

**Fast direct solution methods for  
symmetric banded Toeplitz systems,  
based on the sine transform**

*Jef Hendrickx  
Marc Van Barel*

*Report TW 316, October 2000*



**Katholieke Universiteit Leuven**  
Department of Computer Science  
Celestijnenlaan 200A – B-3001 Heverlee (Belgium)

# Fast direct solution methods for symmetric banded Toeplitz systems, based on the sine transform

*Jef Hendrickx*  
*Marc Van Barel*

*Report TW 316, October 2000*

Department of Computer Science, K.U.Leuven

## Abstract

We present new fast direct methods for solving a large symmetric banded Toeplitz system of order  $n$  with bandwidth  $p$ . We make use of structured matrices which can be diagonalized by the discrete sine transform matrix, sometimes called  $\tau$ -matrices. A first method writes the Toeplitz matrix as the sum of a  $\tau$ -matrix and a low rank matrix. A second method embeds the Toeplitz matrix in a larger  $\tau$ -matrix of order  $m$ . The methods are similar to Jain [A.K. Jain. *IEEE Trans. Acoust. Speech Signal Process.*, 26: 121-126, 1978] and Linzer [E. Linzer. *Linear Algebra Appl.*, 170:1-32, 1992], who worked with circulant matrices. Both algorithms consist in solving two  $\tau$ -systems and two smaller systems. A  $\tau$ -system of order  $n$  can be solved in  $O(n \log n)$  by using a discrete sine transform if  $n+1$  has small prime factors. Therefore, the second algorithm is preferable, since we can choose  $m$  such that  $m+1$  has small prime factors. On the other hand, in the second method the smaller systems can become large when  $m$  differs too much from  $n$ , while in the first method the order is always  $p-1$ . In both methods, the small systems have low displacement rank, so we can use fast methods to solve them.

**Keywords :** banded Toeplitz system, sine transform,  $\tau$ -class, fast direct methods, displacement theory

**AMS(MOS) Classification :** Primary : 65F05, Secondary : 65F50, 47L80.

**Fast direct solution methods for symmetric banded Toeplitz systems,  
based on the sine transform** \* †

Jef Hendrickx and Marc Van Barel

*Department of Computer Science*

*Katholieke Universiteit Leuven*

*Celestijnenlaan 200 A*

*B-3001 Heverlee, Belgium*

*e-mail: {Jef.Hendrickx,Marc.VanBarel}@cs.kuleuven.ac.be*

---

ABSTRACT

We present new fast direct methods for solving a large symmetric banded Toeplitz system of order  $n$  with bandwidth  $p$ . We make use of structured matrices which can be diagonalized by the discrete sine transform matrix, sometimes called  $\tau$ -matrices. A first method writes the Toeplitz matrix as the sum of a  $\tau$ -matrix and a low rank matrix. A second method embeds the Toeplitz matrix in a larger  $\tau$ -matrix of order  $m$ . The methods are similar to Jain [21] and Linzer [24], who worked with circulant matrices. Both algorithms consist in solving two  $\tau$ -systems and two smaller systems. A  $\tau$ -system of order  $n$  can be solved in  $O(n \log n)$  by using a discrete sine transform if  $n + 1$  has small prime factors. Therefore, the second algorithm is preferable, since we can choose  $m$  such that  $m + 1$  has small prime factors. On the other hand, in the second method the smaller systems can become large when  $m$  differs too much from  $n$ , while in the first method the order is always  $p - 1$ . In both methods, the small systems have low displacement rank, so we can use fast methods to solve them.

---

1. Introduction

In this paper we present two new methods for solving a system  $T\mathbf{x} = \mathbf{b}$ , where  $T$  is a symmetric banded Toeplitz matrix of order  $n$  and bandwidth

---

\*The work of the authors is supported by the Belgian Programme on Interuniversity Poles of Attraction, initiated by the Belgian State, Prime Minister's Office for Science, Technology and Culture. The scientific responsibility rests with the authors.

†This research was partially supported by the K.U.Leuven (Bijzonder Onderzoeksfonds), project "SLAP: Structured Linear Algebra Package," grant #OT/00/16.

$p$ , i.e. the elements  $t_{ij}$  of  $T$  satisfy  $t_{ij} = t_{|i-j|}$  and  $t_k = 0$  for  $k > p$ . Such problems occur in many practical problems, like the numerical solution of ordinary differential equations, the numerical solution of Markov chains in the theory of queueing problems, or it can be used as a preconditioner for a dense Toeplitz matrix. We refer to [3, 21] for some references to applications.

The methods are based on the fact that a banded Toeplitz matrix differs in only a few elements from a matrix that can be diagonalized by a sine transform, called  $\tau$ -matrices by Bini and Capovani [1, 2]. A system with a  $\tau$ -matrix of order  $n$  can be solved very fast in  $O(n \log n)$  operations by making use of the discrete sine transform if  $n + 1$  is a power of 2 or has at least small prime factors [27, 29]. This has made matrices from the  $\tau$ -class very suitable as preconditioner for banded or dense Toeplitz systems; see e.g. [5, 8, 10]. These preconditioners are used when solving the system by an iterative method, for instance the preconditioned conjugate gradient method. Yet in this paper we will describe some direct methods for solving banded Toeplitz matrices. A first method, the correction method, uses the fact that the banded Toeplitz matrix can be written as the sum of a  $\tau$ -matrix and a low rank matrix to reduce the system to two  $\tau$ -systems of order  $n$  and two smaller systems of order  $p - 1$ . The operation count is therefore  $O(n \log n + p^3)$  if  $n + 1$  has small prime factors. However, we will show that the smaller systems are generalized Toeplitz-plus-Hankel matrices with displacement rank 3, so the operation count can be reduced to  $O(n \log n + p^2)$  or even  $O(n \log n + p \log^2 p)$  if  $n + 1$  has small prime factors. Of course, the restriction for  $n$  is a disadvantage of the method. The second method embeds the Toeplitz matrix in a larger  $\tau$ -matrix. The order  $m$  of the  $\tau$ -matrix must be at least  $n + p - 1$  and can be chosen such that  $m + 1$  has small prime factors. The Toeplitz system reduces to two  $\tau$ -systems of order  $m$  and two smaller systems of order  $r = \lfloor (m - n)/2 \rfloor$ . Because of the choice for  $m$  the  $\tau$ -systems can always be solved fast in  $O(m \log m)$ . The matrices of the smaller systems are Toeplitz-plus-Hankel, so we can use displacement theory to solve them. Therefore, the operation count for the embedding method is  $O(m \log m + r^2)$  for any value of  $n$ . If we use superfast solvers this can even be reduced to  $O(m \log m + r \log^2 r)$ . However, if  $m$  differs too much from  $n$ ,  $r$  can become rather large and this can have an important impact on the performance of the method.

The methods are similar to the methods of Jain [21] and Linzer [24], who worked with circulant matrices, although the elaboration of the use of displacement theory in solving the smaller systems is new. Circulant matrices can be diagonalized by a discrete Fourier transform matrix, so we can use FFT to solve circulant systems of order  $n$  in  $O(n \log n)$  if  $n$  has small prime factors. Therefore, circulant matrices have become very popular as preconditioner for Toeplitz or other systems [9, 19, 28]. We

mention also that recently Bini and Meini [3] developed a new method for banded Toeplitz systems based on cyclic reduction.

The outline of the paper is as follows. In section 2 we will first review some properties about  $\tau$ -matrices. In section 3 we will present the first method, the correction method. The Toeplitz system can be reduced to two  $\tau$ -systems and two smaller systems. In section 4 we will see that we can use displacement theory to solve the smaller systems. The second method, the embedding method, is described in section 5. Finally, in section 6 we will compare the different methods among each other and to some classical methods in some numerical examples.

## 2. $\tau$ -matrices

We denote by  $F_n$  the discrete Fourier transform matrix of order  $n$ :

$$F_n = \left[ \sqrt{\frac{1}{n}} e^{-\frac{jk\pi}{n}i} \right]_{j,k=0}^{n-1}.$$

$F_n$  is a unitary matrix. It is well known that every matrix that can be diagonalized by  $F_n$  is a circulant matrix and vice versa. In other words, the class of circulant matrices is identical with the class of matrices that can be diagonalized by  $F_n$ .

