A Fast Near- to Far-Field Transform Algorithm

A Thesis

Presented in Partial Fulfillment of the Requirements for
Graduation with Distinction from the
College of Engineering of The Ohio State University

By

Matthew B. Stephanson

* * * * *

The Ohio State University

2007

Distinction Committee:

Professor Jin-Fa Lee, Adviser
Professor Fernando Teixeira

Approved by

Adviser
Department of Electrical &
Computer Engineering

ABSTRACT

A frequent problem in computational electromagnetics is to calculate the electromagnetic field far from a radiating object. This has numerous applications, such as antenna analysis and determining the radar cross sections of airplanes, ships, etc. Another application to converting near field measurements, taken in an indoor measurement range for instance, and transforming them to the far-zone. However, when the field of a large object needs to be calculated in many directions, however, this computation can take too long to be practical. The goal of this project is to develop an algorithm to greatly speed up this calculation by using a divide-and-conquer approach. First, the necessary background information in electromagnetics and numerical methods is presented. Then, a near to far field transform with potential for $O(N^{1.5})$ time complexity is discussed, where N is the number of basis elements used to approximate the object's surface. Finally, the running time and accuracy of the algorithm is discussed, with comparisons to another near to far field computer program for two problem geometries, a VFY-218 airplane with $N = 53,054$ and an antenna array with $N = 106,792$. The results show the fast near to far transform can be 30 times faster with an RMS relative error of 2.05×10^{-5}, compared to traditional methods.

ACKNOWLEDGMENTS

Foremost, I must thank my adviser, Professor Jin-Fa Lee, for his patience and guidance as I slowly worked out the algorithm. Without his trust, I would never have thought about electromagnetics research.

I also wish to thank the Computational Science Group at the Electroscience Lab, especially Seung-Choel Lee, Vineet Rawat, Seung Mo Seo, and Kezhong Zhao, for providing source code to work from, data files, and advice for porting the source code.

TABLE OF CONTENTS

Chapters:

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

ELECTROMAGNETICS AND NUMERICAL METHODS BACKGROUND

This chapter present a derivation of the electromagnetic equations upon which the transform algorithm is based, as well as an overview of relevant numerical methods topics.

1.1 The Maxwell Equations and Potential Functions

We begin with the symmetric Maxwell equations in linear media. If an $e^{j\omega t}$ time dependence is assumed and suppressed, these can be written as

$$\begin{aligned} \boldsymbol{\nabla}\cdot\epsilon\mathbf{E} &= \rho_e & -\boldsymbol{\nabla}\times\mathbf{E} &= j\omega\mu\mathbf{H}+\mathbf{J_m} \\ \boldsymbol{\nabla}\cdot\mu\mathbf{H} &= \rho_m & \boldsymbol{\nabla}\times\mathbf{H} &= j\omega\epsilon\mathbf{E}+\mathbf{J_e}, \end{aligned} \tag{1.1}$$

where ρ_e and $\mathbf{J_e}$ are the electric charge and current, while ρ_m and $\mathbf{J_m}$ are the magnetic charge and current. With the assumption of linearity, electric and magnetic sources can be considered independently. Then one of the divergence equations in (1.1) will be homogeneous, allowing either $\mathbf{E}$ or $\mathbf{H}$ to be written as the curl of a vector field. With no magnetic sources, this vector field is denoted $\mathbf{A}$ and called the magnetic vector potential. With no electric sources, it is given the symbol $\mathbf{F}$ and called the electric vector potential [1, pp. 135–138]. Thus,

$$\mathbf{H_A} = \frac{1}{\mu}\,\boldsymbol{\nabla}\times\mathbf{A} \tag{1.2a}$$

$$\mathbf{E_F} = -\frac{1}{\epsilon}\,\boldsymbol{\nabla}\times\mathbf{F}, \tag{1.2b}$$

where $\mathbf{H_A}$ is the magnetic field due only to electric sources, and $\mathbf{E_F}$ is the electric field due only to magnetic sources. Corresponding definitions apply for $\mathbf{H_F}$ and $\mathbf{E_A}$. Then the Maxwell equations can be divided into two groups, one for electric sources and another for magnetic sources.

$$\boldsymbol{\nabla}\cdot\epsilon\mathbf{E_A} = \rho_e \qquad \boldsymbol{\nabla}\cdot\epsilon\mathbf{E_F} = 0 \tag{1.3a}$$

$$\boldsymbol{\nabla}\cdot\mu\mathbf{H_A} = 0 \qquad \boldsymbol{\nabla}\cdot\mu\mathbf{H_F} = \rho_m \tag{1.3b}$$

$$-\boldsymbol{\nabla}\times\mathbf{E_A} - j\omega\mu\mathbf{H_A} = 0 \qquad -\boldsymbol{\nabla}\times\mathbf{E_F} - j\omega\mu\mathbf{H_F} = \mathbf{J_m} \tag{1.3c}$$

$$\boldsymbol{\nabla}\times\mathbf{H_A} - j\omega\epsilon\mathbf{E_A} = \mathbf{J_e} \qquad \boldsymbol{\nabla}\times\mathbf{H_F} - j\omega\epsilon\mathbf{E_F} = 0 \tag{1.3d}$$

If (1.2) is substituted into the homogeneous curl equations of (1.3), the following relations result:

$$\boldsymbol{\nabla}\times\begin{Bmatrix}\mathbf{E_A} + j\omega\mathbf{A}\\ \mathbf{H_F} + j\omega\mathbf{F}\end{Bmatrix} = 0. \tag{1.4}$$

An irrotational field may be written as the gradient of a scalar field, so (1.4) becomes

$$\begin{Bmatrix}\mathbf{E_A}\\ \mathbf{H_F}\end{Bmatrix} = -\boldsymbol{\nabla}\begin{Bmatrix}\Phi\\ \Psi\end{Bmatrix} - j\omega\begin{Bmatrix}\mathbf{A}\\ \mathbf{F}\end{Bmatrix}. \tag{1.5}$$

The scaler fields are known as the scaler electric and magnetic potentials, Φ and Ψ.

To obtain expressions for the vector potentials, (1.2) is substituted into the inhomogeneous curl equations of (1.3). This yields

$$(\nabla^2 + k^2)\begin{Bmatrix}\mathbf{A}\\ \mathbf{F}\end{Bmatrix} + \boldsymbol{\nabla}\left(\boldsymbol{\nabla}\cdot\begin{Bmatrix}\mathbf{A}\\ \mathbf{F}\end{Bmatrix} + \frac{j\omega}{u^2}\begin{Bmatrix}\Phi\\ \Psi\end{Bmatrix}\right) = -\begin{Bmatrix}\mu\mathbf{J_e}\\ \epsilon\mathbf{J_m}\end{Bmatrix}, \tag{1.6}$$

where $k = \omega/u$ is the wavenumber and u is the speed of propagation, $(\epsilon\mu)^{-1/2}$. These equations still contain both the scaler and vector potentials. To eliminate one, use can be made of the fact that the electric and magnetic fields are the physically real quantities, not the potentials, a fact formally known as gauge invariance [2, pp. 240–242]. A common choice is the Lorenz gauge, which is defined by the Lorenz condition:

$$\boldsymbol{\nabla}\cdot\left\{\begin{matrix}\mathbf{A}\\ \mathbf{F}\end{matrix}\right\} + \frac{j\omega}{u^2}\left\{\begin{matrix}\Phi\\ \Psi\end{matrix}\right\} = 0. \tag{1.7}$$

Then the vector potentials satisfy the Helmholtz equation

$$(\nabla^2 + k^2)\left\{\begin{matrix}\mathbf{A}\\ \mathbf{F}\end{matrix}\right\} = -\left\{\begin{matrix}\mu\mathbf{J_e}\\ \epsilon\mathbf{J_m}\end{matrix}\right\}. \tag{1.8}$$

