



UNIVERSITY
of York

Reduction Algorithms for Simulation of Laser Wakefield Acceleration

Jasper Grimm

MSc (by Research)

University of York

Physics

September, 2020

Abstract

We present results from simulations, using the 2D particle-in-cell (PIC) code EPOCH, in which we analyse the suitability of low transverse field resolution for the study of laser wakefield acceleration (LWFA). The simulations use a tailored density profile to stimulate injection, maintaining a constant post-injection density. We also examine the applicability of reducing simulations post-injection, both through algorithmic nearest-neighbour (NN) particle coalescing, and through a combination of variation in the number of particles-per-cell (PPC) with field interpolation.

For low transverse field resolution, we demonstrate that both the peak energy injected and the resultant beam emittance converge rapidly, when the overall domain size is held constant, with little variation beyond 1.2 cells-per-wavelength (CPW) in the transverse direction. Results also show that whilst the proposed NN particle coalescing method maintains the injected electron beam, this degrades sufficiently that it does not improve accuracy over low transverse resolution, whilst being more computationally expensive. In the case of variable PPC, we show that this method can exactly preserve the electron beam profile for a PPC decrease from 32 to 1, whilst offering an order-of-magnitude speedup in post-reduction computation. Through the incorporation of field interpolation, we demonstrate minimal degrading in the simulated electron beam profile for a 20-fold decrease in the number of transverse simulation cells post-injection, with a corresponding computational speedup of factor 90.

Keywords— laser wakefield acceleration – pseudo-particle – macro-particle – simulation reduction – particle coalescing – particle-in-cell – particle simulation

Table of Contents

Abstract	2
Contents	6
List of Tables	8
List of Figures	16
Acknowledgements	17
Declaration	18
Abbreviations	19
1 Introduction	20
2 Theoretical Background	23
2.1 Laser Wakefield Acceleration	23
2.1.1 Ponderomotive Force	24
2.1.2 Langmuir Waves	26
2.1.3 Self-injection	27
2.1.4 Shock Injection	28
2.2 Quantifying Beam Quality	29

2.2.1	Beam Energy and Charge	30
2.2.2	Normalised RMS Emittance	30
2.3	Plasma Particle-in-cell Simulation	32
2.3.1	Fundamental Theory	32
2.3.1.1	Pseudo-particles and Weighting Functions	32
2.3.1.2	Particle Pusher	33
2.3.1.3	Field Solver	35
2.3.1.4	Current Calculation	36
2.3.1.5	Stability Conditions	36
2.3.2	Increasing Simulation Speed	37
2.3.2.1	Existing Schemes	37
2.3.2.2	Particle Reduction	38
3	Simulation Reduction	40
3.1	Nearest-Neighbour Reduction Method	40
3.1.1	Moving window	40
3.1.2	Transverse Cell Reduction	42
3.2	Parity Considerations	44
3.3	Variable PPC	45
4	Methodology	48
4.1	Aim and Objectives	48

4.2	Simulation Parameters	48
4.3	Convergence Tests	49
4.4	Transverse Simulation Cells	50
4.5	Reduction Schemes	51
4.5.1	Nearest-Neighbour Particle Averaging	52
4.5.2	Super-cell size	53
4.5.3	Variable PPC	55
5	Results	56
5.1	Convergence	56
5.1.1	Particles-per-cell	56
5.1.2	Cells-per-wavelength	58
5.2	Transverse Simulation Cells	59
5.3	Nearest-Neighbour Simulation Reduction	61
5.3.1	Nearest-Neighbour Attribute Averaging	62
5.3.2	Large Super-cells	64
5.3.3	Small Super-cells	65
5.4	Variable PPC Simulation Reduction	67
5.5	Reduction Spectra	70
5.6	Computation Time	72
6	Discussion	74

List of Tables

4.1	Simulation parameters to test convergence in particles-per-cell (n_{ppc}) and cells-per-wavelength (CPW), by varying n_{ppc} and the length of the moving window (l_x), respectively. For both, the number of longitudinal and transverse cells remains constant in accordance with the base parameters laid out in Chapter 4.2. Simulation 13 is identical to 6, corresponding to the base parameters, and is included for completeness although the same results will be used for both.	51
4.2	Simulation parameters describing the variation in transverse cells, with corresponding cell widths (size in transverse direction), whilst the absolute window dimension remains constant. Runs are grouped since each parameter set is repeated a total of 3 times, each with different random seeds.	52
4.3	Simulation parameters for comparing the effects of different particle attribute averaging methods when using nearest-neighbour reduction, where $n_{y'}$ is the reduced number of transverse cells, and n'_{ppc} the reduced PPC. Simulation 5 corresponds to a reduction in PPC only, of equivalent factor - since this is non-integer, this is rounded to 2 in the restarted simulation.	53
4.4	Simulation parameters for comparing reduction to varying numbers of transverse cells $n_{y'}$, for different "super-cell" resolutions. Simulations 1 – 6 here correspond to "large" super-cells, where each super-cell comprises 400,000 original simulation cells, whilst 7 – 12 correspond to "small" super-cells, being a factor 100 smaller.	54

4.5	Simulation parameters for variable PPC simulations; simulations 1 – 3 correspond to reducing to different numbers of particles-per-cell, whilst 4 – 6 reduce to 1 PPC with linear field interpolation. 7 is an additional run without field interpolation, but is based on a 301 cell simulation instead of 2001.	55
-----	---	----

List of Figures

1.1	Sample simulation frame showing the charge density (top), and momentum distribution as a function of longitudinal position (bottom).	21
2.1	Simulated laser pulse (red) propagating through a plasma from right to left, forming a wake [35]. (a) depicts the linear regime, with normalised laser intensity $a_0 = 0.5$, whilst (b) has an $a_0 = 4.0$, corresponding to the non-linear bubble regime. Vertical scale has been magnified by factor 10 for subfigure (a). Reprinted by permission from Springer: Nature, Nature Photonics, <i>Developments in laser-driven plasma accelerators</i> , S. M. Hooker, Copyright Macmillan Publishers Ltd (2013) [35].	23
2.2	Schematic of the displacement of a slab of electrons n_e , thickness L , within the plasma by a small distance δx . Light blue here corresponds to the displaced electrons, leaving behind an electropositive region of space.	26
2.3	Example of typical density profile used for shock injection (blue). As electrons ionised in close proximity to the density transition between (I) and (II) spend disproportionately longer in the accelerating field (since the plasma wavelength scales inversely with density), they are susceptible to trapping. Reproduced from Springer: Nature, Scientific Reports, <i>Shock assisted ionization injection in laser-plasma accelerators</i> , C. Thaury <i>et al</i> , Copyright Springer Nature (2015) under CC BY license [48].	29

- 2.4 A selection of 12 representative shots for both self injection **(a)** and shock injection **(b)**. It is clear that shock injection results in significantly narrower energy spectra in the injected bunch, although this comes at the cost of decreased total injected charge. Reproduced from APS: Phys. Rev. ST Accel. Beams, *Density-transition based electron injector for laser driven wakefield accelerators*, K. Schmid *et al*, under Creative Commons Attribution 3.0 license [26]. 30
- 2.5 Cycle by which the PIC scheme operates. **(top)**: charge and current density at the gridpoints is calculated from the current particle positions and velocities. **(right)**: weighting is applied to closest gridpoints to the pseudo-particles, by proximity to grid-point. **(bottom)**: Maxwell equations evaluated to calculate electric and magnetic fields at gridpoints. **(left)**: Weighting applied to particles, computing the force on particles and updating their position and velocity [56, 57]. Reprinted from Comp. Phys. Comm., **204**, M. Vranic *et al*, *Classical radiation reaction in particle-in-cell simulations*, Copyright (2016), with permission from Elsevier [58]. 33
- 2.6 Sketch of the leap-frog FDM integration method, depicting the staggering of the two calculations. Even though \mathbf{x}_t and \mathbf{v}_t are not known at the same time, they are advanced to $\mathbf{x}_{t+\Delta t}$ and $\mathbf{v}_{t+\Delta t}$. Reprinted by permission from Taylor and Francis: *Plasma Physics via Computer Simulation*, C. K. Birdsall and A. B. Langdon, Copyright Taylor & Francis Group LLC (2004) [57]. 34