Let  $S_n$  denote the discrete sine transform matrix of order  $n$ :

$$S_n = \left[ \sqrt{\frac{2}{n+1}} \sin \frac{jk\pi}{n+1} \right]_{j,k=1}^n.$$

The matrix  $S_n$  is an orthogonal, symmetric matrix, this means  $S_n^{-1} = S_n^T = S_n$ . We consider the matrices that can be diagonalized by  $S_n$ :

$$S_n \Lambda S_n \tag{1}$$

where  $\Lambda$  is an arbitrary diagonal matrix of order  $n$ . These matrices are called  $S$ -matrices by Mertens and Van de Vel [25]. Like the matrices that can be diagonalized by  $F_n$  are circulant matrices, we look similarly for a specific property that characterizes the matrices that can be diagonalized by  $S_n$ . Bini and Capovani [1, 2] introduced the  $\tau$ -class, this is the class of matrices  $A = (a_{ij})$  of order  $n$  that satisfy the following ‘‘cross-sum’’ condition:

$$a_{i-1,j} + a_{i+1,j} = a_{i,j-1} + a_{i,j+1},$$

where we assume that  $a_{n+1,j} = a_{i,n+1} = a_{0,j} = a_{i,0} = 0$ . They showed that the  $\tau$ -class is identical with the class of  $S$ -matrices.

As an immediate consequence of the cross-sum property we have that a  $\tau$ -matrix is completely defined by its first row. On the other hand, from (1) it follows that the eigenvalues  $\lambda_1, \dots, \lambda_n$  of a matrix  $A$  from the  $\tau$ -class can also be computed from the first row [25, 5]:

$$\lambda_j = \left( \sin \frac{j\pi}{n+1} \right)^{-1} \sum_{k=1}^n \sin \frac{jk\pi}{n+1} a_{1,k}. \quad (2)$$

A  $\tau$ -matrix is symmetric and persymmetric (i.e. symmetric about the cross-diagonal). Furthermore, a  $\tau$ -matrix can be written as a special sum of a Toeplitz and a Hankel matrix. Indeed, using (1) we can write an element  $a_{jk}$  from a  $\tau$ -matrix  $A$  as

$$a_{jk} = \frac{2}{n+1} \sum_{t=1}^n \sin \frac{jt\pi}{n+1} \sin \frac{kt\pi}{n+1} \lambda_t.$$

By using trigonometric formulas, it can be seen that [25, 20]

$$A = \begin{bmatrix} c_0 & c_1 & \cdots & c_{n-1} \\ c_1 & c_0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & c_1 \\ c_{n-1} & \cdots & c_1 & c_0 \end{bmatrix} = \begin{bmatrix} c_2 & c_3 & \cdots & c_{n+1} \\ c_3 & c_4 & \ddots & \vdots \\ \vdots & \ddots & & c_3 \\ c_{m_k+1} & \cdots & c_3 & c_2 \end{bmatrix}, \quad (3)$$

where the elements  $c_r$  can be found from the eigenvalues as

$$c_r = \frac{1}{n+1} \sum_{t=1}^n \cos \frac{rt\pi}{n+1} \lambda_t. \quad (4)$$

The same result can be found as a consequence of the bases that can be derived for the  $\tau$ -class [1, 5, 20]. Finally, we remark that computations involving  $\tau$ -matrices, like computing eigenvalues, matrix-vector products or solving linear systems of equations, can be computed very fast in  $O(n \log n)$  operations using the fast sine transform [27, 29] if  $n+1$  is a power of 2 or has at least small prime factors.

### 3. Correction method

In this section we will use the fact that a symmetric band Toeplitz matrix can be written as the sum of a  $\tau$ -matrix and a low rank matrix to solve a linear system of equations, following a method similar to Jain [21] and Linzer [24] who worked with circulant matrices.

We want to solve a linear system of equations  $T\mathbf{x} = \mathbf{b}$ , where  $T$  is a symmetric band Toeplitz matrix, this is  $T = (t_{i,j})$  with  $t_{i,j} = t_{|i-j|}$  and  $t_k = 0$  if  $k > p$ .

Consider the  $\tau$ -matrix  $M$  such that the first row of  $M$  is given by

$$[ t_0 - t_2 \quad t_1 - t_3 \quad \cdots \quad t_{p-2} - t_p \quad t_{p-1} \quad t_p \quad 0 \quad \cdots \quad 0 ],$$

then the matrix  $T$  can be written as  $T = M - P$  where  $P$  is the low rank matrix

$$P = \begin{bmatrix} F & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & JFJ \end{bmatrix},$$

with  $F$  and  $J$  the matrices

$$F = \begin{bmatrix} -t_2 & -t_3 & \cdots & -t_p \\ -t_3 & & & -t_p \\ \vdots & \ddots & & \\ -t_p & & & \end{bmatrix}, \quad J = \begin{bmatrix} & & & 1 \\ & & & \\ & & 1 & \\ & \ddots & & \\ 1 & & & \end{bmatrix}.$$

(see e.g. [1, 5]). Using (2) it can be easily seen that the eigenvalues  $\lambda_k$  of  $M$  are

$$\lambda_j = t_0 + 2 \sum_{k=1}^p t_k \cos \frac{jk\pi}{n+1}, \quad j = 1, \dots, n. \quad (5)$$

We can rewrite the system  $T\mathbf{x} = \mathbf{b}$  as

$$\mathbf{x} = M^{-1}(P\mathbf{x}) + M^{-1}\mathbf{b}, \quad (6)$$

so we can solve the system via two systems with the  $\tau$ -matrix  $M$  if we can deduce  $P\mathbf{x}$ . We partition the vector  $\mathbf{x}$  as  $\mathbf{x}^T = [\mathbf{x}_i^T \quad \mathbf{x}_m^T \quad \mathbf{x}_f^T]$ , where  $\mathbf{x}_i$  and  $\mathbf{x}_f$  are the  $p-1$  *initial* and *final* values, respectively. We partition the matrix  $B := M^{-1}$  in the same way as  $\mathbf{x}$  and because  $M^{-1}$  also belongs to the  $\tau$ -class, it can be written as

$$M^{-1} = \begin{bmatrix} B_{ii} & B_{im} & B_{if} \\ B_{im}^T & B_{mm} & JB_{im}^T J \\ B_{if}^T & JB_{im} J & JB_{ii} J \end{bmatrix}.$$

Remark that

$$P\mathbf{x} = \begin{bmatrix} F\mathbf{x}_i \\ \mathbf{0} \\ JFJ\mathbf{x}_f \end{bmatrix},$$

so we only have to deduce  $\mathbf{x}_i$  and  $\mathbf{x}_f$ . Further we denote  $\mathbf{z} := M^{-1}\mathbf{b}$  and we partition it in the same way as  $\mathbf{x}$ . From (6) it is easy to obtain

$$\begin{bmatrix} I - B_{ii}F & -B_{if}JFJ \\ -JB_{if}JF & J(I - B_{ii}F)J \end{bmatrix} \begin{bmatrix} \mathbf{x}_i \\ \mathbf{x}_f \end{bmatrix} = \begin{bmatrix} \mathbf{z}_i \\ \mathbf{z}_f \end{bmatrix}, \quad (7)$$

so we can calculate  $\mathbf{x}_i, \mathbf{x}_f$  from this system. We denote the matrix of the system with  $R$ . Since  $B$  is a  $\tau$ -matrix,  $B$  can be written as the sum of a Toeplitz and a Hankel matrix by formula (3), where the elements  $c_j$  in this case must be computed from the eigenvalues of  $B$  by formula (4):

$$c_j = \frac{1}{n+1} \sum_{k=1}^n \lambda_k^{-1} \cos \frac{jk\pi}{n+1}, \quad j = 0, 1, \dots, n+1. \quad (8)$$

Once the elements  $c_j$  are known, the matrix  $R$  can be easily computed. The next proposition shows that  $R$  is well conditioned if  $T$  and  $M$  are well conditioned.

PROPOSITION 1.

$$\kappa_F(R) \leq \kappa_F(T)\kappa_F(M),$$

where  $\kappa_F(\cdot)$  denotes the condition number with respect to the Frobenius norm.

The proof is very similar to Linzer [24], where  $T$  is corrected by a circulant matrix, and will be omitted here.