By using (1.5) and(1.7) in the inhomogeneous divergence equations of (1.3), it is found that the scaler potentials satisfy similar equations,

$$(\nabla^2 + k^2)\left\{\begin{matrix}\Phi\\ \Psi\end{matrix}\right\} = -\left\{\begin{matrix}\rho_e/\epsilon\\ \rho_m/\mu\end{matrix}\right\}. \tag{1.9}$$

1.2 Far-Zone Solution of the Helmholtz Equation

With no boundary surfaces and assuming an outwardly-radiating solution, the Helmholtz equation $(\nabla^2 + k^2)f = -g$ has the solution

$$f(\mathbf{x}) = \frac{1}{4\pi}\int \frac{e^{-jk|\mathbf{x}-\mathbf{x}'|}}{|\mathbf{x}-\mathbf{x}'|} g(\mathbf{x}')\, d^3x'. \tag{1.10}$$

In the far zone ($|\mathbf{x}| \gg \max|\mathbf{x}'|$), the inverse distance term may be approximated by $r = |\mathbf{x}|$. The oscillatory nature of the complex exponential requires more care with respect to the phase term, however. To this end, it will be expanded using the binomial theorem. First,

$$|\mathbf{x}-\mathbf{x}'| = (|\mathbf{x}|^2 + |\mathbf{x}'|^2 - 2|\mathbf{x}||\mathbf{x}'|\cos\gamma)^{1/2}, \tag{1.11}$$

where γ is the angle between $\mathbf{x}$ and $\mathbf{x}'$. Then

$$\begin{aligned}|\mathbf{x}-\mathbf{x}'| &= |\mathbf{x}| - \frac{|\mathbf{x}'|}{2|\mathbf{x}|}(2|\mathbf{x}|\cos\gamma - |\mathbf{x}'|) - \frac{|\mathbf{x}'|^2}{8|\mathbf{x}|^3}(2|\mathbf{x}|\cos\gamma - |\mathbf{x}'|)^2 + \cdots \\ &\approx |\mathbf{x}| - |\mathbf{x}'|\cos\gamma \\ &= r - \hat{\mathbf{r}}\cdot\mathbf{x}'\end{aligned} \tag{1.12}$$

where $\hat{\mathbf{r}} = \mathbf{x}/r$. With these two simplifications, (1.10) may be written as

$$f(\mathbf{x}) = \frac{e^{-jkr}}{4\pi r}\int g(\mathbf{x}')e^{jk\hat{\mathbf{r}}\cdot\mathbf{x}'}\,d^3x'. \tag{1.13}$$

The above equation has an important result when applied to the scaler and vector potentials. First, note that the vector potential has the form

$$\mathbf{A} = \frac{e^{-jkr}}{4\pi r}\left[\hat{\mathbf{r}}A_r(\theta,\phi) + \hat{\boldsymbol{\theta}}A_\theta(\theta,\phi) + \hat{\boldsymbol{\phi}}A_\phi(\theta,\phi)\right].$$

Taking the curl of this expression, it is seen that

$$\begin{aligned}\boldsymbol{\nabla}\times\mathbf{A} = \frac{e^{-jkr}}{4\pi}&\left(\frac{\hat{\mathbf{r}}}{r^2\sin\theta}\left[\frac{\partial}{\partial\theta}(A_\theta\sin\theta) - \frac{\partial A_\theta}{\partial\phi}\right] + \frac{\hat{\boldsymbol{\theta}}}{r}\left[\frac{1}{r\sin\theta}\frac{\partial A_r}{\partial\phi} + jkA_\phi\right]\right. \\ &\left. -\frac{\hat{\boldsymbol{\phi}}}{r}\left[jkA_\theta + \frac{1}{r}\frac{\partial A_r}{\partial\theta}\right]\right).\end{aligned} \tag{1.14}$$

For large r, the four r^{-2} terms can be discarded in favor of the two r^{-1} terms. Therefore (1.2) may be re-written as

$$\begin{Bmatrix}\mu\mathbf{H_A}\\ \epsilon\mathbf{E_F}\end{Bmatrix} = \boldsymbol{\nabla}\times\begin{Bmatrix}\mathbf{A}\\ -\mathbf{F}\end{Bmatrix} \approx jk\begin{Bmatrix}\mathbf{A}\\ -\mathbf{F}\end{Bmatrix}\times\hat{\mathbf{r}}. \tag{1.15}$$

This equality shows that the far zone $\mathbf{H_A}$ and $\mathbf{E_F}$ fields decay as r^{-1}, have no radial component, and may be obtained by a simple cross product between the vector potentials and the radial vector.

Turning to the electric scaler potential, note that by using the Lorenz condition the contribution to $\mathbf{E_A}$ can be written as

$$\begin{aligned}-\boldsymbol{\nabla}\Phi &= \frac{\boldsymbol{\nabla}(\boldsymbol{\nabla}\cdot\mathbf{A})}{j\omega\mu\epsilon} \\ &= \boldsymbol{\nabla}\left[\mathbf{L}(\theta,\phi)\cdot\boldsymbol{\nabla}\frac{e^{-jkr}}{4\pi j\omega\epsilon r} + \frac{e^{-jkr}}{4\pi j\omega\epsilon r}\,\boldsymbol{\nabla}\cdot\mathbf{L}(\theta,\phi)\right],\end{aligned} \tag{1.16}$$

where $\mathbf{L}(\theta,\phi) = \int \exp(jk\hat{\mathbf{r}}\cdot\mathbf{x}')\mathbf{J_e}(\mathbf{x}')\,d^3x'$. The second term in the square brackets decays as r^{-2}, so its gradient has r^{-3} terms and can therefore be ignored. The first term expands to

$$\begin{aligned} \boldsymbol{\nabla}\left[\mathbf{L}(\theta,\phi)\cdot\boldsymbol{\nabla}\frac{e^{-jkr}}{4\pi j\omega\epsilon r}\right] &= -\boldsymbol{\nabla}\left[L_r\frac{e^{-jkr}}{4\pi\omega\epsilon}\left(\frac{k}{r}+\frac{1}{jr^2}\right)\right] \\ &= \hat{\mathbf{r}}j\omega\mu\frac{e^{-jkr}}{4\pi r}L_r + \frac{1}{r^2}(\cdots) + \frac{1}{r^3}(\cdots). \end{aligned} \tag{1.17}$$

The leading r^{-1} term is simply the radial component of $j\omega\mathbf{A}$ and the remaining higher-order terms are negligible in the far zone. Thus, in the equation $\mathbf{E_A} = -\boldsymbol{\nabla}\Phi - j\omega\mathbf{A}$, we find that the transverse components of the first term are negligible and the radial components of both terms cancel, leaving

$$\mathbf{E_A} = -j\omega(A_\theta\hat{\boldsymbol{\theta}} + A_\phi\hat{\boldsymbol{\phi}}). \tag{1.18}$$

Combined with the expression for $\mathbf{E_F}$ given in (1.15), the far-zone electric field is written

$$\mathbf{E_{far}} = \mathbf{E_F} + \mathbf{E_A} = -\frac{jk}{\epsilon}\mathbf{F_{far}}\times\hat{\mathbf{r}} - j\omega\mathbf{A_{far}}, \tag{1.19}$$

with the understanding that the radial contributions from $\mathbf{A}$ should be ignored and the **far** subscript indicating that the far field approximations have been used.

We now have all the results necessary to write the far-zone electric field explicitly in terms of the electric and magnetic sources. Using the far field solution to the Helmholtz Equation given in (1.13), the vector potentials are

$$\begin{Bmatrix}\mathbf{A_{far}}\\ \mathbf{F_{far}}\end{Bmatrix} = \begin{Bmatrix}\mu\\ \epsilon\end{Bmatrix}\frac{e^{-jkr}}{4\pi r}\int\begin{Bmatrix}\mathbf{J_e}\\ \mathbf{J_m}\end{Bmatrix}e^{jk\hat{\mathbf{r}}\cdot\mathbf{x}'}\,d^3x'. \tag{1.20}$$

Combined with (1.19), the electric field is then

$$\begin{aligned} \mathbf{E_{far}}(\mathbf{x}) &= -\frac{e^{-jkr}}{4\pi r}\int(jk\mathbf{J_m}\times\hat{\mathbf{r}} + j\omega\mu\mathbf{J_e})e^{jk\hat{\mathbf{r}}\cdot\mathbf{x}'}\,d^3x' \\ &= -jk\frac{e^{-jkr}}{4\pi r}\int(\mathbf{J_m}\times\hat{\mathbf{r}} + \eta\mathbf{J_e})e^{jk\hat{\mathbf{r}}\cdot\mathbf{x}'}\,d^3x', \end{aligned} \tag{1.21}$$

where $\eta = \sqrt{\mu/\epsilon}$ is the wave impedance.

