2014

TR-2014005: Fast Approximation Algorithms for Computations with Cauchy Matrices and Extensions

Victor Y. Pan

Fast Approximation Algorithms for Computations with Cauchy Matrices and Extensions *

Victor Y. Pan[1,2],[a]

[1] Department of Mathematics and Computer Science
Lehman College of the City University of New York
Bronx, NY 10468 USA
[2] Ph.D. Programs in Mathematics and Computer Science
The Graduate Center of the City University of New York
New York, NY 10036 USA
[a] victor.pan@lehman.cuny.edu
http://comet.lehman.cuny.edu/vpan/

Abstract

The papers [MRT05], [CGS07], [XXG12], and [XXCB14] combine the techniques of the Fast Multipole Method of [GR87], [CGR98] with the transformations of matrix structures, traced back to [P90]. The resulting numerically stable algorithms approximate the solutions of Toeplitz, Hankel, Toeplitz-like, and Hankel-like linear systems of equations in nearly linear arithmetic time, versus the classical cubic time and the quadratic time of the previous advanced algorithms. We extend this progress to decrease the arithmetic time of the known numerical algorithms from quadratic to nearly linear for computations with a large class of matrices that have structure of Cauchy or Vandermonde type and for the evaluation and interpolation of polynomials and rational functions. We detail and analyze the new algorithms, and in [Pa] we extend them further.

Key words: Vandermonde matrices; Cauchy matrices; Fast Multipole Method; HSS matrices; Matrix compression; Polynomial evaluation; Rational evaluation; Interpolation

AMS Subject Classification: 12Y05, 15A04, 47A65, 65D05, 68Q25

1 Introduction

The numerically stable algorithms of [MRT05], [CGS07], [XXG12], and [XXCB14] approximate the solution of Toeplitz, Hankel, Toeplitz-like, and Hankel-like linear systems of equations in nearly linear arithmetic time versus the classical cubic time and the previous record quadratic time of [GKO95]. All five cited papers first transform the matrix structures of Toeplitz and Hankel types into the structure of Cauchy type, which is a special case of the general technique proposed in [P90]. Then [GKO95] exploits the invariance of the Cauchy matrix structure in row and column interchange, whereas the other four papers apply numerically stable FMM to operate efficiently with HSS approximation of the basic Cauchy matrix. "HSS" and "FMM" are the acronyms for "Hierarchically

*Some results of this paper have been presented at the 18th Conference of the International Linear Algebra Society (ILAS'2013), Providence, RI, 2013 and at the 15th Annual Conference on Computer Algebra in Scientific Computing (CASC 2013), Berlin, Germany, 2013, and are scheduled to be presented at the Ninth International Computer Science Symposium in Russia (CSR 2014), Moscow, Russia, June 2014. Our research has been supported by the NSF Grant CC 1116736 and the PSC CUNY Awards 64512–0042 and 65792–0043.

Semiseparable" and "Fast Multipole Method", respectively. "Historically HSS representation is just a special case of the representations commonly exploited in the FMM literature" [CDG06].

In our present paper we extend the successful algorithms of [MRT05], [CGS07], [XXG12], and [XXCB14] to computations with Cauchy and Vandermonde matrices, namely to approximation of their products by a vector and of the solution of linear systems of equations with these matrices, and further to approximate multipoint polynomial and rational evaluation and interpolation. The arithmetic time of the known numerical approximation algorithms for these tasks is quadratic (cf. [BF00] and [BEGO08]), and we decrease it to nearly linear.

As in the papers [MRT05], [CGS07], [XXG12], and [XXCB14], we approximate Cauchy matrices by HSS matrices and exploit the HSS matrix structure, and our present work can be also viewed as a specification of the FMM to an important subclass of Cauchy matrices. As in these papers our basic computational blocks are the numerically stable FFT and FMM algorithms, which have been efficiently implemented on both serial and parallel computers [GS66], [B99], [BY13], although Cauchy and particularly Vandermonde linear systems of equations as well as the equivalent problems of rational and polynomial interpolation are frequently ill conditioned and thus unfavorable to approximate numerical solution.

Unlike the cited papers we treat a large subclass of Cauchy matrices $C = (\frac{1}{s_i - t_j})_{i,j=0}^{n-1}$ (we call them CV matrices because they are linked to Vandermonde matrices via FFT-based unitary transformations) rather than just the single CV matrix involved in the fast Toeplitz solvers. For that matrix, $\{s_0, \dots, s_{n-1}\}$ is the set of the nth roots of unity, and $\{t_0, \dots, t_{n-1}\}$ is the set of the other $(2n)$-th roots of unity, but for a CV matrix C only the knots $\{t_0, \dots, t_{n-1}\}$ are assumed to be equally spaced on the unit circle, whereas $\{s_0, \dots, s_{n-1}\}$ is an unrestricted set of n knots. We still yield the desired HSS approximation of CV matrices by exploiting a proper partition of the complex plane into congruent sectors that share the origin 0. To decrease the cost of computing this approximation and of the subsequent computations with HSS matrices, we handle the harder and so far untreated case where the diagonal blocks are rectangular and have row indices that pairwise overlap. We provide some new insights into the subject as well as the detailed analysis of our algorithms and the background for the paper [Pa], where our present algorithms have been extended to various other classes of structured matrices and to minor acceleration of the known algorithms for solving Toeplitz and Hankel linear systems of equations.

We refer the reader to the papers and books [GKK85], [DV98], [T00], [EGH13], [VVGM], [MRT05], [CDG06], [CGS07], [VVM07], [VVM08], [X12], [XXG12], [X13], [XXCB14], [B10], [BY13], [GR87], [DGR96], [CGR98], [LRT79], [P93], [PR93], and the bibliography therein on FMM, HSS matrices, and Matrix Compression (e.g., Nested Dissection) algorithms.

We organize our paper as follows. In the next section we recall some basic results on computations with general matrices. In Section 3 we study polynomial and rational evaluation and interpolation as computations with Vandermonde and Cauchy matrices. In Sections 4–6 we extend the known results on HSS matrix computations. In Section 7 we apply these results to treat CV matrices. In Section 8 we discuss extensions and implementation. In Section 9 we summarize our study. The Appendix contains figures and the legends. Most part of Sections 7 and 8 can be read independently of the previous sections, except for their concluding Theorem 6.1 and Corollary 6.1.

2 Definitions and auxiliary results

We measure the computational cost by the number of arithmetic operations performed in the field $\mathbb{C}$ of complex numbers with no error. $|\mathcal{S}|$ denotes the cardinality of a set $\mathcal{S}$. $M = (m_{i,j})_{i,j=0}^{m-1,n-1}$ is an $m \times n$ matrix. M^T is its transpose, M^H is its Hermitian transpose. $\mathcal{C}(B)$ and $\mathcal{R}(B)$ are the index sets of the rows and columns of its submatrix B, respectively. For two sets $\mathcal{I} \subseteq \{1, \dots, m\}$ and $\mathcal{J} \subseteq \{1, \dots, n\}$ define the submatrix $M(\mathcal{I}, \mathcal{J}) = (m_{i,j})_{i \in \mathcal{I}, j \in \mathcal{J}}$. $\mathcal{R}(B) = \mathcal{I}$ and $\mathcal{C}(B) = \mathcal{J}$ if and only if $B = M(\mathcal{I}, \mathcal{J})$. Write $M(\mathcal{I}, .) = M(\mathcal{I}, \mathcal{J})$ where $\mathcal{J} = \{1, \dots, n\}$. Write $M(., \mathcal{J}) = M(\mathcal{I}, \mathcal{J})$ where $\mathcal{I} = \{1, \dots, m\}$. $(B_0 \ \dots \ B_{k-1})$ and $(B_0 \mid \dots \mid B_{k-1})$ denote a $1 \times k$ block matrix with k blocks $B_0, \dots, B_{k-1}$, whereas $\mathrm{diag}(B_0, \dots, B_{k-1}) = \mathrm{diag}(B_j)_{j=0}^{k-1}$ is a $k \times k$ block diagonal matrix with k diagonal blocks $B_0, \dots, B_{k-1}$, possibly rectangular. $O = O_{m,n}$ is the $m \times n$ matrix filled with

zeros. $I = I_n$ is the $n \times n$ identity matrix. M is a $k \times l$ *unitary* matrix if $M^H M = I_l$ or $MM^H = I_k$. An $m \times n$ matrix M has a nonunique *generating pair* (F, G^T) *of a length* ρ if $M = FG^T$ for two matrices $F \in \mathbb{C}^{m \times \rho}$ and $G \in \mathbb{C}^{n \times \rho}$. The rank of a matrix is the minimum length of its generating pairs. An $m \times n$ matrix is *regular* or nonsingular if it has full rank $\min\{m, n\}$.

Theorem 2.1. *A matrix M has a rank at least ρ if and only if it has a nonsingular $\rho \times \rho$ submatrix $M(\mathcal{I}, \mathcal{J})$, and if so, then $M = M(., \mathcal{J})M(\mathcal{I}, \mathcal{J})^{-1}M(\mathcal{I}, .)$.*

The theorem defines a *generating triple* $(M(., \mathcal{J}), M(\mathcal{I}, \mathcal{J})^{-1}, M(\mathcal{I}, .))$ and two generating pairs $(M(., \mathcal{J}), M(\mathcal{I}, \mathcal{J})^{-1}M(\mathcal{I}, .)$ and $(M(., \mathcal{J})M(\mathcal{I}, \mathcal{J})^{-1}, M(\mathcal{I}, .)$ for a matrix M of a length ρ. We call such pairs and triples *generators*. One can obtain some generators of the minimum length for a given matrix by computing its SVD or its less costly rank revealing factorizations such as ULV and URV factorizations in [CGS07], [XXG12], and [XXCB14], where the factors are unitary, diagonal or triangular.

$\alpha(M)$ and $\beta(M)$ denote the arithmetic cost of computing the vectors $M\mathbf{u}$ and $M^{-1}\mathbf{u}$, respectively, maximized over all vectors $\mathbf{u}$ and minimized over all algorithms, and we write $\beta(M) = \infty$ where the matrix M is singular. The straightforward algorithm supports the following bound.

Theorem 2.2. *$\alpha(M) \le 2(m + n)\rho - \rho - m$ for an $m \times n$ matrix M given with its generating pair of a length ρ.*

$||M|| = ||M||_2$ denotes the spectral norm of an $m \times n$ matrix $M = (m_{i,j})_{i,j=0}^{m-1,n-1}$, and we also write $|M| = \max_{i,j} |m_{i,j}|$, $||M|| \le \sqrt{mn}|M|$. It holds that $||U|| = 1$ and $||MU|| = ||UM|| = ||M||$ for a unitary matrix U. A vector $\mathbf{u}$ is unitary if and only if $||\mathbf{u}|| = 1$, and if this holds we call it a *unit vector*. A matrix $\tilde{M}$ is an *ϵ-approximation* of a matrix M if $|\tilde{M} - M| \le \epsilon$. The *$\epsilon$-rank* of a matrix M denotes the integer $\min_{|\tilde{M}-M| \le \epsilon} \operatorname{rank}(\tilde{M})$. An *$\epsilon$-basis* for a linear space $\mathbb{S}$ of dimension k is a set of vectors that ϵ-approximate the k vectors of a basis for this space. An ϵ-generator of a matrix is a generator of its ϵ-approximation. $\alpha_\epsilon(M)$ and $\beta_\epsilon(M)$ replace the bounds $\alpha(M)$ and $\beta(M)$ where we ϵ-approximate the vectors $M\mathbf{u}$ and $M^{-1}\mathbf{u}$ instead of evaluating them. The *numerical rank* of a matrix M, which we denote $\operatorname{nrank}(M)$, is its ϵ-rank for a small ϵ. A matrix M is *ill conditioned* if its rank exceeds its numerical rank.

Theorem 2.3. *(See [S98, Corollary 1.4.19] for $P = -M^{-1}E$.) Suppose M and $M + E$ are two nonsingular matrices of the same size and $||M^{-1}E|| = \theta < 1$. Then $||I - (M + E)^{-1}M|| \le \frac{\theta}{1-\theta}$ and $||(M + E)^{-1} - M^{-1}|| \le \frac{\theta}{1-\theta}||M^{-1}||$. In particular $||(M + E)^{-1} - M^{-1}|| \le 1.5\ \theta\ ||M^{-1}||$ if $\theta \le 1/3$.*

3 Polynomial and rational evaluation and interpolation as operations with structured matrices

Let $T = (t_{i-j})_{i,j=0}^{m-1,n-1}$, $H = (h_{i+j})_{i,j=0}^{m-1,n-1}$, $V = V_{\mathbf{s}} = (s_i^j)_{i,j=0}^{m-1,n-1}$, and $C = C_{\mathbf{s},\mathbf{t}} = \left(\frac{1}{s_i - t_j}\right)_{i,j=0}^{m-1,n-1}$ denote $m \times n$ *Toeplitz, Hankel, Vandermonde, and Cauchy* matrices, respectively, which are four classes of highly popular structured matrices, each having mn entries defined by at most $m + n$ parameters. (Some authors define Vandermonde matrices as the transposes V^T, rather than the above matrices V.) The four matrix structures have quite distinct features. The matrix structure of Cauchy type is invariant in row and column interchange, in contrast to the structures of Toeplitz and Hankel types. This structure is also stable in shift and scaling its basic knots (cf. (3.5)), unlike the structure of Vandermonde type, and it supports approximation by HSS matrices, unlike the structures of the three other types. The paper [P90], however, has linked the four structures to each other by means of structured matrix multiplication and *proposed to exploit this link in order to extend any successful matrix inversion algorithm for the matrices of any of the four classes to the matrices of the three other classes.* In this paper we study Vandermonde and Cauchy matrices, linked to polynomial and rational evaluation and interpolation.

Problem 1. Multipoint polynomial evaluation or Vandermonde-by-vector multiplication.