2.7	A single Yee grid cell, in 2D. Starting from the centre of the cell, the point where E-field components are evaluated is shifted half a cell in the direction the field points, if possible. The B-field components are moved half a grid in all directions other than the direction it points. Reprinted from IOP Publishing: Plasma Phys. Control. Fusion, <i>Contemporary particle-in-cell approach to laser-plasma modelling</i> , T. D. Arber <i>et al</i> , under Creative Commons 3.0 Attribution (CC BY) licence [59].	36
3.1	Example simulation window with grid size (n_x, n_y) , at some arbitrary time post-injection. As the window moves, simulation cells are deleted from the back (left) and added to the front (right), with an initial n_{ppc} particles-per-cell. If we reduce, this thus involves a change in either transverse resolution (n_y'), particles-per-cell (n'_{ppc}), or both. This has been represented in red.	41
3.2	Total number of pseudo-particles in each simulation cell of a 32 PPC, 301 transverse cell simulation at some time after injection. It is clear that the actual number of pseudo-particles in each cell can vary substantially from the initial number (32), corresponding to the position in the wake.	42
3.3	Sample simulation cell at some arbitrary time after injection, showing the distribution of pseudo-particles (blue) and the corresponding momentum distribution.	42

3.4	Nearest-neighbour reduction on a single cell, sampling particles uniformly from the momentum distribution, and adding weights of removed particles to their closest (spatially) neighbouring sampled particle. (1) depicts a boundary problem, where the closest sampled particle may lie in an adjacent cell.	43
3.5	Nearest-neighbour reduction on a group of cells ("super-cell"), sampling particles uniformly from the momentum distribution, and adding weights of removed particles to their closest (spatially) neighbouring sampled particle.	44
3.6	Interpolation of the E_x field component at fixed x -position, corresponding to the location of an injected electron bunch. Due to the rapidly varying fields, the on-axis field strength is significantly overestimated when interpolating with even parity, which samples either side, whilst it is exactly sampled with odd parity.	45
4.1	The plasma density profile used for simulations (black), which is based on the profile (red, dashed) simulated by Swanson <i>et al.</i> [9], but with some modifications. Instead of an effective step width of $100\mu\text{m}$, a width of $25\mu\text{m}$ was used, with a flat post-step density as suggested by Baird [12].	50
5.1	The E_{peak} and E_{FWHM} dependence on the number of pseudo-particles per cell. The upper and lower boundaries of the FWHM energy are here plotted in (yellow), about the peak (red).	56

5.2	The dependence of beam emittance ($\varepsilon_{n, \text{tr}, \text{rms}}$) on the number of pseudo-particles per cell. Although the emittance appears to converge from 16 PPC, it increases again at 128 PPC, although this could be due to statistical variations.	57
5.3	The E_{peak} and E_{FWHM} dependence on the number of simulation cells (in the longitudinal direction) per laser wavelength.	58
5.4	The dependence of beam emittance ($\varepsilon_{n, \text{tr}, \text{rms}}$) on the number of simulation cells (in the longitudinal direction) per laser wavelength. The datapoint for 10 cells per wavelength has been distinguished since the beam emittance was incalculable (NaN).	59
5.5	The E_{peak} dependence on number of transverse simulation cells. The datapoints depicted are averaged over 3 simulations with different random seeds, with the standard deviation of these values being represented by an error bar.	60
5.6	The dependence of beam emittance ($\varepsilon_{n, \text{tr}, \text{rms}}$) on the number of simulation cells in the transverse direction. Data has been averaged over 3 simulations with different random seeds, depicting the standard deviation as in the form of an error bar. For 5 and 11 transverse cells, the beam was incalculable (NaN).	61

- 5.7 Comparison of the different particle attribute averaging (momentum and position) methods, relative to unreduced simulations of full width. **2001** and **101** are the full width simulations with respective numbers of transverse cells. **PPC** is a simulation reduced in PPC only using the NN algorithm, but maintains the field resolution. The others (**none**, **pos.**, **mom.**, **pos. + mom.**) correspond to: no averaging, position averaging, momentum averaging, and the averaging over both position and momentum, respectively. . . . 62
- 5.8 Here shown are the positions of the peak energies (E_{peak}), and positions of the upper- and lower FWHM energies (E_{FWHM}), for reduced simulations with "large" super-cells, as compared to the full-width control. 64
- 5.9 Here shown is the calculated beam emittance ($\varepsilon_{n, \text{tr}, \text{rms}}$), for the full-width control (black) and reduced (red) simulations with "large" super-cells. Again, the values for unreduced narrow-width simulations have been included for comparison, and the emittance for 5 and 11 transverse cells was incalculable for the unreduced simulation. 65
- 5.10 Here shown are the positions of the peak energies (E_{peak}), and positions of the upper- and lower FWHM energies (E_{FWHM}), for "small" super-cell reduced simulations as compared to the full-width control. For reduced widths of 5 and 11 cells, no beam was present. 66

- 5.11 Here shown is the calculated beam emittance ($\varepsilon_{n, \text{tr}, \text{rms}}$), for the full-width control (black) and reduced (red) simulations with "small" super-cells. Additionally, the values for unreduced narrow-width simulations have been included for comparison. Once again, the emittance for 5 and 11 transverse cells was incalculable for both reduced and unreduced simulations. 67
- 5.12 Depicted are the peak energies E_{peak} for unreduced (black) and variable PPC simulations (red), where the number of particles per cell is reduced from 32 to 1, and fields are interpolated linearly from the original 2001 transverse cell simulation. Additionally, the energy when reducing from a narrower simulation ($n_y = 301$ cells), with identical decrease in PPC, is depicted in blue. 68
- 5.13 Depicted are the calculated beam emittances $\varepsilon_{n, \text{tr}, \text{rms}}$ for unreduced (black) and variable PPC simulations (red), where the number of particles per cell is reduced from 32 to 1, and fields are interpolated linearly from the original 2001 transverse cell simulation. Additionally, the emittance when reducing from a narrower simulation ($n_y = 301$ cells), with identical decrease in PPC, is depicted in blue. 69
- 5.14 Depicted are the full energy spectra for various reduction methods. **(A)** full simulation with 2001 transverse cells, and 32 PPC. **(B)** NN reduction, using the "small" super-cells. **(C)** NN reduction, using the "large" super-cells. **(D)** variable PPC. **(E)** variable PPC, with additional field interpolation. 71

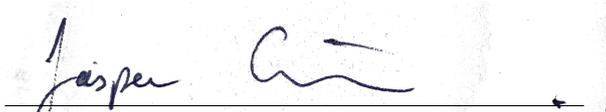
- 5.15 We here show the relationship between the (CPU-independent) computation time per simulated picosecond, and the reduction factor (the inverse product of relative number of cells, and relative PPC), for various reduction schemes corresponding to **(A, ..., E)** in Figure 5.14. In solid black **(A)**, we represent the full-width (2001 transverse cells, 32 PPC) simulation, used as a basis for reduction. In dotted black, we represent unreduced simulations with fewer transverse cells, assigning a virtual reduction factor of $(2001/ny)$. Red and orange represent nearest-neighbour approaches with large- and small super-cells, respectively, whilst the variable PPC method is depicted in blue. **(B)** and **(C)** correspond to 301 reduced cell simulations using small- and large super-cell NN reduction respectively. Variable PPC reduction to 1 PPC corresponds to **(D)**, whereas **(E)** further includes field interpolation to 301 transverse cells. 73
- 6.1 For a variable PPC simulation, the number of simulated macro-particles is shown (red) as function of simulation time post-reduction. In blue, the equivalent CPU-normalised computation time per simulated femtosecond is plotted, showing strong correlation with the number of particles as expected. 75

Acknowledgements

I would like to thank my supervisor, Dr. Chris Murphy, for his unwavering support and guidance throughout my project, and many inspiring discussions. For access to computational resources, essential for the many simulations, I also extend my gratitude to Dr. Raoul Trines, and the STFC Scientific Computing Department's SCARF cluster.

Declaration

I, Jasper Grimm, hereby declare that this thesis is a presentation of original work, of which I am the sole author. This work has not previously been presented for an award at this, or any other, University. All sources are acknowledged as References.


Signature

Date: 28/12/2019

List of Abbreviations

3DV	3-dimensional velocity
CPW	Cells-per-(laser)wavelength
EM	Electro-magnetic
FDM	Finite difference method
FEM	Finite element method
FWHM	Full width at half maximum
LPA	Laser-plasma accelerator
LWFA	Laser wakefield accelerator
LINAC	Linear accelerator
NN	Nearest neighbour
PBWA	Plasma beat wave accelerator
PIC	Particle-in-cell
PPC	Particles per cell
RF	Radio-frequency

1 | Introduction

The laser-plasma accelerator (LPA) was originally proposed by Tajima and Dawson in 1979 [1], following theoretical work that showed the suitability of plasmas as an accelerator medium. Akin to conventional particle accelerators, such as linear accelerators (LINACs) and synchrotrons, the principal aim of LPAs is the generation of high energy particles – in this case, electrons. Spanning a wide range of applications, including the production of radio-isotopes and diagnostics [2, 3] as well as ultrafast electron diffraction [4], the electron beam can also be used to generate X-rays, through methods such as Thompson backscattering [5–7] or betatron radiation [8]. This offers further uses, spanning from lithography, medical imaging and treatments, to the imaging of the structural dynamics of various biological and chemical systems [9].

