New Density Estimation Methods for Charged Particle Beams
With Applications to Microbunching Instability∗

Balsa Terzić†
Jefferson Lab, Newport News, Virginia 23606, USA

Gabriele Bassi‡
Brookhaven National Laboratory, Upton, New York 11973, USA

In this paper we discuss representations of charge particle densities in particle-in-cell (PIC) simulations, analyze the sources and profiles of the intrinsic numerical noise, and present efficient methods for their removal. We devise two alternative estimation methods for charged particle distribution which represent significant improvement over the Monte Carlo cosine expansion used in the 2d code of Bassi et al. [1, 2], designed to simulate coherent synchrotron radiation (CSR) in charged particle beams. The improvement is achieved by employing an alternative beam density estimation to the Monte Carlo cosine expansion. The representation is first binned onto a finite grid, after which two grid-based methods are employed to approximate particle distributions: (i) truncated fast cosine transform (TFCT); and (ii) thresholded wavelet transform (TWT). We demonstrate that these alternative methods represent a staggering upgrade over the original Monte Carlo cosine expansion in terms of efficiency, while the TWT approximation also provides an appreciable improvement in accuracy. The improvement in accuracy comes from a judicious removal of the numerical noise enabled by the wavelet formulation. The TWT method is then integrated into Bassi’s et al. [1] CSR code, and benchmarked against the original version. We show that the new density estimation method provides a superior performance in terms of efficiency and spatial resolution, thus enabling high-fidelity simulations of CSR effects, including microbunching instability.

PACS numbers: 07.05.Tp, 52.65.Rr 02.60.Gf, 41.75.-i,

I. INTRODUCTION

When a charged particle beam travels along a curved trajectory (e.g., bending magnet), it emits synchrotron radiation. If the wavelength of the synchrotron radiation is longer than the bunch length, the resulting radiation is coherent. Therefore, CSR is the low frequency part of the radiation power spectrum, and is more pronounced when the beam bunch is short. Short beam bunch lengths are desirable in many different contexts — free electron lasers (FELs), energy recovering linacs, B-factories — and their importance is only expected to grow in the future. This necessitates a development of a trustworthy code which is capable of properly simulating CSR and its effects, such as an appreciable degradation of the beam’s emittance and its fragmentation. For a recent study of CSR effects and its modeling see [3], where measurements of CSR-induced energy loss and transverse emittance growth have been done for the two bunch compressors of the Linac Coherent Light Source and compared with particle tracking codes. Despite the good agreement found between measurements and simulations, more studies are needed to validate numerical codes, especially in the study of the microbunching instability, where CSR can be emitted also at wavelength shorter than the bunch length if the beam has high-frequency density modulations caused by instability itself. This effect was first observed in numerical simulations with the code ELEGANT and led to a serious concern for its impact on FEL performance [4]. Several theoretical studies advanced our understanding of the microbunching instability [5–7] leading to the development of various suppression schemes [8], while simple numerical methods have been successful in simulating its basic mechanism [9]. Encouraged by these advancements, the FEL community set out to organize a series of workshops [10] to further our theoretical understanding and numerical modeling of the microbunching instability, as well as improve experimental observations and diagnostic techniques. In addition to having detrimental effects in bunch compressors, the microbunching instability is a concern for manipulation schemes such as high harmonic generation for FELs that require accurate modeling of very small-scale structures.

The numerical method adopted in this paper to model CSR effects is based on a Green’s function approach. Its use faces two major difficulties: (i) storing the history of the beam bunch is very intensive for simulations with reasonable spectral resolution; and (ii) the manner in which the computation of the beam’s self-interaction scales. The numerical study of the CSR effects can be made computationally feasible by introducing different approximations/simplifications. The line-charge (1d) approximation [5] of a beam bunch is one such approximation, in

∗Notice: Authored by Jefferson Science Associates, LLC under U.S. DOE Contract No. DE-AC05-06OR23177. The U.S. Government retains a non-exclusive, paid-up, irrevocable, world-wide license to publish or reproduce this manuscript for U.S. Government purposes.
†Email: terzic@jlab.org
‡Email: gbassi@bnl.gov
which the CSR effects are likely overrated by the collapse of the transverse distribution onto a rigid line. Point-to-
point approaches mimic the self-interaction of the beam by direct interaction among N macroparticles which rep-
resent the beam charge density [11, 12]. The drawback here is that the computation of the self-interaction scales as the square of the number of macroparticles, which gets prohibitively expensive rather quickly, thus limiting N to the order of a thousand. This, in turn, yields poor spectral resolution unable to account for small-scale structures present in CSR simulations. The mean-field (mesh, grid, particle-in-cell) approach alleviates this problem by replacing direct interaction of the particles with the inter-
action of the particles with the mean field computed from depositing the particles onto a finite grid. This results in an algorithm which scales better with the number of particles (linear instead of square in the point-to-point approach) and allows for a superior spatial resolution.

In this study we improve the 2d CSR mean-field code developed by Bassi et al. [1, 13]. The code is based on integration of the retarded potential for a 2d charge distribution (no vertical size), considering the shielding effects by infinite parallel plates. In the original implementa-
tion of Bassi et al. [1] Monte Carlo code, the particle distribution sampled by N point-charge particles is first approximated by the cosine expansion, then evaluated on the finite grid and stored for computation of retarded potentials. We improve the code by employing two new alternative, grid-based techniques for representing a 2d particle distribution. Both of them are extremely efficient in comparison to the traditional Monte Carlo cosine expansion — they are about three orders of magnitude faster. Each of the new methods starts with a simple gridded distribution and uses different approaches to denoise it, thus improving the accuracy of the gridded approximation.

The first alternative technique — truncated fast co-
sine transform (TFCT) — uses fast cosine transforms to transform the gridded distribution to Fourier space, where it truncates higher order frequencies by retaining only a fraction of the cosine coefficients. This non-
discriminatory removal of the small scale structure re-
sults in smoothing of the distribution, and represents an overly simplistic method for noise removal. In terms of accuracy, the TFCT approximation is equivalent to the Monte Carlo cosine expansion.

The second alternative technique uses discrete fast wavelet transforms to transform the gridded distribution to wavelet space, where it performs wavelet thresholding, thereby largely removing the numerical noise intrinsic in numerical simulations. Transforming back to the physical space yields a truncated wavelet transform (TWT) approximation to the particle distribution which is appreciably more accurate than the cosine expansion.

In Section II we outline the original algorithm behind Bassi et al.’s 2d CSR code [1, 13]. In Section III we first analyze the origin and profile of numerical noise associated with particle deposition using various particle
deposition schemes (PDS). We then present the two al-
ternative methods in detail. Special attention is devoted to the wavelet-denoised approximation, along with the discussion of how wavelets can be used to remove noise from PIC simulations. In Section IV we use typical particle distributions from Bassi et al. [1] simulation of the microbunching instability to compare the spatial reso-
novation, accuracy, smoothness and efficiency of the two alternative methods to the original Monte Carlo cosine approximation. In Section V we apply two versions of the code — one with the original and the other with the new wavelet-denoised density estimation algorithm — to the simulation of FERMI@Elettra first bunch compressor system, and compare the results. Finally, in Section VI we discuss the significance and use of the new approach and outline its future applications.

II. ORIGINAL ALGORITHM WITH MONTE-CARLO COSINE DENSITY ESTIMATION

We outline the algorithm for density estimation cur-
cently implemented in [1] where the equations of motion are solved in beam frame with phase space coordi-
nates ξ = (z, p_z, x, p_x) and independent variable s. The quantities ρ(z, x, s) = ∫dz dp_z dp_x f(ξ, s) and τ_B(z, x, s) = ∫dz dp_z f(ξ, s) dp_x dp_x must be estimated for the calculation of the self fields. To this end, using the grid transformation (z, x) ↔ (x_1, x_2) ∈ [0, 1] × [0, 1] and denoting ρ_g, τ_g in the grid coordinates (x_1, x_2) we expand ρ_g(x_1, x_2, s) and τ_g(x_1, x_2, s) in a finite Fourier series

\begin{align}
ρ_g(x_1, x_2, s) &= \sum_{i=0}^{I} \sum_{j=0}^{J} \theta_{ij}(s) φ_i(x_1) φ_j(x_2), \\
τ_g(x_1, x_2, s) &= \sum_{i=0}^{I} \sum_{j=0}^{J} Θ_{ij}(s) φ_i(x_1) φ_j(x_2),
\end{align}

where

\begin{align}
θ_{ij}(s) &= \int_A dx_1 dx_2 φ_i(x_1) φ_j(x_2) ρ_g(x_1, x_2, s), \\
Θ_{ij}(s) &= \int_A dx_1 dx_2 φ_i(x_1) φ_j(x_2) τ_g(x_1, x_2, s).
\end{align}

Here \{φ_i\} is the orthonormal basis:

\begin{align}
φ_i(x) = \begin{cases}
1 & i = 0, \\
\sqrt{2} \cos(πx) & i \neq 0,
\end{cases}
\end{align}

for x ∈ [0, 1].

Since ρ_g is a probability density the Fourier co-
ecients θ_{ij} may be written as the expected value E of φ_i(X_1)φ_j(X_2) with respect to ρ_g(\cdot, s)

\begin{align}
θ_{ij}(s) &= E[φ_i(X_1)φ_j(X_2)] \\
&= \int_A dx_1 dx_2 φ_i(x_1) φ_j(x_2) ρ_g(x_1, x_2, s).
\end{align}
where \( X = (X_1, X_2) \) is the random variable with probability density \( \rho_g \). To estimate \( \tau_g \), which is not a probability density, we notice that the Fourier coefficients \( \Theta_{ij} \) may be written as the expected value \( E \) of \( \phi_i(X_1)\phi_j(X_2)P_X \) with respect to \( f_g(\cdot, s) \)

\[
\Theta_{ij}(s) = E\{\phi_i(X_1)\phi_j(X_2)P_X\} = \int_A dx_1 \int_A dx_2 \int_{R^2} dp_x dp_y \phi_i(x_1)\phi_j(x_2)p_x \times f_g(x_1, p_z, x_2, p_x, s),
\]

(7)

where \( X = (X_1, P_Z, X_2, P_X) \) is the random variable with probability density \( f_g(\cdot, s) \).

