

2013

Structured Matrices and the Algebra of Displacement Operators

Ryan Takahashi
Harvey Mudd College

Recommended Citation

Takahashi, Ryan, "Structured Matrices and the Algebra of Displacement Operators" (2013). *HMC Senior Theses*. 45.
https://scholarship.claremont.edu/hmc_theses/45

This Open Access Senior Thesis is brought to you for free and open access by the HMC Student Scholarship at Scholarship @ Claremont. It has been accepted for inclusion in HMC Senior Theses by an authorized administrator of Scholarship @ Claremont. For more information, please contact scholarship@cuc.claremont.edu.

Structured Matrices and the Algebra of Displacement Operators

Ryan Takahashi

Michael Orrison, Advisor

Nicholas Pippenger, Reader



Department of Mathematics

May, 2013

Copyright © 2013 Ryan Takahashi.

The author grants Harvey Mudd College and the Claremont Colleges Library the nonexclusive right to make this work available for noncommercial, educational purposes, provided that this copyright statement appears on the reproduced materials and notice is given that the copying is by permission of the author. To disseminate otherwise or to republish requires written permission from the author.

Abstract

Matrix calculations underlie countless problems in science, mathematics, and engineering. When the involved matrices are highly structured, displacement operators can be used to accelerate fundamental operations such as matrix-vector multiplication. In this thesis, we provide an introduction to the theory of displacement operators and study the interplay between displacement and natural matrix constructions involving direct sums, Kronecker products, and blocking. We also investigate the algebraic behavior of displacement operators, developing results about invertibility and kernels.

Contents

Abstract	iii
1 Introduction	1
2 The Theory of Displacement	3
2.1 Structured Matrices	3
2.2 The Displacement Rank Approach	8
2.3 Recovery Formulae	16
2.4 Generating Recovery Formulae	19
3 The Algebra of Displacement Operators	23
3.1 Singular Displacement Operators	23
3.2 Displacement Kernels	25
3.3 The Method of Blocking	30
4 Future Directions	35
4.1 Matrix Symmetries	35
4.2 Group Matrices	37
4.3 Matrix Algebras	38
5 Conclusion	39
Bibliography	41

List of Tables

2.1	Displacing Matrices Associated with Families of Structured Matrices	10
-----	---	----

Chapter 1

Introduction

The motivation for the work presented in this thesis comes from structured matrices. Matrix calculations underlie countless problems in applied mathematics, and in many cases, the involved matrices have a particular structure. We shall see in Section 2.1 that these structured matrices arise in fields as diverse as image processing, communications, and data approximation. These matrices have been studied extensively, and methods have been discovered for expediting structured matrix computations (see, for instance, Horn and Johnson (1985)). Displacement operators provide a way to leverage these structured matrix algorithms to obtain computational improvements with matrices that are only *near*-structured.

The study of displacement operators began only relatively recently, in the wake of the groundbreaking paper by Kailath et al. (1979). At the time, fast algorithms were already known for performing basic computations, such as inversion or matrix-vector multiplication, with the structured family of Toeplitz matrices. Kailath et al. noted that in real problems, computations often involved matrices that were not Toeplitz, but were in some sense “nearly” Toeplitz. For instance, the inverses of Toeplitz matrices need not be Toeplitz themselves, but exhibit certain Toeplitz-like qualities.

Pursuing the intuition that these “nearly” Toeplitz matrices should also be nearly as computationally tractable, Kailath et al. developed a formula for decomposing non-Toeplitz matrices into Toeplitz parts, where the number of parts required is low when the matrix is Toeplitz-like. This method is the first branch of what is now known as “the displacement rank approach.”

Following the success of Kailath et al. (1979), other papers (see, for instance, Heinig and Rost (1984)) presented analogous results for other

classes of structured matrices. Such results are still of interest—more recent examples are developed in Kailath and Olshevsky (1997) and Kailath and Olshevsky (1995). However, the first attempt at unification of these results was Pan (1990). Pan’s paper was the first to extensively explore algebraic manipulation of the associated “displacement operators,” and was able to relate the formulae for different classes of structured matrices. This work foreshadowed the author’s comprehensive text Pan (2001), to which this thesis is deeply indebted.

The relatibility of different displacement operators is significant because it lends a certain universality to our theory. While fast algorithms for specific classes of matrices are of great practical value, the ideas we develop in this thesis will, in principle, be applicable to *any* matrix, although practical utility may rely on the degree of structure in the matrix.

We have three primary goals in this thesis. The first is to provide a clear introduction to the field. The study of displacement operators can be nuanced, and the comparative youth of the field means that results may be scattered across several works, or given in varying notations. Chapter 2 gives a deliberate, accessible overview of displacement operators and the associated methodology.

Our second objective is to explore the algebraic properties of displacement operators. Chapter 3 is much more theory-driven, developing original results about the kernels of displacement operators and the relationship between different types of displacement.

Finally, we have found that while many families of structured matrices have been studied extensively, few results exist involving constructed matrices, such as direct sums and Kronecker products. Throughout the thesis, we find opportunities to extend results, including our own, to include these natural matrix constructions.

Chapter 2

The Theory of Displacement

In this chapter, we introduce the mechanics necessary to work with displacement operators. Section 2.1 presents some of the structured matrix families we commonly work with and describes applications in which these matrix families arise. Section 2.2 collects and extends some basic definitions and results about displacement operators. The remainder of the chapter is devoted to the more complex step of recovery. Section 2.3 introduces some canonical examples of recovery formulae, and Section 2.4 shows how we can generate new recovery formulae to use in specific situations.

2.1 Structured Matrices

By a *structured matrix*, we typically mean an $n \times n$ matrix whose entries have a formulaic relationship, allowing the matrix to be specified by significantly fewer than n^2 parameters. The precise meaning of “significantly fewer” is left deliberately vague—matrices may have different degrees of structure. The theory we develop is, in principle, applicable to any matrix, although practically, the effectiveness of our tools will be commensurate with the amount of structure.

Example 2.1. *All of the matrix forms below are structured, to varying degrees, as seen in the different numbers of parameters.*

$$\begin{bmatrix} a & b & c & d \\ e & a & b & c \\ f & e & a & b \\ g & f & e & a \end{bmatrix} \begin{bmatrix} a & a & a & a \\ a & a & a & a \\ a & a & a & a \\ a & a & a & a \end{bmatrix} \begin{bmatrix} a & a^2 & a^3 & a^4 \\ b & b^2 & b^3 & b^4 \\ c & c^2 & c^3 & c^4 \\ d & d^2 & d^3 & d^4 \end{bmatrix} \begin{bmatrix} a & b & c & a \\ d & e & f & g \\ h & i & j & k \\ a & l & m & a \end{bmatrix}$$

In practice, we typically deal with matrices having a number of parameters on the order of n . Certain families of structured matrices are particularly relevant and commonly studied because they arise in applications. While these applications are not our primary focus, we give brief overviews of some interesting examples to motivate our study of the associated matrix families.

Example 2.2. *The primary goal of system identification is to reverse-engineer the constraints of an unknown system from measurements of its output. From the response data, a sequence of Markov parameters $\{Y_i\}$ is generated and used to populate matrices of the form*

$$H = \begin{bmatrix} Y_0 & Y_1 & \dots & Y_{n-1} \\ Y_1 & Y_2 & \dots & Y_n \\ \vdots & \dots & \dots & \vdots \\ Y_{n-1} & Y_n & \dots & Y_{2n-2} \end{bmatrix}.$$

Manipulation of this matrix yields the desired system parameters. An in-depth description of this process can be found in Ljung (1999). We call matrices with the above structure Hankel matrices.

Example 2.3. *Matrices arise naturally in problems of data approximation. Suppose, for instance, that we are given a set of $n + 1$ points in the plane, and wish to find a polynomial that interpolates the data. Then we seek a polynomial*

$$p(x) = a_0 + a_1x + \dots + a_nx^n$$

satisfying a set of constraints of the form $p(x_i) = y_i$. We can encode this problem in the matrix-vector equation

$$\begin{bmatrix} 1 & x_0 & \dots & x_0^n \\ 1 & x_1 & \dots & x_1^n \\ \vdots & \vdots & & \vdots \\ 1 & x_n & \dots & x_n^n \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{bmatrix}.$$

Thus, solving for the coefficients a_i relies on the inversion of the left-hand matrix. See, for instance, Axler (1997), for examples like this. We call matrices with the above structure Vandermonde matrices.

Example 2.4. *In coding theory, an error-correcting code facilitates communication over unreliable channels by packaging a transmission with redundant information to allow reconstruction of a partially garbled message. The class of Reed-Solomon codes are ubiquitous in modern communications technology, and are*

encoded via a matrix of the form

$$C = \begin{bmatrix} \frac{1}{s_0 - t_0} & \cdots & \frac{1}{s_0 - t_{n-1}} \\ \frac{1}{s_1 - t_0} & \cdots & \frac{1}{s_1 - t_{n-1}} \\ \vdots & & \vdots \\ \frac{1}{s_{n-1} - t_0} & \cdots & \frac{1}{s_{n-1} - t_{n-1}} \end{bmatrix}.$$

We call matrices with this structure Cauchy matrices.

Example 2.5. Nagy (1996) describes a model for image restoration,

$$g = Hf + \eta,$$

where g is the observed image, f is the true image, η is a noise term, and H is a matrix representing “blur,” a kind of image degradation. Nagy observes that the matrix H here is often a block matrix whose blocks take the form

$$T = \begin{bmatrix} t_0 & t_{-1} & \cdots & t_{1-n} \\ t_1 & t_0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & t_{-1} \\ t_{n-1} & \cdots & t_1 & t_0 \end{bmatrix}.$$

We call such a matrix a Toeplitz matrix. In this particular case, not only does H have Toeplitz blocks, but it is also block Toeplitz—that is, its blocks are aligned in Toeplitz fashion. Note that this need not imply that the matrix H is itself Toeplitz.