1.3 Surface Currents and the Equivalence Theorem

In the preceding section, the far field potentials were determined from volume current densities. This solution can be generalized naturally to surface and line currents be replacing the integration domain with a lower-dimensional region. There are some cases where surface currents are directly applicable. For example, if the object under study is a perfect electric conductor (PEC), then electric currents will exist only on the object's surface. This is a desirable situation, since surface and line integrals tend to be easier than volume integrals.

For non-PEC objects, a similar result can be obtained with the equivalence theorem [1, pp. 653–657], which is a formal statement of Huygens' Principle. The theorem states that the fields at the boundary of a surface that completely encloses all sources are sufficient to determine the field everywhere outside the surface. Since the theorem says nothing of the field inside the surface, we are free to define it as desired. A common choice, known as the Love's Equivalent, is $\mathbf{E} = \mathbf{H} = \mathbf{0}$. Then the electromagnetic boundary conditions give

$$-\hat{\mathbf{n}} \times \mathbf{E} = \mathbf{J_{ms}} \qquad \text{and} \qquad \hat{\mathbf{n}} \times \mathbf{H} = \mathbf{J_{es}}, \tag{1.22}$$

where $\hat{\mathbf{n}}$ is the outward unit normal to the surface; the electric and magnetic fields are taken infinitesimally outside the surface; and the additional subscript s on the currents indicates surface currents. This result shows that calculation of the far field from surface currents can also be viewed as a near-to-far field transformation. One must simply choose a closed surface S that completely encloses the sources, use (1.22) to transform the fields on S to equivalent surface currents, and use (1.21), which is written as

$$\mathbf{E}_{\mathbf{far}}(\mathbf{x}) = -jk\frac{e^{-jkr}}{4\pi r}\int_S (\mathbf{J}_{\mathbf{ms}} \times \hat{\mathbf{r}} + \eta \mathbf{J}_{\mathbf{es}}) e^{jk\hat{\mathbf{r}}\cdot\mathbf{x}'}\, d^2x' \tag{1.23}$$

when generalized to a two-dimensional domain. Typically S corresponds with the surface of the object being studied (an antenna, aircraft, ship, etc.), but this is not strictly necessary. As a result, the equivalence theorem and (1.23) can be used to find the radiated field from any object, even when good conductivity cannot be assumed *a priori*.

1.4 Numerical Methods

1.4.1 Discretization

Consider the surface integral

$$\int_S f(\mathbf{x})\, d^2x.$$

In general, the function f cannot be represented in closed form. Furthermore, even if it could, it may be impractical to do so for the purposes of computer calculations. In addition, it may be impossible or impractical to represent the domain S exactly. The solution to both of these problems lies in discretization: S is approximated by a patchwork of many simple shapes, called elements, and f is approximated by a set of localized, linearly independent functions with support only on or in the neighborhood of a single element [2, pp. 79–82]. This thesis will focus on triangle elements and linear basis functions, but many other choices are available [3, pp. 40–52].

The prototypical triangle element e is shown in Figure 1.1. An arbitrary point $\mathbf{x}$ in the triangle can be described by the areas of the sub-triangles formed by drawing a line segment from $\mathbf{x}$ to two of the triangle's vertices. These areas, Δ_1, Δ_2, and Δ_3,

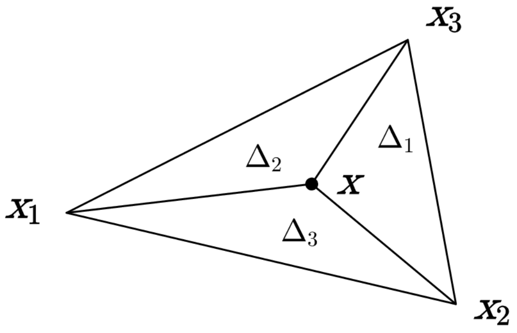

Figure 1.1: Prototypical triangle element

are called homogeneous barycentric coordinates [4]. If the Δ_i's are normalized by the area of the triangle, they are called areal coordinates:

$$\lambda_i = \frac{\Delta_i}{\Delta^{(e)}}. \tag{1.24}$$

Then the coordinates of $\mathbf{x}$ can be written as

$$\mathbf{x} = \lambda_1 \mathbf{x_1^{(e)}} + \lambda_2 \mathbf{x_2^{(e)}} + \lambda_3 \mathbf{x_3^{(e)}}.$$

Clearly, $\lambda_1 + \lambda_2 + \lambda_3 = 1$, so only two coordinates need to be given to uniquely specify a point, such as $\eta = \lambda_1$ and $\xi = \lambda_2$. Based on the areal coordinates, we define the basis functions

$$N_i^{(e)} = \lambda_i, \tag{1.25}$$

or, more explicitly,

$$\begin{aligned} N_1^{(e)}(\eta, \xi) &= \eta \\ N_2^{(e)}(\eta, \xi) &= \xi \\ N_3^{(e)}(\eta, \xi) &= 1 - \eta - \xi. \end{aligned}$$

These functions satisfy the equality

$$N_i^{(e)}(\mathbf{x_j}) = \delta_{ij}. \tag{1.26}$$

In other words, $N_i^{(e)}$ is equal to one at vertex i and decreases linearly to zero at the other two vertices. The function f over region S is now approximated in terms of these basis functions, so that

$$f \approx \sum_e \sum_{j=1}^{3} f_j^{(e)} N_j^{(e)}. \tag{1.27}$$

The coefficients $f_i^{(e)}$ may be thought of as the value of f at the i-th vertex of triangle e.

1.4.2 Numerical Integration

In order to integrate the approximation to f given in (1.27), one could determine the expressions for the λ_i's and integrate them analytically. There are several reasons why this may not be desirable. First, a new formula would have to be derived for each type of basis function, or a closed-form expression for the integral may not exist. More importantly, the desired integrand may be f times a weighting function, so a closed form expression for $\int_e f\,dS$ would not be useful.

For these reasons, a more fruitful approach is to approximate the desired integral numerically. Several algorithms are based on taking a weighed sum of function samples [5]. In algorithms such as Gaussian quadrature, these samples are taken at specified points in a normalized interval, for example,

$$\int_{-1}^{1} f(x)\,dx \approx \sum_{i=1}^{p} w_i f(x_i), \tag{1.28}$$

where p is the number of samples. Choosing the w_i and x_i is an art unto itself, but there are two essential features to notice in the context of this thesis. First, this method can be easily generalized to a two-dimensional triangular domain using areal coordinates (η_i, ξ_i) instead of x_i. Then (1.28) can be generalized to two dimensions and combined with (1.27) to yield

$$\int_e f(\mathbf{x})\, dS \approx \sum_{i=1}^{p} \sum_{j=1}^{3} w_i f_j^{(e)} N_j^{(e)}(\eta_i, \xi_i). \tag{1.29}$$

This brings us to the second essential feature: an integral over a single triangular element can be performed in constant computer time, since the summation limits in (1.29) are constant, regardless of the number of elements used to approximated the domain S. Furthermore, and integral over all of S requires a computational time that is proportional to N, the number of elements used to approximate S.

1.4.3 Big O Notation

As we have just seen, the topic of complexity frequently arises when discussing algorithms. We want to know how the running time or memory usage of a program changes as the problem size changes. Asymptotic notation is typically used for this purpose, especially "big O" notation $O(g(n))$ [6].