The matrix  $M$  though can be ill conditioned even if  $T$  is well conditioned (and vice versa), e.g. consider the matrix

$$T(\epsilon) = \begin{bmatrix} 1 + \epsilon & 0 & 1/2 & 0 & 0 \\ 0 & 1 + \epsilon & 0 & 1/2 & 0 \\ 1/2 & 0 & 1 + \epsilon & 0 & 1/2 \\ 0 & 1/2 & 0 & 1 + \epsilon & 0 \\ 0 & 0 & 1/2 & 0 & 1 + \epsilon \end{bmatrix}.$$

For small  $\epsilon$ ,  $T(\epsilon)$  is well conditioned :  $\kappa(\epsilon) < 6$ . On the other hand, the matrix  $M$ :

$$M = \begin{bmatrix} 1/2 + \epsilon & 0 & 1/2 & 0 & 0 \\ 0 & 1 + \epsilon & 0 & 1/2 & 0 \\ 1/2 & 0 & 1 + \epsilon & 0 & 1/2 \\ 0 & 1/2 & 0 & 1 + \epsilon & 0 \\ 0 & 0 & 1/2 & 0 & 1/2 + \epsilon \end{bmatrix},$$

has eigenvalues

$$\lambda_j = 1 + \epsilon + \cos \frac{2j\pi}{6}, \quad j = 1, 2, \dots, 5,$$

leading to  $\kappa_2(M) = \frac{1.5+\epsilon}{\epsilon}$ , so  $M$  is ill conditioned for  $\epsilon$  small. Notice that  $M$  even becomes singular for  $\epsilon = 0$ , while  $T(0)$  is nonsingular.

Under certain conditions, we can say something about the condition of  $M$ . We will first review some properties of a sequence of nested symmetric Toeplitz matrices  $(A_n)_{n=1}^{\infty}$ . This sequence belongs to the Wiener class if for the sequence  $(a_n)_{n=0}^{\infty}$  of elements on the first row holds  $\sum_{k=0}^{\infty} |a_k| < \infty$ . We can then define the generating function  $f$  as  $f(x) = a_0 + 2 \sum_{k=1}^{\infty} a_k \cos(kx)$  for  $x \in [0, \pi]$ . On the other hand, the generating function determines completely the sequence of Toeplitz matrices via the relation

$$a_k = \frac{1}{\pi} \int_0^{\pi} f(x) \cos(kx) dx.$$

The generating function can be used to study the spectrum of the matrices  $A_n$ . We recall from the work of Szegő [6, 13] that the spectrum of every matrix  $A_n$  lies in the interval  $[f_{\min}, f_{\max}]$ , where  $f_{\min}$  and  $f_{\max}$  denote the minimum, respectively the maximum of  $f$  over  $[0, \pi]$ . Moreover, if we denote the eigenvalues of  $A_n$  as  $\lambda_1^{(n)} \leq \lambda_2^{(n)} \leq \dots \leq \lambda_n^{(n)}$ , then

$$\lim_{n \rightarrow \infty} \lambda_1^{(n)} = f_{\min}, \quad \lim_{n \rightarrow \infty} \lambda_n^{(n)} = f_{\max}.$$

If  $f$  is a positive function, then a consequence of these properties is that  $A_n$  is positive definite for all  $n$ . Moreover, we can see that  $\kappa_2(A_n) = \lambda_n^{(n)} / \lambda_1^{(n)}$  approaches  $f_{\max} / f_{\min}$  from the left as  $n \rightarrow \infty$ .

We can embed the symmetric band Toeplitz matrix  $T$  in a sequence of nested Toeplitz matrices  $(T_n)_{n=1}^{\infty}$  by adding zeros at the right and at the bottom. The generating function  $f$  belongs then trivially to the Wiener class. By (5) the eigenvalues  $\lambda_j$  of the associated  $\tau$ -matrix  $M$  satisfy

$$\lambda_j = f\left(\frac{j\pi}{n+1}\right), \quad j = 1, 2, \dots, n,$$

such that  $M$  is positive definite if  $f$  is a positive function. Moreover, we have for the condition number of  $M$ :

$$\kappa_2(M) = \max_{j=1, \dots, n} \lambda_j / \min_{j=1, \dots, n} \lambda_j \leq f_{\max} / f_{\min}.$$

Thus we have proven the following property:

**PROPOSITION 2.** *Suppose we embed the symmetric banded Toeplitz matrix  $T$  in a sequence of nested banded Toeplitz matrices  $(T_n)_{n=1}^{\infty}$  by adding zeros at the right and at the bottom, then the associated matrix  $M$  is positive definite if all the matrices  $T_n$  are positive definite. Moreover,  $M$  is well conditioned if all the matrices  $T_n$  are well conditioned.*

Notice that we have given a much simpler proof for the first part of the theorem than Boman and Koltracht [5].

We can simplify system (7) of order  $2p-2$  to two systems of order  $p-1$  by multiplying the second equation by  $J$  and, on the one hand adding, on the other hand subtracting the two equations, which gives

$$\begin{cases} (I - (B_{ii} + B_{if}J)F)(\mathbf{x}_i + J\mathbf{x}_f) &= \mathbf{z}_i + J\mathbf{z}_f \\ (I - (B_{ii} - B_{if}J)F)(\mathbf{x}_i - J\mathbf{x}_f) &= \mathbf{z}_i - J\mathbf{z}_f \end{cases} \quad (9)$$

This is a system in the unknowns  $\mathbf{x}_i + J\mathbf{x}_f$  and  $\mathbf{x}_i - J\mathbf{x}_f$  from which we can easily calculate  $\mathbf{x}_i$  and  $\mathbf{x}_f$ . Moreover, since system (9) can be deduced from system (7) by an orthogonal transformation, the condition number does not change. We can solve the system by simple Gaussian elimination in  $O(p^3)$  operations. In the next section however we will see that we can also use displacement theory to solve the system in  $O(p^2)$ .

We can now describe the full algorithm.

**ALGORITHM 1 (CORRECTION METHOD)** *Solve the system  $T\mathbf{x} = \mathbf{b}$  where  $T$  is a symmetric band Toeplitz matrix of order  $n$  with bandwidth  $p$ .*

1. *Compute the eigenvalues of the  $\tau$ -matrix  $M$  by (5)*
2. *Compute  $\mathbf{z} = M^{-1}\mathbf{b}$*
3. *Compute the elements  $c_j$  of the matrix  $B := M^{-1}$  by (8)*
4. *Compute  $\mathbf{x}_i$  and  $\mathbf{x}_f$  by solving system (9)*
5. *Compute  $\mathbf{y} = M^{-1}(P\mathbf{x})$*
6.  *$\mathbf{x} = \mathbf{y} + \mathbf{z}$*

We compute the asymptotic operation count for the algorithm. The first and third step can be computed in  $2.5n \log n$  flops by using a fast cosine transform. The  $\tau$ -systems in step 2 and 5 can be computed in  $5n \log n$  flops using a fast sine transform. However, this performance for the sine and cosine transform can only be reached if  $n+1$  is a power of 2 or has at least small prime factors. For small  $p$  we can reduce the computational work of step 5. Indeed, since  $\mathbf{x}_i$  and  $\mathbf{x}_f$  are already computed in step 4, we only have to compute  $\mathbf{x}_m$ . From (6) we have

$$\mathbf{x}_m = \mathbf{z}_m + B_{im}^T F \mathbf{x}_i + J B_{im}^T F J \mathbf{x}_f,$$

where  $B_{im}$  can be written as a Toeplitz-plus-Hankel matrix with the elements  $c_j$ . The two systems of order  $p-1$  in step 4 can be computed in  $O(p^3)$  if we use classical Gaussian elimination. However, for  $p$  large this



Since  $A_1$  is the product of the Toeplitz-plus-Hankel matrix  $T_1 + H_1$  and the triangular Hankel matrix  $F$ , it seems obvious to work with the displacement operator  $\nabla_{\{Y_{00}, Y_{11}\}}$ . It is easy to show that

$$\nabla_{\{Y_{00}, Y_{11}\}}(A_1) = Y_{00} - Y_{11} - \nabla_{\{Y_{00}, Y_{00}\}}(T_1 + H_1) F - (T_1 + H_1) \nabla_{\{Y_{00}, Y_{11}\}}(F). \quad (10)$$