This observation, that matrices may not have a particular structure, but may be in some way *constructed* from parts with that structure, informs some of our original work throughout this thesis. It seems that we should be able to understand a constructed matrix as long as we understand all the components that form it. In this spirit, we extend several results, both new and existing, to include constructed matrices. In particular, we will often be interested in two common matrix constructions, the *direct sum* and the *Kronecker product*.

Definition 2.1. The direct sum of matrices A and B is the block matrix

$$A \oplus B = \begin{bmatrix} A & 0 \\ 0 & B \end{bmatrix}.$$

Definition 2.2. Let A be an $m_1 \times n_1$ matrix, and denote its i, j entry by $a_{i,j}$ (we use the shorthand $A = [a_{i,j}]$). Let B be an $m_2 \times n_2$ matrix. Then the Kronecker product of A and B is the $m_1 m_2 \times n_1 n_2$ block matrix

$$A \otimes B = [a_{i,j} B] = \begin{bmatrix} a_{1,1} B & \dots & a_{1,n_1 n_2} B \\ \vdots & & \vdots \\ a_{m_1 m_2, 1} B & \dots & a_{m_1 m_2, n_1 n_2} B \end{bmatrix}.$$

A final matrix class of interest, which will have plentiful applications within this thesis, is the family of *circulant matrices*.

Definition 2.3. A e -circulant matrix is a matrix of the form

$$Z = \begin{bmatrix} a_1 & ea_n & \dots & ea_2 \\ a_2 & a_1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & ea_n \\ a_n & \dots & a_2 & a_1 \end{bmatrix}.$$

In other words, circulants are a special case of Toeplitz matrices that also obey a certain diagonal wrapping condition. When we refer simply to a circulant matrix without qualification, e is assumed to be 1.

The unit e -circulant, in turn, is a special case of the e -circulant, denoted by Z_e and having the form

$$Z_e = \begin{bmatrix} 0 & 0 & \dots & e \\ 1 & 0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \dots & 1 & 0 \end{bmatrix}.$$

These unit circulants represent shift operators with scaled wrapping, and will pervade our work, beginning in Section 2.2.

In general, it is important to note that while the structure condition represents a certain internal redundancy, a structured matrix may still have full rank. For instance, the matrix

$$T = \begin{bmatrix} 1 & 2 & 5 & 7 \\ 0 & 1 & 2 & 5 \\ 3 & 0 & 1 & 2 \\ -2 & 3 & 0 & 1 \end{bmatrix}$$

is Toeplitz, but has rank 4. A major motivation for the theory we will introduce is the intuition that even in such cases, computations like inversion and matrix-vector multiplication should be faster or easier for structured matrices than for arbitrary matrices.

2.2 The Displacement Rank Approach

We begin the section with two related definitions.

Definition 2.4. For fixed matrices A and B with compatible dimensions, the Sylvester-type displacement operator, denoted $\nabla_{A,B}$, is defined by

$$\nabla_{A,B}(M) = AM - MB.$$

Definition 2.5. For fixed matrices A and B , the Stein-type displacement operator, denoted $\Delta_{A,B}$, is defined by

$$\Delta_{A,B}(M) = M - AMB.$$

In both cases, the matrices A and B are known as the *displacing* or *operator matrices*. For the purposes of this thesis, we shall mostly assume A and B to be square, but this need not necessarily be the case.

We shall use the two types of displacement operators more or less interchangeably. This is mostly a matter of convenience, but is justifiable. For one thing, we can often convert easily between the two, according to the following result from Pan (2001).

Theorem 2.1 (Pan, Theorem 1.3.1). *If the matrix A is non-singular, then*

$$\nabla_{A,B} = A\Delta_{A^{-1},B}.$$

If the matrix B is non-singular, then

$$\nabla_{A,B} = -\Delta_{A,B^{-1}}B.$$

Even when both displacing matrices are singular, however, we regard the two kinds of displacement as serving the same purpose—namely, to exploit redundancies in a structured matrix and reduce it to something of low rank. In this sense, while many useful displacement operators have the above forms, we should bear in mind that displacement operators *are* simply linear operators with certain desirable effects. Let us see an example of how these operators function.

Example 2.6. *Suppose we wish to reduce the Toeplitz matrix*

$$M = \begin{bmatrix} 1 & 2 & 0 & 5 \\ 3 & 1 & 2 & 0 \\ -1 & 3 & 1 & 2 \\ 1 & -1 & 3 & 1 \end{bmatrix}$$

to a low rank matrix.

Choose the displacing matrices A and B to be the unit circulants Z_1 and Z_0^T . Then if we apply a Stein-type displacement, we get

$$\begin{aligned} \Delta_{Z_1, Z_0^T}(M) &= M - Z_1 M Z_0^T \\ &= \begin{bmatrix} 1 & 2 & 0 & 5 \\ 3 & 1 & 2 & 0 \\ -1 & 3 & 1 & 2 \\ 1 & -1 & 3 & 1 \end{bmatrix} - \begin{bmatrix} 0 & 1 & -1 & 3 \\ 0 & 1 & 2 & 0 \\ 0 & 3 & 1 & 2 \\ 0 & -1 & 3 & 1 \end{bmatrix} \\ &= \begin{bmatrix} 1 & 1 & 1 & 2 \\ 3 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}, \end{aligned}$$

which has rank 2.

It is, of course, no coincidence that this displacement has low rank. Note that the action of left-multiplying by Z_1 is to shift the entries of a matrix down, with wrapping, while the action of right-multiplying by Z_0^T is to shift the entries right, without wrapping. We can see how the composition of these actions produced the matrix $Z_1 M Z_0^T$ in the computation above.

We also note that Toeplitz matrices have a kind of diagonal shift “resistance”—that is, not invariance, but something close to it. Because of the Toeplitz structure condition, shifting an entry down and to the right can only change the edges of the matrix, and thus we see that the lower right 3×3 blocks of M and $Z_1 M Z_0^T$ agree, producing the low rank difference. In this sense, we can consider the operator Δ_{Z_1, Z_0^T} to be associated with the family of Toeplitz matrices, because it reveals their structure. We will often refer to these associated displacements as *effective displacements* for their corresponding matrix families.

The question of what choices of displacing matrices yield operators associated with various families of matrices has already been thoroughly investigated. Table 1.1, borrowed with slight modifications from Pan (2001), lists some effective choices of displacing matrices for Sylvester-type operators. It is worth mentioning, however, that these are not the *only* operators that effectively reduce these classes of matrices. Note that in this context, the matrix $D(v)$ is the diagonal matrix whose diagonal entries are given by the column vector v . The vectors x, s , and t in the table refer back to our definitions of the corresponding matrices.

A	B	class of structured matrices M	rank of $\nabla_{A,B}(M)$
Z_1	Z_0	Toeplitz	≤ 2
Z_1	Z_0^T	Hankel	≤ 2
$Z_0 + Z_0^T$	$Z_0 + Z_0^T$	Toeplitz + Hankel	≤ 4
$D(v)$	Z_0	Vandermonde	≤ 1
Z_0	$D(v)$	inverse of Vandermonde	≤ 1
$D(s)$	$D(t)$	Cauchy	≤ 1
$D(t)$	$D(s)$	inverse of Cauchy	≤ 1

Table 2.1 Displacing Matrices Associated with Families of Structured Matrices

Note that in each case, the rank of $\nabla_{A,B}(M)$, called the *displacement rank*, is bounded by a constant regardless of the dimension of the matrices. This is typically what we mean in saying that a displacement rank is “low,” although slow-growing functions of the matrix dimension may occasionally be permitted. Bostan et al. (2006) provides an interesting discussion of some cases in which the displacement rank is linear, rather than constant.

There is comparatively little information about displacing constructed matrices such as direct sums and Kronecker products. We saw in Section 2.1, however, that these matrices can be relevant and arise in applications. Accordingly, we have derived two results to augment Table 1.1. The first is straightforward.

Theorem 2.2. *Let $M = M_1 \oplus M_2$. If ∇_{A_1,B_1} is an effective displacement for M_1 and ∇_{A_2,B_2} is an effective displacement for M_2 , then*

$$\nabla_{A_1 \oplus A_2, B_1 \oplus B_2}$$

is an effective displacement for M .

Proof. This is easily verified. We have

$$\begin{aligned} \nabla_{A_1 \oplus A_2, B_1 \oplus B_2}(M) &= \begin{bmatrix} A_1 & 0 \\ 0 & A_2 \end{bmatrix} \begin{bmatrix} M_1 & 0 \\ 0 & M_2 \end{bmatrix} - \begin{bmatrix} M_1 & 0 \\ 0 & M_2 \end{bmatrix} \begin{bmatrix} B_1 & 0 \\ 0 & B_2 \end{bmatrix} \\ &= \begin{bmatrix} A_1 M_1 - M_1 B_1 & 0 \\ 0 & A_2 M_2 - M_2 B_2 \end{bmatrix} = \begin{bmatrix} \nabla_{A_1, B_1}(M_1) & 0 \\ 0 & \nabla_{A_2, B_2}(M_2) \end{bmatrix}. \end{aligned}$$

The blocks of this matrix have constant bounded (or at least, asymptotically small) rank, and thus so does the whole matrix. \square

To develop a similar result for Kronecker products requires slightly more machinery. We claim that if ∇_{A_1, B_1} is an effective displacement for M_1 , and ∇_{A_2, B_2} is an effective displacement for M_2 , an effective displacement operator for the Kronecker product $M_1 \otimes M_2$ is the composition

$$\nabla_{A_1 \otimes I, B_1 \otimes I} \circ \nabla_{I \otimes A_2, I \otimes B_2}.$$

To prove this, first recall the following:

Definition 2.6. A complex square matrix M is called unitary if $MM^* = M^*M = I$, the identity matrix, where M^* denotes the conjugate transpose of M .