Definition. A function $f(n)$ is $O(g(n))$ as $n \to \infty$ iff $\exists n_0 > 0$, $\exists C > 0 : |f(n)| \leq C|g(n)|\ \forall n > n_0$.

Notations exist for other types of asymptotic behaviour, but O will suffice for our purposes. Using this notation, Section 1.4.2 showed that the number of operations to numerically integrate over a single triangular element is $O(1)$ and integrating over S is $O(N)$.

CHAPTER 2

NUMERICAL CALCULATION OF FAR ZONE FIELDS

2.1 Direct Summation

With the background developed in Chapter 1, we are ready to rewrite the far zone electric field equation,

$$\mathbf{E_{far}}(\mathbf{x}) = -jk\frac{e^{-jkr}}{4\pi r}\int_S (\mathbf{J_{ms}} \times \hat{\mathbf{n}} + \eta \mathbf{J_{es}}) e^{jk\hat{\mathbf{n}}\cdot\mathbf{x}'}\, d^2x',$$

in a form suitable for computer evaluation. By replacing the currents with basis function expansions and the integral with a weighted sum, we obtain

$$\mathbf{E_{far}}(\mathbf{x}) \approx -jk\frac{e^{-jkr}}{4\pi r}\sum_{(e)=1}^{N}\sum_{i=1}^{p}\sum_{j=1}^{3} w_i N_j^{(e)}(\eta_i, \xi_i)\left[\mathbf{J^{(e)}_{ms,j}} \times \hat{\mathbf{r}} + \eta \mathbf{J^{(e)}_{es,j}}\right] e^{jk\hat{\mathbf{r}}\cdot\mathbf{x}'}, \tag{2.1}$$

where $\mathbf{x}' = \eta_i \mathbf{x^{(e)}_1} + \xi_i \mathbf{x^{(e)}_2} + (1 - \eta_i - \xi_i)\mathbf{x^{(e)}_3}$; and $\mathbf{J^{(e)}_{ms,j}}$ and $\mathbf{J^{(e)}_{es,j}}$ are constant vectors that may be thought of as the value of the currents at vertex j of triangle e. Since the exact form of this function's r dependence is known, sampling the function consists only of choosing a direction $\hat{\mathbf{r}}$ and calculating everything in the summations.

Now that we know how to calculate the far field in a single direction, a natural question to ask is how many samples are needed to characterize the far field. Too few samples and important features of the field will be missed, but oversampling wastes time and computational resources. In [7], Coifman, Rokhlin, and Wandzura suggest that an appropriate sampling rate is L theta direction and $2L$ phi directions, for a total of $2L^2$ samples, where $L = O(kd)$ and d is twice the maximum value of $|\mathbf{x}'|$. Specifically,

$$L = \begin{cases} kd + 5\ln(kd+\pi) & \text{for single precision} \\ kd + 10\ln(kd+\pi) & \text{for double precision.} \end{cases} \tag{2.2}$$

These expression are intended for interpolation via a spherical harmonic series and Gaussian quadrature along the theta direction, but this project has so far found good accuracy using equispaced theta and phi directions with

$$\Delta\theta = \frac{2\pi}{2L+1} \quad \text{and} \quad \Delta\phi = \frac{2\pi}{2L},$$

which gives $2L(L+1)$ samples in total. If the size of the elements used to approximate the object's surface S stays approximately the same size, then the surface area of S is $O(N)$. Since the area is also proportional to d^2, it is clear that $2L(L+1) = O(k^2d^2) = O(N)$. Thus, $O(N)$ samples are needed, and each sample takes $O(N)$ time, so the total sampling time is $O(N^2)$.

As will be shown in Chapter 3, this time can rise to several days on a personal computer for $N \sim 10^5$. Reducing this time could have a significant impact to engineers working in such fields as antenna design and radar cross section optimization.

2.2 The Fast Far Field Transform

The fast algorithm is a divide and conquer approach. Instead of calculating the far field of the entire object, it is partitioned into many groups. The fields from the groups are calculated independently and then added together to obtain the complete far field.

Suppose, for example, that $L = 200$ for the entire object, so that 80,400 samples are needed. Naive divide-and-conquer might calculate 80,400 samples for each group and add them together, but this would not result in any savings. The key idea is that since the groups are smaller, many fewer samples are needed to characterize their fields. In our example, we might make the groups small enough that their average L is around 50, requiring sampling in only 5,100 directions. These samples can then be interpolated to the 80,400 directions needed for the entire object. Since interpolation is much faster than calculating the additional 75,300 samples by direct summation, a significant amount of time may be saved.

To be specific, suppose that the object is partitioned into M similarly-sized groups, so that each group has $\sim N/M$ elements. The previous section showed that the total time to calculate the far field of N elements is $O(N^2)$, so determining the field from one group is $O(N^2/M^2)$. Assuming a method can be found so that each interpolation costs $O(1)$ time, the time required to interpolate to the higher sampling rate will be $O(N)$, since $O(N - N/M)$ interpolations must be performed, and the number of groups M is assumed to be an increasing function of N, or at least constant. For all of the groups, then, the total time is $O(N^2/M + NM)$. This may be minimized by choosing $M = \sqrt{N}$, yielding $O(N^{1.5})$ time. A basic flowchart of this algorithm is shown in Figure 2.1. There are several fine points, however, that are not included on this flowchart; a complete description is given in Section 2.5

One of the first subtleties is as follows: note that d was defined as twice the maximum value of $|\mathbf{x}'|$, that is, the diameter of the smallest sphere centered on the origin and enclosing the sources. Thus, the number of samples needed for a group will not appreciably decrease if the origin remains near the center of the object. Instead, it

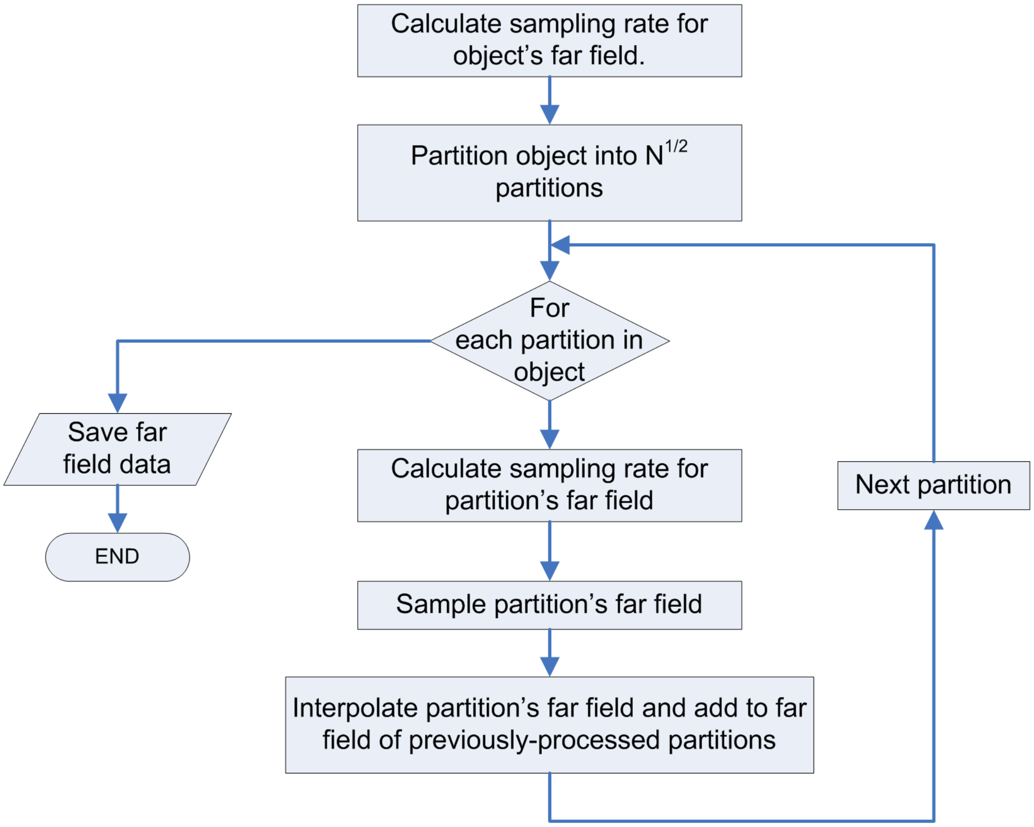
Calculate sampling rate for object's far field.
Partition object into N^1/2 partitions
For each partition in object
Save far field data
END
Calculate sampling rate for partition's far field
Next partition
Sample partition's far field
Interpolate partition's far field and add to far field of previously-processed partitions