INPUT: $m+n$ complex scalars $p_0,\dots,p_{n-1}; s_0,\dots,s_{m-1}$.
OUTPUT: n complex scalars $v_0,\dots,v_{m-1}$ satisfying

$$v_i = p(s_i) \text{ for } p(x) = p_0 + p_1 x + \cdots + p_{n-1}x^{n-1} \text{ and } i = 0,\dots,m-1 \tag{3.1}$$

or equivalently

$$V\mathbf{p} = \mathbf{v} \text{ for } V = V_{\mathbf{s}} = (s_i^j)_{i,j=0}^{m-1,n-1},\ \mathbf{p} = (p_j)_{j=0}^{n-1},\ \text{and } \mathbf{v} = (v_i)_{i=0}^{m-1}. \tag{3.2}$$

Problem 2. Polynomial interpolation or the solution of a Vandermonde linear system of equations.
INPUT: $2n$ complex scalars $v_0,\dots,v_{n-1}; s_0,\dots,s_{n-1}$, the last n of them distinct.
OUTPUT: n complex scalars $p_0,\dots,p_{n-1}$ satisfying equations (3.1) and (3.2) for $m=n$.

Problem 3. Multipoint rational evaluation or Cauchy-by-vector multiplication.
INPUT: $2m+n$ complex scalars $s_0,\dots,s_{m-1}; t_0,\dots,t_{n-1}; v_0,\dots,v_{m-1}$.
OUTPUT: m complex scalars $v_0,\dots,v_{m-1}$ satisfying

$$v_i = \sum_{j=0}^{n-1} \frac{u_j}{s_i - t_j} \text{ for } i = 0,\dots,m-1 \tag{3.3}$$

or equivalently

$$C\mathbf{u} = \mathbf{v} \text{ for } C = C_{\mathbf{s},\mathbf{t}} = \Big(\frac{1}{s_i - t_j}\Big)_{i,j=0}^{m-1,n-1},\ \mathbf{u} = (u_j)_{j=0}^{n-1},\ \text{and } \mathbf{v} = (v_i)_{i=0}^{m-1}. \tag{3.4}$$

Problem 4. Rational interpolation or the solution of a Cauchy linear system of equations.
INPUT: $3n$ complex scalars $s_0,\dots,s_{n-1}; t_0,\dots,t_{n-1}; v_0,\dots,v_{n-1}$, the first $2n$ of them distinct.
OUTPUT: n complex scalars $u_0,\dots,u_{n-1}$ satisfying equations (3.3) and (3.4) for $m=n$.

The scalars $s_0,\dots,s_{m-1},t_0,\dots,t_{n-1}$ define the Vandermonde and Cauchy matrices $V_{\mathbf{s}}$ and $C_{\mathbf{s},\mathbf{t}}$, are basic for Problems 1–4, and are said to be the *knots*. A Cauchy matrix is stable in shifting its knots and scaling them by constants, that is

$$aC_{a\mathbf{s},a\mathbf{t}} = C_{\mathbf{s},\mathbf{t}} \text{ and } C_{\mathbf{s}+a\mathbf{e},\mathbf{t}+a\mathbf{e}} = C_{\mathbf{s},\mathbf{t}} \text{ for } a \neq 0 \text{ and } \mathbf{e} = (1,\dots,1)^T. \tag{3.5}$$

Theorem 3.1. *(i) An $m\times n$ Vandermonde matrix $V_{\mathbf{s}} = (s_i^j)_{i,j=0}^{m-1,n-1}$ has full rank if and only if all m knots $s_0,\dots,s_{m-1}$ are distinct. (ii) An $m\times n$ Cauchy matrix $C_{\mathbf{s},\mathbf{t}} = \Big(\frac{1}{s_i-t_j}\Big)_{i,j=0}^{m-1,n-1}$ is well defined if and only if its two knot sets $s_0,\dots,s_{m-1}$ and $t_0,\dots,t_{n-1}$ share no elements. (iii) If this matrix is well defined, then it has full rank if and only if all its $m+n$ knots $s_0,\dots,s_{m-1},t_0,\dots,t_{n-1}$ are distinct and also (iv) if and only if all its submatrices have full rank.*

Proof. Parts (i)–(iii) are implied by the following equations of independent interest (see, e.g., [P01, Section 3.6]),

$$\det V_{\mathbf{s}} = \prod_{i>j}(s_i - s_j),\ \ \det C_{\mathbf{s},\mathbf{t}} = \prod_{i<j}(s_j - s_i)(t_i - t_j)/\prod_{i,j}(s_i - t_j). \tag{3.6}$$

Part (iv) follows from part part (iii) and the observation that every submatrix of a Cauchy matrix is a Cauchy matrix itself. □

How many arithmetic operations do we need for solving Problems 1–4? The algorithms of [F72], [GGS87], and [MB72] solve Problems 1–3 by using $O((m+n)\log^2(n)\log(\log(n)))$ arithmetic operations over any field of constants. For $m \geq n$ this is within a factor of $\log(n)\log(\log(n))$ from the optimum [S73], [B-O83]. Equation (3.7) of this subsection extends the latter upper bound to Problem 4. For numerical solution of Problems 1–4, however, the users employ quadratic time

algorithms to avoid error propagation (cf. [BF00], [P64], [BP70], [BEGO08]), in spite of substantial research progress reported in [PRT92], [PSLT93], [P95], [PZHY97], and particularly [DGR96].

We can solve Problems 1–4 numerically by using $O(n\log(n))$ arithmetic operations in the important special case where the knots $s_i = \omega^i$ are the nth roots of 1, $\omega = \omega_n = \exp(2\pi\sqrt{-1}/n)$, $i = 0, \dots, n-1$, and $V_{\mathbf{s}} = (\omega^{ij})_{i,j=0}^{n-1}$, and hereafter we write $\Omega = \frac{1}{\sqrt{n}}(\omega^{ij})_{i,j=0}^{n-1}$. In this case Problems 1 and 2 turn into the computational problems of the forward and inverse discrete Fourier transforms (hereafter *DFT* and *IDFT*). The *FFT* (Fast Fourier transform) and the Inverse FFT are two numerically stable algorithms that perform DFT and IDFT at the arithmetic cost $1.5n\log_2(n)$ and $1.5n\log_2(n)+n$, respectively, if $m = n$ is a power of 2 (cf. [BP94, Sections 1.2 and 3.4]), whereas the Generalized FFT and the Generalized Inverse FFT use $O(n\log(n))$ arithmetic operations to perform DFT and IDFT for any n [P01, Problem 2.4.2]. Note that $\Omega^H\Omega = I_n$, that is $\Omega = \Omega^T$ and $\Omega^H = \Omega^{-1} = \frac{1}{\sqrt{n}}(\omega^{-ij})_{i,j=0}^{n-1}$ are unitary matrices. The following equation links Problems 1 and 2 to Cauchy matrix computations (cf. [P01, Section 3.6]),

$$C_{\mathbf{s},\mathbf{t}} = \operatorname{diag}(t(s_i)^{-1})_{i=0}^{m-1} V_{\mathbf{s}} V_{\mathbf{t}}^{-1} \operatorname{diag}(t'(t_j))_{j=0}^{n-1},\ t(x = \prod_{i=0}^{n-1}(x - t_j)). \tag{3.7}$$

Remark 3.1. *Assume the latter polynomial $t(x)$ and write $v(x) = t(x) - x^m$. Then one can compute the values $v(t_0) = -t_0^n \dots, v(t_{n-1}) = -t_{n-1}^n$ of the polynomial $v(x)$ at the knots $t_0, \dots, t_{n-1}$ by using $O(n\log^2(n))$ arithmetic operations. Given these knots, one can compute the values $v(t_i) = -t_i^n$ for $i = 0, \dots, n-1$ by using $O(n\log(n))$ arithmetic operations, and then one can recover the coefficients of the polynomial $v(x)$ by solving Problem 2 of polynomial interpolation.*

For $\mathbf{t} = (f\omega^j)_{j=0}^{n-1}$, $f \neq 0$, the knots t_j are the nth roots of 1 scaled by f, $t(x) = x^n - f^n$, $t'(x) = nx^{n-1}$, $V_{\mathbf{t}} = \sqrt{n}\Omega \operatorname{diag}(f^j)_{j=0}^{n-1}$, $V_{\mathbf{t}}^{-1} = \frac{1}{\sqrt{n}} \operatorname{diag}(f^{-j})_{j=0}^{n-1}\Omega^H$, and we write $C_{\mathbf{s},f} = (\frac{1}{s_i - f\omega^j})_{i,j=0}^{n-1}$ and deduce from equation (3.7) that

$$C_{\mathbf{s},f} = \sqrt{n} \operatorname{diag}\Big(\frac{f^{n-1}}{s_i^n - f^n}\Big)_{i=0}^{m-1} V_{\mathbf{s}} \operatorname{diag}(f^{-j})_{j=0}^{n-1}\Omega^H \operatorname{diag}(\omega^{-j})_{j=0}^{n-1}, \tag{3.8}$$

thus linking Vandermonde matrices and their inverses to the $m \times n$ Cauchy matrices $C_{\mathbf{s},f}$ (for $f \neq 0$), which we call *CV matrices*. The $n \times m$ matrices $C_{e,\mathbf{t}} = -C_{\mathbf{t},e}^T = \Big(\frac{1}{e\omega^i - t_j}\Big)_{i,j=0}^{n-1,m-1}$ for $e \neq 0$ have the knot set $\mathcal{S} = \{s_i = e\omega^i,\ i = 0, \dots, n-1\}$, are linked to the transposed Vandermonde matrices, and are said to be CV^T *matrices*. Let us display these links more explicitly,

$$V_{\mathbf{s}} = \frac{f^{1-n}}{\sqrt{n}} \operatorname{diag}\Big(s_i^n - f^n\Big)_{i=0}^{m-1} C_{\mathbf{s},f} \operatorname{diag}(\omega^j)_{j=0}^{n-1}\Omega \operatorname{diag}(f^j)_{j=0}^{n-1}, \tag{3.9}$$

$$V_{\mathbf{s}}^T = -\frac{f^{1-n}}{\sqrt{n}} \operatorname{diag}(f^j)_{j=0}^{n-1}\Omega \operatorname{diag}(\omega^j)_{j=0}^{n-1} C_{f,\mathbf{s}} \operatorname{diag}(s_i^n - f^n)_{i=0}^{m-1}, \tag{3.10}$$

and for $m = n$ also

$$V_{\mathbf{s}}^{-1} = \sqrt{n} \operatorname{diag}(f^{-j})_{j=0}^{n-1}\Omega^H \operatorname{diag}(\omega^{-j})_{j=0}^{n-1} C_{\mathbf{s},f}^{-1} \operatorname{diag}\Big(\frac{f^{n-1}}{s_i^n - f^n}\Big)_{i=0}^{n-1}, \tag{3.11}$$

$$V_{\mathbf{s}}^{-T} = -\sqrt{n} \operatorname{diag}\Big(\frac{f^{n-1}}{s_i^n - f^n}\Big)_{i=0}^{n-1} C_{f,\mathbf{s}}^{-1} \operatorname{diag}(\omega^{-j})_{j=0}^{n-1}\Omega^H \operatorname{diag}(f^{-j})_{j=0}^{n-1}. \tag{3.12}$$

Remark 3.2. *By linking together the Vandermonde and Cauchy matrix structures, equations (3.7)–(3.12) also link Problems 1 and 2 to Problems 3 and 4. Equivalently, assume $p(x)$ of equation (3.1), $t(x) = \prod_{j=0}^{n-1}(x - t_j)$, and n distinct knots $t_0, \dots, t_{n-1}$ and then represent the rational function $v(x) = \frac{p(x)}{t(x)}$ as $v(x) = \sum_{j=0}^{n-1} \frac{u_j}{x - t_j}$. We obtain equations (3.3) by writing $v_i = v(s_i)$ for $i = 0, \dots, m-1$.*

Theorem 3.2 below as well as [P01, Equation (3.4.1)] link together a Vandermonde matrix, its transpose, inverse and the inverse of the transpose. [P01, Sections 4.7 and 4.8] and [Pa] cover more comprehensively such links among matrix structures as well as the links between the computations with structured matrices and polynomials, exemplified by equivalent formulations of Problems 1–4 in terms of either polynomials and rational functions or Vandermonde and Cauchy matrices.

Theorem 3.2. *(i) JH and HJ are Toeplitz matrices if H is a Hankel matrix, and vice versa. (ii) $H = V^TV = (\sum_{k=0}^{m-1} s_k^{i+j})_{i,j=0}^{n-1}$ is a Hankel matrix for any $m \times n$ Vandermonde matrix $V = (s_i^j)_{i,j=0}^{m-1,n-1}$.*

4 Quasiseparable and HSS matrices

Next we define HSS matrices and study their multiplication by vectors and the solution of nonsingular HSS linear systems of equations.

4.1 Quasiseparable matrices and generators

Definition 4.1. *A matrix given with its block diagonal is* (l, u)-quasiseparable *if l and u are the maximum ranks of its sub- and superdiagonal blocks, respectively. By replacing ranks with ϵ-ranks we define* (ϵ, l, u)-quasiseparable *matrices.*

In particular the matrices with a lower bandwidth l and an upper bandwidth u as well as their inverses (if defined) are (l, u)-quasiseparable. We are going to study a variation of this class, which we call the balanced ρ-HSS matrices (see Definition 4.4). To provide better insight into this subject, next we recall some results on quasiseparable matrices and compare them with the balanced ρ-HSS matrices in Theorems 4.4 and 4.5 and Corollaries 4.1 and 4.2.

Theorem 4.1. *[DV98], [EG02]. Suppose that an (l, u)-quasiseparable matrix M is given with $m_q \times n_q$ diagonal blocks Σ_q, $q = 0, \dots, k-1$, such that $\sum_{q=0}^{k-1} m_q = m$, $\sum_{q=0}^{k-1} n_q = n$, and $s = \sum_{q=0}^{k-1} m_q n_q = O((l+u)(m+n))$. Then*

$$\alpha(M) \le 2\sum_{q=0}^{k-1}((m_q + n_q)(l+u) + s) + 2l^2k + 2u^2k = O((l+u)(m+n)).$$

Furthermore if $m_q = n_q$ for all q and if the matrix M is nonsingular, then

$$\beta(M) = O(\sum_{q=0}^{k-1}((l+u)^2(l+u+n_q)n_q + n_q^3)).$$

The algorithms of [DV98], [EG02] supporting the theorem as well as the study in [CGS07], [VVM07], [VVM08], [XXG12], and [EGH13] rely on the representation of (l, u)-quasiseparable matrices with *quasiseparable generators*, demonstrated by the following 4×4 example and defined in Theorem 4.2,

$$M = \begin{pmatrix} \Sigma_0 & S_0T_1 & S_0B_1T_2 & S_0B_1B_2T_3 \\ P_1Q_0 & \Sigma_1 & S_1T_2 & S_1B_2T_3 \\ P_2A_1Q_0 & P_2Q_1 & \Sigma_2 & S_2T_3 \\ P_3A_2A_1Q_0 & P_3A_2Q_1 & P_3Q_2 & \Sigma_3 \end{pmatrix}. \tag{4.1}$$

Note that M is a block tridiagonal matrix where $A_p = B_q = O$ for all p and q.