Theorem 2.3 (Singular Value Decomposition). If M is a real- or complex-valued (rectangular) matrix, there exists a factorization

$$M = U\Sigma V^*,$$

where U and V are square and unitary, and Σ is diagonal with non-negative real entries. These diagonal entries are called the singular values of M , and by convention, are listed in decreasing order down the main diagonal.

Proof. The proof is via the Spectral Theorems, and can be found in most linear algebra texts. See Axler (1997), for instance. \square

Since the matrix Σ in a singular value decomposition $M = U\Sigma V^*$ is diagonal, we can rewrite the expression for M . Call the number of non-zero singular values n . Then

$$M = \sum_{i=1}^n \sigma_i u_i v_i^*, \quad (2.1)$$

where σ_i is the i th singular value of M , u_i is the i th column of U , and v_i is the i th column of V . This leads us to the following observation.

Theorem 2.4. The rank of a matrix M is equal to the number of non-zero singular values of M .

Proof. Let

$$M = U\Sigma V^*$$

be the singular value decomposition of M . Then U and V are unitary, and thus V^* is also unitary, since $V^*(V^*)^* = V^*V = I$, and similarly $(V^*)^*V^* = VV^* = I$.

In particular, U and V^* are invertible. Thus the nullity of $M = U\Sigma V^*$ is exactly the nullity of Σ , so that M and Σ have the same rank. Since Σ is diagonal, the rank is simply the number of non-zero diagonal entries, which is precisely the number of non-zero singular values of M , as desired. \square

We now give a brief proof of a fact stated in Laub (2005).

Lemma 2.1. *If $A = U_A \Sigma_A V_A^*$ and $B = U_B \Sigma_B V_B^*$ are singular value decompositions, then*

$$A \otimes B = (U_A \otimes U_B)(\Sigma_A \otimes \Sigma_B)(V_A^* \otimes V_B^*)$$

is also a singular value decomposition, up to row/column order.

Proof. It is easy to check that the Kronecker product has the properties

$$(A \otimes B)(C \otimes D) = AC \otimes BD$$

and

$$(A \otimes B)^* = A^* \otimes B^*.$$

Thus we do have

$$A \otimes B = (U_A \Sigma_A V_A^*) \otimes (U_B \Sigma_B V_B^*) = (U_A \otimes U_B)(\Sigma_A \otimes \Sigma_B)(V_A^* \otimes V_B^*).$$

Moreover,

$$\begin{aligned} (U_A \otimes U_B)(U_A \otimes U_B)^* &= (U_A \otimes U_B)(U_A^* \otimes U_B^*) \\ &= U_A U_A^* \otimes U_B U_B^* = I \otimes I = I, \end{aligned}$$

since U_A and U_B are unitary, and thus $U_A \otimes U_B$ is unitary, and similarly for $V_A^* \otimes V_B^*$. Finally, $\Sigma_A \otimes \Sigma_B$ is the Kronecker product of diagonal matrices, and is thus diagonal. The diagonal entries are products of entries from Σ_A and Σ_B , which are real and non-negative, so the entries of $\Sigma_A \otimes \Sigma_B$ are likewise real and non-negative. \square

Finally, Theorem 2.4 and Lemma 2.1 lead us to the following lemma:

Lemma 2.2. *Let $A = B \otimes C$. Then*

$$\text{rank}(A) = \text{rank}(B)\text{rank}(C).$$

Proof. Let $\{\beta_1, \dots, \beta_m\}$ be the multiset (that is, including multiplicity) of non-zero singular values of B , and let $\{\gamma_1, \dots, \gamma_n\}$ be the multiset of non-zero singular values of C . By Lemma 2.1, the non-zero singular values of A are $\{\beta_i \gamma_j\}$, $i = 1, \dots, m$, $j = 1, \dots, n$, which has mn elements including multiplicity. By Theorem 2.4, however, the number of non-zero singular values is precisely the rank of the matrix, so $\text{rank}(A) = mn = \text{rank}(B)\text{rank}(C)$. \square

At last, we are in a position to state and prove our initial claim about displacement.

Theorem 2.5. *Let $M = M_1 \otimes M_2$. If ∇_{A_1, B_1} is an effective displacement for M_1 and ∇_{A_2, B_2} is an effective displacement for M_2 , then*

$$\nabla_{A_1 \otimes I, B_1 \otimes I} \circ \nabla_{I \otimes A_2, I \otimes B_2}$$

is an effective displacement for M .

Proof. We have

$$\begin{aligned} & \nabla_{A_1 \otimes I, B_1 \otimes I} \circ \nabla_{I \otimes A_2, I \otimes B_2}(M) \\ &= \nabla_{A_1 \otimes I, B_1 \otimes I}((I \otimes A_2)(M_1 \otimes M_2) - (M_1 \otimes M_2)(I \otimes B_2)) \\ &= \nabla_{A_1 \otimes I, B_1 \otimes I}(M_1 \otimes A_2 M_2 - M_1 \otimes M_2 B_2) \\ &= \nabla_{A_1 \otimes I, B_1 \otimes I}(M_1 \otimes (A_2 M_2 - M_2 B_2)) \\ &= \nabla_{A_1 \otimes I, B_1 \otimes I}(M_1 \otimes \nabla_{A_2, B_2}(M_2)) \\ &= (A_1 \otimes I)(M_1 \otimes \nabla_{A_2, B_2}(M_2)) - (M_1 \otimes \nabla_{A_2, B_2}(M_2))(B_1 \otimes I) \\ &= (A_1 M_1 \otimes \nabla_{A_2, B_2}(M_2)) - (M_1 B_1 \otimes \nabla_{A_2, B_2}(M_2)) \\ &= (A_1 M_1 - M_1 B_1) \otimes \nabla_{A_2, B_2}(M_2) \\ &= \nabla_{A_1, B_1}(M_1) \otimes \nabla_{A_2, B_2}(M_2). \end{aligned}$$

By Lemma 2.2, the rank of this matrix is then the product of the displacement ranks of M_1 and M_2 , which were assumed to be small, completing the proof. \square

Note that this result is easily generalized to k -fold Kronecker products, since the Kronecker product is associative.

Having gone into some detail with the preceding proofs, it may be helpful at this point to take stock of our position with respect to our larger goals. We have thus far been developing a catalogue of effective displacements for reducing certain classes of matrices. These displacements have low rank, and operations involving them will naturally be easy and quick. To get practical value out of displacements, however, we are still missing one crucial component. Having compressed a structured matrix and performed fast operations on the result, we must have a way to decompress and recover results about the original matrix that we were interested in. This process of compress-operate-decompress is sometimes called the *displacement rank approach*. Of the three steps, the recovery stage is, in a sense, the most complex, and we devote Sections 2.3 and 2.4 to its study.

We conclude our introduction to displacement operators with a comment about *near-structured matrices*. A matrix is near-structured if it has low displacement rank with respect to a displacement operator associated with a particular family of structured matrices. For instance, if M is a matrix such that $\nabla_{Z_1, Z_0}(M)$ is low rank, then M is near-structured, and we say that M is *Toeplitz-like*. The concept of near-structuredness will prove invaluable to our work. One interesting observation is that displacement rank provides a metric for judging how close a matrix is to having a particular structure, which is difficult to do based on intuition alone. For instance, consider the Toeplitz matrix T , below, and two perturbations, T' and T'' .

$$T = \begin{bmatrix} 0 & 1 & 2 & 3 \\ 1 & 0 & 1 & 2 \\ 3 & 1 & 0 & 1 \\ 1 & 3 & 1 & 0 \end{bmatrix}$$

$$T' = \begin{bmatrix} 0 & -1 & 2 & 3 \\ -1 & 0 & 1 & 2 \\ 3 & 1 & 0 & 1 \\ 1 & 3 & 1 & 0 \end{bmatrix}, T'' = \begin{bmatrix} 0 & 1 & 2 & 3 \\ -1 & 0 & 1 & 2 \\ 3 & 1 & 0 & -1 \\ 1 & 3 & 1 & 0 \end{bmatrix}$$

It seems reasonable to say that both T' and T'' look “close” to Toeplitz in some sense, and arguably, they look equally close. In both cases, we have negated two entries from the original Toeplitz matrix, in different rows, columns, and diagonals. However, one of these matrices behaves more like a Toeplitz matrix than the other. We find that

$$\nabla_{Z_1, Z_0}(T') = \begin{bmatrix} 2 & 1 & -2 & 0 \\ 0 & -2 & 0 & 3 \\ -2 & 0 & 0 & 2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

has rank 4, while

$$\nabla_{Z_1, Z_0}(T'') = \begin{bmatrix} 0 & 1 & -2 & 0 \\ 0 & 0 & 0 & 3 \\ -2 & 0 & 2 & 2 \\ 0 & 0 & 0 & -1 \end{bmatrix}$$

has rank 3. Thus T'' is more Toeplitz-like than T' .

The ability to perform such comparisons is itself of theoretical interest, and this idea will appear again in Chapter 4, but we will also be able to

extract enormous practical value from this idea of near-structured matrices. As alluded to in the introduction, numerous algorithmic tricks already exist for dealing with matrices that are Toeplitz, Hankel, etc. The greater power and universality of the displacement rank approach will become apparent in Section 2.4, when we extend our results to matrices that are only near-structured.