Figure 2.1: Simplified algorithm for fast near to far field transform

should be moved to the center of the group, in order to minimize d. The exponential in the summation, however, will introduce a phase shift because of the change in origin. Specifically, if $\mathbf{O_g}$ is the vector from the object origin to the group origin, then

$$\begin{aligned} e^{jk\hat{\mathbf{r}}\cdot\mathbf{x}'} &= e^{jk\hat{\mathbf{r}}\cdot(\mathbf{x}'-\mathbf{O_g}+\mathbf{O_g})} \\ &= e^{jk\hat{\mathbf{r}}\cdot\mathbf{O_g}} e^{jk\hat{\mathbf{r}}\cdot(\mathbf{x}'-\mathbf{O_g})}. \end{aligned}$$

So, after calculating and interpolating a group's field relative to $\mathbf{O_g}$, the resulting samples must be multiplied by

$$e^{jk\hat{\mathbf{r}}\cdot\mathbf{O_g}} \tag{2.3}$$

before adding them to the other groups' fields.

2.3 Trigonometric Interpolation

Because of the periodic nature of the far fields with respect to θ and ϕ, the interpolation method chosen for this project was trigonometric interpolation via fast Fourier transform [8]. The Fastest Fourier Transform in the West (FFTW) software library was used in order to allow arbitrary-length transforms and maximize portability [9]. The disadvantage of this approach is that after each group's Fourier coefficients are calculated, $L+1$ IFFTs of length $2L$ and $2L$ IFFTs of length $L+1$ must be performed to interpolate along the ϕ and θ directions. This is an $O(L^2 \log L) = O(N \log N)$ computation for each group, for a total of $N^{1.5} \log N$ for the entire object. Note that calculation of the Fourier coefficients involves fewer, shorter FFTs, so this process is also $O(N^{1.5} \log N)$. This is less than the desired complexity of $O(N^{1.5})$, but it still a significant improvement over direct summation, as will be seen in Chapter 3.

Do derive an appropriate formula for trigonometric interpolation, consider interpolation along the θ direction. We begin with $L_g + 1$ samples for each ϕ direction, which we want to interpolate to $L + 1$ samples. Note that L_g is the value of L given in (2.2), except that d is now the diameter of the sphere enclosing the group, so that the number of group samples is $2L_g(L_g + 1)$. It is not entirely trivial how to arrange the $L_g + 1$ Fourier coefficients (denoted X) into a list of L coefficients (called sequence X'), so that an IFFT may be taken. First, note that the unnormalized IDFT is given by

$$Y_k = \sum_{j=1}^{n-1} X_j e^{2\pi jk\sqrt{-1}/n}.$$

In order for the method to qualify as interpolation, the length-$(L_g + 1)$ and length-$(L+1)$ inverse transforms must yield the same number when the "angle" terms $2\pi k/n$ are the same. Specifically, we must have

$$\sum_{j=0}^{L_g-1} X_j e^{2\pi jp\sqrt{-1}/L_g} = \sum_{j=0}^{L-1} X'_j e^{2\pi jq\sqrt{-1}/L}$$

when there exist integers $p \in [0, L_g]$ and $q \in [0, L]$ such that $2\pi p/L_g = 2\pi q/L$. One way to accomplish this is to have $X'_j = X_j$ for $j \leq L_g$ and $X'_j = 0$ for all other j. This, however, has a disadvantage. If there are an odd number of group samples that are all real, then all but one of the Fourier coefficients will be complex conjugate pairs. In order for the inverse transform (and thus the interpolated data) to yield real values, these coefficients must be multiplied by conjugate complex exponential terms during the IDFT. In this summation, the multiplicative factors of the $(1 + j)$-th and $(n - j)$-th terms are conjugates. This suggests that for odd $(L_g + 1)$,

$$X'_j = \left\{X_0, X_1, \ldots, X_{L_g/2}, 0, \ldots, 0, X_{L_g/2+1}, \ldots, X_{L_g}\right\}, \tag{2.4}$$

will interpolate the group samples and return real data if the original samples were real.

When $L_g + 1$ is even, a similar argument can be made. If the X_j are real, then the X_0 and $X_{(L_g+1)/2}$ will be real and the rest will be conjugate pairs. An additional complication for this case is the real $X_{(L_g+1)/2}$ term. In order to ensure a real result from the inverse transform, $X_{(L_g+1)/2}$ must be split in half, and each half multiplied by a conjugate complex exponential pair. This will be accomplished if

$$X'_j = \left\{ X_0, X_1, \ldots, X_{(L_g-1)/2}, \frac{1}{2} X_{(L_g+1)/2}, 0, \ldots, 0, \frac{1}{2} X_{(L_g+1)/2}, X_{(L_g+3)/2}, \ldots, X_{L_g} \right\}. \tag{2.5}$$

The sequence for X'_j in equations (2.4) and (2.5) can be built in a unified manner according to the following algorithm:

1. Calculate X, the Fourier coefficients of the group samples
2. Calculate $nyquist = \left\lceil \frac{L_g}{2} + 1 \right\rceil$.
3. Initialize X' to all zeros.
4. Copy the first $nyquist$ terms of X, in order, to the beginning of X'.
5. Copy the last $(L_g + 1 - nyquist)$ terms of X, in order, to the end of X'.
6. If $(L_g + 1)$ is even:
    (a) $X'_{nyquist-1} \leftarrow \frac{1}{2} X'_{nyquist-1}$.
    (b) $X'_{L-nyquist+2} \leftarrow X'_{nyquist-1}$.

Taking the IFFT of X' then gives the interpolated samples.

There is yet another issue with trigonometric interpolation for this particular application, or any application involving sampling on a sphere. Imagine the group samples being placed in an $(L+1) \times 2L$ matrix, with columns corresponding to constant ϕ and rows to constant θ. Then, taking a Fourier transform along a column would imply that the far field is π-periodic in the the θ direction, that is, that $\mathbf{E}(\theta + \pi, \phi) = \mathbf{E}(\theta, \phi)$. This is not true; the actual relation is

$$\mathbf{E}(\theta + \pi, \phi) = \mathbf{E}(\pi - \theta, \phi \pm \phi). \tag{2.6}$$

A simple, if not very efficient, method to overcome this issue is to add rows to the sample matrix, corresponding to values of θ greater than or equal to π. These extra samples need not be calculated, of course, with the possible exception of a single sample in the $\theta = \pi$ direction. All the other samples can be determined with (2.6).

2.4 Varying the Sample Rate

So far, we have only discussed interpolating the group samples to $L+1$ theta directions and $2L$ phi directions. This is very limiting, since the user may only be interested in a few theta or phi cuts. Therefore, we should alow the user to specify that the final data should have *numTheta* and *numPhi* steps in the theta and phi directions. This corresponds to angular increments of

$$\Delta\theta = \frac{\pi}{numTheta}$$
$$\Delta\phi = \frac{2\pi}{numPhi},$$

so that there are $numTheta + 1$ theta directions and *numPhi* phi directions being sampled. However, if we unthinkingly sample each group in $2L_g$ phi directions, for instance, and interpolate to *numPhi* directions, we will waste time in the cases where

$numPhi < 2L_g$. This is because we will take more samples than we ultimately want, and the interpolation will degenerate to decimation. So, for any group with $numPhi < 2L_g$, we should sample in $numPhi$ directions and skip the interpolation along the phi direction. A similar argument can obviously also be made for the theta direction, except that the appropriate criterion is $numTheta < L_g$.

