

A butterfly algorithm for synthetic aperture radar imaging

Laurent Demanet¹, Matthew Ferrara², Nicholas Maxwell³, Jack Poulson⁴, and Lexing Ying⁵

1. Massachusetts Institute of Technology, Department of Mathematics,
77 Massachusetts Avenue, Cambridge, MA 02139
2. Matrix Research, Inc.,
1300 Research Park Drive, Dayton, OH 45432
3. University of Houston, Department of Mathematics,
651 PGH, Houston, TX 77204
4. University of Texas at Austin, ICES
1 University Station, C0200, Austin, TX 78712
5. University of Texas at Austin, Department of Mathematics and ICES,
1 University Station, C1200, Austin, TX 78712

September 2010, revised October 2011

Abstract. In spite of an extensive literature on fast algorithms for synthetic aperture radar (SAR) imaging, it is not currently known if it is possible to accurately form an image from N data points in provable near-linear time complexity. This paper seeks to close this gap by proposing an algorithm which runs in complexity $O(N \log N \log(1/\epsilon))$ without making the far-field approximation or imposing the beam pattern approximation required by time-domain backprojection, with ϵ the desired *pixelwise* accuracy. It is based on the butterfly scheme, which unlike the FFT works for vastly more general oscillatory integrals than the discrete Fourier transform. A complete error analysis is provided: the rigorous complexity bound has additional powers of $\log N$ and $\log(1/\epsilon)$ that are not observed in practice.

Acknowledgment. LD would like to thank Stefan Kunis for early discussions on error propagation analysis in the butterfly algorithm. MF and LD are grateful for AFOSR support from Arje Nachman. MF and NM were partially supported by the AFRL Automatic Target Recognition Center. LD, JP and LY were partially supported by the National Science Foundation. A short version of this paper appeared in [16].

1 Introduction

1.1 Setup

Synthetic-aperture radar (SAR) is an imaging modality that produces images of a scene from measurements of scattered electromagnetic waves. Pulses of microwaves are sent from an antenna aboard an airplane or a satellite, scattered by objects on the surface of the Earth, and recorded by the same (or a different) antenna. The imaging problem consists in recovering a reflectivity profile that explains the recorded pulse-echo data.

- *Image space* is indexed by $\mathbf{x} = (x, y) \in \mathbb{R}^2$, the horizontal coordinates. The scatterers are assumed to be at a known elevation $z = h(x, y)$, so we have the embedding $\mathbf{x}_T = ((x, y), h(x, y)) \in \mathbb{R}^3$. The reflectivity profile is a function $m(\mathbf{x})$ whose magnitude indicates the strength of the reflection by the object at x_T , as an impedance contrast for instance.
- *Data space* is indexed by ω , the frequency of the recorded signal, and s , a parameter that defines the position of the antenna through a function $\gamma(s) \in \mathbb{R}^3$. Data are given by a function $d(\omega, s)$, whose value is the result of a measurement of the strength of the recorded signal at angular frequency $\omega = 2\pi f$, when the antenna is at $\gamma(s)$.

Under very general and widely accepted assumptions¹, this imaging map is an oscillatory integral. We make three additional but unessential assumptions that can easily be removed: 1) monostatic SAR in which the transmitter antenna is also the receiver, 2) no consideration of the orientation of the plane, and 3) the phase-center approximation, in which the antenna is far enough from the targets that it is considered as a point. Imaging is then done by some “generalized” filtered backprojection:

$$m(\mathbf{x}) = \int_{\Omega} e^{-2i\omega|\gamma(s)-\mathbf{x}_T|/c} B(\omega, s, \mathbf{x}) d(\omega, s) ds d\omega, \quad (1)$$

where $B(\omega, s, \mathbf{x})$ is an amplitude function, and $\mathbf{x}_T = (x_1, x_2, h(x_1, x_2))$ is the target point. We will comment later on the backprojection interpretation. See [14] for the justification of this formula. The figure below illustrates some of the notations.

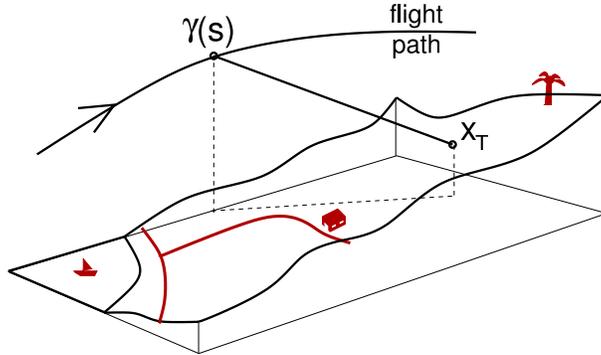


Figure 1: The SAR imaging setup

Here Ω is the acquisition manifold, normally a rectangle $[\omega_{\min}, \omega_{\max}] \times [s_1, s_2]$. The amplitude factor $B(\omega, s, x)$ is chosen so that the formula above is a good approximate inverse to the forward/modeling/reprojection operator

$$d(\omega, s) = \int e^{2i\omega|\gamma(s)-x_T|/c} A(\omega, s, x) m(x) dx_1 dx_2. \quad (2)$$

In this integral, the amplitude factor $A(\omega, s, x)$ is

$$A(\omega, s, x) = -\omega^2 P(\omega) \frac{J(\omega, \widehat{x_T - \gamma(s)}) W(\omega, \widehat{x_T - \gamma(s)})}{(4\pi|x_T - \gamma(s)|^2)},$$

where $P(\omega)$ is the transfer function of the pulse, and J and W are the respective antenna beam patterns at transmission and reception. The hat over a vector denotes unit length normalization. The corresponding amplitude B for imaging is cumbersome to write precisely without introducing the so-called Stolt change of variables; suffice it to say that

$$B = \frac{\chi}{A d_B},$$

where A is the amplitude above, d_B is the so-called Beylkin determinant, and χ is an ad-hoc cutoff that prevents division by zero. The details are in [14], but they do not matter at this level of exposition.

It is difficult to form a large image by the integral (1) in real-time with a single instruction thread, hence the need for fast algorithms. That is the price to pay for opening up the beam and leveraging *synthetic* aperture in order to get a good resolution. Contrast this situation to phased-array transducer systems with narrow beams used in low-resolution medical ultrasound imaging, where imaging at 20 frames per second is commonplace, but where no such mathematical transform as (1) is needed.

The contribution of this paper is to propose a fast and accurate way of evaluating oscillatory integrals such as (1). We start by reviewing the existing algorithms and their range of applicability.

¹Single scattering in the Born approximation, scalar wavefields, no dispersion, no attempt at addressing three-dimensional effects such as shadowing and layover, start-stop setup, no attempt at estimating target motion. This is the setup in [14].

1.2 Existing algorithms

Denote by $\Delta\omega = \omega_{\max} - \omega_{\min}$ the bandwidth of the measurements. For simplicity, we will assume that the bandwidth is on the same order of magnitude as the representative “carrier” frequency $\omega_0 \simeq (\omega_{\min} + \omega_{\max})/2$, so we have broadband measurements.

The Nyquist-Shannon sampling rate should be respected both in image space and in data space.

- In image space, we expect variations on the order of the wavelength c/ω_0 in both directions², and the scene to be imaged has sidelength L , so the total number of pixels is proportional to $L^2\omega_0^2/c^2$.
- In data space, a frequency grid spacing of $O(c/L)$ is called for to access distances on the order of L , so we need $O(\omega_0 L/c)$ samples. The distance between pulses should be on the order of the wavelength $O(c/\omega_0)$ to attain the same wavelength resolution on the ground, so we need $O(\omega_0 L/c)$ samples in slow time as well. So the total number of data points is proportional to $L^2\omega_0^2/c^2$.

The complexity of specifying either a dataset, called N , is therefore proportional to the complexity of specifying an image, and increases quadratically in the frequency ω_0 :

$$N = O(L^2\omega_0^2/c^2).$$

It is the scaling of the complexity of the imaging algorithm as a function of this parameter N which is of interest. The “naive algorithm” consists in performing the direct summation from a quadrature of (1). It has complexity $O(N^2)$.

Traditionally, it is only in contexts where the problem formulation is *simplified* that designing faster algorithms is possible. Two levels of simplification are popular in the literature:

1. **The separability assumption** $B(\omega, s, x) = P(\omega)Q(s, x)$. This assumption only makes sense if the antenna beam patterns are independent of frequency. In this setting, we may evaluate equation (1) by the following sequence of steps: for each s , multiply the data by $1/P(\omega)$, perform a Fourier transform in ω evaluated at $2|\gamma(s) - \mathbf{x}_T|/c$, and multiply by $Q(s, x)$. Iterate and sum over s . This procedure results in an algorithm of complexity $O(N^{3/2})$. It is called filtered backprojection (proper), because it can be seen as integration along curves of equal range when expressed as acting on data $\hat{d}(t, s)$ of time t . One would also speak of a generalized Radon transform [4]. The computation of the remaining sum over s can be further simplified by an ingenious trick of multiscale (computational) beamforming in certain settings. The resulting class of algorithms has come to be known as “fast backprojection” (FBP). It includes work by Nilsson and Andersson [27] and Yegulalp [37] on computation of circular averages (monostatic SAR), and by Ding and Munson on computation of elliptical averages (bistatic SAR) [18]. A subsequent contribution by Ulander, Hellsten, and Stenström [35] covered a more general imaging setting, but still within the limitation of an omnidirectional antenna, flat topography, and perturbative deviations from a linear track. Most of these papers operate in the $O(N \log N)$ complexity regime yet still seem to suffer from accuracy concerns. It is unclear at this point whether a pointwise or mean-square error estimate would hold for any variant of fast backprojection.
2. **The far-field assumption** $\|\gamma(s) - x_T\| \simeq \|\gamma(s)\| - \hat{x}_T \cdot \gamma(s)$. This further simplification makes sense if the target is so far from the antenna that the circles of equal distance can be treated as straight lines, like in spotlight SAR. In this setting equation (1) becomes a 2D Fourier transform, albeit not on a uniform grid [26]. In the time domain, we would speak of a Radon transform instead of a generalized Radon transform. The USFFT method of Dutt and Rokhlin [19] and its variants [5, 10] apply to this problem and yield algorithms of complexity $O(N \log N)$. The Polar Format Algorithm (PFA) [36], which interpolates the data from polar raster onto a rectilinear grid, is also a reasonable approach. A comparison between PFA, USFFT, and NUFFT techniques for SAR is given in Andersson et al [1]. Fast backprojection algorithms were originally developed for the Radon transform in the tomographic setting by Basu and Bresler [3] and Boag, Bresler, and Michielssen [2], and then adapted to monostatic SAR in the far field regime by Xiao, Munson, Basu, and Bresler [40]. (As discussed earlier, this line of work on FBP continued without the far-field approximation at least in [18, 35].)

²This can be refined by considering range direction and cross-range direction, in the case of narrowband measurements.

It should be noted that ultrasound tomography (a.k.a. diffraction tomography) is algorithmically similar to SAR in the far-field regime. Tomographic data can be interpreted as unequally-spaced samples in the Fourier domain via the projection-slide theorem, both in diffraction tomography and in far-field SAR [26]. Fast algorithms for ultrasound tomography fall into the same two categories as above, namely FFT-based reconstruction [9] and fast backprojection [17].

In contrast, this paper presents a fast “butterfly” algorithm good for much more general radar setups. None of the assumptions above are made; only minimal smoothness properties of the functions $\gamma(s)$ and $B(\omega, s, x)$ are required. In fact, the butterfly scheme is intrinsically robust and we anticipate that it would easily accommodate refinements such as multistatic SAR (several sources and antennas), or taking into account the orientation of the antenna via the pitch, roll and yaw angles as measured by the Inertial Navigation System (INS).

The main idea behind the butterfly scheme is that of *low-rank interaction*. This idea conveys a very important and general principle of quantification of the “information content” in high-frequency scattering.

1.3 Low-rank interactions

The phase center of an antenna is the point $\gamma(s)$ introduced earlier, about which the antenna beam patterns are constructed as functions of angular frequency ω direction $\widehat{x - \gamma}$. It draws its name from the fact that a more accurate representation of the radiation field from an antenna Γ is (we drop the s dependence of γ for the time being.)

$$\begin{aligned} u(x, \omega) &= \int_{\Gamma} \frac{e^{ik|x-y|}}{4\pi|x-y|} j(y, \omega) dS_y, \\ &\simeq \frac{e^{ik|x-\gamma|}}{4\pi|x-\gamma|} \int_{\Gamma} e^{-ik(\widehat{x-\gamma}) \cdot y} j(y, \omega) dS_y, \\ &=: \frac{e^{ik|x-\gamma|}}{4\pi|x-\gamma|} J(\omega, \widehat{x-\gamma}) \quad (\omega = kc), \end{aligned}$$

hence γ should be chosen as a good “center” for the approximately circular phase lines of $u(x, \omega)$. Here $j(y, \omega)$ is a scalar counterpart of the current density on the antenna³. To pass to the second line the well-known far-field approximation $|x - \gamma| \gg |y - \gamma|$ was used. While the full surface integral over the antenna is impractical for radar imaging, this phase center reduction has the advantage of presenting the antenna beam patterns as functions on the sphere of outgoing directions. (A similar argument can be made for the receiving antenna.)

Another way of reducing the complexity of the integral, without making the far-field approximation, consists in finding several *equivalent sources* γ_i , with weights J_i ; as well as several regions \mathcal{A} such that

$$u(x, \omega) = \sum_i \frac{e^{ik|x-\gamma_i|}}{4\pi|x-\gamma_i|} J_i + O(\epsilon), \quad x \in \mathcal{A}.$$

Here, the error is under control and denoted ϵ . In other words, if we are willing to restrict ourselves to a certain region \mathcal{A} of space, how many “phase centers” γ_i indexed by i are really needed to synthesize the radiation field to prescribed accuracy? A simple numerical experiment shown in the Figure below recovers the radiation field created by a Y-shaped antenna, observed in the box in the lower-right corner, for any possible current distribution, using only 9 equivalent points on the antenna. They are the red dots, and their choice guarantees an accuracy of 1 percent.

³We apologize in passing to the engineers who are used to $j = \sqrt{-1}$.

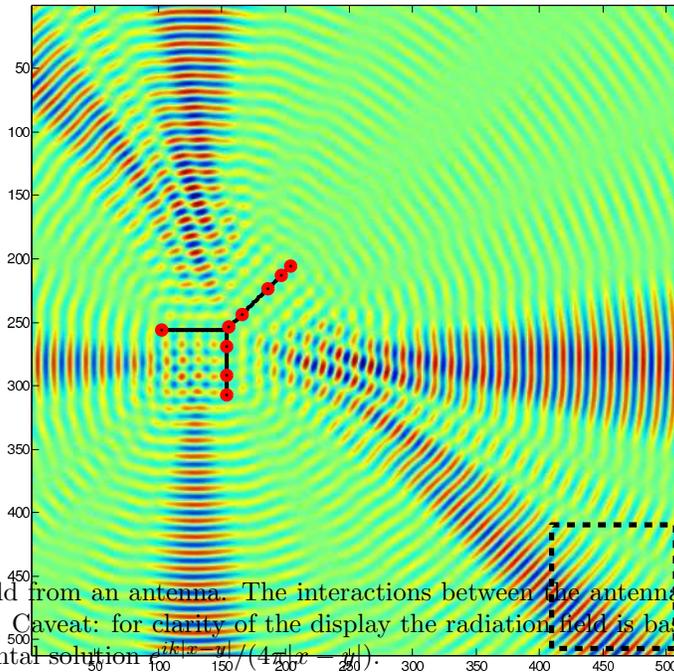


Figure 2: Radiation field from an antenna. The interactions between the antenna and the box surrounded by a dashed line is low. *Caveat:* for clarity of the display the radiation field is based on the kernel $e^{ik|x-y|}$, instead of the fundamental solution $e^{ik|x-y|}/(4\pi|x-y|)$.