Unlike conventional accelerators, which are limited by radio-frequency (RF) breakdown, plasma accelerators can sustain significantly stronger electromagnetic (EM) fields, and thus offer the potential to be significantly more compact. Whereas superconducting RF cavities are effective up to ~ 100 MeV/m [10], plasmas are able to sustain acceleration gradients in excess of 100 GeV/m – three orders of magnitude greater [11, 12]. This corresponds to a sizeable difference in scale (and thus cost) – whilst LINACs are typically in the range of several metres to kilometres, LPAs can achieve GeV electron energies on scales of millimetres to centimetres. Furthermore, LPAs also exhibit further useful characteristics such as femtosecond-level duration, and ultra-small bunch emittance [13, 14]; since the resultant electron bunches generated by an LPA is synchronised with the laser pulse(s), this provides further benefits in pump-probe applications [15].

Two distinct types of plasma accelerator were initially proposed [1] – the Plasma Beatwave Accelerator (PWBA), and the Laser Wakefield Accelerator (LWFA) –

with the latter not being feasible at time of conception due to insufficient development of compact, ultra-high power lasers capable of producing ultra-short pulses [15, 16]. Additionally, electrons were originally injected externally to be accelerated [17], before the development of more powerful lasers capable of producing intense electron beams solely from propagation of the laser pulse through the underdense plasma, as demonstrated in 2002 by [18].

The PBWA, first observed by Clayton *et al.* in 1985 [19], uses two (or more) long laser pulses, of frequency ω_1, ω_2 ; this resonantly excites a plasma wave when the difference between the pulse frequencies is similar to the electron plasma frequency [1, 20]: $\Delta\omega = \omega_1 - \omega_2 \approx \omega_p$. However, whilst this method provides control over the injection process, it introduces a number of problems – alignment of the lasers is critical [21], and resonant detuning [22] provides further difficulties.

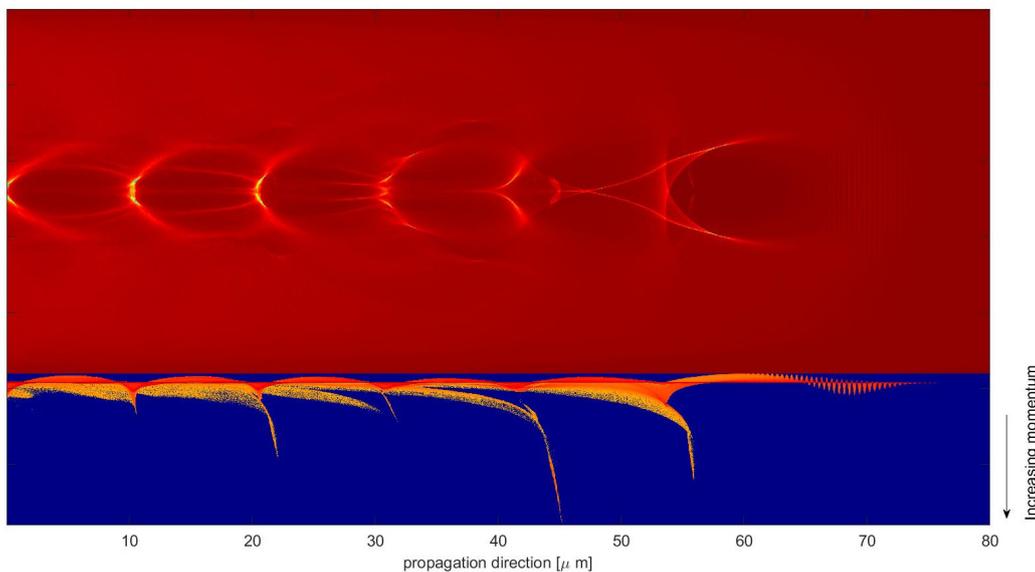


Figure 1.1: Sample simulation frame showing the charge density (top), and momentum distribution as a function of longitudinal position (bottom).

Unlike PBWA, only a single laser pulse is used in LWFA, and the pulse length $c\tau_L$ is directly matched to the plasma wavelength $c\tau_L \approx \lambda_p$. With the innovation

of chirped pulse amplification in 1985 [23] having allowed development the necessary power combined with short pulse length, experimental laser-wakefield accelerators became viable [10, 24]. Due to the inherently short pulses used, LWFA has the benefit of not being susceptible to Raman backscattering, and can operate with less uniform plasmas as it is not resonantly excited [15, 25]. The point of injection can then be controlled through various methods, such as tailored density profiles [26], ionization injection [4], and colliding pulses [17]. With control over the location of injection, this then allows for further fine-tuning in parameters affecting the characteristics of the generated electron beam.

When simulating density ramp injection, we would expect that the accuracy is largely dependant on the resolution in the interaction of the non-linear wake with a sharp density transition, and in the formation of the initial wake. We thus hypothesise that decreasing simulation resolution in the post-injection phase can significantly reduce computation time whilst preserving the detail in the injected bunch. Particularly for a constant post-injection background density, thereby effectively reducing the problem to one akin to an electrostatic accelerator, we expect significant potential speedups.

2 | Theoretical Background

2.1 Laser Wakefield Acceleration

In LWFA, a short, intense laser pulse is used to perturb oscillations in an underdense plasma, forming an electrostatic wake in the path of the laser [15, 27]; pulse lengths are typically around $\tau_L \lesssim 1$ ps, with a laser intensity $I_L \gtrsim 10^{18}$ W.cm⁻² [28]. Broadly speaking, there are two main regimes in LWFA – self-modulation [28, 29], and the bubble¹ regime [30, 31], first proposed in 2002 by Pukhov *et al.* [32]. We shall henceforth focus on the latter, which is capable of producing quasi-monoenergetic electron beams [21, 33], in which the laser induced ponderomotive force expels electrons radially, creating a highly non-linear, relativistic wake [34]. Whilst analytical solutions exist in the 3D linear, computational simulation is required to model the non-linear regime in more than one dimension [15].

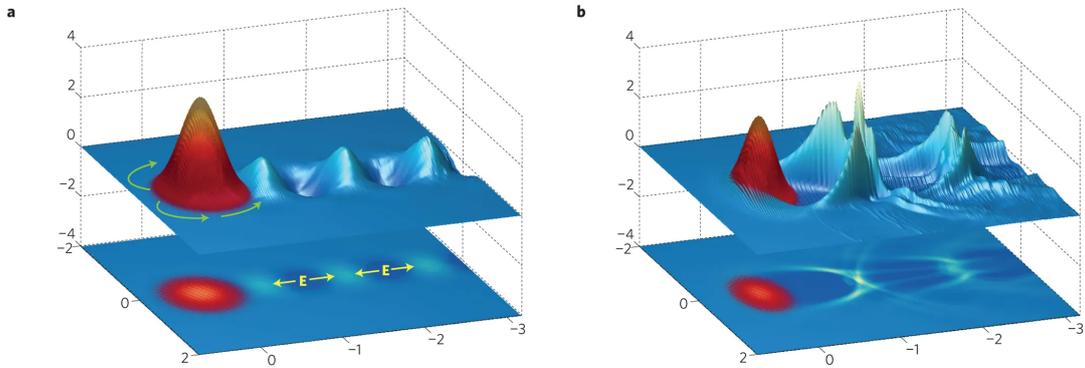


Figure 2.1: Simulated laser pulse (red) propagating through a plasma from right to left, forming a wake [35]. (a) depicts the linear regime, with normalised laser intensity $a_0 = 0.5$, whilst (b) has an $a_0 = 4.0$, corresponding to the non-linear bubble regime. Vertical scale has been magnified by factor 10 for subfigure (a). Reprinted by permission from Springer: Nature, Nature Photonics, *Developments in laser-driven plasma accelerators*, S. M. Hooker, Copyright Macmillan Publishers Ltd (2013) [35].

¹Sometimes also referred to as the "cavitation", or "blow-out" regime [21, 34].

