Optimized Nonuniform FFTs and Their Application to Array Factor Computation

Amedeo Capozzoli, *Member, IEEE*, Claudio Curcio, and Angelo Liseno

***Abstract*— We deal with developing an optimized approach for implementing nonuniform fast Fourier transform (NUFFT) algorithms under a general and new perspective for 1-D transformations. The computations of nonequispaced results, nonequispaced data, and Type-3 nonuniform discrete Fourier transforms are tackled in a unified way. They exploit "uniformly sampled" exponentials to interpolate the "nonuniformly sampled" ones involved in the nonuniform discrete Fourier transforms (NUFDTs), so as to enable the use of standard fast Fourier transforms, and an optimized window. The computational costs and the memory requirements are analyzed, and their convenient performance is assessed also by comparing them with other approaches in the literature. Numerical results demonstrate that the method is more accurate and does not introduce any additional computational or memory burden. The computation of the window functions amounts to that of a Legendre polynomial expansion, i.e., a simple polynomial evaluation. This is convenient in terms of computational burden and of the proper arrangement of the calculations. A case study of electromagnetic interest has been carried out by applying the developed NUFFTs to the radiation of linear regular or irregular arrays onto a set of regular or irregular spectral points. Guidelines for multidimensional extension of the proposed approach are also presented.**

***Index Terms*— Array factor computation, computational electromagnetics, nonuniform fast Fourier transform (NUFFT), optimized window.**

I. Introduction

THE nonuniform Fourier transform is of interest in diverse domains of applied electromagnetics, such as medical imaging [1], radio astronomy [2], numerical solution of partial differential equations [3], [4], antenna analysis and synthesis [5]–[7], antenna near-field/far-field transformations [8], microwave imaging [9], and synthetic aperture radar [10]. Typically, in those applications, part of the computation amounts to evaluating the "spectral" values corresponding to a given set of N "spatial" sample data.

When the sampling is uniform in both spatial and spectral domains, the spectral and spatial samples are related by a standard discrete Fourier transform (DFT) relation. The fast Fourier transform (FFT) enables the calculation of standard DFTs in $O(N\log N)$ operations rather than $O(N^2)$, thanks to the multilevel algorithm based on the "divide et impera" concept [11]. On the contrary, when the sampling is nonuniform in at least one domain, the spectral and spatial samples will be related by a nonuniform DFT (NUDFT) relation, and hence the standard FFT algorithm does not apply anymore.

Several algorithms have been developed over the years to overcome this limitation [12]–[25]. They are referred to as nonuniform FFTs (NUFFTs), which rely on interpolation schemes designed purposely for "nonuniformly" sampled exponentials involved in the NUDFT. Such an interpolation serves to enable the "nonuniformly" sampled exponentials to be represented by uniformly sampled ones, so that the standard FFT can be used conveniently on an oversampled grid. In this way, the $O(N\log N)$ complexity of a standard FFT can be restored.

The abovementioned approaches differ mainly in their choice of the window function, which governs the tradeoff between accuracy and computational complexity.

The NUFFTs of interest are of three kinds, depending on the type of the underlying NUDFT computational problem.

1) Nonequispaced results (NER) NUFFT, or Type-1 NUFFT, if the spatial samples are located regularly, and the spectral samples irregularly.
2) Nonequispaced data (NED) NUFFT, or Type-2 NUFFT, if the spatial samples are located irregularly, and the spectral samples regularly.
3) Type-3 NUFFT, if both the spatial and spectral samples are located irregularly.

For the first two cases, a unified and general approach has been given in [18]. To limit the computational burden, the approach uses window functions having the property of being compactly supported with essentially compactly supported Fourier transforms. However, practical applications of the approach have been provided by using a Kaiser–Bessel window [12]–[18] that is regarded as an approximation to the zeroth order prolate spheroidal wave function (PSWF) [26]–[28], which is in fact compactly supported and has a Fourier transform essentially compactly supported. A few more schemes, for cases 1 and 2, have been proposed, involving least-squares window optimization [15], [22], min–max window optimization [17], least mean square error window optimization [23], [25], and different choices of the window functions that approximately fit the general formulation in [18] (see [13], [14], [16], [21], [24]).

The third case has been dealt with, throughout the literature, from a slightly different perspective. It has been viewed as a gridding problem, involving the use of Gaussian windows as interpolation functions [13], [19], [20]. Greengard and Lee [19] and Lee and Greengard [20] indicate the choice of the algorithm parameters that guarantee Fourier transforms of the Gaussian window with moderate essential compact

Manuscript received August 9, 2016; revised November 5, 2017; accepted January 22, 2018. Date of publication August 7, 2018; date of current version May 31, 2019. *(Corresponding author: Amedeo Capozzoli.)*

The authors are with the Dipartimento di Ingegneria Elettrica e delle Tecnologie dell'Informazione, Università di Naples Federico II, I 80125 Naples, Italy (e-mail: a.capozzoli@unina.it).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TAP.2018.2826368

support. This, in turn, guarantees a limited number of interpolation points, which entail only limited computational burden. Unfortunately, they offer no analytical justifications for their choice [19], [20]. Furthermore, the parameter choice is independent of target accuracy and, especially for the Type-3 problem, can lead to unsatisfactory results. Most recently, based on a thorough performance analysis of the resulting Type-3 NUFFT (referred to as NUFFT-3, henceforth), an approach is proposed to appropriately select the Gaussian window parameters that guarantee a predetermined accuracy [29]. In this approach, it can be seen that when high accuracy is needed, an exceedingly large number of interpolation points would be required for using Gaussian windows.

The literature survey reveals two missing points.

1) Devising a strategy, which helps in choosing window functions that enable the NER and NED NUFFTs with "optimized" performance, as compared to the performance of the Kaiser–Bessel window.
2) Extending the NUFFT-3 to more general, and possibly "optimized," windows, which would be capable of circumventing the above cited inefficiency of the Gaussian windows.

Indeed, with any fixed window function, some parameters, such as the oversampling factor or the interpolation window length, can be used to improve the accuracy or computational lightness, and thus for getting better results than the existing ones.

The aim of this paper is to tackle these two points, specifically for 1-D transformations. As regards the first point, the involved window must have compact support and its Fourier transform must be essentially compact supported, as stressed in Section II. Accordingly, the window must belong to the space spanned by the PSWFs, corresponding to singular values, significantly different from zero, i.e., to those before the step [26]–[28]. Indeed, the PSWFs have the same property required for the window function, i.e., compact support and essential compact support for the transform. The coefficients of the PSWF expansion are then so selected that they "optimize" the representation of the "nonuniformly" sampled exponential of the NUDFT. As regards the second point, a general, "optimized" NUFFT-3 algorithm has been developed, using an approach similar to that in [18], but exploiting the mentioned "optimized" PSWF representation of the involved window function.

It should be noted that in this approach, as the PSWFs will be expressed in terms of the Legendre polynomials, the computation of the window functions amounts to that of a Legendre polynomial expansion, namely, a simple polynomial evaluation. This is convenient from the viewpoint of not only mere computational burden but also a proper arrangement of the calculations.

The computational costs and the memory requirements of the proposed schemes are theoretically analyzed, and the very convenient performance is evaluated also by comparing the results with those of the approaches in the literature.

For this paper, the performance is evaluated in terms of the number of operations required, memory occupancy, and accuracy, but not in terms of computation time. This is an extremely relevant point because the time taken for the performance of actual implementations may strongly depend on software/hardware factors, such as memory latencies or proper exploitation of the cache memories and computation pipelines, which are beyond the scope of this paper. To what extent the implementation aspects may significantly influence the performance, in terms of computation time, is indicated, for example, by the famous FFT code, known as the fastest FFT in the West, which has revolutionized the computation of the FFT [30].

As mentioned at the beginning of this section, many areas of applied electromagnetics can benefit from the use of such efficient and effective tool. The developed "optimized" NUFFTs are applied here to a case of electromagnetic interest, namely, the radiation of linear regular or irregular arrays onto a set of regular or irregular spectral points. A problem of this type has indeed been attracting the attention of the electromagnetics community since long. The possibility of computing the far-field pattern of (regular) aperture-like antennas onto a regular output grid, by using the FFT algorithm, has been first recognized in [31]–[33]. Most recently, the NUFFT algorithm has been exploited, for the first time, to compute the far-field pattern radiated by irregular arrays onto regular grids [34], [35]. This approach has been subsequently extended to the more general case of regular or irregular, also conformal, arrays onto regular or irregular grids [5].

This paper is organized as follows. Section II recalls the definitions of NER and NED NUDFTs, as also of the corresponding NUFFT algorithms. Section III addresses Type-3 NUDFT and NUFFT, extending the latter from a more general perspective. Section IV introduces the PSWF expansion of the relevant window function [36], besides dealing with the optimization of such a function, in terms of expansion coefficient. Section V analyzes the computational costs and the memory requirements of the developed schemes. Section VI briefly recalls the methods in the literature, against which the developed approach is compared. Section VII reports and discusses the results of the numerical analysis and compares them with other schemes to prove how the calculation of the array factor for nonuniformly distributed data and/or results can benefit by using the optimized NUFFT. Section VIII proposes the guidelines for multidimensional extension of the proposed approach. Finally, Section IX presents the conclusions and the possible scenario of future developments in this field.

II. NER AND NED NUDFTS AND NUFFTS

In this section, the authors recall NER and NED NUFFTs from a perspective, different from that in [18]. They propose to use index l for the domain where nonuniformity occurs, and index k for the domain where uniformity occurs.

The NER NUDFT of uniform samples $\{z_k\}_{k=-(N/2)}^{(N/2)-1}$, evaluated at nonequispaced grid points $x_l \in [-(N/2), (N/2)]$, is defined as

$$\hat{z}_l = \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} z_k e^{-j2\pi x_l \frac{k}{N}}, \quad l = 1, \ldots, M. \tag{1}$$

On the other side, the NED NUDFT of samples $\{z_l\}_{l=1}^{M}$, located at nonequispaced points $\{x_l\}_{l=1}^{M}$ and evaluated on an equispaced grid, is defined as

$$\hat{z}_k = \sum_{l=1}^{M} z_l e^{-j2\pi x_l \frac{k}{N}}, \quad k = -\frac{N}{2}, \ldots, \frac{N}{2} - 1. \tag{2}$$

To guarantee fast and accurate processing, the "nonuniformly sampled" exponentials, $e^{-j2\pi x_l (k/N)}$, appearing in (1) and (2) require to be properly interpolated. The idea behind NER and NED NUFFTs is to use "uniformly sampled" exponentials $e^{jm\xi}$. This is considered convenient for three reasons.

1) Unlike other interpolation schemes, the representation is exact when the sampling, associated with x_l's, is uniform.
2) Thanks to the Poisson summation formula, the representation is purposely tailored to the family of functions to be represented, namely, "nonuniformly sampled" exponentials.
3) "Uniformly sampled" exponentials, $e^{jm\xi}$, as interpolation functions, enable "divide et impera" approaches [11], which lead to the use of standard FFTs. As is well-known, FFTs dramatically reduce the computational complexity.

As already mentioned, the exponential representation is made possible by using the Poisson summation formula [37]. Indeed, given the function $f \in \mathcal{L}(\mathbb{R})$, then $\sum_{m\in\mathbb{Z}} f(\xi + 2m\pi)$ converges absolutely almost everywhere to the 2π-periodic, locally integrable function, which can be expressed simply through a Fourier series as

$$\sum_{m\in\mathbb{Z}} f(\xi + 2m\pi) \sim (2\pi)^{-1/2} \sum_{m\in\mathbb{Z}} \hat{f}(m) e^{jm\xi} \tag{3}$$

where $\hat{f}(m) = \mathcal{F}[f; m]$ is the Fourier transform of f, calculated at m. The symbol $\sim$ can be replaced by full equality, if f is of bounded variation and continuous.