Theorem 4.2. *(Cf. [EGH13], [VVM07], [X12], the bibliography therein, and our Table 4.1.) Assume a $k \times k$ matrix M with a block diagonal $\widehat{\Sigma} = (\Sigma_0, \dots, \Sigma_{k-1})$, where $\Sigma_q = M(I_q, J_q)$,*

$q = 0, \dots, k-1$. *Then* M *is an* (l, u)*-quasiseparable matrix if and only if there exists a nonunique family of* quasiseparable generators $\{P_i,\ Q_h,\ S_h,\ T_i,\ A_g,\ B_g\}$ *such that*

$$M(\mathcal{I}_i, \mathcal{J}_h) = P_i A_{i-1} \cdots A_{h+1} Q_h \text{ and } M(\mathcal{I}_h, \mathcal{J}_i) = S_h B_{h+1} \cdots B_{i-1} T_i$$

for $0 \le h < i < k$. *Here* P_i, Q_h *and* A_g *are* $|\mathcal{I}_i| \times l_i$, $l_{h+1} \times |\mathcal{J}_h|$, *and* $l_{g+1} \times l_g$ *matrices, respectively, whereas* S_h, T_i *and* B_g *are* $|\mathcal{I}_h| \times u_{h+1}$, $u_i \times |J_i|$, *and* $u_g \times u_{g+1}$ *matrices, respectively,* $g = 1, \dots, k-2$, $h = 0, \dots, k-2$, $i = 1, \dots, k-1$, *and the integers* $l = \max_g\{l_g\}$ *and* $u = \max_h\{u_h\}$ *are called the* lower and upper lengths *or* orders *of the quasiseparable generators.*

Table 4.1: The sizes of quasiseparable generators of Theorem 4.2

P_i	Q_h	A_g	S_h	T_i	B_g
$\|\mathcal{I}_i\| \times l_i$	$l_{h+1} \times \|\mathcal{J}_h\|$	$l_{g+1} \times l_g$	$\|\mathcal{I}_h\| \times u_{h+1}$	$u_i \times \|J_i\|$	$u_g \times u_{g+1}$

By virtue of this theorem one can redefine the (l, u)-quasiseparable matrices as the ones allowing representation with the families $\{P_h,\ Q_i,\ A_g\}$ and $\{S_h,\ T_i,\ B_g\}$ of quasiseparable generators having lower and upper orders l and u, respectively. Definition 4.1 and Theorem 4.2 provide two useful insights into the properties of (l, u)-quasiseparable matrices. In the next subsections we employ the third equivalent definition, providing yet another insight and linked to the study of the Cauchy matrix $C_{1,\omega_{2n}}$ in the papers [CGS07], [XXG12], [XXCB14].

4.2 Recursive merging of diagonal blocks of a matrix

Definition 4.2. *Assume a* $1 \times k$ *block matrix* $M = (M_0 \mid \dots \mid M_{k-1})$ *with* k basic block columns M_q, *each partitioned into a* diagonal block Σ_q *and a* basic neutered block column N_q, $q = 0, \dots, k-1$ *(cf. our Figures 2–4 and [MRT05, Section 1]). A matrix given with its block diagonal is* basically ρ-neutered *if all its basic neutered block columns have ranks at most* ρ. *By replacing ranks with* ϵ*-ranks we define basically* (ϵ, ρ)*-neutered matrices.*

Definition 4.3. *Fix two positive integers* l *and* q *such that* $l + q \le k$ *and* merge *the* l *basic block columns* $M_q, M_{q+1}, \dots, M_{q+l-1}$, *the* l *diagonal blocks* $\Sigma_q, \Sigma_{q+1}, \dots, \Sigma_{q+l-1}$, *and the* l *basic neutered block columns* $N_q, N_{q+1}, \dots, N_{q+l-1}$ *into their* union $M_{q,l} = M(., \cup_{j=0}^{l-1} \mathcal{C}(\Sigma_{q+j}))$, *their* diagonal union $\Sigma_{q,l}$, *and their* neutered union $N_{q,l}$, *respectively, such that* $\mathcal{R}(\Sigma_{q,l}) = \cup_{j=0}^{l-1} \mathcal{R}(\Sigma_{q+j})$ *and the block column* $M_{q,l}$ *is partitioned into the diagonal union* $\Sigma_{q,l}$ *and the neutered union* $N_{q,l}$.

Define *recursive merging* of all diagonal blocks $\Sigma_0, \dots, \Sigma_{k-1}$ by a binary tree whose leaves are associated to these blocks and whose every internal vertex is the union of its two children. For every vertex v define the sets $L(v)$ and $R(v)$ of its left and right descendants, respectively. Then a binary tree is *balanced* if $0 \le |L(v)| - |R(v)| \le 1$ for all its vertices v. Such a tree identifies *balanced merging* of its leaves, in our case the diagonal blocks. We can uniquely define a balanced tree with n leaves by removing the $2^{l(n)} - n$ rightmost leaves of the complete binary tree that has $2^{l(n)}$ leaves for $l(n) = \lceil \log_2(n) \rceil$. All leaves of the resulting *heap structure* with n leaves lie in its two lowest levels.

For example, the complete binary tree of Figure 1 represents balanced recursive merging of eight diagonal blocks $\Sigma_0, \Sigma_1, \dots, \Sigma_7$. At first we merge them into the four diagonal unions of the four pairs $\Sigma_{0,1} = \Sigma(\Sigma_0, \Sigma_1), \dots, \Sigma_{6,7} = \Sigma(\Sigma_6, \Sigma_7)$, then into the two diagonal unions of two quadruples

$$\Sigma_{0,1,2,3} = \Sigma(\Sigma_{0,1}, \Sigma_{2,3}) = \Sigma(\Sigma_0, \Sigma_1, \Sigma_2, \Sigma_3),$$

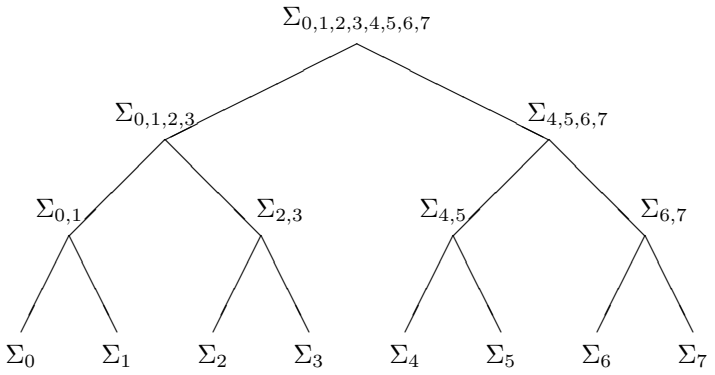
$$\Sigma_{4,5,6,7} = \Sigma(\Sigma_{4,5}, \Sigma_{6,7}) = \Sigma(\Sigma_4, \Sigma_5, \Sigma_6, \Sigma_7),$$

and finally into the diagonal union of the single 8-tuple,

$$\Sigma_{0,1,2,3,4,5,6,7} = \Sigma(\Sigma_{0,1,2,3}, \Sigma_{4,5,6,7}) = \Sigma(\Sigma_0, \Sigma_1, \Sigma_2, \Sigma_3, \Sigma_4, \Sigma_5, \Sigma_6, \Sigma_7).$$

By removing the h rightmost leaves for $h < 7$ we can represent balanced merging of $8 - h$ diagonal blocks.

Figure 1: Balanced merging of diagonal blocks.



4.3 HSS and balanced HSS matrices, their link to quasiseparable matrices, and the cost of basic operations with them

Theorem 4.3. *Assume an $m \times n$ matrix M with a block diagonal $\Sigma = \mathrm{diag}(\Sigma_q)_{q=0}^{k-1}$ and k generators $(F_0, G_0), \dots, (F_{k-1}, G_{k-1})$ of lengths at most ρ for the k basic neutered block columns. Then*

$$\alpha(M) \le \alpha(\Sigma) + (2m + 2n - 1)k\rho.$$

Proof. Write $M = M' + \mathrm{diag}(\Sigma_q)_{q=0}^{k-1}$. Note that $\alpha(M) \le \alpha(\Sigma) + \alpha(M') + m$. The basic neutered block columns of the matrix M share their entries with the matrix M', whose other entries are zeros. So the k pairs $(F_0, G_0), \dots, (F_{k-1}, G_{k-1})$ together form a single generating pair of a length at most $k\rho$ for the matrix M'. Therefore $\alpha(M') \le (2m + 2n - 1)k\rho - m$ by virtue of Theorem 2.2. □

Basically ρ-neutered matrices are precisely the input class of Theorem 4.3, whose cost estimates are weaker than in Theorem 4.1. By adding row-wise links among the basic neutered block columns of a basically ρ-neutered matrix we can turn it into (ρ, ρ)-quasiseparable, as we show next (see Theorem 4.4).

Definition 4.4. *(i) A matrix given with its block diagonal is a* balanced ρ-HSS *matrix if it is basically ρ-neutered throughout the process of balanced merging of its diagonal blocks, that is if all neutered unions of its basic neutered block columns involved into this process have ranks at most ρ. (ii) This is a* ρ-HSS *matrix if it is basically ρ-neutered throughout any process of recursive merging of its diagonal blocks. (iii) By replacing ranks with ϵ-ranks we define* balanced (ϵ, ρ)-HSS matrices *and* (ϵ, ρ)-HSS matrices.

Fact 4.1. *(i) Let a matrix be basically ρ_j-neutered at the j-th step of recursive balanced merging for every j. Then this is a balanced ρ-HSS matrix for $\rho = \max_j \rho_j$. (ii) Likewise, let a matrix be basically (ϵ_j, ρ_j)-neutered at the j-th step of recursive balanced merging for every j. Then this is a balanced (ϵ, ρ)-HSS matrix for $\epsilon = \max_j \epsilon_j$ and $\rho = \max_j \rho_j$.*

Theorem 4.4. *(i) Every (l, u)-quasiseparable matrix M is an $(l+u)$-HSS matrix. (ii) Every ρ-HSS matrix is (ρ, ρ)-quasiseparable.*

Proof. A basic neutered block column N_q of a matrix can be partitioned into its basic block sub- and superdiagonal parts L_q and U_q, respectively, and so $\mathrm{rank}(N_q) \le \mathrm{rank}(L_q) + \mathrm{rank}(U_q)$, which implies that $\mathrm{rank}(N_q) \le l + u$ for $q = 0, \dots, k-1$ if the matrix M is (l, u)-quasiseparable. This proves part (i). Next note that the union N of any set of basic neutered block columns of a matrix M can be turned into a basic neutered block column at some stage of an appropriate process of recursive merging. Therefore $\mathrm{rank}(N) \le \rho$ where M is a ρ-HSS matrix. Now for every off-diagonal block B of a matrix M define the set of its basic neutered block columns that share some column

indices with the block B and then note that the block B is a submatrix of the neutered union of this set. Therefore $\mathrm{rank}(B) \le \mathrm{rank}(N) \le \rho$, and we obtain part (ii). $\square$

By combining Theorems 4.1 and 4.4 we obtain the following results.

Corollary 4.1. *Assume a ρ-HSS matrix M given with $m_q \times n_q$ diagonal blocks Σ_q, $q = 0, \dots, k-1$, and write $m = \sum_{q=0}^{k-1} m_q$, $n = \sum_{q=0}^{k-1} n_q$, and $s = \sum_{q=0}^{k-1} m_q n_q$. Then $\alpha(M) < 2s + 4\rho^2 k + 4\sum_{q=0}^{k-1}(m_q + n_q)\rho = O((m+n)\rho + s)$. Furthermore if $m_q = n_q$ for all q and if the matrix M is nonsingular, then $\beta(M) = O(\sum_{q=0}^{k-1}((\rho + n_q)\rho^2 n_q + n_q^3))$.*

For a balanced ρ-HSS matrix M we only have a little weaker representation than in Theorem 4.2; so the proof of the estimates of Corollary 4.1 for $\alpha(M)$ and $\beta(M)$ does not apply. We bound $\alpha(M)$ and $\beta(M)$ by adjusting the algorithms of [CGS07, Sections 3 and 4], [XXG12], and [XXCB14], devised for a distinct matrix class. Unlike Theorem 4.1 and Corollary 4.1, we allow $m_q \ne n_q$ for all q.

Theorem 4.5. *Assume a balanced ρ-HSS matrix M with $m_q \times n_q$ diagonal blocks Σ_q, $q = 0, \dots, k-1$, having $s = \sum_{q=0}^{k-1} m_q n_q$ entries overall and write $l = \lceil \log_2(k) \rceil$, $m = \sum_{q=0}^{k-1} m_q$, $n = \sum_{q=0}^{k-1} n_q$, $m_+ = \max_{q=0}^{k-1} m_q$, $n_+ = \max_{q=0}^{k-1} n_q$, and $s = \sum_{q=0}^{k-1} m_q n_q$, $s \le \min\{m_+ n, m n_+\}$. (i) Then*

$$\alpha(M) < 2s + (m + 4(m+n)\rho)l. \tag{4.2}$$

(ii) Moreover if $m = n$ and if the matrix M is nonsingular, then

$$\beta(M) = O(n_+ s + (n_+^2 + \rho n_+ + l\rho^2)n + (k\rho + n)\rho^2). \tag{4.3}$$

(iii) Furthermore the same bounds (4.2) and (4.3) hold for the transpose of a balanced ρ-HSS matrix M matrix having $n_q \times m_q$ diagonal blocks Σ_q for $q = 0, \dots, k-1$.