2.1.1 Ponderomotive Force

In LWFA, it is the ponderomotive force which drives the wake, expelling charged particles from the path of the laser. To understand this force, let us consider the force on the electron fluid in a homogeneous plasma (in the cold fluid limit) exposed to some laser field [15]:

$$m_e \frac{d\mathbf{v}_e}{dt} = -e(\mathbf{E} + \mathbf{v}_e \times \mathbf{B}) \quad (2.1)$$

where \mathbf{v}_e is the plasma fluid element velocity, e and m_e the charge and mass of an electron respectively, and \mathbf{E} and \mathbf{B} the electric and magnetic fields. Let us now assume the laser field has a spatially-dependent amplitude (e.g. $\mathbf{E} = \mathbf{E}(\mathbf{x}) \sin(\omega_L t)$), with laser frequency $\omega_L \gtrsim \omega_{pe} \gg \omega_{pi}$ greater than or similar to the electron plasma frequency, which is much greater than the electron ion frequency. If we also substitute the total derivative² $d/dt = \partial/\partial t + (\mathbf{v}_e \cdot \nabla)$:

$$\frac{\partial \mathbf{v}_e}{\partial t} + (\mathbf{v}_e \cdot \nabla) \mathbf{v}_e = -\frac{e}{m_e} (\mathbf{E} + \mathbf{v}_e \times \mathbf{B}) \quad (2.2)$$

To the first order in $|\mathbf{E}|$, we thus find:

$$m_e \frac{\partial \mathbf{v}_e^{(1)}}{\partial t} = -e \mathbf{E}(\mathbf{x}) \sin(\omega_L t) \quad (2.3)$$

$$\mathbf{v}_e^{(1)} = \frac{e}{m_e \omega_L} \mathbf{E}(\mathbf{x}) \sin(\omega_L t) \quad (2.4)$$

²also commonly referred to as the: convective derivative, material derivative, Lagrangian derivative, Stokes derivative, particle derivative, advective derivative, and substantive derivative, amongst others [36].

where the superscript (1) is used to denote order. If we now consider the second order response [37]:

$$\frac{\partial \mathbf{v}_e^{(2)}}{\partial t} = -(\mathbf{v}_e^{(1)} \cdot \nabla) \mathbf{v}_e^{(1)} - \frac{e}{m_e} (\mathbf{v}_e^{(1)} \times \mathbf{B}^{(1)}) \quad (2.5)$$

Through a temporal integration of the Maxwell-Faraday equation ($\nabla \times \mathbf{E} = -\partial \mathbf{B}/\partial t$), we can find $\mathbf{B}^{(1)}$:

$$\mathbf{B}^{(1)} = \frac{1}{\omega_L} \nabla \times \mathbf{E}(\mathbf{x}) \cos(\omega_L t) = \frac{m_e}{e} (\nabla \times \mathbf{v}_e^{(1)}) \quad (2.6)$$

Substituting this back into equation (2.5) together with equation (2.4), we find:

$$\frac{\partial \mathbf{v}_e^{(2)}}{\partial t} = -(\mathbf{v}_e^{(1)} \cdot \nabla) \mathbf{v}_e^{(1)} - \mathbf{v}_e^{(1)} \times (\nabla \times \mathbf{v}_e^{(1)}) \quad (2.7)$$

$$\begin{aligned} m_e \frac{\partial \mathbf{v}_e^{(2)}}{\partial t} &= -\frac{e^2}{m_e \omega_L^2} (\mathbf{E} \cdot \nabla) \mathbf{E} - \frac{e^2}{m_e \omega_L^2} [\mathbf{E} \times (\nabla \times \mathbf{E})] \\ &= -\frac{e^2}{m_e \omega_L^2} [(\mathbf{E} \cdot \nabla) \mathbf{E} + \mathbf{E} \times (\nabla \times \mathbf{E})] \end{aligned} \quad (2.8)$$

Using the identity $\mathbf{a} \times (\nabla \times \mathbf{a}) \equiv \frac{1}{2} \nabla (\mathbf{a}^2) - (\mathbf{a} \cdot \nabla) \mathbf{a}$:

$$m_e \frac{\partial \mathbf{v}_e^{(2)}}{\partial t} = -\frac{e^2}{m_e \omega_L^2} \left[(\mathbf{E} \cdot \nabla) + \frac{1}{2} \nabla (\mathbf{E}^2) - (\mathbf{E} \cdot \nabla) \right] \quad (2.9)$$

$$= -\frac{e^2}{2m_e \omega_L^2} \nabla (\mathbf{E}^2) \quad (2.10)$$

Taking a temporal average, denoted by $\langle \rangle_T$, over the rapid laser field oscillations of period $T = \frac{2\pi}{\omega_L}$, we find that

$$\begin{aligned} \langle \mathbf{E}^2 \rangle_T &= \langle \mathbf{E}^2(\mathbf{x}) \sin^2(\omega_L t) \rangle_T = \mathbf{E}^2(\mathbf{x}) \langle \sin^2(\omega_L t) \rangle_T \\ &= \frac{1}{2} \mathbf{E}^2(\mathbf{x}) \end{aligned} \quad (2.11)$$

Substituting equation (2.11) back into equation (2.10), we thus arrive at the ponderomotive force

$$m_e \frac{\partial \langle \mathbf{v}_e^{(2)} \rangle_T}{\partial t} = -\frac{e^2}{4m_e \omega_L^2} \nabla (E^2) =: \mathbf{F}_p \quad (2.12)$$

It should be noted that this force contains both radial and axial components – it thus dispels electrons radially outward, as well as in the propagation direction of the laser.

2.1.2 Langmuir Waves

To understand the mechanism by which waves in the plasma (also referred to as Langmuir waves) are excited, let us consider the response of a quasi-neutral homogeneous plasma in the 1D linear regime. If a slab of electrons, thickness L , is displaced in x within the plasma by a small distance $\delta x \ll L$ (as in Figure 2.2), the charge density σ of the leading face is $\sigma = n_e e \delta x$.

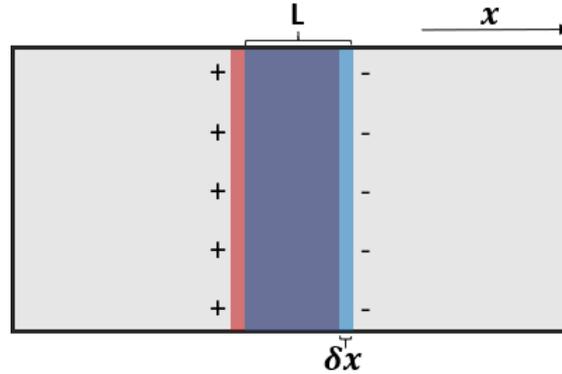


Figure 2.2: Schematic of the displacement of a slab of electrons n_e , thickness L , within the plasma by a small distance δx . Light blue here corresponds to the displaced electrons, leaving behind an electropositive region of space.

Since an equal but opposite charge density also develops on the opposite face, an electric field $E_x = -\frac{\sigma}{\epsilon_0} = -\frac{n_e e}{\epsilon_0} \delta x$ is generated [38]. Applying Newton's law,

each individual particle is thus subject to the restoring force [15]:

$$\begin{aligned} m_e \frac{d^2(\delta x)}{dt^2} &= eE_x = -\frac{n_e e^2}{\epsilon_0} \delta x \\ &= -m_e \left(\frac{n_e e^2}{\epsilon_0 m_e} \right) \delta x \\ &= -m_e \omega_{pe}^2 \delta x \end{aligned}$$

where we have defined the electron plasma frequency

$$\omega_{pe} = \sqrt{\frac{n_e e^2}{m_e \epsilon_0}} \quad (2.13)$$

This has the form of a harmonic oscillator, with associated period $\tau_{pe} = 2\pi/\omega_{pe}$, and describes the oscillation of electrons about their equilibrium position within the plasma.

For LWFA, it is critical that the plasma frequency $\omega_{pe} < \omega_L$ is less than the laser frequency; with a corresponding characteristic time scale longer than one period of the incident laser, the plasma is unable to stop the laser propagation [17]. This is defined as an "under-dense" plasma, and contrasts to an "over-dense" plasma which is capable of reflecting the laser pulse.

2.1.3 Self-injection

It is important to understand that the wake is not itself comprised of accelerated electrons, it is simply the collective oscillation of electrons about their equilibria. Instead, as the amplitude of the plasma wave continues to grow, eventually this exceeds a threshold E_{\max} , at which point the wave directly transfers energy to constituent particles [39]. This process does not necessarily destroy the wake, and can instead continuously "inject" electrons, thereby reducing its amplitude, until

this is once again below the threshold; this has further benefits in enabling injected particles to continue to gain energy from the plasma wave [33].

