5-2005

Wavelet Factorization via a Homogenization Analogy for Solutions of Linear Systems

Andrea Van Sickle

Utah State University

Follow this and additional works at: https://digitalcommons.usu.edu/gradreports

Part of the Mathematics Commons

Recommended Citation

https://digitalcommons.usu.edu/gradreports/1283
WAVELET FACTORIZATION VIA A HOMOGENIZATION ANALOGY FOR SOLUTIONS OF LINEAR SYSTEMS

by

Andrea Van Sickle

A project submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE in

Mathematics

UTHAH STATE UNIVERSITY
Logan, Utah
2005
ABSTRACT

Wavelet Factorization via a Homogenization Analogy for Solutions of Linear Systems

by

Andrea Van Sickle, Master of Science

Utah State University, 2005

Major Professor: Dr. Joseph V. Koebbe
Department: Mathematics and Statistics

A system of linear equations can be solved using a factorization method that produces a wavelet structure and is akin to a homogenization process used in determining the solution of differential equations. The method is dependant on the particular structure of Hadamard matrices and their implementation in a similarity transform. This paper details the development of such a method for systems of size $2^n \times 2^n$, including establishing the theoretical underpinnings necessary to define an orthogonal transform. Also, we present the development of an algorithm to implement the method for a simple $2 \times 2$ system, which will then be the basic building block for developing algorithms to solve higher-ordered systems.

(41 pages)
DEDICATION

As I may never again have a chance to dedicate a piece of work, I would like to dedicate this to those whom I love most.

First to my parents, Rock and Sharon, for always providing a way for me to experience the things I really wanted to and for encouraging and supporting every step I take. To my brothers: Cory, who once said I would never be a mediocrity (although I'm still not quite a believer); Scott, for his unconditional love and endless service; and Derek, who always makes me feel wanted and impresses me more each day.

And finally, to Ryan, without whom I would never have made it to the end. I love you and look forward to many more beginnings and endings with you by my side.
I would like to acknowledge several individuals for their help and support in the completion of this project.

First, I owe a huge thank you to my advisor, Dr. Joe Koebbe, for putting up with and not giving up on me, for giving me ideas when I had run out (which was pretty much all the time,) and for taking the time to talk about non-math topics so that we both might keep our sanity. I would like to acknowledge Dr. LeRoy Beasley for providing some of the knowledge needed to begin the project and thank him for his help and suggestions in improving this document. Thanks also to Dr. Peg Howland, who was willing to give of her time and fill a vacancy at the last minute and gave me a fresh pair of eyes on the ideas presented.

I also wish to acknowledge a small group of women who were invaluable to me personally: the staff assistants (both past and present) of the USU Math Department. Thanks to Nancy Smart for having all the answers, Kayla Olsen for getting me through the graduate student process, Becky Hirst for understanding and waiting a few days to put up flyers announcing my defense, and Linda Skabelund for always keeping me laughing.

Andrea Van Sickle
# CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ABSTRACT</td>
<td>iii</td>
</tr>
<tr>
<td>DEDICATION</td>
<td>iv</td>
</tr>
<tr>
<td>ACKNOWLEDGMENTS</td>
<td>v</td>
</tr>
<tr>
<td>LIST OF TABLES</td>
<td>vii</td>
</tr>
<tr>
<td>1 INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>2 LITERATURE REVIEW ON HADAMARD MATRICES</td>
<td>3</td>
</tr>
<tr>
<td>2.1 Basic Definitions and Theorems</td>
<td>3</td>
</tr>
<tr>
<td>2.1.1 Equivalency of Hadamard Matrices</td>
<td>4</td>
</tr>
<tr>
<td>2.1.2 Properties of Hadamard Matrices</td>
<td>4</td>
</tr>
<tr>
<td>2.1.3 Normalizing Hadamard Matrices</td>
<td>5</td>
</tr>
<tr>
<td>2.2 Existence of Hadamard Matrices</td>
<td>6</td>
</tr>
<tr>
<td>2.3 Applications Using Hadamard Matrices</td>
<td>8</td>
</tr>
<tr>
<td>3 PROBLEM DEFINITION AND RESULTS</td>
<td>10</td>
</tr>
<tr>
<td>3.1 Problem Definition</td>
<td>10</td>
</tr>
<tr>
<td>3.2 Applying Hadamard Matrices to Matrices and Vectors</td>
<td>10</td>
</tr>
<tr>
<td>3.3 Homogenization Based Transform of a $2 \times 2$ System</td>
<td>11</td>
</tr>
<tr>
<td>3.3.1 Verification of the Solution</td>
<td>18</td>
</tr>
<tr>
<td>3.4 Results for a Block $2 \times 2$ System</td>
<td>19</td>
</tr>
<tr>
<td>3.5 Extension to Higher Order Systems</td>
<td>20</td>
</tr>
<tr>
<td>4 ALGORITHM DEVELOPMENT</td>
<td>27</td>
</tr>
<tr>
<td>4.1 Development for a $2 \times 2$ system</td>
<td>27</td>
</tr>
<tr>
<td>4.2 Algorithm for the $2 \times 2$ case</td>
<td>30</td>
</tr>
<tr>
<td>4.3 Extending the Algorithm to $2^n \times 2^n$</td>
<td>31</td>
</tr>
<tr>
<td>5 CONCLUSION</td>
<td>33</td>
</tr>
<tr>
<td>REFERENCES</td>
<td>34</td>
</tr>
<tr>
<td>A IDENTITIES</td>
<td>35</td>
</tr>
</tbody>
</table>
# LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>A .1 Identities involving the multiplication of two permutation matrices, $LR$.</td>
<td>36</td>
</tr>
<tr>
<td>A .2 Identities involving the multiplication of a matrix $A$ by two permutation matrices, $LAR$.</td>
<td>36</td>
</tr>
<tr>
<td>A .3 Identities involving the Kronecker product of the pieces of $T_n$, $L \otimes R$.</td>
<td>36</td>
</tr>
</tbody>
</table>
CHAPTER 1
INTRODUCTION

In the field of applied mathematics, a computational method has been developed in which properties of homogenization have been used to construct wavelet characterizations of coefficients present in differential equations modeling water flow [7]. The averaging formula from homogenization motivated this process, as the average and detail appear explicitly as part of the expression. In solving a two-cell local problem via homogenization techniques, a structure similar to

\[
\begin{align*}
  x_0 &= \frac{\alpha + \beta}{2} \\
  x_1 &= \frac{\alpha - \beta}{2}
\end{align*}
\]

occurs throughout. While a simplification, this can similarly be expressed as a linear system of

\[
\begin{bmatrix}
  x_0 \\
  x_1
\end{bmatrix} = \frac{1}{2} \begin{bmatrix}
  1 & 1 \\
  1 & -1
\end{bmatrix} \begin{bmatrix}
  \alpha \\
  \beta
\end{bmatrix}
\]

This structure appears to continue in the four-cell problem (and beyond), where the solution process is based on two-cell blocks.

The matrix used in the above representation,

\[
\begin{bmatrix}
  1 & 1 \\
  1 & -1
\end{bmatrix}
\]

is what is known as a Hadamard matrix, introduced by Jacques Hadamard in 1893 [2]. Hadamard matrices possess a distinct structure and set of properties, including orthogonality between all rows and columns. The above Hadamard matrix of degree two has a structure which provides the averages and differences representative of a wavelet decomposition. Indeed, applying such a matrix to vectors and other matrices produces a set of equations which demonstrate a wavelet structure. It would make sense, then, that one could extend the wavelet structure from the continuous realm to the discrete case in working out a process to solve systems of linear equations.
In this paper we prove the theoretical underpinnings necessary in developing an orthogonal transform based on such a process, as well as work out much of the development of the method, using homogenization as our guide. We want to first understand the relevant properties of and theorems involving Hadamard matrices, so a brief synopsis has been provided. The next chapter develops a wavelet factorization technique using a homogenization analogy for systems of size $2^n \times 2^n$ and proves the key properties of commutativity and orthogonality within our method. Finally, we take a brief look at the development of an algorithm to implement the defined method.
2.1 Basic Definitions and Theorems

We begin with a standard definition of a Hadamard matrix.

**Definition 2.1.1** A Hadamard matrix of degree \( n \) \((H^{(n)})\) is an \( n \times n \) matrix of \( \pm 1 \)'s that satisfies

\[
HH^T = H^T H = nI_n
\]

where \( I_n \) is the \( n \times n \) identity matrix.

This is equivalent to saying that a square \((1,-1)\)-matrix of degree \( n \) is Hadamard iff any two rows/columns of the matrix are orthogonal [3]. The truth of this statement can be easily seen as \((HH^T)_{ij} = \langle h_i, h_j^T \rangle = 0 \) for \( i \neq j \). Reversing the argument, we look at the \( ij^{th} \) elements for \( HH^T \). The diagonal elements are the dot product of each row with itself, and since each element of \( H \) is \( \pm 1 \), \( \langle h_i, h_i^T \rangle = n \). Such a matrix, \( H \), was originally known as a Hadamard determinant [9], as it satisfies the equality in Hadamard's Maximal Determinant theorem (stated here for matrices with real entries).