Corollary 4.2. *Under the assumptions of parts (i)–(iii) of Theorem 4.5 suppose that $k\rho = O(n)$ and $n_+ + \rho = O(\log(n))$. Then $\alpha(M) = O((m+n)\log^2(n))$ and $\beta(M) = O(n\log^3(n))$.*

5 Proof of Theorem 4.5

5.1 A proof of bound (4.2)

With no loss of generality assume that the $(l-1)$st, that is the final stage of a balanced merging process has produced a 2×2 block representation

$$M = \begin{pmatrix} \bar{\Sigma}_0^{(l)} & \bar{S}_{01}^{(l)} \bar{T}_1^{(l)} \\ \bar{S}_{10}^{(l)} \bar{T}_0^{(l)} & \bar{\Sigma}_1^{(l)} \end{pmatrix}$$

where $\bar{\Sigma}_j^{(l)}$ is an $\bar{m}_j^{(l)} \times \bar{n}_j^{(l)}$ matrix, $\bar{T}_j^{(l)}$ is an $\bar{n}_j^{(l)} \times \bar{\rho}_j^{(l)}$ matrix, $\bar{\rho}_j^{(l)} \le \rho$, $j = 0, 1$, $\bar{m}_1^{(l)} + \bar{m}_2^{(l)} = m$, and $\bar{n}_1^{(l)} + \bar{n}_2^{(l)} = n$.

Clearly $\alpha(M) \le m + \sum_{j=0}^{1} \alpha(\bar{\Sigma}_j^{(l)}) + \sum_{j=0}^{1} \alpha(\bar{T}_j^{(l)}) + \alpha(\bar{S}_{01}^{(l)}) + \alpha(\bar{S}_{10}^{(l)})$. Apply Theorem 2.2 and obtain that $\sum_{j=0}^{1} \alpha(\bar{T}_j^{(l)}) + \alpha(\bar{S}_{01}^{(l)}) + \alpha(\bar{S}_{10}^{(l)}) < 4(m+n)\rho$.

The second last stage of the balanced merging process produces a similar 2×2 block representation for each of the diagonal blocks $\bar{\Sigma}_j^{(l)}$, $j = 0, 1$, and therefore $\sum_{j=0}^{1} \alpha(\bar{\Sigma}_j^{(l)}) < m + 4(m+n)\rho + \sum_{j=0}^{k(1)} \alpha(\bar{\Sigma}_j^{(1-1)})$ where $\bar{\Sigma}_0^{(1-1)}, \dots, \bar{\Sigma}_{k(1)-1}^{(1-1)}$ are the diagonal blocks output at the second last merging stage (cf. Figures 3 and 4). By recursively going back through the merging process, we obtain that $\alpha(M) < (m + 4(m+n)\rho)l + \sum_{j=0}^{k-1} \alpha(\Sigma_j)$ where $\Sigma_q = \bar{\Sigma}_q^{(0)}$ is an $m_q \times n_q$ matrix for $m_q = \bar{m}_q^{(0)}$, $n_q = \bar{n}_q^{(0)}$, and $q = 0, \dots, k-1$. Consequently $\sum_{q=0}^{k-1} \alpha(\Sigma_q) < 2\sum_{q=0}^{k-1} m_q n_q = 2s$, and we arrive at bound (4.2).

5.2 Some introductory comments for proving bound (4.3)

The algorithm of [CGS07, Section 3] factors an (l,u)-quasiseparable matrix M into the product of unitary and block triangular matrices. This enables unitary reduction of a nonsingular linear systems of equations $M\mathbf{y}=\mathbf{b}$ to triangular linear systems, and then one can compute the solution vector $\mathbf{y}$ in nearly linear arithmetic time. We apply the algorithm to a balanced ρ-HSS matrix M and obtain similar factorization and unitary reduction to deduce the cost bounds of Theorem 4.5. We rearrange the computations to facilitate the proof of the arithmetic cost estimates, not presented in [CGS07]. As in [CGS07, Section 3] we demonstrate the algorithm for a 4×4 block matrix, although instead of (l,u)-HSS matrix of (4.1) we work with a basically ρ-HSS matrix

$$M=\begin{pmatrix} \Sigma_0 & S_{01}T_1 & S_{02}B_{12}T_2 & S_{03}B_{13}B_{23}T_3\\ S_{10}T_0 & \Sigma_1 & S_{12}T_2 & S_{13}B_{23}T_3\\ S_{20}B_{20}T_0 & S_{21}T_1 & \Sigma_2 & S_{23}T_3\\ S_{30}B_{20}B_{10}T_0 & S_{31}B_{32}T_1 & S_{23}T_2 & \Sigma_3 \end{pmatrix} \tag{5.1}$$

having $m_q\times n_q$ diagonal blocks Σ_q for any pairs $m_q\times n_q$ and $q=0,1,2,3$. For balanced ρ-HSS matrices M we could have written $B_{p,q}=I$ for all pairs of p and q, but we use expression (5.1) to simplify comparison with [CGS07]). As soon as we complete the description of the construction for $k=4$, we outline its generalization to the case of any positive integer k.

5.3 Compression and merging stages

At first, for $k=4$ and $q=0,1,2,3$, we compute the QR factors of the matrices T_q^H, that is compute square unitary matrices U_q (in factored form) and $\rho_q\times\widehat{n}_q$ matrices $\widehat{T}_q$ of full column ranks $\widehat{n}_q$ such that $T_qU_q^H=(O\mid\widehat{T}_q)$ and $\widehat{n}_q\le\rho_q\le\rho$. Write $\widehat{U}=\mathrm{diag}(U_q)_{q=0}^3$, $\widehat{M}=M\widehat{U}^H$, $M=\widehat{M}\widehat{U}$, and obtain

$$\widehat{M}=\begin{pmatrix} \Sigma_{00} & \Sigma_{01} & O & S_{01}\widehat{T}_1 & O & S_{02}B_{12}\widehat{T}_2 & O & S_{03}B_{13}B_{23}\widehat{T}_3\\ O & S_{10}\widehat{T}_0 & \Sigma_{10} & \Sigma_{11} & O & S_{12}\widehat{T}_2 & O & S_{13}B_{2,3}\widehat{T}_3\\ O & S_{20}B_{20}\widehat{T}_0 & O & S_{21}\widehat{T}_1 & \Sigma_{20} & \Sigma_{21} & O & S_{23}\widehat{T}_3\\ O & S_{30}B_{20}B_{10}\widehat{T}_0 & O & S_{31}B_{32}\widehat{T}_1 & O & S_{32}\widehat{T}_2 & \Sigma_{30} & \Sigma_{31} \end{pmatrix}.$$

Choose a permutation matrix P_0 such that $\widehat{M}P_0=(\mathrm{diag}(\Sigma_{q0})_{q=0}^3\mid M_1)$,

$$M_1=\begin{pmatrix} \Sigma_{01} & S_{01}\widehat{T}_1 & S_{02}B_{12}\widehat{T}_2 & S_{03}B_{13}B_{23}\widehat{T}_3\\ S_{10}\widehat{T}_0 & \Sigma_{11} & S_{12}\widehat{T}_2 & S_{13}B_{2,3}\widehat{T}_3\\ S_{20}B_{20}\widehat{T}_0 & S_{21}\widehat{T}_1 & \Sigma_{21} & S_{23}\widehat{T}_3\\ S_{30}B_{20}B_{10}\widehat{T}_0 & S_{31}B_{32}\widehat{T}_1 & S_{32}\widehat{T}_2 & \Sigma_{31} \end{pmatrix},$$

and the four diagonal blocks Σ_{q0} have sizes $m_q\times(n_q-\widehat{n}_q)$ for $q=0,1,2,3$. Note that M_1 is a balanced ρ-HSS matrix. Write $M=M^{(0)}$, $\Sigma^{(0)}=\mathrm{diag}(\Sigma_{q0})_{q=0}^3$, and $U^{(0)}=\widehat{U}P_0$, and obtain that

$$M^{(0)}=(\Sigma^{(0)}\mid M_1)U^{(0)}. \tag{5.2}$$

By following [CGS07] we call the above computation of the matrices $U^{(0)}$, $\Sigma^{(0)}$ and M_1 the *compression* of the matrix M. For a fixed ρ we cannot compress the matrix M_1 any further because its every diagonal block Σ_{q0} has at most ρ columns. At this point (cf. [CGS07]) we *merge* pairwise the diagonal blocks Σ_{01}, Σ_{11}, Σ_{21} and Σ_{31} of the matrix M_1 into the diagonal unions of the two pairs, $\Sigma_0^{(1)}=\begin{pmatrix}\Sigma_0^{(1)} & \widehat{S}_{01}^{(1)}T_1^{(1)}\\ \widehat{S}_{10}^{(1)}T_0^{(1)} & \Sigma_1^{(1)}\end{pmatrix}$ and $\Sigma_1^{(1)}=\begin{pmatrix}\Sigma_2^{(1)} & \widehat{S}_{23}^{(1)}T_3^{(1)}\\ \widehat{S}_{32}^{(1)}T_2^{(1)} & \Sigma_3^{(1)}\end{pmatrix}$. By definition, merging preserves the property of being a basically ρ-HSS matrix, and we redefine M_1 as a 2×2 block matrix $M^{(1)}=\begin{pmatrix}\Sigma_0^{(1)} & \widehat{S}_{01}^{(1)}T_1^{(1)}\\ \widehat{S}_{10}^{(1)}T_0^{(1)} & \Sigma_1^{(1)}\end{pmatrix}$ where $\Sigma_q^{(1)}$ are $m_q^{(1)}\times n_q^{(1)}$ matrices, $\widehat{S}_{pq}^{(1)}$ are $m_p^{(1)}\times\rho_q^{(1)}$ matrices, $T_q^{(1)}$ are $\rho_q^{(1)}\times\bar{n}_q$ matrices, $m_q^{(1)}=m_{2q}+m_{2q+1}$, $\bar{n}_q=\widehat{n}_{2q}+\widehat{n}_{2q+1}\le 2\rho$, and $\rho_q^{(1)}\le\rho$ for $p,q\in\{0,1\}$.

5.4 Recursive alternation of compression and merging

By following [CGS07, Section 3] we recursively alternate compression and merging, and next we compress the 2×2 block matrix $M^{(1)}$. We compute unitary matrices $U_0^{(1)}$ and $U_1^{(1)}$ (the Q factors) such that $T_q^{(1)}(U_q^{(1)})^H = (O \mid \widehat{T}_q^{(1)})$ and $\widehat{T}_q^{(1)}$ is an $n_q^{(1)} \times \rho_q^{(1)}$ matrix of full rank $n_q^{(1)}$ for $n_q^{(1)} \le \rho_q^{(1)} \le \rho$ and $q = 0, 1$. Then we write $\widehat{U}^{(1)} = \mathrm{diag}(U_0^{(1)}, U_1^{(1)})$ and obtain $M^{(1)} = \widehat{M}^{(1)}\widehat{U}^{(1)}$,

$$\widehat{M}^{(1)} = M^{(1)}(\widehat{U}^{(1)})^H = \begin{pmatrix} \Sigma_{00}^{(1)} & \Sigma_{01}^{(1)} & O & S_{01}^{(1)}\widehat{T}_1^{(1)} \\ O & S_{10}^{(1)}\widehat{T}_0^{(1)} & \Sigma_{10}^{(1)} & \Sigma_{11}^{(1)} \end{pmatrix}$$

and $\widehat{M}^{(1)}P_1 = \begin{pmatrix} \Sigma_{00}^{(1)} & O & \Sigma_{01}^{(1)} & S_{01}^{(1)}\widehat{T}_1^{(1)} \\ O & \Sigma_{10}^{(1)} & S_{10}^{(1)}\widehat{T}_0^{(1)} & \Sigma_{11}^{(1)} \end{pmatrix}$ for a permutation matrix P_1. Now write $\Sigma_q^{(1)} = \Sigma_{q0}^{(1)}$ for $q = 0, 1$, $\Sigma^{(1)} = \mathrm{diag}(\Sigma_q^{(1)})_{q=0}^1$, $U^{(1)} = \widehat{U}^{(1)}P_1$, and $M_2 = \begin{pmatrix} \Sigma_{01}^{(1)} & S_{01}^{(1)}\widehat{T}_1^{(1)} \\ S_{10}^{(1)}\widehat{T}_0^{(1)} & \Sigma_{11}^{(1)} \end{pmatrix}$ and obtain

$$M^{(1)} = (\Sigma^{(1)} \mid M_2)U^{(1)}. \tag{5.3}$$

We cannot compress the 2×2 block matrix M_2 any further because each of its diagonal blocks $\Sigma_{q1}^{(1)}$, $q = 0, 1$, has at most ρ columns. We merge these two blocks to rewrite M_2 as a 1×1 block matrix, to which we refer hereafter as $\Sigma^{(2)}$. Now we combine equations (5.2) and (5.3) and write $U = U^{(0)}\,\mathrm{diag}(I, U^{(1)})$ to obtain $M = M^{(0)} = DU$ for $D = (\Sigma^{(0)} \mid \Sigma^{(1)} \mid \Sigma^{(2)})$, $\Sigma^{(0)} = \mathrm{diag}(\Sigma_q^{(0)})_{q=0}^3$, $\Sigma^{(1)} = \mathrm{diag}(\Sigma_0^{(1)}, \Sigma_1^{(1)})$, and so

$$D = \left(\begin{array}{cccc|cc|c} \Sigma_0^{(0)} & & & & \Sigma_0^{(1)} & & \\ & \Sigma_1^{(0)} & & & & & \\ & & \Sigma_2^{(0)} & & & & \Sigma_0^{(2)} \\ & & & \Sigma_3^{(0)} & & \Sigma_1^{(1)} & \end{array}\right),$$

where (cf. (5.1)) $\Sigma_0^{(1)} = \begin{pmatrix} \Sigma_{01} & S_{01}\widehat{T}_1 \\ S_{10}\widehat{T}_0 & \Sigma_{11} \end{pmatrix} U_1^{(0)}$, $\Sigma_1^{(1)} = \begin{pmatrix} \Sigma_{21} & S_{23}\widehat{T}_3 \\ S_{32}\widehat{T}_2 & \Sigma_{31} \end{pmatrix} U_1^{(1)}$, and $\Sigma_q^{(0)} = \Sigma_{q0}$ for $q = 0, 1, 2, 3$. This completes the recursive process of compression and merging of the 4×4 block matrix M.