Considering a window $\varphi(\xi)$, having compact support $(-\xi_M, \xi_M)$, which is continuous and of bounded variation [see Fig. 1(a)], the Poisson formula can then be applied to the function $\varphi(\xi)e^{-jx\xi}$, which is again continuous and of bounded variation, and hence returns

$$e^{-j\xi x} = (2\pi)^{-1/2} \frac{\sum_{m\in\mathbb{Z}} \mathcal{F}[\varphi(\xi)e^{-j\xi x}; m] e^{jm\xi}}{\sum_{m\in\mathbb{Z}} \varphi(\xi + 2m\pi) e^{-j2m\pi x}}. \tag{4}$$

To obtain a computationally convenient expression for $e^{-j\xi x}$, the denominator must be expressed as a factorized function of x and ξ. A straightforward way of achieving this is to avoid overlapping of the replicas $\varphi(\xi + 2m\pi)e^{-j2m\pi x}$ which are related to the replication period 2π of the Poisson summation formula (3), and to choose an appropriate support for $\varphi(\xi)$. In other words, a natural choice, which corresponds to the periodicity of the expanding $e^{jm\xi}$ exponentials at the numerator of (4), is $\xi_M = \pi$ [see Fig. 1(b)], so that

$$e^{-j\xi x} = (2\pi)^{-1/2} \frac{\sum_{m\in\mathbb{Z}} \mathcal{F}[\varphi(\xi)e^{-j\xi x}; m] e^{jm\xi}}{\varphi(\xi)}, \quad |\xi| \le \pi \text{ and } \forall x \in \mathbb{R}. \tag{5}$$

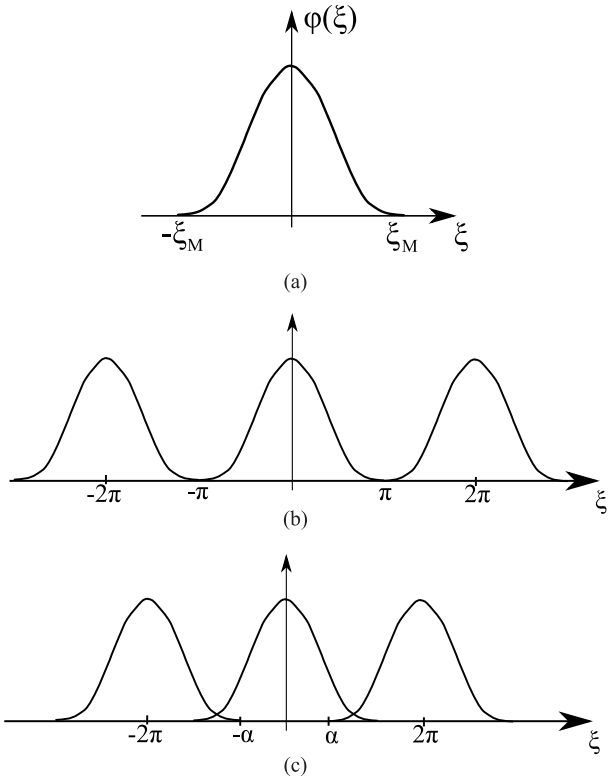


Fig. 1. Illustrating the use of the Poisson summation formula. (a) Window function. (b) Case $\xi_M = \pi$. (c) Case $\xi_M > \pi$.

However, by this method, divisions by very small numbers in (5) would take place because of unavoidable decay of $\varphi(\xi)$ toward the ends of $(-\pi, \pi)$. To avoid this problem, the support of $\varphi(\xi)$ can be chosen larger than π, namely, $\xi_M > \pi$ [Fig. 1(c)]. With this choice, different replicas overlap, and hence the representation of $e^{-j\xi x}$ must be considered only in the nonoverlapping interval $(-\alpha, \alpha)$. For the ensuing discussions, it is assumed that $\alpha = (\pi/c)$, with $c > 1$, and that c will be interpreted in the sequel as an oversampling factor.

According to the above reasoning and after some straightforward manipulations, the following final identity is obtained:

$$e^{-jx\xi} = \frac{(2\pi)^{-1/2}}{\varphi(\xi)} \sum_{m\in\mathbb{Z}} \hat{\varphi}(x-m) e^{-jm\xi}, \quad |\xi| \le \pi/c, \ \forall x \in \mathbb{R} \tag{6}$$

where $\hat{\varphi}(x) = \mathcal{F}[\varphi(\xi); x]$, i.e., the Fourier transform of $\varphi(\xi)$.

Equation (6) is amenable to meaningful interpretation (Fig. 2). Indeed, if x is considered a parameter, then $e^{-jx\xi}$ is represented in the spectral domain by a Dirac delta function, located at x. On the other side, $e^{-jm\xi}$'s are represented, again in the spectral domain, by Dirac delta functions located at m. So, it is not possible to express $e^{-jx\xi}$ as a summation of $e^{-jm\xi}$'s, unless x is an integer. To render such representation possible, it is necessary to smooth $e^{-jx\xi}$ by a window function $\varphi(\xi)$, and thus spread its spectrum first. Following the smoothing, $\varphi(\xi)e^{-jm\xi}$ becomes an ordinary function, which can then be expressed as a Fourier series summation provided by $(2\pi)^{-1/2} \sum_{m\in\mathbb{Z}} \hat{\varphi}(x-m)e^{-jm\xi}$. Therefore, the so-called scaling window φ and interpolation window $\hat{\varphi}$ are necessarily related, in terms of the Fourier transform relationship. Having established this, the problem now amounts to simple handling

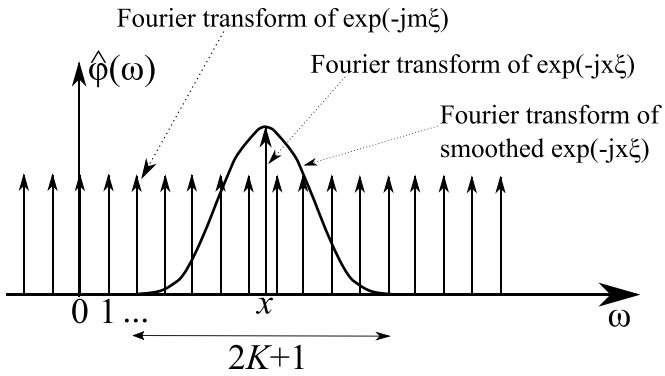


Fig. 2. Illustrating (6).

of the truncation of the summation in (6), as also of the division by the scaling window in the interval of interest.

It should be noted that (6) holds true for all real values of x. However, optimization of the scaling window φ, which will be addressed in the following, will be performed for a finite interval of interest $(-x_M, x_M)$, with a "sufficiently large" x_M. Furthermore, it should be noted that restricting the representation in (6) to $|\xi| \leq \pi/c$ is by no means a limitation, because whenever larger intervals are needed, it is enough to perform a stretching in ξ and x variables. In other words, whenever a representation is needed in $(-\xi_M, \xi_M)$, with $\xi_M > (\pi/c)$, then it is possible to consider the exponential $e^{-j\xi' x'}$ with $\xi' = \xi\xi_M/(\pi/c)$ and $x' = x(\pi/c)/\xi_M$.

Finally, it should be noted that to be of practical interest, the summation in (6) should be finite.

For further presentation of this paper, the following natural assumptions are made on $\varphi(\xi)$.

1) $\varphi(\xi)$ is real, without any loss of generality.
2) $\varphi(\xi)$ is even, because of the symmetry of the function $e^{-j\xi x}$.
3) The Fourier transform $\hat{\varphi}$ has support, essentially bounded to $[-K, K]$, so as to ensure a finite summation in (6).
4) cN is an integer; this can be achieved by ensuring that $c \in \mathbb{Q}$ and by suitably increasing N, i.e., the length of the input or output sequence, if necessary.

Equation (6), which leads to a representation of the exponential kernels of (1) and (2), permits efficient evaluation of the NUDFTs. In particular, such kernels are "uniformly sampled" in one variable and "nonuniformly sampled" in the other. Accordingly, it is assumed that $\xi = (2\pi k/cN)$ and $x = cx_l$, which lead to obtaining

$$e^{-j2\pi x_l \frac{k}{N}} = \frac{(2\pi)^{-\frac{1}{2}}}{\varphi\left(\frac{2\pi k}{cN}\right)} \sum_{m\epsilon\mathbb{Z}} \hat{\varphi}(cx_l - m) e^{-j2\pi \frac{mk}{cN}} \qquad l = 1, \ldots, M; \quad k = -\frac{N}{2}, \ldots, \frac{N}{2} - 1. \tag{7}$$

In (7), the summation over m can be truncated, following assumption c). In this way, (7) can be rewritten as

$$e^{-j2\pi x_l \frac{k}{N}} = \frac{(2\pi)^{-\frac{1}{2}}}{\varphi\left(\frac{2\pi k}{cN}\right)} \sum_{|m|\leq K} \hat{\varphi}(cx_l - (\mu_l + m)) e^{-j2\pi \frac{(\mu_l+m)k}{cN}} \qquad l = 1, \ldots, M; \quad k = -\frac{N}{2}, \ldots, \frac{N}{2} - 1 \tag{8}$$

where equality now holds true in an approximate sense.

Accordingly, for the NER case, the following equation is obtained:

$$\hat{z}_l = (2\pi)^{-1/2} \underbrace{\sum_{|m|\leq K} \hat{\varphi}_{lm} \underbrace{\sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} \underbrace{\frac{z_k}{\varphi\left(\frac{2\pi k}{cN}\right)}}_{Step\ 1} e^{-j2\pi \frac{(\mu_l+m)k}{cN}}}_{Step\ 2}}_{Step\ 3} \tag{9}$$

where $\hat{\varphi}_{lm} = \hat{\varphi}(cx_l - (\mu_l + m))$, $\mu_l = [cx_l]$, and $[\nu]$ is the integer, nearest to ν. Obviously, larger values of K increase accuracy at the expense of increased computational burden, because more terms are needed in the above summation for m.

From an algorithmic point of view, (9) can be regarded as involving three sequential steps [18]. Step 1 amounts at a scaling and a zero padding up to a length equal to cN, and Step 2 can be performed by a standard FFT over cN points; Step 3 is an interpolation step. Because $cN > N$, the "effective" length of the sequence to be transformed c can be interpreted as an oversampling factor.

Similarly, for the NED case, the following is obtained:

$$\hat{z}_k = \frac{(2\pi)^{-1/2}}{\varphi\left(\frac{2\pi k}{cN}\right)} \sum_{|m|\leq K} \sum_{l=1}^{M} z_l \hat{\varphi}_{lm} e^{-j2\pi \frac{(\mu_l+m)k}{cN}} \tag{10}$$

If the domains of definition of z_l, μ_l, and $\hat{\varphi}_{lm}$ are extended so that z_l and μ_l vanish for $l < 1$ and $l > M$, and $\hat{\varphi}_{lm}$ vanishes for $l < 1$, $l > M$, and $|m| > K$, then (10) becomes

$$\hat{z}_k = \frac{(2\pi)^{-1/2}}{\varphi\left(\frac{2\pi k}{cN}\right)} \underbrace{\underbrace{\sum_{i=-\frac{cN}{2}}^{\frac{cN}{2}-1} \underbrace{U_i}_{Step\ 1} e^{-j2\pi \frac{ik}{cN}}}_{Step\ 2}}_{Step\ 3}, \tag{11}$$

where

$$U_i = \sum_{l\epsilon\mathbb{Z}} \sum_{m\epsilon\mathbb{Z}} z_l \hat{\varphi}_{li+cmN-\mu_l}. \tag{12}$$

If $\hat{\varphi}$ is small outside some interval, $[-K, K]$, then $\hat{\varphi}_{li+cmN-\mu_l}$ is nonzero only for $i + cmN - \mu_l \leq K$, and the summation in m can be truncated, whereas that in l is automatically truncated by the support of z_l. Again, for the NED case, (11) involves three steps [16]: Step 1, i.e., the calculation of U_i, is an interpolation step; Step 2 can be performed by a standard FFT over cN points, and Step 3 is scaling.