2.3 Recovery Formulae

We now turn our attention to the recovery step of the displacement rank method. In the abstract, this is the payoff of our approach; having displaced a structured matrix and operated on the low-rank result, we now “decompress,” in the parlance of Pan (2001), and extract what we originally wanted to compute using the structured matrix. Concretely, recovery entails the expression of a matrix in terms of several simple components (for instance, displacements of the matrix and/or the displacing matrices A and B). It may seem circular, having defined the displacement in terms of the matrix, to now write the matrix in terms of its displacement, but our aim is to decompose the matrix as a sum of products involving highly tractable matrices such as the displacement. We shall see this idea in action later in the section. These simple decompositions allow us to reduce a single weighty computation to several very fast ones. Before we can give an example of a recovery formula, we need the following definition.

Definition 2.7. *Let A be a square matrix. If $A^p = aI$ for some positive integer p and some scalar a , we say that A is a -potent of order p .*

The following theorem and its corollary, given in Pan (2001), have served as prototypical examples of recovery formulae in our work.

Theorem 2.6 (Pan (2001), Theorem 4.3.6). *For all $k \geq 1$, we have*

$$M = A^k M B^k + \sum_{i=0}^{k-1} A^i \Delta_{A,B}(M) B^i.$$

Theorem 2.7 (Pan (2001), Corollary 4.3.7). *If A is a -potent of order p and/or B is b -potent of order q , then*

$$M = \left(\sum_{i=0}^{p-1} A^i \Delta_{A,B}(M) B^i \right) (I - aB^p)^{-1}$$

and/or

$$M = (I - bA^q)^{-1} \left(\sum_{i=0}^{q-1} A^i \Delta_{A,B}(M) B^i \right),$$

respectively.

Let us see how this formula might be used.

Example 2.7. Suppose we wish to recover the Toeplitz matrix

$$T = \begin{bmatrix} 1 & 2 & 0 & 5 \\ 3 & 1 & 2 & 0 \\ -1 & 3 & 1 & 2 \\ 1 & -1 & 3 & 1 \end{bmatrix}$$

from a displacement of T . Recall that Δ_{Z_1, Z_0^T} is an effective displacement for Toeplitz matrices. We see that $Z_1^{4 \times 4}$ is 1-potent of order 4, so by Theorem 2.7, we have

$$T = \left(\sum_{i=0}^3 Z_1^i \Delta_{Z_1, Z_0^T}(T) (Z_0^T)^i \right) (I - (Z_0^T)^4)^{-1}.$$

Recalling that Z_0 (and thus Z_0^T) represents a shift without wrapping, it is always nilpotent of order equal to its dimension, so

$$(I - (Z_0^T)^4)^{-1} = I^{-1} = I.$$

We previously computed

$$\Delta_{Z_1, Z_0^T}(T) = \begin{bmatrix} 1 & 1 & 1 & 2 \\ 3 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix},$$

which is the term of the summation corresponding to $i = 0$. Then we simply apply down-shifts with wrapping and right-shifts without wrapping to obtain

$$Z_1 \Delta_{Z_1, Z_0^T}(T) Z_0^T = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 3 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix},$$

$$Z_1^2 \Delta_{Z_1, Z_0^T}(T) (Z_0^T)^2 = \begin{bmatrix} 0 & 0 & -1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 3 & 0 \end{bmatrix},$$

and

$$Z_1^3 \Delta_{Z_1, Z_0^T}(T) (Z_0^T)^3 = \begin{bmatrix} 0 & 0 & 0 & 3 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

We can verify that the sum of these matrices is indeed the original matrix T . Note that we could have used any displacement operator here, and successfully recovered T , but our choice of a Toeplitz-associated operator produced a decomposition into matrices of very low rank.

It is worth noting that recovery formulae may take very different forms. For instance, let us examine one of the earliest recovery formulae, from Kailath et al. (1979). Here, $L(x)$ is the lower-triangular Toeplitz matrix whose first column is x . That is,

$$L([x_1 \ \dots \ x_n]^T) = \begin{bmatrix} x_1 & 0 & \dots & 0 \\ x_2 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ x_n & \dots & x_2 & x_1 \end{bmatrix}.$$

Similarly, $U(y^T)$ denotes the upper-triangular Toeplitz matrix whose first row is y^T . We modernize the authors' notation slightly for the sake of clarity.

Theorem 2.8 (Kailath et al. (1979), Lemma 2). *Given column vectors $\{f_i, g_i\}$, the unique solution of*

$$\Delta_{Z_0, Z_0^T}(M) = \sum_{i=1}^{\alpha} f_i g_i^T$$

is

$$M = \sum_{i=1}^{\alpha} L(f_i)U(g_i^T).$$

This is a very different statement than was given for Pan's formulae, and bears some unpacking. The key observation is that the sum expression

$$\Delta_{A,B}(M) = \sum_{i=1}^{\alpha} f_i g_i^T \tag{2.2}$$

is somewhat idiomatic in this kind of work. In stating (2.2), we will typically mean that the displacement rank is α . This convention makes sense because of Equation 2.1, from the singular value decomposition, which always allows us to write a rank α matrix as an α -fold sum of vector outer products.

So Theorem 2.8 can be understood as giving a simple sum of products decomposition, where the number of terms needed is the displacement rank of the target matrix. This convenient form and dependence on displacement rank are the unifying properties of the many disparate recovery formulae.

2.4 Generating Recovery Formulae

Pan (2001) develops several explicit formulae like Theorem 2.8 from the more general theorems 2.6 and 2.7. This idea will prove to be very powerful, and we take a moment to justify one of Pan's statements.

Theorem 2.9 (Pan (2001), Example 4.4.1). *Let M have displacement rank α with respect to the operator $\Delta_{Z_e, Z_f^T}, ef \neq 1$, so that $\Delta_{Z_e, Z_f^T}(M) = \sum_{i=1}^{\alpha} g_i h_i^T$ as in 2.2. Then*

$$(1 - ef)M = \sum_{i=1}^{\alpha} \begin{bmatrix} g_{1,i} & eg_{n,i} & \cdots & eg_{2,i} \\ g_{2,i} & g_{1,i} & \ddots & \vdots \\ \vdots & \ddots & \ddots & eg_{n,i} \\ g_{n,i} & \cdots & g_{2,i} & g_{1,i} \end{bmatrix} \begin{bmatrix} h_{1,i} & h_{2,i} & \cdots & h_{n,i} \\ fh_{n,i} & h_{1,i} & \ddots & \vdots \\ \vdots & \ddots & \ddots & h_{2,i} \\ fh_{2,i} & \cdots & fh_{n,i} & h_{1,i} \end{bmatrix}$$

Proof. We will use Theorem 2.7. Since $Z_e^{n \times n}$ is e -potent of order n , we have $fZ_e^n = efI$. Since we have taken $ef \neq 1$, $1 - ef \neq 0$, so multiply through to obtain

$$(1 - ef)M = \sum_{j=0}^{n-1} Z_e^j \Delta_{Z_e, Z_f^T}(M) (Z_f^T)^j.$$

Now we make the substitution $\Delta_{Z_e, Z_f^T}(M) = \sum_{i=1}^{\alpha} g_i h_i^T$, writing

$$(1 - ef)M = \sum_{j=0}^{n-1} Z_e^j \left(\sum_{i=1}^{\alpha} g_i h_i^T \right) (Z_f^T)^j.$$

Note that Z_e^j and $(Z_f^T)^j$ are independent of the index i , so their multiplication distributes over the inner sum. Associating in a convenient way, we get

$$(1 - ef)M = \sum_{j=0}^{n-1} \sum_{i=1}^{\alpha} (Z_e^j g_i) (h_i^T (Z_f^T)^j).$$

Now we use the fact that $(AB)^T = B^T A^T$ and interchange the order of summation to obtain

$$(1 - ef)M = \sum_{i=1}^{\alpha} \sum_{j=0}^{n-1} (Z_e^j g_i) (Z_f^j h_i)^T. \quad (2.3)$$

Finally, we rewrite the inner sum as the product of matrices, so that

$$(1 - ef)M = \sum_{i=1}^{\alpha} G_i H_i^T, \quad (2.4)$$

where the j th column of G_i is $Z_e^j g_i$ and the j th column of H_i (hence the j th row of H_i^T) is $Z_f^j h_i$. It is straightforward to verify that this matrix product gives the same result as the previous sum of vector products, and that these matrices take the forms shown in the theorem statement. \square

This technique is easily extended to a wide variety of displacements and matrices, and gives us a great deal of power. In particular, note that we have reduced the problem of recovery to understanding the action of the displacing matrices A and B (and thus, of their powers) on column vectors. Inasmuch as A and B are likely to be computationally tractable (for instance, unit circulants), analogues of Equation 2.4 are easily produced and provide evocative recovery formulae.

Moreover, these kinds of formulae are the tools we use to improve computational efficiency. Some quick estimation will demonstrate the kind of improvement we can see.

Example 2.8. *Naïve matrix-vector multiplication requires n^2 multiplications for an $n \times n$ matrix, while Toeplitz-vector multiplication can be reduced to about $n \log n$ multiplications by other means (see Horn and Johnson (1985)). Suppose that, for a large computational problem, we need to multiply a 100×100 Toeplitz-like matrix against a vector. Using the raw matrix, this would require $100^2 = 10,000$ multiplications, but the matrix is Toeplitz-like. Say it has displacement rank 3 with respect to the operator in Theorem 2.9. Then the matrix decomposes as the sum of three Toeplitz products as shown. Each of these terms represents two Toeplitz-vector multiplications, so at a rough estimate, we have to run only about $6 \cdot 100 \log(100) \approx 2,763$ basic multiplications. Of course, this effect only becomes more dramatic as the dimensions of the involved matrices increase.*

Pan and others have produced formulae like Theorem 2.9 for various families of structured matrices (that is, for effective displacements of the families). In these cases, the displacing matrices A and B are commonly unit circulants, or sometimes diagonal matrices, as seen in Table 1.1. However, we have already seen that there may be motivation to displace by natural *constructed* matrices. Specifically, we wish to consider formulae for displacements by matrices that take the form of direct sums and Kronecker

products. Let us consider what Equation 2.4 looks like when we displace by two matrices

$$A_1 = B_1 \oplus C_1 = \begin{bmatrix} B_1 & 0 \\ 0 & C_1 \end{bmatrix} \text{ and } A_2 = B_2 \oplus C_2 = \begin{bmatrix} B_2 & 0 \\ 0 & C_2 \end{bmatrix}$$

that have the same block dimensions.