Given an $m \times n$ balanced ρ-HSS matrix M with k diagonal blocks Σ_q of sizes $m_q \times n_q$ for $q = 0, \dots, k-1$, we generalize this recursive process and successively obtain matrices $U^{(j)}$ (unitary), $\Sigma^{(j)}$ (block diagonal), and $M_{j+1} = M^{(j+1)}$ (basically ρ-HSS) for $j = 0, \dots, l-1$ and $l = \lceil \log_2(k) \rceil$. At the end we arrive at the factorization $M = DU$. Here $U = U^{(0)} \prod_{j=0}^{l-1} \mathrm{diag}(I, U^{(j)})$ is a unitary matrix, $D = (\Sigma^{(0)} \mid \Sigma^{(1)} \mid \dots \mid \Sigma^{(l-1)})$, $\Sigma^{(j)} = \mathrm{diag}(\Sigma_q^{(j)})_{q=0}^{k(j)-1}$, $\Sigma_q^{(0)} = \Sigma_{q0}$ is an $m_q \times (n_q - \rho_q^{(0)})$ matrix for $\rho_q^{(0)} \le \rho$ and $q = 0, \dots, k-1$, whereas $\Sigma_q^{(j)}$ are $m_q^{(j)} \times \rho_q^{(j)}$ matrices for $q = 0, \dots, k(j)-1$, where $k(j) \le \lceil k/2^j \rceil$, $m_q^{(j)} = m_{2q-1}^{(j-1)} + m_{2q}^{(j-1)}$, $m_q^{(0)} = m_q$ for $q < k$, $m_q^{(0)} = 0$ for $q \ge k$, $\rho_q^{(j)} \le \rho$ for $q = 0, \dots, k(j)-1$ and $j = 1, \dots, l-1$.

5.5 Reduction to an auxiliary linear system

Observe that

$$\beta(M) \le \beta(D) + \beta(U) + \sum_{j=0}^{l-1} (a(U^{(j)}) + a(\Sigma^{(j)})) \tag{5.4}$$

where $a(W)$ denotes the arithmetic cost of computing a matrix W. For the solution of a linear system $D\mathbf{y} = \mathbf{b}$ we need the entries of the matrices $\Sigma^{(j)}$, and so bound (5.4) includes the terms $a(\Sigma^{(j)})$.

The value $a(U^{(0)})$ is equal to the arithmetic cost of computing the QR factorization of the $n_q \times \rho_q$ matrices $T_1^H, \dots, T_{k-1}^H$ where $\rho_q \le \rho$ for all q, and so $a(U^{(0)}) = O(\sum_{q=0}^{k-1} \rho_q^2 n_q) = O(\rho^2 n)$. The values $a(U^{(j)})$ for $j > 0$ are bounded similarly, except that we compute the QR factors of $k(j) \le \lceil k/2^j \rceil$ matrices of sizes at most $\rho \times \rho$ for every $j > 0$, and so $\sum_{j=1}^{l-1} a(U^{(j)}) = O(k\rho^3)$ and

$$\sum_{j=0}^{l-1} a(U^{(j)}) = O((n + k\rho)\rho^2). \tag{5.5}$$

Next estimate $\beta(U) = \alpha(U^H)$. At the jth merging the block diagonal matrix $U^{(j)}$ has $k(j) \le \lceil k/2^j \rceil$ diagonal blocks, which are the Q factors of the QR factorization for the matrices of sizes of at most $\rho \times n_q$ for $q = 0, \dots, k-1$ and $j = 0$ and of at most $(2\rho) \times \rho$ for all positive j and all q. Therefore $\alpha(U^{(0)}) = O(\rho \sum_{q=0}^{k-1} n_q) = O(n\rho)$, whereas $\alpha(U^{(j)}) \le ck\rho^2/2^j$ for a constant c and all $j > 0$, and so

$$\beta(U) = O((n + k\rho)\rho), \tag{5.6}$$

dominated by bound (5.5). It remains to estimate $\beta(D)$ and $a(\Sigma^{(j)})$ for $j = 0, \dots, l-1$.

Write $a(\Sigma^{(j)}) = a_0(\Sigma^{(j)}) + a_1(\Sigma^{(j)})$ where $a_0(\Sigma^{(j)})$ and $a_1(\Sigma^{(j)})$ denote the arithmetic cost of computing the block products $\Sigma^{(j)} U^{(j)} = \mathrm{diag}(\Sigma_p^{(j)} U_p^{(j)})_{p=0}^{k(j)}$ and the blocks appended to the diagonal blocks at the jth merging, respectively.

Compute the block product $\Sigma^{(j)} U^{(j)}$ by using less than $2\sum_{q=0}^{k-1} m_q n_q \rho \le 2mn_+\rho$ arithmetic operations for $j = 0$ and less than $2\sum_{q=0}^{k(j)-1} m_q^{(j)} n_q^{(j)} \rho \le 2m\rho^2$ for every positive $j = 0$. Hence $\sum_{j=0}^{l=1} a_0(\Sigma^{(j)}) \le 2(n_+ + l\rho)m\rho$.

Next observe that $a_1(\Sigma^{(0)})$ amounts to the cost of computing the products $S_{10}\widehat{T}_0$, $S_{01}\widehat{T}_1$, $S_{32}\widehat{T}_2$, and $S_{23}\widehat{T}_3$ in the displayed case of (5.1), where $k = 4$. In the general case the two factors of such a product in a block row q have sizes of at most $m_q \times \rho$ and $\rho \times \rho$, respectively, for $q = 0, \dots, k-1$. Therefore $a_1(\Sigma^{(0)}) < 2\rho^2 \sum_{q=0}^{k-1} m_q = 2\rho^2 m$. Likewise $a_1(\Sigma^{(j)}) < 2\rho^2 m$ for every j because the overall number of rows of the factors $S_{pq}^{(j)}$ is equal to m, whereas the factors $\widehat{T}_q^{(j)}$ have sizes at most $\rho \times \rho$. Consequently $\sum_{j=0}^{l-1} a_1(\Sigma^{(j)}) < 2l\rho^2 m$ and

$$\sum_{j=0}^{l-1} a(\Sigma^{(j)}) < 2(n_+ + 2l\rho)m\rho. \tag{5.7}$$

To estimate $\beta(M)$ it remains to bound $\beta(D)$.

5.6 The second recursive factorization

By following [CGS07, Section 3] compute the QR factors of the matrices $\Sigma_q^{(0)}$ for $q = 0, \dots, k-1$, that is compute some unitary matrices $V_q^{(0)}$ (in factored form) and $\widehat{\rho}_q^{(0)} \times \widehat{\rho}_q^{(0)}$ nonsingular upper triangular matrices $\widehat{\Sigma}_q^{(0)}$ such that $\Sigma_q^{(0)} = V_q^{(0)} \begin{pmatrix} O \\ \widehat{\Sigma}_q^{(0)} \end{pmatrix}$ and $\widehat{\rho}_q^{(0)} \le \min\{m_q, n_q\}$ for all q. Write $V^{(0)} = \mathrm{diag}(V_q^{(0)})_{q=0}^{k-1}$, $D_1^{(j)} = (V^{(0)})^H \Sigma^{(j)}$ for $j = 0, \dots, l-1$, and $\widehat{D} = (D_1^{(1)} \mid \dots \mid D_1^{(l-1)})$. Note that all nonzero blocks of the matrices $\Sigma^{(j)}$ for all positive j keep their sizes and positions and do not increase their ranks in the transition to the matrices $D_1^{(j)}$ and that the matrix $D_1^{(0)}$ has exactly $\sum_{q=0}^{k-1} \widehat{\rho}_q^{(0)}$ nonzero rows. Remove all rows of the matrix $\widehat{D}$ sharing indices with these rows and let D_1 denote the resulting matrix. Substitution reduces the solution of a linear system $D\mathbf{y} = \mathbf{b}$ to computing the matrices $D_1^{(j)}$, $j = 0, \dots, l-1$, and to solving two linear systems of equations with the matrices D_1 and $D_1^{(0)}$. Recursively apply this process to the matrix D_1. Note that in l recursive steps the substitution reduces the original linear system $D\mathbf{y} = \mathbf{b}$ to block diagonal systems with triangular diagonal blocks.

5.7 Completion of the proof of the theorem

We have showed that $\beta(D) \le \sigma + \sum_{j=0}^{l-1}(\beta(V^{(j)}) + a(V^{(j)}) + a(D_j))$. Here σ denotes the cost of the substitution and of the solution of all triangular linear systems involved, $V^{(j)}$ denotes the unitary multiplier computed at the jth stage of the above process for $j = 0, \dots, l-1$, and $a(D_j)$ denotes the arithmetic cost of the multiplication of the matrix $V^{(j)}$ by the submatrix (we denote it $\widehat{D}_j$) obtained by removing the entries of the block column $D_j^{(j)}$ from the matrix D_j. Hereafter let $\nu(W)$ denote the overall number of the nonzero entries of a matrix W, observe that $\sigma < 2\nu(D)$, and obtain

$$\sigma < 2\nu(D) \le 2s + 2(l-1)m\rho. \tag{5.8}$$

The arithmetic cost of the computation of the unitary multipliers $V^{(j)}$ is $O(\sum_{q=0}^{k-1} m_q n_q^2) = O(mn_+^2)$ at Stage 0 of the process and is $O(\sum_{p=0}^{k(j)-1} m_p^{(j)}(n_p^{(j)})^2)$ at its jth stage for every positive j. Here $n_p^{(j)} \le \rho$, and the sum $\sum_{p=0}^{k(j)-1} m_p^{(j)}$ is monotone decreasing from m as j increases from 0. Therefore $\sum_{j=1}^{l-1} a(V^{(j)}) = O(lm\rho^2)$, and so

$$\sum_{j=0}^{l-1} a(V^{(j)}) = O(n_+^2 + l\rho^2)m. \tag{5.9}$$

This bound strongly dominates the sum $\sum_{j=0}^{l-1}(\beta(V^{(j)}) = \sum_{j=0}^{l-1} \alpha((V^{(j)})^H)$.

To compute the product $V^{(j)}\widehat{D}_j$ we need $O(\rho m n_+)$ arithmetic operations for $j = 0$ and $O(m\rho^2)$ for any $j > 0$. Consequently we perform this computation for $j = 0, \dots, l-1$ by using $O((n_+ + l\rho)m\rho)$ arithmetic operations, which matches (5.7). Combine estimates (5.4)–(5.9) to deduce bound (4.3).

To prove part (iii) of the theorem, apply bounds (4.2) and (4.3) to the transposed matrix M^T, thus extending them to the case where the matrix M is the transpose of a balanced ρ-HSS matrix that has $n_q \times m_q$ diagonal blocks Σ_q for $q = 0, \dots, k-1$.

Remark 5.1. *At the jth stage of the merging process we deal with matrix D_j, which has at most $l_j = \lceil n_+(j)/n \rceil \le 2^{l-j}$ block columns, each of at most ρ columns, that is at most $\rho 2^{l-j}$ columns overall. Suppose we stop merging process at this stage and compute the QR factors of the matrix at the arithmetic cost $O(m\rho^2 2^{l-j+1})$. For $\rho = O(\log(n))$ this modification does not affect the cost bound $\beta(M) = O(n\log^3(n))$ of Corollary 4.2 where the integer $l - j$ is bounded from above by a constant and even where $l - j = O(\log(\log(n)))$.*

6 Extension to tridiagonal blocks

We wish to approximate CV matrices by balanced ρ-HSS matrices, but this only works when we extend this class. We are going to do this and to extend Theorem 4.5 and Corollary 4.2 accordingly. We begin with demonstration by example (cf. Figure 5 and [B10]).

Example 6.1. *The following 8×8 block matrix turns into a block tridiagonal matrix if we glue together its lower and upper boundaries,*

$$M = \begin{pmatrix} \Sigma_0 & B_0 & O & O & O & O & O & A_0 \\ A_1 & \Sigma_1 & B_1 & O & O & O & O & O \\ O & A_2 & \Sigma_2 & B_2 & O & O & O & O \\ O & O & A_3 & \Sigma_3 & B_3 & O & O & O \\ O & O & O & A_4 & \Sigma_4 & B_4 & O & O \\ O & O & O & O & A_5 & \Sigma_5 & B_5 & O \\ O & O & O & O & O & A_6 & \Sigma_6 & B_6 \\ B_7 & O & O & O & O & O & A_7 & \Sigma_7 \end{pmatrix}. \tag{6.1}$$

Define the eight tridiagonal blocks,

$$\Sigma_0^{(c)} = \begin{pmatrix} B_7 \\ \Sigma_0 \\ A_1 \end{pmatrix},\ \Sigma_1^{(c)} = \begin{pmatrix} B_0 \\ \Sigma_1 \\ A_2 \end{pmatrix},\ \Sigma_2^{(c)} = \begin{pmatrix} B_1 \\ \Sigma_2 \\ A_3 \end{pmatrix},\ \Sigma_3^{(c)} = \begin{pmatrix} B_2 \\ \Sigma_3 \\ A_4 \end{pmatrix},$$

$$\Sigma_4^{(c)} = \begin{pmatrix} B_3 \\ \Sigma_4 \\ A_5 \end{pmatrix},\ \Sigma_5^{(c)} = \begin{pmatrix} B_4 \\ \Sigma_5 \\ A_6 \end{pmatrix},\ \Sigma_6^{(c)} = \begin{pmatrix} B_5 \\ \Sigma_6 \\ A_7 \end{pmatrix},\ \text{and } \Sigma_7^{(c)} = \begin{pmatrix} B_6 \\ \Sigma_7 \\ A_0 \end{pmatrix}.$$