It follows that the natural estimate of \( E \) is the sample mean, \textit{i.e.}, we have the following two Monte Carlo formulae:

\[
\theta_{ij}(s) \approx \frac{1}{N} \sum_{n=1}^N \phi_i(X_{1n})\phi_j(X_{2n}),
\]

(8)

\[
\Theta_{ij}(s) \approx \frac{1}{N} \sum_{n=1}^N \phi_i(X_{1n})\phi_j(X_{2n})P_{Xn},
\]

(9)

where a realization of the random variable \( X = (X_1, P_Z, X_2, P_X) \) is obtained from beam frame scattered phase space points \( z_i, p_z, x_i, p_x \), at \( s \), \( i = 1, \ldots, N \) (via the transformation: \( (z_i, p_z, x_i, p_x) \rightarrow (x_{1i}, p_{zi}, x_{2i}, p_{xi}) \)). This is a density estimation used in statistical estimation \textit{e.g.}, see [14]).

III. PARTICLE-IN-CELL METHOD: ALTERNATIVE DENSITY ESTIMATION METHODS

Bassi \textit{et al.} [1] code is not a classic PIC code, since the point-charge particles are not directly deposited on a grid, but instead are used to compute a cosine approximation to the distribution which is then evaluated at predefined grid points. This yields a smooth particle distribution, which, in turn, produces a well-behaved integral in the evaluation of the retarded fields, at the price of a costly cosine approximation evaluation.

Here we propose to do away with the cosine approximation and directly deposit particles on the grid, in the spirit of traditional PIC codes. Such direct deposition yields a noisy gridded approximation to the particle distribution that is less smooth than the cosine approximation (see Section IV). The gridded approximation can then be denoised/smoothed out in several different ways:

1. Applying the higher-order PDS;
2. Truncating high frequencies from the Fourier expansion;
3. Thresholding the wavelet expansion.

In the remainder of this section we discuss each of these approaches and argue that only the wavelet thresholding provides a natural setting for \textit{judicious} noise removal, while others systematically remove physical small-scale structure along with small-scale fluctuations due to numerical noise. This is reflected in the accuracy of the wavelet denoised representation being clearly superior to the others.

A. Error in Particle Deposition Schemes

In PIC simulations, the beam’s charge distribution sampled by particles is given by Klimontovich particle distribution

\[
f(z, p_z, x, p_x) = q_0 \sum_{i=1}^N \delta(z-z_i)\delta(p_z-p_{zi})\delta(x-x_i)\delta(p_x-p_{xi}),
\]

(10)

where all \( N \) particles carry the same charge \( q_0 \).

To get to the charge density and current distributions at an arbitrary point in \( (x, z) \) space, the Klimontovich particle distribution must be convolved with a \( k \)-th order PDS \( d_k \) over some neighborhood \( V_k \) around it:

\[
\rho_g(z, x) = \int \int_{V_k} \int_{\mathcal{P}} f(z - \bar{z}, p_z, x - \bar{x}, p_x)dp_z dp_x dk(\bar{z})dk(\bar{x})d\bar{z} d\bar{x},
\]

(11)

The particle density distribution above is evaluated at the specific set of equally spaced points — a rectangular grid. In our 2d case here, the grid is given by the set of grid points \( (z_i \equiv ih_z, x_j \equiv jh_x)_{i=1}^{N_z}, \) with the total number of grid points \( N_{grid} = N_z N_x \). In any realization sampled by \( N \) particles, \( n_j \) is the number of particles inside a \( V_k \)-neighborhood of the \( j \)-th grid point, where \( V_k = \{(-kh_{1/2}, kh_{1/2}), (kh_{2/2}, -kh_{2/2}), ..., (-kh_{D/2}, kh_{D/2})\} \), and \( D \) is the dimensionality of the problem.

PDSs \( d_k \) can be classified according to the number of
grid points to which the total charge is deposited in 1d (also one more than the order of the local polynomial which describes its shape) [15]:

1. \( k = 1 \) is the Nearest Grid Point (NGP) PDS, in which all of particle’s charge is deposited onto the nearest grid point (see Figure 1):

\[
d_k(x) = h^{-1} \left\{ \begin{array}{ll} 1 & \text{if } |x| \leq \frac{h}{2} \\ 0 & \text{otherwise}, \end{array} \right. \tag{12}
\]

where \( x \) is a general coordinate and \( h \) is the spacing in that coordinate;

2. \( k = 2 \) is the Cloud-In-Cell (CIC) PDS, in which each particle’s charge is deposited to all vertices of the cell it occupies (2 vertices in 1d, 4 in 2d, 8 in 3d), with the weight inversely proportional to its distance from the vertex (see Figure 1):

\[
d_k(x) = h^{-1} \left\{ \begin{array}{ll} 1 - \frac{|x|}{h} & \text{if } |x| \leq h \\ 0 & \text{otherwise}. \end{array} \right. \tag{13}
\]

3. \( k = 3 \) is the Triangular Shaped Cloud (TSC) PDS, which deposits the charge onto three nearest cells (3 vertices in 1d, 9 in 2d, 27 in 3d) using a quadratic (see Figure 1):

\[
d_k(x) = h^{-1} \left\{ \begin{array}{ll} \frac{1}{3} - \frac{2}{3} \left( \frac{|x|}{h} \right)^2 & \text{if } |x| \leq \frac{h}{2} \\ \frac{1}{3} \left[ \frac{3}{2} - \left( \frac{|x|}{h} \right)^2 \right] & \text{if } \frac{h}{2} \leq |x| \leq \frac{3h}{2} \\ 0 & \text{otherwise}. \end{array} \right. \tag{14}
\]

Other higher-order PDS are obtained by convolving the previous PDS with a CIC PDS [15]. They are, however, rarely used because of their computational cost: the next higher order PDS than TSC increases the number of points to which the charge is deposited from \( 3^D \) to \( 4^D \), which in 3d is from 27 to 64 — an increase of more than a factor of two. Furthermore, the smoothing of the particle distribution obtained by using higher-order PDS comes at the expense of spectral resolution — if the domain of the PDS is too wide (the order of the PDS too high), small scale structures of the order of the size of the domain or smaller will be smoothed out. This is why in practice most PIC simulations use either CIC or TSC PDS. In the present study, we also, for the same reasons, deal only with \( k = 1, 2, 3 \) PDS.

In analyzing the origin of the numerical noise associated with PDS, we follow and expand on the discussion of [16]. For the NGP PDS \((k = 1)\), the probability of a particle being deposited into the \( j \)th grid point is given by the binomial distribution

\[
P(n_j = n|p_j) = \frac{N_1}{n!(N-n)!} p_j^n (1-p_j)^{N-n}, \tag{15}
\]

where \( p_j \equiv \bar{n}_j/N \) and \( \bar{n}_j \) is the expected number of particles in the \( j \)th grid point. In \( N \)-body simulations, \( N \) is quite large, in which case the binomial distribution converges to the Poisson distribution with mean \( \bar{n}_j \):

\[
P(n_j = n|p_j) = \frac{\bar{n}_j^n e^{-\bar{n}_j}}{n!}, \tag{16}
\]

where \( n \) is an integer.

One of the ways to quantify noise in a PIC simulation is to define a “global” measure of the error associated with noise induced by sampling and particle deposition, defined as the algebraic average of variances:

\[
\sigma^2 = \frac{1}{N_{\text{grid}}} \sum_{i=1}^{N_{\text{grid}}} (q_i - \bar{q}_i)^2 = \frac{1}{N_{\text{grid}}} \sum_{i=1}^{N_{\text{grid}}} \text{Var}(q_i), \tag{17}
\]

where \( q_i \) is the charge in the \( V \)-neighborhood of the \( i \)th gridpoint, and \( \bar{q}_i \) the exact value of the particle distribution at the location of the \( i \)th gridpoint.

For the PDS considered in this paper \((k = 1, 2, 3...)\), their values are (see Appendix A):

\[
\sigma_k^2 \approx \frac{a_k^2 N_0}{N_{\text{grid}}} = \frac{a_k^2 N q_0}{N_{\text{grid}}} \tag{18}
\]

where

\[
a_1 = 1, \quad a_2 = \left( \frac{2}{3} \right)^{D/2}, \quad a_3 = \left( \frac{11}{20} \right)^{D/2} \tag{19}
\]

and \( Q_{\text{tot}} = N q_0 \) is the total charge of the distribution. Therefore, the noise distribution for the higher order PDS \((k = 2, 3...)\) is a contracted Poissonian, given by eq. (16) with \( n_j \to a_k n_j \). This also means that the signal obtained by depositing the particle distribution onto a grid using a higher order PDS is \( a_k^{-1} \) more accurate than the NGP \((k = 1)\). Table I shows the values of \( a_k \) for dimensions \( D = 1, 2, 3 \).
TABLE I: Values of $a_k$ for different order PDS.

| $D$ | $k = 1$ | $k = 2$ | $k = 3$ |
|-----|---------|---------|---------|
| 1   | 0.82   | 0.74    | 0.55    |

FIG. 2: Accuracy (measured by the IMSE given in eq. (20)), scaled to $a_k^2$, of the deposition scheme, versus the number of particles $N$ for the three lowest order PDS: NGP (red line), CIC (green) and TSC (blue) for the 2d sinusoidally modulated flat-top distribution used in Section IV.

The global variance as defined above is closely related to the integrated mean square error (IMSE), or $L_2$-error, on the grid:

$$E = h_x h_z \sum_{i=1}^{N_{\text{grid}}} (q_i - \bar{q}_i)^2 = \frac{L_x L_z}{N_{\text{grid}}} \sum_{i=1}^{N_{\text{grid}}} (q_i - \bar{q}_i)^2$$

$$= L_x L_z \sigma^2,$$  

where $h_x$ and $h_z$ are the resolution and $L_x$ and $L_z$ the size of the grid in $z$- and $x$-directions, respectively. The IMSE is often used to quantify the accuracy of the approximations of the analytically known particle distribution.