In general, a Toeplitz-plus-Hankel matrix  $T + H$  of order  $m$  with  $T = (t_{i-j})_{i,j=0}^{m-1}$  and  $H = (h_{i+j})_{i,j=0}^{m-1}$  has  $\{Y_{00}, Y_{00}\}$ -displacement rank 4 and we can immediately derive the generators. Indeed, from [12, 16] we have

$$\nabla_{\{Y_{00}, Y_{00}\}}(T + H) = \sum_{j=1}^4 \mathbf{g}_j \mathbf{b}_j^T,$$

where

$$\begin{aligned} \mathbf{g}_1 = \mathbf{b}_3 = \mathbf{e}_0 &= [1 \ 0 \ \cdots \ 0]^T, & \mathbf{g}_2 = \mathbf{b}_4 = \mathbf{e}_{m-1} &= [0 \ 0 \ \cdots \ 1]^T, \\ \mathbf{g}_3 &= (t_{i+1} + h_{i-1})_{i=0}^{m-1}, & \mathbf{g}_4 &= (t_{i-m} + h_{m+i})_{i=0}^{m-1} \\ \mathbf{b}_1 &= -(t_{-(i+1)} + h_{i-1})_{i=0}^{m-1}, & \mathbf{b}_2 &= -(t_{m-i} + h_{m+i})_{i=0}^{m-1} \end{aligned} \quad (11)$$

and  $h_{-1}, h_{2m-1}, t_m, t_{-m}$  are arbitrary real numbers. Applied to the matrices  $T_1$  and  $H_1$  however, gives

$$\mathbf{g}_1 = \mathbf{b}_3 = \mathbf{e}_0, \quad \mathbf{g}_2 = \mathbf{b}_4 = \mathbf{e}_{p-2}, \quad \mathbf{g}_3 = \mathbf{b}_1 = \mathbf{0},$$

$$\mathbf{g}_4 = -\mathbf{b}_2 = (c_{p-1-i} + c_{n-p-i} - c_{n-p+2+i} - c_{p+i+1})_{i=0}^{p-2},$$

leading to

$$\nabla_{\{Y_{00}, Y_{00}\}}(T_1 + H_1) = \mathbf{g}_1 \mathbf{e}_{p-2}^T - \mathbf{e}_{p-2} \mathbf{g}_1^T,$$

so  $T_1 + H_1$  has only  $\{Y_{00}, Y_{00}\}$ -displacement rank 2.

On the other hand, it is easy to show that for the triangular Hankel matrix  $F$  holds

$$\nabla_{\{Y_{00}, Y_{11}\}}(F) = \mathbf{e}_0 \mathbf{f}_1^T + \mathbf{f}_2 \mathbf{e}_0^T,$$

where  $\mathbf{f}_1 = [\alpha, t_2, \dots, t_{p-2}, t_{p-1} + t_p]^T$ ,  $\mathbf{f}_2 = [-\alpha + t_2, -t_2 + t_3, \dots, -t_{p-1} + t_p]^T$  and  $\alpha$  an arbitrary real number. If we use that  $Y_{00} - Y_{11} = -\mathbf{e}_0 \mathbf{e}_0^T - \mathbf{e}_{p-2} \mathbf{e}_{p-2}^T$  and  $\mathbf{e}_{p-2}^T F = t_p \mathbf{e}_0^T$ , then (10) becomes

$$\begin{aligned} \nabla_{\{Y_{00}, Y_{11}\}}(A_1) &= \mathbf{e}_{p-2} (F^T \mathbf{g}_1 - \mathbf{e}_{p-2})^T + ((T_1 + H_1) \mathbf{e}_0) (-\mathbf{f}_1)^T \\ &\quad + ((T_1 + H_1) \mathbf{f}_2 + t_p \mathbf{g}_1 + \mathbf{e}_0) (-\mathbf{e}_0)^T. \end{aligned} \quad (12)$$

This proves that  $A_1$  has  $\{Y_{00}, Y_{11}\}$ -displacement rank 3 and we immediately find the generators. A similar derivation can of course be done for  $A_2$ .

Now that we know the displacement rank and the generators, we have to select a fast method to solve the system (9). Since we do not know

in advance if the matrix  $A_1$  is positive definite or strongly regular, a possible method is to transform the generalized Toeplitz-plus-Hankel matrix to a generalized Cauchy matrix, because then we can use the fast GEPP-algorithm (fast Gaussian elimination with partial pivoting). This transformation technique and the fast GEPP-algorithm can be found in Gohberg, Kailath and Olshevsky [11]. For a detailed elaboration of the application of the technique in this case and the full algorithms we refer to [18].

## 5. Embedding method

The correction method from section 3 has the drawback that it is only fast when  $n + 1$  is a power of 2, or has at least small prime factors, because of the sine transform of order  $n$ . In this section we will embed the Toeplitz matrix in a larger  $\tau$ -matrix  $M$  of order  $m$ , where  $m$  is chosen such that  $m + 1$  has small prime factors. The method is similar to Jain [21] who embedded the Toeplitz matrix in a circulant matrix.

So again suppose that we want to solve the system  $T\mathbf{x} = \mathbf{b}$  with  $T$  a symmetric band Toeplitz matrix of order  $n$  and bandwidth  $p$ . We choose  $m$  such that  $m \geq n + p - 1$  and  $m + 1$  has small prime factors. Let  $M$  be the  $\tau$ -matrix with first row

$$[ t_0 - t_2 \quad t_1 - t_3 \quad \cdots \quad t_{p-2} - t_p \quad t_{p-1} \quad t_p \quad 0 \quad \cdots \quad 0 ].$$

Notice that  $M$  has the same shape as in section 3, but now of order  $m$  instead of  $n$ . It can be easily seen that  $T$  is embedded in  $M$ , by other words we can write  $M$  as

$$M = \begin{bmatrix} M_{ii} & M_{im} & 0 \\ M_{mi} & T & M_{mf} \\ 0 & M_{fm} & M_{ff} \end{bmatrix}.$$

Suppose that  $r$  is the number of rows added at the the top and  $s$  at the bottom of  $T$ . We extend the system  $T\mathbf{x} = \mathbf{b}$  to

$$\begin{bmatrix} M_{ii} & M_{im} & 0 \\ M_{mi} & T & M_{mf} \\ 0 & M_{fm} & M_{ff} \end{bmatrix} \begin{bmatrix} \mathbf{0} \\ \mathbf{x} \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{b}_i \\ \mathbf{b} \\ \mathbf{b}_f \end{bmatrix}, \quad (13)$$

where  $\mathbf{b}_i$  and  $\mathbf{b}_f$  have to be defined such that the first  $r$  and last  $s$  components of the solution vector are equal to zero. If we partition  $B := M^{-1}$  the same way as  $M$ , we get

$$\begin{bmatrix} B_{ii} & B_{im} & B_{if} \\ B_{mi} & B_{mm} & B_{mf} \\ B_{fi} & B_{fm} & B_{ff} \end{bmatrix} \begin{bmatrix} \mathbf{b}_i \\ \mathbf{b} \\ \mathbf{b}_f \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{x} \\ \mathbf{0} \end{bmatrix}. \quad (14)$$

Therefore we can find  $\mathbf{b}_i$  and  $\mathbf{b}_f$  from the system

$$\begin{cases} B_{ii}\mathbf{b}_i + B_{if}\mathbf{b}_f &= -B_{im}\mathbf{b} \\ B_{fi}\mathbf{b}_i + B_{ff}\mathbf{b}_f &= -B_{fm}\mathbf{b} \end{cases} . \quad (15)$$

As in section 3 we can split the latter system in two smaller systems. Dependent on the values of  $m$  and  $n$ , we can always choose  $s$  and  $r$  such that  $s = r$  or  $s = r + 1$ . In the first case,  $B_{fi} = JB_{ff}J$ ,  $B_{ff} = JB_{ii}J$  and  $B_{fm} = JB_{im}J$  because of symmetry, so we can analogously to section 3 rewrite the system as two systems of order  $r$ :

$$\begin{cases} (B_{ii} + B_{if}J)(\mathbf{b}_i + J\mathbf{b}_f) &= -B_{im}(\mathbf{b} + J\mathbf{b}) \\ (B_{ii} - B_{if}J)(\mathbf{b}_i - J\mathbf{b}_f) &= -B_{im}(\mathbf{b} - J\mathbf{b}) \end{cases} .$$

If  $s = r + 1$ , we have to partition (13) otherwise, taking into account the symmetry in the matrix  $B$ .

$$\begin{bmatrix} B_{ii} & B_{im} & \alpha & B_{if} \\ B_{mi} & B_{mm} & \gamma & B_{mf} \\ \alpha^T & \gamma^T & \omega & \beta^T \\ JB_{if}J & B_{fm} & \beta & JB_{ii}J \end{bmatrix} \begin{bmatrix} \mathbf{b}_i \\ \mathbf{b} \\ b_s \\ \mathbf{b}_t \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \mathbf{x} \\ 0 \\ \mathbf{0} \end{bmatrix} .$$