**Theorem 2.1.1** [4] Hadamard's Maximal Determinant Theorem

Let \( X = (x_{ij}) \) be a \( n \times n \) real matrix whose entries satisfy \( |x_{ij}| \leq 1 \) for all \( i, j \). Then \( |\det(X)| \leq n^{\frac{3}{2}} \).

**Corollary 2.1.2** [3] Equality for the Maximal Determinant theorem holds iff \( X \) is a matrix of type Hadamard.

Examples of Hadamard matrices, for degrees \( n = 2 \) and \( n = 4 \), include the following.

\[
H^{(2)} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}
\]

\[
H^{(4)} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}
\]
Notice that the determinant of $H^{(2)}$ is -2 (with absolute value 2) and the determinant of $H^{(4)}$ is 16, both of which reflect theorem 2.1.2.

2.1.1 Equivalency of Hadamard Matrices

There exists only one distinct Hadamard matrix of degree two, as all Hadamard matrices of degree two are equivalent to the matrix in equation (2.1). However, for higher orders there are often many distinct Hadamard matrices of the same degree. This leads to the following definition.

**Definition 2.1.2** [6] Two Hadamard matrices of equal degree, $H_1$ and $H_2$, are said to be equivalent if $H_2 = PH_1Q$, where $P$ and $Q$ are monomial matrices (matrices with only one non-zero entry in each row and column) with non-zero entries $\pm 1$.

In this case, $P$ represents the permutation and change of sign of the rows of $H_1$ and $Q$ represents the permutation and change of sign on the columns. For example,

$$H_1 = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \text{ and } H_2 = \begin{bmatrix} 1 & -1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & -1 \\ 1 & -1 & -1 & -1 \end{bmatrix}$$

are equivalent as $H_1 = PH_2Q$ where

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 1 & 0 & 0 \end{bmatrix} \text{ and } Q = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Although there may be many nonequivalent matrices of a given order, they may still be categorized in equivalence classes. For example, the sixty distinct $24 \times 24$ Hadamard matrices fall into only six equivalence classes [2]. Such equivalence classes, as well as those for skew-symmetric, symmetric, and complex Hadamard matrices are detailed in [11].

2.1.2 Properties of Hadamard Matrices

The previous definitions and theorems produce the following properties.

**Remark 1** **Basic Properties of Hadamard matrices:**

Let $H$ be a Hadamard matrix of degree $n$. Then
a) The dot product of any two rows/columns of $H$ with itself is $n$.

b) $\frac{1}{\sqrt{n}} H$ is an orthogonal matrix.

c) For all $x$ in $\mathbb{R}^n$, $\|Hx\| = \sqrt{n}\|x\|$.

d) All (complex) eigenvalues of $H$ have absolute value $\sqrt{n}$.

**Proof:** Property (a) is obvious from the definition of Hadamard matrices. Property (b) also follows from the definition:

$\left( \frac{1}{\sqrt{n}} H \right) \left( \frac{1}{\sqrt{n}} H \right)^T = \frac{1}{n} HH^T = \frac{1}{n} nI_n = I_n$

and $\frac{1}{\sqrt{n}} H$ is an orthogonal matrix. Property (c) follows from (b): since $\frac{1}{\sqrt{n}} H$ is orthogonal,

$\|x\| = \left\| \frac{1}{\sqrt{n}} Hx \right\| = \frac{1}{\sqrt{n}} \|Hx\|$ and we have the result. $\lambda$ is said to be an eigenvalue of $H$ if $Hx = \lambda x$, so this implies

$\|\lambda x\| = |\lambda| \|x\| = \|Hx\| = \sqrt{n}\|x\|$

and by equality, $|\lambda| = \sqrt{n}$

We can see a demonstration of property (d) by looking at two equivalent Hadamard matrices of degree two,

$H_1 = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$ and $H_2 = \begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix}$

The eigenvalues of $H_1$ are $\pm \sqrt{2}$ and the eigenvalues of $H_2$ are $1 \pm i$, all of which have absolute value $\sqrt{2}$.

Also, from property (c) of remark 1 we see that the transformation $T$ defined by $T(x) = L_H x$ (multiplication of a vector on the left by a Hadamard matrix) is a conformal transformation, or a transformation that preserves angles. As we will be applying transformations involving Hadamard matrices, this property will be useful.

2.1.3 Normalizing Hadamard Matrices

In many operations and processes involving matrices, it is beneficial to have a normalized matrix. The following theorem and corollary show that the properties of a Hadamard matrix may be retained through normalization.

The proof of theorem 2.1.3 relies on the definition and properties of the determinant of a matrix and can be found in [11]. The resulting corollary is used in proving future existence theorems.

Corollary 2.1.4 [11] Any Hadamard matrix can be normalized, ie, permuted so that the entries in the first row and first are equal to $+1$.

Permuting the rows or columns (except the first) of a normalized Hadamard matrix leaves it normalized, but generally not all normalized matrices of the same degree are equivalent. Certain properties of a normalized Hadamard matrix are also key to the proofs of existence theorems to be stated shortly. These properties are as follows.

Remark 2 [4] Properties of a normalized Hadamard matrix:

a) Exactly half of the entries of any row/column (except the first) are $+1$.

b) Any two rows/columns agree in $n/2$ places and both have $+1$ in exactly half of these $n/2$ places.

2.2 Existence of Hadamard Matrices

As Hadamard matrices have such convenient structure and properties it would be advantageous to be able to construct such a matrix for every degree $n$. However, this is not the case. Corollary 2.1.4, the properties listed in remark 2, orthogonality, and theorem 2.1.3 are used in the proof of the following theorem.

Theorem 2.2.1 [6] If $H$ is a Hadamard matrix of degree $n$, for $n > 2$, $n$ is necessarily a multiple of 4.

One effect this theorem has is that it limits the type (degree) of linear systems we’ll be able to solve with a factorization based on Hadamard matrices. Although Hadamard
matrices have been shown to exist for all degrees \( n \equiv 0(\text{mod}4), n < 428 \) (often by construction [11]), theorem 2.2.1 gives us only a necessary, not sufficient, condition for the existence of Hadamard matrices. One of the current open problems in coding theory is Hadamard's conjecture.

**Conjecture 2.2.2 Hadamard's Conjecture**

There exists a Hadamard matrix of degree \( n \) for every \( n \equiv 0(\text{mod}4) \).

There have been a variety of approaches in attempting to prove this conjecture. One approach involves orthogonal designs (or symmetric block designs,) a thorough discussion of which can be found in [5] and [6]. Jacques Hadamard actually focused the majority of his studies on complex Hadamard matrices, and his inequality dealt with matrices whose entries are from the unit circle and with matrices of entries ±1 and ±i with orthogonal rows and columns. It has been conjectured that complex Hadamard matrices exist for every order \( n \equiv 0(\text{mod}2) \), and the truth of this conjecture would imply the truth of Hadamard's conjecture [9].

One of the easiest cases to prove for the existence of a Hadamard matrix is for matrices of degree \( n = 2^m \) for some \( m \in \mathbb{N} \). This will be shown through the use of Kronecker (or tensor) products.

**Definition 2.2.1** If \( A = (a_{ij}) \) and \( B = (b_{kl}) \) are matrices of degree \( m \times n \) and \( p \times q \), respectively, then the Kronecker product of \( A \) and \( B \), \( A \otimes B \), is the \( mp \times nq \) matrix made up of \( p \times q \) blocks, where the \((i, j)\) block is \( a_{ij}B \).

**Theorem 2.2.3** [3] The Kronecker product of Hadamard matrices is a Hadamard matrix.

Therefore, a Hadamard matrix of order \( 2^m \) can be generated by taking the Kronecker product of \( m \) copies of the \( 2 \times 2 \) Hadamard matrix. The matrix in equation (2.2), for example, was generated by taking the Kronecker product of two copies of the \( 2 \times 2 \) Hadamard matrix. Such a matrix is also often referred to as a Sylvester matrix [3]. Sylvester matrices also demonstrate the property that the Kronecker product of two symmetric Hadamard
matrices is a symmetric Hadamard matrix. Obviously, Kronecker products can be used in generating other Hadamard matrices not of order $2^m$:

**Corollary 1** [6] If there exist $k \times k$ and $l \times l$ Hadamard matrices, there exists a Hadamard matrix of degree $kl \times kl$.

Many other specific cases for the existence of Hadamard matrices have been proven by construction, several types of which are given a thorough treatment in [1] and [11]. A summary of many of these is given by the following theorem.