Here $\Sigma_1^{(c)}$, $\Sigma_2^{(c)}$, $\Sigma_3^{(c)}$, $\Sigma_4^{(c)}$, $\Sigma_5^{(c)}$, *and* $\Sigma_6^{(c)}$ *are six blocks of the matrix* M *of equation (6.1), whereas each of the two submatrices* $\Sigma_0^{(c)}$ *and* $\Sigma_7^{(c)}$ *has been made up of a pair of blocks of this matrix. Each pair, however, turns into a single block if we glue together the lower and upper boundaries of the matrix* M. *We still call a block column* basic *and denote it* M_q *if* $\mathcal{C}(M_q) = \mathcal{C}(\Sigma_q^{(c)})$, *that is if it shares the column indices with the diagonal block* Σ_q *and the* tridiagonal block $\Sigma_q^{(c)}$. *The* admissible block $N_q^{(c)}$ *(playing the role of a basic neutered block column of Definition 4.2) complements the tridiagonal block* $\Sigma_q^{(c)}$ *in its basic block column. The admissible block* $N_q^{(c)}$ *is filled with zeros in the case of the matrix* M *of (6.1) for every* q, $q = 0, \dots, 7$, *but not so in the case of general* 8×8 *block matrix embedding the matrix* M *of (6.1). Here are some sample unions of the tridiagonal blocks of the matrix* M *of (6.1),* $\Sigma_{0,1,\dots,7}^{(c)} = M$,

$$\Sigma_{0,1,2,3}^{(c)} = \begin{pmatrix} B_7 & O & O & O \\ \Sigma_0 & B_0 & O & O \\ A_1 & \Sigma_1 & B_1 & O \\ O & A_2 & \Sigma_2 & B_2 \\ O & O & A_3 & \Sigma_3 \\ O & O & O & A_4 \end{pmatrix},\ \Sigma_{0,1}^{(c)} = \begin{pmatrix} B_7 & O \\ \Sigma_0 & B_0 \\ A_1 & \Sigma_1 \\ O & A_2 \end{pmatrix},\ \text{and } \Sigma_{2,3}^{(c)} = \begin{pmatrix} B_1 & O \\ \Sigma_2 & B_2 \\ A_3 & \Sigma_3 \\ O & A_4 \end{pmatrix}.$$

Let us generalize this demonstration. Assume a block matrix M with k diagonal blocks Σ_q, of sizes $m_q^{(c)} \times n_q$, for $q = 0, \dots, k-1$, and glue together its lower and upper block boundaries. Then each diagonal block, including the two extremal blocks Σ_0 and Σ_{k-1}, has exactly two *neighboring blocks* in its basic block column: they are given by the pair of the subdiagonal and superdiagonal blocks. Define the *tridiagonal blocks* $\Sigma_0^{(c)}, \dots, \Sigma_{k-1}^{(c)}$ of sizes $m_q^{(c)} \times n_q$ by combining such triples of blocks where $m_q^{(c)} = m_{q-1 \mod k} + m_q + m_{q+1 \mod k}$, $q = 0, \dots, k-1$. Write $m^{(c)} = \sum_{q=0}^{k-1} m_q^{(c)}$ and note that $m^{(c)} = 3m$ because the number of rows in each of the three block diagonals sums to m. Therefore $s^{(c)} = \sum_{q=0}^{k-1} m_q^{(c)} n_q \le m^{(c)} n_+ \le 3mn_+$. The complements of the tridiagonal blocks in their basic block columns are also blocks, called *admissible* (cf. [B10]).

Working with tridiagonal rather than diagonal blocks, we extend our definitions of the unions of blocks, recursive and balanced merging, basically ρ-neutered, balanced ρ-HSS, and ρ-HSS matrices (cf. Definitions 4.2, 4.3, and 4.4) as well as basically (ϵ, ρ)-neutered, balanced (ϵ, ρ)-HSS, and (ϵ, ρ)-HSS matrices, and we call such matrices *extended*. Can we extend Theorem 4.5 and Corollary 4.2 to the case of extended balanced ρ-HSS matrices M where we replace the integer parameters m and s by $m^{(c)} = 3m$ and $s^{(c)} \le m^{(c)} n_+ = 3mn_+$, respectively? The extension of part (i) of Theorem 4.5 is immediate, but to extend the algorithms supporting its part (ii) we must impose some restriction on the input matrix M.

Definition 6.1. *An extended balanced* ρ*-HSS matrix is* hierarchically regular *if all its diagonal blocks at the second factorization stage of the associated balanced merging process have full rank. This matrix is* hierarchically well conditioned *if these blocks are also well conditioned.*

Theorem 6.1. *Suppose that the matrix* M *in Theorem 4.5 is replaced by an extended* $m \times n$ *balanced* ρ*-HSS matrix* $M^{(c)}$ *and also suppose that the integer parameters* m *and* s *in bounds (4.2) on* $\alpha(M)$ *and (4.3) on* $\beta(M)$ *are replaced by* $m^{(c)} = 3m$ *and* $s^{(c)} \le 3mn_+$, *respectively. (i) Then bound (4.2) still holds. (ii) Moreover bound (4.3) holds if* $m = n$ *and if the matrix* M *is hierarchically regular.*

Proof. Revisit the proof of the bound on $\alpha(M)$ and $\beta(M)$ of Theorem 4.5, replacing the integer parameters m and $\bar{s}^{(j)}$ according to the assumptions of the new theorem. Then verify that with the only exception, all auxiliary and final bounds remain valid. The exception is the impact of the QR factorizations at the second factorization stage in our proof of bound (4.3) on $\beta(M)$. Because of the transition to the tridiagonal blocks, the sizes and ranks of the nonzero blocks of the matrices $\Sigma^{(j)}$ for positive j can increase in the transition to the matrices $D_1^{(j)}$. To avoid this increase, we restrict the QR factorizations at that stage to the diagonal blocks and use the computed triangular factors as the pivot blocks to eliminate the other entries of the tridiagonal blocks in these columns by means of substitution. We readily verify that the recipe works (that is we avoid divisions by 0) and still supports bound (4.3) where the matrix M is hierarchically regular. □

Corollary 6.1. *Under the assumptions of Theorem 6.1 suppose that* $k\rho = O(n)$ *and* $n_+ + \rho = O(\log(n))$. *Then* $\alpha(M) = O((m+n)\log^2(n))$ *and* $\beta(M) = O(n\log^3(n))$.

Remark 6.1. *Clearly an extended balanced HSS process supporting Theorem 6.1 can fail numerically unless the input matrix is hierarchically regular and hierarchically well conditioned. It is a challenge to investigate how frequently this necessary condition is sufficient. Here is a simple initial observation. Denote by* ∇ *the maximum norm of the inverses of the triangular factors in the recursive HSS process. Then the solution of the associated triangular linear systems of equations magnifies the input error by a factor of* ∇^l, $l \le \log_2 n$. *Even for reasonably bounded values* ∇ *this growth can be fairly large but never exponential.*

7 Approximation of the CV and CV^T matrices by HSS matrices and algorithmic implications

7.1 Small-rank approximation of certain Cauchy matrices

Definition 7.1. *(See [CGS07, page 1254].) For a* separation bound $\theta < 1$ *and a complex* separation center c, *two complex points* s *and* t *are* (θ, c)-separated *from one another if* $|\frac{t-c}{s-c}| \le \theta$. *Two sets of complex numbers* $\mathcal{S}$ *and* $\mathcal{T}$ *are* (θ, c)-separated *from one another if every two points* $s \in \mathcal{S}$ *and* $t \in \mathcal{T}$ *are* (θ, c)*-separated from one another.* $\delta_{c,\mathcal{S}} = \min_{s\in\,\mathcal{S}} |s-c|$ *denotes the distance from the center* c *to the set* $\mathcal{S}$.

Lemma 7.1. *(See [R85] and [CGS07, equation (2.8)].) Suppose two complex values* s *and* t *are* (θ, c)*-separated from one another for* $0 \le \theta < 1$ *and a complex* center c *and write* $q = \frac{t-c}{s-c}$, $|q| \le \theta$. *Then for every positive integer* ρ *it holds that*

$$\frac{1}{s-t} = \frac{1}{s-c}\sum_{h=0}^{\rho-1}\frac{(t-c)^h}{(s-c)^h} + \frac{q_\rho}{s-c} \text{ where } |q_\rho| = \frac{|q|^\rho}{1-|q|} \le \frac{\theta^\rho}{1-\theta}. \tag{7.1}$$

Proof. $\frac{1}{s-t} = \frac{1}{s-c}\frac{1}{1-q}$, $\frac{1}{1-q} = \sum_{h=0}^{\infty} q^h = (\sum_{h=0}^{\rho-1} q^h + \sum_{h=\rho}^{\infty} q^h) = (\sum_{h=0}^{\rho-1} q^h + \frac{q^\rho}{1-q})$. □

Corollary 7.1. *(Cf. [CGS07, Section 2.2] and [B10].) Suppose two sets of* $2n$ *distinct complex numbers* $\mathcal{S} = \{s_0, \dots, s_{m-1}\}$ *and* $\mathcal{T} = \{t_0, \dots, t_{n-1}\}$ *are* (θ, c)*-separated from one another for* $0 < \theta < 1$ *and a* global complex center c. *Define the Cauchy matrix* $C = (\frac{1}{s_i - t_j})_{i,j=0}^{m-1,n-1}$ *and write* $\delta = \delta_{c,\mathcal{S}} = \min_{i=0}^{m-1} |s_i - c|$ *(cf. Definition 7.1). Fix a positive integer* ρ *and define the* $m \times \rho$ *matrix* $F = (1/(s_i-c)^{\nu+1})_{i,\nu=0}^{m-1,\rho-1}$ *and the* $n \times \rho$ *matrix* $G = ((t_j - c)^\nu)_{j,\nu=0}^{n-1,\rho-1}$. *(We can compute these matrices by using* $(m+n)\rho + m$ *arithmetic operations.) Then*

$$C = FG^T + E,\ |E| \le \frac{\theta^\rho}{(1-\theta)\delta}. \tag{7.2}$$

Proof. Apply (7.1) for $s = s_i$, $t = t_j$, and all pairs (i, j) to deduce (7.2). □

Remark 7.1. *Assume an $m \times n$ Cauchy matrix $C = (\frac{1}{s_i - t_j})_{i,j=0}^{m-1,n-1}$ with $m+n$ distinct knots $s_0, \dots, s_{m-1}, t_0, \dots, t_{n-1}$. Then* $\mathrm{rank}(C) = \min\{m,n\}$ *(cf. Theorem 3.1). Further assume that the sets $\mathcal{S} = \{s_0, \dots, s_{m-1}\}$ and $\mathcal{T} = \{t_0, \dots, t_{n-1}\}$ are (θ, c)-separated from one another for a global complex center c and $0 < \theta < 1$ such that the value $(1-\theta)\delta/\sqrt{mn}$ is not small. Then by virtue of the corollary the matrix C, having full rank, can be closely approximated by a matrix FG^T of a smaller rank $\rho < \min\{m,n\}$, and therefore is ill conditioned. Furthermore if we have such (θ, c)-separation just for a $k \times l$ submatrix $C_{k,l}$ of the matrix C (this implies that* $\mathrm{nrank}(C_{k,l}) \le \rho$*), then it follows that* $\mathrm{nrank}(C) \le m - k + n - l + \rho$*. Consequently if $m - k + n - l + \rho < \min\{m,n\}$, then again the matrix C is ill conditioned. This class of ill conditioned Cauchy matrices contains a large subclass of CV and CV^T matrices. In particular a CV matrix is ill conditioned if all its knots s_i or all knots s_i of its submatrix of a large size lie far enough from the unit circle $\{z : |z| = 1\}$, because in this case the origin serves as a global center for the matrix or the submatrix.*

7.2 Block partition of a Cauchy matrix

Generally neither CV matrix nor its submatrices of a large size have global separation centers. So instead of the approximation of a CV matrix by a low-rank matrix we seek its approximation by an extended balanced ρ-HSS matrix for a bounded integer ρ. We first fix a reasonably large integer k and then partition the complex plane into k congruent sectors sharing the origin 0, which induces a *uniform k-partition* of the knot sets $\mathcal{S}$ and $\mathcal{T}$ and thus a block partition of the associated Cauchy matrix. In the next subsection we specialize these partitions to the case of a CV matrix.

Definition 7.2. *$\mathcal{A}(\phi, \phi') = \{z = \exp(\psi\sqrt{-1}) : 0 \le \phi \le \psi < \phi' \le 2\pi\}$ is the* semi-open arc *of the unit circle $\{z : |z| = 1|\}$ having* length *$\phi' - \phi$ and the endpoints $\tau = \exp(\phi\sqrt{-1})$ and $\tau' = \exp(\phi'\sqrt{-1})$. $\Gamma(\phi, \phi') = \{z = r\exp(\psi\sqrt{-1}) : r \ge 0,\ 0 \le \phi \le \psi < \phi' \le 2\pi\}$ is the* semi-open sector *bounded by the two rays from the origin to the two endpoints of the arc. $\bar{\Gamma}(\phi, \phi')$ denotes the exterior (that is the complement) of this sector.*

Fix a positive integer l_+, write $k = 2^{l_+}$, $\phi_q = 2q\pi/k$, and $\phi'_q = \phi_{q+1 \mod k}$, partition the unit circle $\{z : |z| = 1|\}$ by k equally spaced points $\phi_0, \dots, \phi_{k-1}$ into k semi-open arcs $\mathcal{A}_q = \mathcal{A}(\phi_q, \phi'_q)$, each of the length $2\pi/k$, and define the semi-open sectors $\Gamma_q = \Gamma(\phi_q, \phi'_q)$ for $q = 0, \dots, k-1$. Now assume the polar representation $s_i = |s_i|\exp(\mu_i\sqrt{-1})$ and $t_j = |t_j|\exp(\nu_j\sqrt{-1})$, and reenumerate the knots in the counter-clockwise order of the angles μ_i and ν_j beginning with the knots in the sector $\Gamma(\phi_0, \phi'_0)$ and breaking ties arbitrarily. Induce the block partition of a Cauchy matrix $C = (C_{p,q})_{p,q=0}^{k-1}$ and its partition into basic block columns $C = (C_0 \mid \dots \mid C_{k-1})$ such that $C_{p,q} = \left(\frac{1}{s_i - t_j}\right)_{s_i \in \Gamma_p, t_j \in \Gamma_q}$ and $C_q = \left(\frac{1}{s_i - t_j}\right)_{s_i \in \{0,\dots,n-1\}, t_j \in \Gamma_q}$ for $p, q = 0, \dots, k-1$. Further, for every q define the diagonal block $\Sigma_q = C_{q,q}$, its two neighboring blocks $C_{q-1 \mod k,q}$ and $C_{q+1 \mod k,q}$, the tridiagonal block $\Sigma_q^{(c)}$ (made up of the block C_q and its two neighbors), and the admissible block $N_q^{(c)}$, which complements the tridiagonal block $\Sigma_q^{(c)}$ in its basic block column C_q.