III. NUDFT-3 and Improved NUFFT-3

NUDFT-3 of the nonuniform samples $\{z_l\}_{l=1}^{M}$ acquired at $\{x_l\}_{l=1}^{M}$ and evaluated at the nonequispaced grid points $\{s_k\}_{k=1}^{N}$ is defined as

$$\hat{z}_k = \sum_{=1}^{N} z_l e^{-js_k x_l}, \quad k = 1, \ldots, N. \tag{13}$$

Here, unlike NER and NED NUDFTs, both the indices l and k address domains where nonuniformity occurs.

The NUFFT-3 algorithm of [19] and [20] uses Gaussian window interpolation functions to represent the nonuniformly sampled exponentials. We now propose to improve the NUFFT-3 algorithm with more general window functions, using the same approach, exploited for the NER and NED NUFFTs of the foregoing section. To this end, also in this case, (6) leads to a representation of the exponential kernel in (13), which permits efficient evaluation of NUDFT-3. Particularly, in (6), if it is assumed that $\xi = (\pi s_k/cS)$ and $x = (cSx_l/\pi)$, where $S = \max\{|s_k|\}_{k=1}^{N}$, then $|\xi| \leq (\pi/c)$ and

$$e^{-js_k x_l} = \frac{(2\pi)^{-1/2}}{\varphi\left(\frac{\pi s_k}{cS}\right)} \sum_{m\epsilon\mathbb{Z}} \hat{\varphi}\left(\frac{cSx_l}{\pi} - m\right) e^{-jm\frac{\pi}{c}\frac{s_k}{S}}. \quad (14)$$

Using (14) in (13), and after some straightforward manipulations, one gets

$$\hat{z}_k = \underbrace{\frac{(2\pi)^{-1/2}}{\Phi\left(\frac{\pi sk}{cS}\right)} \underbrace{\sum_{m\in\mathbb{Z}} \underbrace{\left[\sum_{l=1}^{M} z_l \hat{\varphi}\left(\frac{cS}{\pi}x_l - m\right)\right]}_{Step\ 1} e^{-j\frac{2\pi m}{2N}\left(\frac{s_k N}{cS}\right)}}_{Step\ 2}}_{Step\ 3} \quad (15)$$

In (15), Step 1 is an interpolation step. Step 2 is the expression of NER-NUDFT of a $2N$ long sequence, evaluated at the output points $(s_k N/cS)$. Hence, it can be computed via the same scheme, expounded in the foregoing section. In other words, Step 2 effectively includes three steps. Finally, Step 3 is scaling. Having noticed that Step 2 is composed of three internal steps, it can be summed up that NUFFT-3 amounts to comprising overall five steps, as the schemes in [19], [20], and [29].

IV. PSWFs Representation of the Window Function and Its Optimization

For fixed values of x_M, K, and c, the scaling window φ, "minimizing" the representation error of $e^{-j\xi x}$ in (6), remains to be determined. It is therefore appropriate to introduce here the proposed optimization process. To maintain the general purpose of the results, the optimization of the representation is performed on the individual exponential $e^{-j\xi x}$, rather than on its summations as in the NUFFTs.

For the three foregoing NUFFTs to be effective, the function $\hat{\varphi}$ must be small, outside some interval $[-K, K]$. Therefore, the scaling window φ must have compact support containing $[-\alpha, \alpha]$ and its Fourier transform must be essentially supported in $[-K, K]$ for all the three NUFFTs. Consequently, the scaling window φ must belong to the space spanned by the PSWFs corresponding to the singular values of the Fourier transform operator having amplitude larger than the "knee value" determined by the step-like behavior of the singular values [26]–[28]. In other words

$$\varphi(\xi) = \sum_{t=0}^{T} \gamma_t \Psi_{2t}[w, \xi] \quad (16)$$

where Ψ_k denotes the kth PSWFs with space-bandwidth product (SBP) w, T the number of expansion functions, and γ_t the unknown expansion coefficients. As can be seen from (15), owing to the symmetry of φ, only the PSWFs with even order, namely, PSFWs with even symmetry, are considered [28]. To enforce the desired properties on φ, the SBP is chosen as

$$w = \chi(2\pi - \frac{\pi}{c})K \quad (17)$$

where χ is a coefficient, which can be subject to further optimization. The number of retained expansion PSWFs is chosen as $2T \leq \lfloor(2w/\pi)\rfloor$, where $\lfloor\nu\rfloor$ is the largest integer smaller than υ [36]. In this way, φ has compact support in $[-\chi(2\pi - (\pi/c)), \chi(2\pi - (\pi/c))]$ and $\hat{\varphi}$ is essentially supported in $[-K, K]$.

According to (16), the expansion coefficients γ_t are determined to optimize the error functional

$$\Gamma(\underline{\gamma}) = \int_{-x_M}^{x_M}\int_{-\frac{\pi}{c}}^{\frac{\pi}{c}} d\xi dx \left| e^{-jx\xi} - \frac{(2\pi)^{-1/2}}{\varphi\left(\xi;\underline{\gamma}\right)} \sum_{|m|\leq K} \hat{\varphi} \times (x - (\lfloor x\rfloor + m);\underline{\gamma}) e^{-j(\lfloor x\rfloor + m)\xi} \right| \quad (18)$$

where $\underline{\gamma}$ is the vector of γ_t's.

Throughout the optimization, the setting of $x_M = 10\pi$ is shown to be enough for the numerical results in the following. In (18), the dependence of φ and $\hat{\varphi}$ on $\underline{\gamma}$ is highlighted explicitly.

Note that the optimization of $\Gamma(\underline{\gamma})$ depends only on c and K, but not on N, M or on the nonuniform (input and/or output) sample locations. Therefore, all the NUFFTs, sharing the same values of c and K, require only a single optimization. Moreover, the results of optimization of $\Gamma(\underline{\gamma})$ can be saved for future reuse. For example, as will be discussed later in the sequel, the results of optimization for the cases $c = 1.5$, $K = 3$; $c = 1.5$, $K = 6$; $c = 2$, $K = 3$; and $c = 2$, $K = 6$ are provided in [43]. Those interested in computing NUFFTs with such values of c and K can utilize those results, rather than optimizing $\Gamma(\underline{\gamma})$ again.

V. Computational Complexity and Memory Requirements

The theoretical performance of the three (NER, NED, and Type-3) proposed NUFFT schemes is analyzed now, in terms of asymptotic complexity and memory requirements. For analysis, it is assumed that $N \sim M$.

The analysis will not account for the optimization of $\Gamma(\underline{\gamma})$, which, as mentioned, can be performed offline *una tantum*. Moreover, the cost analysis of the windows φ and $\hat{\varphi}$ in Sections V-A and V-B is peculiar to our approach, while the computational/memory analysis in Sections V-C–V-E is common to all the approaches and is based on the formulation in [18] (see [13], [14], [16], [19], [20]), which is reported here for convenience. According to these analyses, and as can be inferred from Section VI, where the computational

and memory requirements for other schemes are summarized, the proposed approach gains in accuracy without introducing criticalities, in terms of computations and memory storage, as compared to other NUFFT schemes.

A. Computation of φ

Before proceeding with computation, we point out that the evaluation of the window function φ, according to (16), requires accurate calculation of the PSWFs, which is a sensitive problem. For this paper, the PSWFs are calculated accurately by an established computational scheme, based on a Legendre polynomial representation as in [38], some details of which are provided in the Appendix.

According to (A.5), the calculation of φ amounts to the computation of a Legendre polynomial expansion [39], i.e., φ can be computed as a simple polynomial evaluation. This is convenient from the viewpoint of not only computational burden but also proper arrangement of the calculations.

If C is the maximum cost for computing one sample of an individual polynomial,[1] then for S output points, expansion (A.5) costs at the most $CS(0.5(K_{\text{leg}}-1)+1)$ operations. For NER and NED NUFFTs, $S=N$, and for NUFFT-3, $S=N+M$. In the computational scheme of [38], K_{leg} grows linearly with SBP. Consequently, C and K_{leg} being independent on both $N$and M (the optimized windows depend only on c and $2K+1$), the calculation of φ costs $\mathcal{O}((2K+1)N)$ operations.

B. Computation of $\hat{\varphi}$

To avoid the possibility of introducing numerical errors, $\hat{\varphi}$ is not computed by Fourier transforming φ numerically but by using the analytical expression of the Fourier transform of the Legendre polynomials [40, Formula 18.17.19]. Again, some details of the numerical evaluation of $\hat{\varphi}$ are provided in the Appendix. According to (A.6), the calculation of $\hat{\varphi}$ amounts to computing a Bessel function expansion, where the computation of each Bessel function can be performed as a power series [39]. If $D$is the maximum cost (in the same sense as above) for computing each sample of $\sqrt{(2\pi/x)}J_{k+(1/2)}(x)$, then the cost for T output points, $\hat{\varphi}$, would be $DS(0.5(K_{leg}-1)+1)$ operations. Again, D and K_{leg} being independent on T, $\hat{\varphi}$ costs, at the most, $DT(0.5(K_{\text{leg}}-1)+1)$ operations. Concerning the number of points at which $\hat{\varphi}$ is to be computed, it is observed that for NER and NED NUFFTs, the samples of interest are $\hat{\varphi}_{lm}$'s in (9). Accordingly, they are $T=(2K+1)M$ for these two cases. For NUFFT-3, an interpolation, using the interpolation window $\hat{\varphi}$, is first used, and then NER NUFFT is applied, so that they are $T=2(2K+1)M$. Consequently, and taking into account that K_{leg} grows linearly with the SBP, the computation of $\hat{\varphi}$ costs $O((2K+1)^2N)$ operations.

C. "Precomputed Mode" Versus "Reduced Memory Mode"

For all the three NUFFTs, the values of the windows, φ and $\hat{\varphi}$, can be either precomputed or computed "on-the-fly." Following the terminology in vogue (see [17], [21]), the former case is addressed as "precomputed mode" and the latter as "reduced memory mode."

[1]The cost for computing the values of $P_{2k}(\xi)$ depends on k. C is the maximum cost over k.

TABLE I
COMPUTATIONAL COMPLEXITY AND MEMORY REQUIREMENTS OF THE PROPOSED NER NUFFT ($N \sim M$)

Computational cost		
Step	***Precomputed mode***	***Reduced memory mode***
Step #1	N	$\mathcal{O}((2K+1)N)$
Step #2	$\mathcal{O}(cNlogN)$	$\mathcal{O}(cNlogN)$
Step #3	$\mathcal{O}((2K+1)N)$	$\mathcal{O}((2K+1)^2N)$
Memory		
Step #1	$(c+1)N$	cN
Step #2	/	/
Step #3	$(2K+1)N$	/

The precomputed mode applies to cases when the NUFFT must be calculated several times and consists in storing the values of the involved window functions, for example, the $\hat{\varphi}_{lm}$'s and/or of the $\varphi(2\pi k/cN)$'s for NER and NED NUFFTs, provided x_l's and/or N remains constant during the processing. A typical case when precomputed mode is possible is represented by iterative image reconstruction [17]. Different kinds of precomputed modalities are possible, depending on how many parameters remain constant during processing. For example, in the NUFFT-3 case, during the iterations in the synthesis of aperiodic arrays, s_k's may remain fixed, while x_l's may vary [5], [34], [35]. The advantage of the precomputed mode is that the windows' cost can be traded off for additional memory occupancy and amortized during the whole iterative process.

The reduced memory mode applies when storage of the window values has to be avoided. In this case, no additional memory storage is required and the computational costs of Sections V-A and V-B become relevant.