**Theorem 2.2.4** [6] If $p$ is an odd prime (or $p = 0$) and $r$ is any positive integer, then it is possible to construct Hadamard matrices of the following orders $n$:

a) $n = 2^r$

b) $n = p^r + 1 \equiv 0 \pmod{4}$

c) $n = h(p^r + 1), h \geq 2$, $h$ order of a Hadamard matrix (this is a generalization of Paley's theorem [12])

d) $n = h(h - 1)$, $h$ a product of numbers of forms (a), (b)

e) $n = h(h + 3)$, where $h$, $h + 4$ both products from (a), (b)

f) $n = h_1h_2(p^r + 1)p^r$, where $h_1$, $h_2 > 1$ orders of Hadamard matrices

g) $n = h_1h_2s(s + 3)$, where $h_1$, $h_2 > 1$ orders of Hadamard matrices and $s$, $s + 4$ both of form $p^r + 1$

h) $n = q(q + 2) + 1$, where $q$, $q + 2$ both of form $p^r$

i) $n$ a product of numbers in (a) to (h)

### 2.3 Applications Using Hadamard Matrices

One of the most common applications that utilizes the structure of Hadamard matrices is in coding and information theory. Hadamard matrices can be used to produce optimal code [11] and improve data acquisition and noise reduction in frequency functions [8]. Transforms
of this nature have been used throughout the sciences (an example being spectroscopy in chemistry [10]) and engineering. Engineers also use Hadamard matrices in direct correlation with Walsh functions in working with semi-conductor technology [11]. A Walsh function is a function consisting of a square pulse (with the allowed states being 1 and -1) such that transitions may only occur at fixed intervals of a unit time step. The $2^n$ Walsh functions of order $n$ are given by the rows of the Hadamard matrix $H_{2^n}$ [13].

Hadamard matrices have been applied throughout mathematics. Statisticians have used them in cooperation with block and weighing designs and in determining pairwise statistical independence. Hadamard matrices have a place in graph theory in working with tournaments and groups, as well as in combinatorics, cryptography, and other fields of applied mathematics. A thorough treatment of these and other applications can be found in [1] and [11].
3.1 Problem Definition

Let $Ax = b$ be a linear system of size $2^n \times 2^n$. Our goal is to develop a factorization method utilizing the wavelet structure that results from applying a Hadamard transform to solve the system. The homogenization process used in finding the solution to a system of differential equations will be our guide throughout the development of this method.

We want to first find a viable method to solve a basic $2 \times 2$ system and then work with that as a basic building block to see if the method could perhaps be extended to find a solution of a larger system.

3.2 Applying Hadamard Matrices to Matrices and Vectors

Before we look at the factorization of a linear system using a homogenization analogy, it is essential to understand how Hadamard matrices relate to this process. It was noted earlier that the Hadamard structure occurs repeatedly in the wavelet and homogenization process, and as we switch to a discrete setting it is the basis of our factorization method.

Consider the degree two Hadamard matrix,

$$H^{(2)} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

Notice that applying this matrix to a two-vector, $a = [a_1 \ a_2]^T$, results in

$$H^{(2)}a = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} a_1 + a_2 \\ a_1 - a_2 \end{bmatrix}$$

The first value represents a constant multiple of the average of the entries of the original vector, and the second value gives the difference, or detail, of the two. Knowing these two values, we can easily reconstruct the original vector. This structure is representative of the wavelet structure found throughout the solution of a two cell local problem in homogenization.

If the Hadamard matrix of degree two is applied to both sides of an arbitrary $2 \times 2$
matrix, 

\[
A = \begin{bmatrix}
  a_{11} & a_{12} \\
  a_{21} & a_{22}
\end{bmatrix}
\]

the result is

\[
H^{(2)} A H^{(2)} = \begin{bmatrix}
  1 & 1 \\
  1 & -1
\end{bmatrix} \begin{bmatrix}
  a_{11} & a_{12} \\
  a_{21} & a_{22}
\end{bmatrix} \begin{bmatrix}
  1 & 1 \\
  1 & -1
\end{bmatrix} = \begin{bmatrix}
  \bar{a} & \hat{a} \\
  \hat{a} & \bar{a}
\end{bmatrix}
\]

where

(3.1) \quad \bar{a} = a_{11} + a_{12} + a_{21} + a_{22}

(3.2) \quad \hat{a} = a_{11} - a_{12} + a_{21} - a_{22}

(3.3) \quad \hat{a} = a_{11} + a_{12} - a_{21} - a_{22}

(3.4) \quad \bar{a} = a_{11} - a_{12} - a_{21} + a_{22}

In this case, \( \bar{a} \) represents a multiple of the average of the entries, \( \hat{a} \) is the difference between the columns of \( A \), \( \hat{a} \) represents the difference between the rows of \( A \), and \( \bar{a} \) is the difference between the diagonals of the matrix. Once again, given these four values the original entries of \( A \) can be easily computed. Similarly, if the Kronecker product of two Hadamard matrices of degree two is applied to a four-vector, we get essentially the same results:

\[
\begin{bmatrix}
  1 & 1 & 1 & 1 \\
  1 & -1 & 1 & -1 \\
  1 & 1 & -1 & -1 \\
  1 & -1 & -1 & 1
\end{bmatrix} \begin{bmatrix}
  a_1 \\
  a_2 \\
  a_3 \\
  a_4
\end{bmatrix} = \begin{bmatrix}
  \bar{a} \\
  \hat{a} \\
  \hat{a} \\
  \bar{a}
\end{bmatrix}
\]

Here, \( \bar{a} = a_1 + a_2 + a_3 + a_4 \), etc.

These two results suggest that perhaps the best way to produce a wavelet structure from a \( 2^n \times 2^n \) matrix \( A \) would be by implementing a similarity transform using a matrix of Hadamard structure. This will be our approach.

### 3.3 Homogenization Based Transform of a 2 x 2 System

In our factorization method the basic building block will be the solution of a simple \( 2 \times 2 \) linear system with coefficient matrix \( A \) defined by

\[
A = \begin{bmatrix}
  a_{11} & a_{12} \\
  a_{21} & a_{22}
\end{bmatrix}
\]
with arbitrary \( a_{ij} \in \mathbb{R} \). We can compute an alternate linear system by transforming the original \( 2 \times 2 \) system of \( Ax = b \) using the matrix \( H \), defined by

\[
H = P_0 + \epsilon^{-1} P_1 = \begin{bmatrix}
\frac{3}{2} & -\frac{1}{2} \\
-\frac{1}{2} & \frac{3}{2}
\end{bmatrix}
\]

where \( \epsilon = \frac{1}{2} \) and the two projection matrices, \( P_0 \) and \( P_1 \), are defined by

\[
P_0 = \begin{bmatrix}
\frac{1}{2} & \frac{1}{2} \\
\frac{1}{2} & \frac{1}{2}
\end{bmatrix}
\]

and

\[
P_1 = \begin{bmatrix}
\frac{1}{2} & -\frac{1}{2} \\
-\frac{1}{2} & \frac{1}{2}
\end{bmatrix}
\]

While \( H \) is not a Hadamard matrix, notice that the column spaces of the rank one projection matrices, \( P_0 \) and \( P_1 \), coincide with the spaces spanned by the individual columns of the Hadamard matrix of degree two. Also, applying the two projections, \( P_0 \) and \( P_1 \), to a vector results in the average of the two components and a multiple of the difference of the two components, respectively, similar to the transformation discussed before. With respect to homogenization, this is exactly the same type of projection that occurs in an appropriate description of the solution of a two-cell problem in homogenization techniques [7].

The inverse of the transformation matrix, \( H \), is easy to compute and is given by

\[
H^{-1} = \begin{bmatrix}
\frac{3}{4} & \frac{1}{4} \\
\frac{1}{4} & \frac{3}{4}
\end{bmatrix} = P_0 + \epsilon P_1
\]

where again, \( \epsilon = \frac{1}{2} \). The parameter is left in place to take advantage of the analogy to computation of effective values in the solution of the local problems in homogenization.

With these definitions, the alternate system we will use is

\[
HAH^{-1}x = Hb.
\]

This system can be written as

\[
HAH y = r
\]

with

\[
y = H^{-1}x
\]
and

\( (3.9) \quad r = Hb. \)

As in the definition of the local problem in homogenization theory we need an auxiliary equation of the form

\( (3.10) \quad v = AHy \)

and substitute this into the system in (3.7). The result is

\( (3.11) \quad Hv = r. \)

These two equations look like the first order system of differential equations that define the local problem in homogenization.

Using the decompositions of \( H \) and \( H^{-1} \), our system can be thought of as a perturbation problem in the parameter \( \epsilon \). Thus, the ansatz in solving the perturbed system is that the solutions can be expanded in terms of \( \epsilon \) as follows:

\[
Y = Y_0 + \epsilon Y_1 \\
V = V_0 + \epsilon V_1
\]

Substitution of the above solution expansions into equations (3.10) and (3.11) results in the system

\( (3.12) \quad v_0 + \epsilon v_1 = A(P_0 + \epsilon^{-1}P_1)(y_0 + \epsilon y_1) \)

\( (3.13) \quad (P_0 + \epsilon^{-1}P_1)(v_0 + \epsilon v_1) = r \)

The computation of the solution for these equations is analogous to the process used in homogenization.

There is, however, an important difference. If this system is calculated with the transformed value, \( r \), unperturbed, the system is inconsistent. Therefore, in this finite dimensional setting, \( r \) must be expressed in terms of the perturbation series. In particular, we need to write equation (3.9) as

\( (3.14) \quad Hb = (P_0 + \epsilon^{-1}P_1)b = r_0 + \epsilon^{-1}r_1 \)
Perturbing \( r \) also causes equation (3.13) to look like

\[
(P_0 + \varepsilon^{-1}P_1)(v_0 + \varepsilon v_1) = r_0 + \varepsilon^{-1}r_1
\]

This decomposition will allow us to compute a unique solution.