With a maximum electric field amplitude that can be supported by a plasma of $E_0 = m_e \omega_{pe} c / e$ in the cold, non-relativistic limit [40], it is possible for this to be exceeded in the non-linear regime [15]. Theoretical calculations in the 1D limit of the non-linear, relativistic, cold fluid equations [41] predict a maximum amplitude of

$$E_{\text{WB}} = \sqrt{2(\gamma_p - 1)} E_0$$

with relativistic Lorentz factor $\gamma_p = (1 - v_p^2/c^2)^{-1/2}$ associated with the plasma wave phase velocity v_p [15].

A simple estimate of the corresponding energy gain, assuming normalised laser parameter $a_0 \sim 1$, is given by $W \sim E_0 L_{\text{deph}} \sim (mc\omega_{pe})(\lambda_{pe}^3/\lambda_L)$ [42]. Here, L_{deph} is the linear dephasing length (the length over which electrons outrun the wake and are instead decelerated) and λ_{pe}, λ_L are the electron plasma- and laser wavelengths respectively.

2.1.4 Shock Injection

In order to exert better control over the precise location and timing of the injection, and to further improve the injected electron energy spectra, a variety of methods – such as colliding counter-propagating pulses – have been explored [43]. Another method involves the use of a tailored density profile, reducing the length over which electrons are injected and thus their energy spread, as demonstrated by Bulanov in 1998 through analytical theory supported by simulations [44], and various other groups [8, 26, 45]. This controls the wake phase-velocity v_p , and a sharp decrease in density (as in Figure 2.3) causes a phase mismatch between some electrons and the wave, causing these to be trapped in the accelerating re-

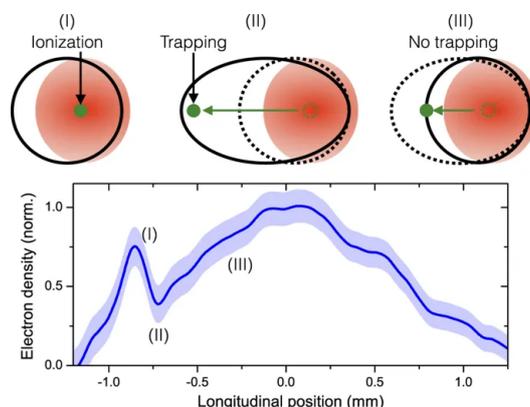


Figure 2.3: Example of typical density profile used for shock injection (blue). As electrons ionised in close proximity to the density transition between (I) and (II) spend disproportionately longer in the accelerating field (since the plasma wavelength scales inversely with density), they are susceptible to trapping. Reproduced from Springer: Nature, Scientific Reports, *Shock assisted ionization injection in laser-plasma accelerators*, C. Thaury *et al*, Copyright Springer Nature (2015) under CC BY license [48].

gion and thus injected [46, 47]. With the density gradient usually being produced by methods such as placing a razor blade in a supersonic gas jet [8, 9, 49], the density profile is comparatively easy to control, thus making the system – point of injection, as well as various beam properties – tuneable [12].

2.2 Quantifying Beam Quality

In order to be able to analyse and compare the resultant electron beam from LWFA simulations, we need to quantify the quality of the beam through macroscopic variables, which should also be experimentally measurable. Since the principal motivation is the generation of quasi-monoenergetic electron beams, it is essential to identify the peak in the injected electron kinetic energy spectrum E_{pk} , as well as the width of this peak ΔE_{pk} . Also of interest is the total charge injected Q_{tot} , as well as the charge in the principal peak, Q_{pk} ; a further useful quantity is the

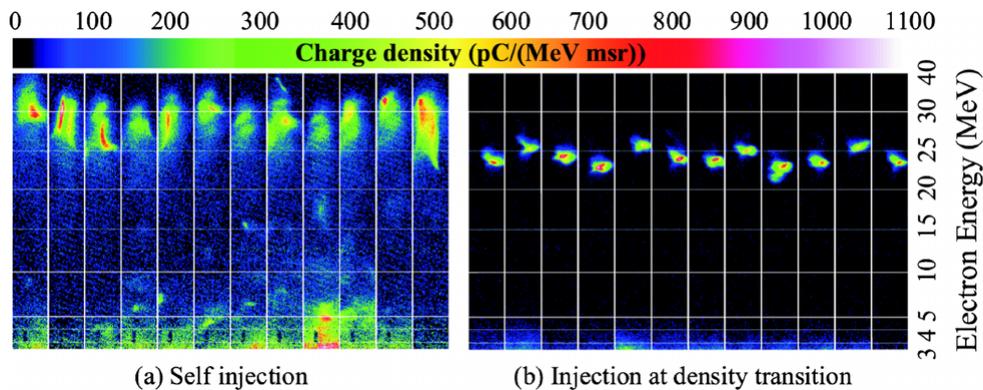


Figure 2.4: A selection of 12 representative shots for both self injection (a) and shock injection (b). It is clear that shock injection results in significantly narrower energy spectra in the injected bunch, although this comes at the cost of decreased total injected charge. Reproduced from APS: Phys. Rev. ST Accel. Beams, *Density-transition based electron injector for laser driven wakefield accelerators*, K. Schmid *et al*, under Creative Commons Attribution 3.0 license [26].

normalised beam emittance $\varepsilon_{n, \text{tr}, \text{rms}}$, which effectively represents the divergence in the beam.

2.2.1 Beam Energy and Charge

Whilst the calculation of the peak energy E_{pk} is self-evident, it is important to define the calculation of the width of this peak; we here define ΔE_{pk} to be the full width at half-maximum (FWHM).

The total charge Q_{tot} is similarly simple to calculate, simply summing over all injected electrons. For the charge in the principal peak Q_{pk} , however, we sum over electrons within the central peak of the electron kinetic energy spectrum, cutting the peak off at 10% of the height of E_{pk} , either side of the peak.

2.2.2 Normalised RMS Emittance

In addition to the previously discussed quantities, a further useful measure of beam quality is the beam emittance, which measures the average spread of the

beam particles in position-momentum phase space [50–52]. The RMS emittance, ε_{rms} , for the y - p_y projection of a beam propagating in x is given by [50]:

$$\varepsilon_{\text{rms}} = \sqrt{\langle y^2 \rangle \langle p_y^2 \rangle - \langle y p_y \rangle^2} \quad (2.14)$$

where $\langle \rangle$ represents the particle distribution's second central moment, defined as:

$$\begin{aligned} \langle AB \rangle &= \frac{\sum AB}{n} - \frac{\sum A \sum B}{n^2} & A, B \in \{y, p_y\} \\ &= \frac{\sum A^2}{n} - \left(\frac{\sum A}{n} \right)^2 & (\text{when } A = B) \end{aligned} \quad (2.15)$$

and summations are implicitly over all n particles ($\sum A \equiv \sum_{i=1}^n A_i$).

In trace-space, we calculate the phase-space areas in the y - y' space, instead of y - p_y , where we define [50, 51]:

$$\begin{aligned} y' &:= \frac{p_y}{p_x} \\ \Rightarrow \varepsilon_{\text{tr, rms}} &= \sqrt{\langle y^2 \rangle \langle y'^2 \rangle - \langle y y' \rangle^2} \end{aligned} \quad (2.16)$$

Normalising this, using³

$$\varepsilon_{n, \text{tr, rms}} = \overline{\gamma\beta} \varepsilon_{\text{tr, rms}} = \frac{\bar{\mathbf{p}}_x}{m_0 c} \varepsilon_{\text{tr, rms}}$$

we thus find the normalised rms emittance in trace-space:

$$\varepsilon_{n, \text{tr, rms}} = \frac{\bar{\mathbf{p}}_x}{m_0 c} \sqrt{\langle y^2 \rangle \langle y'^2 \rangle - \langle y y' \rangle^2} \quad (2.17)$$

³To show that these are equivalent, consider the relativistic momentum, $\mathbf{p} = \gamma m_0 \mathbf{v}$. Substituting the relativistic beta (β) for the velocity (\mathbf{v}), we find $\mathbf{p} = \gamma m_0 \beta c$, which can be re-arranged to $\gamma\beta = \frac{\mathbf{p}}{m_0 c}$. Assuming a radially symmetric beam, we know that $\bar{\mathbf{p}}_y = \bar{\mathbf{p}}_z = 0 \Rightarrow \bar{\mathbf{p}} = \bar{\mathbf{p}}_x$, and thus $\overline{\gamma\beta} = \frac{\bar{\mathbf{p}}_x}{m_0 c}$.