Eqs. (18) and (20) imply that the quantities $\sigma_k^2/a_k^2$ and $E/a_k^2$ are the same for all PDSs, and that, for a fixed resolution ($N_{\text{grid}} = \text{const}$), they both scale as $\propto N^{-1}$. This is clearly demonstrated in Figure 2. Therefore, depositing point-charge particles over several grid points — i.e., using higher-order PDS — essentially cuts off the highest order frequencies of the resulting distribution. Such smoothened distributions are closer to the smooth distribution function being sampled, resulting in a lower IMSE, at the expense of reduced spatial resolution.

The algebraic average of variances of noise in a PIC simulation $\sigma_k^2$ depends sensitively on the parameters of the simulation (number of particles and the grid resolution) and on the PDS (through $a_k$), but only very weakly on the particle distribution (see Appendix A of [16]). After a systematic study of the many types of distributions of particles arising in beam dynamics, this weak dependence appears to be negligible (see [16] for a more detailed discussion). This global measure of numerical noise in PIC simulations is needed to compute the noise threshold which is used in wavelet thresholding (see Section III C).

B. Alternative Method 1: Truncated Fast Cosine Transform

A finite cosine expansion of a charge and current density distribution is given by

$$\rho(z, x) = \sum_{n=0}^{M_x} \sum_{m=0}^{M_z} \theta_{nm} \phi_n(z) \phi_m(x),$$  

$$\tau(z, x) = \sum_{n=0}^{M_x} \sum_{m=0}^{M_z} \Theta_{nm} \phi_n(z) \phi_m(x),$$  

where

$$\theta_{nm} = \int_0^1 \int_0^1 \rho(z, x) \phi_n(z) \phi_m(x) dz dx,$$  

$$\Theta_{nm} = \int_0^1 \int_0^1 \tau(z, x) \phi_n(z) \phi_m(x) dz dx,$$

and the basis functions $\phi$ given in eq. (5). Both the Monte Carlo and the TFCT evaluate the coefficients from eq. (22) using the charge density distribution function $\rho_q(z, x)$ from eq. (11) with different PDSs:

- The Monte Carlo cosine expansion uses a “zeroth” order PDS, the Dirac delta function, which “deposits” the entire charge of the particle onto a point at which it is located:

$$d_0(x) = \delta(x).$$  

Upon substituting eq. (23) in eq. (11), the resulting charge and current density distributions become

$$\rho(z, x) = q_0 \sum_{i=1}^{N} \delta(z - z_i) \delta(x - x_i),$$  

$$\tau(z, x) = q_0 \sum_{i=1}^{N} p_{xi} \delta(z - z_i) \delta(x - x_i).$$

When inserted in the expression for the coefficients

[1] This is a PDS only in a loose sense. It is meaningless to talk about properties of this PDS as we did for the higher order ones $(\sigma_0, a_0, E)$, because they are not defined.
(22), this charge density distribution yields
\[
\theta_{nm} = q_0 \sum_{i=1}^{N} \phi_n(z_i) \phi_m(x_i),
\]
\[
\Theta_{nm} = q_0 \sum_{i=1}^{N} p_{xi} \phi_n(z_i) \phi_m(x_i).
\]

- The TFCT uses higher (non-zero) order PDSs \(d_k\) (§III A) evaluated on a grid of resolution \((N_z, N_x)\). The coefficients are then computed from this gridded distribution via fast cosine transform, which is just an optimized way to solve eq. (22). Finally, the full cosine expansion containing \((N_z, N_x)\) coefficients is truncated to retain only the lowest \((M_z, M_x)\).

When the resolution of the grid is fine, the \(V_k\)-neighborhood is small and the PDSs are narrow. The narrowest PDS of non-zero order is NGP, which, as the resolution grows, becomes increasingly a better approximation to the delta function, and hence closer to the zeroth order PDS. Therefore, we expect the cosine coefficients computed from a grid resulting from a NGP PDS to be very close to those of Monte Carlo cosine, given in eq. (25). In Appendix B, we prove that the two are the same to within a second order approximation. Figure 3 shows this: the IMSE as a function of the number of particles for the two expansions overlap.

The full (non-truncated) fast cosine expansion generated solving eq. (25) with \(M_z = N_z\) and \(M_x = N_x\) is as accurate as the approximation generated using NGP PDSs, because neither of the two lowest PDSs (zeroth order delta function or the first order NGP) smoothen the contributions of individual point-charges over space: in both cases the charges are deposited onto a single point, unlike in the case of the higher order PDSs. As discussed earlier, smoothing by a higher-order PDS decreases the IMSE (Figure 3) by removing the highest order frequencies (thereby reducing its spectral resolution). However, more powerful smoothing is accomplished by truncation of the cosine representation (see Section IV B).

We introduce the alternative cosine expansion — TFCT — which is equivalent to the one currently used in Bassi et al. [1] code, yet considerably faster. The algorithm can be outlined as follows:

1. Deposit particles on the \((N_z, N_x)\) grid.
2. Apply 2d fast cosine transform on the grid, thus yielding \((N_z, N_x)\) cosine coefficients.
3. Remove the high-frequency contribution to the density by truncating coefficients higher than \(M_z\) and \(M_x\) in \(z\)- and \(x\)-coordinate, respectively. This removal of the high-frequency components results in a smoother distribution, but it also removes small scale structures that may not be due to noise. Therefore, the truncation of the Fourier coefficients also restricts the spatial resolution of the representation (see Section IV B).
4. Apply 2d inverse fast cosine transform on the grid, to obtain the smoothened distribution in physical space.

C. Alternative Method 2: Thresholded Wavelet Transform

The details of the wavelet thresholding in the context of the PIC simulations of beams have been outlined in [16]. Here we highlight the main points of their discussion.

Since their inception in the 1980’s [17–21], wavelets and wavelet transforms have been used to denoise signals contaminated with noise. Just like the discrete Fourier transform (DFT), the discrete wavelet transform (DWT) is a fast, linear operation on data sets with size of integer power of two in each dimension, yielding a transformed data set of the same size. The most important difference between DWT and DFT is that the DWT provides localization in both frequency space and configuration space, whereas the DFT is only localized in frequency space. The reason for this is that, while still providing a hierarchy of frequencies, the basis functions of the DWT have compact support (finite range over which they are non-zero); in contrast, the basis functions of the DFT have infinite range). This dual localization renders a number of operators and functions sparse in the wavelet space. Essential background on wavelets and wavelet transforms is available in literature [17, 20, 21].

Upon transformation of the noisy data to wavelet space, the signal is generally given by a few large co-
The transformed signal is a judiciously adopting a noise threshold $T$, one of the two common thresholding techniques can be applied:

1. **Hard thresholding** sets the coefficients with magnitudes below the noise threshold $T > 0$ to zero:

   $$\bar{w}_i = \begin{cases} w_i & \text{if } |w_i| > T \\ 0 & \text{if } |w_i| \leq T \end{cases} \quad (26)$$

2. **Soft thresholding** sets the coefficients with magnitudes below the noise threshold $T > 0$ to zero, and contracts coefficients above it by $T$:

   $$\bar{w}_i = \begin{cases} \text{sign}(w_i)(|w_i| - T) & \text{if } |w_i| > T \\ 0 & \text{if } |w_i| \leq T \end{cases} \quad (27)$$

Thresholding the wavelet coefficients therefore removes the smallest-scale fluctuations usually associated with numerical noise. However, the noise threshold must be chosen judiciously so as to avoid uncontrolled “smoothing out” over the physical fine-scale details that serve as a seed for the onset of global features.

The discussion on choosing the noise threshold is extensively discussed in literature [16, 22–28]. The most commonly used noise threshold is given in terms of the standard deviation $\sigma$ of the noise as

$$T = \sqrt{2\log N_{\text{grid}}} \sigma. \quad (28)$$

This is a universal threshold for signals with Gaussian white noise, i.e., better noise removal cannot be achieved for all signals in all wavelet bases. It results in a denoised signal which is within a small factor of ideal denoising [24, 25].

Most, if not all, of the work on wavelet denoising deals with distributions (signals) polluted with additive (distribution-independent) Gaussian noise [24, 25, 27]. In Section III A, however, we demonstrate that the noise in PIC simulations is Poisson-distributed (multiplicative) and weakly distribution-dependent. We assure that the wavelet thresholding theory outlined in earlier work directly applies to PIC simulations by transforming a signal with Poisson-distributed noise into one for which the noise is approximately Gaussian-distributed via a variance-stabilizing transformation due to Anscombe [29]:

$$X_G = 2\sqrt{X_P + \frac{3}{8}}. \quad (29)$$

The transformed signal $X_G$ has unit variance ($\sigma = 1$) and mean

$$m_G = \sqrt{m_P + \frac{3}{8}} - \frac{1}{8m_P^{1/2}} + \frac{1}{64m_P^{3/2}}. \quad (30)$$

where $m_P$ is the mean of the Poissonian signal. Applying the Anscombe transformation leads to a bias in the data [28, 29], which is removed by ensuring that the denoised and noisy data have the same mean (in PIC simulations, this is equivalent to enforcing conservation of total charge).

Application of the Anscombe transformation to a signal contaminated with Poisson noise as in eq. (16) yields a signal which is approximately normally distributed, with variance $\sigma_{\text{NGP}} = 1$ for the NGP PDS. For the higher-order PDSs, the data distribution is a contracted Poissonian, given by eq. (16) with $n_j \to a_k n_j$, which implies that the resulting variance will be appropriately contracted by a factor $a_k$, $(\sigma_{\text{CIC}} = a_2, \sigma_{\text{TSC}} = a_3)$. Combining these with eq. (28) yields optimal noise threshold for the Anscombe-transformed (variance-stabilized) data obtained by depositing a distribution using a $k^{\text{th}}$ order PDS:

$$T_k = \sqrt{2\log N_{\text{grid}}} a_k. \quad (31)$$

The analytical distribution used is the sinusoidally modulated flat-top distribution described in Section IV. Figure 4 shows the IMSE as a function of the threshold $T$. The limits are $T \to 0$ — no thresholding is done, the signal is unaltered, yielding the original gridded distribution; and $T \to \infty$ — all wavelet coefficients are set to zero, and there is no remaining signal. We notice that the threshold $T_2$ is consistently at or near the maximum denoising factor. This has also been observed in our earlier study with different kind of distributions [16].