Note, first of all, that if $A = B \oplus C$ is block diagonal, its powers are simply the matrices $A^k = B^k \oplus C^k$. Let us suppose for now that A_1 is a k -potent matrix. This may seem difficult to achieve, but consider for instance

$$A_1 = Z_2^{3 \times 3} \oplus Z_4^{6 \times 6}.$$

We have

$$(Z_2^{3 \times 3})^6 = (Z_2^{3 \times 3})^3 (Z_2^{3 \times 3})^3 = (2I)(2I) = 4I,$$

and

$$(Z_4^{6 \times 6})^6 = 4I,$$

so

$$A_1^6 = 4I \oplus 4I = 4I.$$

We have seen that to write a recovery formula for displacement by these matrices, we need to evaluate the actions of A_1 and A_2 on column vectors. To this end, let us consider a column vector g_i (as in Equation 2.4) to be the concatenation of g'_i and g''_i , of lengths conforming to the blocks of A_1 and A_2 , and similarly for h_i . Thus

$$A_1 g_i = \begin{bmatrix} B_1 & 0 \\ 0 & C_1 \end{bmatrix} \begin{bmatrix} g'_i \\ g''_i \end{bmatrix} = \begin{bmatrix} B_1 g'_i \\ C_1 g''_i \end{bmatrix},$$

and similarly for A_2 . So the matrix product in Equation 2.4 becomes.

$$\sum_{i=1}^{\alpha} \begin{bmatrix} g'_i & B_1 g'_i & B_1^2 g'_i & \dots \\ g''_i & C_1 g''_i & C_1^2 g''_i & \dots \end{bmatrix} \begin{bmatrix} h_i'^T & h_i''^T \\ h_i'^T B_2 & h_i''^T C_2 \\ h_i'^T B_2^2 & h_i''^T C_2^2 \\ \vdots & \vdots \end{bmatrix}.$$

An interesting result is obtained simply by expanding this matrix multiplication in the block form we have written (that is, $2 \times n$ times $n \times 2$). The result is the 2×2 block matrix

$$\sum_{j=1}^{\alpha} \begin{bmatrix} \sum_{i=0}^{k-1} B_1^i f'_j g_j'^T B_2^i & \sum_{i=0}^{k-1} B_1^i f'_j g_j''^T C_2^i \\ \sum_{i=0}^{k-1} C_1^i f''_j g_j'^T B_2^i & \sum_{i=0}^{k-1} C_1^i f''_j g_j''^T C_2^i \end{bmatrix}. \quad (2.5)$$

Comparing this expression to Equation 2.3, we note that the entries of the matrices being summed here are themselves the quantities that we would sum to recover for lower-dimensional displacements. In particular, note that the absence of transposes in (2.5) means that the entries are recovery sums for the displacements Δ_{B_1, B_2^T} , Δ_{B_1, C_2^T} , Δ_{C_1, B_2^T} , and Δ_{C_1, C_2^T} .

Thus (2.5) lets us understand how to recover from direct sum displacements, provided we have an understanding of displacements involving the blocks of the direct sum. We have encountered difficulties in developing an analogous result for Kronecker products, but it seems to be an interesting question.

Chapter 3

The Algebra of Displacement Operators

In this chapter, we move away from mechanics and applications to study displacement operators as algebraic objects. In Section 3.1, we consider what conditions cause a displacement operator to be non-invertible, as well as the significance of non-invertible displacements. Section 3.2 explores the question of characterizing displacement kernels for operators of specific forms. Finally, Section 3.3 presents an innovative result, inspired by Orri-son (2012), on the relationship between displacement operators.

3.1 Singular Displacement Operators

In Section 2.3, we considered a recovery formula for Δ_{Z_0, Z_0^T} . Ammar and Gader (1991) give the following analogue for Z_1 .

Theorem 3.1. *If*

$$\Delta_{Z_1, Z_1^T}(M) = \sum_{i=1}^{\alpha} f_i g_i^T,$$

then

$$M = \text{Circ}_{lr} + \sum_{i=1}^{\alpha} L(f_i) \cdot \text{Circ}_1(g_i).$$

As before, $L(f_i)$ denotes the lower-triangular Toeplitz matrix with first column f_i . Here, $\text{Circ}_1(g_i)$ is the circulant matrix with first column g_i , and Circ_{lr} is the circulant matrix with the same last row as M .

To understand the difference between this formula and that given by Theorem 2.8, we take a slightly broader view and observe that, all matrices aside, displacement *operators* are simply that: linear operators. While some displacement operators, such as ∇_{Z_1, Z_0} , may be invertible, displacement operators may in general have non-trivial kernels. The question of operator invertibility explains the difference between Theorem 2.8 and Theorem 3.1. When we displace in a non-invertible way, we are transforming with loss of information, so to recover the original matrix completely, we have to store additional information about the matrix. In our examples, while Δ_{Z_0, Z_0^T} is invertible, Δ_{Z_1, Z_1^T} is not, and thus the extra term $Circ_{I_r}$ in Theorem 3.1 is precisely a kernel representative, representing information about M that we have held onto while displacing.

In general, this will be our strategy for generating recovery formulae for non-invertible displacement operators. Following Pan (2001), let L be any displacement operator. We will then write a matrix as the sum

$$M = M_N + M_C,$$

where M_N lies in the kernel $N(L)$ of L , and M_C in the orthogonal complement (under the usual Frobenius inner product) $C(L)$ of the kernel. Then the restriction $L|_{C(L)}$ is invertible by construction, and we can recover M_C in the usual way. To complete our recovery formulae, we then wish to understand the kernel $N(L)$.

For a general displacement operator $\nabla_{A,B}$, characterizing the elements in the kernel is a difficult problem. However, it is comparatively easy to determine whether or not a particular displacement kernel is trivial. Recall that the spectrum $\sigma(M)$ of a matrix M is the set of its eigenvalues.

Theorem 3.2. *The operator $\nabla_{A,B}$ is invertible if and only if $\sigma(A) \cap \sigma(B) = \emptyset$.*

A proof of this theorem is given in Pan (2001), but relies on some results that we shall not refer to here. We give a proof from elementary linear algebra.

Proof. Suppose $M \in \ker(\nabla_{A,B})$, so that $AM - MB = 0$, or $AM = MB$, and suppose that $\sigma(A) \cap \sigma(B) = \emptyset$. We show briefly that for any polynomial p , we must have

$$p(A)M = Mp(B).$$

Since matrix multiplication distributes over addition, we need only show that $A^n M = M B^n$ for any $n \in \mathbb{N}$. We have $AM = MB$ by hypothesis.

Suppose $A^{k-1}M = MB^{k-1}$ for some $k \in \mathbb{N}$. Then

$$A^k M = A(A^{k-1}M) = A(MB^{k-1}) = (AM)B^{k-1} = (MB)B^{k-1} = MB^k,$$

so the statement follows by induction.

In particular, let p_A be the characteristic polynomial of A . Then $p_A(A)M = Mp_A(B)$ by the above. By the Cayley-Hamilton Theorem, $p_A(A) = 0$, so $Mp_A(B) = 0$. On the other hand, we have

$$p_A(B) = \prod_{\lambda \in \sigma(A)} (B - \lambda I).$$

If any of the operators $(B - \lambda I)$ is singular, then it has a non-trivial kernel, so $(B - \lambda I)v = 0$ for some non-zero vector v , and thus $Bv = \lambda v$, contradicting our assumption that A and B share no eigenvalues. Thus $p_A(B)$ is the composition of invertible operators, and must itself be invertible. Since $Mp_A(B) = 0$ and $p_A(B)$ is invertible, we must have $M = 0$, and thus $\ker(\nabla_{A,B})$ is trivial.

Conversely, suppose A and B have a common eigenvalue $\lambda \in \sigma(A) \cap \sigma(B)$. Since $\sigma(B) = \sigma(B^T)$, say $Au = \lambda u$ and $B^T v = \lambda v$ for non-zero vectors u and v . Hence $(B^T v)^T = (\lambda v)^T$, or $v^T B = v^T \lambda$. Then

$$Auv^T = \lambda uv^T = uv^T \lambda = uv^T B,$$

so $uv^T \in \ker(\nabla_{A,B})$, and since u and v are non-zero, so is uv^T . Thus $\nabla_{A,B}$ is non-invertible, completing the proof. \square

We have the following analogous result for Stein-type operators.

Theorem 3.3. *The operator $\Delta_{A,B}$ is invertible if and only if, for all $\alpha \in \sigma(A)$, $\beta \in \sigma(B)$, $\alpha\beta \neq 1$.*

Proof. We refer the reader to Pan (2001) for the proof of this theorem. \square

While these conditions may have to suffice in general, we can say more in specific cases. In the next Section, we explore some of these results.

3.2 Displacement Kernels

Suppose we are interested in the operator $\nabla_{A,A}$, for some matrix A . Clearly, the kernel of this operator contains at least A itself, since $AA - AA = 0$. In fact, we can see that the kernel of $\nabla_{A,A}$ consists of precisely the matrices

that commute with A . Characterizing these matrices is not easy. It is well known that diagonalizable matrices commute if and only if they are simultaneously diagonalizable (see, for instance, Horn and Johnson (1985)), but we frequently make use of (non-zero) nilpotent matrices, such as Z_0 , which cannot be diagonalizable, so we seek a more general result. One half of the condition has a natural generalization.