D. NER NUFFT

1) Step #1 (scaling and zero padding) involves the scaling of N complex numbers z_k by $\varphi(2\pi k/cN)$ and zero padding to a sequence of cN elements. Besides, calculation/storage of $\varphi(2\pi k/cN)$'s requires N divisions and memory space for cN complex elements.
2) Step #2 (FFT) involves calculation of standard FFT of length cN, which costs $O(cN\log N)$; such an in-place calculation does not cost extra memory space.
3) Step #3 (interpolation) requires, for each value of l, and besides the calculation/storage of $\hat{\varphi}_{lm}$'s: 1) performing $2K+1$ complex multiplications between the nonzero elements of $\hat{\varphi}_{lm}$ and the corresponding $2K+1$ outputs of the foregoing FFT and 2) a final summation of $2K+1$ complex numbers, which costs $O((2K+1)N)$.

Table I summarizes the computational and memory requirements of the NER NUFFT scheme.

E. NED NUFFT

The discussion of the foregoing section applies to NED NUFFT also, subject to changes in steps #1 and #3.

F. NUFFT-3

The foregoing arguments apply even to the analysis of the computational cost and memory requirements of NUFFT-3.

In more detail, step #1 is a convolution analogous to step #3 of the NER NUFFT, step #2 is calculated by the NER NUFFT algorithm, and step #3 involves scaling, analogous to step #1 of the NER NUFFT.

G. Computational Complexity and Accuracy

From above, it appears that for NER, NED, or NUFFT-3, FFT is the most demanding step, computationally. Furthermore, and generally speaking, the accuracy of NUFFT scheme improves by increasing either the oversampling factor c or the support $2K+1$ of the interpolation window. Therefore, devising an efficient NUFFT scheme, which is both accurate and computationally convenient, requires an interpolation window, having maximum possible energy in $(-K, K)$, so as to reduce c for a fixed prescribed accuracy.

VI. Other Approaches in Literature

This section recalls briefly some NUFFT approaches in the literature, which will be used in the next section as a benchmark for evaluating the performance of the proposed scheme. These approaches differ essentially in the way they represent the exponential $e^{-jx\xi}$ or its discrete counterpart $e^{-j2\pi x_l(k/N)}$. The compared approaches will be then presented in this section under this perspective.

A. Approach of Liu and Nguyen

The approach in [15] and [41] exploits the following approximation of $e^{-j2\pi x_l(k/N)}$, which will be used for NED NUFFT:

$$e^{-j2\pi x_l \frac{k}{N}} = \frac{1}{\varphi\left(\frac{2\pi k}{cN}\right)} \sum_{|m|\le K} \tilde{\varphi}(cx_l - (\mu_l + m)) e^{-j2\pi \frac{(\mu_l+m)k}{cN}} \quad l = 0, \ldots, M-1; \quad k = -\frac{N}{2}, \ldots, \frac{N}{2} - 1 \tag{19}$$

where c is an integer, larger or equal to 2. The approach does not exploit the Poisson summation formula and is thus at variance with (8). Consequently, the window functions $\varphi(\xi)$ and $\tilde{\varphi}(x)$ will no longer be in a Fourier transform relationship. In more detail, $\varphi(\xi)$ is chosen as

$$\varphi(\xi) = \cos\left(\frac{\xi}{2}\right) \tag{20}$$

while $\tilde{\varphi}(x)$ is determined by matching, in the least-squares sense, the left- and right-hand sides of (19). The matching gives

$$\tilde{\varphi}(x_l) = \underline{\underline{F}}^{-1} \underline{a}(x_l) \tag{21}$$

where the elements of matrix $\underline{\underline{F}}$ are given by

$$F_{pq} = \begin{cases} N, & p = q \\ \dfrac{w^{(\mathrm{p-q})N/2} - w^{(q-p)N/2}}{1 - w^{q-p}}, & p \neq q \end{cases} \tag{22}$$

and the vector by

$$a_k(x_l) = j \sum_{\gamma=-1,1} \frac{\sin\left[\frac{\pi}{2c}(2k - \gamma - 2K - 2(cx_l - [cx_l]))\right]}{1 - e^{j\frac{\pi}{cN}[2(cx_l - [cx_l]) + 2K - 2k + \gamma]}} \tag{23}$$

TABLE II

Computational Complexity of the Approaches by Liu and Nguyen and by Kuo and Lee ($N \sim M$)

Computation	
Calculation of the $a_k(x_l)$	$\mathcal{O}((2K+1)N)$
Calculation of the products in eq. (21) (assuming to having precalculated $\underline{\underline{F}}^{-1}$)	$\mathcal{O}((2K+1)^2 N)$
Calculation of $\varphi\left(\frac{2\pi k}{cN}\right)$	$\mathcal{O}(N)$
Calculation of the coefficients $\tilde{\varphi}(cx_l - (\mu_l + m))e^{-j2\pi\frac{\mu_l k}{cN}}$	$\mathcal{O}((2K+1)N)$
FFT	$\mathcal{O}(cN log N)$
Memory	
Storage of $\underline{\underline{F}}^{-1}$	$(2K+1)^2$

Matrix $\underline{\underline{F}}^{-1}$ is precomputed and stored, so as to amortize its computational cost. Table II shows the computational complexity of different steps involved [41].

B. Approach of Kuo and Lee

The approach in [22] is similar to that in [15], the only difference being that

$$\varphi(\xi) = \cos^n\left(\frac{\xi}{2}\right), \quad n > 1. \tag{24}$$

Following this choice, the expression of $\underline{a_k}(x)$ differs from that in (23) and is not reported here for the sake of brevity. The exponent n is used as a further degree of freedom to possibly improve the accuracy.

For integer n, the computational complexity and the memory storage requirements are the same as those in [15] and [41].

C. Approach of Fessler and Sutton

For NER NUFFT, Fessler and Sutton [17] use the same approximation in (19). The window functions, $\varphi(\xi)$ and $\tilde{\varphi}(x)$, are of course different from those in [15] and [22], but as in [15] and [22], they are not in the Fourier transform relationship.

Concerning the choice of $\varphi(\xi)$, the authors propose a truncated Fourier series expansion retaining $2L+1$ terms, namely,

$$\varphi(\xi) = \sum_{t=-L}^{L} \alpha_t e^{j\beta\xi t}. \tag{25}$$

In this way, $\varphi(\xi)$ is parametrically represented by $2L+1$ expansion coefficient α_t, as well as by β. Equation (25) generalizes other literature choices. For example, the case $L = 1$, $\alpha_0 = 0$, $\alpha_{-1} = \alpha_1 = 0.5$, and $\beta = 0.5$ corresponds to (20).

Regarding the choice of $\tilde{\varphi}(x)$, it is analytically found by using a min–max criterion under the parametric representation (25). Of course, at this stage, the expression of $\tilde{\varphi}(x)$ will also be parameterized to α_t's and to β.

Finally, α_t's and β are selected as those minimizing the residual error, after optimization of $\tilde{\varphi}(x)$. The authors have found that best results can be achieved by choosing $\varphi(\xi)$ as a Kaiser–Bessel window with optimized parameters.

TABLE III
COMPUTATIONAL COMPLEXITY OF THE APPROACHES BY FESSLER AND SUTTON ($N \sim M$)

Computation	
Precomputation of $\tilde{\varphi}(cx_l - (\mu_l + m))$ (precomputed mode)	$\mathcal{O}((2K+1)(L+1+2K+1))$
Computation of $\tilde{\varphi}(cx_l - (\mu_l + m))$ (reduced memory mode)	$\mathcal{O}((2K+1)(L+1+2K+1)N)$
Interpolation using the $\tilde{\varphi}(cx_l - \mu l + m$'s	$\mathcal{O}(N(2K+1))$
Calculation of $\varphi\left(\frac{2\pi k}{cN}\right)$	$\mathcal{O}(N)$
FFT	$\mathcal{O}(cN log N)$
Memory	
Storage of $\tilde{\varphi}(cx_l - (\mu_l + m))$ (precomputed mode)	$(2K+1)N$
Storage requirements (reduced memory mode))	$(2K+1)^2$

They propose different strategies to tradeoff between computational complexity and memory storage. Table III summarizes the most popular precomputed modes and the reduced memory modes.

D. Approach of Yang and Jacob

The scheme in [23] and [25] is similar to that of Fessler and Sutton [17], but for one difference: for the fixed scaling window, [2] the interpolation window $\tilde{\varphi}(x)$ is searched for according to a mean square error criterion instead of a min–max one.

To prevent the evaluation of the interpolator from dominating the computational complexity [23] and *to precompute the interpolators on a uniform grid,* Yang and Jacob [25] suggest that the interpolation window $\tilde{\varphi}(x)$ be given in the following representation:

$$\tilde{\varphi}(x) = \sum_{k=-OK+1}^{OK-1} q_k \mathrm{BS}(Ox - k) \tag{26}$$

where $\mathrm{BS}(y)$ is a B-spline function and O an "oversampling" factor. Ideally, q_k's should coincide with $\tilde{\varphi}((k/O))$. The $2OK - 1 q_k's$ are subject to the optimization process.

As regards $\varphi(\xi)$, its expression is selected to be

$$\varphi(\xi) = \frac{\mathcal{F}[\tilde{\varphi}](\xi)}{\sum_{k\in\mathbb{Z}} |\mathcal{F}[\tilde{\varphi}](\xi + 2k\pi)|^2}. \tag{27}$$

As no satisfactory results could be achieved by this method (see Section VII-B), discussion on computational/memory requirements is skipped.

E. Approach of Dutt and Rokhlin (Gaussian Windows)

The following NUFFT schemes employ (8) with different choices of windows, φ and $\hat{\varphi}$. For them, the following common parameter δ is defined thus:

$$\delta = \pi\left(2 - \frac{1}{c}\right). \tag{28}$$

[2]Note that Jacob [23] and Yang and Jacob [25] denote the scaling window by $\tilde{\varphi}$ and the interpolation window by φ, which is at variance from the viewpoint of the authors.

In particular, the scheme of Dutt and Rokhlin [13] uses the following windows:

$$\begin{cases} \varphi(\xi) = e^{-b\left(\frac{\xi}{2}\right)^2} \\ \hat{\varphi}(\xi) = \sqrt{\frac{2}{b}} e^{-\frac{x^2}{b}} \end{cases} \qquad b = \frac{3c}{2c-1}\frac{K}{\pi}. \tag{29}$$

Actually, a criterion to select the parameters of the Gaussian window for a fixed accuracy has been derived in [29]. However, such a criterion leads to numerous interpolation points, which prevent comparison with other windows for fixed values of K. Therefore, the authors have preferred the choice in [13] for this paper.

The discussion on computational/memory requirements for this method, as also for those in the following sections, follows the same lines as those in Section V for the proposed scheme, apart from the computation of the window functions.