This numerical experiment is a simple exercise in linear algebra, solved by QR with pivoting, or more generally with an interpolative interpolation [23]. Ultimately, the numerical experiment is successful because of the *low-rank factorization* property of the Green’s function when y is restricted to the antenna and x is restricted to the box \mathcal{A} . The underlying fundamental property of Green’s functions is that factorization is guaranteed to work, with a low rank independent of ω , if the following adimensional number is low,

$$F = \frac{\text{diam}(\Gamma) \times \text{diam}(\mathcal{A})}{\lambda \times d(\Gamma, \mathcal{A})}.$$

We may call F an “algorithmic Fresnel number” in analogy with the discussion of Fraunhofer vs. Fresnel diffraction in physics textbooks. Its value should be comparable to 1 or lower for the low-rank property to hold. Here $\text{diam}(\Gamma)$ is the antenna diameter, $\text{diam}(\mathcal{A})$ is the largest diagonal of the box \mathcal{A} , $\lambda = 2\pi/\omega$ is the wavelength, and $d(\Gamma, \mathcal{A})$ is the distance between the antenna and the box. Similar ideas appear in work of Michielssen and Boag [25], Engquist and Ying [20], Candès, Demanet, and Ying [12], Rokhlin [29], Brandt [7], and likely many others.

Note that if an expansion is valid in a box \mathcal{A} , it is also valid in a large truncated cone in the shadow of this box, as seen from Γ . Martinsson and Rokhlin studied the weak dependence of the length of this truncated cone on the desired accuracy [24].

1.4 The butterfly algorithm

The butterfly algorithm is a systematic way of leveraging low-rank interactions in the scope of a fast algorithm for oscillatory integrals. The pulse-echo data now replaces the antenna as a virtual “source” of radiation, so the physical problem is different from that presented in the previous section, but the ideas remain the same.

The butterfly algorithm originates from work of Michielssen and Boag [25], and has recently been popularized in a string of papers by Rokhlin and O’Neil [28], Ying [38], Candès, Demanet and Ying [12] and Tygert [34]. Note that the algorithmic variant presented in our earlier work [12] is particularly well suited for the application to SAR imaging: unlike [28] it does not have a precomputation step. The butterfly is a natural descendant of, or variant of, the fast multipole method [22, 29] for high-frequency scattering, in the sense that low-rank interactions are adequate “summaries” that serve as a substitute for multipole expansions [39].

If we let $y = (\omega', s)$, with $\omega' = \omega/\omega_0$ a rescaled frequency variable, then we may succinctly write any quadrature for (1) as

$$m(x) = \sum_y K(x, y)d(y), \tag{3}$$

with $K(x, y)$ the oscillatory kernel, and $d(y)$ the data samples multiplied by the quadrature weights. We refer to the direct summation in (3) as the “naive algorithm”. Low-rank interactions come into play through the problem of finding a good approximation

$$m(x) = \sum_{j=1}^r K(x, y_j) \delta_j + O(\epsilon), \quad (4)$$

where (y_j, δ_j) are equivalent sources. In order for the number r of terms to be independent of the carrier frequency ω_0 (or $N \sim \omega_0^2$), it suffices to take $x \in A$, and to restrict the sum to $y \in B$, in such a way that the “algorithmic Fresnel number” is small, i.e.

$$\text{diam}(A) \times \text{diam}(B) \leq \frac{H}{\omega_0}, \quad (5)$$

for some constant H that has the dimension of a length, with value comparable to the altitude of the antenna. This property of r was established in earlier work of two of the authors, in the more general setting of Fourier integral operators [12]. It holds for SAR imaging if the trajectory $\gamma(s)$ is smooth, i.e., a C^∞ function of s .

If B would cover the whole data space, we would be in presence of the full sum. In that case the range of validity of a formula like (4) would be restricted to very small boxes A – of diameter $O(1/\omega_0)$ – and to each such small box A would correspond a given set of equivalent sources (y_j, δ_j) . If the information of the (y_j, δ_j) were available for each small box A , then we would be in presence of a very fast algorithm: a superposition of $r = O(1)$ terms for each of the $O(\omega_0^2) = O(N)$ boxes in A would suffice for imaging. This is unfortunately not the case.

The butterfly scheme is a way of computing these equivalent sources by playing on the sizes of A and B in a multiscale manner. It is possible to tile model and data space with boxes that satisfy the scaling (5), and consider the low-rank interactions between any pair of such boxes. It is advantageous to generate such tilings by means of quadtree partitions of model and data space. See Figure 3, where data space (y) is on the right, and model space (x) on the left.

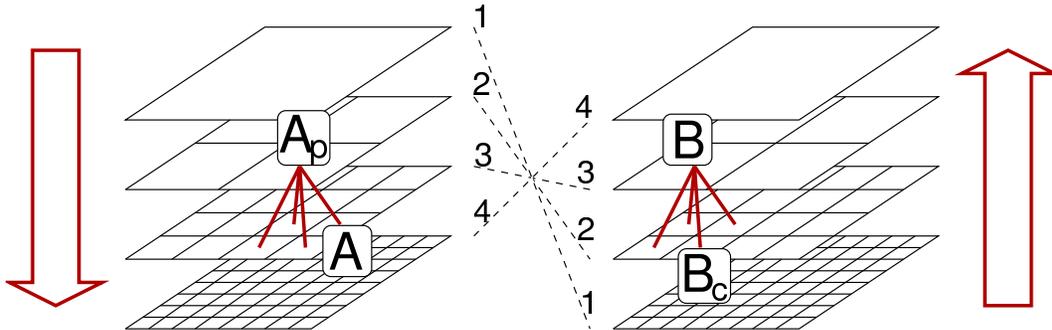


Figure 3: The two quadtrees of the butterfly scheme.

For instance,

- The fine boxes at the leaves (bottom) of the tree on the right can be paired with a large box at the root (top) of the tree on the left. The pairing corresponds to the dashed line labeled “1”. If the boxes B are small enough ($1/\omega_0$ by $1/\omega_0$), then the scaling (5) is respected. This choice of tiling corresponds to sums (4) restricted to only a few terms: it is straightforward to compute directly, without the δ_j . But it is not very useful since we want the whole sum.
- On the other hand, the large box B at the root of the tree can be paired with small boxes A at the leaves. This pairing goes by the number “4”. It corresponds to a low-rank view of the whole sum (3), only valid in certain very small sets A on the x -side. It is exactly what we are interested in, but the δ_j in the expansion are unknown to us.

The core of the butterfly algorithm is the ability to update low-rank interactions in a multiscale fashion, down the left tree and up the right tree, by respectively grouping and subdividing boxes. In the picture this allows to iteratively obtain the δ_j at all scales, from the pairing “1” to the pairing “4”.

The details of the butterfly scheme concern the choice of y_j in (4), how to realize the low-rank expansion as an interpolation problem, and how to update the δ_j weights from one scale to the next. These details are presented in section 2 for completeness, and follow from our previous work in [12]. Let us mention that it is the “Chebyshev interpolation” version of the butterfly algorithm which is used in this paper; it is unclear that the other variants would be equally well suited for SAR imaging.

We now switch to the rigorous performance guarantee enjoyed by the butterfly algorithm, which was missing in our previous work [12].

1.5 Accuracy and complexity bounds

In this paper, like in [11, 12], we choose the radius of convergence of Taylor expansions as a measure of smoothness of real-analytic functions. In one spatial dimension, a function $f(x)$ is (Q, R) -analytic if it is infinitely differentiable and its derivatives obey

$$|f^{(n)}(x)| \leq Q n! R^{-n}.$$

The number R is simply a lower bound on the radius of convergence of Taylor expansions of f , uniform over all points where f is considered. We say that a function $f(x)$ of $x \in \mathbb{R}^2$ is (Q, R) -analytic if its directional derivative along any line obeys the same estimate: for any unit-length d ,

$$|(d \cdot \nabla)^n f(x)| \leq Q n! R^{-n}.$$

Our main assumption on the kernel $K(x, y)$ is that it can be written in Fourier integral form as $a(x, y)e^{iM\phi(x, y)}$, with the amplitude a and the phase ϕ both analytic in x and y separately. (For convenience we use the same values of the constants Q and R in the following two equations.)

$$\begin{aligned} |(d_1 \cdot \nabla_x)^{n_1} (d_2 \cdot \nabla_y)^{n_2} a(x, y)| &\leq Q n_1! n_2! R^{-n_1} R^{-n_2}, \\ |(d_1 \cdot \nabla_x)^{n_1} (d_2 \cdot \nabla_y)^{n_2} \phi(x, y)| &\leq Q n_1! n_2! R^{-n_1} R^{-n_2}. \end{aligned}$$

Manifestly, the SAR kernel of equation (1) is of the form $ae^{iM\phi}$ with $M = \omega_0 = O(\sqrt{N})$.

The following complexity result depends on N and ϵ . The dependence on Q and R will not be kept track of.

Theorem 1. *Assume that the flight path $\gamma(s)$ and the amplitude $B(\omega, x, s)$ are both real-analytic functions of their arguments. Write $y = (\omega/\omega_0, s)$, and*

$$K(x, y) = a(x, y)e^{iM\Phi(x, y)}.$$

Then the variant of the butterfly method presented in this paper, which uses Chebyshev interpolation, provides an approximation $\tilde{m}(x) = \sum_y \tilde{K}(x, y)d(y)$ to $m(x) = \sum_y K(x, y)d(y)$ obeying

$$\|\tilde{m} - m\|_\infty \leq \epsilon \sum_y |d(y)|,$$

in exact arithmetic, and in (sequential) algorithmic complexity

$$C(Q, R) \times \max\{\log^4(\frac{1}{\epsilon}), (\log^4 N) \log^4(C \log N)\} \times N \log N.$$

The proof is in Section 4. Recall that $d(y)$ are data samples normalized with the proper quadrature weights, such that sums over y approximate integrals. Hence $\sum_y |d(y)|$ is a quantity upper-bounded uniformly over N . It is also independent of ϵ since the choice of discretization of the integral has nothing to do with the truncation inherent to fast summation. Let us also note that the above theorem contains no statement about the discretization error; only the ℓ_∞ discrepancy between the result of naive summation (3) and the result of the fast algorithm is controlled.

2 The butterfly algorithm for oscillatory integrals

Let us denote by X the set of all x (positions) indexing model space, and by Y the set of all y (normalized frequencies and slow times) indexing data space. From the discussion above, it is clear that both $|X|$ and $|Y|$ are on the order of $N = O(M^2)$. By rescaling the geometry if necessary, we can assume that X and Y are both supported in the unit square $[0, 1]^2$. In this section, unlike in the numerical code, we do not worry about the values of numerical constants: for brevity only the asymptotic behavior in terms of M is accounted for. The computational problem is then to approximate $m(x)$ defined by

$$m(x) = \sum_{y \in Y} a(x, y) e^{iM\Phi(x, y)} d(y).$$

We now give a brief discussion of the butterfly algorithm for computing this oscillatory summation. The presentation follows closely to the one of [12] and the new contribution is an easy way to address the amplitude function $a(x, y)$. It is useful to keep an eye on Figure 3 while following the description of the algorithm.

Suppose that A and B are two square boxes in $[0, 1]^2$, while A is considered to be a box in the X domain and B in the Y domain. We denote their centers, respectively, by $x_0(A)$ and $y_0(B)$; and the length of their diagonals, respectively, by $\text{diam}(A)$ and $\text{diam}(B)$. The most important component of the butterfly algorithm is the existence of a low-rank approximation for the kernel

$$\left| a(x, y) e^{iM\Phi(x, y)} - \sum_{t=1}^{r_\epsilon} \alpha_t^{AB}(x) \beta_t^{AB}(y) \right| \leq \epsilon \quad (6)$$

for $x \in A$ and $y \in B$ when $\text{diam}(A)\text{diam}(B) \lesssim 1/M$. The quantities $\alpha_t^{AB}(x)$ and $\beta_t^{AB}(y)$ will be defined below in equations (9, 11). Define $m^B(x)$ to be the partial sum restricted to, or ‘‘potential generated by’’, $y \in B$. The benefit of the low-rank approximation is that it gives rise to a compact representation for $m^B(x)$ when restricted to $x \in A$:

$$m^B(x) \simeq \sum_{t=1}^{r_\epsilon} \alpha_t^{AB}(x) \left(\sum_{y \in B} \beta_t^{AB}(y) d(y) \right) \quad \forall x \in A.$$

Therefore, any coefficients $\{\delta_t^{AB}\}_t$ obeying

$$\delta_t^{AB} \simeq \sum_{y \in B} \beta_t^{AB}(y) d(y), \quad (7)$$

offer a good approximation to $m^B(x)$ for $x \in A$.

In order to find a low-rank approximation, we introduce the residual phase associated with the pair (A, B) ,

$$R^{AB}(x, y) := \Phi(x, y) - \Phi(x_0(A), y) - \Phi(x, y_0(B)) + \Phi(x_0(A), y_0(B)). \quad (8)$$

Under the condition that $\Phi(x, y)$ is real-analytic both in x and in y , and $\text{diam}(A)\text{diam}(B) \lesssim 1/M$, it is easy to show that $R^{AB}(x, y) = O(1/M)$ for $x \in A$ and $y \in B$. As a result, it was shown in [12], in the case $a(x, y) = 1$, that r_ϵ in equation (6) can be bounded by a constant times $\log^4(1/\epsilon)$. This bound can be further refined to a constant times $\log^2(1/\epsilon)$, and made valid for arbitrary analytic $a(x, y)$, using the proof methods presented in Section 4. The point is that those bounds on r_ϵ are *independent of M* , and only depend weakly on the desired accuracy.

One way to realize such low values of r_ϵ , as explained in [12], is to use polynomial interpolation in x when $\text{diam}(A) \lesssim 1/\sqrt{M}$ and in y when $\text{diam}(B) \lesssim 1/\sqrt{M}$. The interpolation points are placed on tensor Chebyshev grids for efficiency. For some small positive integer q , the Chebyshev grid of order q on the centered unit interval $[-1/2, 1/2]$ is defined by

$$\left\{ z_j = \frac{1}{2} \cos \left(\frac{j\pi}{q-1} \right) \right\}_{0 \leq j \leq q-1}.$$

The Lagrange basis polynomials $L_i(z)$ of this grid are given by

$$L_j(z) := \prod_{0 \leq k \leq q-1, k \neq j} \frac{z - z_k}{z_j - z_k}.$$

By taking tensor products, we can define the two dimensional Chebyshev grid $\{(z_{t_1}, z_{t_2})\}$ for the centered unit square and its Chebyshev basis functions

$$L_t(z_1, z_2) := L_{t_1}(z_1) \cdot L_{t_2}(z_2), \quad \text{for } t = (t_1, t_2).$$

For a general square box B in the Y domain, its Chebyshev grid can be defined similarly by appropriate scaling and shifting. We denote this grid by $\{y_t^B\}$ and its Lagrange basis functions for its Chebyshev grid by $\{L_t^B\}$. When $\text{diam}(B) \lesssim 1/\sqrt{M}$, Lagrange interpolation on the grid adapted to B provides the approximation

$$a(x, y)e^{iMR^{AB}(x, y)} \simeq \sum_t a(x, y_t^B)e^{iMR^{AB}(x, y_t^B)} L_t^B(y).$$

Similarly, for a box A in the X domain, its Chebyshev grid and Lagrange basis functions are denoted by $\{x_t^A\}$ and $\{L_t^A\}$, respectively. When $\text{diam}(A) \lesssim 1/\sqrt{M}$, Lagrange interpolation on the grid adapted to A provides the approximation

$$a(x, y)e^{iMR^{AB}(x, y)} \simeq \sum_t L_t^A(x)a(x_t^A, y)e^{iMR^{AB}(x_t^A, y)}$$

It will be shown in Section 4 that the number q of Chebyshev points grows logarithmically in the error level ϵ , resulting in $r_\epsilon = q^2 = O(\log^2 1/\epsilon)$ as announced earlier. Alternatively, one could generalize Theorem 3.3 in [12] to the case of a non-constant amplitude $a(x, y)$. In practice, it is advantageous to take q with values ranging from 5 to 10 in order to obtain ‘‘a few’’ to ‘‘several’’ digits of accuracy. The section on numerical experiments contains more details on the choice of q vs. accuracy.