Lemma 3.1. *Let $A, B : V \rightarrow V$ be linear operators on a complex vector space V . If $AB = BA$, then there exists a basis \mathcal{B} for V whose elements are generalized eigenvectors for both A and B .*

Proof. Suppose A and B commute, and decompose V as the direct sum of generalized eigenspaces as determined by the eigenvalues $\{\lambda_i\}$ of A . So we have

$$V = G_{\lambda_1} \oplus G_{\lambda_2} \oplus \cdots \oplus G_{\lambda_k}.$$

Suppose that $v \in G_{\lambda_i}$. By construction, $(A - \lambda_i I)^k v = 0$, for some positive integer k . Here $(A - \lambda_i I)^k$ is simply some polynomial $p(A)$, and since A commutes with B , so does any polynomial in A . Hence,

$$p(A)(Bv) = (p(A)B)v = (Bp(A))v = B(p(A)v) = B(0) = 0.$$

Hence $Bv \in G_{\lambda_i}$ also, so B maps the generalized eigenspace to itself. Thus the restriction $B|_{G_{\lambda_i}}$ of B to the generalized eigenspace is well-defined as an operator on G_{λ_i} , and we can further decompose each G_{λ_i} as the direct sum of generalized eigenspaces corresponding to the eigenvalues $\{\mu_j\}$ of B . Say

$$G_{\lambda_i} = H_{\lambda_i, \mu_1} \oplus H_{\lambda_i, \mu_2} \oplus \cdots \oplus H_{\lambda_i, \mu_l}.$$

Then

$$V = \bigoplus_{i,j} H_{\lambda_i, \mu_j}.$$

Let $\mathcal{B}_{i,j}$ be a basis for H_{λ_i, μ_j} . By construction, every element of $\mathcal{B}_{i,j}$ is a generalized eigenvector of both A and B , so

$$\mathcal{B} = \bigcup_{i,j} \mathcal{B}_{i,j}$$

is the desired basis for V . □

To state a sufficient condition for possibly non-diagonalizable matrices to commute is more challenging. Note, in particular, that the converse of the above lemma does not hold. If there exists a basis \mathcal{B} for V consisting

of shared generalized eigenvectors, we can simultaneously upper triangularize A and B (and in fact, can write one of the matrices in Jordan normal form), but this simultaneous triangularizability is insufficient to guarantee commutation. Counterexamples are plentiful, but we might take

$$A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, B = \begin{bmatrix} 1 & 2 \\ 0 & 3 \end{bmatrix}.$$

A is in Jordan normal form, and B is upper triangular, but certainly $AB \neq BA$. In fact, we find that neither simultaneous Jordanizability, nor even simultaneous Jordanizability with corresponding eigenvalues is sufficient, since for instance,

$$\begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix} \text{ and } \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

do not commute. We can at least say the following.

Lemma 3.2. *Let $A, B : V \rightarrow V$ be linear operators on a complex vector space V . If there exists a basis \mathcal{B} for V such that the encodings $[A]_{\mathcal{B}}$ and $[B]_{\mathcal{B}}$ are both in Jordan normal form with identical Jordan block structure, then $AB = BA$.*

Proof. Commutativity is independent of choice of basis, since if $AB = BA$ and P is a change of basis matrix, we have

$$(PAP^{-1})(PBP^{-1}) = PABP^{-1} = PBAP^{-1} = (PBP^{-1})(PAP^{-1}).$$

Since $[A]_{\mathcal{B}}$ and $[B]_{\mathcal{B}}$ are block diagonal, they commute if and only if the diagonal blocks commute. So we need only show that two Jordan blocks of the same size (but possibly corresponding to different eigenvalues) commute.

Write each matrix as the sum of a diagonal and a nilpotent part.

$$[A]_{\mathcal{B}} = \begin{bmatrix} \lambda & 1 & 0 & \dots & 0 \\ 0 & \lambda & 1 & \dots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \dots & 0 & \lambda & 1 \\ 0 & \dots & 0 & 0 & \lambda \end{bmatrix} = \lambda I + Z_0^T,$$

and similarly

$$[B]_{\mathcal{B}} = \mu I + Z_0^T,$$

for some eigenvalues λ and μ . Then

$$[A]_{\mathcal{B}}[B]_{\mathcal{B}} = \lambda\mu I + \lambda I Z_0^T + Z_0^T \mu I + (Z_0^T)^2,$$

and

$$[B]_{\mathcal{B}}[A]_{\mathcal{B}} = \mu\lambda I + \mu I Z_0^T + Z_0^T \lambda I + (Z_0^T)^2.$$

Since scalars and the identity matrix commute with any matrix, these expressions are equal, completing the proof. \square

Note that the above proof suggests a valid weakening of the assumption in the lemma. In the final term-by-term comparison, the fact that $(Z_0^T)^2 = (Z_0^T)^2$ is a special case of the two matrices' nilpotent parts commuting—we can permit the blocks to have different structures so long as the main diagonals are homogeneous (that is, we force the eigenvalues to match up) and the nilpotent parts commute with each other.

While we would like, for the sake of completeness, to develop a full characterization of $\ker(\nabla_{A,A})$ in this fashion, it is perhaps impractical to determine kernel elements via these kinds of theorems. In practice, however, we often work with displacements that have intuitive interpretations. For instance, consider the kernel of ∇_{Z_1, Z_1} . Since Z_1 represents the cyclic shift operator, to say that $Z_1 M - M Z_1 = 0$ is to say that shifting the entries of M downwards and shifting them left produces the same result. Thus we can see that $\ker(\nabla_{Z_1, Z_1})$ consists of precisely the circulant matrices. We may also be able to say more about displacement kernels when the matrix A has a particular form.

For instance, consider $\ker(\nabla_{A_1, A_2})$ when A_1 and A_2 have the form of direct sums or Kronecker products. While we have not yet discovered exact characterizations of these kernels, we can state two natural results. For elegance of notation, we identify an element (S, T) in the Cartesian product $\ker(\nabla_{B_1, B_2}) \times \ker(\nabla_{C_1, C_2})$ with the block matrix $S \oplus T$. We assume square matrices for simplicity. Then we have the following results:

Lemma 3.3. *Let $A_1 = B_1 \oplus C_1$ and $A_2 = B_2 \oplus C_2$ have the same block dimensions. Then*

$$\ker(\nabla_{B_1, B_2}) \oplus \ker(\nabla_{C_1, C_2}) \subset \ker(\nabla_{A_1, A_2}).$$

Lemma 3.4. *Let $A_1 = B_1 \otimes C_1$ and $A_2 = B_2 \otimes C_2$ have the same block dimensions. Then*

$$\ker(\nabla_{B_1, B_2}) \otimes \ker(\nabla_{C_1, C_2}) \subseteq \ker(\nabla_{A_1, A_2}).$$

Proof (Direct Sums). This is easy to see. If $A_1 = B_1 \oplus C_1 = \begin{bmatrix} B_1 & 0 \\ 0 & C_1 \end{bmatrix}$, block a matrix M conformally (that is, with dimensions corresponding to those of B_1 and C_1) as $M = \begin{bmatrix} M_1 & M_2 \\ M_3 & M_4 \end{bmatrix}$. Then

$$\nabla_{A_1, A_2}(M) = \begin{bmatrix} B_1 M_1 - M_1 B_2 & B_1 M_2 - M_2 C_2 \\ C_1 M_3 - M_3 B_2 & C_1 M_4 - M_4 C_2 \end{bmatrix}.$$

Clearly, then, $M \in \ker(\nabla_{A_1, A_2})$, if and only if $M_1 \in \ker(\nabla_{B_1, B_2})$, $M_2 \in \ker(\nabla_{B_1, C_2})$, $M_3 \in \ker(\nabla_{C_1, B_2})$, and $M_4 \in \ker(\nabla_{C_1, C_2})$. The neater stated result follows *a fortiori*. \square

Proof (Kronecker Products). The crux of the proof is showing the result for simple tensors. Let $D \in \ker(\nabla_{B_1, B_2})$, and $E \in \ker(\nabla_{C_1, C_2})$. We wish to show that $M = D \otimes E$ is in the kernel of ∇_{A_1, A_2} . We have

$$\begin{aligned} \nabla_{A_1, A_2}(M) &= (B_1 \otimes C_1)(D \otimes E) - (D \otimes E)(B_2 \otimes C_2) \\ &= B_1 D \otimes C_1 E - D B_2 \otimes E C_2, \end{aligned}$$

by a property of the Kronecker product. Thus the i, j block of $\nabla_{A, A}(M)$ is

$$(b_{i,*} \cdot d_{*,j}^T) C_1 E - (d_{i,*} \cdot \beta_{*,j}^T) E C_2,$$

where $b_{i,*}$ is the i th row of B_1 , $\beta_{*,j}$ is the j th column of B_2 , and similarly for the d in D . Since $D \in \ker(\nabla_{B_1, B_2})$, and thus $B_1 D = D B_2$, we know that $b_{i,*} \cdot d_{*,j}^T = d_{i,*} \cdot \beta_{*,j}^T$ for all i and j . Hence, the i, j entry of $\nabla_{A, A}(M)$ can be written as

$$(b_{i,*} \cdot d_{*,j}^T)(C_1 E - E C_2).$$

Since $E \in \ker(\nabla_{C_1, C_2})$ by construction, $C_1 E - E C_2$ is zero, and we find $\nabla_{A_1, A_2}(M) = 0$. An appeal to the linearity of the displacement operator completes the proof. \square

3.3 The Method of Blocking

Up to this point, we have considered only two specific constructions: the direct sum and the Kronecker product. A broader question is whether we can relate the displacement by a matrix A to the displacement by a matrix A' containing A as a block. This seems to be a very big question, and we do not nearly exhaust its study, but our next result, inspired by Orrison (2012), shows that these relationships can exist, in principle. We refer to the already stated theorems 2.8 and 3.1, which we reproduce in abbreviated form.