2.5 A Complete Prescription

The following steps present a complete description of the fast far field algorithm:

1. From the user, read $numTheta$ and $numPhi$, the number of steps to take in the θ and ϕ directions when sampling the object's far field.
2. Calculate L according to (2.2), where d is the diameter of a sphere enclosing the object.
3. Partition the N elements into $O(\sqrt{N})$ groups, each with $O(\sqrt{N})$ elements.
4. For each group:
    (a) Calculate $\mathbf{O_g}$, the center of the sphere surrounding the group and determine L_g according to (2.2), where d is now the diameter of a sphere enclosing the group.

    (b) Determine the actual group sampling rates $sizeTheta$ and $sizePhi$ according to

$$sizeTheta = \begin{cases} 2 \cdot numTheta & \text{if } L_g > numTheta \\ 2L_g + 1 & \text{if } L_g \leq numTheta \end{cases}$$

$$sizePhi = \begin{cases} numPhi & \text{if } 2L_g > numPhi \\ 2L_g & \text{if } 2L_g \leq numPhi, \end{cases}$$

then determine the angular increments with

$$\Delta\theta = \frac{2\pi}{sizeTheta} \quad \text{and} \quad \Delta\phi = \frac{2\pi}{sizePhi}.$$

(c) Using the increments from step 4b, sample the theta direction over the interval $[0, \pi]$ and phi directions over the interval $[0, 2\pi)$.

(d) Extend the sampling interval to $\theta \in (\pi, 2\pi)$ using (2.6).

(e) Using the algorithm in Section 2.3, perform FFT-based interpolation on the group samples. If $sizeTheta = 2 \cdot numTheta$ or $sizePhi = numPhi$, skip the interpolation along the appropriate direction(s).

(f) Multiply the interpolated values by the phase shift given in (2.3).

(g) Add the shifted values to those of the previously-processed groups.

CHAPTER 3

RESULTS

3.1 Comparison with Direct Summation

The fast near to far field transform has been tested with two geometries: a VFY-218 at 500 MHz and a 50×50 Vivaldi array at 5 GHz. Both tests were carried out on a 750 MHz Pentium 3 with 512 MB of RAM. The results of these tests are summarized in Table 3.1.

	VFY-218		50×50 Vivaldi	
N	53,054		106,792	
f (MHz)	500		5,000	
L	218		227	
Groups	242		280	
$numTheta$	180	240	180	240
$numPhi$	360	480	360	480
DS Time (hh:mm)	—	85:36	—	173:00
FNFT Time (hh:mm)	2:50	2:53	7:58	8:00
Relative Error				
RMS	—	2.05×10^{-5}	—	1.38×10^{-4}
Maximum	—	0.0021	—	0.0104

Table 3.1: Running time and accuracy comparisons for direct summation (DS) and the fast near to far transform (FNFT).

The first test with the VFY-218 used $numTheta = 240$ and $numPhi = 480$. The fast near to far transform, at under three hours, fared much better than direct summation, which took over three and a half days to generate the same number of samples. The fast algorithm also maintained good accuracy, with a maximum relative error one-fifth of one percent when compared to the direct summation data. Figures 3.1 and 3.2 give visual confirmation to the accuracy. Even though the VFY-218 is usually considered a scattering problem, these plots are in terms of directivity, which is the native output variable of the computer program. Knowledge of the incident and radiated power could be used to calculate a more meaningful parameter, such as radar cross section, but such a computation is beside the main goal of this project and was not pursued.

An interesting aspect of the VFY-218 results is the difference in running time between a sampling space of size 240×480 versus 180×360. In both cases, $numTheta$ and $numPhi$ were greater than L_g and $2L_g$, respectively, for all groups. Therefore, the same number of group samples were calculated in both cases, so the only difference was the number of interpolations that had to be carried out. In this case, the extra interpolations only took an additional three minutes, a change of less than 2%. This shows that even though the time for trigonometric interpolation is greater than $O(N^{1.5})$, it represents a small part of the total time, at least for modest N. Thus, for N not much greater than 10^5, the algorithm is nearly $O(N^{1.5})$, since the $O(N^{1.5})$ group sampling operation dominates the running time.

The second problem studied was a Vivaldi array with 106,792 elements. The fast algorithm still performed admirably compared to direct summation, at 8 hours versus 7.2 days. The directivity is shown in Figures 3.3 and 3.4 and displays good agreement

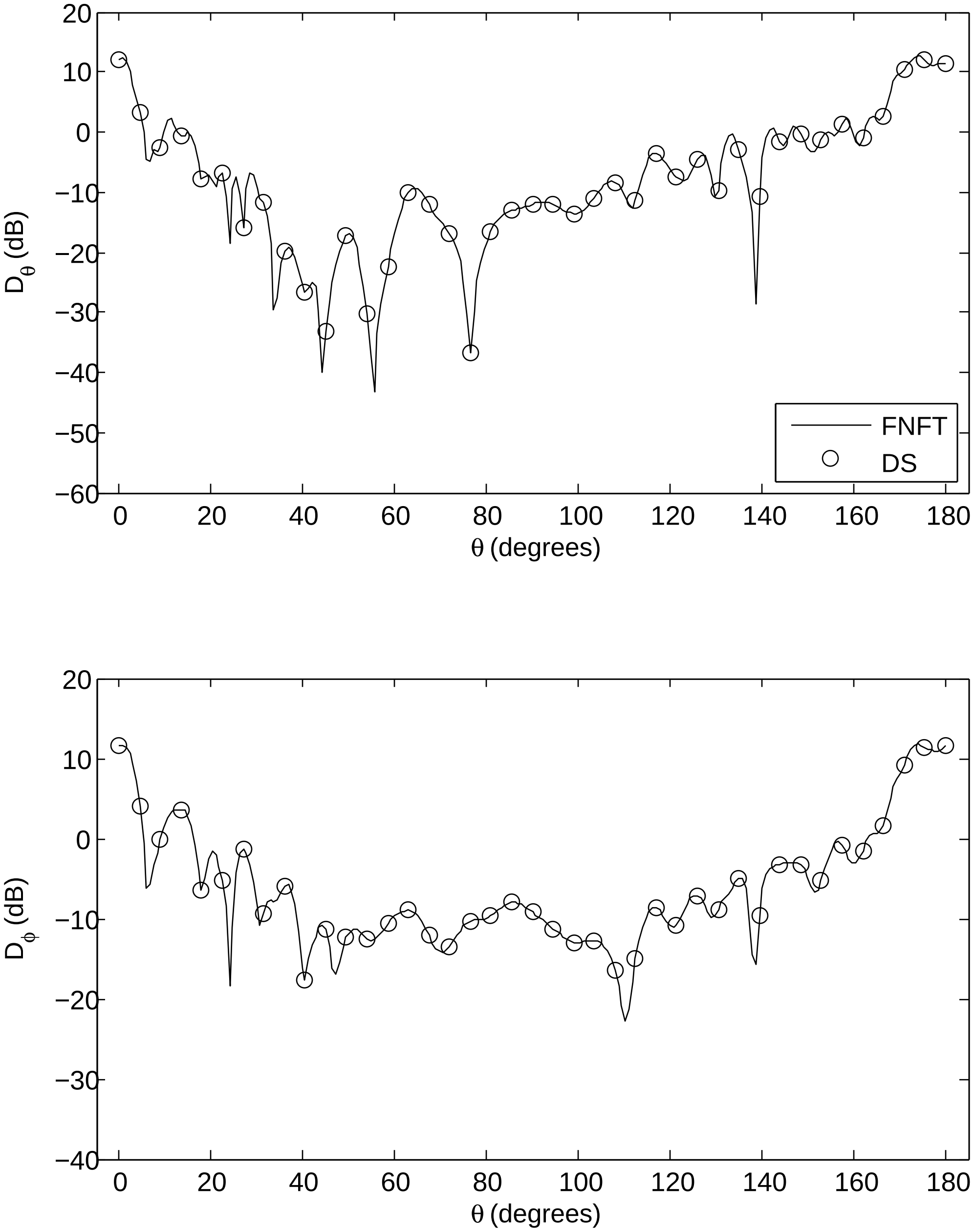

Figure 3.1: VFY-218 directivity, $\phi = 0$ cut, using direct summation (DS) and the fast near to far transform (FNFT).