One might want to know how the inclusion of \( r_1 \) relates to the homogenization analysis and the development of the local problem. In the perturbation analysis one computes the average of the forcing function, \( f \), on the local or microscopic scale using

\[
f^\# = \int_Y f(y)dy
\]

where \( Y \) denotes the domain associated with the local problem. If one thinks of \( r \) as a vector that is used to sample a function like the forcing function in the homogenization analysis, then \( r_1 \) represents the difference between the exact function definition and the average, \( f^\# \), defined on the local problem domain. In the homogenization process, it is assumed that as \( \varepsilon \) tends to zero that \( f^\# \) will converge to the exact value of the function and guarantee that the solution of the heterogeneous problem will converge to the solution of the homogenized problem. The important point is that this will be true as \( \varepsilon \) tends to zero. If we assume that \( f \) is smooth enough (actually \( f \in L_2 \)) it is not hard to see how \( r_1 \) will tend to zero.

Expanding equations (3.12), (3.14), and (3.15) and equating powers results in the following system:

- **\( O(\varepsilon^{-1}) \): Perturbation Equations**

  \[
  \begin{align*}
  (3.16) & \quad P_1 v_0 = r_1 \\
  (3.17) & \quad AP_1 y_0 = 0 \\
  (3.18) & \quad P_1 b = r_1
  \end{align*}
  \]

- **\( O(\varepsilon^0) \): Perturbation Equations**

  \[
  \begin{align*}
  (3.19) & \quad P_0 v_0 + P_1 v_1 = r_0 \\
  (3.20) & \quad AP_0 y_0 + AP_1 y_1 = v_0 \\
  (3.21) & \quad P_0 b = r_0
  \end{align*}
  \]
• $O(\epsilon^1)$: Perturbation Equations

(3.22) \[ P_0 v_1 = 0 \]

(3.23) \[ A P_0 y_1 = v_1 \]

The next step is to interpret these equations and compute a solution.

First, we can use equations (3.18) and (3.21) to find the general form of the components of $r$. For $r_0$ we have the vector equation

\[
\begin{align*}
  r_0 &= \begin{bmatrix} r_{01} \\ r_{02} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \\
  &= \frac{1}{2} \begin{bmatrix} b_1 + b_2 \\ b_1 + b_2 \end{bmatrix} \\
  &= \frac{1}{2}(b_1 + b_2) \begin{bmatrix} 1 \\ 1 \end{bmatrix}
\end{align*}
\]

and for $r_1$ we expand the equation to get

\[
\begin{align*}
  r_1 &= \begin{bmatrix} r_{11} \\ r_{12} \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} \\
  &= \frac{1}{2} \begin{bmatrix} b_1 - b_2 \\ -b_1 + b_2 \end{bmatrix} \\
  &= \frac{1}{2}(b_1 - b_2) \begin{bmatrix} 1 \\ 1 \end{bmatrix}
\end{align*}
\]

Thus, $r_0$ and $r_1$ correspond to the average and difference of the components of the solution vector.

If we assume that $A$ is not fixed, equation (3.17) implies that $y_0$ must be orthogonal to the column space of $P_1$. So, we can write

\[
y_0 = \begin{bmatrix} y_{01} \\ y_{11} \end{bmatrix} = y_{01} \begin{bmatrix} 1 \\ 1 \end{bmatrix}
\]

This corresponds to a projection onto an average.

The other quick interpretation involves equation (3.22). In this case, the vector $v_1$ must be orthogonal to the column space of $P_0$. This implies that

\[
v_1 = \begin{bmatrix} v_{11} \\ v_{12} \end{bmatrix} = v_{11} \begin{bmatrix} 1 \\ -1 \end{bmatrix}
\]
In turn this relationship can be used to compute the vector \( y_1 \) using equation (3.23). Multiplying both sides of the equation by \( P_0 \), we get

\[
P_0 v_1 = 0 = P_0 A P_0 y_1 = \frac{1}{2} \tilde{a} P_0 y_1
\]

with \( \tilde{a} \) as described in section 3.1. Again, for general \( A \), this implies that \( y_1 \) must be in the column space orthogonal to \( P_0 \), and we end up with

\[
y_1 = \begin{bmatrix} y_{11} \\ y_{12} \end{bmatrix} = y_{11} \begin{bmatrix} 1 \\ -1 \end{bmatrix}
\]

Going back to the original equation, this in turn implies that \( v_1 = 0 \).

There are three equations left, and it is worth noting that the analysis to this point has decomposed the solution into a linear combination of vectors from the ranges of \( P_0 \) and \( P_1 \) provided we can find \( y_{01} \) and \( y_{11} \).

Equation (3.16) can be used to make progress on \( v_0 \). We have

\[
\begin{bmatrix} \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} v_{01} \\ v_{02} \end{bmatrix} = r_{11} \begin{bmatrix} 1 \\ -1 \end{bmatrix}
\]

This implies that \( v_{02} = v_{01} - 2r_{11} \) and

\[
v_0 = \begin{bmatrix} v_{01} \\ v_{02} \end{bmatrix} = \begin{bmatrix} v_{01} \\ v_{01} - 2r_{11} \end{bmatrix}
\]

This value will be used in the \( \epsilon^0 \) equations to complete the solution process.

In a previous analysis we found that \( v_1 = 0 \). So, we can replace equation (3.19) with

\[
P_0 v_0 = r_0
\]

Note that this equation corresponds to the average Darcy velocity equation in the homogenization analysis [7]. This result implies

\[
r_{01} = \frac{1}{2} (v_{01} + v_{02}) = \frac{1}{2} (v_{01} + v_{01} - 2r_{11}) = v_{01} - r_{11}
\]
and thus we obtain \( v_0 = r_0 + r_{11} \) and

\[
\begin{bmatrix}
  r_{01} + r_{11} \\
  r_{01} - r_{11}
\end{bmatrix} = \begin{bmatrix}
  b_1 \\
  b_2
\end{bmatrix}
\]

This makes sense because when we rewrite equation (3.7) in terms of \( v \), we get

\[
Hv = r = Hb
\]

implying \( v = b \), and since \( v_1 = 0 \), \( v_0 \) must equal \( b \).

After all this work we can use equation (3.20) to obtain our final result. We end up with

\[
\begin{bmatrix}
  r_{01} + r_{11} \\
  r_{01} - r_{11}
\end{bmatrix} = AP_0 y_0 + AP_1 y_1
\]

\[
= \begin{bmatrix}
  (a_{11} + a_{12}) & (a_{11} + a_{12}) \\
  (a_{21} + a_{22}) & (a_{21} + a_{22})
\end{bmatrix} \begin{bmatrix}
  y_0 \\
  y_0
\end{bmatrix}
+ \begin{bmatrix}
  (a_{11} - a_{12}) & (-a_{11} + a_{12}) \\
  (a_{21} - a_{22}) & (-a_{21} + a_{22})
\end{bmatrix} \begin{bmatrix}
  y_1 \\
  -y_1
\end{bmatrix}
\]

By adding and subtracting the two rows of this system together, one can see that it can be written in the form

\[
\begin{bmatrix}
  \hat{a} & \hat{a} \\
  \hat{a} & \hat{a}
\end{bmatrix} \begin{bmatrix}
  y_{01} \\
  y_{11}
\end{bmatrix} = 2 \begin{bmatrix}
  r_{01} \\
  r_{11}
\end{bmatrix}
\]

where \( \hat{a}, \hat{a}, \hat{a}, \) and \( \hat{a} \) are defined as before. The values of \( y_0 \) and \( y_1 \) can be computed from this system of equations.

In particular, the solution for this system is

\[
\begin{bmatrix}
  y_{01} \\
  y_{11}
\end{bmatrix} = \frac{2}{\alpha} \begin{bmatrix}
  \hat{a} r_0 - \hat{a} r_1 \\
  -\hat{a} r_0 + \hat{a} r_1
\end{bmatrix}
\]

with \( \alpha \) the determinant of the coefficient matrix for our latest incarnation of the system. Thus the solution can be computed by substituting the values above into the perturbation representation for \( y \) to obtain

\[
y = y_0 + \epsilon y_1
\]

with \( \epsilon = \frac{1}{2} \) as before. This solution is the exact solution one can obtain using the usual closed form of the solution of a 2 \( \times \) 2 linear system. Once we have the solution of the
transformed system it is a simple matter to compute the solution, \( x \), for the original system of equations.

One might ask why we would be interested in such a complicated way of computing the solution of such a simple system. However, the recursive application of this idea combined with orthogonality will produce a transform approach for solving larger linear systems, as we will show.