1. **Choosing a Wavelet Family**

Wisely choosing a wavelet family should yield a compact representation of the signal. Donoho and Johnstone [24, 25] showed that for a given application there exists the “ideal” wavelet basis in which the optimal compression is achieved. This means that the closer the basis is to the “ideal” basis, the better the compression that results, and vice versa.

Figure 5 shows the fraction of wavelet coefficients of the charge density retained after thresholding versus the number of particles sampling the particle distribution (at the fixed resolution). In the present study, we use two types of wavelets: *orthogonal* — represented by Daubechies family of wavelets of order 2, 3, 6 and 10, symlets of order 4, 6, 8 and coiflets of order 1, 2, 3; and *biorthogonal* — represented by bior (4,4) and bior (6,8) [19].

As the order of the wavelet family increases, so does the width of its compact support. Better localization in space is obtained through wavelets with small compact support (lower order), while better localization in frequency is achieved by higher-order ones. In this study, the more important of the two is the localization in frequency, which is why we use Daubechies wavelets of order
FIG. 4: Accuracy of the approximation (measured by the IMSE) as a function of the wavelet threshold $T$ for the sinusoidally modulated flat-top distribution used in Section IV. The resolution of the grid is fixed at $1024 \times 128$ in $(z,x)$ space. Number of particles sampling the distribution is given in the title of each panel. The four curves correspond to the four different wavelets from Daubechies family: order 2 (red), order 3 (green), order 6 (blue), order 10 (purple). Light blue horizontal line represents the accuracy of the grid deposition and black the cosine expansion. Vertical line is the analytical value of $T_2$ given in eq. (31), because CIC PDS ($k=2$) is used.

FIG. 5: Sparsity of the wavelet-denoised representation, quantified by the fraction of non-zero wavelet coefficients after thresholding. Wavelet families used are: Daubechies of order 2, 3, 6, 10; symlets of order 4, 6, 8; coiflets of order 1, 2, 3; and biorthonormal (4,4) and (6,8).

10. From Figure 4, it seems to be the best-performing wavelet in its family.

2. Combining Wavelet Denoising With Different Particle Deposition Schemes

The TWT method is grid based, so it is natural to ask which of the PDS investigated in Section III A, when combined with wavelet thresholding will yield the most accurate approximation.

Figure 6 shows the IMSE for the three PDS — NGP ($k=1$), CIC ($k=2$) and TSC ($k=3$) — before (thin lines) and after wavelet thresholding (thick lines). Wavelet-denoised CIC and TSC representations are clearly more accurate than the wavelet-denoised NGP. For small and intermediate $N$, the CIC and TSC PDS are quite similar in accuracy. For large $N$, however, the lower-order PDS CIC yields a better approximation than the TSC, because of the added smoothing used in the higher PDS. This sends a clear message that,
if wavelet-denoising is employed, there is no need to use PDS higher than CIC \((k = 2)\). That is why in the remainder of the present study we only use CIC PDS.

3. Outline of the Thresholded Wavelet Method

The wavelet-denoised algorithm for estimating the particle density can be outlined as follows:

1. Deposit particles on the \((N_{z}, N_{x})\) grid.

2. Apply Anscombe transformation to convert the resulting gridded signal polluted by the Poissonian noise (the gridded particle distribution) to the signal with Gaussian noise.

3. Apply 2d DWT on the grid, thus yielding \((N_{z}, N_{x})\) wavelet coefficients.

4. Perform wavelet thresholding on the wavelet coefficients in order to remove numerical noise and smoothen the particle distribution. Unlike the filtering of the cosine coefficients in Fourier space, this judicious noise removal does not restrict the spatial resolution of the distribution: small-scale structures which are not deemed to be noise during wavelet thresholding (i.e., the amplitude of the corresponding wavelet coefficients is larger than the threshold) are retained in the denoised distribution.

5. Apply 2d inverse DWT on the grid, to obtain the smoothed distribution in physical space.

6. Apply inverse Anscombe transformation.

IV. COMPARISON OF THE DENSITY ESTIMATION METHODS

We now compare the TFCT and the TWT estimation methods in the following areas: spatial resolution, accuracy, smoothness and efficiency. The simulations in this section are based on an analytical model charge distribution, carefully chosen to represent small-scale structure in the beam. Knowing the underlying analytical distribution allows computation of the IMSE for the two methods. It is our expectation that the comparison presented here, which is based on the analytically known distribution, extends to general particle distributions arising in realistic simulations.

A. Analytical Model Charge Distribution: Sinusoidally Modulated Flat-Top

In order to study the microbunching instability, Bassi et al. [1] designed a simulation in which a flat-top distribution, modulated with a single sinusoidal frequency is passed through FERMI@Elettra first bunch compressor system. The resulting amplification of the initial modulation, denoted as the gain factor (see Section V), is a good indicator as to how small-scale structure may respond to bunch compression. It is therefore important to show how the gain factor varies as a function of the frequency (wavelength) of the modulations.

In what follows, we investigate only the approximations to the particle charge distribution. However, the analysis also applies to approximations to the current density, because the two have the same form (see eq. (21)).

Figure 7 shows the flat top charge distribution in physi-
\[ \rho(x, z) = \left[ 1 + A \cos \left( \frac{2\pi z}{\lambda} \right) \right] \mu(z) g(x), \] (32)

where
\[ \mu(z) = \frac{1}{4a} \left[ \tanh \left( \frac{z + a}{b} \right) - \tanh \left( \frac{z - a}{b} \right) \right], \]
\[ g(x) = \frac{1}{\sqrt{2\pi} \sigma x_0} e^{-\frac{x^2}{2\sigma^2}}. \] (33)

Parameters \( a \) and \( b \) determine the width and the steepness of the flat-top in \( z \)-coordinate; \( \sigma x_0 \) the width in \( x \)-coordinate, and \( A \) and \( \lambda \) are the magnitude and the wavelength of the modulation, respectively. The parameters and variables are non-dimensionalized in the manner given in Table II. The charge distribution above is a particular case of the full phase-space distribution function used in the code, given by eq. (48).

### B. Spatial Resolution

In general, the smallest structure representable by a finite cosine approximation (including the TFCT method) is determined by the highest order basis function retained in the expansion. The highest order basis function in \( z \)-coordinate is \( M_z \) and \( M_x \) in \( x \). The arguments of the basis functions (given in eq. (5)) are
\[ M_z \pi \bar{z} = \frac{M_z \pi z}{L_z} + \phi_z, \quad \text{in } z \text{-coordinate,} \] (34)
\[ M_x \pi \bar{x} = \frac{M_x \pi x}{L_x} + \phi_x, \quad \text{in } x \text{-coordinate,} \]
where \( L_z \) and \( L_x \) are the size of the computational box around the beam in \( z \)-and \( x \) coordinates, and \( \phi_z \) and \( \phi_x \) are dimensionless constants (and therefore just simple phase shifts). Therefore, the smallest wavelengths representable by this expansion are:
\[ \lambda_z^{\min, \cos} = \frac{2L_z}{M_z}, \quad \text{in } z \text{-coordinate}, \] (35)
\[ \lambda_x^{\min, \cos} = \frac{2L_x}{M_x}, \quad \text{in } x \text{-coordinate.} \]

In the microbunching simulations, we use a sinusoidally-modulated flat-top distribution to study the evolution of the small-scale structures. The wavelength of these small-scale longitudinal structures is given by \( \lambda \) in eq. (32). The cosine approximation will be able to approximate accurately only the particle distributions with small-scale structure with wavelengths \( \lambda \) larger than the smallest wavelength representable by the cosine expansion \( \lambda_z^{\min, \cos} \). This imposes a limit on usability of the cosine expansion:
\[ \lambda > \lambda_z^{\min, \cos} = \frac{2L_z}{M_z}. \] (36)

Figure 8 shows the accuracy in approximating the gain factor of a distribution sampled with \( N = 10^8 \) particles by a cosine approximation with \( M_z = 100 \) and \( M_x = 40 \) and the finite grid (both with and without wavelet denoising) of resolution \( N_z = 1024 \) and \( N_x = 128 \). The relation (37) predicts that the precipitous drop in accuracy (increased in error) for the cosine approximation with \( M_z = 100 \) should occur for
\[ \lambda < \frac{7200 \mu m}{M_z} = 72 \mu m, \] (38)
which is exactly what is observed. This general constraint is manifested in Figure 9, where the IMSE is plotted as a function of the longitudinal modulation wavelength \( \lambda \) and number of cosine basis functions in the longitudinal direction (\( M_z \)). The steep drop-off in accuracy of cosine approximation, as outlined by the contour plot, coincides with the line \( \lambda M_z \approx 7200 \mu m \), given by eq. (38). It is also interesting to note that for the range where cosine expansion provides a good approximation to the particle distribution function (\( \lambda M_z > 7200 \mu m \)), increasing \( M_z \) does not yield a more accurate approximation, as evidenced by an upward sloping surface of the error \( E \) as a function of \( M_z \). All of this essentially means that the optimal simulation with cosine expansion — one that would provide the best accuracy at the low-
The number of transverse basis functions is fixed at 
\( N \) and number of cosine basis functions in the lon-
ditudinal direction (\( N_z \)), for \( N = 10^8 \), \( N_z = 1024 \), \( N_x = 128 \). The number of transverse basis functions is fixed at \( M_x = 40 \). The contour curve denoting the drop-off in accuracy of about two orders of magnitude corresponds to \( \lambda M_z \approx 7200 \mu m \), as predicted by eq. (35).