7.3 $(0.5, c_q)$-separation of the tridiagonal and admissible blocks of a CV matrix

The following lemma can be readily verified (cf. Figure 6).

Lemma 7.2. *Suppose $0 \le \chi \le \phi \le \eta < \phi' < \chi' \le \pi/2$ and write $\tau = \exp(\phi\sqrt{-1})$, $c = \exp(\eta\sqrt{-1})$, and $\tau' = \exp(\phi'\sqrt{-1})$. Then $|c - \tau| = 2\sin((\eta - \phi)/2)$ and the distance from the point c to the sector $\bar{\Gamma}(\chi, \chi')$ is equal to $\sin(\psi)$, for $\psi = \min\{\eta - \chi, \chi' - \eta\}$.*

Now let C be actually a CV matrix $C_{\mathbf{s},f}$ for a fixed complex f such that $|f| = 1$, and so $t_j = f\omega_k^j$ for $\omega_k = \exp(2\pi\sqrt{-1}/k)$, $j = 0, \dots, n-1$. In this case all knots t_j are lying on the arcs $\mathcal{A}_q$ and each arc contains $\lceil n/k \rceil$ or $\lfloor n/k \rfloor$ knots.

Theorem 7.1. *(Cf. Figure 7.) Assume a uniform k-partition of the knot sets of a CV matrix above for $k \geq 12$. Let Γ'_q denote the union of the sector Γ_q and its two neighbors on both sides, that is $\Gamma'_q = \Gamma_{q-1 \mod k} \cup \Gamma_q \cup \Gamma_{q+1 \mod k}$, let $\bar{\Gamma}'_q$ denote its exterior, and let c_q denote the midpoints of the arcs $\mathcal{A}_q = \mathcal{A}(\phi_q, \phi'_q)$ for $q = 0, \dots, k-1$. Then for every q the arc $\mathcal{A}_q$ and the sector $\bar{\Gamma}'_q$ are (θ, c_q)-separated for $\theta = 2\sin((\phi'_q - \phi_q)/4)/\sin(\phi'_q - \phi_q)$.*

Proof. Apply Lemma 7.2 for $\phi = \phi_q$, $c = c_q$, $\phi' = \phi'_q$, $\chi = 2\phi_q - \phi'_q$, and $\chi' = 2\phi'_q - \phi_q$. □

Recall that $x/\sin x \approx 1$ as $x \approx 0$, and therefore $\theta = 2\sin((\phi'_q - \phi_q)/4)/\sin(\phi'_q - \phi_q) \approx 0.5$ provided $\phi'_q - \phi_q \approx 0$, that is the integer k is large enough. Note that for every q the admissible block $N_q^{(c)}$ is defined by the knots t_j lying on the arc $\mathcal{A}_q$ and the knots s_i lying in the sector $\bar{\Gamma}'_q$, apply Corollary 7.1. For every q, $q = 0, \dots, k-1$, write $\delta_q = \min_{s_i \in \bar{\Gamma}'_q} |s_i - c_q|$ and obtain the following result.

Corollary 7.2. *Assume a sufficiently large integer k, $2k < n$, and let a uniform k-partition of the knot sets of an $m \times n$ CV matrix C define k admissible blocks $N_0^{(c)}, \dots, N_{k-1}^{(c)}$. Then all of them have the $|E|$-ranks at most ρ, that is C is an extended basically $(|E|, \rho)$-neutered matrix, where $|E|$ and ρ satisfy bound (7.2) for $\theta \approx 0.5$ and $\delta = \min_{q=0}^{k-1} |\delta_q|$.*

One can seek separation of the blocks of CV matrices by using various alternative partitions of the complex plane, e.g., one can employ the following observations (not applied in this paper). Let $D(c, r) = \{z : |z - c| \leq r\}$ denote the *disc* on the complex plane with a center c, a radius r, and the exterior $\bar{D}(c, r) = \{z : |z - c| > r\}$. Lemma 7.2 implies the following result.

Theorem 7.2. *Assume the numbers θ, ϕ, ϕ', and c such that $0 < \theta < 1$, $0 \leq \phi < \phi' \leq 2\pi$, and $c = \exp(0.5(\phi' + \phi)\sqrt{-1})$ is the midpoint of the arc $\mathcal{A}(\phi, \phi')$. Write $r = r(\phi, \phi', \theta) = \frac{2}{\theta}\sin\frac{\phi' - \phi}{4}$ and let $\bar{D} = \bar{D}(c, r)$ denote the exterior of the disc $D(c, r)$. Then the two sets $\mathcal{A}(\phi, \phi')$ and $\bar{D}$ are (θ, c)-separated.*

7.4 Approximation of a CV matrix by a balanced ρ-HSS matrix and the complexity of approximate computations with CV matrices

The angles $2\pi/k$ of the k congruent sectors $\Gamma_0, \dots, \Gamma_{k-1}$ are recursively doubled in every merging. So Lemma 7.2 implies that $\delta \leq \delta_h = \sin(3\pi 2^h/k)$ after the hth merging, $h = 1, \dots, l$. Choose the integers $k = 2^{l_+}$ and $l < l_+$ such that the integer $k/2^l = 2^{l_+ - l}$ is reasonably large, to support separation with parameters θ of about 0.5 or less at all stages of recursive merging. Then $\delta_h \approx 3\pi 2^h/k$, and $\delta_{h+1}/\delta_h \approx 2$ for all h. Now Corollary 7.2 implies the following result.

Theorem 7.3. *The CV matrix C Corollary 7.2 is an extended balanced (ϵ, ρ)-HSS matrix where the values ϵ and ρ are linked by bound (7.2) for $\theta \approx 0.5$, $|E| = \epsilon$, $\delta = \delta_h \approx 3\pi 2^h/k$, and $h = 0, \dots, l$.*

Combine Corollary 6.1 with this theorem applied for $k = 2^{l_+}$ of order $n/\log(n)$, for ρ and $\log(1/\epsilon)$ of order $\log(n)$, and for $l < l_+$ such that the integer $l_+ - l$ is reasonably large (verify that the assumptions of the corollary are satisfied), and obtain the following result.

Theorem 7.4. *Assume an $m \times n$ CV matrix C and a positive ϵ such that $\log(1/\epsilon) = O(\log(n))$. Then $\alpha_\epsilon(C) = O((m+n)\log^2(n))$. If in addition $m = n$ and if the matrix C is ϵ-approximated by a hierarcically regular extended balanced ρ-HSS matrix, then $\beta_\epsilon(C) = O(n\log^3(n))$.*

Because of the dual role of the rows and columns in our constructions we can readily extend all our results (and in particular Theorem 7.4) from CV matrices C to CV^T matrices C^T.

Corollary 7.3. *The estimates of Theorem 7.4 also hold for a CV^T matrix C.*

Remark 7.2. *Suppose we extend diagonal blocks to v-diagonal blocks for an odd integer $v > 3$. How would this change our complexity bounds? The separation parameter θ would increase by a factor of v, but the implied decrease of the cost bound would be offset by the increase of the overall numbers of the entries in the diagonal blocks.*

8 Extensions and implementation

8.1 Computations with Vandermonde matrices and their transposes

Let us employ equations (3.9)–(3.12) to extend Theorem 7.4 to computations with Vandermonde matrices, their transposes, and polynomials.

Theorem 8.1. *Suppose that we are given two positive integers m and n and a vector $\mathbf{s} = (s_i)_{i=0}^{m-1}$ defining an $m \times n$ Vandermonde matrix $V = V_{\mathbf{s}}$. Write $s_+ = \max_{i=0}^{m-1} |s_i|$ and let $\log(1/\epsilon) = O(\log(m+n) + n\log(1+s_+))$. (i) Then*

$$\alpha_\epsilon(V) + \alpha_\epsilon(V^T) = O((m+n)(\rho \log^2(m+n) + n\log(1+s_+)). \tag{8.1}$$

(ii) Suppose that in addition $m = n$ and for some complex f, $|f| = 1$, the matrix $C_{\mathbf{s},f}$ of equation (3.8) is approximated by a hierarchically nonsingular extended balanced (ϵ, ρ)-HSS matrix. Then

$$\beta_\epsilon(V) + \beta_\epsilon(V^T) = O(n\rho^3 \log(n)). \tag{8.2}$$

(iii) Bounds (8.1) and (8.2) on $\alpha_\epsilon(V)$ and $\beta_\epsilon(V)$ can be applied also to the solution of Problems 1 and 2 of Section 3, respectively.

Proof. Combine Theorem 7.4, Corollary 7.3 and equations (3.9)–(3.12). The matrices $\text{diag}(\omega^{-j})_{j=0}^{n-1}$, $\text{diag}(f^{-j})_{j=0}^{n-1}$, $\Omega/\sqrt{n} = (\sqrt{n}\Omega^H)^{-1}$, and their inverses are unitary, and so multiplication by them makes no impact on the output error norms. Multiplication by the matrix $\text{diag}(s_i^n - f^n)_{i=0}^{m-1}$ can increase the value $\log_2(1/\epsilon)$ by at most $\log_2(1+s_+^n)$, whereas multiplication by its inverse for $m = n$ can increase this value by at most $\log_2(\Delta)$ for $\Delta = 1/\max_{f:\ |f|=1} \min_{i=0}^{m-1} |s_i^n - f^n|$. We can ensure that $\Delta \le 2m$ by choosing a proper value f, and so $\log_2(\Delta) \le 1 + \log_2(m)$. Such an increase makes no impact on the asymptotic bounds of Theorem 8.1, and so we complete the proof of parts (i) and (ii). Equations (3.1) and (3.2) extend the proof to part (iii). □

Note that the term $n\log(s_+)$ is dominated and can be removed from the bound on $\log(1/\epsilon)$ and (8.1) provided that $s_+ = 1 + O(\frac{\log^2(m+n)}{n})$.

8.2 Computations with other structured matrices, polynomials, and rational functions

The FMM/HSS techniques of [GR87], [DGR96], [CGR98], and [B10] combined with the algebraic techniques of [P90] and [Pa] work efficiently for other classes of structured matrices, and our complexity estimates can be extended and in some cases strengthened. Next we recall some relevant results from [Pa].

For $m \times n$ Toeplitz and Hankel matrices W one yields the bound $\beta_\epsilon(W) = O((n)\log^2(1/\epsilon)\log(n))$ where $m = n$ (see [Pa]). Our estimates for CV matrices can be extended to general Cauchy matrices $C_{\mathbf{s},\mathbf{t}}$ having arbitrary sets of knots s_i and t_j provided that we allow to increase the approximation errors by factors of $||C||\ ||C^{-1}||$ for $C = C_{\mathbf{s},f}$ or/and $C = C_{e,\mathbf{t}}$ for constants e and f of our choice such that $|e| = |f| = 1$. These estimates and the ones of the previous subsection are immediately extended to approximate solution of Problems 3 and 4 of rational interpolation and multipoint evaluation, assuming the latter restriction on the parameters e and f. Furthermore all algorithms and estimates can be extended from Cauchy to generalized Cauchy matrices $(f(s_i - t_j))_{i,j=0}^{m-1,n-1}$ for various functions $f(z)$ such as x^{-p} for a positive integer p, $\ln z$, and $\tan z$ (cf. [DGR96]).

Finally the classes of Toeplitz, Hankel, Vandermonde and Cauchy matrices W have been extended to larger classes of $m \times n$ matrices M that have structures of Toeplitz, Hankel, Vandermonde and Cauchy types. They allow compressed expressions through their displacements $AM - MB$ of small ranks d for operator matrices A and B fixed for each of the four structures, that is through at most $(m+n)(d+1)$ parameters per matrix (cf. [PW03]). The known fast algorithms for computations with Toeplitz, Hankel, Vandermonde and Cauchy matrices are extended to these classes. In particular our fast approximation algorithms are extended, with their complexity estimates changed into $\alpha_{\epsilon'}(M) = O(d\alpha_\epsilon(W))$ and $\beta_{\epsilon''}(M) = O(d\beta_\epsilon(W))$ for $\epsilon' = O(d|F|\epsilon)$ and $\epsilon'' = O(d|F|\ ||M^{-1}||\epsilon)$ (cf. [Pa]).

8.3 Simplified implementation by means of bounding the numerical ranks of the admissible blocks of a CV matrix

To implement our algorithms one can compute the centers c_q and the admissible blocks $\widehat{N}_q$ of bounded ranks throughout the merging process, but one can avoid a large part of these computations by following the papers [CGS07], [X12], [XXG12], and [XXCB14]. They bypass the computation of the centers c_q and immediately compute the HSS generators for the admissible blocks $\widehat{N}_q$, defined by HSS trees. The length (size) of the generators at every merging stage (represented by a fixed level of the tree) can be chosen equal to the available upper bound on the numerical ranks of these blocks or can be adapted empirically.

9 Conclusions

The papers [MRT05], [CGS07], [XXG12], and [XXCB14] combine the advanced FMM/HSS techniques with a transformation of matrix structures (traced back to [P90]) to devise numerically stable algorithms that compute approximate solution of Toeplitz, Hankel, Toeplitz-like, and Hankel-like linear systems of equations in nearly linear arithmetic time (versus cubic time of the classical numerical algorithms). We yield similar results for multiplication of Vandermonde and Cauchy matrices by a vector and the solution of linear systems of equations with these matrices (with the extensions to polynomial and rational evaluation and interpolation). The resulting decrease of the running time of the known approximation algorithms is by order of magnitude, from quadratic to nearly linear. Our study provides new insight into the subject and the background for further advances in [Pa], which include the extension of our results to Cauchy-like matrices and further acceleration of the known approximation algorithms in the case of Toeplitz inputs. The 2D FMM can help decrease similarly our cost bound (4.2) (cf. [B10, Section 3.6]).

Appendix

A FIGURES 2–7

In Figures 2–4 we mark by green color the diagonal blocks and by blue color the basic neutered block columns.