To pass from low-rank approximations of $a(x, y)e^{iMR^{AB}(x, y)}$ to those for the true kernel $a(x, y)e^{iM\Phi(x, y)}$, we restore the other factors in (8). When $\text{diam}(B) \lesssim 1/\sqrt{M}$, this gives

$$a(x, y)e^{iM\Phi(x, y)} \simeq \sum_t \left(a(x, y_t^B)e^{iM\Phi(x, y_t^B)} \right) \left(e^{-iM\Phi(x_0(A), y_t^B)} L_t^B(y)e^{iM\Phi(x_0(A), y)} \right)$$

In terms of the earlier notations,

$$\alpha_t^{AB}(x) = a(x, y_t^B)e^{iM\Phi(x, y_t^B)}, \quad \beta_t^{AB}(y) = e^{-iM\Phi(x_0(A), y_t^B)} L_t^B(y)e^{iM\Phi(x_0(A), y)}, \quad (9)$$

and the expansion coefficients $\{\delta_t^{AB}\}_t$ for the potential should obey the condition

$$\delta_t^{AB} \simeq \sum_{y \in B} \beta_t^{AB}(y)f(y) = e^{-iM\Phi(x_0(A), y_t^B)} \sum_{y \in B} \left(L_t^B(y)e^{iM\Phi(x_0(A), y)} d(y) \right). \quad (10)$$

Similarly when $\text{diam}(A) \lesssim 1/\sqrt{M}$, we have

$$a(x, y)e^{iM\Phi(x, y)} \simeq \sum_t \left(e^{iM\Phi(x, y_0(B))} L_t^A(x)e^{-iM\Phi(x_t^A, y_0(B))} \right) \left(a(x_t^A, y)e^{iM\Phi(x_t^A, y)} \right)$$

In terms of the earlier notations,

$$\alpha_t^{AB}(x) = e^{iM\Phi(x, y_0(B))} L_t^A(x)e^{-iM\Phi(x_t^A, y_0(B))}, \quad \beta_t^{AB}(y) = a(x_t^A, y)e^{iM\Phi(x_t^A, y)}, \quad (11)$$

and the expansion coefficients $\{\delta_t^{AB}\}$ should obey

$$\delta_t^{AB} \simeq \sum_{y \in B} \beta_t^{AB}(y)d(y) = \sum_{y \in B} a(x_t^A, y)e^{iM\Phi(x_t^A, y)} d(y) = m^B(x_t^A). \quad (12)$$

Combining these expansions with the general structure of the butterfly scheme, we arrive at the following algorithm. It is a slight modification of the one proposed in [12].

1. *Preliminaries.* Construct two quadrees T_X and T_Y for X and Y . Each leaf node of T_X and T_Y is of size (a constant times) $1/M \times 1/M$. We denote the number of levels of T_X and T_Y by L .
2. *Initialization.* Set A to be the root of T_X . For each leaf box $B \in T_Y$, construct the expansion coefficients $\{\delta_t^{AB}, 1 \leq t \leq r_\epsilon\}$ from (10) by setting

$$\delta_t^{AB} = e^{-iM\Phi(x_0(A), y_t^B)} \sum_{y \in B} \left(L_t^B(y) e^{iM\Phi(x_0(A), y)} d(y) \right). \quad (13)$$

3. *Recursion.* For each $\ell = 1, 2, \dots, L/2$, construct the coefficients $\{\delta_t^{AB}, 1 \leq t \leq r_\epsilon\}$ for each pair (A, B) with A at level ℓ and B at the complementary level $L - \ell$ as follows: let A_p be A 's parent and $\{B_c, c = 1, 2, 3, 4\}$ be B 's children. For each child, we have available from the previous level an approximation of the form

$$m^{B_c}(x) \simeq \sum_{t'} e^{iM\Phi(x, y_{t'}^{B_c})} \delta_{t'}^{A_p B_c}, \quad \forall x \in A_p.$$

Summing over all children gives

$$m^B(x) \simeq \sum_c \sum_{t'} e^{iM\Phi(x, y_{t'}^{B_c})} \delta_{t'}^{A_p B_c}, \quad \forall x \in A_p.$$

Since $A \subset A_p$, this is also true for any $x \in A$. This means that we can treat $\{\delta_{t'}^{A_p B_c}\}$ as equivalent sources in B . As explained below, we then set the coefficients $\{\delta_t^{AB}\}_t$ as

$$\delta_t^{AB} = e^{-iM\Phi(x_0(A), y_t^B)} \sum_c \sum_{t'} L_t^B(y_{t'}^{B_c}) e^{iM\Phi(x_0(A), y_{t'}^{B_c})} \delta_{t'}^{A_p B_c}. \quad (14)$$

4. *Switch.* The interpolant in p may be used as the low-rank approximation as long as $\ell \leq L/2$ whereas the interpolant in x is a valid low-rank approximation as soon as $\ell \geq L/2$. Therefore, at $\ell = L/2$, we need to switch representation. Recall that for $\ell \leq L/2$ the expansion coefficients $\{\delta_t^{AB}, 1 \leq t \leq r_\epsilon\}$ may be regarded as equivalent sources while for $\ell \geq L/2$, they approximate the values of the potential $m^B(x)$ on the Chebyshev grid $\{x_t^A, 1 \leq t \leq r_\epsilon\}$. Hence, for any pair (A, B) with A at level $L/2$ (and likewise for B), we have $\delta_t^{AB} \simeq m^B(x_t^A)$ from (12) so that we may set

$$\delta_t^{AB} = \sum_s a(x_t^A, y_s^B) e^{iM\Phi(x_t^A, y_s^B)} \delta_s^{AB} \quad (15)$$

(we abuse notations here since $\{\delta_t^{AB}\}$ denotes the new set of coefficients and $\{\delta_s^{AB}\}$ the older set).

5. *Recursion (end).* The rest of the recursion is analogous. For $\ell = L/2 + 1, \dots, L$, construct the coefficients $\{\delta_t^{AB}, 1 \leq t \leq r_\epsilon\}$ as follows. With $\{\alpha_t^{AB}\}$ and $\{\beta_t^{AB}\}$ given by (11), we have

$$m^B(x) = \sum_c m^{B_c}(x) \simeq \sum_{t', c} \alpha_{t'}^{A_p B_c}(x) \sum_{p \in B_c} \beta_{t'}^{A_p B_c}(y) d(y) \simeq \sum_{t', c} \alpha_{t'}^{A_p B_c}(x) \delta_{t'}^{A_p B_c}.$$

Hence, since δ_t^{AB} should approximate $m^B(x_t^A)$ by (12), we simply set

$$\delta_t^{AB} = \sum_{t', c} \alpha_{t'}^{A_p B_c}(x_t^A) \delta_{t'}^{A_p B_c}.$$

Substituting α_t^{AB} with its value gives the update

$$\delta_t^{AB} = \sum_c e^{iM\Phi(x_t^A, y_0(B_c))} \sum_{t'} \left(L_{t'}^{A_p}(x_t^A) e^{-iM\Phi(x_{t'}^{A_p}, y_0(B_c))} \delta_{t'}^{A_p B_c} \right). \quad (16)$$

6. *Termination.* Finally, we reach $\ell = L$ and set B to be the root box of T_P . For each leaf box A of T_X , we have

$$m^B(x) \simeq \sum_t \alpha_t^{AB}(x) \delta_t^{AB}, \quad x \in A,$$

where $\{\alpha_t^{AB}\}$ is given by (11). Hence, for each $x \in A$, we set

$$m(x) = e^{iM\Phi(x, y_0(B))} \sum_t \left(L_t^A(x) e^{-iM\Phi(x_t^A, y_0(B))} \delta_t^{AB} \right). \quad (17)$$

Most of the computation is in (14) and (16). Because of the tensor product structures, the computations in (14) and (16) can be accelerated by performing Chebyshev interpolation one dimension at a time, reducing the number operations from $O(q^4)$ to $O(q^3)$, where q is the size of the Chebyshev grid in each dimension. As there are at most $O(M^2 \log M)$ pairs of boxes (A, B) , the recursion steps take at most $O(q^3 M^2 \log M)$ operations. The cost of (15) is of order $O(q^4 M^2)$ operations since for each pair (A, B) on the middle level a $q^2 \times q^2$ linear transformation is required. Hence, the overall complexity estimate of this algorithm $O(q^3 M^2 \log M + q^4 M^2) = O(q^3 N \log N + q^4 N)$. For brevity we bound this further as $O(q^4 N \log N)$. The value of q will be determined in Section 4. For the purpose of the rigorous error estimate it depends on M and $1/\epsilon$ logarithmically.

3 Numerical results

We present numerical experiments for two standard setups of synthetic aperture radar imaging: Stripmap SAR, where the plane goes along a straight line, and Spotlight SAR, where the plane goes around in a circle.

3.1 Stripmap SAR, synthetic data

Assume that the electromagnetic pulse is $P(\omega) = 1$, i.e., that no deconvolution is necessary. Let the acquisition manifold be

$$\Omega = [-\omega_2, -\omega_1] \cup [\omega_1, \omega_2] \quad \times \quad [s_1, s_2], \quad \text{with} \quad s_1 = 0, \quad s_2 = 1.$$

For the trajectory we let

$$\gamma(s) = (s, 0, H) \quad \Rightarrow \quad \dot{\gamma}(s) = (1, 0, 0).$$

With this choice, the range is $R = \|\gamma(s) - x_T\| = \sqrt{(s - x_1)^2 + x_2^2 + H^2}$. For the reference of the interested specialist, in the notations of [14] the Stolt variables are

$$\xi = \frac{2k}{R} \begin{pmatrix} s - x_1 \\ -x_2 \end{pmatrix},$$

and it is easily calculated that the Beylkin determinant is

$$\left| \frac{\partial \xi}{\partial(\omega, s)} \right| = \frac{4k|x_2|}{R^2}.$$

After properly taking into account all these contributions in the amplitude, the oscillatory integral becomes

$$\tilde{m}(\mathbf{x}) = 64\pi^2 |x_2| \int_{\Omega} e^{-2i\frac{\omega}{c} \sqrt{(s-x_1)^2 + x_2^2 + H^2}} d(\omega, s) d\omega ds.$$

This is a straight backprojection formula, without any filtering. The only slight subtlety is the domain of integration, a rectangle in (ω, s) space. Imaging artifacts will be reduced if an additional smooth indicator $\chi(\omega, s)$ equal to zero outside of Ω multiplies the integrand – as was done in the numerical experiment.

The results show that the complexity is consistently that of about a thousand 2D FFT (for an operation more complicated than a Fourier transform, of course), and the relative ℓ_2 error is consistently on the order of $2e-3$. The table below was obtained using $q = 5$ Chebyshev points per dimension, in each square in which interpolation is performed.

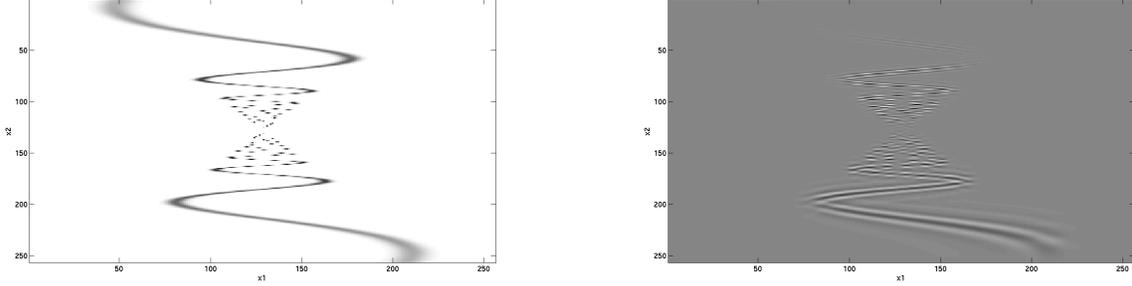


Figure 4: Left: synthetic 2D reflectivity model. Right: result of imaging from data limited by bandwidth and aperture. The linear flight track runs alongside the bottom side of the model, at an altitude equal to the maximal horizontal range of the reflectivity model.

\sqrt{N}	Time (s)	Speedup	Error
64	.5	5	2e-3
128	2.7	17	2e-3
256	14	61	2e-3
512	68	220	2e-3
1024	320	760	2e-3
2048	1500	2500	2e-3

Table 1. Comparisons are made against direct summation. Note that $\sqrt{N} = M$. Other error levels can be reached: runtimes are directly proportional to the desired number of correct digits.

3.2 Spotlight SAR, the Gotcha dataset

In this section we demonstrate the technique on the Air Force Research Laboratory’s publicly-released “Volumetric SAR Data Set, Version 1.0” [13]. The pulsed Circular SAR (CSAR) collection system used a 9.6 GHz center-frequency (X-band) Linearly Frequency Modulated (LFM) waveform with 640 MHz bandwidth, corresponding to a range resolution of 0.234 meters. The received data underwent the standard I/Q demodulation, digitization, and match-filtering processes described in [36]. A wideband horn antenna was used, thus the 3-dB antenna beampattern approximately satisfied the frequency-independence assumption discussed in Section 1.2. The antenna beampattern covered more than a one-kilometer radius of Wright-Patterson Air Force Base, but the phase history data were spatially lowpass filtered and downsampled [31] to include only a $100\text{m} \times 100\text{m}$ region containing calibration targets and carefully-placed civilian vehicles. Given the small size of the scene relative to the half-peak-power beampattern, spatial variation of the antenna beampattern within the scene can be neglected for these data.

The imaging operator we applied is as follows,

$$\mathcal{I}(i_1, i_2) = \sum_{(j_1, j_2) \in J} e^{i\Phi(i_1, i_2, j_1, j_2)} A(i_1, i_2, j_1, j_2) D(j_1, j_2), \quad (i_1, i_2) \in I, \quad (18)$$

where

- I, J are two subsets of $[0, 1]^2$,
- D is the phase history data, $D(j_1, j_2) = d(\omega(j_1), \gamma(j_2))$,
- the phase $\Phi(i_1, i_2, j_1, j_2) = \frac{2}{c}\omega(j_1) (\|\gamma(j_2) - \mathbf{x}(i_1, i_2)\| - r_0(j_2))$, and
- the amplitude $A(i_1, i_2, j_1, j_2) = \|\gamma(j_2) - \mathbf{x}(i_1, i_2)\|^2 w_1(j_1) w_2(j_2)$,

- γ the flight path in \mathbb{R}^3 ,
- ω spans the angular frequency range of the LFM chirp (in radians per second),
- r_0 the range to scene center from the antenna location (its presence is only due to some preprocessing that needs to be undone),
- and with w_1, w_2 numerical weights allowing for irregularly sampled data.

The butterfly algorithm needs to evaluate A, D and Φ at arbitrary points (j_1, j_2) in between sample points, so we interpolate the data d, γ, r_0, ω ; in these experiments we made use of the cubic spline interpolants of the GNU Scientific Library. The scene is defined by $\{\mathbf{x}(i_1, i_2) : (i_1, i_2) \in [0, 1]^2\}$, and may be a general surface in \mathbb{R}^3 parameterized by i_1, i_2 ; here we chose $\mathbf{x}(i_1, i_2) = ((b_1 - a_1)i_1 + a_1, (b_2 - a_2)i_2 + a_2, 0)$, with a_1, b_1, a_2, b_2 defining the boundary of the scene.