2.3 Plasma Particle-in-cell Simulation

Due to the lack of an analytic theory capable of describing full 3D non-linear LWFA, laser-plasma simulations have been essential for progress in the field [10, 15]. Since plasma dynamics are well-described through the theory of electromagnetism, most numerical codes used in the simulation of LWFA implement the particle-in-cell (PIC) method [53], moving particles according to the Lorentz force, and calculating fields using Maxwell's equations [54]. Whilst there are a multitude of PIC codes widely used in the simulation of LWFA, we shall here focus specifically on the open-source code EPOCH.

2.3.1 Fundamental Theory

2.3.1.1 Pseudo-particles and Weighting Functions

Due to the extreme computational expense associated with simulating all particles that exist even within the small region of interest (10^{18} electrons per cm^3), the concept of "pseudo-particles" (also referred to as "macro-particles") was introduced. Corresponding to many real electrons, each pseudo-particle describes the expected collective motion of these localised groups.

For a pseudo-particle representing k electrons, this particle would also be assigned a mass of $m_k = \sum_i m_i = km_e$ and a "particle weighting" of $w = k$; the particle weighting here describes how many real electrons the pseudo-particle represents. Since the Lorentz force is dependent only on the charge-to-mass ratio, pseudo-particles follow the same trajectory as individual electrons with equivalent position and momentum [55].

These pseudo-particles are used to populate a simulation grid representing the physical space to be simulated, with the dimensions of the grid cells being depen-

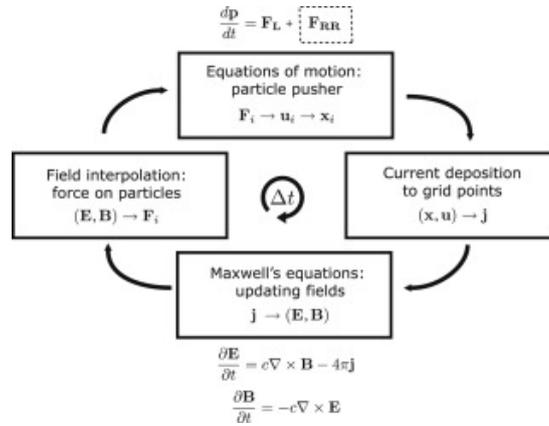


Figure 2.5: Cycle by which the PIC scheme operates. **(top)**: charge and current density at the gridpoints is calculated from the current particle positions and velocities. **(right)**: weighting is applied to closest gridpoints to the pseudo-particles, by proximity to grid-point. **(bottom)**: Maxwell equations evaluated to calculate electric and magnetic fields at gridpoints. **(left)**: Weighting applied to particles, computing the force on particles and updating their position and velocity [56, 57]. Reprinted from *Comp. Phys. Comm.*, **204**, M. Vranic *et al*, *Classical radiation reaction in particle-in-cell simulations*, Copyright (2016), with permission from Elsevier [58].

dent on the phenomena being investigated [12]. For LWFA, sufficient resolution in the laser pulse is required in order to model the plasma response [57]; whilst ~ 10 cells per laser wavelength in the direction of propagation are generally sufficient to see injection, the accuracy can be significantly improved by increasing this resolution, and many simulations use of the order ~ 30 – 100 cells per wavelength [4, 9, 43]. Figure 2.5 is a flowchart depicting the overall method used in EPOCH.

2.3.1.2 Particle Pusher

The particle pusher used in EPOCH is a Birdsall-Langdon type PIC scheme [59], and is responsible for integrating the equations of motion for the pseudo-particles. With each pseudo-particle i having an assigned position \mathbf{x}_i and velocity \mathbf{v}_i , ideally one would track previous time steps to improve accuracy in the time integration

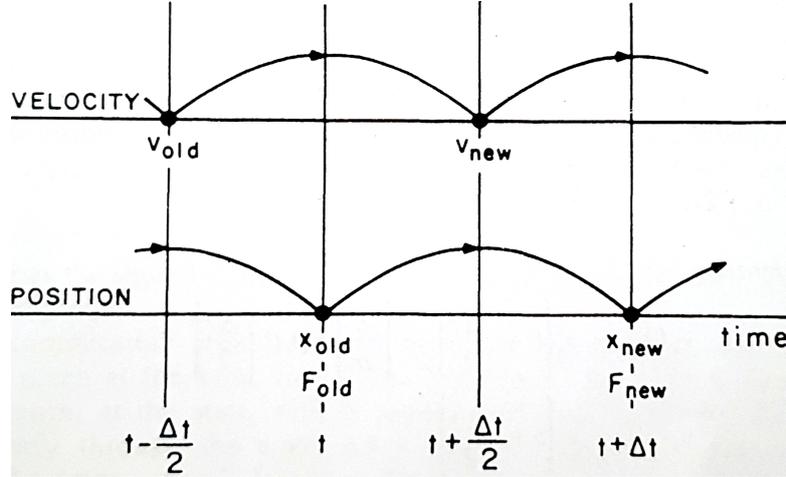


Figure 2.6: Sketch of the leap-frog FDM integration method, depicting the staggering of the two calculations. Even though x_t and v_t are not known at the same time, they are advanced to $x_{t+\Delta t}$ and $v_{t+\Delta t}$. Reprinted by permission from Taylor and Francis: *Plasma Physics via Computer Simulation*, C. K. Birdsall and A. B. Langdon, Copyright Taylor & Francis Group LLC (2004) [57].

[57]. However, this is impractical due to the large number of particles simulated in 2D and 3D, and thus a trade-off in memory, computational expense and accuracy is made.

We solve this by combining the "leap-frog" method, in which the first-order force and velocity equations are integrated separately, with the "finite-difference" method (FDM), giving the following equations of motion [60] for each particle (Figure 2.6):

$$\frac{\mathbf{x}^{n+\frac{3}{2}} - \mathbf{x}^{n+\frac{1}{2}}}{\Delta t} = \mathbf{v}^n \quad (2.18)$$

$$\frac{\mathbf{v}^{n+1} - \mathbf{v}^n}{\Delta t} = \frac{e}{m} \left(\mathbf{E}^{n+\frac{1}{2}} + \frac{\mathbf{v}^{n+1} + \mathbf{v}^n}{2} \times \mathbf{B}^{n+\frac{1}{2}} \right) = \frac{\mathbf{F}}{m} \quad (2.19)$$

Although this method inherently has some error, with appropriate simulation parameters this method can be surprisingly accurate, and offers a good compromise in accuracy, speed and number of numerical operations [57]. Like most PIC codes,

EPOCH uses the Boris rotation algorithm, which separates the force calculation 2.19 into several components [61]:

$$\mathbf{v}^- = \mathbf{v}^n + \frac{e}{2m} \Delta t \mathbf{E}^{n+\frac{1}{2}} \quad (2.20)$$

$$\mathbf{v}^+ = \mathbf{v}^- + \frac{e}{2m} \Delta t (\mathbf{v}^+ + \mathbf{v}^-) \times \mathbf{B}^{n+\frac{1}{2}} \quad (2.21)$$

$$\mathbf{v}^{n+1} = \mathbf{v}^+ + \frac{e}{2m} \Delta t \mathbf{E}^{n+\frac{1}{2}} \quad (2.22)$$

This can be understood as splitting the acceleration due to the \mathbf{E} -field into two half-accelerations (2.20, 2.22), in between which the velocity vector rotation due to the magnetic field is calculated (2.21). Equation (2.21) can also be transformed to be time-explicit [61]:

$$\mathbf{v}^+ = \mathbf{v}^- + (\mathbf{v}^- + \mathbf{v}^- \times \mathbf{t}) \times \mathbf{s} \quad (2.23)$$

where we have defined $\mathbf{t} = \frac{\Omega \Delta t}{2} \hat{\mathbf{b}}$, $\mathbf{s} = \frac{\Omega \Delta t}{1 + (\Omega \Delta t / 2)^2} \hat{\mathbf{b}}$, unit vector $\hat{\mathbf{b}} = \frac{\mathbf{B}}{B}$ and gyro-frequency $\Omega = \frac{eB}{m}$.

