>0.53</td>
</tr>
</tbody>
</table>

Table 6.4
Numerical result for Example 5.

<table>
<thead>
<tr>
<th>( m )</th>
<th>( |x - x_{OUR}|_\infty )</th>
<th>Time (s)</th>
<th>( |x - x_{SMW}|_\infty )</th>
<th>Time(_{SMW}) (s)</th>
<th>( |x - x_{CR}|_\infty )</th>
<th>Time(_{CR}) (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>8</td>
<td>( 8.50 \times 10^{-14} )</td>
<td>0.62</td>
<td>( 5.11 \times 10^{-14} )</td>
<td>0.85</td>
<td>( 2.40 \times 10^{-14} )</td>
<td>0.22</td>
</tr>
<tr>
<td>16</td>
<td>( 3.53 \times 10^{-10} )</td>
<td>0.8</td>
<td>( 1.44 \times 10^{-9} )</td>
<td>1.46</td>
<td>( 2.94 \times 10^{-10} )</td>
<td>0.29</td>
</tr>
<tr>
<td>32</td>
<td>( 6.3 \times 10^{-13} )</td>
<td>0.73</td>
<td>( 3.00 \times 10^{-6} )</td>
<td>1.57</td>
<td>( 6.47 \times 10^{-9} )</td>
<td>0.29</td>
</tr>
<tr>
<td>64</td>
<td>( 1.18 \times 10^{-10} )</td>
<td>0.51</td>
<td>( 3.48 \times 10^{-9} )</td>
<td>2.13</td>
<td>( 1.89 \times 10^{-9} )</td>
<td>0.25</td>
</tr>
<tr>
<td>128</td>
<td>( 1.97 \times 10^{-10} )</td>
<td>0.58</td>
<td>does not work</td>
<td>7.89 \times 10^{-9}</td>
<td>does not work</td>
<td>0.52</td>
</tr>
<tr>
<td>256</td>
<td>( 8.18 \times 10^{-10} )</td>
<td>0.6</td>
<td>does not work</td>
<td>1.35 \times 10^{-9}</td>
<td>does not work</td>
<td>1.58</td>
</tr>
<tr>
<td>512</td>
<td>( 9.83 \times 10^{-8} )</td>
<td>0.69</td>
<td>does not work</td>
<td>2.24 \times 10^{-7}</td>
<td>6.09</td>
<td></td>
</tr>
<tr>
<td>1024</td>
<td>( 6.18 \times 10^{-8} )</td>
<td>0.7</td>
<td>does not work</td>
<td>2.85 \times 10^{-6}</td>
<td>31.31</td>
<td></td>
</tr>
</tbody>
</table>

Table 6.5
Numerical result for Example 6.

<table>
<thead>
<tr>
<th>( n )</th>
<th>( |x - x_{OUR}|_\infty )</th>
<th>Time (s)</th>
<th>( |x - x_{SMW}|_\infty )</th>
<th>Time(_{SMW}) (s)</th>
<th>( |x - x_{CR}|_\infty )</th>
<th>Time(_{CR}) (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2^1</td>
<td>( 1.665 \times 10^{-14} )</td>
<td>0.001</td>
<td>( 2.977 \times 10^{-13} )</td>
<td>0.25</td>
<td>( 1.9667 )</td>
<td>0.0082</td>
</tr>
<tr>
<td>2^2</td>
<td>( 9.226 \times 10^{-14} )</td>
<td>0.002</td>
<td>does not work</td>
<td>( 1.9667 )</td>
<td>( 0.0085 )</td>
<td></td>
</tr>
<tr>
<td>2^3</td>
<td>( 3.619 \times 10^{-14} )</td>
<td>0.004</td>
<td>does not work</td>
<td>( 1.9667 )</td>
<td>0.01</td>
<td></td>
</tr>
<tr>
<td>2^4</td>
<td>( 1.654 \times 10^{-13} )</td>
<td>0.004</td>
<td>does not work</td>
<td>( 1.9667 )</td>
<td>0.055</td>
<td></td>
</tr>
<tr>
<td>2^5</td>
<td>( 1.722 \times 10^{-14} )</td>
<td>0.005</td>
<td>does not work</td>
<td>( 1.9667 )</td>
<td>0.086</td>
<td></td>
</tr>
</tbody>
</table>
Table 6.6
The condition number of $T$ and the embedding triangular matrix $M$ for different sizes $n$: Ill-conditioned $M$ case

<table>
<thead>
<tr>
<th>$n$</th>
<th>$\text{cond}_\infty(T)$</th>
<th>$\text{cond}_\infty(M)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$2^7$</td>
<td>1.500</td>
<td>$3.676 \times 10^{28}$</td>
</tr>
<tr>
<td>$2^8$</td>
<td>1.500</td>
<td>$1.02 \times 10^{26}$</td>
</tr>
<tr>
<td>$2^9$</td>
<td>1.500</td>
<td>inf</td>
</tr>
<tr>
<td>$2^{10}$</td>
<td>1.500</td>
<td>inf</td>
</tr>
<tr>
<td>$2^{11}$</td>
<td>1.500</td>
<td>inf</td>
</tr>
</tbody>
</table>

Table 6.7
The condition number of $T$ and the embedding triangular matrix $M$ for different sizes $n$: Ill-conditioned $T$ case

<table>
<thead>
<tr>
<th>$n$</th>
<th>$\text{cond}_\infty(T)$</th>
<th>$\text{cond}_\infty(M)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$2^7$</td>
<td>$2 \times 10^{39}$</td>
<td>130</td>
</tr>
<tr>
<td>$2^8$</td>
<td>$2 \times 10^{39}$</td>
<td>258</td>
</tr>
<tr>
<td>$2^9$</td>
<td>$2 \times 10^{39}$</td>
<td>514</td>
</tr>
<tr>
<td>$2^{10}$</td>
<td>$2 \times 10^{39}$</td>
<td>1026</td>
</tr>
<tr>
<td>$2^{11}$</td>
<td>$2 \times 10^{39}$</td>
<td>2050</td>
</tr>
<tr>
<td>$2^{12}$</td>
<td>$2 \times 10^{39}$</td>
<td>4098</td>
</tr>
</tbody>
</table>