Theorem (Theorem 2.8).

$$\Delta_{Z_0, Z_0^T}(M) = \sum_{i=1}^{\alpha} f_i g_i^T$$

if and only if

$$M = \sum_{i=1}^{\alpha} L(f_i)U(g_i^T).$$

Theorem (Theorem 3.1). *If*

$$\Delta_{Z_1, Z_1^T}(M) = \sum_{i=1}^{\alpha} f_i g_i^T,$$

then

$$M = \text{Circ}_{1r} + \sum_{i=1}^{\alpha} L(f_i) \cdot \text{Circ}_1(g_i).$$

We saw in Section 3.1 that the difference between these forms can be explained in terms of displacement kernels. We now demonstrate that we can nonetheless derive a result very close to Theorem 2.8 from Theorem 3.1. The result is close but not precisely the stated theorem in that we cannot guarantee α as an upper bound for the number of terms in the sum, but the difference is small. This derivation may seem a curious idea to consider, but it is motivated by the observation that

$$Z_1^{(n+1) \times (n+1)} = \begin{bmatrix} Z_0^{n \times n} & e_1 \\ e_n^T & 0 \end{bmatrix},$$

where e_i is the i th standard basis vector for \mathbb{R}^n .

We begin with two lemmas. While these results are unsurprising, we are unaware of any explicit proofs in the literature, and offer brief justifications here.

Lemma 3.5. *The product of $n \times n$ lower (resp. upper) triangular Toeplitz matrices is an $n \times n$ lower (resp. upper) triangular Toeplitz matrix.*

Proof. We begin by showing the lower-triangular case. The proof is by induction. Since

$$\begin{bmatrix} 0 & 0 \\ a & 0 \end{bmatrix} \begin{bmatrix} 0 & 0 \\ b & 0 \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

is technically lower triangular Toeplitz, we have a base case. For the inductive step, write a blockwise product of $n \times n$ lower triangular Toeplitz matrices as

$$\begin{bmatrix} L_1 & 0 \\ l_1 & a \end{bmatrix} \begin{bmatrix} L_2 & 0 \\ l_2 & b \end{bmatrix} = \begin{bmatrix} L_1 L_2 & 0 \\ l_1 L_2 + a l_2 & ab \end{bmatrix},$$

where L_1 and L_2 are $(n-1) \times (n-1)$. It is tedious but straightforward to verify that the bottom row entries in the product matrix agree with the diagonals of $L_1 L_2$.

Now, if U_1, U_2 are upper-triangular Toeplitz, then U_1^T and U_2^T are lower-triangular Toeplitz. By the above, $U_2^T U_1^T = (U_1 U_2)^T$ is lower-triangular Toeplitz, and thus $U_1 U_2$ is upper-triangular Toeplitz. \square

Definition 3.1. *Say that a matrix M has an α -fold Lower-Upper Toeplitz (LUT) decomposition if $M = \sum_{j=1}^{\alpha} L(f_j) \cdot U(g_j)$ in the sense of Theorem 2.8.*

Lemma 3.6. *Let A be a square matrix. If the block matrix $\begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix}$ has an α -fold LUT decomposition, then so does A .*

Proof. Write the lower and upper triangular matrices in block form conforming to the above matrix. Then

$$\begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix} = \sum_{j=1}^{\alpha} L(f_j) \cdot U(g_j) = \sum_{j=1}^{\alpha} \begin{bmatrix} L_j & 0 \\ l_j & a_j \end{bmatrix} \begin{bmatrix} U_j & u_j \\ 0 & b_j \end{bmatrix} = \sum_{j=1}^{\alpha} \begin{bmatrix} L_j U_j & L_j u_j \\ l_j U_j & l_j u_j + a_j b_j \end{bmatrix}.$$

By simply equating entries, we obtain

$$A = \sum_{j=1}^{\alpha} L_j U_j,$$

as desired. \square

We now show how a result close to Theorem 2.8 can be derived from Theorem 3.1.

Theorem 3.4. *If A is an $n \times n$ matrix, and if the $(n + 1) \times (n + 1)$ block matrix $A' = \begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix}$ has displacement rank α with respect to the operator Δ_{Z_1, Z_1^T} , then A has an $(\alpha + 1)$ -fold LUT decomposition.*

Proof. Let α be the Δ_{Z_1, Z_1^T} displacement rank of the block matrix $A' = \begin{bmatrix} A & 0 \\ 0 & 0 \end{bmatrix}$. By Theorem 3.1,

$$A' = \text{Circ}_{lr} + \sum_{j=1}^{\alpha} L(f_j) \cdot \text{Circ}_1(g_j)^T.$$

Here Circ_{lr} is the circulant with the same last row as A' , i.e. all zeroes, so it is the zero matrix.

Note that we can write a circulant as the sum of a lower and an upper Toeplitz matrix. So

$$A' = \sum_{j=1}^{\alpha} L(f_j) \cdot (L_j + U_j) = \sum_{j=1}^{\alpha} L(f_j) \cdot L_j + \sum_{j=1}^{\alpha} L(f_j) \cdot U_j,$$

where the L_j and U_j are lower and upper triangular Toeplitz, respectively. By Lemma 3.5, matrix multiplication preserves lower triangular Toeplitzness, and clearly addition does as well, so

$$\sum_{j=1}^{\alpha} L(f_j) \cdot L_j = L$$

is simply another lower triangular Toeplitz matrix. On the other hand, the identity matrix is certainly upper triangular Toeplitz, so we can write

$$A' = L + \sum_{j=1}^{\alpha} L(f_j) \cdot U_j = L \cdot I + \sum_{j=1}^{\alpha} L(f_j) \cdot U_j = \sum_{j=1}^{\alpha+1} L(f_j) \cdot U_j,$$

where

$$L(f_{\alpha+1}) = L, U_{\alpha+1} = I.$$

This is an $(\alpha + 1)$ -fold LUT decomposition for A' . By Lemma 3.6, we must have an $(\alpha + 1)$ -fold LUT decomposition for A , completing the proof. \square

This theorem demonstrates another fact of possible interest: we are able, in principle, to understand displacements by non-invertible *matrices* (care

should be taken to distinguish this idea from that of non-invertible *operators*) such as Z_0 in terms of displacements by invertible matrices like Z_1 . We have already seen, in Section 2.2, some examples in which invertible matrices are more easily treated. In our next and final Section, we shall see another reason that invertible matrix displacements are appealing, and indeed, our original motivation for considering this kind of blocking.

Chapter 4

Future Directions

In this chapter, we offer some possible avenues for further investigation into the topics discussed in this thesis. We mention some areas of the thesis that can be directly expanded, and also describe some ideas presented in other papers that have interesting connections with our work.

4.1 Matrix Symmetries

To begin with, we describe in some depth an idea that has often informed the direction of our research, but has not fully materialized at this stage. In broad terms, the idea is that displacement operators are closely related to certain ideas from representation theory. To see some of these connections, let us first recall a basic definition.

Definition 4.1. *Let G be a finite group. By a (matrix) representation of G , we mean a homomorphism $\phi : G \rightarrow GL_n(\mathbb{F})$ mapping the elements of G to invertible matrices over a base field \mathbb{F} . We say that ϕ is a representation of degree n .*

We now introduce an idea due to Egner and Püschel (2003), called *matrix symmetries*. This idea has a very natural association with displacement operators, and will provide the connection between displacement operators and group representations. We give two related definitions.

Definition 4.2. *Let M be a rectangular matrix over a field \mathbb{F} , and let ϕ and ψ be two representations of the same finite group G . We say that the pair (ϕ, ψ) is a symmetry of the matrix M if for all $g \in G$ we have*

$$\phi(g) \cdot M = M \cdot \psi(g).$$

Definition 4.3. Let $\phi : G \rightarrow GL_m(\mathbb{F})$ and $\psi : G \rightarrow GL_n(\mathbb{F})$ be two representations of the same finite group G . The intertwining space of ϕ and ψ is the vector space

$$\text{Int}(\phi, \psi) = \{M \in \mathbb{F}^{m \times n} \mid \phi(g) \cdot M = M \cdot \psi(g) \text{ for all } g \in G\}.$$

In other words, the intertwining space of representations ϕ and ψ comprises precisely the matrices of which (ϕ, ψ) is a symmetry.

These objects are closely related to displacement operators. In Egner and Püschel (2003), the authors note that a symmetry of a matrix M “captures redundancy in M given by linear relationships among the entries of M .” This is the same intuition that led us to consider displacement operators—structured matrices have more entries than parameters. Since ϕ and ψ are matrix representations, let $A = \phi(g), B = \psi(g)$ in the definitions above. We see that if (ϕ, ψ) is a symmetry of a matrix M , then $AM = MB$, or $AM - MB = \nabla_{A,B}(M) = 0$. The relationship between displacement operators and matrix symmetries can be described succinctly by the following lemma, which follows easily from the preceding definitions.

Lemma 4.1. Let ϕ and ψ be representations of the same finite group G . Then

$$\text{Int}(\phi, \psi) = \bigcap_{g \in G} \ker(\nabla_{\phi(g), \psi(g)}).$$

In this light, consider our original recovery formula, reproduced below.