F. Approach by Beylkin (B-Splines)

The approach in [14] (see also [16]) uses the following windows:

$$\begin{cases} \varphi(\xi) = (2\pi)^{-0.5} \mathrm{sinc}^{2K}\left(\frac{\xi}{2\pi}\right) \\ \hat{\varphi}(\xi) = B_{2K-1}(x) \end{cases} \tag{30}$$

where $\mathrm{sinc}(x) = (\sin(\pi x))/(\pi x)$ and $B_n(x)$ is the B-spline of order n, recursively defined as

$$B_0(x) = \begin{cases} 1, & |x| \le 0.5 \\ 0, & otherwise \end{cases} \tag{31}$$

for $n = 0$, and

$$B_n(x) = \frac{1}{n}\left[\left(\frac{n+1}{2} + x\right) B_{n-1}\left(x + \frac{1}{2}\right) + \left(\frac{n+1}{2} - x\right) B_{n-1}\left(x - \frac{1}{2}\right)\right]. \tag{32}$$

G. Approach by Potts (Dilated Sinc Functions)

The scheme in [42] (see [24]) uses the following couple of windows:

$$\begin{cases} \varphi(\xi) = (2\pi)^{-0.5} B_{2K-1}\left(\frac{Kc\xi}{\pi(2c-1)}\right) \\ \hat{\varphi}(\xi) = \frac{2c-1}{2cK} \mathrm{sinc}^{2K}\left(\frac{(2c-1)x}{2cK}\right). \end{cases} \tag{33}$$

H. Approach by Jackson et al. and by Fourmont (Kaiser–Bessel Windows)

Finally, the approach in [12] (see also [18]) exploits the Kaiser–Bessel windows, having the following expressions:

$$\varphi(\xi) = \begin{cases} I_0(K\sqrt{\alpha^2 - \xi^2}), & |\xi| \le \alpha \\ 0, & |\xi| > \alpha \end{cases} \tag{34}$$

where I_0 is the modified Bessel function of the zeroth order and

$$\tilde{\varphi}(x) = \sqrt{\frac{2}{\pi}} \frac{\sinh(\alpha\sqrt{K^2 - x^2})}{\sqrt{K^2 - x^2}}. \tag{35}$$

TABLE IV

COMPARISON OF THE OPTIMIZED ERROR FUNCTIONAL WITH THAT OF THE KAISER–BESSEL WINDOW, EMPLOYED IN [18]

c	K	χ_{opt}	Γ_{opt}	Γ Kaiser-Bessel
1.5	3	1.5	1.284	6.46
2	3	1.2	0.0942	0.671
1.5	6	1.1	1.18e-05	7.11e-05
2	6	1.1	1.00e-07	8.15e-07

VII. NUMERICAL PERFORMANCE OF THE APPROACH AND NUFFT-BASED EVALUATION OF THE FAR FIELD RADIATED BY PERIODIC AND APERIODIC ARRAYS

In this section, the numerical performance of the optimized NUFFT approach is analyzed and compared with the performance of the schemes recalled in the previous section.

First, it needs to be stressed that FFTs lead to exact evaluations of standard DFTs, whereas NUFFTs are approximations of the NUDFTs. However, by properly setting the shape of the window, the oversampling parameter c, and the support K of $\hat{\varphi}$, the NUFFT results can be drawn numerically closer and closer to the NUDFT. Second, the optimized NUFFT will be applied to a case of electromagnetic interest, namely, the radiation of linear regular or irregular arrays onto a set of regular or irregular spectral points. As already mentioned, because of widespread use of the Fourier transform and its discrete counterpart in electromagnetic applications, the present tool can be of unprecedented utility.

Optimization of the error functional in (18) has been performed by exploiting the fminsearch MATLAB function and by the progressive enlargement of the number of unknowns. For all the test cases in the following, the optimization has been initialized with $T = 1$, $\gamma_0 = 1$,and $\gamma_1 = 0$. Such a choice corresponds to a scaling window φ, coinciding with the PSWF of zeroth order, namely, to a window function very close to the Kaiser–Bessel window used in [18]. Actually, it has been found that by using a global optimization algorithm, the results have not appreciably improved. The optimization against χ has been performed by exhaustive search against that parameter.

Four test cases, with typical values of c and K, have been considered, leading to increased levels of accuracy.

1) *Case a)*: $c = 1.5$ and $K = 3$.
2) *Case b)*: $c = 2$ and $K = 3$.
3) *Case c)*: $c = 1.5$ and $K = 6$.
4) *Case d)*: $c = 2$ and $K = 6$.

The optimization results are reported in Table IV, along with the corresponding χ, leading to the least error.

Table IV shows that in all the cases, the obtained functional values are lower than those obtained for the Kaiser–Bessel window. Fig. 3 illustrates the comparison between the optimized window and the Kaiser–Bessel window for $c = 2$ and $K = 3$.

The expansion coefficient obtained for the optimized window functions, in terms of the Legendre polynomials, is reported in [43, Tables I–IV], for cases a)–d), respectively. It should be noted that the expansion coefficients in the mentioned Tables I–IV represent the only algorithm parameters needed to implement the NUFFTs at hand, once c and K have been fixed. Indeed, they are independent of the sample locations. In other words, the optimization that leads to the coefficients in Tables I–IV can be performed only once and offline. The variations in the number of Legendre coefficients involved in cases a)–d) are due to the differences in the values of c.

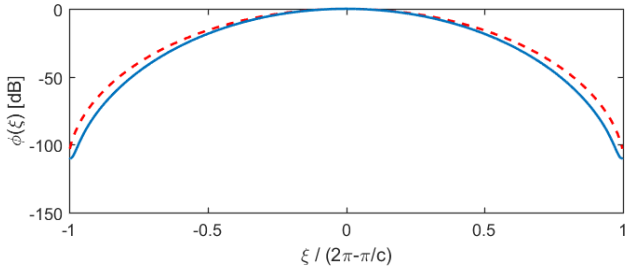


Fig. 3. Comparison between optimized window (blue solid line) and Kaiser–Bessel window (red dashed line).

After optimization, the scaling window φ, corresponding to Table IV, is considered to compare the proposed approach with those in Section VI and to examine the mentioned three test cases of electromagnetic interest.

1) Evaluation of the field, radiated by a periodic array onto a set of irregularly spaced spectral sampling points; in this case, the relevant NUFFT is the NER.
2) Evaluation of the field, radiated by an aperiodic array onto a set of regularly distributed spectral sampling points; in this case, the relevant NUFFT is the NED [5].
3) Evaluation of the field, radiated by an aperiodic array onto a set of irregularly spaced spectral sampling points; in this case, the relevant NUFFT is Type-3 [29].

It should be mentioned here that for all the comparisons in this paper, the accuracies are compared keeping the width of the interpolation window fixed. This is because the width of $\hat{\varphi}$ is a measure of both computational complexity and memory requirements of the approach (see Sections V and VI), and the comparison between different approaches has to be made for a fixed tradeoff between these requirements.

A. Comparison With the Approaches of Liu and Nguyen and of Kuo and Lee

The comparison is performed for different values of $K$and for $c = 2$, as integer c is a requirement of the method. For the proposed approach, the error is evaluated by a normalized, discrete version of the metrics in (18), namely,

$$\frac{1}{2x_M N}\sum_{k=-\frac{N}{2}-x_M}^{\frac{N}{2}-1}\int_{-x_M}^{x_M} dx \left| e^{-j2\pi x_l \frac{k}{N}} - \frac{(2\pi)^{-1/2}}{\varphi\left(\frac{2\pi k}{cN}\right)} \times \sum_{|m|\le K} \hat{\varphi}(x-(\lfloor x\rfloor + m))e^{-j(\lfloor x\rfloor+m)\xi}\right| \tag{36}$$

and for those in [13] and [20], by using (36) and substituting $\tilde{\varphi}$ for $\hat{\varphi}$.

The results are shown in Fig. 4 when $n = 1, 3$, and 5. The results show that the proposed approach outperforms those in [15] and [22]. Although what is shown in Fig. 4 relates to

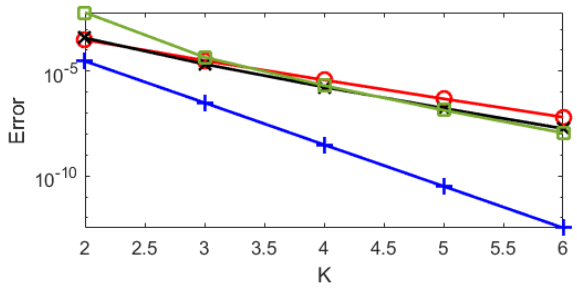


Fig. 4. Proposed approach against those in [15] and [22] for $c = 2$. Blue plus line: optimized approach. Red circle line: approach in [15]. Black cross line: approach in [22] with $n = 3$. Green square line: approach in [22] with $n = 5$.

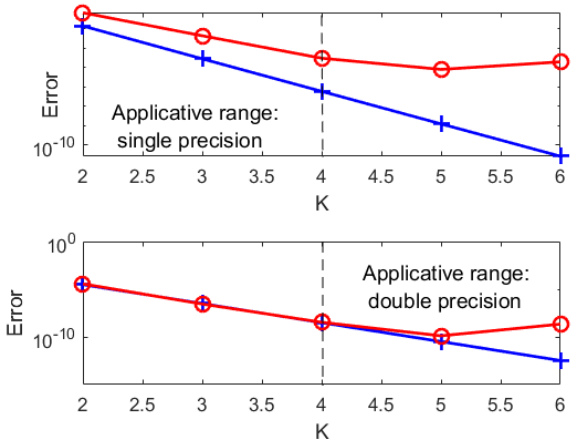


Fig. 5. Proposed approach against that in [17]. Blue plus line: optimized approach. Red circle line: approach in [17]. $c = 1.5$ (top). $c = 2$ (bottom).

the particular case of $N = 1024$ and $x_M = 70\pi$, the outperformance continues even for other choices of these parameters. It is relevant to point out here that increasing n does not significantly improve the accuracy; so, Kuo and Lee [22] do not significantly improve [15]. Finally, quoting [41], Liu and Nguyen [15] claim that "the cosine scaling factors ... are by no means the 'best' ones, and the problem of finding better scaling factors should be followed up." The results of the authors are consistent with this contention.

B. Comparison With the Approach of Fessler and Sutton

The comparison is performed for different values of K of the interpolation window support and for $c = 1.5$ and $c = 2$ (Fig. 5). The ranges of K, typically employed to reach single- or double-precision accuracy, are also indicated in the two panels of Fig. 5. The error is evaluated on the same lines as those of the foregoing section. Again, and without the loss of generality, the case $N = 1024$ and $x_M = 70\pi$ has been considered.

For this, the NUFFT routines of the authors, available at https://web.eecs.umich.edu/~fessler/code/index.html, have been exploited. More elaborately, the default routine, using a min–max interpolator with Kaiser–Bessel scaling factor, has been used.[3] Fessler and Sutton [17] have optimized the parameters of Kaiser–Bessel scaling factor only for the case $c = 2$.

From the results shown in Fig. 5, it can be seen that the proposed approach outperforms the approach in [17] for $c = 1.5$. On the other hand, for $c = 2$, the performance is very close to $K = 4$. It should be, however, noted that oversampling of $c = 2$ has been used to achieve double precision accuracy; so, it is typically employed for connection to "large" (e.g., 6) values of K. Actually, beyond that value, the proposed algorithm outperforms again the compared one.

[3]nufft_init routine with minmax:kb option.

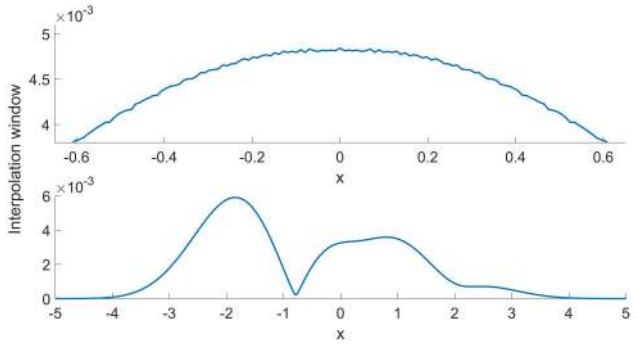


Fig. 6. Particulars of the interpolation window for the approach in [23] and [25] for $K = 2$, $N = 64$, $O = 101$, $R = 6$, and $c = 68/64$ (top) and $K = 10$, $N = 64$, $O = 101$, $R = 6$, and $c = 2$ (bottom).

C. Comparison With the Approach of Yang and Jacob

Unfortunately, even by using the authors' own code, available at https://research.engineering.uiowa.edu/cbig/content/accurate-nufft-using-optimized-interpolator-and-scale-factors, no meaningful results could be obtained.

Fig. 6 (top) shows the details of the interpolation window displayed in [25, Fig. 2(a)], namely, for $K = 2$, $N = 64$, $O = 101$, $R = 6$, and $c = 68/64$. From Fig. 6 (top), it can be seen that the interpolation window shows oscillating behavior, which is perhaps due to the ill-conditioning of the window optimization process. On the other hand, Fig. 6 (bottom) depicts the interpolation window obtained for the case when $K = 10$, $N = 64$, $O = 101$, $R = 6$, and $c = 2$. From Fig. 6 (bottom), it can be seen that the window is asymmetrical, with a null in the center. This is perhaps due to the lack of convergence of the optimization algorithm. Here, the comments of Yang and Jacob [25] are worth recalling: "We currently do not have guarantees for the convergence of this algorithm to the global minimum of the cost function." It should be noted that even with "high-order" B-Splines (say, 5 or 6), optimizing for $2O + 1$ parameters is not an easy task [44].