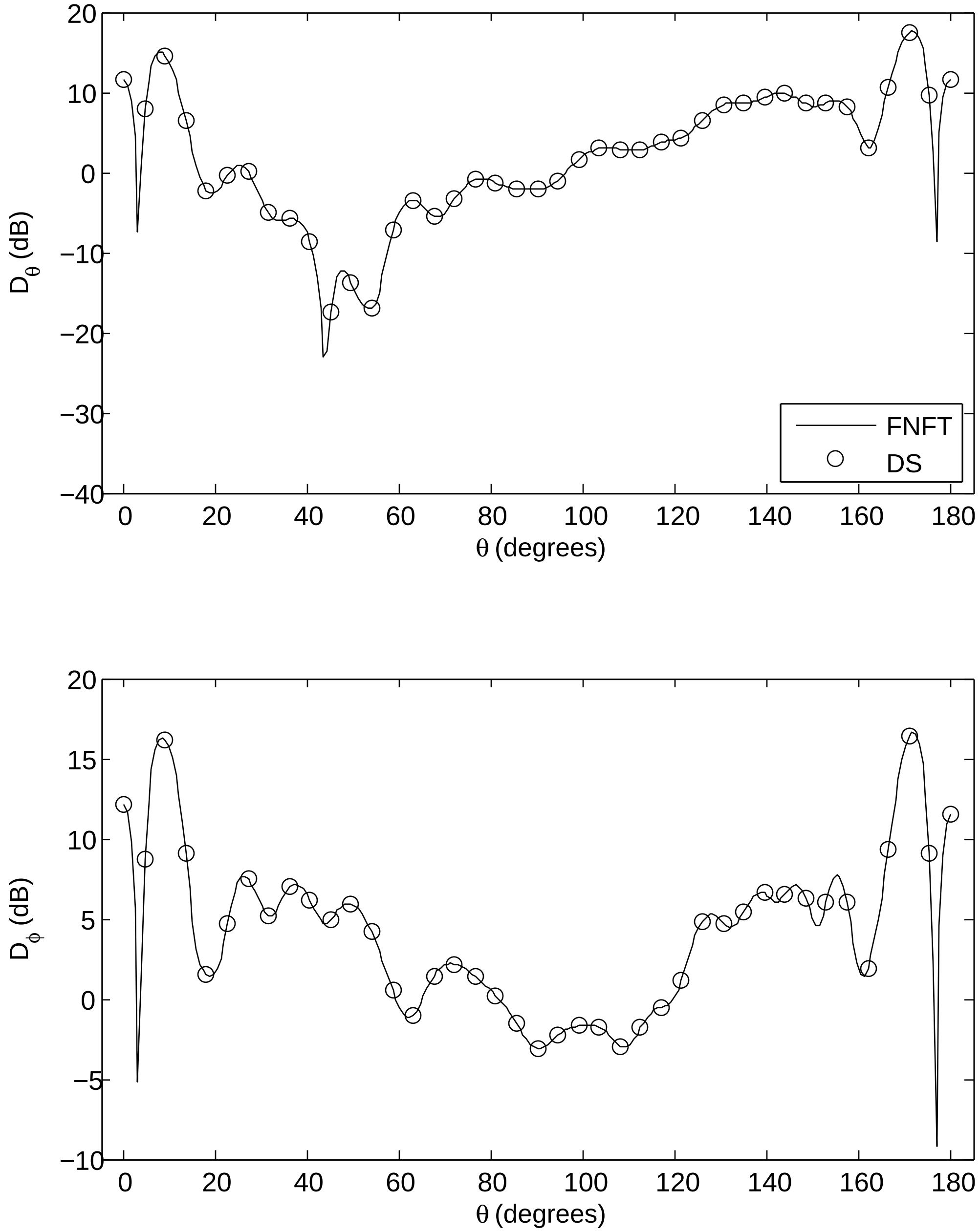

Figure 3.2: VFY-218 directivity, $\phi = \pi/2$ cut, using DS and FNFT.

with direct summation. The performance, however, is disappointing compared to the VFY-218 because the number of elements increased by less than a factor of two, but the running time more than doubled. This can be explained by looking at the number of groups. For the VFY-218, 242 groups were used, which is reasonably close to the ideal value of $\sqrt{N} \approx 230$. The array, on the other hand, used 280 groups but has $\sqrt{N} \approx 327$. During these tests, the partitioning code was set to allow more than $\sqrt{N}$ elements per group, in order to achieve good partitions of small objects like the VFY-218. These results show that as N becomes large, the maximum number of elements per group should decreased to approach the ideal value, $\sqrt{N}$. Also, note that as was the case with the VFY-218, changing $numTheta$ and $numPhi$ only changed the running time by a few minutes. This is more indication that the $O(N^{1.5} \log N)$ interpolation is not as serious a concern as it may have appeared.

3.2 Future Work

There are numerous possibilities for future work on this project. First, if it is desired to continue using trigonometric interpolation, it should be investigated whether data symmetry can be used to shorten the FFT lengths. That is, the current implementation extends the group sample matrix to be 2π-periodic in θ by (approximately) doubling the matrix size and copying redundant data from the $\theta \in [0, \pi)$ range to $\theta \in [\pi, 2\pi)$. There may be a more memory- and time-efficient way to do this. Next, different methods of interpolation could be used, such as spherical harmonics [7] or splines [10, pp. 123–128]. Finally, the partitioning code needs to be refined, as detailed at the end of Section 3.1.