\( \lambda_{\text{min, grid}} = \frac{4L_z}{N_z} \). \hspace{1cm} (39)

This means that the requirement for the accurate represen-
tation of the distribution on the finite grid is then given by

\( \lambda > \lambda_{\text{min, grid}}^z = \frac{4L_z}{N_z} \).

\( N_z > \frac{4L_z}{\lambda} = \frac{3600 \mu m}{\lambda} \). \hspace{1cm} (40)

For the simulations in Figure 8, where \( N_z = 1024 \), the

\( \lambda < \frac{3600 \mu m}{N_z} \approx 14 \mu m \). \hspace{1cm} (41)

relation (40) predicts that for

the grid approximation will be inaccurate, as evidenced by the increase in error of the grid approximation.

Therefore, the spatial resolution for the cosine expansion is determined by the number of expansion coeffi-
cients \( M_z \) and \( M_x \), while for the grid approximation it is given by the number of grid points \( N_z \) and \( N_x \). In the current implementation of Bassi et al. [1] code, the longitudinal resolution is fixed so that the number of grid points over the period of the highest retained basis func-
tion is fixed to 20. After using the eq. (39), we obtain

\( \lambda_{\text{min, cos}}^z = 20h_z = 5\lambda_{\text{min, grid}}^z \). \hspace{1cm} (42)

and, with the help of the eq. (35)

\( N_z = 10M_z \). \hspace{1cm} (43)
In other words, the smallest structure representable by the truncated cosine expansion (Monte Carlo or TFCT on the grid) is five times larger than the smallest structure representable on the grid.

Thresholding of wavelet coefficients does not automatically remove highest frequency components in the manner of the TFCT method. Instead, only the coefficients whose magnitude is below a certain predetermined threshold are set to zero. The process of wavelet thresholding, including the Anscombe transform, removes only the coefficients adjudged to be noise, regardless of their spatial resolution. As a consequence of this, small-scale structures which are not classified as noise remain a part of the denoised signal. This means that the wavelet thresholding on a grid does not further limit the spatial resolution. Therefore, the spatial resolution of the TWT is the same as that of the grid (see Figure 8).

### C. Accuracy

Figure 10 shows the IMSE for the grid, Monte Carlo cosine, TFCT and TWT approximations as a function of the number of particles $N$ used in the simulation. It is evident that the IMSE for all approximations scales as $N^{-1}$, consistent with the well-known result that the signal quality (as measured by the square root of the normalized inverse of the IMSE, also known as the signal-to-noise ratio) scales as the square root of the number of particles. As we have shown in Appendix B, the Monte Carlo cosine expansion and TFCT are equivalent to within second order in the grid spacing. It is also clear that the wavelet denoising significantly improves the accuracy of the approximation, surpassing that of the cosine expansion. The increase in accuracy when using the denoised grid approximation is even more pronounced for other values of the modulation wavelength $\lambda$ (see Figure 8).

As we mentioned earlier, we require that the simulations are set up in such a way that the highest-frequency basis function retained is sampled by at least 15-20 gridpoints, in order to ensure smooth finite-differencing required for the numerical computation of derivatives. Also, in order to maintain the same signal quality, as represented by the distribution of point-charges sampling the underlying distribution deposited on a finite grid, we maintain the same number of particles per grid for the entire simulation. The parameters for the simulation are given in Table III. Figure 11 shows the accuracy of the grid, TFCT and TWT representations (top), as well as the number of coefficients retained in the ideal representation, normalized to the number of gridpoints (bottom). It is evident that both truncated representations are significantly more accurate than the gridded representation alone, and that the TWT is appreciably more accurate than the TFCT. Furthermore, TWT is about an order of magnitude more sparse, which reduces memory requirements and leads to a more efficient algorithm.

### D. Smoothness

In order to estimate the smoothness of the particle distribution, we compute the IMSE of its derivatives obtained using a finite difference scheme on the grid. Figure 12 compares the smoothness of the three gridded approximations — (non-smoothened) grid, TFCT and TWT — measured by the IMSE of the $z$-derivatives computed using a third-order-accurate four-point asymmetric stencil (this is how the fields are computed from the grid-ded particle distribution in the code of Bassi et al. [1]). Both smoothening schemes — TFCT and TWT — improve the accuracy of the first derivative by more than three orders of magnitude.
the execution times of grid methods become quite similar.

This is also why the execution times of grid methods become quite similar for large \( N \) (Figure 14).

Figure 14 shows the execution time for the two alternative methods normalized with the execution time of the Monte Carlo cosine approximation used in the original code as a function of the number of particles \( N \). It is evident that the ratio of execution times asymptotically approaches \( 1/(M_x M_z) \), because Monte Carlo cosine is \( \propto O(M_x M_z N) \) and the grid-based methods (for large \( N \) used in realistic simulations) as \( \propto O(N) \).

An honest comparison of the efficiency of Monte Carlo cosine expansion and the expansions with TFCT and TWT has to take into account the optimum number of basis functions in the cosine expansion. The execution times for the computation of cosine coefficients and grid-based methods is given by:

\[
t_{\text{MC cos}} = k_1 M_x M_z N, \\
t_{\text{grid}} = k_2 N,  \
\]

where \( k_1 \) and \( k_2 \) are some constants which depend on the computer executing the simulation. Here we assume that for large \( N \), as required by realistic simulations, the dominant cost is due to grid deposition, and not fast transforms (see Figure 13). Therefore, the ratio of the execution times is simply

\[
r \equiv \frac{t_{\text{MC cos}}}{t_{\text{grid}}} = k M_x M_z,  \
\]

where the constant \( k \), and therefore the ratio \( r \) should be computer-independent (as it can be seen from Figure 14,

### Table III: Values for the parameters used in simulations in Figures 11, 12 and 23.

| \( \lambda \) | \( N_x \) | \( N_z \) | \( M_x \) | \( M_z \) | \( N \) |
|---|---|---|---|---|---|
| 25 - 50 | 4096 | 128 | 400 | 40 | \( 2 \times 10^9 \) |
| 60 - 90 | 2048 | 128 | 200 | 40 | \( 10^8 \) |
| 100 - 200 | 1024 | 128 | 100 | 40 | \( 5 \times 10^7 \) |

A detailed analysis of the accuracy of derivatives computed using a finite difference scheme is given in Appendix C.

### E. Efficiency

Figure 13 shows how execution times of different methods and their constituent parts scale with the number of particles \( N \). The Monte Carlo-based computation of cosine coefficients requires integration over all of the particles in the distribution, therefore scaling as \( \propto O(M_x M_z N) \). Using cosine coefficients to approximate the particle distribution on the grid requires summing over all the cosine coefficients, thus scaling as \( \propto O(M_x M_z N_{\text{grid}}) \). In realistic simulations \( N \gg N_{\text{grid}} \), so the computation of cosine coefficients will be by far more time-consuming of the two.

Both grid-based methods use a particle deposition, which scales as \( \propto O(N) \), and fast transforms: FCT \( \propto O(N_{\text{grid}} \log(N_{\text{grid}})) \) and DWT \( \propto O(K N_{\text{grid}}) \), where \( K \) is the width of the wavelet family. This means that for large numbers of particles \( N \) used in realistic simulations, the most time-consuming part of the approximation is particle deposition (Figure 13). This is also why the execution times of grid methods become quite similar.
it is on the order of unity). For the near-optimal number of basis function in the cosine expansion along the longitudinal direction $\bar{M}_z = 7200 \mu m/\lambda$, the ratio is:

$$\bar{r} = k \bar{M}_z M_x = k M_x \bar{M}_z \frac{M_z}{\lambda M_z} = r \frac{7200 \mu m}{\lambda M_z}. \quad (46)$$

From our earlier analysis and Figure 14, we get that for large $N$ and $M_z = 100$, $r \approx M_x M_z$ (ignoring the contributions from evaluation of cosine on the grid and wavelet denoising, which are negligible for large $N$, as suggested by the top two panels of the figure), so

$$\bar{r} \approx \frac{7200 \mu m}{\lambda} M_x, \quad (47)$$

which means that the cost-effectiveness of the both TFCT and TWT over the Monte Carlo cosine expansion will be even more pronounced for simulations of small-scale perturbations, as expected. For example, this means that for $M_x = 40$, the increase in computational efficiency for $\lambda < 100 \mu m$ should be by at least a factor of about 2880, and double that at $\lambda < 50 \mu m$.

**V. SIMULATION COMPARISON: TFCT VS. TWT**

As an application, we study the microbunching instability in the first bunch compressor system of the FERMI@Elettra FEL. We discuss the numerical results and compare the performance of the TWT and TFCT methods.

In order to reach the desired electron peak current capable of inducing the collective FEL instability in FELs, the pulse length of a low-emittance electron bunch generated from the photocathode rf gun is magnetically compressed in the linear accelerator by more than an order of magnitude. Numerical and theoretical investigations of high-brightness electron bunch compression lead to a microbunching instability driven by CSR that can significantly degrade the beam quality [8].

The mechanism for microbunching instability can be understood as follows. A high-brightness electron beam with a small amount of longitudinal density modulation can create self-fields that lead to beam energy modulation. Since a magnetic bunch compressor (usually a chicane) introduces path length dependence on energy, the induced energy modulation is then converted to additional density modulation that can be much larger than the initial density modulation. This amplification process (the gain in microbunching) is accompanied by a growth of energy modulation and a possible growth of emittance if significant energy modulation is induced in a dispersive region such as the chicane. Thus, the instability can be harmful to FEL performance, which depends critically on the high quality of the electron beam [8].