D. Comparison With Other Windows

The results of the comparison between the proposed scheme and those in Sections VI-E–VI-H are depicted in Fig. 7 for the cases $c = 1.5$ (top) and $c = 2$ (bottom), respectively. From these figures, it can be seen, once again, that the proposed method outperforms the compared ones.

E. Radiation of a Periodic Array Onto a Set of Irregularly Spaced Spectral Sampling Points

Let a periodic array comprising N elements spaced at a distance of d be assumed. The array factor can be written as

$$f(u) = \sum_{k=0}^{N-1} c_k e^{jku} \tag{37}$$

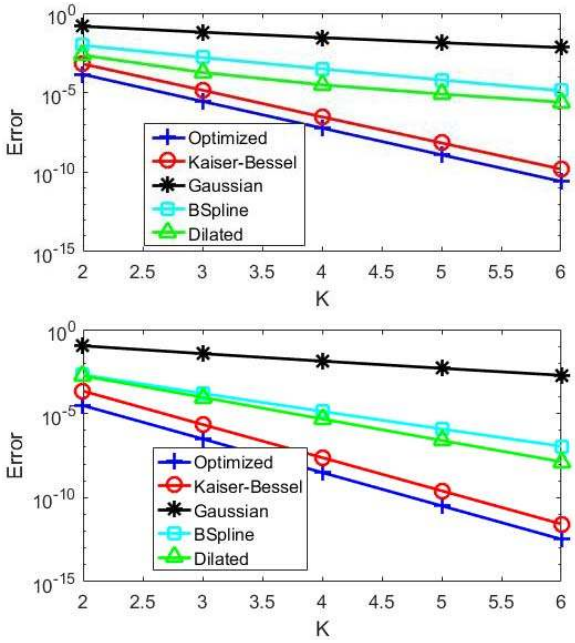


Fig. 7. Proposed approach against those employing the window functions described in Sections VI-E–VI-H. Case $c = 1.5$ (top). Case $c = 2$ (bottom).

where $u = \beta d\cos\psi$, β is the wavenumber of the embedding medium, $\{c_k\}_{k=0}^{N-1}$ are the complex element excitation coefficients, and ψ is the angle between the array axis and the observation direction. If the array factor is computed at $\{u_l\}_{l=0}^{M-1}$, then the samples of f can be computed by an NER NUFFT, evaluated at the points $\{(2\pi/N)u_l\}_{l=0}^{M-1}$.

A comparison of the performance of the optimized window of the Kaiser–Bessel window, proposed in [18], with that of the Gaussian window, proposed in [13] and [19], for the implementation of an NER NUFFT is now relevant. Although from the previous sections, the approach of Fessler and Sutton [17] has been found to be by far the best one, it should be noted that it is available only for the NER NUFFT case. Therefore, Kaiser–Bessel, Gaussian, and optimized windows will be compared here, thanks to the possibility of easily setting up all the three kinds of NUFFT.

To perform the comparisons, 100 realizations of arrays and spectral sampling points, with randomly chosen excitations, have been addressed. The array elements are located onto a segment 40λ sized, with a spacing of $\lambda/2$, i.e., $N = 80$, and an equal number of spectral points have been considered. For different realizations, the complex element excitations have been randomly selected with Gaussian distributions for real and imaginary parts, and the spectral points according to a uniform distribution in $(-\beta_0 d, \beta_0 d)$, d being the interelement distance. Table V reports the average rms and maximum errors for Kaiser–Bessel, Gaussian, and optimized windows, for the four considered cases a)–d). It can be seen that by controlling c and K, it is possible to progressively increase the accuracy of all the NUFFT algorithms, and that the algorithm, relating to the optimized window, shows the least error. Furthermore, the accuracy of the optimized NUFFT can be driven very close to machine precision.

TABLE V
RADIATION BY A PERIODIC ARRAY ONTO AN IRREGULAR SET OF SPECTRAL POINTS: AVERAGE RMS AND MAXIMUM ERRORS

Case	K-B	Gaussian	Optimized
RMSa	2.23e-3	2.10e-1	4.65e-4
RMSb	2.85e-4	5.13e-2	4.29e-5
RMSc	3.42e-8	1.39e-3	6.07e-9
RMSd	3.99e-10	6.63e-5	6.39e-11
Maxa	9,27e-4	8.18e-2	2.47e-4
Maxb	1.21e-4	1.80e-2	3.24e-5
Maxc	1.30e-8	4.82e-4	3.47e-9
Maxd	1.49e-10	2.47e-5	2.56e-11

TABLE VI
RADIATION BY AN APERIODIC ARRAY ONTO A REGULAR SET OF SPECTRAL POINTS: AVERAGE RMS AND MAXIMUM ERRORS

Case	K-B	Gaussian	Optimized
RMSa	2.18e-3	2.10e-1	4.38e-4
RMSb	2.85e-4	5.25e-2	4.34e-5
RMSc	3.36e-8	1.28e-3	6.13e-9
RMSd	4.00e-10	8.23e-4	6.19e-11
Maxa	3.14e-3	2.34e-1	7.11e-4
Maxb	3.84e-4	5.52e-2	6.97e-5
Maxc	4.40e-8	2.18e-3	1.12e-8
Maxd	4.47e-10	6.72e-4	1.05e-10

F. Radiation of an Aperiodic Array Onto a Set of Regularly Spaced Spectral Sampling Points

Let an aperiodic array comprising N elements located at the points $\{x_k\}_{k=0}^{N-1}$ be considered now. The array factor can be written as

$$f(\bar{u}) = \sum_{k=0}^{N-1} c_k e^{jx_k\bar{u}} \tag{38}$$

where $\bar{u} = \beta\cos\psi$. If the array factor is computed at a regular set of M points in $\bar{u}$ variable, then the samples of f can be computed by NED-NUFFT [5]. Furthermore, 100 realizations of aperiodic arrays have been considered. The array elements are randomly located, according to a uniform distribution, onto a segment measuring $40\lambda$with $N = 80$, and an equal number of spectral sampling points have been dealt with for different realizations, the complex element excitation coefficients have been randomly selected with Gaussian distributions for real and imaginary parts. Table VI reports the average rms errors and the maximum errors for the three windows. It can be seen that, once again, the accuracy of the optimized NUFFT runs toward the machine precision.

For the case of aperiodic arrays, the effect of the approximations involved in the NUFFT algorithm on the radiated pattern has also been examined. In particular, for the sake of illustration, an aperiodic array of 34 elements, synthesized by the algorithm in [34] and [35] for a choice of $c = 1.5$ and $K = 1$, has been considered. Fig. 8 illustrates the comparison between the exact radiated field and that evaluated by the NUFFT algorithm, based on the Gaussian (top), Kaiser–Bessel

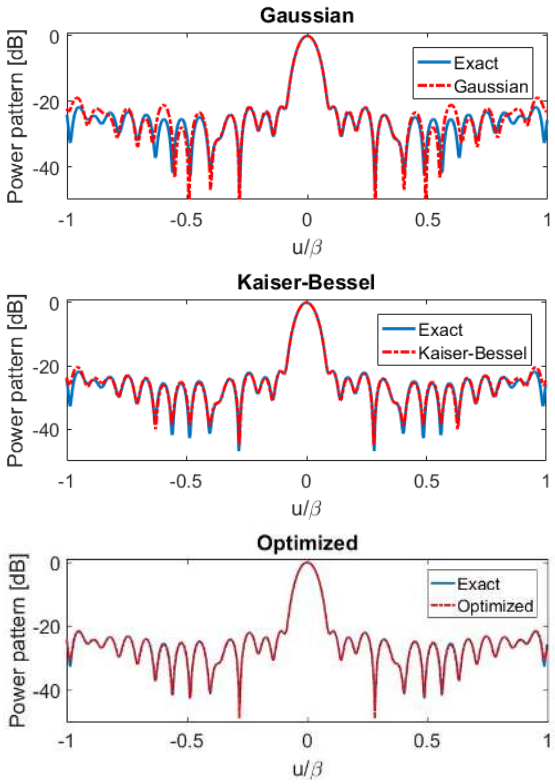


Fig. 8. Field radiated by the aperiodic array (blue solid line), as evaluated by an NUFFT (red dashed-dotted line), using Gaussian (top), Kaiser–Bessel (middle), or optimized (bottom) windows.

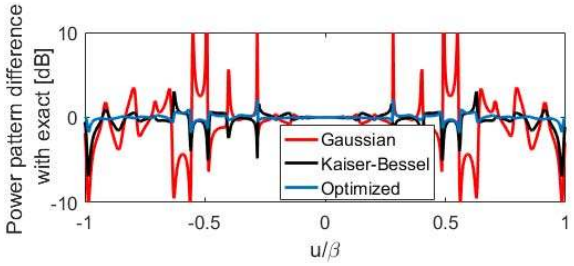


Fig. 9. Differences among the exact evaluations of the field radiated by the 34-element aperiodic array and those evaluated by the three compared NUFFTs.

(middle), and optimized (bottom) windows, respectively. The differences between the exact pattern and the evaluated patterns are shown in Fig. 9. It can be seen that even with significantly low values for c and K, the accuracy of the optimized NUFFT appears to be very satisfactory, which is at variance with the other NUFFT implementations. Having low values of c and K, of course, reduces the computational complexity of the approach.

G. Radiation of an Aperiodic Array Onto a Set of Irregularly Spaced Spectral Sampling Points

Finally, let an aperiodic array comprising N elements be assumed to have been located at points $\{x_k\}_{k=0}^{N-1}$, so that the array factor can be written again as in (23). If the array factor is computed at an irregular set of M sampling points in $\bar{u}$ variable, namely, $\{\bar{u}_l\}_{l=0}^{M-1}$, then the samples of f can be calculated by Type-3 NUFFT [5].

TABLE VII
RADIATION BY AN APERIODIC ARRAY ONTO AN IRREGULAR SET OF SPECTRAL POINTS: AVERAGE RMS AND MAXIMUM ERRORS

Case	K-B	Gaussian	Optimized
RMSa	2.43e-3	2.19e-1	5.10e-4
RMSb	3.81e-4	5.56e-2	5.65e-5
RMSc	3.42e-8	1.56e-3	6.54e-9
RMSd	4.42e-10	7.75e-5	7.28e-11
Maxa	3.81e-3	2.19e-1	8.16e-4
Maxb	3.61e-4	4.47e-2	6.57e-5
Maxc	5.21e-8	2.31e-3	1.41e-8
Maxd	4.90e-10	9.46e-5	1.39e-10

To appreciate the performance of the optimized window for the implementation of Type-3 NUFFT, the use of the Kaiser–Bessel window has been extended to this case also, following the same approach as the one in Section III.

Again, 100 realizations of aperiodic arrays have been considered. The array elements are randomly located, according to a uniform distribution, onto a segment 40λsized, with $N = 80$. An equal number of spectral points have been considered. For different realizations, the complex excitations have been randomly selected with Gaussian distributions for real and imaginary parts, and the spectral points according to the uniform distribution in $(-\beta, \beta)$. Table VII shows the average and maximum rms errors for the three windows. The results again show that the accuracy has progressively increased by changing c and K, and that the optimized NUFFT has led to the most accurate results, which are close to the machine precision.

VIII. EXTENSION TO HIGHER DIMENSIONS

Finally, we sketch the extension of our approach to higher dimensions. Multidimensional NUFFTs have already been proposed in the literature (see [18], [45] for Kaiser–Bessel case, [17] for the approach of Fessler and Sutton, and [21] for a general discussion). The extension to higher dimensions for the proposed approach can be done by following the ideas in the literature, and for other approaches, following the same tradeoff (accuracy/computation/memory).