FIGURE 2

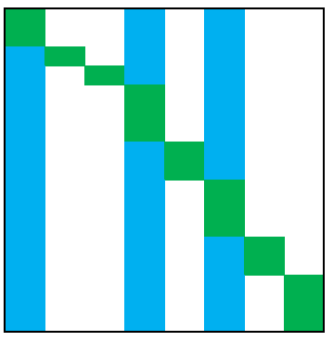
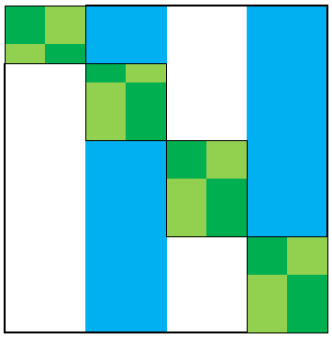
2.jpg 

Figure 2: FIGURE 2

In Figures 3 and 4 the pairs of smaller diagonal blocks (marked by light green color) are merged into their diagonal unions, each made up of four smaller blocks, marked by light and dark green colors.

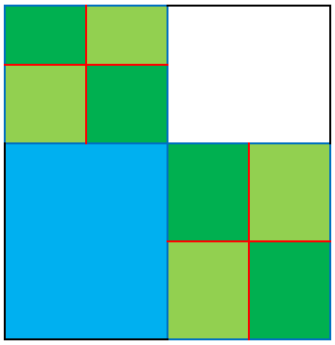
FIGURE 3



3.jpg

Figure 3: FIGURE 3

FIGURE 4



4.jpg

Figure 4: FIGURE 4

In Figure 5 the admissible blocks are shown by blue, each green diagonal block has two red neighboring blocks, and the triples of green and red blocks form the tridiagonal blocks.

FIGURE 5

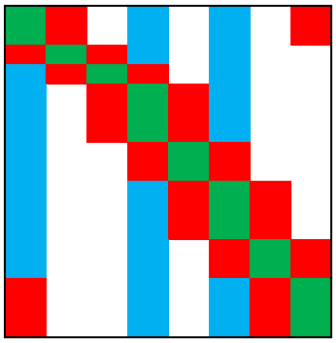
5.jpg 

Figure 5: FIGURE 5

In Figure 6 we mark by black color an arc of the unit circle $\{z : \ |z = 1|\}$, and we mark by blue the five line intervals $[0, \tau]$, $[0, c]$, $[0, \tau']$, $[\tau, c]$, and $[c, \tau]$. We mark by red the two line intervals bounding the intersection of the sector $\Gamma(\psi, \psi')$ and the unit disc $D(0, 1) = \{z : \ |z| \leq 1\}$ as well as the two perpendiculars from the point c onto these two bounding line intervals.

FIGURE 6

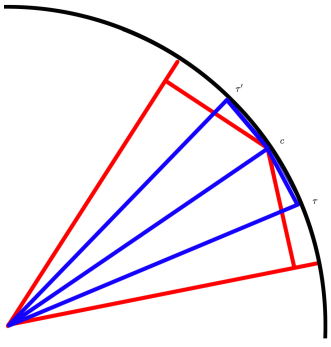


Figure 6: FIGURE 6

In Figure 7 we assume that $q = 0$, $\phi_0 = 0$ and $\exp(\phi'\sqrt{-1}) = t_h$.

FIGURE 7

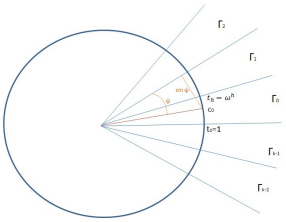
7.jpg 

Figure 7: FIGURE 7

References

[B99] R. Bracewell, *The Fourier Transform and Its Applications*. McGraw-Hill, New York, 1999 (3rd edition).

[B10] S. Börm, *Efficient Numerical Methods for Non-local Operators: $\mathcal{H}^2$-Matrix Compression, Algorithms and Analysis*, European Math. Society, 2010.

[BEGO08] T. Bella, Y. Eidelman, I. Gohberg, V. Olshevsky, Computations with Quasiseparable Polynomials and Matrices, *Theoretical Computer Science*, *Special Issue on Symbolic–Numerical Algorithms* (D. A. Bini, V. Y. Pan, and J. Verschelde editors), **409**, **2**, 158–179, 2008.

[BF00] D. A. Bini, G. Fiorentino, Design, Analysis, and Implementation of a Multiprecision Polynomial Rootfinder, *Numer. Algs.*, **23**, 127–173, 2000.

[B-O83] M. Ben-Or, Lower Bounds for Algebraic Computation Trees, *Proceedings of 15th Annual ACM Symposium on Theory of Computing (STOC'83)*, 80–86, ACM Press, New York, 1983.

[BP70] A. Björck, V. Pereyra, Solution of Vandermonde Systems of Equations, *Math. of Computation*, **24**, 893–903, 1970.

[BP94] D. Bini, V. Y. Pan, *Polynomial and Matrix Computations, Volume 1: Fundamental Algorithms*, Birkhäuser, Boston, 1994.

[BY13] L. A. Barba, R. Yokota, How Will the Fast Multipole Method Fare in Exascale Era? *SIAM News*, **46**, **6**, 1–3, July/August 2013.

[CDG06] S. Chandrasekaran, P. Dewilde, M. Gu, W. Lyons, T. Pals, A Fast Solver for HSS Representations via Sparse Matrices, *SIAM J. Matrix Anal. Appl.*, **29, 1**, 67–81, 2006.

[CGR98] J. Carrier, L. Greengard, V. Rokhlin, A Fast Adaptive Algorithm for Particle Simulation, *SIAM J. Scientific Computing*, **9**, 669–686, 1998.

[CGS07] S. Chandrasekaran, M. Gu, X. Sun, J. Xia, J. Zhu, A Superfast Algorithm for Toeplitz Systems of Linear Equations, *SIAM J. Matrix Anal. Appl.*, **29**, 1247–1266, 2007.

[DGR96] A. Dutt, M. Gu, V. Rokhlin, Fast Algorithms for Polynomial Interpolation, Integration, and Differentiation, *SIAM Journal on Numerical Analysis*, **33, 5**, 1689–1711, 1996.

[DV98] P. Dewilde and A. van der Veen, *Time-Varying Systems and Computations*, Kluwer Academic Publishers, Dordrecht, The Netherlands, 1998.

[EG02] Y. Eidelman, I. Gohberg, A Modification of the Dewilde–van der Veen Method for Inversion of Finite Structured Matrices, *Linear Algebra and Its Applications*, **343**, 419–450, 2002.

[EGH13] Y. Eidelman, I. Gohberg, I. Haimovici, *Separable Type Representations of Matrices and Fast Algorithms. Volume 1. Basics. Completion Problems. Multiplication and Inversion Algorithms. Volume 2. Eigenvalue method*, Birkhauser, 2013.

[F72] C. M. Fiduccia, Polynomial Evaluation via the Division Algorithm: The Fast Fourier Transform Revisited, *Proc. 4th Annual ACM Symp. on Theory of Computing (STOC'72)*, 88–93, 1972.

[G98] M. Gu, Stable and Efficient Algorithms for Structured Systems of Linear Equations, *SIAM J. Matrix Anal. Appl.*, **19**, 279–306, 1998.

[GI88] W. Gautschi, G. Inglese, Lower Bounds for the Condition Number of Vandermonde Matrices, *Numerische Mathematik*, **52**, 241–250, 1988.

[GKK85] I. Gohberg, T. Kailath, I. Koltracht, Linear Complexity Algorithms for Semiseparable Matrices, *Integral Equations and Operator Theory*, **8, 6**, 780–804, 1985.

[GGS87] A. Gerasoulis, M. D. Grigoriadis, L. Sun, A Fast Algorithm for Trummer's Problem, *SIAM Journal on Scientific and Statistical Computing*, **8, 1**, 135–138, 1987.

[GKO95] I. Gohberg, T. Kailath, V. Olshevsky, Fast Gaussian Elimination with Partial Pivoting for Matrices with Displacement Structure, *Mathematics of Computation*, **64**, 1557–1576, 1995.

[GL96] G. H. Golub, C. F. Van Loan, *Matrix Computations*, 3rd edition, The Johns Hopkins University Press, Baltimore, Maryland, 1996.

[GR87] L. Greengard, V. Rokhlin, A Fast Algorithm for Particle Simulation, *Journal of Computational Physics*, **73**, 325–348, 1987.

[GS66] W. Gentelman, G. Sande, Fast Fourier Transform for Fun and Profit, *Full Joint Comput. Conference*, **29**, 563–578, 1966.

[H72] E. Horowitz, A Fast Method for Interpolation Using Preconditioning, *Information Processing Letters*, **1, 4**, 157–163, 1972.

[LRT79] R. J. Lipton, D. Rose, R. E. Tarjan, Generalized Nested Dissection, *SIAM J. on Numerical Analysis*, **16**, **2**, 346–358, 1979.

[MB72] R. Moenck, A. Borodin, Fast Modular Transform via Division, *Proceedings of 13th Annual Symposium on Switching and Automata Theory*, 90–96, IEEE Computer Society Press, Washington, DC, 1972.

[MRT05] P. G. Martinsson, V. Rokhlin, M. Tygert, A Fast Algorithm for the Inversion of Toeplitz Matrices, *Comput. Math. Appl.*, **50**, 741–752, 2005.

[P64] F. Parker, Inverses of Vandermonde matrices, *Amer. Math. Monthly*, **71**, 410–411, 1964.

[P90] V. Y. Pan, On Computations with Dense Structured Matrices, *Math. of Computation*, **55, 191**, 179–190, 1990. Proceedings version in *Proc. Intern. Symposium on Symbolic and Algebraic Computation (ISSAC'89)*, 34–42, ACM Press, New York, 1989.

[P93] V. Y. Pan, Parallel Solution of Sparse Linear and Path Systems, in *Synthesis of Parallel Algorithms* (J.H. Reif, editor), Chapter 14, pp. 621–678, Morgan Kaufmann publishers, San Mateo, California (1993).

[P95] V. Y. Pan, An Algebraic Approach to Approximate Evaluation of a Polynomial on a Set of Real Points, *Advances in Computational Mathematics*, **3**, 41–58, 1995.

[P01] V. Y. Pan, *Structured Matrices and Polynomials: Unified Superfast Algorithms*, Birkhäuser/Springer, Boston/New York, 2001.

[P11] V. Y. Pan, Nearly Optimal Solution of Rational Linear Systems of Equations with Symbolic Lifting and Numerical Initialization, *Computers and Mathematics with Applications*, **62**, 1685–1706, 2011.

[P13] V. Y. Pan, Polynomial Evaluation and Interpolation and Transformations of Matrix Structures, *Proceedings of CASC 2013* (V.P. Gerdt et al. editors), *Lecture Notes in Computer Science*, **8136**, 273–287, Springer, Heidelberg (2013).

[Pa] V. Y. Pan, Transformations of Matrix Structures Work Again II, in arxiv:1311.3729[math.NA]

[PR93] V. Y. Pan, J. Reif, Fast and Efficient Parallel Solution of Sparse Linear Systems, *SIAM J. on Computing*, **22**, **6**, 1227–1250, 1993.

[PRT92] V. Y. Pan, J. H. Reif, S. R. Tate, The Power of Combining the Techniques of Algebraic and Numerical Computing: Improved Approximate Multipoint Polynomial Evaluation and Improved Multipole Algorithms, *33th Annual IEEE Symposium on Foundations of Computer Science (FOCS'92)*, 703–713, IEEE Computer Societty Press, 1992.

[PSLT93] V. Y. Pan, A. Sadikou, E. Landowne, O. Tiga, A New Approach to Fast Polynomial Interpolation and Multipoint Evaluation, *Computers and Math. (with Applications)*, **25, 9**, 25–30, 1993.

[PW03] V. Y. Pan, X. Wang, Inversion of Displacement Operators, *SIAM J. on Matrix Analysis and Applications*, **24, 3**, 660–677, 2003.

[PZHY97] V. Y. Pan, A. Zheng, X. Huang, Y.Yu, Fast Multipoint Polynomial Evaluation and Interpolation via Computation with Structured Matrices, *Annals of Numerical Math.*, **4**, 483–510, 1997.

[R85] V. Rokhlin, Rapid Solution of Integral Equations of Classical Potential Theory, *Journal of Computational Physics*, **60**, 187–207, 1985.

[S73] V. Strassen, Die Berechnungskomplexetät von elementarysymmetrischen Funktionen und von Interpolationskoeffizienten, *Numerische Mathematik*, **20, 3**, 238–251, 1973.

[S98] G. W. Stewart, *Matrix Algorithms, Vol I: Basic Decompositions*, SIAM, Philadelphia, 1998.

[T00] E.E. Tyrtyshnikov, Incomplete Cross-Approximation in the Mosaic-Skeleton Method, *Computing*, **64**, 367–380, 2000.

[VVGM] R. Vandebril, M. Van Barel, G. Golub, N. Mastronardi, A Bibliography on Semiseparable Matrices, *Calcolo*, **42, 3–4,** 249–270, 2005.

[VVM07] R. Vandebril, M. Van Barel, N. Mastronardi, *Matrix Computations and Semiseparable Matrices: Linear Systems* (Volume 1), The Johns Hopkins University Press, Baltimore, Maryland, 2007.

[VVM08] R. Vandebril, M. Van Barel, N. Mastronardi, *Matrix Computations and Semiseparable Matrices: Eigenvalue and Singular Value Methods* (Volume 2), The Johns Hopkins University Press, Baltimore, Maryland, 2008.

[X12] J. Xia, On the Complexity of Some Hierarchical Structured Matrix Algorithms, *SIAM J. Matrix Anal. Appl.*, **33**, 388–410, 2012.

[X13] J. Xia, Randomized Sparse Direct olvers, *SIAM J. Matrix Anal. Appl.*, **34**, 197–227, 2013.

[XXCB14] J. Xia, Y. Xi, S. Cauley, V. Balakrishnan, Superfast and Stable Structured Solvers for Toeplitz Least Squares via Randomized Sampling, *SIAM J. Matrix Anal. and Applications*, **35**, 44–72, 2014.

[XXG12] J. Xia, Y. Xi, M. Gu, A Superfast Structured Solver for Toeplitz Linear Systems via Randomized Sampling, *SIAM J. Matrix Anal. Appl.*, **33**, 837–858, 2012.