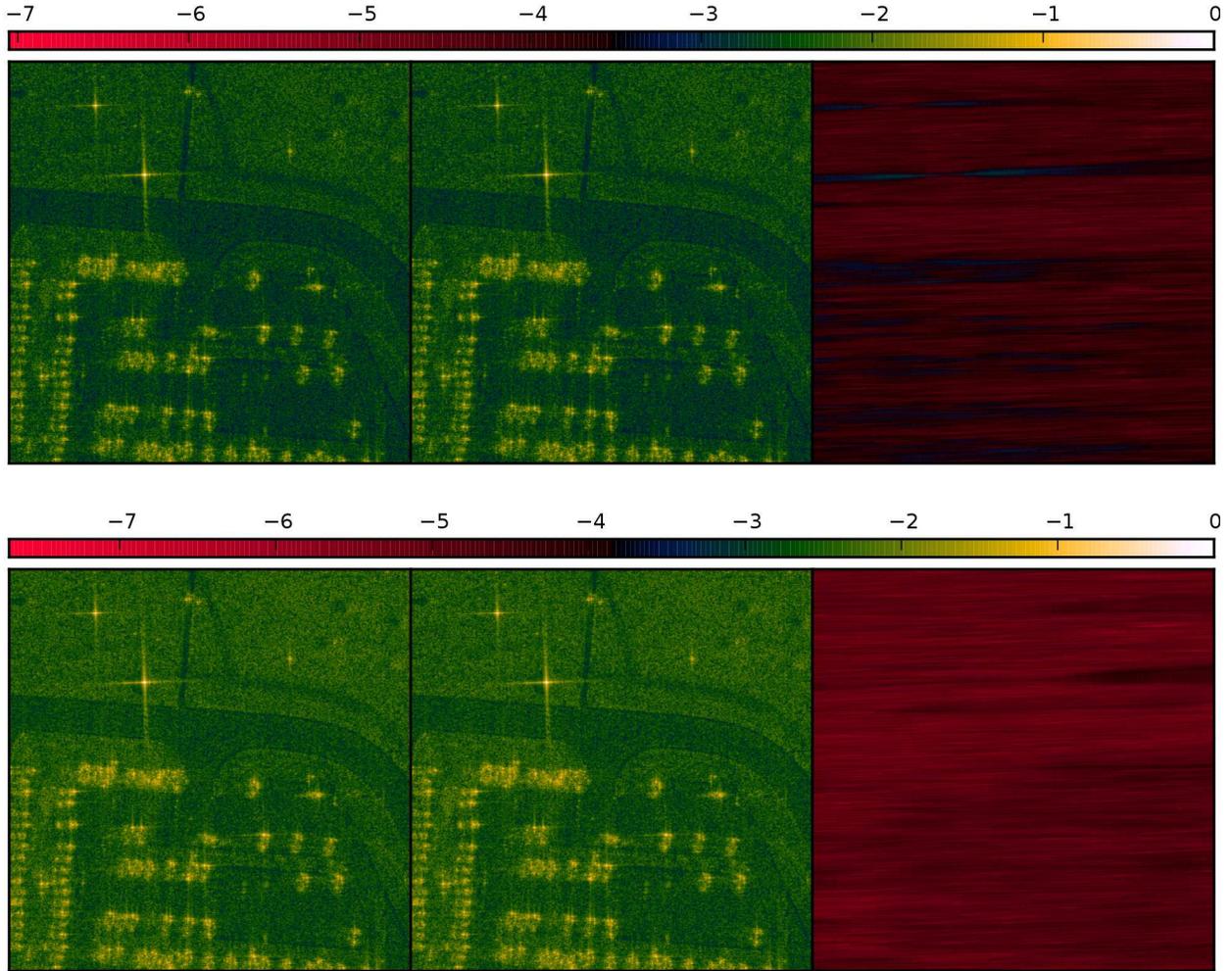


Figure 5: SAR images formed from the Gotcha data set, using the butterfly algorithm with $\sqrt{N} = 2^{10}$, and 4 degrees of azimuthal range, starting at 0 degrees, rendered on a logarithmic scale. Left images are of an exact summation, the center images are from the butterfly algorithm with $q = 4$ on top, and $q = 12$ on the bottom, and the right images are the differences between the two. All images are divided by the maximum norm of the exact image to set the scale.

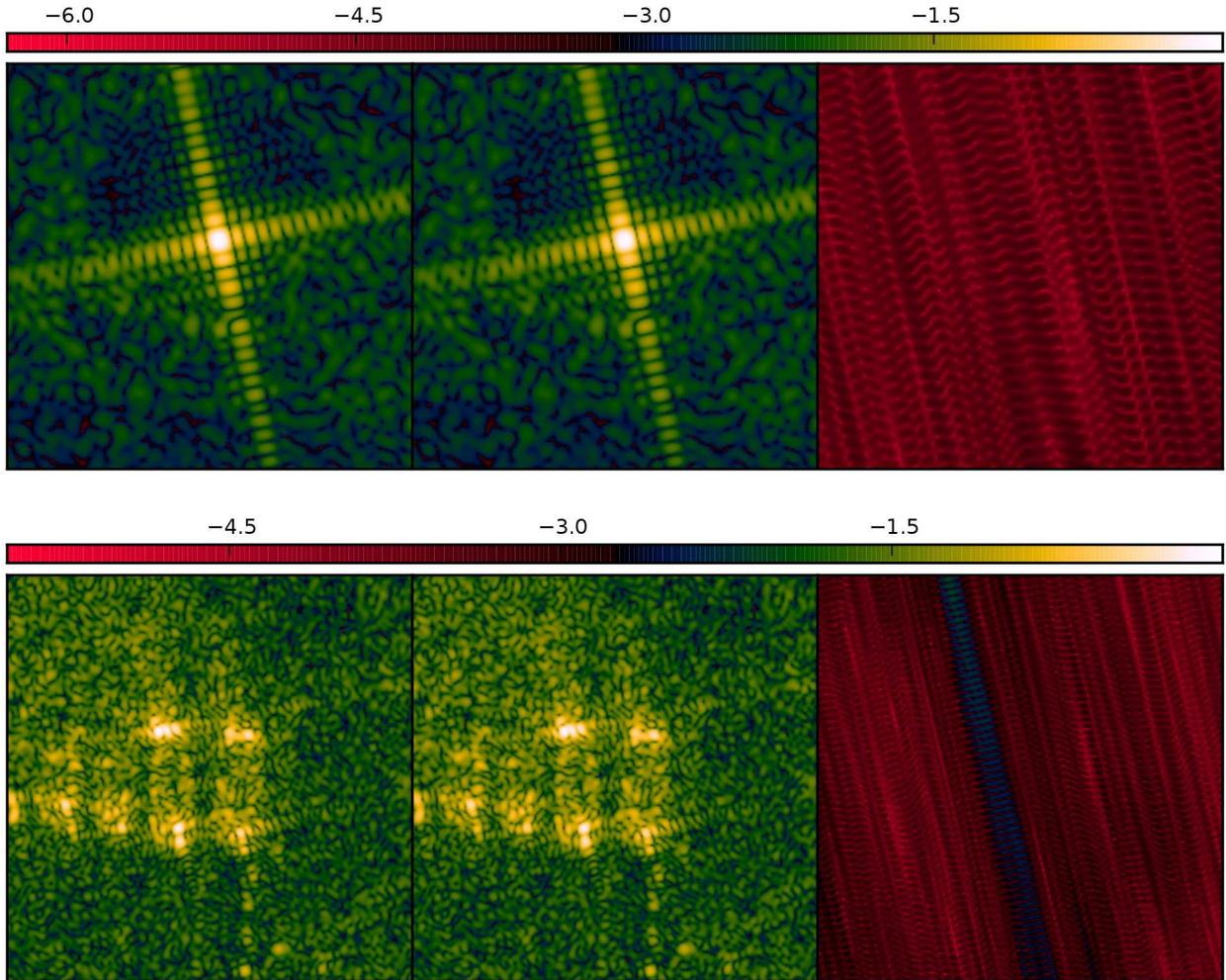


Figure 6: Zoom on features in the scene, here the azimuthal range is from 280 to 284 degrees, and $\sqrt{N} = 2^8$. The left images are the exact summation, center are the images from the butterfly algorithm, with $q = 7$, and the right images are the difference computed as before. Clearly, visual interpretation is not hampered by choosing $q = 7$ in these examples.

The total number of available frequency samples is $M_\omega = 426$. In the spotlight configuration, the position $\gamma(s)$ of the aircraft is indexed by the “azimuthal range” s . The latter is an angle: it is first partitioned into 360 1-degree angular sectors, and within each sector by 118 angular samples. All the images above are obtained by using only 4 contiguous sectors⁴. We let M_γ the total number of angular samples; in the example above, $M_\gamma = 472$. Note also that the Gotcha dataset contains data at 8 different altitudes and 4 different polarizations (HH, HV, VH, VV). Only one altitude and one polarization (HH) have been used for the tests in this paper, but it is clear that the technique can be extended to process the full dataset.

In accordance with the previous sections, the letter M is reserved for the fixed (large) quantity ω_0 . In units where the observed time span has length 1, M is (proportional to) a lower bound on the number of samples required for sampling the signal accurately. This suggest that M_ω and M_γ be at least of the same order of magnitude as M .

The current version of the butterfly algorithm handles square domains and boxes. A rectangular data space

⁴The quality of the image increases with more angular sectors. The speedup and relative RMS figures for the algorithm are essentially independent of the azimuthal range.

Y or image space X can easily be accommodated by stretching and resampling the dataset in the direction of least number of samples. This results in a total number of samples given by

$$N = (\max\{M_\omega, M_\gamma\})^2.$$

The two tuning parameters are q , the number of Chebyshev points per dimension per square, and L the depth of the tree. The slight oversampling mentioned above — when M_ω, M_γ are larger than M — alleviates the need for considering values of q greater than 10 or 15 in practice. L should be chosen in such a way that each leaf box in the data domain Y contains fewer than q^2 points, and such that the size of the leaf boxes in image space X matches the desired level of resolution. In our tests, we chose $L = \frac{1}{2} \log_2(N) - 6$.

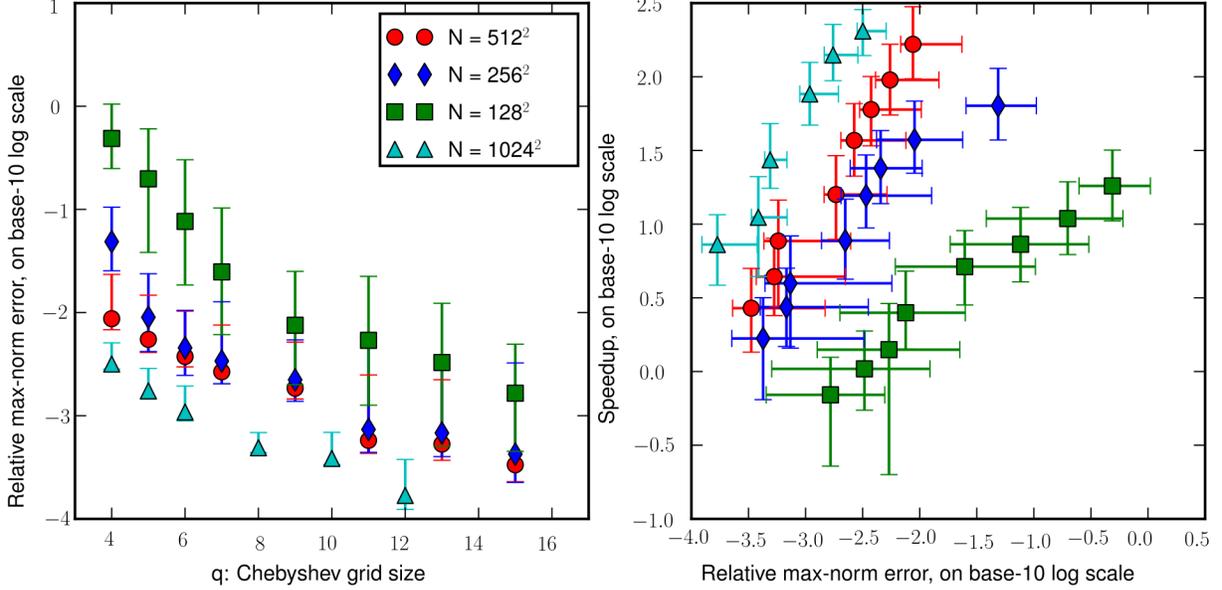


Figure 7: Left: plot of the relative max-norm-error as a function of q , the number of Chebyshev interpolation points per dimension in each butterfly box, at various values of N . Right: plot of the speedup as a function of the relative max-norm-error, speedup relative to exact sum. Each circle / diamond / square / triangle corresponds to a choice of q and N as earlier. The error bars arise from the fact that several images were computed, corresponding to different intervals in azimuthal angles, in the range $(0, 4, 8, \dots, 356)$ for $(N = 128, 256, 512)$, and $(0, 30, 60, \dots, 330)$ for $N = 1024$. The error bars were obtained from the minimum and maximum values in each case. It should be noted that these experiments were conducted on shared computational resources, thus we expect some variability in actual run time, and hence, speedup, depending on the variability of system load during the experiment.

Fast algorithms based on the FFT would make the far-field approximation. While we did not compare the butterfly directly to the unequally-spaced FFT in terms of speed, in the figure below we compute the error which would accompany the far-field approximation. This error is predicted to grow with the angular aperture. While the far-field image looks acceptable, the *pointwise relative error is orders of magnitude larger than that of butterfly truncation, even when only 4 degrees of azimuthal range are used*. Further numerical errors could come from the delicate spectral interpolation step in FFT-based processing.

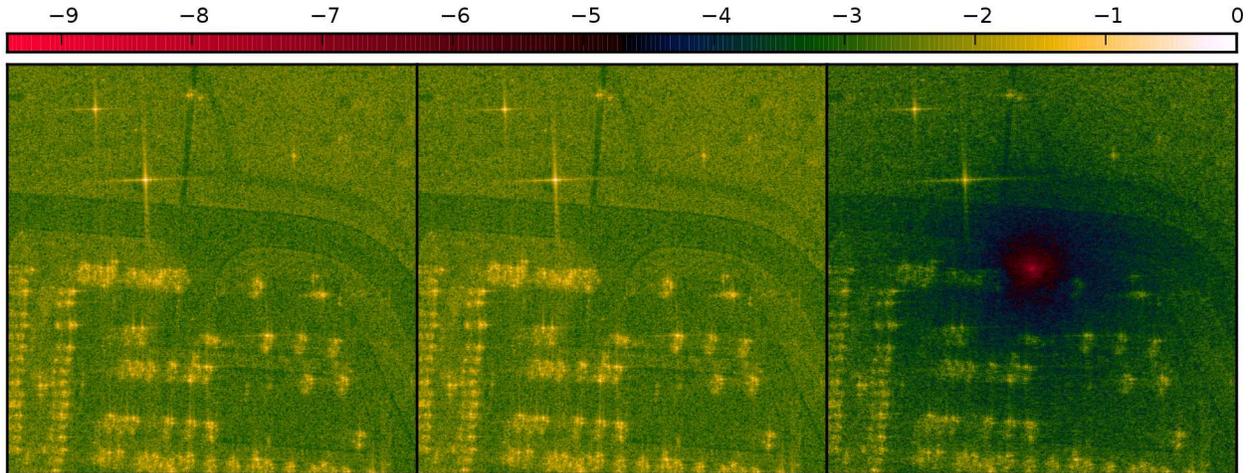


Figure 8: Left: image obtained by exact full summation, in log scale. Middle: full sum with the far-field approximation, i.e, that $\|\gamma(j_2) - \mathbf{x}(i_1, i_2)\| \approx \|\gamma(j_2)\| - \mathbf{x}(i_1, i_2) \cdot \gamma(j_2) / \|\gamma(j_2)\|$ (relative to the scene center, which is 0,) again in log scale. Right: absolute value of the difference of the moduli of the two images, in log scale, normalized with respect to the maximum of the exact image. The relative error in maximum norm between the left and middle images is 26.92 percent. The modulus was taken before subtracting the images because the phase errors would be even larger than the modulus errors from taking the far-field approximation. Note the error drops to zero in the center, where the ratio $\|x\|/\|\gamma\|$ goes to zero. See the next figure for a quantitative analysis.

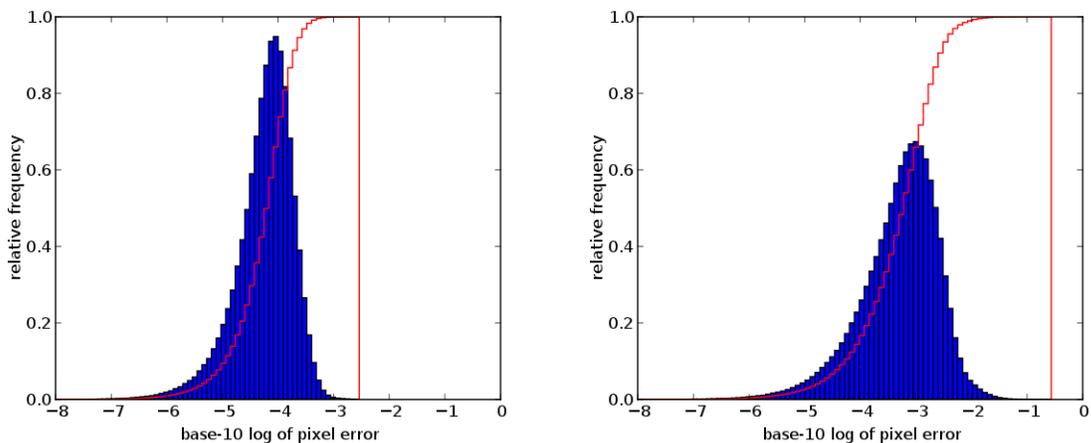


Figure 9: Histograms of pixel values in the two difference images, butterfly algorithm on the left, far-field approximation on the right. By “difference”, we mean difference of the moduli as in the previous figure. We can see that the relative max-norm of the far-field difference image is very high, 0.2692 (admittedly an outlier.) We also compute the median errors as $6.31e - 4$ for the far-field approximation, and $6.31e - 5$ for the butterfly algorithm, with $q = 4$.