2.3.1.3 Field Solver

The field solver is responsible for solving the Maxwell equations for each cell in the simulation grid. EPOCH uses a Yee staggered second-order finite-difference time domain (FDTD) method [59]. The use of a Yee cell [62] improves the accuracy of the field solver, by calculating each field component at the appropriate cell boundary (see Figure 2.7). The appropriate time-centered equations to evaluate are, for electric and magnetic fields \mathbf{E} , \mathbf{B} and current \mathbf{j} are:

$$\mathbf{E}^{n+\frac{1}{2}} = \mathbf{E}^n + \frac{\Delta t}{2} (c^2 \nabla \times \mathbf{B}^n - \mathbf{j}^n) \quad (2.24)$$

$$\mathbf{B}^{n+\frac{1}{2}} = \mathbf{B}^n - \frac{\Delta t}{2} (\nabla \times \mathbf{E}^{n+\frac{1}{2}}) \quad (2.25)$$

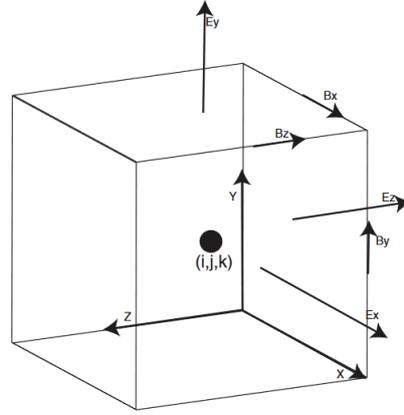


Figure 2.7: A single Yee grid cell, in 2D. Starting from the centre of the cell, the point where E-field components are evaluated is shifted half a cell in the direction the field points, if possible. The B-field components are moved half a grid in all directions other than the direction it points. Reprinted from IOP Publishing: Plasma Phys. Control. Fusion, *Contemporary particle-in-cell approach to laser-plasma modelling*, T. D. Arber *et al*, under Creative Commons 3.0 Attribution (CC BY) licence [59].

2.3.1.4 Current Calculation

In order to better preserve the charge on the grid, rather than only globally, EPOCH implements the Villasenor and Buneman scheme, which solves an additional equation at each time step [59]. Specifically, the continuity equation $\partial\rho/\partial t = \nabla \cdot \mathbf{J}$ is solved [55], with the particle pusher evaluating the \mathbf{j}^{n+1} currents:

$$\mathbf{B}^{n+1} = \mathbf{B}^{n+\frac{1}{2}} - \frac{\Delta t}{2} (\nabla \times \mathbf{E}^{n+\frac{1}{2}}) \quad (2.26)$$

$$\mathbf{E}^{n+1} = \mathbf{E}^{n+\frac{1}{2}} + \frac{\Delta t}{2} (c^2 \nabla \times \mathbf{B}^{n+1} - \mathbf{j}^{n+1}) \quad (2.27)$$

2.3.1.5 Stability Conditions

Of course, the stability of the simulation is dependent on the appropriate choice of various simulation parameters – the grid resolution must be sufficiently high to accurately describe the rapidly varying laser field. Additionally, simulating too

few pseudo-particles can reduce the resolution in the motion of particles to a point where injection may no longer be observed. Whilst one would also typically have to worry especially about selecting an appropriate time step, in order to minimise error in the particle pusher and field calculations, thereby better conserving energy, EPOCH automatically determines the optimal time step Δt in accordance with the Courant condition [59].

Tsung *et al.* [63] showed in 2006 that in addition to a suitable choice of time step, the cell size must be carefully chosen to ensure good numerical dispersion. A transverse cell size $\Delta y \sim \frac{0.2c}{\omega_p}$, where ω_p is the plasma frequency, provides sufficient resolution to resolve the laser evolution, for underdense plasmas with $\omega_p \ll \omega_L$ and $\Delta x \ll \Delta y$. It is further shown that with a suitable time step choice, it is required that $k_L \Delta x = \frac{2\pi}{\lambda_L} \Delta x < 0.2$ is satisfied in order to maintain a relative error in the group velocity of less than 1% [63]. This means that with approximately 30 cells per laser wavelength in the longitudinal direction, the errors in group velocity (and thus electron energies) begin to converge.

2.3.2 Increasing Simulation Speed

With the significant computational expense involved with high-resolution LWFA simulations – even in only 2 dimensions – it is highly desirable to develop schemes to reduce this. Of course, there are many different possible approaches one might take, and coupled with varying sets of assumptions, result in varying success depending on simulated system.

2.3.2.1 Existing Schemes

To simulate LWFA in conventional PIC codes, it is essential to sufficiently resolve the laser wavelength ($\lambda \sim 800\text{nm}$); comparatively, the generated wakefields are ~ 2 orders of magnitude larger, with total simulation lengths typically being in

the order of centimetres. In an attempt to alleviate this disparity, some simulation software introduce models such as the guiding centre approximation – this averages over the laser frequency, thereby removing the need to resolve the laser wavelength, and instead introduces an envelope equation, adding corresponding terms for the laser and ponderomotive force to the particle pusher [64]. However, this model has difficulty accurately simulating full pump depletion distance, and self-injection [65].

Another model, known as the Lorentz boosted frame technique, was initially proposed in 1992 by Mori *et al.* [66], although not selected for funding; it was independently re-discovered by Vay [67] in 2007. Fundamentally, the approach exploits the fact that in a Lorentz boosted frame, the number of cells in the propagation direction is invariant, whilst cell size – and thus time step – are Lorentz expanded [65]. This combination of the increase in simulation time step, coupled with the spatial reduction, results in a significant reduction in simulation time [68]. However, this approach introduces an instability in multi-dimensional simulations [69], which impacts the accuracy in simulations of self-injection [65].

2.3.2.2 Particle Reduction

In addition to the variety of schemes which offer increases in speed by exploring different physical models, another potential approach involves algorithmically reducing the number of particles simulated mid-simulation. Multiple algorithms have been proposed that describe the method for reducing the total number of simulated macro-particles, through coalescing, in an attempt to increase the simulation speed whilst preserving the accuracy associated with the initial higher resolution [70–73]. Applying this to LWFA, the highest resolution is required in the region where injection occurs – whether stimulated or self-injected – whereas the post-injection acceleration phase, for example, converges at lower resolutions.

Whilst there are various different algorithms, they generally merge 2 particles into 1 (or 4-2) conserving charge and mass, as well as total energy or total momentum.

3 | Simulation Reduction

3.1 Nearest-Neighbour Reduction Method

Whilst it is important to attempt to conserve the total energy (and momentum) in a simulation, for LWFA it is also important to preserve the energy- and momentum distributions. Due to the large field gradients that are key to laser plasma accelerators, and the resulting strong relationship between spatial position in the wake with particle momentum, one must be careful averaging over momentum and position when coalescing particles. We thus propose a method¹, which attempts to preserve the kinetic energy spectrum of simulated particles, by redistributing particle weightings of those particles removed by the algorithm, without averaging over position or momentum.

3.1.1 Moving window

The concept of a moving window is frequently used in PIC simulations, since this restricts the simulation to the current region of interest, rather than simulating the full domain continuously. In LWFA, the (fixed size) window moves with the laser; this is implemented through the deletion of cells that are now outside the domain of the new window, and spawning new cells correspondingly (Figure 3.1). It is important to note that since pseudo-particles are free to move between cells after they have been initialised, the number of pseudo-particles in a particular cell $n_{i,j}$ is only guaranteed to be identical to the specified particle-per-cell count n_{ppc} at initialisation (Figure 3.2).

¹Source code for the nearest-neighbour algorithmic reduction is available from <https://bitbucket.org/JFGrimm/epoch-simulation-reduction/>. Written in MATLAB, it includes further options beyond what is discussed here, such as non-uniform sampling, and can reduce 2 or 3 dimensional simulations.

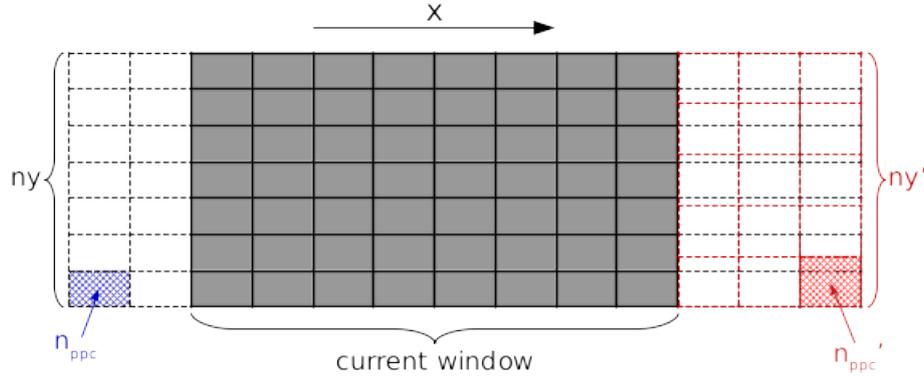


Figure 3.1: Example simulation window with grid size (n_x, n_y) , at some arbitrary time post-injection. As the window moves, simulation cells are deleted from the back (left) and added to the front (right), with an initial n_{ppc} particles-per-cell. If we reduce, this thus involves a change in either transverse resolution (n'_y), particles-per-cell (n'_{ppc}), or both. This has been represented in red.