We study the microbunching instability with parameters for the first bunch compressor system of FERMI@Elettra. The layout of the system is shown in Figure 15 and the chicane and beam parameters at first dipole are shown in Table IV. The phase space density on entrance to the chicane is a smooth function $a_0(z, p_z, x, p_x)$ modulated by a factor $1 + A \cos(2\pi z/\lambda)$ where $A$ is a small amplitude and $\lambda$ is the perturbation wavelength

$$f(z, p_z, x, p_x, 0) = (1 + \varepsilon(z)) a_0(z, p_z, x, p_x), \quad (48)$$
where

$$a_0(z, p_z, x, p_x) = \mu(z) \rho_c(p_z - h z) \times \exp[-(x^2 + (\alpha_0 x + \beta_0 p_x)^2)/2\epsilon_0 \beta_0]/2\pi \epsilon_0, \quad (49)$$

$$\rho_c(p_z) = \exp[-p_z^2/2\sigma_u^2]/\sqrt{2\pi \sigma_u}, \quad (50)$$

$$\mu(z) = [\tanh((z + a)/b) - \tanh((z - a)/b)]/4a, \quad (51)$$

$$\varepsilon(z) = A \cos(2\pi z/\lambda) = A \cos(k_0 z). \quad (52)$$

The function $a_0$ contains the $z - p_z$ correlation that is necessary for bunch compression, called energy chirp.

A standard approach to study the microbunching instability consists in calculating a gain factor for a given initial modulation wavelength $k_0$. This gain factor is defined as

$$G(k_0) = |\tilde{\rho}(k_0 C(s_f), s_f)/\tilde{\rho}(k_0, 0)|, \quad (53)$$

where

$$\tilde{\rho}(k, s) = \int dz \exp(-ikz) \rho_L(z, s), \quad (54)$$

for a given initial modulation wavelength of $\lambda = 2\pi k_0$.

Here $\rho_L(z, s) = \int dx \rho(p, z, x, s)$ is the longitudinal spatial density, $C(s) = 1/[1 + hR_{56}(s)]$ is the compression factor of the chicane, $s$ is the arclength along the reference orbit, $s_f$ is the value of $s$ at the exit of the chicane, and $h$ is the linear chirp parameter. This definition is motivated by the fact that when the system is linearized the initial modulation at wave number $k_0$ will appear at wave number $k_0 C(s_f)$ at the end of the chicane due to the bunch compression. We calculate the gain factor numerically and compare it with the analytical formula derived in [6].

The derivation of the analytical gain factor is based on a Vlasov approach with a 1d steady state CSR model and with a coasting beam form of $a_0$ in eq. (49). The gain factor is obtained by linearizing the Vlasov equation and solving it with an iterative procedure. The numerical gain factor is calculated with the 2d model of CSR [1] and with a 1d field approximation to the 2d model [30]. Due to the several assumptions made in the derivation of the analytical gain factor, we do not expect a detailed agreement between analytical and numerical results. In what follows, we sketch out the 2d model and 1d field and discuss the numerical results.

### A. Self-Consistent 2d model

We briefly outline the two-dimensional mean field treatment of CSR effects discussed in [1]. We use Frenet-Serret coordinates with respect to the reference curve and in addition to the lab system we have two coordinate systems, which we call “beam system 1” and “beam system 2.” The former uses $u = ct$ as independent variable, and the latter uses $s$ is the independent variable and $u$ is a dependent variable [31]. The equations of motion in the beam system are

| Parameter | Symbol | Value | Unit |
|-----------|--------|-------|------|
| Energy reference particle | $E_r$ | 233 | MeV |
| Peak current | $I$ | 120 | A |
| Bunch charge | $Q$ | 1 | nC |
| Norm. transverse emittance | $\gamma \epsilon_0$ | 1 | $\mu m$ |
| Alpha function | $\alpha_0$ | 0 | |
| Beta function | $\beta_0$ | 10 | m |
| Linear energy chirp | $h$ | -27.5 | 1/m |
| Uncorrelated energy spread | $\sigma_E$ | 2 | KeV |
| Momentum compaction | $R_{56}$ | 0.025 | m |
| Radius of curvature | $\tau_0$ | 5 | m |
| Magnetic length | $L_b$ | 0.5 | m |
| Distance 1st-2nd, 3rd-4th bend | $L_1$ | 2.5 | m |
| Distance 2nd-3rd bend | $L_2$ | 1 | m |

### B. 1d Field Approximation Scheme

We now discuss a 1d field approximation to our 2d model. This approximation is based on the fact that the beam system spatial density is almost stationary in the $(\tilde{z}, \tilde{x})$ coordinates given by eq. (38) of [1]. The 1d field approximation takes $\tilde{x} = 0$ and the $\tilde{R}$ in eq. (55) becomes

$$\tilde{R} = R_c(s) + M(s)/[1 + hR_{56}(s), hD(s)]^2 \tilde{z}. \quad (56)$$

The accuracy
FIG. 16: Comparison of the mean power calculated with the 2d model [1] and the 1d field approximation scheme [30].

FIG. 17: Comparison of the x-emittance calculated with the 2d model [1] and the 1d field approximation scheme [30].

FIG. 18: Plot of $F_{z1}(\tilde{z}, \tilde{x}, s)$ in normalized coordinates at $s = 5\text{m}$ (exit third magnet). The 1d field approximation scheme uses $F_{z1}(\tilde{z}, 0, s)$ (blue line).

FIG. 19: The same as in Figure 18 for $s = 8\text{m}$ (end of chicane).

of the scheme is shown in Figures 16 and 17. The final value of the transverse emittance is 1.44\text{\textmu m}, to be compared with 1.45\text{\textmu m} of the exact 2d model. The calculation is considerably faster than the 2d calculation as it reduces the computational cost by a factor proportional to the number of grid points in $\tilde{x}$, i.e., the computational effort is $O(N_zN_vN_\theta)$. This approximation relies on a weak dependency of $F_{z1}$ on its second argument for fixed values of the first, as illustrated in Figures 18 and 19 where we have plotted $F_{z1}(\tilde{z}, \tilde{x}, s) = (q/P_e c)E_{ij}(\mathbf{R}, s) \cdot t(s)$ in $(\tilde{z}, \tilde{x})$ coordinates that put the 5\sigma range of the tilde variables on the square $[-1, 1]^2$. This weak dependence is satisfied to good approximation where the CSR force has its maximum strength, namely inside the magnets, with the worst case scenario for $s = 5\text{m}$ illustrated by Figure 18.

C. Gain Factor Calculation

We calculate the gain factor for $25\text{\textmu m} \leq \lambda \leq 200\text{\textmu m}$ with the 1d approximation scheme and compare the results with the 2d model and analytical theory. The computational cost of the field calculation in the 2d model is $O(N_zN_vN_\theta)$, while in the 1d field approximation scheme is $O(N_zN_vN_\theta)$. The computational cost for the density estimation based on TWT and TFCT methods is dominated, for large particle numbers $N$, by the cost in the particle deposition scheme which is $O(N)$, while the Monte Carlo cosine approximation has a computational load of $O(M_zM_xN)$. According to Table III, for the calculation of the gain factor at $\lambda = 100\text{\textmu m}$ we have $N_z = 1024$, $N_x = 128$, $M_z = 100$, $M_x = 40$ and $N = 5 \times 10^7$. In typical simulations $N_v = 1000$, and, given that $N_\theta = N_z$, we have that the computational cost of the field calculation in the 2d model and in the Monte Carlo cosine approximation are both $O(10^{11})$. Therefore the use of both the TWT and TFCT methods reduces the computational load by a factor of two for $\lambda = 100\text{\textmu m}$. Notice that the computational cost of the field calculation in the 1d approximation scheme is $O(10^9)$, still dominating over the cost in the particle deposition scheme which is $O(10^7)$. The numerical simulations discussed in this paper have been done on the parallel cluster NERSC. The
FIG. 20: Comparison of the longitudinal density at the exit of the chicane for $\lambda = 100\mu m$ obtained with the TWT and TFCT methods. Top right: comparison of TWT and TFCT with $N_z = 2048$. Top left: comparison of TWT with $N_z = 2048$ and $N_z = 1024$. Bottom left: comparison of TFCT with $N_z = 2048$ and $N_z = 1024$. Bottom right: 2d spatial density at the end of the chicane for $\lambda = 100\mu m$ obtained with TWT. The results are obtained in accordance to Table III.

Simulation for $\lambda = 100\mu m$ was run on the NERSC Cray XT4 system Franklin and took approximately 8 hours with the 2d model and approximately 5 minutes with the 1d field approximation scheme.

In Figure 20 we compare the longitudinal density for $\lambda = 100\mu m$ at the end of the chicane calculated with TWT and TCFT. We see clearly that TFCT with $N_z = 2048$ gives the same accuracy of TWT with $N_z = 1024$, thus proving that TWT is more efficient than TFCT in its ability to resolve small scale structures. It is encouraging to see, however, that the results from the two methods converge as the grid resolution is increased.

In Figure 21 we plot the gain factor as a function of the initial modulation wavelength $\lambda$. The blue line gives the results with the 1d field approximation scheme obtained with the TWT method, compared to the results with the 2d model obtained with the TFCT method, as discussed in [32]. Simulations akin to those in Figure 20 show that the gain factor calculation with the two methods are quite close to each other. The 1d and 2d numerical results differ at short wavelengths from the analytical formula and agree quite well with each other, showing a decrease of the gain factor for the range $\lambda = 70 - 100\mu m$. Figure 20 (bottom right) shows the 2d spatial density at the end of the chicane for $\lambda = 100\mu m$. Figure 22 shows the longitudinal density at the end of the chicane for a range of values of $\lambda$; it clearly demonstrates (top left) that the modulation wavelength $\lambda = 25\mu m$ leads to a mild microwave instability.

The discussion so far has been focused on the calculation of the gain factor, that by definition gives the amplification factor of an initial perturbation at a certain wavelength determined by the initial modulation and compression factor of the chicane. A more detailed study would consider an initial modulation with a more general spectral content, to study coupling between modes, and would analyze the amplification process in the full spectrum of the longitudinal density. The latter study has been performed in [1] with the 2d model. Moreover, in design studies of FELs, the distribution at entrance of
the bunch compressor is obtained by numerically tracking the beam from the injector to the bunch compressor. A more realistic modeling of the microbunching instability would therefore consist in the study of the evolution of an arbitrary distribution at entrance of the bunch compressor. These studies are on our agenda.