Here  $B_{ii}$  and  $B_{if}$  are  $(r \times r)$ -matrices,  $\alpha$ ,  $\beta$ ,  $\mathbf{b}_i$  and  $\mathbf{b}_f$  vectors of order  $r$  and  $\omega$  and  $b_s$  scalars. We have to define the unknowns  $\mathbf{b}_i$ ,  $\mathbf{b}_f$  and  $b_s$  from the first, the third and the fourth equation. If we multiply the first equation by  $J$  and replace the first and last equation by, on the one hand the sum and on the other hand the difference of the two equations, we get the equivalent system:

$$\begin{cases} (B_{ii} + B_{if}J)(\mathbf{b}_i + J\mathbf{b}_f) + (\alpha + J\beta)b_s &= -(B_{im} + JB_{fm})\mathbf{b} \\ (\alpha + J\beta)^T(\mathbf{b}_i + J\mathbf{b}_f) + (\alpha - J\beta)^T(\mathbf{b}_i - J\mathbf{b}_f) + 2\omega b_s &= -2\gamma^T\mathbf{b} \\ (B_{ii} - B_{if}J)(\mathbf{b}_i - J\mathbf{b}_f) + (\alpha - J\beta)b_s &= -(B_{im} - JB_{fm})\mathbf{b} \end{cases} .$$

We remark that the new system can be derived from the old by an orthogonal transformation. Let us for simplicity introduce the notations  $A_1 := B_{ii} + B_{if}J$ ,  $A_2 := B_{ii} - B_{if}J$ ,  $\mathbf{f}_1 := -(B_{im} + JB_{fm})\mathbf{b}$ ,  $\mathbf{f}_2 := -(B_{im} - JB_{fm})\mathbf{b}$ ,  $\mathbf{a}_1 := \alpha + J\beta$ ,  $\mathbf{a}_2 := \alpha - J\beta^T$  and the new unknowns  $\mathbf{y}_1 := \mathbf{b}_i + J\mathbf{b}_f$ ,  $\mathbf{y}_2 := \mathbf{b}_i - J\mathbf{b}_f$ , then we can rewrite the first and last equation as

$$\begin{cases} \mathbf{y}_1 = A_1^{-1}\mathbf{f}_1 - b_s A_1^{-1}\mathbf{a}_1 \\ \mathbf{y}_2 = A_2^{-1}\mathbf{f}_2 - b_s A_2^{-1}\mathbf{a}_2 \end{cases} .$$

If we insert this in the second equation we get

$$(2\omega - \mathbf{a}_1^T A_1^{-1}\mathbf{a}_1 - \mathbf{a}_2^T A_2^{-1}\mathbf{a}_2) b_s = -2\gamma^T\mathbf{b} - \mathbf{a}_1^T A_1^{-1}\mathbf{f}_1 - \mathbf{a}_2^T A_2^{-1}\mathbf{f}_2,$$

so we can compute  $\mathbf{b}_i$ ,  $\mathbf{b}_f$  and  $b_s$  by solving two linear systems of order  $r$ , each with two right-hand sides, and some scalar products.

In both cases we have to deal with matrices  $A_1 := B_{ii} + B_{if}J$  and  $A_2 := B_{ii} - B_{if}J$  of order  $r$ , which are Toeplitz-plus-Hankel matrices, since  $B_{ii}$  and  $B_{if}$  are of this type. Therefore we can use fast algorithms based on displacement theory to solve the system, as explained in section 4. We could again work with the displacement operator  $\nabla_{\{Y_{00}, Y_{11}\}}$ . One can prove that  $A_1$  has  $\{Y_{00}, Y_{11}\}$ -displacement rank 3. However, in section 4 we saw that the  $\{Y_{00}, Y_{00}\}$ -displacement rank of  $A_1$  was only 2. Therefore it seems better to work with the displacement operator  $\nabla_{\{Y_{00}, Y_{00}\}}$ . On the other hand, the  $\{Y_{00}, Y_{00}\}$ -displacement of  $A_1$  does not completely determine the matrix  $A_1$  and this gives extra difficulties. We can still transform the matrix to a generalized Cauchy matrix, but the diagonal of the generalized Cauchy matrix can not be determined from the generators of the generalized Cauchy matrix, and will have to be stored and computed separately. The LU-Cauchy algorithm from Heinig and Bojanczyk [14, 15] computes an LU-decomposition with partial pivoting of the generalized Cauchy matrix in this case. It is a generalization of the fast GEPP algorithm from Gohberg, Kailath and Olshevsky [11]. However, since  $A_1$  is symmetric also the transformed generalized Cauchy matrix will be symmetric and the LU-decomposition does not take the symmetry into account. Therefore, it is better to look for a decomposition that preserves the symmetry. For a general matrix  $A$ , the Bunch-Kaufman-Parlett algorithm [7] computes a factorization  $P^T A P = M D M^T$ , with  $P$  a permutation matrix,  $M$  a lower triangular matrix with ones on the diagonal and  $D$  a block diagonal matrix with diagonal blocks of order 1 or 2. The algorithm BKP-Cauchy from Heinig and Bojanczyk [15] is an adaptation of this algorithm for generalized Cauchy matrices.

It is not always necessary to transform the matrix to a generalized Cauchy matrix. When  $M$  is positive definite, which can be easily seen from its calculated eigenvalues, also the matrix  $B = M^{-1}$  will be positive definite and therefore also the matrices  $B_{ii} + B_{if}J$  and  $B_{ii} - B_{if}J$ . Remark that also  $T$  will be positive definite because of the embedding of  $T$  in  $M$ . In this case we can compute a Cholesky decomposition without the need for pivoting. One can find several methods for solving strongly non-singular or positive definite Toeplitz-plus-Hankel systems in the literature, see e.g. [12, 16, 26]. We will ground on [26] and the generalized Schur algorithm from [22]. We start with the displacement operator

$$\nabla_{\{Z, I+Z^2, I+Z^2, Z\}}(A) = ZA(I + Z^2)^T - (I + Z^2)AZ^T$$

where  $Z$  is the lower triangular matrix

$$Z = \begin{bmatrix} 0 & & & & \\ 1 & 0 & & & \\ & \ddots & \ddots & & \\ & & & 1 & 0 \end{bmatrix}.$$

A Toeplitz-plus-Hankel matrix has in general displacement rank 4 with respect to this displacement operator. However, for the matrix  $A = B_{ii} + B_{if}J$  it can be easily shown that<sup>‡</sup>

$$ZA(I + Z^2)^T - (I + Z^2)AZ^T = \mathbf{g}\mathbf{e}_0^T - \mathbf{e}_0\mathbf{g}^T = G\tilde{J}G^T, \quad (16)$$

where  $\mathbf{g} = (c_{i-1} - c_{i+1} + c_{m-i} - c_{m+2-i})_{i=0}^{r-1}$ ,  $G = [\mathbf{e}_0 \ \mathbf{g}]$ ,

$$\tilde{J} = \begin{bmatrix} & -1 \\ 1 & \end{bmatrix},$$

and  $c_{-1}$ ,  $c_{m+2}$  arbitrary numbers. This proves that  $A$  has displacement rank 2. The last row of  $A$  can not be derived from the generators and will have to be stored and calculated separately. We will denote it by  $\mathbf{h}_0$  and we have

$$\mathbf{h}_0 = (c_{r-1-i} - c_{m+2-r+i} + c_{m-r-i} - c_{r+1+i})_{i=0}^{r-1}.$$

From the generators and the last row, we want to calculate a Cholesky-decomposition or equivalently a decomposition of the form  $A_1 = \tilde{L}\tilde{D}^{-1}\tilde{L}^T$  with  $\tilde{D} = \text{diag}\{d_0, d_1, \dots\}$  and  $L$  a lower triangular matrix with  $d_0, d_1, \dots$  on the diagonal. The first column of  $L$  will be equal to the first column of  $A_1$ . If we denote it by  $\mathbf{l}_0$  we have from (16) that  $Z\mathbf{l}_0 = G\tilde{J}\mathbf{g}_0$ . This determines  $\mathbf{l}_0$  except for the last element, which can be calculated from  $\mathbf{h}_0$ . Remark that  $d_0$  is the first element of  $\mathbf{l}_0$  and that

$$A - \mathbf{l}_0 d_0^{-1} \mathbf{l}_0^T = \begin{bmatrix} 0 & 0 \\ 0 & A_1 \end{bmatrix}. \quad (17)$$

The matrix  $A_1$  is called the Schur complement of  $d_0$  in  $A$ . For the second column of  $L$  we have to proceed with the Schur complement. If we denote the matrix in the right-hand side of (17) by  $\tilde{A}_1$ , one can easily proof that

$$Z\tilde{A}_1(I + Z^2)^T - (I + Z^2)\tilde{A}_1Z^T = \tilde{G}_1\tilde{J}\tilde{G}_1^T,$$

with

$$\tilde{G}_1 = G - (I + Z^2)\mathbf{l}_0 d_0^{-1} \mathbf{g}_0 =: \begin{bmatrix} 0 \\ G_1 \end{bmatrix}.$$

---

<sup>‡</sup>For simplicity we have omitted the subindex of  $A_1$ .