In higher dimensions, particularly, the key function to be represented is a complex exponential, representing a multivariate Fourier kernel that can be factored into complex exponentials, representing single variable Fourier kernels. Thanks to factorization, the representation at hand amounts to that of individual complex exponentials. The analysis contained in this paper, which considers individual complex exponential function, can then be immediately applied to the multidimensional case also.

To recall higher dimensional extensions of NUFFT algorithms, the 2-D NED NUFFT case is considered here, by way of an example, but without the loss of generality.

The corresponding NUDFT is defined as

$$\hat{z}_{kp} = \sum_{l=0}^{M-1} z_l e^{-j2\pi x_l \frac{k}{N}} e^{-j2\pi y_l \frac{p}{N}}, \quad k, p = 0, \ldots, N-1. \tag{39}$$

In Section II, the problem of accurately estimating the exponential function $e^{-j2\pi x_l(k/N)}$, by using properly defined window functions, has been examined. The same arguments can obviously be applied to the representation of the product $e^{-j2\pi x_l(k/N)}e^{-j2\pi y_l(p/N)}$. This idea corresponds to using windows $\varphi_{2D}(\xi, \eta)$ and $\hat{\varphi}_{2D}(x, y)$, which are factorized into the product of those exploited in the 1-D case, namely, $\varphi_{2D}(\xi, \eta) = \varphi(\xi)\varphi(\eta)$ and $\hat{\varphi}_{2D}(x, y) = \hat{\varphi}(x)\hat{\varphi}(y)$. Accordingly, optimized 1-D window functions are readily made available to be employed in the multidimensional case.

It is to be noted that for a 2-D problem, the interpolation involves $(2K+1)^2$ terms, and, similarly, for a d-dimensional problem, $(2K+1)^d$ terms. Consequently, for higher dimensional problems, the need to reach accuracy with small values of $(2K+1)$ becomes more compelling. As before, the problem concerns the tradeoff between operation count and memory storage.

In the "precomputed mode," and concerning the memory requirements, the factorization of window functions enables storing of 1-D arrays only. Indeed, each variable kernel requires $(2K+2)N$ terms to be stored (see Table I), and thus for d exponentials, the memory requirement would be $d(2K+2)N$. Accordingly, the memory storage increases linearly with increase in the problem dimensions d, and not with the power of d, as stressed in the discussion in [21]. Conversely, in the "reduced memory mode" and concerning the operations count, each kernel requires $\mathcal{O}((2K+1)^2N)$ operations (see Table I), and thus for d exponentials, the resulting operations count would be $\mathcal{O}(d(2K+1)^2N)$. Accordingly, the computational burden increases again linearly with increase in the problem dimensions d, and not with the power of d, as stressed in the discussion in [21].

In both cases, a linear storage strategy is paid in terms of complex multiplications between the involved 1-D windows. Indeed, due to the required tensor multiplications between the window functions, the number of multiplications increases with the power of d, again as underlined in [21]. Such tradeoff is common to all the compared approaches, detailed in Section VI.

Of course, the performance of the multidimensional extension of our approach is expected to be more accurate than that of the approaches detailed in Section VI, because our representation of each exponential factor is more accurate, as stressed in Section VII. Furthermore, from the above discussion, it emerges that the multidimensional extension of our approach does not introduce unusual computational/memory trends as the extensions of other schemes do.

IX. Conclusion

This paper deals with the development of an optimized approach for the implementation of NUFFT algorithms, based on a general and new perspective.

First, the NUFFT approach has been extended to Type-3 NUDFTs by using the Gaussian windows as general windows. Second, the choice of the window function employed in NER, NED, and Type-3 NUFFTs has been optimized to obtain more accurate results than those available in the literature.

The computational costs and the memory requirements of the proposed schemes have been theoretically analyzed, and their very convenient performance assessed, theoretically and numerically, by comparing them with a number of relevant and popular approaches in the literature. The proposed method has proved to be more accurate than the compared schemes, without burdening the computational and memory requirements.

Notably, the IEEE 754 double-precision floating-point arithmetic has a precision of at least 15 significant digits, or, at the most, 17 significant digits (16 on average) [46], [47]. A maximum error of 10^{-11} (see Table VI) means accuracy, approaching the best one possible in double-precision arithmetics. Of course, as the accuracy of the method approaches that of the IEEE 754 double precision, no further improvement in accuracy is warranted and, that being so, the task ahead is how to reduce the operations count and memory storage, without affecting that accuracy.

Finally, the computation of window functions amounts to that of a Legendre polynomial expansion, namely, a simple polynomial evaluation. This is convenient from the viewpoint of not only computational burden but also proper arrangement of calculations.

The developed "optimized" NUFFTs have been applied to a case of electromagnetic interest, i.e., the radiation of linear regular or irregular arrays onto sets of regular or irregular spectral points. Obviously, NUFFT algorithms are important for a host of other electromagnetic applications too [1]–[10].

From the presented analysis, it emerges that:

1) The performance of the "optimized" NUFFT is superior to that of the currently available NUFFT approaches;
2) The accuracy of the "optimized" NUFFT reaches machine precision earlier than that of the other NUFFTs, with which it has been compared, for increasing the values of c and K;
3) Even for low values of c and K, the "optimized" NUFFT is capable of yielding satisfactory results; of course, reducing the values of c and K helps in reducing the computational complexity of the approach.

The guidelines for multidimensional extension of the proposed approach have also been discussed.

As future developments, we plan to extend the approach to the 2-D case and apply it to other cases of electromagnetic interest [6]–[10], using parallelization on graphics processing units. Here, no particular expedient has been adopted to reduce the numerical errors in the operations involved in the NUFFT calculations. Therefore, for further improvement of numerical accuracy in the future, it is planned to employ techniques, like Kahan summation algorithm [47]. Finally, the NUFFT could be used for calculating the field, radiated by arrays, even in the case of nonnegligible mutual coupling [48].

APPENDIX
CALCULATION OF THE PSWFS

The algorithm in [21] expands each PSWF into a number K_{leg} of the Legendre polynomials, where

$$\begin{cases} K_{leg} = (2\lfloor ew \rfloor + 1) + \log_2\left(\frac{1}{\varepsilon}\right), & N < \lfloor \frac{2\sigma}{\pi} \rfloor \\ K_{leg} = (2\lfloor ew \rfloor + 1) + \log_2\left(\frac{1}{\varepsilon}\right) + \log_2\left(\frac{1}{\bar{\lambda}}\right), & \text{otherwise} \end{cases} \tag{A.1}$$

$$\bar{\lambda} = e^{\left(N+\frac{1}{2}\right)\left(\log\left(\frac{ew}{4}\right) - \log\left(N+\frac{1}{2}\right)\right)} \tag{A.2}$$

where N is the number of PSWFs to be generated, w is the SBP, and ε is the required accuracy. In other words

$$\Psi_s[c,\xi] = \sum_{k=0}^{(K_{leg}-1)/2} d_{2k}^{s} P_{2k}(\xi), \quad |\xi| < \frac{\pi}{c} \tag{A.3}$$

where P_k denotes the kth Legendre polynomials, so that

$$\varphi(\xi) = \sum_{k=0}^{(K_{leg}-1)/2} \left\{ \sum_{t=0}^{T} d_{2k}^{2t} \gamma_t \right\} P_{2k}(\xi), \quad |\xi| < \frac{\pi}{c}. \tag{A.4}$$

In (A.3) and (A.4), $K_{leg} - 1$ is rounded off to the closest larger even integer. According to (A.4), the window function φ can be regarded as having been expanded, in terms of the Legendre polynomials, as

$$\varphi(\xi) = \sum_{k=0}^{(K_{leg}-1)/2} d_{2k} P_{2k}(\xi) \tag{A.5}$$

where d_k's are the expansion coefficients.

As regards $\hat{\varphi}$, it can be calculated by considering that the Fourier transform of the Legendre polynomials is given by

$$\int_{-1}^{1} P_k(\xi) e^{j\xi x} = j^k \sqrt{\frac{2\pi}{x}} J_{k+\frac{1}{2}}(x) \tag{A.6}$$

where $J_p(x)$ is the Bessel function of pth order.

REFERENCES

[1] J. A. Fessler, "Model-based image reconstruction for MRI," *IEEE Signal Process. Mag.*, vol. 27, no. 4, pp. 81–89, Jul. 2010.

[2] A. Camps, M. Vall-llossera, I. Corbella, F. Torres, and N. Duffo, "Angular and radiometric resolution of Y-shaped nonuniform synthetic aperture radiometers for earth observation," *IEEE Geosci. Remote Sens. Lett.*, vol. 5, no. 4, pp. 793–795, Oct. 2008.

[3] Q. H. Liu, X. M. Xu, B. Tian, and Z. Q. Zhang, "Applications of nonuniform fast transform algorithms in numerical solutions of differential and integral equations," *IEEE Trans. Geosci. Remote Sens.*, vol. 38, no. 4, pp. 1551–1560, Jul. 2000.

[4] G. Mazzarella and G. Panariello, "Pattern synthesis of conformal arrays," in *Proc. IEEE Int. Symp. Antennas Propag.*, Ann Arbor, MI, USA, Jun./Jul. 1993, pp. 1054–1057.

[5] A. Capozzoli, C. Curcio, G. D'Elia, A. Liseno, and P. Vinetti, "Fast CPU/GPU pattern evaluation of irregular arrays," *Appl. Comput. Electromagn. Soc. J.*, vol. 25, no. 4, pp. 355–372, Apr. 2010.

[6] A. Capozzoli, C. Curcio, A. Liseno, and G. Toso, "Phase-only synthesis of flat aperiodic reflectarrays," *Prog. Electromagn. Res.*, vol. 133, pp. 53–89, Oct. 2013.

[7] A. Fanti and G. Mazzarella, "A finite dsifference polar-Cartesian grid approach for mode computation in rounded-end waveguides," *Appl. Comput. Electromagn. Soc. J.*, vol. 26, no. 9, pp. 768–775, Sep. 2011.

[8] A. Capozzoli, C. Curcio, and A. Liseno, "NUFFT-accelerated plane-polar (also phaseless) near-field/far-field transformation," *Prog. Electromagn. Res. M*, vol. 27, pp. 59–73, Nov. 2012.

[9] A. Capozzoli, C. Curcio, A. Di Vico, and A. Liseno, "NUFFT-&GPU-based fast imaging of vegetation," *IEICE Trans. Commun.*, vol. 94-B, no. 7, pp. 2092–2103, Jul. 2011.

[10] A. Capozzoli, C. Curcio, and A. Liseno, "Fast GPU-based interpolation for SAR backprojection," *Prog. Electromagn. Res.*, vol. 133, pp. 259–283, 2013.

[11] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*. Cambridge, MA, USA: MIT Press, 2009.

[12] J. I. Jackson, C. H. Meyer, D. G. Nishimura, and A. Macovski, "Selection of a convolution function for Fourier inversion using gridding," *IEEE Trans. Med. Imag.*, vol. 10, no. 3, pp. 473–478, Sep. 1991.

[13] A. Dutt and V. Rokhlin, "Fast Fourier transforms for nonequispaced data," *SIAM J. Sci. Comput.*, vol. 14, no. 6, pp. 1368–1393, 1993.

[14] G. Beylkin, "On the fast Fourier transform of functions with singularities," *Appl. Comput. Harmon. Anal.*, vol. 2, no. 4, pp. 363–381, Oct. 1995.

[15] Q. H. Liu and N. Nguyen, "An accurate algorithm for nonuniform fast Fourier transforms (NUFFT's)," *IEEE Microw. Guided Wave Lett.*, vol. 8, no. 1, pp. 18–20, Jan. 1998.

[16] G. Steidl, "A note on fast Fourier transforms for nonequispaced grids," *Adv. Comput. Math.*, vol. 9, no. 3, pp. 337–352, 1998.