A crucial parameter that affects the microbunching instability is the uncorrelated energy spread. An important prediction of the gain factor formula from [6] is that increasing the uncorrelated energy spread reduces the gain factor. This led to a proposal, the laser heater, to increase the uncorrelated energy spread within FEL tolerance in order to damp the microbunching instability without degrading the FEL performance. An analysis of this effect together with the study of the evolution of an arbitrary distribution at the entrance of the bunch compressor will be discussed in a forthcoming paper.

VI. DISCUSSION AND CONCLUSION

We presented an approximation of the charged distribution function which provides a superior alternative to the cosine expansion currently in use in Bassi et al. [1] code. The new method deposits the particles on the finite grid — an approximation to the charged particle distribution which is intrinsically plagued by numerical noise — and then uses wavelet thresholding to remove some of this noise. The resulting wavelet-based grid approximation is appreciably more accurate — as quantified by the IMSE — and more than three orders of magnitude more efficient. As a result of this drastic computational speedup in the density estimation, the computational load for the simulations with the 2d model is reduced by a factor of two, allowing a more efficient study of the evolution of the small-scale structures in the beam. Resolving these small scale structures requires simulations with a very large number of point particles, which is prohibitively expensive with the original Monte Carlo method that scales as \( \propto M_x M_z N \). This improvement is even more dramatic for the simulations with the 1d field approximation scheme, where the computational load is reduced by two orders of magnitude, allowing for a very fast calculation of the microbunching instability.

One of the major computational efforts in the present implementation of Bassi’s code is the storage of the history of the beam on a 3d grid (2d in space and 1d in time). In the future, it will be beneficial to explore possibilities of further optimizing computational efficiency of the new wavelet-based approach by exploiting the sparsity of the charge representation (see Figure 5). Storing the entire history of the charge distribution as a small set of sparse wavelet coefficients instead of the 3d full grid could provide a substantial savings in memory and processor communication, thus significantly reducing simulation times. This study is currently underway.

[1] G. Bassi, J. Ellison, K. Heinemann, and R. Warnock, Phys. Rev. ST Accel. Beams 12, 080704 (2009).
[2] G. Bassi and B. Terzić, Proceedings of PAC 2009, Vancouver. Paper TH5PFP043 (2009).
[3] K. Bane, F.-J. Decke, Y. Ding, D. Dowell, P. Emma, J. Frisch, Z. Huang, R. Iverson, C. Limborg-Deprey, H. Loos, et al., Phys. Rev. ST Accel. Beams 12, 030704 (2009).
[4] M. Borland, Nucl. Instrum. Methods Phys. Res., Sect. A 483, 268 (2002).
[5] E. Saldin, E. Schneidmiller, and M. Yurkov, Nucl. Instrum. Meth. Phys. Res. A 298, 373 (1997).
[6] Z. Huang and K. Kim, Nucl. Instrum. Methods Phys. Res., Sect. A 5, 074401 (2002).
[7] S. Heifets, G. Stupakov, and S. Krinsky, Nucl. Instrum. Methods Phys. Res., Sect. A 5, 064401 (2002).
FIG. 22: Longitudinal density at the exit of the chicane with and without self-fields for $\lambda = 25\mu m$ (top left), $\lambda = 50\mu m$ (top right), $\lambda = 75\mu m$ (center left), $\lambda = 100\mu m$ (center right), $\lambda = 150\mu m$ (bottom left) and $\lambda = 200\mu m$ (bottom right). The results are obtained with the TWT method.

[8] Z. Huang, M. Borland, P. Emma, J. Wu, C. Limborg, G. Stupakov, and J. Welch, Phys. Rev. ST Accel. Beams 7, 074401 (2004).
[9] M. Borland, Phys. Rev. ST Accel. Beams 11, 030701 (2009).
[10] http://www.lnf.infn.it/conference/uBI10/.
[11] R. Li, Nucl. Instrum. Meth. Phys. Res. A 429, 310 (1998).
[12] R. Li, The Second ICFA Advanced Accelerator Workshop on The Physics of High Brightness Beams (1999).
[13] R. Warnock, G. Bassi, and J. Ellison, Nucl. Instrum. Meth. Phys. Res. A 558, 85 (2006).
[14] S. Efroymovich, Nonparametric Curve Estimation: Methods, Theory, and Applications (Springer, Berlin, 1999).
[15] R. W. Hockney and J. Eastwood, Computer Simulations Using Particles (IOP Publishing Ltd, Bristol, 1988).
[16] B. Terzić, I. Pogorelov, and C. Bohn, Phys. Rev. ST Accel. Beams 10, 034201 (2007).
Appendix A: Variance of Noise For the Particle Deposition Schemes

In this Appendix we detail the process of sampling a continuous charge density distribution by \( N \) particles, with subsequent charge deposition on a grid. The detailed discussion of the two lowest-order PDS — NGP \((k = 1)\) and CIC \((k = 2)\) — is presented in the Appendix A of [16]. Here we extend their discussion to TSC \((k = 3)\) and present the generalized approach for an arbitrary order PDS \((k > 3)\).

We calculate the expectation and variance of \( Q^{(i)} \), the total charge assigned to the \( i^{th} \) node of the lattice, for the general PDS (arbitrary \( k \)). Each particle carries the same charge \( q_0 = Q_{tot}/N \).

The total charge \( Q^{(i)} \) deposited onto the gridpoint \( i \) can be interpreted as the sum of \( N \) independent and identically distributed random variables \( \{Q_1, ..., Q_N\} \):

\[
\begin{align*}
Q^{(i)} &= \sum_{k=1}^{N} Q_k; \\
\end{align*}
\]

one may consider this sampling process as particles being added one at the time. All \( \{Q_k\} \) are then distributed as the “prototype” random variable \( Q \), the charge assigned to the gridpoint \( i \) when a new particle is added to the sample. We also introduce an auxiliary variable \( I \), which assumes value 1 if the sampled position of a \( Q_k \) is within the domain \( \Omega^{(i)} \) of the PDS function. For the \( k^{th} \) order PDS, \( \Omega^{(i)} \) is defined to be a \( D \)-dimensional cube of edge length \( kh \), centered on the node \( i \).

The working assumption here is that the grid is sufficiently fine for the probability density function to be approximated by a linear function on \( \Omega^{(i)} \). After adopting this, one can readily show:

- \( k = 1 \):

\[
\begin{align*}
E[Q^2|I = 1] &= \frac{q_0^2}{2D}, \\
E[Q|I = 1] &= \frac{q_0}{2D}, \\
Var(Q|I = 1) &= \left( \frac{1}{2} - \frac{1}{4} \right) q_0^2. \\
\end{align*}
\]

- \( k = 2 \):

\[
\begin{align*}
E[Q^2|I = 1] &= \frac{q_0^2}{3D}, \\
E[Q|I = 1] &= \frac{q_0}{3D}, \\
Var(Q|I = 1) &= \left( \frac{1}{3} - \frac{1}{4} \right) q_0^2. \\
\end{align*}
\]

- \( k = 3 \):

\[
\begin{align*}
E[Q^2|I = 1] &= \frac{q_0^2}{11(60)} D, \\
E[Q|I = 1] &= \frac{q_0}{3D}, \\
Var(Q|I = 1) &= \left( \frac{11}{60} - \frac{1}{9} \right) q_0^2. \\
\end{align*}
\]

- \( k > 3 \):

\[
\begin{align*}
E[Q^2|I = 1] &= q_0^2 \int_{-\frac{k}{2}}^{\frac{k}{2}} d^2_k(x) dx = q_0^2 \sum_{i=1}^{D} a_k^2, \\
E[Q|I = 1] &= \frac{q_0}{kD}, \\
Var(Q|I = 1) &= \left[ \sum_{i=1}^{D} a_k^2 - \left( \frac{1}{k^2} \right) \right] q_0^2. \\
\end{align*}
\]

\( E[Q|I = 0] = 0 \) and \( Var(Q|I = 0) = 0 \) for all PDSs.

We now define a new random variable \( K \) as the number of particles for which \( I = 1 \). In a given realization, \( K \) will have a value \( k \) between 0 and \( N \) (0 < \( k \) < \( N \)). For any given \( k \),

\[
E(Q|K = k) = kE[Q|I = 1] 
\]
and
\[
\text{Var}(Q^{(i)}|K = k) = k\text{Var}(Q|I = 1), \quad (A11)
\]
because the \(\{Q_1, ..., Q_N\}\) are independent and identically distributed (as \(Q\)). \(K\) itself is a random variable, so the expectation and variance of \(Q^{(i)}\) has to be calculated using the double expectation theorem:
\[
E[Q^{(i)}] = E[E[Q^{(i)}|K]] = E[K]\text{Var}(Q|I = 1), \quad (A12)
\]
\[
\text{Var}(Q^{(i)}) = E[\text{Var}(Q^{(i)}|K)] + \text{Var}(E[Q^{(i)}|K]) \quad (A13)
\]
\[
= E[K]\text{Var}(Q|I = 1) + (E[Q|I = 1])^2\text{Var}(K).
\]
The calculation of \(E[K]\) and \(\text{Var}(K)\) that enters the expressions above is aided by the realization that \(K\) is the number of “successes” \((I = 1)\) in a series of \(N\) trials. It is, therefore, distributed according to a binomial distribution with “probability of success” \(P_i\) equal to the continuous probability density function integrated over the \(\Omega^{(i)}\):
\[
E[K] = NP_i, \quad (A14)
\]
\[
\text{Var}(K) = NP_i(1 - P_i). \quad (A15)
\]
Analogously to \(P_i\), we define \(p_i\) as the value of the integral of the continuous probability density over \(\omega^{(i)}\), a \(D\)-dimensional cube of edge length \(h\) centered on the node \(i\). With this notation:

- \(k = 1:\)
  \[
P_i \approx 2^D p_i, \quad (A16)
  \]