4 Proof of the error bound

In this section we prove Theorem 1. Although this may not be stated explicitly, the letter C refers to constants that may depend on unessential parameters (clear from context), and which may change from line

to line.

4.1 Preliminaries on analyticity and interpolation

The analyticity assumption on the amplitude and phase functions translates into fast converging Chebyshev interpolation. The following result will not surprise specialists, but we were unable to find in the literature; its proof is in the appendix. Throughout this paper we use the notation

$$\|f\|^2 = \int_{-1}^1 |f(x)|^2 \frac{dx}{\sqrt{1-x^2}}.$$

Theorem 2. *Let f be (Q, R) -analytic on $[-1, 1]$:*

$$|f^{(n)}(x)| \leq Q n! R^{-n}.$$

Denote by p the N -point Chebyshev interpolant of f on $[-1, 1]$. Assume $N \geq 1/(2R)$. Then

$$\|f - p\| \leq C Q N \left[1 + \frac{1}{R^2}\right]^{1/4} \left[R + \sqrt{R^2 + 1}\right]^{-N}, \quad (19)$$

for some numerical constant $C > 0$.

A fortiori, the result is also true for the weaker L^2 norm $\|f\|_2 = (\int |f(x)|^2 dx)^{1/2}$. The proof involves analytic continuation of f in the complex plane, and contour deformation to bound (i) the coefficients of the Chebyshev expansion of f , and (ii) the difference between those coefficients and the approximate ones obtained from the function samples. The following corollary treats the case of the uniform norm.

Corollary 3. *In the setting of Theorem 2,*

$$\|f - p\|_\infty \leq C Q N^2 \left[1 + \frac{1}{R^2}\right]^{1/4} \left[R + \sqrt{R^2 + 1}\right]^{-N}, \quad (20)$$

where the constant C is possibly different from that of Theorem 2.

Proof. Use the fundamental theorem of calculus and the Cauchy-Schwarz inequality to write the elementary relation

$$|f(x) - f(x^*)|^2 \leq \left[\int_{-1}^1 \frac{dx}{\sqrt{1-x^2}} \right] \int_{-1}^1 |f'(x)|^2 \sqrt{1-x^2} dx,$$

valid for all $x, x^* \in [-1, 1]$. The factor in bracket has value π .

Denote by $\varepsilon_2(N; Q, R)$ the right-hand side of (19). It is easily established (for instance by contradiction) that there exists $x^* \in [-1, 1]$ for which $|f(x^*) - p(x^*)| \leq \varepsilon_2(N; Q, R)$, where p is the N -term Chebyshev interpolant of f . For any other $x \in [-1, 1]$, we have

$$|f(x) - p(x)| \leq |f(x^*) - p(x^*)| + \pi \int_{-1}^1 |f'(x) - p'(x)|^2 \sqrt{1-x^2} dx.$$

A reasoning entirely parallel to that of the proof of Theorem 2 can be made to show that the integral term is no bigger than $C N \varepsilon_2(N; Q, R)$, which proves the claim. Indeed,

$$\int_{-1}^1 |f'(x) - p'(x)|^2 \sqrt{1-x^2} dx = \int_0^\pi \left| \frac{d}{d\theta} (f(\cos \theta) - p(\cos \theta)) \right|^2 d\theta,$$

and differentiation in the θ domain amounts to multiplication by in of the n -th Fourier series coefficient. This extra factor plays a role in equation (38) for the aliasing error, where the sum over n now takes the form

$$\sum_{|n| \leq N}'' n^2 \rho^{2n},$$

which leads to an extra N^2 in the bound. As for the truncation error, it now reads

$$\sum_{|n|>N} n^2 \rho^{-2n} = \frac{\rho}{4} \frac{d}{d\rho} \rho \frac{d}{d\rho} \frac{\rho^{-2N}}{1 - \rho^{-2}},$$

where the two differentiations also lead to an extra factor N^2 . A square root is later taken to turn these N^2 factors into N . \square

The following simple lemmas record the behavior of analytic functions under exponentiation, multiplication, and subtraction respectively.

Lemma 1. *Let $f(x)$ be (Q, R) -analytic on some domain Ω . Then, on Ω , $g(x) = e^{iMf(x)}$ obeys*

$$|g^{(n)}(x)| \leq \begin{cases} 3\sqrt{n} M^n \left(\frac{\epsilon Q}{R}\right)^n & \text{if } 1 \leq n < \frac{MQ}{\sqrt{2}}; \\ e^{\sqrt{2}NQ} n! \left(\frac{\sqrt{2}}{R}\right)^n & \text{if } n \geq \frac{NQ}{\sqrt{2}}. \end{cases}$$

Proof. Since f is (Q, R) -analytic, it extends holomorphically into the strip $\Omega \times (-R, R) \subset \mathbb{C}$. We seek to bound the holomorphic extension of g in the same strip. To this end, notice that

$$|g(z)| = \exp(-M \operatorname{Im} f(z)),$$

so it suffices to consider the imaginary part of $f(z)$. For any $z = x + iy$ we have the Taylor series

$$f(z) = \sum_{n \geq 0} \frac{f^{(n)}(x)}{n!} (iy)^n,$$

so that

$$\operatorname{Im} f(z) = \sum_{n \geq 0} \frac{f^{(2n+1)}(x)}{(2n+1)!} (-1)^n y^{2n+1}.$$

We may use (Q, R) -analyticity of f to bound

$$|\operatorname{Im} f(z)| \leq Q \sum_{n \geq 0} (-1)^n \left(\frac{y}{R}\right)^{2n+1} = \frac{Qy}{R} \frac{1}{1 - \frac{y^2}{R^2}}.$$

It results that, for $z = x + iy \in \Omega \times (-R, R)$,

$$|g(z)| \leq \exp\left(\frac{MQy}{R} \frac{1}{1 - \frac{y^2}{R^2}}\right).$$

We now apply Cauchy's criterion in the region $\Omega \times (-y, y)$, with y as yet unspecified, to obtain

$$|g^{(n)}(x)| \leq \max_{|\operatorname{Im} z| \leq y} |g(z)| n! y^{-n}.$$

As long as $y < R/\sqrt{2}$, a good choice is $y = \frac{nR}{MQ}$. In terms of n this choice is valid provided $n < MQ/\sqrt{2}$. Then the bound simplifies to

$$|g^{(n)}(x)| \leq e^{2n} n! \left(\frac{nR}{MQ}\right)^{-n}.$$

In view of the Stirling bound $n! \leq 3\sqrt{n} n^n e^{-n}$ for $n \geq 1$, we get

$$|g^{(n)}(x)| \leq 3\sqrt{n} M^n \left(\frac{\epsilon Q}{R}\right)^n.$$

If on the other hand $n \geq MQ/\sqrt{2}$, we use $y = R/\sqrt{2}$ and check that the required bound holds. \square

Lemma 2. Let $f(x)$ be (Q, R) -analytic on some domain, and g be essentially bandlimited in the sense that

$$|g^{(n)}(x)| \leq A B^n n^m.$$

Then fg is (Q', R) -analytic, with

$$Q' = AQ (RB)^m e^{RB}.$$

Proof. It suffices to apply the high-order Leibniz rule,

$$(fg)^{(n)} = \sum_{k=0}^n \binom{n}{k} f^{(n-k)} g^{(k)},$$

to get

$$\begin{aligned} |(fg)^{(n)}| &\leq \sum_{k=0}^n \binom{n}{k} Q (n-k)! R^{-(n-k)} A B^k k^m, \\ &= QA n! R^{-n} \sum_{k=0}^n \frac{(RB)^k}{k!} k^m \\ &\leq QA (RB)^m e^{RB} n! R^{-n}. \end{aligned}$$

□

This latter bound is rather coarse, particularly if R is large. However Q' is as good a constant as Q for expressing large- n asymptotics.

Lemma 3. Assume $\phi(x, y)$ is (Q, R_1) -analytic in x , and (Q, R_2) -analytic in y : for all unit-length d_1, d_2 ,

$$|(d_1 \cdot \nabla_x)^{n_1} (d_2 \cdot \nabla_y)^{n_2} \phi(x, y)| \leq Q n_1! n_2! R_1^{-n_1} R_2^{-n_2}.$$

Then $\phi(x_0, y) - \phi(x_1, y)$ is (Q', R_2) -analytic in the y variable, with

$$Q' = Q \frac{|x_0 - x_1|/R_1}{1 - |x_0 - x_1|/R_1}. \quad (21)$$

Proof. Perform a Taylor expansion along the line joining x_0 and x_1 . Let $d_1 = \widehat{x_0 - x_1}$.

$$\begin{aligned} |(d_2 \cdot \nabla_y)^{n_2} (\phi(x_0, y) - \phi(x_1, y))| &= \left| \sum_{n_1 \geq 1} \frac{(d_1 \cdot (x_0 - x_1))^{n_1}}{n_1!} (d_1 \cdot \nabla_x)^{n_1} (d_2 \cdot \nabla_y)^{n_2} \phi(x_1, y) \right| \\ &\leq \sum_{n_1 \geq 1} |x_0 - x_1|^{n_1} Q n_1! R_1^{-n_1} R_2^{-n_2} \\ &\leq Q n_2! R_2^{-n_2} \frac{|x_0 - x_1|/R_1}{1 - |x_0 - x_1|/R_1}. \end{aligned}$$

□

4.2 Recursive dyadic interpolation

In this section we show how the butterfly algorithm in effect performs repeated interpolation of the kernel $K(x, y)$ at different scales.

In the version of the butterfly algorithm presented here, a “switch” occurs midway through the tree traversals. A subscript m will be used to denote any of the boxes A_m or B_m at this midlevel. A couple of boxes (A, B) is said to be located before the switch if there exist midlevel boxes A_m, B_m such that $A \supset A_m$ and $B \subset B_m$. Conversely, (A, B) is said to be located after the switch if there exist midlevel boxes A_m, B_m such that $A \subset A_m$ and $B \supset B_m$. Exactly at the switch, a couple (A_m, B_m) is said to occur either before or after the switch depending on whether the δ_t^{AB} have already been subjected to the switch transformation or not.

When referring to the “midlevel box B that contains the point y ”, we will write $B_m(y)$. Analogous definition for $A_m(x)$.

Boxes at the leaves of the A or B trees will be denoted A_ℓ, B_ℓ respectively. When referring to the “leaf level box B that contains the point y ”, we will write $B_\ell(y)$. Analogous definition for $A_\ell(x)$. Similarly, the index r denotes root boxes. And as previously, we also use the indices c to denote a child box, and p to denote a parent box.

Fix a couple of boxes (A, B) . It was explained earlier that the box B carries an interpolation structure over points y_t^B with basis functions β_t^{AB} , that we denote

$$K_B(x, y) = \sum_t K(x, y_t^B) \beta_t^{AB}(y), \quad x \in A, \quad y \in B.$$

Similarly, interpolation can be performed in the x variable and leads to

$$K^A(x, y) = \sum_t \alpha_t^{AB}(x) K(x_t^A, y), \quad x \in A, \quad y \in B.$$

If interpolation is performed both in the x and y variables, we use the notation $K_B^A(x, y)$.

The *recursive dyadic interpolant* of K from some box B down to some descendant box b results from repeatedly interpolating K in the boxes $B, B_c, B_{cc} \dots, b$ along the tree from B down to b . It is denoted

$$(K_{B \searrow b})(x, y) = \sum_{t_n} \dots \left[\sum_{t_2} \left[\sum_{t_1} K(x, y_{t_1}^B) \beta_{t_1}^{AB}(y_{t_2}^{B_c}) \right] \beta_{t_2}^{A_p B_c}(y_{t_3}^{B_{cc}}) \right] \dots \beta_{t_n}^{ab}(y), \quad x \in A,$$

where a is the (unique) parent of A at a level compatible with b . Analogously,

$$(K^{A \searrow a})(x, y) = \sum_{t_n} \dots \left[\sum_{t_2} \left[\sum_{t_1} K(x_{t_1}^A, y) \alpha_{t_1}^{AB}(x_{t_2}^{A_c}) \right] \alpha_{t_2}^{A_c B_p}(x_{t_3}^{A_{cc}}) \right] \dots \alpha_{t_n}^{ab}(x), \quad x \in A,$$

where b is the (unique) parent of B at a level compatible with a . If recursive interpolation is done in both x and y from (A, B) down to (a, b) , we use the notation

$$\left(K_{B \searrow b}^{A \searrow a} \right) (x, y).$$

(Note that a and b are not necessarily compatible in this last equation.) If $B = b$ we simply have $K_{B \searrow B} = K_B$, similarly $K^{A \searrow A} = K^A$ and $K_{B \searrow B}^{A \searrow A} = K_B^A$. If on the other hand $b \subsetneq B$ or $a \subsetneq A$, by convention no interpolation takes place.

The following lemma quantifies how the low-rank partial sum $\tilde{u}_B(x)$ obtained from the butterfly algorithm deviates from the true partial sum $m_B(x) = \sum_{y_j \in B} K(x, y_j) d_j$.

Lemma 4. *Consider a couple of boxes (A, B) and the corresponding butterfly data δ_t^{AB} .*

If (A, B) is located before the switch, define

$$\tilde{m}_B(x) = \sum_t K(x, y_t^B) \delta_t^{AB}, \quad x \in A.$$

It holds that

$$\tilde{m}_B(x) = \sum_{y_j \in B} (K_{B \searrow B_\ell(y_j)})(x, y_j) d_j, \quad x \in A. \quad (22)$$

If (A, B) is located after the switch, define instead

$$\tilde{m}_B(x) = \sum_t \alpha_t^{AB}(x) \delta_t^{AB}, \quad x \in A.$$

It holds that

$$\tilde{m}_B(x) = \sum_{y_j \in B} \left(K_{B_m(y_j) \searrow B_\ell(y_j)}^{A_m(x) \searrow A} \right) (x, y_j) d_j, \quad x \in A. \quad (23)$$

Proof. The proof is split into three cases: before, at, and after the switch.

- Fix a couple (A, B) located before the switch. Consider B as the root of its subtree, consisting of all its descendants b . Index these descendants by level $k = \ell(b)$ from $k = 0$ (the level of B) to $k = n$ (the leaf level). We wish to prove by induction on k increasing from 0 to n the claim that

$$\tilde{m}_B(x) = \sum_{b:\ell(b)=k} \sum_t K_{B \setminus b_p}(x, y_t^b) \delta_t^{ab}, \quad (24)$$

for $x \in A$, and a is the (unique) parent of A at a level compatible with b such that $\ell(b) = k$. For $k = 0$ no interpolation takes place and the claim is obviously true by definition of $\tilde{u}_B(x)$. Assume now that the claim is true at level k . For each b with $\ell(b) = k$ have

$$\delta_t^{ab} = \sum_c \sum_{t'} \beta_t^{ab}(y_{t'}^{b_c}) \delta_{t'}^{a_p b_c}.$$

So, for all $x \in A$,

$$\begin{aligned} \tilde{m}_B(x) &= \sum_{b:\ell(b)=k} \sum_t K_{B \setminus b_p}(x, y_t^b) \left[\sum_c \sum_{t'} \beta_t^{ab}(y_{t'}^{b_c}) \delta_{t'}^{a_p b_c} \right] \\ &= \sum_{b:\ell(b)=k} \sum_c \sum_{t'} \left[\sum_t K_{B \setminus b_p}(x, y_t^b) \beta_t^{ab}(y_{t'}^{b_c}) \right] \delta_{t'}^{a_p b_c}. \end{aligned}$$

We are in presence of the interpolant of $K_{B \setminus b_p}$ in the box b , which by definition is called $K_{B \setminus b}$. Hence

$$\tilde{m}_B(x) = \sum_{b:\ell(b)=k} \sum_c \sum_{t'} K_{B \setminus b}(x, y_{t'}^{b_c}) \delta_{t'}^{a_p b_c}.$$

Relabeling $b \rightarrow b_p$, $b_c \rightarrow b$, and noticing that $\sum_{b:\ell(b)=k} \sum_c \rightarrow \sum_{b:\ell(b)=k+1}$, we obtain the claim (24).