[17] J. A. Fessler and B. P. Sutton, "Nonuniform fast Fourier transforms using min-max interpolation," *IEEE Trans. Signal Process.*, vol. 51, no. 2, pp. 560–574, Feb. 2003.

[18] K. Fourmont, "Non-equispaced fast Fourier transforms with applications to tomography," *J. Fourier Anal. Appl.*, vol. 9, no. 5, pp. 431–450, Sep. 2003.

[19] L. Greengard and J.-B. Lee, "Accelerating the non-uniform fast Fourier transform," *SIAM Rev.*, vol. 46, no. 3, pp. 443–454, 2004.

[20] J.-Y. Lee and L. Greengard, "The type 3 nonuniform FFT and its applications," *J. Comput. Phys.*, vol. 206, no. 1, pp. 1–5, Jun. 2005.

[21] S. Kunis and D. Potts, "Time and memory requirements of the nonequispaced FFT," *Sampling Theory Signal Image Process.*, vol. 7, no. 1, pp. 77–100, Jan. 2008.

[22] J. T. Kuo and H. Y. Lee, "Improved accuracy factors for the nonuniform fast fourier transform (NUFFT) algorithm," *IEEE Microw. Wireless Compon. Lett.*, vol. 19, no. 1, pp. 3–5, Jan. 2009.

[23] M. Jacob, "Optimized least-square nonuniform fast Fourier transform," *IEEE Trans. Signal Process.*, vol. 57, no. 6, pp. 2165–2177, Jun. 2009.

[24] J. Keiner, S. Kunis, and D. Potts, "Using NFFT 3—A software library for various nonequispaced fast Fourier transforms," *ACM Trans. Math. Softw.*, vol. 36, no. 4, pp. 19-1–19-30, Aug. 2009.

[25] Z. Yang and M. Jacob, "Mean square optimal NUFFT approximation for efficient non-Cartesian MRI reconstruction," *J. Magn. Res.*, vol. 242, pp. 126–135, May 2014.

[26] D. Slepian and H. O. Pollak, "Prolate spheroidal wave functions, fourier analysis and uncertainty—I," *Bell. Syst. Tech. J.*, vol. 40, no. 1, pp. 43–63, Jan. 1961.

[27] H. J. Landau and H. O. Pollak, "Prolate spheroidal wave functions, fourier analysis and uncertainty—III: The dimension of the space of essentially time- and band-limited signals," *Bell Syst. Tech. J.*, vol. 41, pp. 1295–1336, Jul. 1962.

[28] B. R. Frieden and E. Wolf, Ed., "Evaluation, design and extrapolation methods for optical signals, based on use of the prolate functions," in *Progress in Optics*, vol. 9. Amsterdam, The Netherlands: North-Holland, 1971, pp. 311–407.

[29] A. Capozzoli, C. Curcio, A. Liseno, and A. Riccardi, "Parameter selection and accuracy in type-3 non-uniform FFTs based on Gaussian gridding," *Prog. Electromagn. Res.*, vol. 142, pp. 743–770, 2013.

[30] M. Frigo and S. G. Johnson, "The design and implementation of FFTW3," *Proc. IEEE*, vol. 46, no. 2, pp. 216–231, Feb. 2005.

[31] O. M. Bucci, G. D'Elia, and R. Pierri, "Numerical evaluation of antennas field using a new aperture-like expansion," in *Proc. 10th Eur. Microw. Conf.*, Warszawa, Poland, Sep. 1980, pp. 67–71.

[32] G. D'Elia, R. Pierri, "Iterative method and sampling techniques in the analysis of shaped reflectors," in *Proc. 11th Eur. Microw. Conf.*, Amsterdam, The Netherlands, Sep. 1981, pp. 535–539.

[33] O. M. Bucci and G. Franceschetti, "Radiation from reflector antennas: Exact aperture and aperture-like approaches," *Radio Sci.*, vol. 16, no. 6, pp. 1101–1104, Nov. 1981.

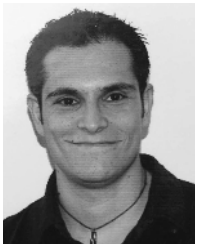
[34] A. Capozzoli, C. Curcio, G. D'Elia, A. Liseno, and P. Vinetti, "FFT & Aperiodic arrays with phase-only control and constraints due to super-directivity, mutual coupling and overall size," in *Proc. 30th ESA Antenna Workshop Antennas Earth Observ., Sci., Telecommun. Navigat. Space Missions*, Noordwijk, The Netherlands, May 2008, pp. 213–216.

[35] A. Capozzoli, C. Curcio, G. D'Elia, A. Liseno, and P. Vinetti, "FFT & Equivalently tapered aperiodic arrays," in *Proc. 29th Gen. Assem. Int. Union Radio Sci.*, Chicago, IL, USA, Aug. 2008, pp. 1–4.
[36] D. Slepian, "Some asymptotic expansions for prolate spheroidal wave functions," *J. Math. Phys.*, vol. 44, nos. 1–4, pp. 99–140, Apr. 1965.
[37] R. M. Trigub and E. S. Belinsky, *Fourier Analysis and Approximation of Functions*. Dordrecht, The Netherlands: Springer, 2004.
[38] H. Xiao, V. Rokhlin, and N. Yarvin, "Prolate spheroidal wavefunctions, quadrature and interpolation," *Inverse Problems*, vol. 17, no. 4, pp. 805–838, Aug. 2001.
[39] S. Zhang and J. Jin, *Computation of Special Functions*. New York, NY, USA: Wiley, 1996.
[40] F. W. J. Olver, D. W. Lozier, R. F. Boisvert, and C. W. Clark, Eds., *NIST Handbook of Mathematical Functions*. Cambridge, U.K.: Cambridge Univ. Press, 2010.
[41] N. Nguyen and Q. H. Liu, "The regular Fourier matrices and nonuniform fast Fourier transforms," *SIAM J. Sci. Comput.*, vol. 21, no. 1, pp. 283–293, 1999.
[42] D. Potts, "Schnelle Fourier-transformationen für nichtäquidistante Daten und Anwendungen," Ph.D. dissertation, Habilitation, Technisch-Naturwissenschaftliche Fakultät, Univ. Lübeck, Lübeck, Germany, 2003.
[43] *Legendre Coefficients Representing the Optimized Window*. Accessed: Aug. 27, 2018. [Online]. Available: https://archive.org/details/LegendreCoefficientsRepresentingTheOptimizedWindow
[44] A. Capozzoli and G. D'Elia, "Global optimization and antennas synthesis and diagnosis, part one: Concepts, tools, strategies and performances," *Prog. Electromagn. Res.*, vol. 56, pp. 195–232, 2006.
[45] A. Capozzoli, C. Curcio, A. Liseno, and G. Toso, "Fast, phase-only synthesis of aperiodic reflectarrays using NUFFTs and CUDA," *Prog. Electromagn. Res. A*, vol. 156, pp. 83–103, 2016.
[46] W. Kahan. *Lecture Notes on the Status of IEEE Standard 754 for Binary Floating-Point Arithmetic*. Accessed: Aug. 27, 2018. [Online]. Available: https://people.eecs.berkeley.edu/~wkahan/ieee754status/IEEE754.PDF
[47] N. J. Higham, "The accuracy of floating point summation," *SIAM J. Sci. Comput.*, vol. 14, no. 4, pp. 783–799, 1993.
[48] Y. Liu, X. Huang, K. D. Xu, Z. Song, S. Yang, and Q. H. Liu, "Pattern synthesis of unequally spaced linear arrays including mutual coupling using iterative FFT via virtual active element pattern expansion," *IEEE Trans. Antennas Propag.*, vol. 65, no. 8, pp. 3950–3958, Aug. 2017.

Amedeo Capozzoli (M'13) received the Laurea degree (*summa cum laude*) in electronic engineering and the Ph.D. degree in electronic engineering and computer science from the Università di Naples Federico II, Naples, Italy.

Since 2005, he has been an Associate Professor of electromagnetic fields with the Università di Naples Federico II, where he has been the Chair of the bachelor's and master's degree studies in telecommunication engineering since 2016, where he is currently the Chair of the Microwave and Millimeter Wave Laboratory and the Numerical Electromagnetics Laboratory. His current research interests include, among others, methods to extract the synthetic information on systems of sources or scatterers from field data, adaptive optics in optical astronomy, antenna synthesis and diagnosis, fast numerical methods in electromagnetics, graphics processing unit computing in electromagnetics, advanced measurement approaches in electromagnetics, inverse problems, and remote sensing.

Dr. Capozzoli won the Open National Competition for a post of Researcher at the Università di Naples Federico II in 1999. He received the Telecom Italia Prize for the best degree dissertation in electronic engineering defended at the Università di Naples Federico II, the Barzilai Prize for young scientists from the Italian Society of Electromagnetism in 2002, the Best Technical Paper Award from the Antenna Measurement Technique Association in 2009 and 2010, the Honorable Mention at the 5th European Conference on Antennas and Propagation (EUCAP 2011), the Nomination for the Best Paper Award at the 8th European Conference on Antennas and Propagation (EUCAP 2014), the Senior Membership from the Antenna Measurement Technique Association in 2015, and the 2016 Best Italian EMC Poster Price at the IEEE EMC Young Professional Paolo Corona Day in 2016. Since 2013, he has been responsible for the Course on Antenna Synthesis in the framework of the European School of Antennas. He is one of the founders, and the Chair, of the Italian AMTA node, the first European node of the Antenna Measurement Technique Association. He is an Associate Editor of the *Express Journal of the Applied Computational Electromagnetic Society*.



Claudio Curcio received the Laurea degree (*summa cum laude*) in electronic engineering and the Ph.D. degree in electronic and telecommunication engineering from the Università di Naples Federico II, Naples, Italy, in 2002 and 2005, respectively.

From 2006 to 2007, he was a Post-Doctoral Researcher with the University of Naples Federico II, where he is currently a Researcher. His current research interests include antenna measurements, standard and phaseless effective near-field/far-field transformation techniques, and optical beamforming techniques for array antennas and reflectarray synthesis.

Dr. Curcio was received the Optimus Award at the SIMAGINE 2002 "Worldwide GSM & Java Card Developer Contest" in 2002, the Best Technical Paper Award at the Antenna Measurement Techniques Association Symposium, in 2009 and 2010, the Honorable Mention for the Best Antenna Measurement Paper at the European Conference on Antennas and Propagation in 2011, the Finalist for the Antenna Measurements Best Paper Award at the European Conference on Antennas and Propagation in 2013, and the IEEE Best Italian EMC Poster Prize from the IEEE EMC Society, Italy Chapter, in 2016.

Angelo Liseno received the Laurea (*summa cum laude*) and the Ph.D. degrees in electrical engineering from the Seconda Università di Naples, Naples, Italy, in 1998 and 2001, respectively.

From 2001 to 2002, he was a Post-Doctoral Researcher with the Seconda Università di Naples. From 2003 to 2004, he was a Research Scientist with the Institut für Hochfrequenztechnik und Radarsysteme of the Deutsches, Zentrum für Luft- und Raumfahrt, Oberpfaffenhofen, Germany. From 2005 to 2015, he has been a Researcher with the Dipartimento di Ingegneria Elettrica e delle Tecnologie dell'Informazione, Università di Naples Federico II, Naples, Italy, where he has been an Associate Professor since 2015. His current research interests include parallel computing techniques for electromagnetics, general purpose graphics processing unit computing, complex and phaseless near-field/far-field transformation techniques, antenna synthesis, remote sensing, inverse problems, inverse scattering, tomography, and imaging.

Dr. Liseno received the Antenna Measurement Techniques Association Best Technical Paper Award in 2009 and 2010, the Honorable Mention for the Best Antenna Measurement Paper at the European Conference on Antennas and Propagation in 2011, the finalist for the "Measurements" Best Paper Award at the European Conference on Antennas and Propagation in 2013, and the Best Italian EMC Poster Prize from the IEEE EMC Society, Italy Chapter, in 2016.