- \(k = 2:\)
  \[
P_i \approx 2^D p_i, \quad (A17)
  \]

- \(k = 3:\)
  \[
P_i \approx 3^D p_i, \quad (A18)
  \]

- \(k > 3:\)
  \[
P_i \approx k^D p_i, \quad (A19)
  \]

After combining eqs. (A2), (A4), (A6), (A8), (A10), (A12), (A14), and (A16)-(A19) we obtain
\[
E[Q^{(i)}] = q_0 NP_i \quad (A20)
\]
for all PDS, and, finally,

- \(k = 1:\)
  \[
  \text{Var}(Q^{(i)}) = q_0^2 NP_i \left( \frac{1}{2^D} - \frac{P_i}{4^D} \right)
  \approx q_0^2 NP_i (1 - P_i); \quad (A21)
  \]

- \(k = 2:\)
  \[
  \text{Var}(Q^{(i)}) = q_0^2 NP_i \left( \frac{1}{3^D} - \frac{P_i}{4^D} \right)
  \approx q_0^2 NP_i \left( \frac{2}{3} \right)^D - p_i \]; \quad (A22)

- \(k = 3:\)
  \[
  \text{Var}(Q^{(i)}) = q_0^2 NP_i \left( \frac{11}{60} \right)^D - \frac{P_i}{3^D}
  \approx q_0^2 NP_i \left[ \frac{k^D a_k^2 - p_i}{2} \right]
  \approx q_0^2 NP_i \left[ a_k^2 - p_i \right]. \quad (A23)
  \]

- \(k > 3:\)
  \[
  \text{Var}(Q^{(i)}) \approx \frac{1}{N_{\text{grid}}} \sum_{i=1}^{N_{\text{grid}}} \text{Var}(Q^{(i)}). \quad (A24)
  \]

After substituting eqs. (A21)-(A24) into eq. (A24), we observe that the second term is bounded
\[
\frac{1}{N_{\text{grid}}} \leq \sum_{i=1}^{N_{\text{grid}}} p_i^2 \leq 1. \quad (A25)
\]
The lower bound is reached for the uniform distribution, while the upper bound is attained when the whole distribution is assigned to a single gridpoint. The charge density distribution in a typical PIC simulation very rarely possesses highly prominent, strongly localized peaks; furthermore, the grid is always adjusted so as to minimize (or, at least, greatly reduce) the number of grid points in the regions of zero density. This means that the value of the term will be much closer to its lower end, thus yielding it essentially negligible in computing \(\sigma^2\) using eq. (A24). Using this simplification, for the PDS schemes considered here we obtain
\[
\sigma^2 \approx \frac{a_k^2 q_0^2 N}{N_{\text{grid}}} = \frac{a_k^2 Q_{\text{tot}}^2}{NN_{\text{grid}}}, \quad (A26)
\]
where \(a_k\) are given in eq. (19). With this formalism, one can compute \(\sigma_k\) for the PDS \(d_k(x)\) of any order \(k\). As we mentioned before, one can easily compute the \(k^{\text{th}}\) order PDS by convolving the \(k - 1^{\text{th}}\) order PDS with the second-order PDS \(d_2(x)\) [15].
Appendix B: Near-Equivalence of Monte Carlo Cosine and TFCT Expansions From a NGP Grid

The cosine coefficients computed from a NGP grid are given by combining eqs. (22) and (24):

\[
\theta_{nm} = q_0 \sum_{i=1}^{N} \int_{0}^{1} \int_{-h_x/2}^{h_x/2} \int_{0}^{1} \int_{-h_z/2}^{h_z/2} \delta(z - z_i) \delta(x - x_i - \bar{x}) d_1(\bar{z}) d_1(\bar{x}) \phi_n(z) \phi_m(x) dx d\bar{x} d\bar{z} dz dx
\]

\[
= q_0 \sum_{i=1}^{N} \int_{-h_x/2}^{h_x/2} \int_{-h_z/2}^{h_z/2} \phi_m(z_i + \bar{z}) \phi_n(x_i + \bar{x}) d_1(\bar{z}) d_1(\bar{x}) d\bar{x} d\bar{z}
\]

\[
= q_0 \sum_{i=1}^{N} \left( \phi_n(z_i) \int_{-h_z/2}^{h_z/2} d_1(\bar{z}) d\bar{z} \right) \left( \phi_m(x_i) \int_{-h_x/2}^{h_x/2} d_1(\bar{x}) d\bar{x} \right) \approx q_0 \sum_{i=1}^{N} \phi_n(z_i) \phi_m(x_i) + O(h_z^2) + O(h_x^2).
\]

The sum \( q_0 \sum_{i=1}^{N} \phi_n(z_i) \phi_m(x_i) \) is the expression for the coefficients in the Monte Carlo expansion, which can be readily seen from eq. (8). The same treatment can easily be extended to the coefficients of the current density \( \Theta_{nm} \). In the previous calculation, we first switched the order of integration, then integrated over \( z \) and \( x \), after which we expanded in Taylor series \( \phi(z_i + \bar{z}) \) around \( z_i \) and \( \phi(x_i + \bar{x}) \) around \( x_i \) (because \( z_i \gg h_z \) and \( x_i \gg h_x \)). In the Taylor expansion, the linear term vanishes because \( h_z \) is over the entire stencil centered at \( z_i \). Earlier, we defined the IMSE as the sum of the square of differences between the numerically-computed solution, such as the one given in eq. (C1), and the exact solution

\[
D_z^{(l)} f(z_i, x_j) = \sum_{n=0}^{M_z} \sum_{m=0}^{M_x} \theta_{nm} \phi_m(x_j) \sum_{\kappa} c_\kappa \phi_n(z_i + \kappa h_z),
\]

\( c_\kappa \) are the relative weights on the points of the stencil (see Table V). The range of \( \kappa \) is over the entire stencil centered at \( z_i \). Earlier, we defined the IMSE as the sum of the square of differences between the numerically-computed solution, such as the one given in eq. (C1), and the exact solution

\[
\theta_{nm}^{(l)} f(z_i, x_j) = \sum_{n=0}^{M_z} \sum_{m=0}^{M_x} \theta_{nm} \phi_n(z_i) \phi_m(x_j),
\]

where \( \theta_{nm} \) are the exact coefficients of the distribution. The exact and the numerically-computed coefficients are related by \( \theta_{nm} = \theta_{nm}^{(l)} + \delta \theta_{nm} \).

It can be readily shown that

\[
\sum_{\kappa} c_\kappa \phi_n(z_i + \kappa h_z) = \phi_n^{(l)}(z_i) + \delta \phi_n,
\]

where

\[
\delta \phi_n \propto O(h_z^p) \propto O(N_z^{-p}).
\]

Appendix C: Error in Using Finite Difference Scheme To Evaluate Derivatives

Finite difference schemes are used to evaluate derivatives of a gridded distribution. The order of the derivative approximation depends on the number of points used in the differencing, i.e., the size of the stencil. The \( p^{th} \)-order accurate \( p^{th} \) \( z \)-derivative computed using a differ-
It follows that the IMSE for the $i^{th}$ $z$-derivative is

$$E_i = h_z h_x \sum_{i=1}^{N_z} \sum_{j=1}^{N_x} \left[ D_z^{(i)} f(z_i, x_j) - \delta_z^{(i)} \bar{f}(z_i, x_j) \right]^2 \quad (C4)$$

$$= h_z h_x \sum_{j=1}^{N_x} \sum_{m=0}^{M_z} \phi_m^2(x_j)$$

$$\times \sum_{i=1}^{N_z} \sum_{n=0}^{M_x} \left[ \theta_{nm} \delta \phi_n + \delta \theta_{nm} \phi_n^0(z_i) + \delta z \phi_n \delta \phi_n \right]^2.$$ 

The first term in the bracket is due to the error of the finite difference scheme, and depends only on the type (order) of the scheme. The second is due to the error in computing expansion coefficients, and depends on the number of particles ($N$) and the grid resolution parameters ($N_z, N_x$). The third term is due to both error in computing expansion coefficients and in the finite difference scheme. This term is of the second order, and therefore not a significant contributor to the overall error, which is why we ignore it in our analysis.

From our earlier discussion, the IMSE for the distribution scales as $N^{-1}$, and can be written as

$$E = h_z h_x \sum_{i=1}^{N_z} \sum_{j=1}^{N_x} \left[ f(z_i, x_j) - \bar{f}(z_i, x_j) \right]^2 \quad (C5)$$

$$= \sum_{n=0}^{M_x} \sum_{m=0}^{M_z} \delta \theta_{nm}^2 \left[ h_z h_x \sum_{i=1}^{N_z} \sum_{j=1}^{N_x} \phi_n^2(z_i) \phi_m^2(x_j) \right],$$

$$\approx \sum_{n=0}^{M_x} \sum_{m=0}^{M_z} \delta \theta_{nm}^2 \propto N^{-1}.$$ 

The last line follows from the fact that in the limit of increasing resolution ($h_z, h_x \to 0$), the bracketed term converges to the integral $\int_0^1 \int_0^1 \phi_n^2(x) \phi_m^2(x) dx dz$. Therefore, the total IMSE in the approximation of derivatives consists of two main constituent errors which scale as $O(N^{-2})$ and $O(N^{-1})$; their relative importance changes with the parameters of the simulation.

Figure 23 shows the accuracy of estimation of the first derivative in $z$ as a function of the length scale $\lambda$ for the grid, TFCT, TWT and the exact distribution (the first term in the brackets of eq. (C4)), evaluated with a difference scheme of the first (top), second (middle) and third (bottom) order. It is evident that the error of the derivative estimate is dictated by the inaccuracy in estimating expansion coefficients, and that using higher-order difference scheme for evaluation of derivatives does not lead to more accurate estimates. We also notice that the estimates of the first derivatives using TFCT and TWT are quite similar.
Acknowledgments

We gratefully acknowledge R. Ryne and P. Spentzouris for an account on NERSC. B.T. was partially supported by ONR contract number N00014-36-1-0587 while at Northern Illinois University.