In other words, the Schur complement  $A_1$  has the same displacement rank as  $A$  and we can easily compute its generators. This is an essential property in displacement theory. Finally, we have to compute the last row  $\mathbf{h}_1$  of the Schur complement  $A_1$  as

$$\tilde{\mathbf{h}}_1 = [0 \ \mathbf{h}_1] = \mathbf{h}_0 - h_{00}d_0^{-1}\mathbf{1}_0^T,$$

with  $h_{00}$  the first element of  $\mathbf{h}_0$ . Now that we know the generators and the last row of the Schur complement, we can compute the second column of  $\tilde{L}$ , similarly as the first column. The third column can be computed if we proceed with the Schur complement of  $d_1$  in  $A_1$  and so on.

The algorithm for the embedding method is very similar to the first method. Remark that the right-hand side of (15) can be computed as  $-M^{-1}[\mathbf{0}^T \ \mathbf{b}^T \ \mathbf{0}^T]^T$ .

ALGORITHM 2 (EMBEDDING METHOD) *Solve the system  $T\mathbf{x} = \mathbf{b}$  where  $T$  is a symmetric band Toeplitz matrix of order  $n$  and with bandwidth  $p$ .*

1. Choose  $m \geq n + p - 1$  such that  $m + 1$  has small prime factors. Set  $r = \lceil (m - n)/2 \rceil$ .
2. Compute the eigenvalues of the matrix  $M$  by formula (5), replacing  $n$  by  $m$ .
3. Compute  $M^{-1}[\mathbf{0}^T \ \mathbf{b}^T \ \mathbf{0}^T]^T$ .
4. Compute the elements  $c_j$  for the  $\tau$ -matrix  $B$  by formula (8), replacing  $n$  by  $m$ .
5. Compute  $\mathbf{b}_i$  and  $\mathbf{b}_f$  by solving system (15).
6. Solve the extended system (13).

Similar to Algorithm 1 it is easy to compute that the asymptotic operation count of the algorithm above is  $15m \log m + O(r^3)$  or  $15m \log m + O(r^2)$  if we use displacement theory. Notice that this performance holds for every  $n$ , since  $m + 1$  is chosen to have small prime factors, and this is the big advantage of the embedding method compared to the correction method of section 3, which is only fast if  $n + 1$  has small prime factors. The disadvantage of the method is that  $r$ , the order of the smaller systems, can become rather large. In the optimal case is  $m = n + p - 1$  and is  $r$  equal to  $(p - 1)/2$ , which is smaller than the order of the systems in the correction method of section 3, but if  $m$  differs too much from  $n$ , the order of the smaller systems can become much greater than  $p$ , leading to a loss of performance of the method. In this case it is certainly recommended to use displacement theory to solve the smaller systems. Finally, we remark that a detailed version of the algorithm, including the use of displacement theory to solve the smaller systems, can be found in [18].

## 6. Numerical examples

We have implemented the methods of the previous sections in Fortran 90. We have written different versions for the correction as well as for the embedding method, dependent on how the smaller systems are solved. We have also implemented some classical methods for comparison. We enumerate the different methods, together with their asymptotic operation count. Remember that  $m$  is chosen such that  $m \geq n + p - 1$  and  $m + 1$  has small prime factors, and that  $r = \lfloor (m - n)/2 \rfloor$ .

**correction (Gauss)** Correction method. Smaller systems are solved by Gaussian elimination. The operation count is  $15n \log n + 4/3p^3$  if  $n + 1$  has small prime factors.

**correction (Cauchy)** Correction method. Smaller systems are transformed to generalized Cauchy matrices and then solved by the fast GEPP-algorithm. The operation count is  $15n \log n + O(p^2)$  if  $n + 1$  has small prime factors.

**embedding (Gauss)** Embedding method. Smaller systems are solved by Gaussian elimination. The operation count is  $15m \log m + 4/3r^3$ .

**embedding (Cauchy)** Embedding method. Smaller systems are transformed to generalized Cauchy matrices and then solved by the BKP-Cauchy algorithm. The operation count is  $15m \log m + O(r^2)$ .

**embedding (Schur)** Embedding method. Smaller systems are solved by a generalized Schur algorithm. The matrix  $M$  has to be positive definite. The operation count is  $15m \log m + O(r^2)$ , but the constant before  $r^2$  will be much smaller compared to *embedding (Cauchy)*.

**Linzer** The algorithm CIRCDE3 from Linzer. This is an analogous method as the correction method of section 3, but by making use of circulant matrices. The smaller systems are solved by classical Gaussian elimination. The operation count is  $15n \log n + 4/3p^3$  if  $n$  has small prime factors.

**Gauss for band** A classical Gauss algorithm for banded matrices. The operation count is  $2np^2$ .

**Schur for band** The classical Schur algorithm for Toeplitz matrices. The method is adapted to banded Toeplitz matrices and we assume  $T$  positive definite. The operation count is  $O(pn)$ .

**Levinson** The classical Levinson algorithm. The operation count is  $4n^2$ .

The programs were executed on an IBM SP2 machine in double precision. The FFTs were calculated via FFTPACK. We will compare the new methods among each other and to the classical methods.

For our first example we have generated symmetric positive definite Toeplitz matrices of order  $n = 32767$  and with varying bandwidth  $p$ . Remark that  $n + 1 = 2^{14}$ . The right hand sides were calculated such that the exact solution is  $[1 \cdots 1]^T$ . In figure 1 we compare the correction and the

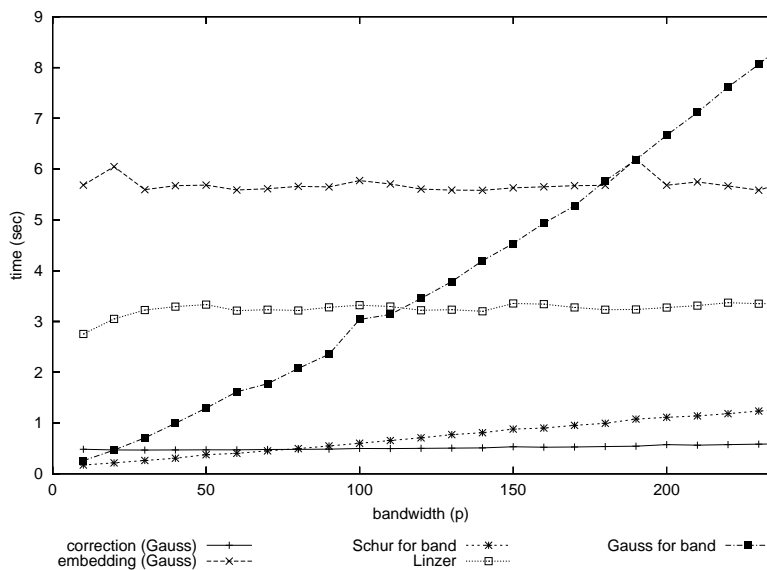


Figure 1: Execution times (in sec) for positive definite matrices of order  $n = 32767$  in function of the bandwidth  $p$

embedding method with the classical methods. The smaller systems in the new methods are solved by classical Gaussian elimination. The execution times for the Levinson algorithm were too high and were not inserted in the figure. We remark that the Gauss and the Schur algorithm for banded matrices are better for small bandwidth  $p$ , but the execution times grow rapidly for increasing  $p$ , such that from more or less  $p = 80$ , the correction method is definitely the fastest. Since  $n$  does not have small prime factors ( $32767 = 7 \times 31 \times 151$ ), the Linzer algorithm is much slower. It is also remarkable that the embedding method does not have a good performance in this example. The reason is that in this method we have to look for a value  $m \geq n + p - 1$  such that  $m + 1$  has small prime factors. An adequate program gives  $m = 35839$  for  $n = 32767$  and  $p$  any value between 1 and 3073.