As $k = n$, the interaction coefficients are

$$\delta_t^{A_r B_\ell} = \sum_{y_j \in B_\ell} \beta^{A_r B_\ell}(y_j) d_j,$$

so equation (24) becomes

$$\begin{aligned} \tilde{m}_B(x) &= \sum_{B_\ell} \sum_t K_{B \setminus B_{\ell,p}}(x, y_t^{B_\ell}) \left[\sum_{y_j \in B_\ell} \beta^{A_r B_\ell}(y_j) d_j \right], \\ &= \sum_{B_\ell} \sum_{y_j \in B_\ell} K_{B \setminus B_\ell}(x, y_j) d_j. \end{aligned}$$

The latter equation is exactly (22).

- Notations change at this point: the pre-switch δ coefficients are now denoted $\tilde{\delta}_t^{AB}$, and the post-switch coefficients simply δ_t^{AB} . They are related by

$$\delta_t^{AB} = \sum_{t'} K(x_t^A, y_{t'}^B) \tilde{\delta}_{t'}^{AB}.$$

This allows to write

$$\begin{aligned} \tilde{u}_B(x) &= \sum_t \alpha_t^{AB}(x) \left[\sum_{t'} K(x_t^A, y_{t'}^B) \tilde{\delta}_{t'}^{AB} \right] \\ &= \sum_{t'} K^A(x, y_{t'}^B) \tilde{\delta}_{t'}^{AB}. \end{aligned}$$

Except for interpolation in the x variable, we are in presence of the pre-switch formula for \tilde{u}_B . Invoking (22), we get

$$\tilde{m}_B(x) = \sum_{y_j \in B} \left(K_{B \setminus B_\ell(y_j)}^A \right) (x, y_j) d_j, \quad x \in A.$$

- It now remains to check that (23) holds by induction down the rest of the A tree. We have just shown that it holds immediately after the switch. The formula for the δ updates after the switch is

$$\delta_t^{AB} = \sum_c \sum_{t'} \alpha_{t'}^{A_p B_c} (x_t^A) \delta_{t'}^{A_p B_c}.$$

The induction assumption can be used directly in this formula: we use (23) for the pair (A_p, B_c) , and evaluated at $x = x_t^A$:

$$\delta_t^{AB} = \sum_c \sum_{y_j \in B_c} \left(K_{B_m(y_j) \setminus B_\ell(y_j)}^{A_m(x) \setminus A_p} \right) (x_t^A, y_j) d_j.$$

Of course $\sum_c \sum_{y_j \in B_c}$ is just $\sum_{y_j \in B}$, so

$$\begin{aligned} \tilde{m}_B(x) &= \sum_t \alpha_t^{AB} (x) \left[\sum_{y_j \in B} \left(K_{B_m(y_j) \setminus B_\ell(y_j)}^{A_m(x) \setminus A_p} \right) (x_t^A, y_j) d_j \right] \\ &= \sum_{y_j \in B} \left[\sum_t \alpha_t^{AB} (x) \left(K_{B_m(y_j) \setminus B_\ell(y_j)}^{A_m(x) \setminus A_p} \right) (x_t^A, y_j) \right] d_j \end{aligned}$$

In the last line, the quantity in square brackets is the interpolant of $K_{B_m(y_j) \setminus B_\ell(y_j)}^{A_m(x) \setminus A_p}$ on the grid x_t^A of the box A . By definition this is just $K_{B_m(y_j) \setminus B_\ell(y_j)}^{A_m(x) \setminus A}$. We deduce (23), as desired. □

Notice that all the interpolations are done *from the top down*, both on the A and B sides, even though the traversals are respectively top-down and bottom-up. This fact is significant because only top-down recursive interpolation is numerically stable.

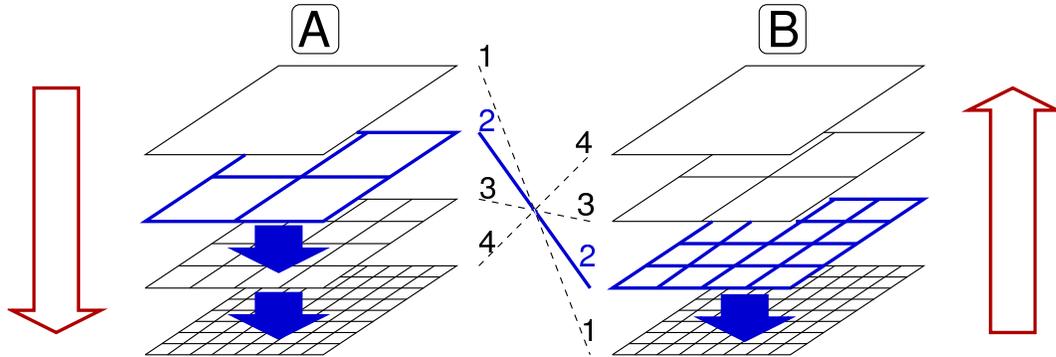


Figure 10: This is a color picture. In red (hollow arrows), direction of traversal of the quadtrees. In blue (solid arrows), direction in which recursive interpolation is implicitly performed by the butterfly algorithm. The blue squares indicate the middle level (level 2), where the switch from interpolation in the y variable to interpolation in the x variable is performed.

If a straight polynomial interpolant were used in the butterfly scheme, *recursive interpolation would be error-free* since the degree- n polynomial interpolant of a degree- n polynomial is itself. Because oscillations are factored in the interpolant, this is however not the case and a full error propagation analysis must be carried out.

4.3 Error propagation analysis

In this section we finish the proof of Theorem 1. The analysis of the previous section is very general in the sense that it holds for arbitrary interpolation kernels α_t^{AB} and β_t^{AB} . Here we specialize $K(x, y) = a(x, y)e^{iM\Phi(x, y)}$ and choose the interpolation kernels as in Section 2:

$$\begin{aligned}\beta_t^{AB}(y) &= e^{-iM\Phi(x_0(A), y_t^B)} L_t^B(y) e^{iM\Phi(x_0(A), y)}, & \text{pre-switch,} \\ \alpha_t^{AB}(y) &= e^{iM\Phi(x, y_0(B))} L_t^A(x) e^{-iM\Phi(x_t^A, y_0(B))}, & \text{post-switch.}\end{aligned}$$

Here L_t^A and L_t^B are the respective elementary Lagrange interpolation polynomials in the boxes A and B . In each box A or B , the polynomial interpolation takes place over q^2 tensor-Chebyshev points.

Corollary (3) will later be applied; let us for now denote the right-hand side of equation (20) as

$$\varepsilon_\infty(N; Q, R) = C Q N^2 \left[1 + \frac{1}{R^2} \right]^{1/4} \left[R + \sqrt{R^2 + 1} \right]^{-N}, \quad (25)$$

where $C > 0$ is adequately large (and N is here any positive integer).

Consider a box B below the switch, and one of its descendants b located k levels lower than B . Let us prove by induction on k from $k = 0$ ($b = B$) to $k = n$ ($b = B_\ell$), that there exists some $C > 0$ such that

$$\|K_{B \setminus b}(x, \cdot) - K(x, \cdot)\|_\infty \leq 2\varepsilon_\infty(q; Q'', R) \cdot (C \log q)^k, \quad x \in A, \quad y \in B, \quad (26)$$

where A is compatible with B , where ε_∞ was defined in (25), and where Q'' is some function of Q and R .

- As $b = B$, we get

$$\begin{aligned}K_B(x, y) &= e^{iM\Phi(x_0(A), y)} \sum_t L_t^B(y) e^{-iM\Phi(x_0(A), y_t^B)} K(x, y_t^B) \\ &= e^{iM\Phi(x_0(A), y)} \sum_t L_t^B(y) a(x, y_t^B) e^{iM(\Phi(x, y_t^B) - \Phi(x_0(A), y_t^B))}.\end{aligned}$$

The difference $|K_B - K|$ is therefore controlled by the error in interpolating the function $a(x, y)e^{iM(\Phi(x, y) - \Phi(x_0(A), y))}$ over $y \in B$, while keeping $x \in A$. All the tools are in place:

- First, remember that both $a(x, y)$ and $\Phi(x, y)$ are (Q, R) -analytic in the y variable. The interpolation takes place in a square of sidelength $\text{diam}(B)/\sqrt{2}$ in y , so we may rescale the problem to $[-1, 1]^2$ by assuming that R is replaced by $R' = 2\sqrt{2}R/\text{diam}B$.
- Second, remember that $\Phi(x, y)$ is also (Q, R) -analytic in the x variable. Assume for a moment that $\text{diam}(A) \leq R$: this implies that the denominator in equation (21) is greater than $1/2$, so we can invoke Lemma 3. We obtain that the difference $\Phi(x, y) - \Phi(x_0(A), y)$ is (Q', R') -analytic in the y variable, with the same R' as $\Phi(x, y)$ but a potentially smaller $Q' \leq 2Q \text{diam}(A)/R$. If on the other hand $\text{diam}(A) > R$, we may use the trivial bound $Q' \leq 2Q$, hence a fortiori $Q' \leq 2Q \text{diam}(A)/R$ as well.
- Third, Lemma 1 asserts that taking the exponential of an analytic function results in an essentially bandlimited function. Putting together the expressions of Q' and R' obtained so far, we find the bound

$$|(d \cdot \nabla_y)^n e^{iM(\Phi(x, y) - \Phi(x_0(A), y))}| \leq 3\sqrt{n} M^n [c_{Q, R} \text{diam}(A) \text{diam}(B)]^n,$$

where $c_{Q, R}$ is some function of Q and R – constant with respect to M , $\text{diam}(A)$ and $\text{diam}(B)$ – and n is not extremely large, $n < MQ/\sqrt{2}$. For $n \geq MQ/\sqrt{2}$ the bound reverts to a factorial: we leave to the reader the easy but tedious task of checking that this detail does not change the conclusions of the argument. The important point is that the essential band limit is a constant times $M \times \text{diam}(A) \times \text{diam}(B)$. The crux of the proof is that *this latter quantity is precisely the algorithmic Fresnel number, assumed to be less than 1.*

- Fourth, we can now invoke Lemma 2 to handle the multiplication of the oscillatory factor with the amplitude. We may disregard the fact that R has been rescaled to R' , and simply assume that $a(x, \cdot)$ is (Q, R) -analytic. In a rescaled variable $y \in [-1, 1]^2$, we obtain the final estimate

$$|\partial_y^n a(x, y) e^{iM(\Phi(x, y) - \Phi(x_0(A), y))}| \leq Q'' n! R^{-n}, \quad (27)$$

with $Q'' = 3Q(Rc_{Q, R}) e^{Rc_{Q, R}}$.

Equation (27) provides the smoothness estimate necessary to invoke Corollary 3. It is applied first in the y_1 variable, then in the y_2 variable. Passing from one dimension to two dimensions doubles the error bound but squares the number of coefficients. With q^2 bivariate Chebyshev polynomials, therefore, Corollary 3 guarantees that the function $a(x, y) e^{iM(\Phi(x, y) - \Phi(x_0(A), y))}$ is interpolated in y with an error

$$2\varepsilon_\infty(q; Q'', R),$$

in the uniform norm, where $\varepsilon_\infty(q; Q'', R)$ comes from equation (25). This settles the base case $k = 0$.

- For the general case of the induction, let a be compatible with b . We have

$$K_{B \setminus b}(x, y) = e^{iM\Phi(x_0(a), y)} \sum_t L_t^b(y) e^{-iM\Phi(x_0(a), y_t^b)} K_{B \setminus b_p}(x, y_t^b) \quad (28)$$

In the right-hand side, split $K_{B \setminus b_p}(x, y) = K(x, y) - (K(x, y) - K_{B \setminus b_p}(x, y))$. Subtract $K(x, y)$ from the whole equation, and consider two contributions.

- First, if K takes the place of $K_{B \setminus b_p}$ in (28), then the latter reduces to

$$K_b(x, y) = e^{iM\Phi(x_0(a), y)} \sum_t L_t^b(y) e^{-iM\Phi(x_0(a), y_t^b)} K(x, y_t^b).$$

Upon subtracting $K(x, y)$ we are left to consider $K_b(x, y) - K(x, y)$, which we have already encountered earlier. Independently of b in the range considered,

$$\|K_b(x, \cdot) - K(x, \cdot)\|_\infty \leq 2\varepsilon_\infty(q; Q'', R).$$

- Second, we are led to consider

$$e^{iM\Phi(x_0(a), y)} \sum_t L_t^b(y) e^{-iM\Phi(x_0(a), y_t^b)} [K(x, y_t^b) - K_{B \setminus b_p}(x, y_t^b)].$$

By assumption the term in square brackets is bounded by $2\varepsilon_\infty(q; Q'', R)(C \log q)^{k-1}$. The oscillatory factor $e^{-iM\Phi(x_0(a), y_t^b)}$ does not change the modulus of this quantity. The interpolation operation may increase the ℓ_∞ norm by a factor $C \log q$, as is well-known for Chebyshev interpolation [6, 12, 32]. Finally, the oscillatory factor $e^{iM\Phi(x_0(a), y)}$ does not change the modulus of the result. We are left with

$$\|K(x, \cdot) - K_{B \setminus b}(x, \cdot)\|_\infty \leq C \log q \cdot \|K(x, \cdot) - K_{B \setminus b_p}(x, \cdot)\|_\infty \leq 2\varepsilon_\infty(q; Q'', R)(C \log q)^k.$$

This concludes the induction argument for interpolation in the y variable, and proves equation (26).

The interpolation problem is completely symmetrical in the x variable: the argument can be repeated and yields an error of the same order of magnitude. Since there are $n = C \log_2 N$ levels between the midlevel nodes and the leaf nodes, for some (very small) constant C , we get an overall error of

$$\| \left(K_{B_m \setminus B_\ell}^{A_m \setminus A_\ell} \right) - K \|_\infty \leq 4\varepsilon_\infty(q; Q'', R)(C \log q)^{C \log N},$$

where A_ℓ is any descendent of A_m , and B_ℓ is any descendent of B_m . If we let $\rho = R + \sqrt{R^2 + 1}$, and use (25), the right-hand side takes the form

$$C(Q, R) q^2 (C \log q)^{C \log N} \rho^{-q}.$$

Let us now fix $\epsilon > 0$, and find conditions on q such that the quantity above is less than ϵ . We require $C(Q, R)q^2 < \rho^{q/3}$, which gives rise to a first condition that q is greater than some other (overloaded) constant $C(Q, R)$. We also require $(C \log q)^{C \log N} < \rho^{q/3}$, which implies

$$q \geq C(\log N) \log(C \log N).$$

(where again C is overloaded.) Finally, we need $\rho^{-q/3} \leq \epsilon$, which implies

$$q \geq 3 \log_{\rho}\left(\frac{1}{\epsilon}\right).$$

In conclusion, we can make the uniform norm of recursive interpolation less than ϵ provided q^2 points are used on each tensor Chebyshev grid, with

$$q \geq C(Q, R) \max\left\{\log\left(\frac{1}{\epsilon}\right), (\log N) \log(C \log N)\right\}.$$

As was discussed earlier, the overall complexity of the butterfly scheme is $O(q^4 N \log N)$. This finishes the proof of Theorem 1.

4.4 Refinement

The radius of analyticity R of the amplitude is here assumed to be identical to that of the phase Φ , but the proof shows that we may relax this condition and let R be as small as a multiple $1/\sqrt{M}$. This corresponds to spatial variations that occur on a length scale proportional to the diameter of boxes at the midlevel in the X and Y domains.

5 Conclusion

We have presented a butterfly scheme for frequency-domain fast backprojection imaging from SAR data. The performance bounds in this paper provide direct user control of the reconstruction error, which may be useful for subsequent image-processing tasks. Unlike fast time-domain methods, our approach is amenable to real-time SAR systems which use “stretch” processing to accomplish I/Q demodulation and matched filtering [36]. Stretch processing has frequency-domain outputs which can easily be distributed amongst processors, making our algorithm especially attractive for extremely large scale imaging problems. Specifically, the pulse-wise range-frequency FFTs required to arrive at time-domain SAR backprojection algorithm can be prohibitively expensive due to data-throughput requirements. Thus in some SAR settings a frequency-domain approach such as the one presented here may be mandatory.