3.3.1 Verification of the Solution

The usual closed form of the solution of a 2 \( \times \) 2 linear system, \( Ax = b \), is calculated as

\[
x = A^{-1} b
\]

\[
= \frac{1}{\det A} \begin{bmatrix}
    a_{22} & -a_{12} \\
    -a_{21} & a_{11}
\end{bmatrix}
\begin{bmatrix}
    b_0 \\
    b_1
\end{bmatrix}
\]

\[
= \frac{1}{a_{11} a_{22} - a_{12} a_{21}} \begin{bmatrix}
    a_{22} b_0 - a_{12} b_1 \\
    -a_{21} b_0 + a_{11} b_1
\end{bmatrix}
\]

To find the solution using this homogenization process, we must first find \( y \):

\[
y = y_0 + \epsilon y_1
\]

\[
= y_0 \begin{bmatrix}
    1 \\
    1
\end{bmatrix} + \frac{1}{2} y_1 \begin{bmatrix}
    1 \\
    -1
\end{bmatrix}
\]

\[
= \frac{2}{\alpha} \begin{bmatrix}
    \hat{a}r_{01} - \bar{a}r_{11} \\
    \bar{a}r_{01} - \bar{a}r_{11}
\end{bmatrix}
+ \frac{1}{\alpha} \begin{bmatrix}
    -\hat{a}r_{01} + \bar{a}r_{11} \\
    \hat{a}r_{01} - \bar{a}r_{11}
\end{bmatrix}
\]

which simplifies to

\[
y = \frac{1}{\alpha} \begin{bmatrix}
    -a_{21} b_1 + 3a_{22} b_1 + a_{11} b_2 - 3a_{12} b_2 \\
    -3a_{21} b_1 + a_{22} b_1 + 3a_{11} b_2 - a_{12} b_2
\end{bmatrix}
\]

Now, to find \( x \), we multiply \( y \) by \( H \) and get the following:

\[
x = Hy
\]

\[
= \frac{1}{\alpha} \begin{bmatrix}
    \frac{3}{2} & -\frac{1}{2} \\
    -\frac{1}{2} & \frac{3}{2}
\end{bmatrix} \begin{bmatrix}
    -a_{21} b_1 + 3a_{22} b_1 + a_{11} b_2 - 3a_{12} b_2 \\
    -3a_{21} b_1 + a_{22} b_1 + 3a_{11} b_2 - a_{12} b_2
\end{bmatrix}
\]

\[
= \frac{1}{\alpha} \begin{bmatrix}
    4a_{22} b_1 - 4a_{12} b_2 \\
    -4a_{21} b_1 + 4a_{11} b_2
\end{bmatrix}
\]

Calculating for \( \alpha \) gives

\[
\alpha = \bar{a}a - \hat{a}a
\]

\[
= 4a_{11} a_{22} - 4a_{12} a_{21}
\]
Plugging everything in, we get the correct solution of

\[ x = \frac{1}{a_{11}a_{22} - a_{12}a_{21}} \begin{bmatrix} a_{22}b_0 - a_{12}b_1 \\ -a_{21}b_0 + a_{11}b_1 \end{bmatrix} \]

3.4 Results for a Block 2 \times 2 System

As this analogy and solution process have been proven to work on a simple 2 \times 2 system, it would be beneficial if we could use this structure to solve larger linear systems, specifically those of size 2^n \times 2^n. One way to approach this is the break the 2^n \times 2^n coefficient matrix into four blocks of size 2^{n-1} \times 2^{n-1} and the corresponding solution vector into two vectors of size 2^{n-1}, causing \( Ax = b \) to be of the form

\[
\begin{bmatrix}
A_{11} & A_{12} \\
A_{21} & A_{22}
\end{bmatrix}
\begin{bmatrix}
X_1 \\
X_2
\end{bmatrix}
= \begin{bmatrix}
B_1 \\
B_2
\end{bmatrix}
\]

In this way, a block 2 \times 2 system is created to which we may apply the same transformation method, writing equation \((3.6)\) as

\[
(3.24) \quad H_n A H_n H_n^{-1} X = H_n B
\]

In this case, \( H_n \) is defined as

\[
(3.25) \quad H_n = P_0^{(n)} + \epsilon^{-1} P_1^{(n)} = \begin{bmatrix}
\frac{3}{2} I_{2^{n-1}} & -\frac{1}{2} I_{2^{n-1}} \\
-\frac{1}{2} I_{2^{n-1}} & \frac{3}{2} I_{2^{n-1}}
\end{bmatrix}
\]

where \( \epsilon = \frac{1}{2} \) and the projection matrices are defined as

\[
P_0^{(n)} = \begin{bmatrix}
I_{2^{n-1}} & I_{2^{n-1}} \\
I_{2^{n-1}} & I_{2^{n-1}}
\end{bmatrix}
\]

and

\[
P_1^{(n)} = \begin{bmatrix}
I_{2^{n-1}} & -I_{2^{n-1}} \\
-I_{2^{n-1}} & I_{2^{n-1}}
\end{bmatrix}
\]

Notice that \( H = H_1 \).

Using the same system of equations derived from the earlier perturbation equations results in

\[
R_0 = \frac{1}{2} \begin{bmatrix}
B_1 + B_2 \\
B_1 + B_2
\end{bmatrix}
\]
\[ R_1 = \frac{1}{2} \begin{bmatrix} B_1 - B_2 \\ -B_1 + B_2 \end{bmatrix} \]

\[ V_0 = \begin{bmatrix} B_1 \\ B_2 \end{bmatrix} \]

\[ V_1 = 0 \]

\[ Y_0 = \begin{bmatrix} Y_{01} \\ Y_{01} \end{bmatrix} \]

\[ Y_1 = \begin{bmatrix} Y_{11} \\ -Y_{11} \end{bmatrix} \]

where \( B_1, B_2, Y_{01}, \) and \( Y_{11} \in \mathbb{R}^{2n-1} \). Solving for \( Y_{01} \) and \( Y_{11} \) gives

\[ Y_{01} = 2\alpha \left[ \tilde{A}(B_0 + B_1) - \tilde{A}(B_0 - B_1) \right] \]

\[ Y_{11} = 2\alpha \left[ -\tilde{A}(B_0 + B_1) + \tilde{A}(B_0 - B_1) \right] \]

where in this case

\[ \alpha = (\tilde{A}\tilde{A} - \tilde{A}\tilde{A})^{-1} \]

This method does not, of course, result in the exact solution to the system, but instead gives an approximation. This is analogous to working with the homogenization process at the coarsest level. The approximation will not be accurate enough to use unless the information contained in the finer scales can be worked into the process. The next section looks at how this may be done.

### 3.5 Extension to Higher Order Systems

The goal is to apply the 2 × 2 results from the previous sections to build a more general solver. To do this, an appropriate transform must be constructed from the building block in the two dimensional case. We will consider systems of order \( n = 2^m \) for some integer \( m \). The solution method will be akin to the application of a fast transform method to a given signal and will look to take advantage of orthogonality properties imposed by higher order Hadamard matrices.

The approach here will use a recursive algorithm defined by taking Kronecker products of \( H \) to create a transform matrix. One motivation for this approach is that the Kronecker products of \( H \) will preserve the orthogonality between the columns and will continue to
produce the column spaces that provide the necessary detail in the perturbation equations.

For a given \( m = 1, 2, \ldots \) we will compute

\[
T_n = \bigotimes_{i=1}^{n} H_1
\]

This product, although a reasonable approach, is not computationally efficient. In addition, we need a way to identify dyadic scales. The basic reformulation depends on the following identity for Kronecker products.

\[
(A \otimes B)(C \otimes D) = (AC) \otimes (BD)
\]

if the matrices involved have appropriate row and column dimensions. The identity can be extended to any number of matrices of appropriate dimension on the right side of the equality, since

\[
(ABC) \otimes (DEF) = (A \otimes D)(BC \otimes EF) = (A \otimes D)(B \otimes E)(C \otimes F)
\]

and so forth.

Applying the identity to \( H_1 \otimes H_1 \), we have

\[
H_1 \otimes H_1 = (I_2 \otimes H_1)(H_1 \otimes I_2) = (I_2 \otimes H_1)(H_1 \otimes I_2)
\]

\[
= \begin{bmatrix} H_1 & 0 \\ 0 & H_1 \end{bmatrix} \begin{bmatrix} 3I_2 & -I_2 \\ -I_2 & 3I_2 \end{bmatrix} = D_{1,1} H_2 = D_{1,1} D_{2,0}
\]

where \( I_2 \) is the identity matrix of order 2 and \( D_{i,j} \) is defined as the \( 2^{i+j} \times 2^{i+j} \) block diagonal matrix consisting of \( 2^j \) blocks of \( H_i \). Notice that, by this definition, \( H_n = D_{n,0} \). This application of the identity can be implemented for any number of Kronecker products of \( H_1 \) and provides us with the following formulation for \( T_n \).