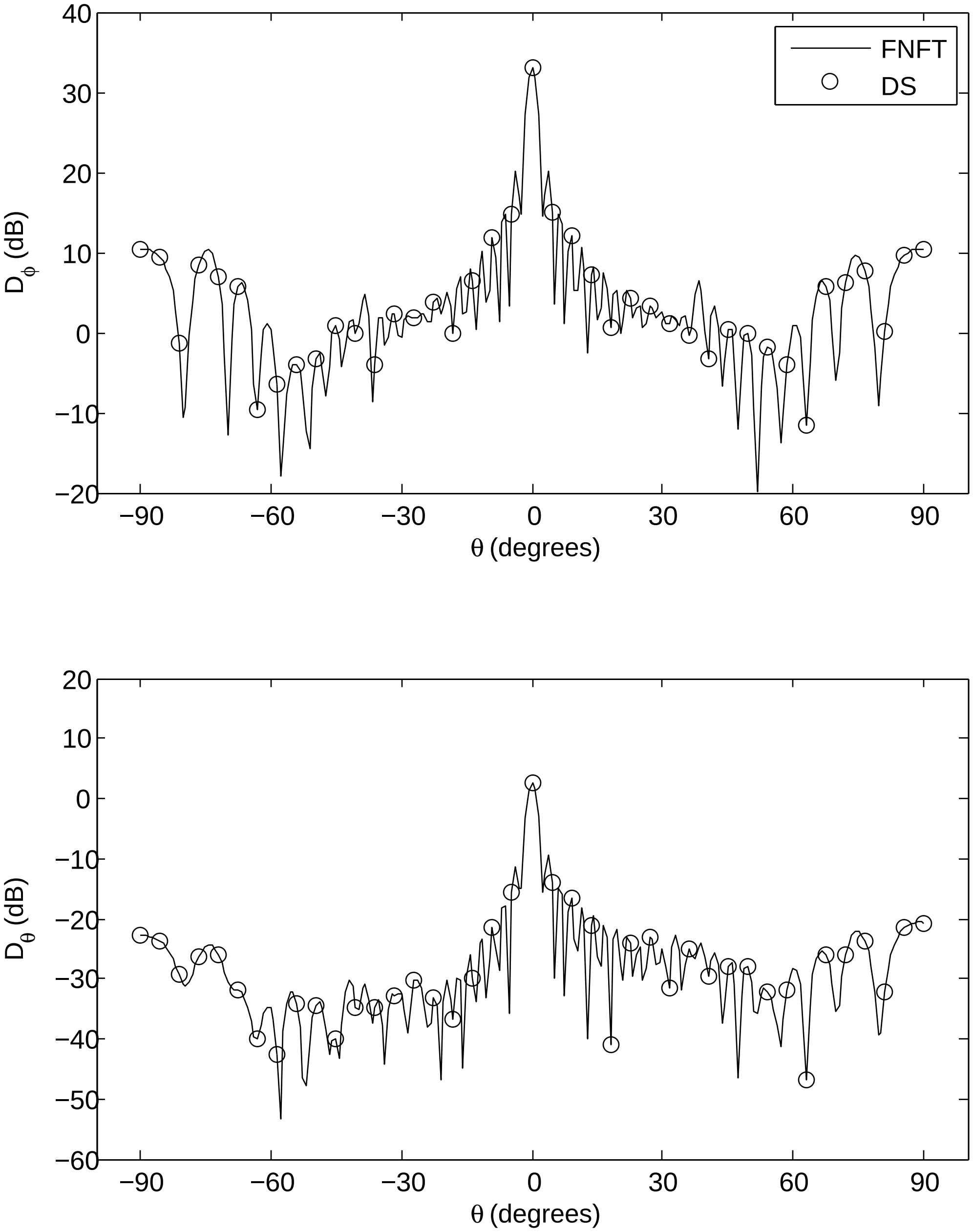

Figure 3.3: Directivity of 50 × 50 Vivaldi Array, $\phi = 0$ cut, using DS and FNFT.

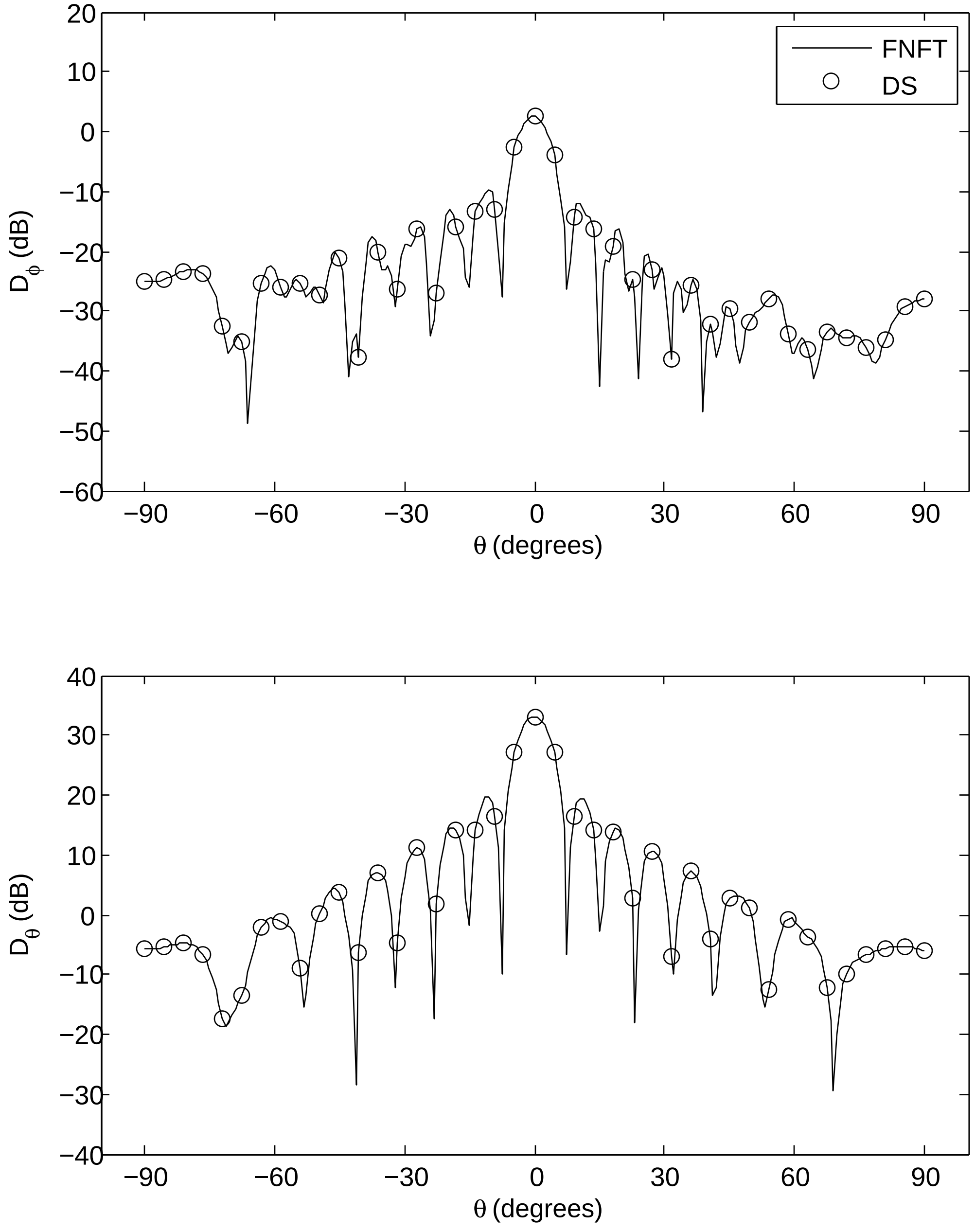

Figure 3.4: Directivity of 50 × 50 Vivaldi Array, $\phi = \pi$ cut, using DS and FNFT.

BIBLIOGRAPHY

[1] C. A. Balanis, *Antenna Theory*, 3rd ed. Hoboken, NJ: John Wiley, 2005.

[2] J. D. Jackson, *Classical Electrodynamics*, 3rd ed. New York: Wiley, 1999.

[3] J. L. Volakis, A. Chatterjee, and L. C. Kempel, *Finite Element Method for Electromagnetics: with Applications to Antennas, Microwave Circuits, and Scattering.* New York: IEEE Press, 1998.

[4] E. W. Weisstein. Barycentric coordinates. From *MathWorld*—A Wolfram Web Resource. [Online]. Available: http://mathworld.wolfram.com/BarycentricCoordinates.html

[5] W. Cheney and D. Kincaid, *Numerical Mathematics and Computing*, 5th ed. Belmont, California: Brooks/Cole-Thomson Learning, 2004.

[6] D. E. Knuth, "Big omicron and big omega and big theta," *SIGACT News*, vol. 8, no. 2, pp. 18–24, 1976.

[7] R. Coifman, V. Rokhlin, and S. Wandzura, "The fast multipole method for the wave equation: A pedestrian prescription," vol. 35, no. 3, pp. 7–12, June 1993.

[8] J. Sarvas, "Performing interpolation and anterpolation entirely by fast fourier transform in the 3-d multilevel fast multipole algorithm," *SIAM J. Numer. Anal.*, vol. 41, no. 6, pp. 2180–2196, Nov. 2003.

[9] M. Frigo and S. G. Johnson, *FFTW*, Massachusetts Institute of Technology, Jan. 2006. [Online]. Available: http://www.fftw.org/fftw3.pdf

[10] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipies in C: The Art of Scientific Computing*, 2nd ed. Cambridge: Cambridge University Press, 2002. [Online]. Available: http://www.nrbook.com/a/bookcpdf.php

[11] R. Beatson and L. Greengard, "A short course on fast multipole methods," in *Wavelets, Multilevel Moethods and Elliptic PDEs*, M. Ainsworth, J. Levesley, W. Light, and M. Martletta, Eds. Oxford University Press, 1997, pp. 1–37.