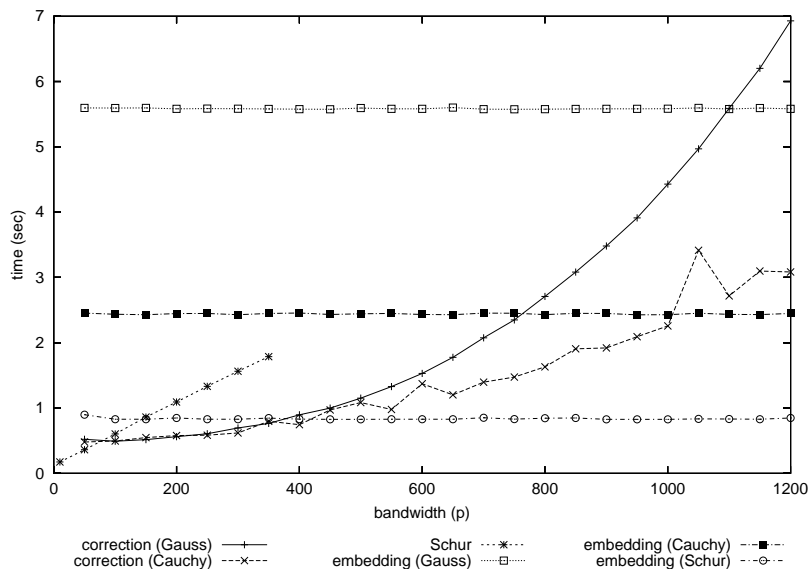


Figure 2: Execution times (in sec) for positive definite matrices of order  $n = 32767$  in function of the bandwidth  $p$  (large  $p$ )

The order of the smaller systems is by consequence  $r = \lfloor (m - n)/2 \rfloor = 1536$  for every  $p$  between 1 and 3073. In the correction method, the order of the smaller systems is only  $p - 1$ . This explains the big difference in execution time between the correction and the embedding method and this illustrates immediately a disadvantage of the embedding method. Of course, in this example it is recommended to use a fast algorithm to solve the smaller systems, as will be seen in figure 2, where we compare the different versions of the correction and the embedding method. Notice that we have taken a wider range for  $p$  compared to figure 1. For comparison we have also plotted the times for the Schur method. Since we compute an LU-factorisation of  $T$  in the Schur method, we have to store  $np$  double precision numbers, which can lead to memory problems. That is why only execution times for  $p \leq 350$  are inserted into the figure. We remark the clear difference between an  $O(p^3)$ -method and an  $O(p^2)$ -method and the profit we can make if we can use a generalized Schur algorithm in stead of transforming the matrices to Cauchy matrices.

In our second example we generate positive definite Toeplitz matrices with fixed bandwidth  $p = 100$  and for varying order  $n$  of the matrix. The execution times are plotted in figure 3. This gives a completely different

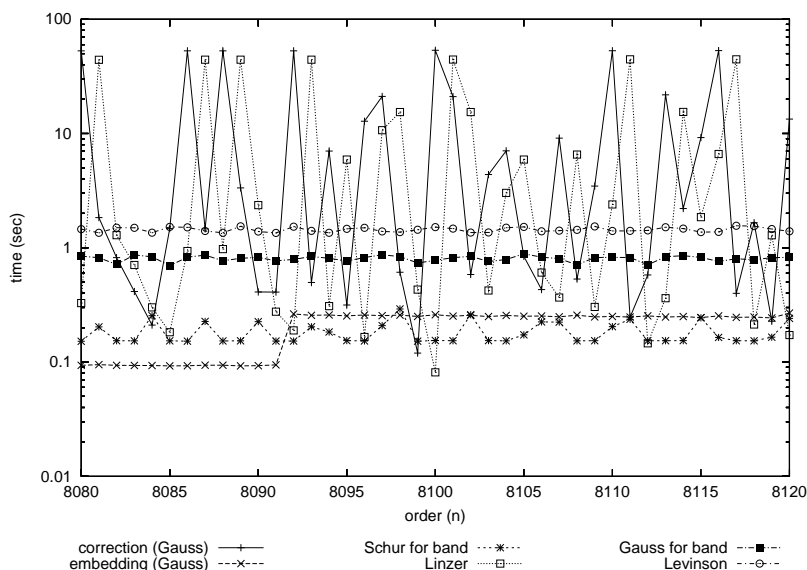


Figure 3: Execution times (in sec) for positive definite matrices of bandwidth  $p = 100$  in function of the order  $n$

picture. Remark that we have used a logarithmic scale for the execution times. The figure illustrates the big disadvantage of the correction method: it is only fast when  $n + 1$  has small prime factors. In the other cases, the performance can be disastrous, even worse than the "slow" Levinson method. The embedding method does not have this drawback. Remark however the "jump" in the execution times for the embedding method, owing to the value for  $m$ . For  $8050 \leq n \leq 8091$  and  $p = 100$  we could choose  $m = 8191$ , but for  $8092 \leq n \leq 8120$  we had to take  $m = 8959$ . This has not only an effect on the order of the sine transform, but more important also on the order of the smaller systems.

Finally we will look at the accuracy of the methods. We have generated arbitrary symmetric banded Toeplitz matrices of fixed order  $n = 32767$  and varying bandwidth. In Figure 4 we have plotted the relative maximal error  $\|\hat{\mathbf{x}} - \mathbf{x}\|_{\infty} / \|\mathbf{x}\|_{\infty}$ , where  $\hat{\mathbf{x}}$  is the computed and  $\mathbf{x}$  the exact solution of the system. We have included the correction and the embedding method and the classical Gauss method for band matrices. In the bottom figure we have plotted the condition numbers (in the 2-norm) of the banded Toeplitz matrix  $T$  and of the associated  $\tau$ -matrix  $M$  for the correction and the embedding method. Remark that there can be a big difference between

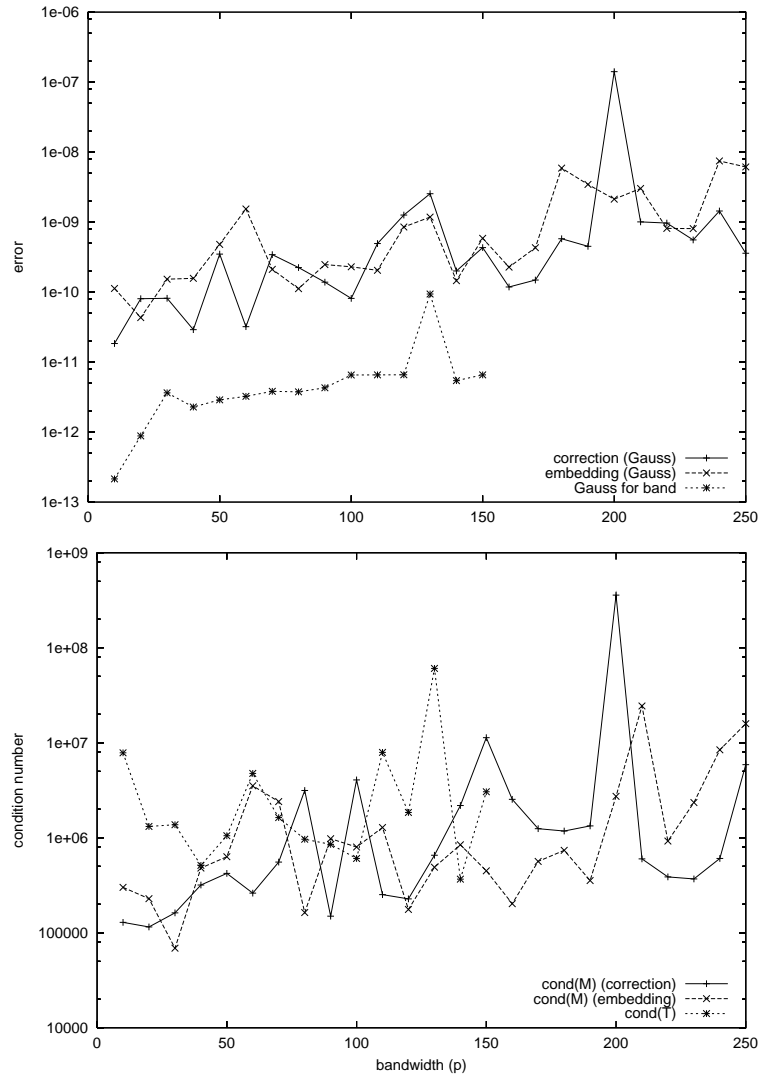


Figure 4: Maximal error and condition number versus the bandwidth  $p$  for arbitrary generated matrices of order  $n = 32767$

$\kappa(T)$  and  $\kappa(M)$ , and this can have its influence on the accuracy.