**Proposition 3.5.1** \( T_n = \prod_{i=1}^{n} D_{i,n-i} \)

**Proof:** By induction:
For $n = 1$, the result holds, as

$$ T_1 = H_1 = D_{1,0} $$

So assume that the equation holds for $T_m$. Then

\[
T_{m+1} = H_1 \otimes T_m
= I_2 H_1 \otimes (D_{1,m-1} D_{2,m-2} \ldots D_{m-1,1} D_{m,0})
= (I_2 \otimes D_{1,m-1} D_{2,m-2} \ldots D_{m-1,1})(H_1 \otimes D_{m,0})
= [(I_2)^{m-1} \otimes D_{1,m-1} D_{2,m-2} \ldots D_{m-1,1}][I_2 H_1 \otimes D_{m,0} I_{2^m}]
= (I_2 \otimes D_{1,m-1}) (I_2 \otimes D_{2,m-2}) \ldots (I_2 \otimes D_{m-1,1}) (I_2 \otimes D_{m,0})(H_1 \otimes I_{2^m})
\]

By the identities of the Kronecker products of $I_{2^i}$, $H_j$, and $D_{k,l}$ listed in Appendix A, this gives

$$ T_{m+1} = D_{1,m} D_{2,m-1} \ldots D_{m,1} D_{m+1,0} = \prod_{i=1}^{m+1} D_{i,m+i-1} $$

and $T_m \Rightarrow T_{m+1}$. \( \blacksquare \)

Using this transform, for a linear system of size $2^n \times 2^n$ we will redefine the alternate system found in equation (3.6) as

$$ T_n A T_n \ T_n^{-1} x = T_n b $$

which is, by definition,

$$ (D_{1,m-1} \ldots D_{m,0}) A (D_{1,m-1} \ldots D_{m,0}) (D_{1,m-1} \ldots D_{m,0})^{-1} x = (D_{1,m-1} \ldots D_{m,0}) b $$

Notice that, concerning the individual $D_{i,m-i}$ matrices of $T_n$, we no longer have a similarity transform. What we would like is to have $A$ acted upon as

$$ (D_{m,0} \ldots D_{1,m-1}) A (D_{1,m-1} \ldots D_{m,0}) $$

With this reformulation, $A$ is first broken down into $2 \times 2$ blocks, and $D_{1,n-1}$ on either side of the matrix allows $H_1$ to act upon each block individually to perform the basic
transform. The next piece of $T_n$, $D_{2,n-2}$, works on the next coarser scale, taking the results of the individual $2 \times 2$ blocks and working the transformation on a $4 \times 4$ (four blocks of $2 \times 2$) piece of the matrix. The process continues in such a manner until $D_{n,0}$ acts upon the final blocks, each of size $2^{n-1}$. This is akin to the wavelet characterization in that we work from the finest scale to the coarsest in performing the transform, and likewise, when reconstructing the solution we will work from the coarsest scale to the finest.

Thus, for the process to be valid, the commutativity of the individual pieces of $T_n$ must be established.

**Proposition 3.5.2** For $1 \leq i \neq j \leq m$,

$$D_{i,m-i} D_{j,m-j} = D_{j,m-j} D_{i,m-i}$$

**Proof:** By verification of the following three claims.

**Claim 1:** Let $M_1 = \text{diag}(A_1, A_2, \ldots, A_k)$ and $M_2 = \text{diag}(B_1, B_2, \ldots, B_k)$. If $A_i B_i = B_i A_i$ for $i = 1, 2, \ldots, k$, then $M_1 M_2 = M_2 M_1$.

**Proof:** By calculation:

$$M_1 M_2 = \text{diag}(A_1, A_2, \ldots, A_k) \cdot \text{diag}(B_1, B_2, \ldots, B_k)$$

$$= \text{diag}(A_1 B_1, A_2 B_2, \ldots, A_k B_k)$$

$$= \text{diag}(B_1 A_1, B_2 A_2, \ldots, B_k A_k)$$

$$= \text{diag}(B_1, B_2, \ldots, B_k) \cdot \text{diag}(A_1, A_2, \ldots, A_k)$$

$$= M_2 M_1$$

**Claim 2:** $D_{i,m-i} D_{j,m-j} = D_{j,m-j} D_{i,m-i}$ for $1 \leq i \neq j < m$.

**Proof:** Without loss of generality, assume $j < i$. Then $H_i$ is the same size as

$$\text{diag} \left( H_j, \ldots, H_j \right)$$

where $l = 2^{i-j}$. Thus,

$$D_{j,m-j} D_{i,m-i} = \text{diag} \left( H_j, \ldots, H_j \right) \cdot \text{diag} \left( H_i, \ldots, H_i \right)$$

$$= \text{diag} \left( H_j, \ldots, H_j \right) \cdot \text{diag} \left( H_i, \ldots, H_i \right)$$
\[= \text{diag} \left[ \text{diag} \left( H_j, \ldots, H_j \right) \right] \cdot \text{diag} \left( H_i, \ldots, H_i \right)\]

\[\text{times} \quad 2^{i-j} \text{ times} \quad \text{times} \quad 2^{m-i} \text{ times} \quad \text{times} \quad 2^{m-i} \text{ times} \]

Notice that at this point there are still \(2^{m-i} \cdot 2^{i-j} = 2^{m-j}\) blocks total blocks of \(H_j\). By claim 1, this is equal to

\[
\text{diag} \left( H_i, \ldots, H_i \right) \cdot \text{diag} \left[ \text{diag} \left( H_j, \ldots, H_j \right) \right] = D_{i,m-i} D_{j,m-j}
\]

Claim 3: \(\text{diag} \left( H_j, \ldots, H_j \right) \cdot H_i = H_i \cdot \text{diag} \left( H_j, \ldots, H_j \right)\)

**Proof:** Using a similar method to the proof of the previous lemma, we have

\[
\text{diag} \left( H_j, \ldots, H_j \right) = \text{diag} \left[ \text{diag} \left( H_j, \ldots, H_j \right), \text{diag} \left( H_j, \ldots, H_j \right) \right]
\]

with each diagonal of dimension \(2^{(i-1)}\). Thus,

\[
diag \left( H_j, \ldots, H_j \right) \cdot H_i
\]

\[\text{times} \quad 2^{i-j} \text{ times} \quad \text{times} \quad 2^{i-j} \text{ times} \quad \text{times} \quad 2^{i-j} \text{ times} \]

Thus by claims 1, 2, and 3 the result is established. \(\Box\)

\(T_n\) applied to a vector works in the same manner as the transform applied to a matrix. \(T_n\) first acts on subvectors of length two, outputting a multiple of the average and difference of the entries of each subvector. Then, \(T_n\) moves to the next coarser scale, finding the average and detail of the entries of two modified subvectors, and so forth until finishing with the
two subvectors of length $2^{n-1}$. Thus, this property of commutativity allows us to identify scaling within our transform method.

One other property that is essential for this transformation process (especially when developing an algorithm based on recurring patterns) is that of orthogonality. Another way to interpret $T_n$ is

$$T_n = \bigotimes_{l=1}^{n} (P_0 + \epsilon^{-1}P_1)$$

This can also be written as

$$T_n = \sum_{i=0}^{n} \left[ \sum_{all\ pert} \left( \bigotimes_{n-i}^{i} P_0 \bigotimes_i P_1 \right) \epsilon^{-i} \right]$$

where the second sum is the sum of all possible perturbations of that particular number of $P_0$ and $P_1$ matrices. For example, $T_2$ can be written as

$$(P_0 \otimes P_0) + [(P_0 \otimes P_1) + (P_1 \otimes P_0)]\epsilon^{-1} + (P_1 \otimes P_1)\epsilon^{-2}$$

Notice that $P_0 \otimes P_0$ and $P_1 \otimes P_1$ are rank one matrices, while $(P_0 \otimes P_1 + P_1 \otimes P_0)$ is a rank two matrix. If these matrices are orthogonal, their column spaces would form a basis for $\mathbb{R}^4$. Accordingly, if we can show orthogonality between the pieces of $T_n$, we can use the column spaces to form a basis for $\mathbb{R}^n$.

In the homogenization analogy, $y_0$ (the coefficient of the $\epsilon^0$ term in $y = y_0 + \epsilon y_1 + \ldots$) represents an approximate solution generated by the averages of the entries of $A$ and $r$. Each power of $\epsilon$ multiplied by $y$ provides a bit more detail to the approximation, producing a more accurate solution. With the column spaces of the pieces of $T_n$ acting as a basis for $\mathbb{R}^n$, we could simply add the subsequent powers of $\epsilon$ to each other without having to recompute the previous terms; hence the need for orthogonality. The following proposition is sufficient to provide the desired orthogonality for the pieces of $T_n$.

**Proposition 3.5.3** For all $0 \leq i \neq j \leq n$,

$$\left( \bigotimes_{n-i}^{i} P_0 \bigotimes_{i} P_1 \right) \cdot \left( \bigotimes_{n-j}^{j} P_0 \bigotimes_{j} P_1 \right) = 0$$
Proof: By induction:

Without loss of generality, assume $i < j$. For the case $n = 1$, this gives

\[
\left( \bigotimes_{1} P_0 \bigotimes P_1 \right) \cdot \left( \bigotimes_{0} P_0 \bigotimes P_1 \right) = P_0 P_1
\]

\[
= 0
\]

Let $S_m$ be the proposition statement for $n = m$ and assume $S_m$ to be true. Without loss of generality, for $S_{m+1}$ assume that $P_0$ is the extra matrix to be taken in the Kronecker product. Then

\[
\left( \bigotimes_{m+1-i} P_0 \bigotimes P_1 \right) \cdot \left( \bigotimes_{m+1-j} P_0 \bigotimes P_1 \right)
\]

\[
= \left[ P_0 \otimes \left( \bigotimes_{m-i} P_0 \bigotimes P_1 \right) \right] \cdot \left[ P_0 \otimes \left( \bigotimes_{m-j} P_0 \bigotimes P_1 \right) \right]
\]

\[
= (P_0 P_0) \otimes \left[ \left( \bigotimes_{m-i} P_0 \bigotimes P_1 \right) \cdot \left( \bigotimes_{m-j} P_0 \bigotimes P_1 \right) \right]
\]

\[
= (P_0 P_0) \otimes 0
\]

\[
= 0
\]

Thus, $S_m \Rightarrow S_{m+1}$ and the proposition holds.