The ideas of the butterfly scheme are reminiscent but distinct from the fast multipole method (FMM) [22, 29]. The authors believe that the butterfly algorithm is the proper way to generalize FMM in an all-purpose way to settings where high-frequency oscillatory integrals are present. “All-purpose” means robustness, and applicability to many problems, such as SAR in the presence of curved flight paths, topography, complicated antenna beam patterns, etc. But it also means that other numerical methods may be faster for certain problems with structure, such as computerized (X-ray) tomography or ultrasound tomography in medical imaging. Adjoint-state seismic migration in heterogeneous media, on the other hand, may be too complicated to be accurately represented by an oscillatory (Kirchhoff) integral, so it is as yet unclear that the butterfly algorithm may be helpful there.

The fast Fourier transform and the USFFT [19] are examples of algorithms which are faster than the butterfly, but which only work for bilinear phases. The table below summarizes ballpark complexity figures and ranges of applicability for the FFT (i.e., slant-plane range-Doppler imaging), the PFA, the USFFT of Dutt and Rokhlin, and the Chebyshev butterfly (B-Cheb) algorithm presented here. The figures are for the one-dimensional transforms, so the complexity multiplier should be squared for the two-dimensional transforms.

Kernel	Algorithm	Complexity vs. FFT
e^{ixk}	FFT	1
$e^{ixk_1(m)k_2(n)}$	PFA	2
$e^{ix_jk_n}$	USFFT	6
$a(x, k)e^{i\phi(x, k)}$, $\phi(x, \alpha k) = \alpha\phi(x, k)$	B-Cheb	30

The variable k is supposed to take on large values, on the order of N , in all cases. By x_jk_n , it is meant xk sampled unevenly, whereas $xk_1(m)k_2(n)$ refers to data samples on a polar raster with $k_1(m)$ indexing frequency and $k_2(n)$ indexing pulse number. The relation $\phi(x, \alpha k) = \alpha\phi(x, k)$ (for $\alpha > 0$) is a homogeneity condition that the butterfly requires, or very nearly so, for operating at the $N \log N$ complexity level. It is ubiquitous in applications to wave propagation.

A On Chebyshev interpolation of analytic functions

The goals of this appendix go beyond the application to radar imaging, so the notations may depart slightly from those in the main text. For instance N here stands for the number of Chebyshev points – what we called q earlier.

The Chebyshev interpolant of a function f on $[-1, 1]$ is a superposition of Chebyshev polynomials $T_n(x)$,

$$p(x) = \sum_{n=0}^N c_n T_n(x),$$

which interpolates f in the sense that $p(x_j) = f(x_j)$ on the Chebyshev grid $x_j = \cos(j\pi/N)$ for $j = 0, \dots, N$.

The rationale for this choice of grid is that under the change of variable $x = \cos\theta$, the Chebyshev points become the equispaced samples $\theta_j = j\pi/N$. Unlike f , the function $g(\theta) = f(\cos\theta)$ is now 2π -periodic. Note that $g(\theta)$ inherits the smoothness of $f(x)$. The samples $g(\theta_j)$ can be made to cover the whole interval $[0, 2\pi]$ if we extend the range of j to be $0 \leq j \leq 2N - 1$ (this corresponds to a mirror extension of the original samples.) The rationale for choosing Chebyshev polynomials is that $T_n(\cos\theta) = \cos(n\theta)$, so that Chebyshev interpolation of f from $f(x_j)$, with $0 \leq j \leq N - 1$, is nothing but interpolation by trigonometric polynomials of g from $g(\theta_j)$, with $0 \leq j \leq 2N - 1$.

This interpolant is built as follows. Start by submitting the $2N$ samples $g(\theta_j)$ to the discrete Fourier transform and back; this gives

$$g(\theta_j) = \sum_{n=-N}^{N-1} e^{in\theta_j} \tilde{g}_n.$$

The spectral interpolant $q(\theta)$ is built from these DFT coefficients as

$$q(\theta) = \sum_{n=-N}^N{}'' e^{in\theta} \tilde{g}_n, \quad (29)$$

where the double prime next to the sum indicates that the first and last terms are halved. This precaution is important to ensure that the interpolant of a real-valued function is real-valued.

The sum (29) reduces to the inverse discrete Fourier transform when $\theta = \theta_j$, so that $q(\theta_j) = g(\theta_j)$. Notice that g is even, so only the cosines are needed in this expansion:

$$q(\theta) = 2 \sum_{n=0}^N{}'' \cos(n\theta) \tilde{g}_n.$$

The Chebyshev interpolant of $f(x)$ is then simply $p(x) = q(\arccos x)$. The coefficients are given explicitly as $c_n = 2\tilde{g}_n$ for $1 \leq n \leq N-1$, or $c_n = \tilde{g}_n$ for $n = 0, N$. Spectral and Chebyshev interpolation methods are not only attractive because the FFT can be used to speed up computations, but because they have remarkable accuracy properties.

A.1 Spectral accuracy of Chebyshev interpolation

The first result concerns the algebraic decay of the interpolation error when f can be differentiated a finite number of times, or super-algebraic decay when f is infinitely differentiable. We consider the native inner product for Chebyshev polynomials,

$$\langle f, g \rangle = \int_{-1}^1 f(x)g(x) \frac{dx}{\sqrt{1-x^2}},$$

with respect to which they are orthogonal. The associated weighted L_w^2 norm

$$\|f\| = \left(\int_{-1}^1 |f(x)|^2 \frac{dx}{\sqrt{1-x^2}} \right)^{1/2}$$

is used throughout this paper to measure the error. (The corresponding measure in $\theta = \arccos x$ is Lebesgue.) The related Sobolev spaces are

$$W_w^s = \{f \in L_w^2 : \|f\|_s^2 = \sum_{k=0}^s \|f^{(k)}\|^2 < \infty\}.$$

The following result is elementary. The ideas can be traced back at least to [21]. A proof of the result as stated is in [32].

Theorem 4. *Let $f \in W_w^s$. Denote by p the N -point Chebyshev interpolant of f on $[-1, 1]$. Then*

$$\|f - p\| \leq C_s \|f\|_s N^{-s}.$$

In [32], Tadmor pushed the analysis further to obtain exponential decay in the case when f is real-analytic. A convenient setting is to assume that f extends analytically in the complex plane, in the ‘‘Bernstein’’ ellipse E_ρ with foci ± 1 , center $z = 0$, and semi-axes

$$a_\rho = \frac{\rho + \rho^{-1}}{2}, \quad b_\rho = \frac{\rho - \rho^{-1}}{2},$$

for some parameter $\rho > 1$ called the elliptical radius. Note that $a_\rho + b_\rho = \rho$. This ellipse has Cartesian equation

$$E_\rho = \left\{ z : \frac{(\operatorname{Re} z)^2}{a_\rho^2} + \frac{(\operatorname{Im} z)^2}{b_\rho^2} = 1 \right\},$$

and parametric equation

$$E_\rho = \left\{ z = \frac{\rho e^{i\theta} + \rho^{-1} e^{-i\theta}}{2} : \theta \in [0, 2\pi) \right\}.$$

Theorem 5 (Tadmor [32]). *Let f have an analytic extension in the open Bernstein ellipse E_{ρ_0} with elliptical radius $\rho_0 > 1$. For each $1 < \rho < \rho_0$, let*

$$M(\rho) = \max_{z \in E_\rho} |f(z)|.$$

Denote by p the N -point Chebyshev interpolant of f on $[-1, 1]$. Then for all $0 < \rho < \rho_0$,

$$\|f - p\| \leq C \frac{M(\rho)}{\rho - \rho^{-1}} \rho^{-N}.$$

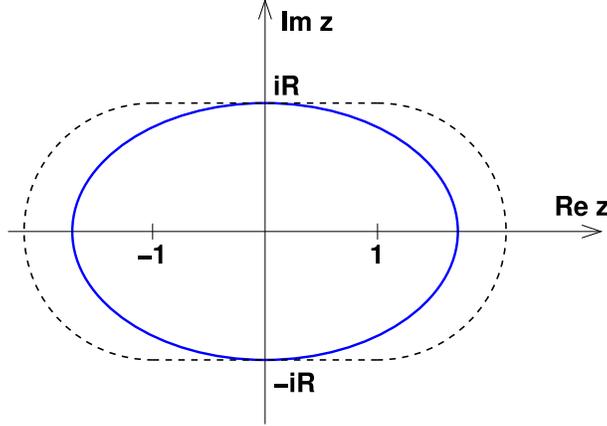


Figure 11: The stadium (dashed line) is the region of analyticity of f . The ellipse (blue, solid line) is the largest inscribed “Bernstein” ellipse with foci at ± 1 .

The next result, which is possibly original, is Theorem 2 presented in Section 4. For this result it is assumed instead that f is (Q, R) analytic, i.e., is real-analytic and obeys the smoothness condition

$$|f^{(n)}(x)| \leq Q n! R^{-n}. \quad (30)$$

As noted in [30], p. 378, f obeys (30) for $x \in \mathbb{R}$ if and only if it can be analytically extended in the strip $|\operatorname{Im} z| \leq R$. This property holds because R is a lower bound on the convergence radius of the Taylor expansion of f at any point x . As a result it is a very natural class of analytic functions; Rudin denotes it by $C\{n!\}$. We only assume that f obeys (30) for $x \in [-1, 1]$, which results in a stadium-shaped analyticity region, as in Figure 11. Note that (Q, R) analyticity has already been used by the authors in [12, 11].

A fortiori, the bound (19) in Theorem 2 also holds for the weaker L^2 norm. The proof gives the value $\frac{5}{2}\sqrt{\frac{45e}{2}}$ for the numerical constant C ; no attempt is made in this paper to find its sharp value. Note that $[R + \sqrt{R^2 + 1}]^{-N}$ corresponds to Tadmor’s ρ^{-N} .

The error bound (19) obeys the following asymptotic behaviors.

- As $R \rightarrow 0$, and if N less than or on the order of $1/R$, then the error bound is large.
- As $R \rightarrow 0$, and if $N \gg 1/R$, then the error bound is roughly proportional to $NR^{-1/2}e^{-RN}$.
- As $R \rightarrow \infty$, then the error bound is roughly proportional to $N(2R)^{-N}$.

A.2 Proof of Theorem 2

As mentioned earlier, f and p are respectively obtained from g and q through the change of variables $x = \cos \theta$. The factor $1/(\sqrt{1-x^2})$ is precisely the Jacobian of this change of variables. Hence it suffices to prove that $\|g - q\|_2$ obeys the bound (19).

We start by listing the consequences of the smoothness condition (30). As is well-known, f has a unique analytic continuation as the Taylor series

$$f(z) = \sum_{n=0}^{\infty} \frac{f^{(n)}(x)}{n!} (z - x)^n,$$

which by (30) is manifestly convergent as soon as $|z - x| \leq R$. Since $x \in [-1, 1]$, the domain of analyticity is the “stadium” illustrated in Figure 11, without its boundary. This shape is a subset of the strip $|\operatorname{Im} z| < R$.

Furthermore, for all $x \in [-1, 1]$ we have the bound

$$\begin{aligned} |f(z)| &\leq Q \sum_{n=0}^{\infty} \left(\frac{|z-x|}{R} \right)^n, \\ &\leq \frac{Q}{1 - |z-x|R^{-1}}, \end{aligned}$$

which results in

$$|f(z)| \leq \begin{cases} \frac{Q}{1 - |z+1|R^{-1}} & \text{if } \operatorname{Re} z < -1; \\ \frac{Q}{1 - |\operatorname{Im} z|R^{-1}} & \text{if } -1 \leq \operatorname{Re} z \leq 1; \\ \frac{Q}{1 - |z-1|R^{-1}} & \text{if } \operatorname{Re} z > 1 \end{cases} \quad (31)$$

The periodic function $g(\theta) = f(\cos \theta)$ also admits an analytic extension, best expressed through the function $h(z)$ such that $h(e^{i\theta}) = g(\theta)$. The result is the following lemma.

Lemma 5. *Let $h(e^{i\theta}) = f(\cos \theta)$, and assume that f is (Q, R) -analytic. Then h has a unique analytic continuation in the open annulus $|z| < R + \sqrt{R^2 + 1} < |z|^{-1}$, and obeys the bound*

$$|h(z)| \leq \frac{Q}{1 - \frac{||z| - |z|^{-1}||}{2} R^{-1}}. \quad (32)$$

Proof of Lemma 5. The analytic extension $h(z)$ of $h(e^{i\theta})$ is related to $f(z)$ by the transformation

$$h(z) = f\left(\frac{z + z^{-1}}{2}\right). \quad (33)$$

Indeed, $h(e^{i\theta}) = f(\cos \theta)$, so the two expressions match when $|z| = 1$. There exists a neighborhood of $|z| = 1$ in which the right-hand side is obviously analytic, hence equal to $h(z)$ by uniqueness. The rationale for this formula is the fact that $\cos \theta = \cos(i \log e^{i\theta})$, and $(z + z^{-1})/2$ is just another expression for $\cos(i \log z)$.

More can be said about the range of analyticity of $h(z)$. The map $z \mapsto \zeta = (z + z^{-1})/2$ is a change from polar to elliptical coordinates [6]. It maps each circle $C_\rho = \{\rho e^{i\theta} : \theta \in [0, 2\pi)\}$ onto the ellipse E_ρ of parametric equation $\{(\rho e^{i\theta} + \rho^{-1} e^{-i\theta})/2 : \theta \in [0, 2\pi)\}$ introduced earlier. Notice that $|z| = \rho_0$ and $|z| = \rho_0^{-1}$ are mapped onto the same ellipse.

Figure 11 shows the open stadium of height $2R$ in which f is analytic, as well as the largest ellipse E_ρ inscribed in that stadium. Its parameter ρ obeys

$$|\rho - \rho^{-1}|/2 = R,$$

corresponding to the case $\theta = \pm\pi/2$. Solving for ρ , we get

$$\rho = R + \sqrt{R^2 + 1} \quad \text{or} \quad \rho = \frac{1}{R + \sqrt{R^2 + 1}}.$$

As a result, any z obeying $|z| < R + \sqrt{R^2 + 1} < |z|^{-1}$ corresponds to a point of analyticity of $f\left(\frac{z+z^{-1}}{2}\right)$, hence of $h(z)$.

To see why the bound (32) holds, substitute $\zeta = (z + z^{-1})/2$ for z in the right-hand-side of (31). The vertical lines $\operatorname{Re} \zeta = \pm 1$ in the ζ plane become cubic curves with equations $(\rho + \rho^{-1}) \cos \theta = \pm 2$ in the z -plane, where $z = \rho e^{i\theta}$. Two regimes must be contrasted:

- In the region $|\operatorname{Re} \zeta| \leq 1$, we write

$$|\operatorname{Im}(z + z^{-1})| = |\rho \sin \theta - \rho^{-1} \sin \theta| \leq |\rho - \rho^{-1}|,$$

which leads to the bound (32) for h .

- Treating the region $\operatorname{Re} \zeta > 1$ is only slightly more involved. It corresponds to the region $(\rho + \rho^{-1}) \cos \theta > 2$ in the z plane; we use this expression in the algebra below. We get

$$\begin{aligned} |z + z^{-1} - 2| &= \left[((\rho + \rho^{-1}) \cos \theta - 2)^2 + (\rho - \rho^{-1})^2 \sin^2 \theta \right]^{1/2} \\ &\leq \left[((\rho + \rho^{-1}) \cos \theta - 2 \cos \theta)^2 + (\rho - \rho^{-1})^2 \sin^2 \theta \right]^{1/2}. \end{aligned}$$

In order to conclude that (32) holds, this quantity should be less than or equal to $|\rho - \rho^{-1}|$. To this end, it suffices to show that

$$(\rho + \rho^{-1} - 2)^2 \leq (\rho - \rho^{-1})^2, \quad \forall \rho > 0.$$

Expanding the squares shows that the expression above reduces to $\rho + \rho^{-1} \geq 2$, which is obviously true.

- The region $\operatorname{Re} \zeta < -1$ is treated in a very analogous manner, and also yields (32).