We can improve the accuracy via iterative refinement. If we look at the algorithms of the embedding and the correction method, we see that the eigenvalues, the elements  $c_j$  and the factorization of the smaller systems only have to be computed once, so the main computation work are the two  $\tau$ -systems, which can be solved via sine transforms. On the other hand, the residue can be computed via FFT (see e.g. [4]). In figure 5 we have plotted the maximal residue  $\|\mathbf{b} - T\hat{\mathbf{x}}\|_\infty / \|\mathbf{b}\|_\infty$  for the embedding method (*embedding*), the embedding method with one step iterative refinement (*embedding+1*) and the Gauss method for bandmatrices for arbitrary generated matrices of order  $n = 10000$ . We notice that one step iterative refinement is sufficient.

## REFERENCES

- 1 D. Bini and M. Capovani. Spectral and computational properties of band symmetric Toeplitz matrices. *Linear Algebra Appl.*, 52/53:99–126, 1983.
- 2 D. Bini and M. Capovani. Tensor rank and border rank of band Toeplitz matrices. *SIAM J. Comput.*, 16:252–258, 1987.
- 3 D. Bini and B. Meini. Effective methods for solving banded Toeplitz systems. *SIAM J. Matrix Anal. Appl.*, 20:700–719, 1999.
- 4 D. Bini and V. Pan. *Polynomial and matrix computations. 1: Fundamental algorithms*. Birkhäuser, Boston, 1994.
- 5 E. Boman and I. Koltracht. Fast transform based preconditioners for Toeplitz equations. *SIAM J. Matrix. Anal. Appl.*, 16:628–645, 1995.
- 6 A. Böttcher and B. Silbermann. *Introduction to large truncated Toeplitz matrices*. Springer, New York, 1999.
- 7 J. Bunch, L. Kaufman, and B. Parlett. Decomposition of a symmetric matrix. *Numer. Math.*, 27:95–109, 1976.
- 8 R. H. Chan and M. K. Ng. Sine transform based preconditioners for symmetric Toeplitz systems. *Linear Algebra Appl.*, 232:237–259, 1996.
- 9 T. Chan. An optimal circulant preconditioner for Toeplitz systems. *SIAM J. Sci. Stat. Comput.*, 9:766–771, 1988.

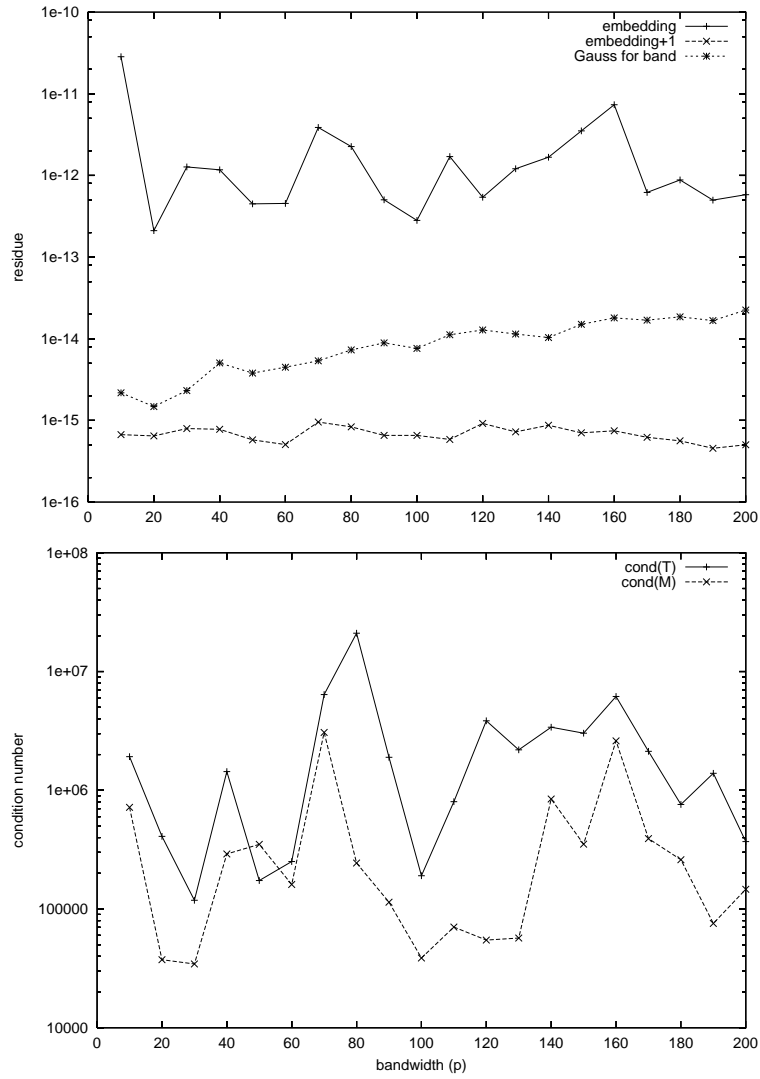


Figure 5: Maximal residue and condition number versus the bandwidth  $p$  for arbitrary generated matrices of order  $n = 10000$

- 10 F. Di Benedetto. Preconditioning of block Toeplitz matrices by sine transforms. *SIAM J. Sci. Comput.*, 18:499–515, 1997.
- 11 I. Gohberg, T. Kailath, and V. Olshevsky. Fast Gaussian elimination with partial pivoting for matrices with displacement structure. *Math. Comput.*, 64:1557–1576, 1995.
- 12 I. Gohberg and I. Koltracht. Efficient algorithm for Toeplitz plus Hankel matrices. *Integral Equations Oper. Theory*, 12:136–142, 1989.
- 13 U. Grenander and G. Szegő. *Toeplitz forms and their applications*. Chelsea, New York, NY, 2nd edition, 1984.
- 14 G. Heinig and A. Bojanczyk. Transformation techniques for Toeplitz and Toeplitz-plus-Hankel matrices. I. Transformations. *Linear Algebra Appl.*, 254:193–226, 1997.
- 15 G. Heinig and A. Bojanczyk. Transformation techniques for Toeplitz and Toeplitz-plus-Hankel matrices. II. Algorithms. *Linear Algebra Appl.*, 278:11–36, 1998.
- 16 G. Heinig, P. Jankowski, and K. Rost. Fast inversion algorithms of Toeplitz-plus-Hankel matrices. *Numer. Math.*, 52:665–682, 1988.
- 17 G. Heinig and K. Rost. *Algebraic methods for Toeplitz-like matrices and operators*, volume 13 of *Operator Theory*. Birkhäuser, Basel, 1984.
- 18 J. Hendrickx. *Veralgemeende S-matrices en hun aanwending bij de eindigedifferentiebenaderingen van differentiaalvergelijkingen en het oplossen van symmetrische band-Toeplitzstelsels*. PhD thesis, K.U. Leuven, 2000. (in Dutch).
- 19 T. Huckle. Circulant and skew-circulant matrices for solving Toeplitz matrix problems. *SIAM J. Matrix Anal. Appl.*, 13:767–777, 1992.
- 20 T. Huckle. Fast transforms for tridiagonal linear equations. *BIT*, 34:99–112, 1994.
- 21 A. K. Jain. Fast inversion of banded Toeplitz matrices by circular decompositions. *IEEE Trans. Acoust. Speech Signal Process.*, 26:121–126, 1978.
- 22 T. Kailath and A. H. Sayed. Displacement structure: theory and applications. *SIAM Rev.*, 17:297–386, 1995.
- 23 T. Kailath and A. H. Sayed. *Fast reliable algorithms for matrices with structure*. SIAM, Philadelphia, 1999.

- 24 E. Linzer. On the stability of solution methods for band Toeplitz systems. *Linear Algebra Appl.*, 170:1–32, 1992.
- 25 L. Mertens and H. Van de Vel. A special class of structured matrices constructed with the Kronecker product and its use for difference equations. *Linear Algebra Appl.*, 106:117–147, 1988.
- 26 A. Sayed, H. Lev-Ari, and T. Kailath. Fast triangular factorization of the sum of quasi-Toeplitz and quasi-Hankel matrices. *Linear Algebra Appl.*, 191:77–106, 1993.
- 27 P. N. Swarztrauber. Symmetric FFTs. *Math. Comput.*, 47:323–346, 1986.
- 28 E. Tyrtyshnikov. Optimal and super-optimal circulant preconditioners. *SIAM J. Matrix Anal. Appl.*, 13:459–473, 1992.
- 29 C. Van Loan. *Computational frameworks for the Fast Fourier Transform*. SIAM, Philadelphia, 1992.