Having established these properties of commutativity and orthogonality, we have all the pieces necessary for an orthogonal transform. We will now take our factorization method and look into the development of an algorithm to implement it.
4.1 Development for a $2 \times 2$ system

In the previous chapter, we discussed the properties of the Hadamard-based transform, many of which are necessary for the development of an accurate and efficient algorithm. The motivation for the algorithm comes from the wavelet structure in homogenization. As we have a similar structure in the discrete form, here too we will be finding a series of averages and differences, but in this case we will then use the column structure of Hadamard matrices to construct a solution. We will start by again looking at the $2 \times 2$ case, although many of the equations and solutions can be used in a similar manner in the block form.

Using the $\epsilon^0$ and $\epsilon^{-1}$ perturbation equations defined in Chapter 3, (the same equations used to define a solution in homogenization,) we will develop a relationship between the pieces of $y, y_0$ and $y_1$. The first and main equation we'll be working with is

$$v_0 = AP_0 y_0 + AP_1 y_1$$

We'll also be using

$$P_1 v_0 = r_1 \quad \text{and} \quad P_0 v_0 = r_0$$

Combining these three equations, $P_1$ and $P_0$ can be applied in turn to both sides of equation (4.1) to produce the following system of equations.

$$P_1 v_0 = r_1 = P_1 AP_0 y_0 + P_1 AP_1 y_1$$

$$P_0 v_0 = r_0 = P_0 AP_0 y_0 + P_0 AP_1 y_1$$

Akin to homogenization, we would like to rewrite $y_1$ in terms of $P_0 y_0$, i.e., we would like to find some $\Omega \in M^{2 \times 2}$ such that

$$y_1 = \Omega P_0 y_0$$

This would allow us to rewrite one of the equations in terms of only one vector unknown, $y_0$, so that a solution to the system may be found.
Before calculating $\Omega$ it is necessary to introduce the matrix $Q$, where

$$Q = \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{bmatrix}$$

Notice that $Q$ maps the column space of $P_1$ into the column space of $P_0$ and that $Q^T$ maps the column space of $P_0$ into the column space of $P_1$. Also,

$$P_0 = QQ^T \quad \text{and} \quad P_1 = Q^T Q$$

We will use these properties to compute $y_1$ in terms of $P_0y_0$.

First, equation (4.4) can be rewritten as

$$y_1 = \Omega QQ^T y_0$$

Expanding these matrices and vectors gives

$$\begin{bmatrix} y_{11} \\ -y_{11} \end{bmatrix} = \Omega \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} y_{01} \\ y_{01} \end{bmatrix}$$

$$= \Omega \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} y_{01} \\ -y_{01} \end{bmatrix}$$

$$= \Omega \begin{bmatrix} y_{01} \\ y_{01} \end{bmatrix}$$

In the second step, multiplying $y_0$ by $Q^T$ had the effect of negating the second entry of $y_0$. So if $\Omega$ contains $Q^T$, this implies that $y_{11} = cy_{01}$ for some constant $c$. Thus, we can write $\Omega$ as $\Omega = cQ^T$ and we have

$$y_1 = cQ^TP_0y_0 \quad \text{(4.5)}$$

Using equation (4.2) we can find a formula for $c$ by substituting for $y_1$.

$$r_1 = P_1AP_0y_0 + cP_1AP_1Q^TP_0y_0$$

$$= P_1AP_{0,0} + cP_1AQ^Ty_0$$

$$= \frac{1}{2}aQ^Ty_0 + \frac{1}{2}c\hat{a}Q^Ty_0$$

(by the identities in Appendix A.) Again, by expanding matrices and vectors we have

$$r_{11} \begin{bmatrix} 1 \\ -1 \end{bmatrix} = \frac{1}{2}\hat{a}y_{01} \begin{bmatrix} 1 \\ -1 \end{bmatrix} + \frac{1}{2}c\hat{a}y_{01} \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$
which gives
\[ r_{11} = \frac{1}{2} \hat{\alpha} y_{01} + \frac{1}{2} c \hat{\alpha} y_{01} \]

Solving for \( c \) we have
\[
(4.6) \quad c = \frac{\hat{\alpha}^{-1} (2r_{11} - \hat{\alpha} y_{01})}{y_{01}}
\]
or, equivalently,
\[
(4.7) \quad y_{11} = \hat{\alpha}^{-1} (2r_{11} - \hat{\alpha} y_{01})
\]

By computation, it can easily be verified that this rewritten formula for \( y_{11} \) produces the same solution as our previous formula.

Now we can use equation (4.3) to solve for \( y_0 \). Substituting for \( y_1 \), we have
\[
\begin{align*}
0 &= P_0 A P_0 + P_0 A P_1 \Omega D_0 y_0 \\
&= P_0 A (I + P_1 \Omega) D_0 y_0
\end{align*}
\]

If we define \( A^# = A (I + P_1 \Omega) \), then the above equation becomes
\[
(4.8) \quad r_0 = P_0 A^# D_0 y_0
\]

Before looking at this structure, one thing that should be mentioned is that multiplying both sides of a matrix \( A \) by \( P_0 \) results in a matrix whose entries are a multiple of the arithmetic average of the entries of \( A \). Thus, equation (4.8) can also be written as
\[
(4.9) \quad r_0 = \frac{1}{2} a^# P_0 y_0
\]

Again, this is similar to homogenization process. When the average is calculated in homogenization it is not the arithmetic average of the data entries, \( \bar{k} \), that is used, but the harmonic average,
\[
\bar{k}^# = \int_0^1 k(y) \, dy
\]

This average provides a better approximation to the solution. Thus, calculating our new average, \( a^# \), should produce a better approximation to the exact solution and eventually provide a more accurate and efficient algorithm.
For the purpose of developing an efficient algorithm, we can again use equation (4.3) to solve for $a^#$ and $y_0$ in the following manner.

\[ r_0 = P_0 A P_0 y_0 + P_0 A P_1 y_1 \]
\[ = \frac{1}{2} \bar{a} P_0 y_0 + \frac{1}{2} \bar{a} Q y_1 \]

Substituting for $y_1$ and expanding vectors and matrices results in

\[ 2r_{01} \begin{bmatrix} 1 \\ 1 \end{bmatrix} = \bar{a} y_{01} \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \bar{a} \bar{a}^{-1} (2r_{11} - \bar{a} y_{01}) \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & -\frac{1}{2} \end{bmatrix} \begin{bmatrix} 1 \\ -1 \end{bmatrix} \]

which gives

\[ 2r_{01} - 2\bar{a} \bar{a}^{-1} r_{11} = \bar{a} y_{01} - \bar{a} \bar{a}^{-1} \bar{a} y_{01} \]

Finally, this equation can be written as

(4.10)
\[ r^# = a^# y_{01} \]

where

(4.11)
\[ r^# = 2(r_{01} - \bar{a} \bar{a}^{-1} r_{11}) \]

and

(4.12)
\[ a^# = \bar{a} - \bar{a} \bar{a}^{-1} \bar{a} \]

Therefore, we can find $y_{01}$ by calculating $r^#/a^#$.

Now we can write a simple algorithm for the $2 \times 2$ case. As with any efficient algorithm, we would like to store as little information as possible while retaining any information that may need to be recalled at a later time and maintaining the accuracy of the solution. Although in the $2 \times 2$ solution we could perhaps store fewer values, we will treat the algorithm as if it is the building block for an algorithm to solve a larger matrix and retain all the information needed to reconstruct the original matrix at a later time.

4.2 Algorithm for the $2 \times 2$ case

Step 1. First, we must compute and store $\bar{a}$, $\bar{a}$, $\bar{a}$, and $\bar{a}$. This can be done by

\[
\begin{bmatrix}
1 & 1 & 1 & 1 \\
1 & -1 & 1 & -1 \\
1 & 1 & -1 & -1 \\
1 & -1 & -1 & 1
\end{bmatrix}
\begin{bmatrix}
a_1 \\
a_2 \\
a_3 \\
a_4
\end{bmatrix}
= 
\begin{bmatrix}
\bar{a} \\
\bar{a} \\
\bar{a} \\
\bar{a}
\end{bmatrix}
\]
Step 2. We must also compute and store $r_{01}$ and $r_{11}$, by
\[
\begin{bmatrix} r_{01} \\ r_{11} \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \end{bmatrix} = \begin{bmatrix} b_1 + b_2 \\ b_1 - b_2 \end{bmatrix}
\]

Step 3. Compute $a^\# = \tilde{a} - \tilde{a}a^{-1}\tilde{a}$ and store $a^\#$ in place of $\tilde{a}$.

Step 4. Compute $r^\# = 2(r_{01} - \tilde{a}a^{-1}r_{11})$ and store $r^\#$ in place of $r_{01}$.

Step 5. Compute and store $y_{01}$ using $y_{01} = r^\#/a^\#$.

Step 6. Compute and store $y_{11}$ by $y_{11} = \tilde{a}^{-1}(2r_{11} - \tilde{a}y_{01})$.