□

The accuracy of trigonometric interpolation is now a standard consequence of the decay of Fourier series coefficient of g . The result below uses the particular smoothness estimate obtained in Lemma 5. The proof technique is essentially borrowed from [32].

Lemma 6. *Let g be a real-analytic, 2π -periodic function of $\theta \in \mathbb{R}$. Define the function h of $z \in \{z : |z| = 1\}$ by $h(e^{i\theta}) = g(\theta)$, and assume that it extends analytically in the complex plane in the manner described by Lemma 5. Consider the trigonometric interpolant $q(\theta)$ of $g(\theta)$ from samples at $\theta_j = j\pi/N$, with $j = 0, \dots, 2N - 1$. Assume $N \geq 1/(2R)$. Then*

$$\|g - q\|_2 \leq C Q N \left[1 + \frac{1}{R^2} \right]^{1/4} \left[R + \sqrt{R^2 + 1} \right]^{-N}, \quad (34)$$

for some number $C > 0$.

Proof of Lemma 6. Write the Fourier series expansion of $g(\theta)$ as

$$g(\theta) = \sum_{n \in \mathbb{Z}} e^{in\theta} \hat{g}_n. \quad (35)$$

A comparison of formulas (35) and (29) shows that two sources of error must be dealt with:

- the truncation error, because the sum over n is finite in (29); and
- the aliasing error, because $\tilde{g}_n \neq \hat{g}_n$.

It is well-known that \tilde{g}_n is a periodization of \hat{g}_n , in the sense that

$$\tilde{g}_n = \sum_{m \in \mathbb{Z}} \hat{g}_{n+2mN}.$$

This equation is (a variant of) the Poisson summation formula. As a result,

$$\|g - q\|_2^2 = \sum_{|n| \leq N}'' \left| \sum_{m \neq 0} \hat{g}_{n+2mN} \right|^2 + \sum_{|n| \geq N}'' |\hat{g}_n|^2. \quad (36)$$

The decay of \hat{g}_n is quantified by considering that the Fourier series expansion of $g(\theta)$ is the restriction to $z = e^{i\theta}$ of the Laurent series

$$h(z) = \sum_{n \in \mathbb{Z}} \hat{g}_n z^n,$$

whereby the coefficients \hat{g}_n are also given by the complex contour integrals

$$\hat{g}_n = \frac{1}{2\pi i} \oint_{|z|=\rho} \frac{h(z)}{z^{n+1}} dz. \quad (37)$$

This formulation offers the freedom of choosing the radius ρ of the circle over which the integral is carried out, as long as this circle is in the region of analyticity of $h(z)$.

Let us first consider the aliasing error – the first term in the right-hand side of (36). We follow [32] in writing

$$\begin{aligned} \sum_{m>0} \hat{g}_{n+2mN} &= \sum_{m>0} \frac{1}{2\pi i} \oint_{|z|=\rho} \frac{h(z)}{z^{n+1+2mN}} dz, \\ &= \frac{1}{2\pi i} \oint_{|z|=\rho} \frac{h(z)}{z^{n+1}(z^{2N}-1)} dz. \end{aligned}$$

For the last step, it suffices to take $\rho > 1$ to ensure convergence of the Neumann series. As a result,

$$\left| \sum_{m>0} \hat{g}_{n+2mN} \right| \leq \rho^{-n} \frac{1}{\rho^{2N}-1} \max_{|z|=\rho} |h(z)|, \quad \rho > 1.$$

The exact same bound holds for the sum over $m < 0$ if we integrate over $|z| = \rho^{-1} < 1$ instead. Notice that the bound (32) on $h(z)$ is identical for ρ and ρ^{-1} .

Upon using (32) and summing over n , we obtain

$$\sum_{|n|\leq N}'' \left| \sum_{m\neq 0} \hat{g}_{n+2mN} \right|^2 \leq \left(\sum_{|n|\leq N}'' \rho^{2n} \right) \frac{4}{(\rho^{2N}-1)^2} \left[\frac{Q}{1 - \frac{\rho-\rho^{-1}}{2} R^{-1}} \right]^2. \quad (38)$$

It is easy to show that the sum over n is majorized by $\rho^{2N} \frac{\rho+\rho^{-1}}{\rho-\rho^{-1}}$.

According to Lemma 5, the bound holds as long as $1 < \rho < R + \sqrt{R^2 + 1}$. The right-hand side in (38) will be minimized for a choice of ρ very close to the upper bound; a good approximation to the argument of the minimum is

$$\rho = \tilde{R} + \sqrt{\tilde{R}^2 + 1}, \quad \tilde{R} = \frac{2N}{2N+1} R,$$

for which

$$\frac{1}{1 - \frac{\rho-\rho^{-1}}{2} R^{-1}} = 2N + 1.$$

The right-hand side in (38) is therefore bounded by

$$4Q^2(2N+1) \frac{1}{(\rho^N - \rho^{-N})^2} \frac{\rho + \rho^{-1}}{\rho - \rho^{-1}}.$$

This expression can be further simplified by noticing that

$$\rho^N - \rho^{-N} \geq \frac{1}{2} \rho^N$$

holds when N is sufficiently large, namely $N \geq 1/(2 \log_2 \rho)$. Observe that

$$\begin{aligned} \log_2 \rho &= \frac{\ln \left(\tilde{R} + \sqrt{\tilde{R}^2 + 1} \right)}{\ln 2} \\ &= \frac{1}{\ln 2} \operatorname{arcsinh}(\tilde{R}) = \frac{1}{\ln 2} \operatorname{arcsinh} \left(\frac{2N}{2N+1} R \right), \end{aligned}$$

so the large- N condition can be rephrased as

$$R \geq \frac{2N+1}{2N} \sinh \left(\frac{\ln 2}{2N} \right).$$

It is easy to check (for instance numerically) that the right hand-side in this expression is always less than $1/(2N)$ as long as $N \geq 2$. Hence it is a stronger requirement on N and R to impose $R \geq 1/(2N)$, i.e., $N \geq 1/(2R)$, as in the wording of the lemma.

The resulting factor $4\rho^{-2N}$ can be further bounded in terms of R as follows:

$$\rho = \tilde{R} + \sqrt{\tilde{R}^2 + 1} \geq \left(\frac{2N+1}{2N} \right) [R + \sqrt{R^2 + 1}],$$

so

$$\begin{aligned} \rho^{-N} &\leq \left(\frac{2N+1}{2N} \right)^{-N} [R + \sqrt{R^2 + 1}]^{-N} \\ &\leq \left(\exp \frac{1}{2N} \right)^{-N} [R + \sqrt{R^2 + 1}]^{-N} \\ &= \sqrt{e} [R + \sqrt{R^2 + 1}]^{-N}. \end{aligned}$$

We also bound the factor $\frac{\rho + \rho^{-1}}{\rho - \rho^{-1}}$ – the eccentricity of the ellipse – in terms of R by following a similar sequence of steps:

$$\begin{aligned} \frac{\rho + \rho^{-1}}{\rho - \rho^{-1}} &= \frac{2\sqrt{\tilde{R}^2 + 1}}{2\tilde{R}} \\ &\leq \frac{2N+1}{2N} \sqrt{1 + \frac{1}{R^2}} \\ &\leq \frac{5}{4} \sqrt{1 + \frac{1}{R^2}}. \end{aligned}$$

After gathering the different factors, the bound (38) becomes

$$\sum_{|n| \leq N}'' \left| \sum_{m \neq 0} \hat{g}_{n+2mN} \right|^2 \leq 20 e Q^2 (2N+1)^2 \sqrt{1 + \frac{1}{R^2}} [R + \sqrt{R^2 + 1}]^{-2N}. \quad (39)$$

We now switch to the analysis of the truncation error, i.e., the second term in (36). By the same type of argument as previously, individual coefficients are bounded as

$$|\hat{g}_n| \leq [\max(\rho, \rho^{-1})]^{-n} \frac{Q}{1 - \frac{\rho - \rho^{-1}}{2} R^{-1}}.$$

The sum over n is decomposed into two contributions, for $n \geq N$ and $n \leq -N$. Both give rise to the same value,

$$\sum_{n \geq N} \rho^{-2n} = \frac{\rho^{-2N}}{1 - \rho^{-2}}.$$

We let ρ take on the same value as previously. Consequently, $\frac{Q}{1 - \frac{\rho - \rho^{-1}}{2} R^{-1}} = 2N + 1$, and, as previously,

$$\rho^{-2N} \leq e [R + \sqrt{R^2 + 1}]^{-2N}.$$

We also obtain

$$\frac{1}{1 - \rho^{-2}} \leq \frac{\rho + \rho^{-1}}{\rho - \rho^{-1}} \leq \frac{5}{4} \sqrt{1 + \frac{1}{R^2}}.$$

As a result, the overall bound is

$$\sum_{|n| \geq N} |\hat{g}_n|^2 \leq \frac{5}{2} e Q^2 (2N+1)^2 \sqrt{1 + \frac{1}{R^2}} [R + \sqrt{R^2 + 1}]^{-2N}. \quad (40)$$

We obtain (34) upon summing (39) and (40), and using $2N + 1 \leq 5N/2$. □

References

- [1] F. Andersson, R. Moses, and F. Natterer, Fast Fourier Methods for Synthetic Aperture Radar Imaging *IEEE Transactions on Aerospace and Electronic Systems* **48** (2012)
- [2] A. Boag, Y. Bresler, and E. Michielssen, A multilevel domain decomposition algorithm for fast $O(N^2 \log N)$ reprojection of tomographic images, *IEEE Trans. Im. Proc.* **9-9** (2000) 1573 – 1582
- [3] S. Basu and Y. Bresler $O(N^2 \log N)$ filtered backprojection algorithm for tomography *IEEE Trans. Im. Proc.* **9-10** (2000) 1760–1773
- [4] G. Beylkin, The inversion problem and applications of the generalized Radon transform, *Comm. Pure Appl. Math.* **37** (1984) 579–599.
- [5] G. Beylkin, On the fast Fourier transform of functions with singularities, *Appl. Comput. Harmon. Anal.* **2-4** (1995) 363–381
- [6] J. Boyd, *Chebyshev and Fourier spectral methods* Dover Publications, Mineola, 2001.
- [7] A. Brandt, Multilevel computations of integral transforms and particle interactions with oscillatory kernels *Comput. Phys. Commun.*, **65** (1991) 2438
- [8] M. Brandfass, W. C. Chew Microwave imaging as applied to remote sensing making use of a multilevel fast multipole algorithm *Proc. SPIE 4053* (2000) 52–63
- [9] M. Bronstein, A. Bronstein, M. Zibulevsky, and H. Azhari, Reconstruction in diffraction ultrasound tomography using Nonuniform FFT, *IEEE Trans. Med. Im.* **21-11** (2002) 1395–1401
- [10] E. Candes, L. Demanet, D. Donoho, L. Ying, Fast Discrete Curvelet Transforms *SIAM Multiscale Model. Simul.* **5-3** (2006) 861–899
- [11] E. Candès, L. Demanet, L. Ying, Fast Computation of Fourier Integral Operators *SIAM J. Sci. Comput.* **29:6** (2007) 2464–2493.
- [12] E. Candès, L. Demanet, L. Ying, A Fast Butterfly Algorithm for the Computation of Fourier Integral Operators *SIAM Multiscale Model. Simul.* **7:4** (2009) 1727–1750
- [13] C. Casteel, L. Gorham et al, M. Minardi, S. Scarborough, K. Naidu, A challenge problem for 2D/3D imaging of targets from a volumetric data set in an urban environment *Algorithms for Synthetic Aperture Radar Imagery XIV, Proc. SPIE 6568* (2007)
- [14] M. Cheney and B. Borden *Fundamentals of radar imaging*, CBMS regional conf. series in applied mathematics, SIAM, Philadelphia, 2009.
- [15] L. Demanet and L. Ying Fast wave computation via Fourier integral operators *Submitted*, 2010
- [16] L. Demanet, M. Ferrara, N. Maxwell, J. Poulson, and L. Ying A butterfly algorithm for synthetic aperture radar imaging, in *Proc. SPIE Defense and Security Symposium, Algorithms for Synthetic Aperture Radar Imagery XVIII*, eds. E. Zelnio and F. Garber, April 2011.
- [17] A. Devaney, A fast filtered backprojection algorithm for ultrasound tomography, *IEEE Trans. Ultr. Ferr. Freq. Contr.* **34-3** (1987) 330–340
- [18] Y. Ding and D.C. Munson A fast back-projection algorithm for bistatic SAR imaging *Proc. IEEE Int. Conf. on Image Processing* **2** (2002) 441–444
- [19] A. Dutt and V. Rokhlin Fast Fourier Transforms for Nonequispaced Data, II *Applied and Computational Harmonic Analysis* **2-1** (1995) 85–100
- [20] B. Engquist and L. Ying. Fast directional multilevel algorithms for oscillatory kernels. *SIAM Journal on Scientific Computing* **29-4** (2007) 1710–1737

- [21] L. Fox and I. B. Parker, *Chebyshev polynomials in numerical analysis* Oxford University Press, Oxford, UK, 1968.
- [22] L. Greengard, V. Rokhlin, A fast algorithm for particle simulations, *Journal of Computers and Physics* **73** (1987) 325–348.
- [23] M. Gu, S. Eisenstat, Efficient algorithms for computing a strong rank-revealing QR factorization, *SIAM J Sci Comput* **17-4** (1996) 848–869.
- [24] P.-G. Martinsson and V. Rokhlin. A fast direct solver for scattering problems involving elongated structures. *J. Comput. Phys.* **221-1** (2007) 288302.
- [25] E. Michielssen and A. Boag, A multilevel matrix decomposition algorithm for analyzing scattering from large structures *IEEE Transactions on Antennas and Propagation* **44** (1996) 1086–1093
- [26] D.C. Munson, J.D. O’Brien and W.K. Jenkins, A tomographic formulation of spotlight-mode synthetic aperture radar, *Proc. IEEE* **8-71** (1983) 917–925
- [27] S Nilsson, LE Andersson Application of fast backprojection techniques for some inverse problems of synthetic aperture radar *Proc. SPIE* **3370** (1998) 62–72
- [28] M. O’Neil and V. Rokhlin, A new class of analysis-based fast transforms *Tech. Rep. 1384, Department of Computer Science, Yale University, August 2007.*
- [29] V. Rokhlin Rapid solution of integral equations of scattering theory in two dimensions *J. Comput. Phys.* **86-2** (1990) 414–439
- [30] W. Rudin, *Real and Complex analysis*, 3rd ed. McGraw-Hill ed., Singapore, 1987.
- [31] M. Soumekh, *Synthetic Aperture Radar Signal Processing with Matlab Algorithms* J. Wiley & Sons (Interscience Div.), 1999.
- [32] E. Tadmor, The exponential accuracy of Fourier and Chebyshev differencing methods *SIAM J. Num. Analysis*, 23:1 (1986) 1–10
- [33] N. Trefethen, *Spectral methods in Matlab* SIAM ed., Philadelphia, 2000.
- [34] M. Tygert Fast algorithms for spherical harmonic expansions, III *Preprint*, 2010.
- [35] L. Ulander, H. Hellsten, and G. Stenstrom, Synthetic-aperture radar processing using fast factorized back-projection *IEEE Trans. Aero. Elect. Systems* **39-3** (2003) 760–776
- [36] D. Wahl, P. Eichel, D. Ghiglia, P. Thompson, C. Jakowatz *Spotlight-Mode Synthetic Aperture Radar: A Signal Processing Approach* Springer, 1996.
- [37] A. Yegulalp Fast backprojection algorithm for synthetic aperture radar *Proc. IEEE radar conf.* (1999) 60–65
- [38] L. Ying Sparse Fourier transform via butterfly algorithm. *SIAM Journal on Scientific Computing*, **31** (2009) 1678
- [39] L. Ying, G. Biros, D. Zorin, A kernel-independent adaptive fast multipole method in two and three dimensions *Journal of Computational Physics*, **196-2** (2004) 591–626
- [40] S. Xiao, D.C. Munson, S. Basu, Y. Bresler An $N^2 \log N$ back-projection algorithm for SAR image formation *Proc. 34th Asilomar Conf. on Signals, Systems and Computers* Pacific Grove, CA, Nov. 2000, pp. 37.