Theorem (Theorem 2.6). For all $k \geq 1$, we have

$$M = A^k M B^k + \sum_{i=0}^{k-1} A^i \Delta_{A,B}(M) B^i.$$

We have seen the unit circulant Z_1 is a common choice for the matrices A and B . At the same time, Z_1 represents the generator of a cyclic group in a natural way. and in these cases, we can see the above formula as a sum over the elements of a (cyclic) group. A natural question is whether we can similarly make sense of these forms when the group involved is non-cyclic, or possibly non-abelian. The ideas of Section 4.2 may also be pertinent here.

As we alluded to in the previous Section, these connections give a strong motivation to interpret displacements by non-invertible matrices in terms of displacements by invertible matrices, since all group elements are invertible, and must therefore be represented by invertible matrices. This observation makes our Theorem 3.4 more relevant, and perhaps worth adjusting.

Remember that we did not quite achieve the desired bound for number of terms in Theorem 3.4. Referring to the proof, we can see that this is because we blocked by “padding” the matrix A with all zeroes. We suspect that by judiciously retaining information about the matrix in this extra space, full recovery of the desired formula can be achieved, but accomplishing this goal has proved challenging.

4.2 Group Matrices

In Gader (1990), the author gives the following definition:

Definition 4.4. Let G be a finite group with $|G| = n$, and let R be a ring with identity. Index a matrix $A \in M_n(R)$ (that is, an $n \times n$ matrix with entries in R) by the elements of G , so that we can write $A = [a_{g_i g_j}]$. Then A is a group matrix for G over R if

$$a_{g_i g_j} = a_{g g_i g g_j}$$

for any $g \in G$.

Example 4.1. Gader (1990) gives the following example: Let G be the dihedral group $D_6 = \{e, r, r^2, f, fr, fr^2\}$. A group matrix for D_6 has the form (note that the entries need not be integers—the matrix is intended only to suggest the structure)

$$\begin{bmatrix} 0 & 1 & 2 & 3 & 4 & 5 \\ 2 & 0 & 1 & 4 & 5 & 3 \\ 1 & 2 & 0 & 5 & 3 & 4 \\ 3 & 4 & 5 & 0 & 1 & 2 \\ 4 & 5 & 3 & 2 & 0 & 1 \\ 5 & 3 & 4 & 1 & 2 & 0 \end{bmatrix}.$$

Here, the columns are indexed by the group elements (in the order given above), and the rows are indexed by the inverses of those elements.

As Gader puts it, the structure of group matrices represents invariance with respect to translation *in the Cayley graph*. Group matrices generalize circulants—if the group G is cyclic, the corresponding group matrices are circulant. Gader (1990) derives a close analogue to our circulant-based Theorem 3.1 using group matrices. Some of the work in this thesis could be similarly generalized—it might be interesting to continue thinking of constructions in the context of, say, direct products of groups.

Alternatively, we have seen that thorough research has been done regarding the choice of matrices A and B that yield low-rank displacements

for various kinds of structured matrices. A dual, but less explored idea is to start with interesting A and B and to determine what family of matrices will have low rank displacement with respect to A and B . Since the unit circulants, in particular, have been invaluable as displacing matrices, group matrices could be an interesting starting point.

4.3 Matrix Algebras

The final connection discussed here concerns matrix algebras. A high level way of looking at formulae like Theorem 2.9 is that we have many ways of (re)constructing a matrix out of highly structured building blocks. We commented, back in Section 2.2, that using displacement rank as a metric for structuredness is potentially a big idea. We can now make this notion more precise: having chosen a displacement and an associated set of building blocks (remember that the blocks come from the action of the displacing matrices on column vectors), the displacement rank tells us how many building block products we need to sum to get our original matrix back.

Note, however, that Theorem 2.9 imposes no constraint on the matrix M . Hence any square matrix can be recovered as a sum of products of the kinds of matrices shown. We can regard this as a statement about the *matrix algebra* generated by these classes of matrices. In particular, it includes all matrices. Note that other kinds of matrices, such as circulants, may form subalgebras.

Bozzo and Di Fiore (1995) and Bozzo (1995) both deal with related material. For instance, Bozzo (1995) derives an interesting result about Toeplitz plus Hankel matrices. While such matrices can be constructed using familiar building blocks like circulants and triangular Toeplitz matrices, it is computationally more efficient to use members of certain higher dimensional matrix algebras. While these matrices may not be as tractable themselves, Bozzo shows that the displacement rank of Toeplitz plus Hankel matrices (with respect to the displacement operator associated to these algebras) is particularly low, giving an overall savings.

An immediate question might be whether we can improve upon decompositions such as the one we produced for direct sum displacements. A larger-scale investigation might consider the balance between algebra dimension and displacement rank. That is, can we say whether a given set of building blocks is computationally optimal for constructing matrices of a given type?

Chapter 5

Conclusion

We conclude by summarizing the course this thesis has taken.

We began, in Chapter 2, by gathering the fundamentals of displacement theory and extending some established results to include two matrix constructions: the direct sum and the Kronecker product. In particular, we were able to augment a list of effective displacements by describing effective displacements for matrices constructed from common structured families. We also provided some expanded discussions and proofs that seem to be missing from the literature.

From a practical perspective, the most important thing to absorb from Chapter 2 is an understanding of the displacement rank approach, which came into full focus in Section 2.4. We saw that near-structured matrices admit a computationally efficient sum-of-products decomposition involving structured building blocks. Relating to our work described above, we developed such a decomposition for displacements by direct sums, showing that the direct sum displacement can be understood in terms of displacements by the direct summands.

In Chapter 3, we turned to a higher-level study of displacement operators as algebraic objects. We began with a discussion of operator singularity and its implications for displacement, giving an alternate proof for a theorem of Pan (2001). We then spent some time attempting to characterize displacement kernels, producing several partial results (the lemmas of Section 3.2). Finally, we explored a new blocking method inspired by Orrison (2012), discovering an interesting result that allows us to relate displacements by invertible and non-invertible matrices.

Finally, in Chapter 4, we proposed some directions for future research by pointing out interesting connections between the ideas in this thesis and

the work of other authors. The representation-theoretic connections with Egner and Püschel (2003), the group matrix work in Gader (1990), and the ideas about matrix algebras in Bozzo (1995) and Bozzo and Di Fiore (1995) all seem to suggest exciting continuations of the work begun in this thesis.

Bibliography

Ammar, Gregory, and Paul Gader. 1991. A variant of the Gohberg-Semencul formula involving circulant matrices. *SIAM Journal on Matrix Analysis and Applications* 12(3):534–540.

Axler, Sheldon. 1997. *Linear Algebra Done Right*. Springer.

Bostan, Alin, Eric Schost, and Claude-Pierre Jeannerod. 2006. Using fast matrix multiplication to solve structured linear systems. In *Challenges in Symbolic Computation Software*, eds. Wolfram Decker, Mike Dewar, Erich Kaltofen, and Stephen Watt. Dagstuhl Seminar Proceedings. URL <http://drops.dagstuhl.de/opus/volltexte/2006/778>.

Bozzo, Enrico. 1995. Algebras of higher dimension for displacement decompositions and computations with Toeplitz plus Hankel matrices. *Linear Algebra and its Applications* 230:127–150.

Bozzo, Enrico, and Carmine Di Fiore. 1995. On the use of certain matrix algebras associated with discrete trigonometric transforms in matrix displacement decomposition. *SIAM Journal on Matrix Analysis and Applications* 16(1):312–326.

Egner, S., and M. Püschel. 2003. Symmetry-based matrix factorization. *Journal of Symbolic Computation* 11:157–186.

Gader, Paul D. 1990. Displacement operator based decompositions of matrices using circulants or other group matrices. *Linear Algebra and its Applications* 139:111–131.

Gohberg, I., and V. Olshevsky. 1992. Circulants, displacements and decompositions of matrices. *Integral Equations and Operator Theory* 15(5):730–743.

- . 1994. Complexity of multiplication with vectors for structured matrices. *Linear Algebra and its Applications* 202:163–192.
- Heinig, Georg. 1995. Inversion of generalized Cauchy matrices and other classes of structured matrices. In *Linear algebra for signal processing (Minneapolis, MN, 1992)*, IMA Vol. Math. Appl., vol. 69, 63–81. New York: Springer.
- Heinig, Georg, and Karla Rost. 1984. *Algebraic methods for Toeplitz-like matrices and operators*, Operator Theory: Advances and Applications, vol. 13. Basel: Birkhäuser Verlag.
- Horn, Roger A., and Charles R. Johnson. 1985. *Matrix Analysis*. Cambridge University Press.
- Kailath, T., and V. Olshevsky. 1995. Displacement structure approach to Chebyshev-Vandermonde and related matrices. *Integral Equations and Operator Theory* 22(1):65–92.
- Kailath, Thomas, Sun Yuan Kung, and Martin Morf. 1979. Displacement ranks of matrices and linear equations. *Journal of Mathematical Analysis and Applications* 68(2):395–407.
- Kailath, Thomas, and Vadim Olshevsky. 1997. Displacement-structure approach to polynomial Vandermonde and related matrices. *Linear Algebra and its Applications* 261:49–90.
- Kailath, Thomas, and Ali H. Sayed. 1995. Displacement structure: theory and applications. *SIAM Review A Publication of the Society for Industrial and Applied Mathematics* 37(3):297–386.
- Laub, Alan J. 2005. *Matrix Analysis for Scientists and Engineers*. SIAM.
- Ljung, Lennart. 1999. *System identification : theory for the user*. Upper Saddle River, NJ: Prentice Hall PTR.
- Nagy, James G. 1996. Decomposition of block Toeplitz matrices into a sum of Kronecker products with applications in image restoration.
- Orrison, M. 2012. Personal communication.
- Pan, Victor. 1990. On computations with dense structured matrices. *Mathematics of Computation* 55(191):179–190.

Pan, Victor Y. 2001. *Structured Matrices and Polynomials: Unified Superfast Algorithms*. Boston: Birkhäuser.