Step 7. Now, to begin to find the solution, compute and store
\[
y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = y_{01} \begin{bmatrix} 1 \\ 1 \end{bmatrix} + \epsilon y_{11} \begin{bmatrix} 1 \\ -1 \end{bmatrix}
\]
where $\epsilon = \frac{1}{2}$.

Step 8. Finally, compute the solution $x$ by
\[
x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = Hy
\]
\[
= \begin{bmatrix} 3 & -\frac{1}{2} \\ -\frac{3}{2} & 3 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}
\]

4.3 Extending the Algorithm to $2^n \times 2^n$

First, we'll look at the components of the algorithm in terms of the block structure. The only hindrance to performing the computations in the same manner as the $2 \times 2$ case comes in the formulation for $c$ used in equation (4.5). In this case, $Y_{01}$ is a vector, by which we cannot divide. However, we can bypass this step in formulating the algorithm as $Y_{11}$ can be determined without computing $c$. Therefore, with the exception of the first step, an algorithm for computing a block $2^n \times 2^n$ system can be written using the same process as above, with values

\[
A^\# = \bar{A} - \bar{A}\bar{A}^{-1}\bar{A}
\]
\[
R^\# = 2(R_{01} - \bar{A}\bar{A}^{-1}R_{11})
\]
\[
Y_{01} = R^\#(A^\#)^{-1}
\]
\[
Y_{11} = \bar{A}^{-1}(2R_{11} - \bar{A}Y_{01})
\]
and the matrix

\[
H_n = \begin{bmatrix}
\frac{3}{2}I_{2^{n-1}} & -\frac{1}{2}I_{2^{n-1}} \\
-\frac{1}{2}I_{2^{n-1}} & \frac{3}{2}I_{2^{n-1}}
\end{bmatrix}
\]

The first step needs to be rewritten as

\[
\begin{bmatrix}
I_{2^{n-1}} & I_{2^{n-1}} & I_{2^{n-1}} & I_{2^{n-1}} \\
I_{2^{n-1}} & -I_{2^{n-1}} & I_{2^{n-1}} & -I_{2^{n-1}} \\
I_{2^{n-1}} & I_{2^{n-1}} & -I_{2^{n-1}} & -I_{2^{n-1}} \\
I_{2^{n-1}} & -I_{2^{n-1}} & -I_{2^{n-1}} & I_{2^{n-1}}
\end{bmatrix}
\begin{bmatrix}
A_{11} \\
A_{12} \\
A_{21} \\
A_{22}
\end{bmatrix}
\]

by which \( \tilde{A}, \tilde{\tilde{A}}, \tilde{\tilde{A}}, \text{ and } \tilde{\tilde{A}} \) can be computed.

Of course, this block form will not produce the exact solution of a \( 2^n \times 2^n \) system. To produce the exact solution, we need a recursive algorithm that nests the simple \( 2 \times 2 \) algorithm within it. This would act much like an algorithm for a fast Fourier transform, where in this case the finer scales are first computed and the resulting information is used in the following iterations.

The final solution for a system of size \( 2^n \times 2^n \) should be of the form

\[
y_{01} \cdot \text{colsp}[\bigotimes_{n} F_0] + y_{11} \cdot \text{colsp}[\bigotimes_{n-1} F_0 \bigotimes P_1] + \ldots + y_{n1} \cdot \text{colsp}[\bigotimes_{n} P_1]
\]

As there are many approaches to developing this algorithm and one would want to know the various error estimates and efficiency of each approach, this will be left to a future study. The basic building block, however, should be the algorithm set out above.
In this paper, a method for solving a system of linear equations using a wavelet factorization was presented. The idea was to apply the structure and properties present in Hadamard matrices and incorporate the steps and reasoning of the homogenization process to guide us to a viable solution. The method was motivated by the average and difference structure found throughout homogenization and the recognized ability to rewrite that structure in terms of a linear system. This paper detailed the development of an algorithm to apply the method to a simple $2 \times 2$ system and to use as an iterative building block for future algorithms, but has left the development and exploration of such algorithms to future work. Future study may also be focused on integrating this method of factorization of linear systems into the process of homogenization in solving differential equations.
REFERENCES


Throughout this paper, various identities have been referred to in a number of calculations. The following is a summary of these and other related identities.

Let $A$ be an arbitrary $2 \times 2$ matrix, and let $\bar{a}$, $\tilde{a}$, $\hat{a}$, and $\check{a}$ be defined as

\[ \bar{a} = a_{11} + a_{12} + a_{21} + a_{22} \]
\[ \tilde{a} = a_{11} - a_{12} + a_{21} - a_{22} \]
\[ \hat{a} = a_{11} + a_{12} - a_{21} - a_{22} \]
\[ \check{a} = a_{11} - a_{12} - a_{21} + a_{22} \]

Also, we define the $2 \times 2$ projection matrices $P_0$, $P_1$, $Q$, and $Q^T$ as

\[ P_0 = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ \frac{1}{2} & \frac{1}{2} \end{bmatrix}, \quad P_1 = \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} \end{bmatrix}, \quad Q = \begin{bmatrix} \frac{1}{2} & -\frac{1}{2} \\ \frac{1}{2} & -\frac{1}{2} \end{bmatrix}, \quad Q^T = \begin{bmatrix} \frac{1}{2} & \frac{1}{2} \\ -\frac{1}{2} & -\frac{1}{2} \end{bmatrix}. \]

The products of these matrices with each other as well as applied in left- and right-multiplication with an arbitrary $2 \times 2$ matrix $A$ are shown in tables A.1 and A.2, respectively.

Now, for

\[ H_n = \begin{bmatrix} \frac{3}{2}I_{2^{n-1}} & -\frac{1}{2}I_{2^{n-1}} \\ -\frac{1}{2}I_{2^{n-1}} & \frac{3}{2}I_{2^{n-1}} \end{bmatrix} \]

and $D_{k,l}$ defined as the $2^{k+l} \times 2^{k+l}$ block diagonal matrix consisting of $2^l$ blocks of $H_k$, the Kronecker products of $H_n$ and $D_{k,l}$ with the identity matrix $I_{2^n}$ are defined in table A.3.
Table A.1. Identities involving the multiplication of two permutation matrices, $LR$.

<table>
<thead>
<tr>
<th>$L \setminus R$</th>
<th>$P_0$</th>
<th>$P_1$</th>
<th>$Q$</th>
<th>$Q^T$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_0$</td>
<td>$P_0$</td>
<td>0</td>
<td>$Q$</td>
<td>0</td>
</tr>
<tr>
<td>$P_1$</td>
<td>0</td>
<td>$P_1$</td>
<td>$Q^T$</td>
<td>0</td>
</tr>
<tr>
<td>$Q$</td>
<td>0</td>
<td>$Q^T$</td>
<td>0</td>
<td>$P_1$</td>
</tr>
<tr>
<td>$Q^T$</td>
<td>$Q^T$</td>
<td>0</td>
<td>$P_0$</td>
<td>0</td>
</tr>
</tbody>
</table>

Table A.2. Identities involving the multiplication of a matrix $A$ by two permutation matrices, $LAR$.

<table>
<thead>
<tr>
<th>$L \setminus R$</th>
<th>$P_0$</th>
<th>$P_1$</th>
<th>$Q$</th>
<th>$Q^T$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P_0$</td>
<td>$\frac{1}{2}\bar{a}P_0$</td>
<td>$\frac{1}{2}\bar{a}Q$</td>
<td>$\frac{1}{2}\bar{a}Q$</td>
<td>$\frac{1}{2}\bar{a}P_0$</td>
</tr>
<tr>
<td>$P_1$</td>
<td>$\frac{1}{2}\bar{a}Q^T$</td>
<td>$\frac{1}{2}\bar{a}P_1$</td>
<td>$\frac{1}{2}\bar{a}P_1$</td>
<td>$\frac{1}{2}\bar{a}Q^T$</td>
</tr>
<tr>
<td>$Q$</td>
<td>$\frac{1}{2}\bar{a}P_0$</td>
<td>$\frac{1}{2}\bar{a}Q$</td>
<td>$\frac{1}{2}\bar{a}Q$</td>
<td>$\frac{1}{2}\bar{a}P_0$</td>
</tr>
<tr>
<td>$Q^T$</td>
<td>$\frac{1}{2}\bar{a}Q^T$</td>
<td>$\frac{1}{2}\bar{a}P_1$</td>
<td>$\frac{1}{2}\bar{a}P_1$</td>
<td>$\frac{1}{2}\bar{a}Q^T$</td>
</tr>
</tbody>
</table>

Table A.3. Identities involving the Kronecker product of the pieces of $T_n$, $L \otimes R$.

<table>
<thead>
<tr>
<th>$L \setminus R$</th>
<th>$I_{2i}$</th>
<th>$H_n$</th>
<th>$D_{k,i}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$I_{2i}$</td>
<td>-</td>
<td>$D_{n,i}$</td>
<td>$D_{k,l+i}$</td>
</tr>
<tr>
<td>$H_n$</td>
<td>$H_{i+n}$</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>$D_{k,l}$</td>
<td>$D_{l+k,l}$</td>
<td>-</td>
<td>-</td>
</tr>
</tbody>
</table>