To consistently reduce the number of particles being simulated after a reduction algorithm has been applied, we can consider three options: reduce the particle-per-cell count n_{ppc} , reduce the number of cells in the transverse direction n_y , or a combination of these:

- if $n'_y = n_y$ and $n'_{ppc} = n_{ppc}$: no reduction
- if $n'_y = n_y$ and $n'_{ppc} < n_{ppc}$: reduction in PPC only
- if $n'_y < n_y$ and $n'_{ppc} \leq n_{ppc}$: reduction in both field resolution and PPC

Reducing in n_y has the added speed-up of decreasing the field resolution, whereas a reduction only in PPC preserves this. Whilst one could also conceivably reduce only the field resolution whilst maintaining the same number of pseudo-particles throughout, by increasing the PPC by a factor of (n_y/n'_y) , we shall not consider increases in PPC henceforth.

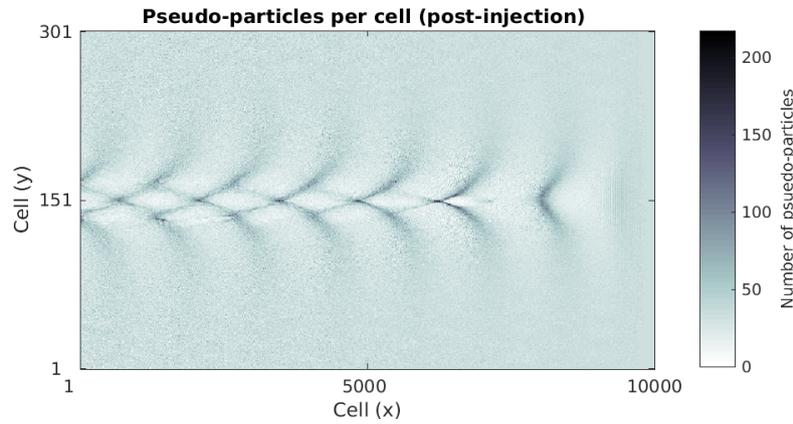


Figure 3.2: Total number of pseudo-particles in each simulation cell of a 32 PPC, 301 transverse cell simulation at some time after injection. It is clear that the actual number of pseudo-particles in each cell can vary substantially from the initial number (32), corresponding to the position in the wake.

3.1.2 Transverse Cell Reduction

To best describe the process of the proposed reduction algorithm, let us first consider a single cell in a higher resolution simulation, some arbitrary time after injection (Figure 3.3). Before applying a nearest-neighbour (NN) algorithm on the

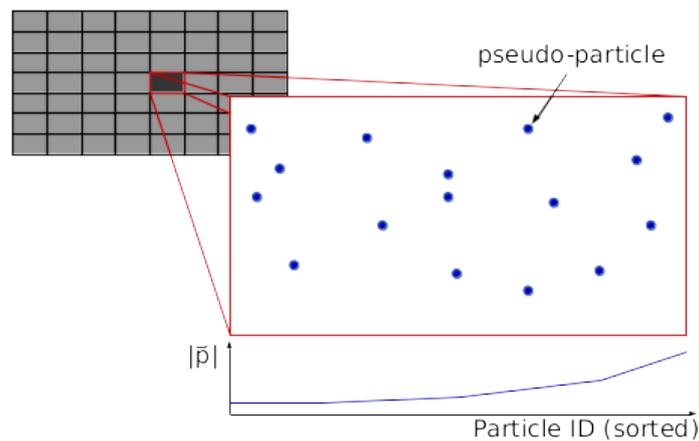


Figure 3.3: Sample simulation cell at some arbitrary time after injection, showing the distribution of pseudo-particles (blue) and the corresponding momentum distribution.

single cell, we first need to select which particles are kept and which are removed from the simulation; there are several possible choices in how to sample the particles: at random, uniformly from the pseudo-particle momentum distribution, non-uniformly from the momentum distribution, or some other structured method.

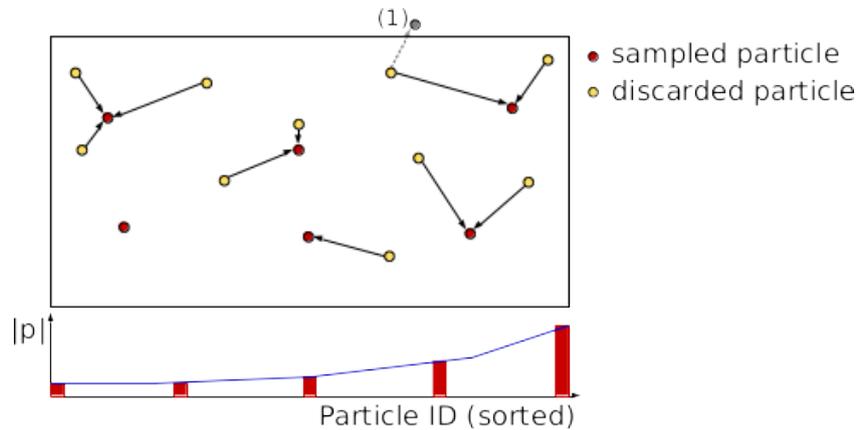


Figure 3.4: Nearest-neighbour reduction on a single cell, sampling particles uniformly from the momentum distribution, and adding weights of removed particles to their closest (spatially) neighbouring sampled particle. (1) depicts a boundary problem, where the closest sampled particle may lie in an adjacent cell.

Assuming we now sample uniformly from the momentum distribution, we use an NN algorithm to determine the closest sampled particle (spatially) for each discarded particle, and sum the particle weighting functions (Figure 3.4). As is evident in (Figure 3.4), this method has several issues; principally, there is a boundary problem where the nearest sampled particle may actually lie in an adjacent cell, which is not considered currently. Additionally, whilst it conserves the momentum-distribution locally, this may not translate to a global conservation. A further difficulty arises when considering cells which contain significantly fewer pseudo-particles than the mean, raising questions in how many particles to sample or discard.

One simple solution which addresses the issues discussed involves operating on larger groups of cells, henceforth referred to as "super-cells", thereby minimising

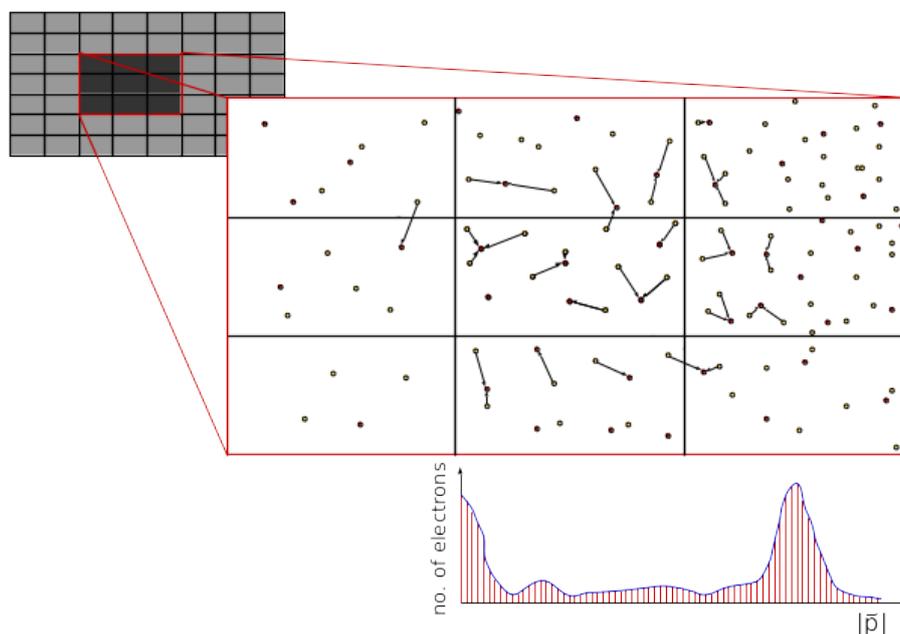


Figure 3.5: Nearest-neighbour reduction on a group of cells ("super-cell"), sampling particles uniformly from the momentum distribution, and adding weights of removed particles to their closest (spatially) neighbouring sampled particle.

these problems (Figure 3.5). Whilst operating on the entire simulation window would likely be ideal, it requires a very robust NN algorithm with high memory capacity even in 2D, hence the division into super-cells. It also allows further control over the comparative preservation of the local momentum distribution relative to the global distribution – one would expect that smaller super-cells are likely to better preserve the local distribution.

3.2 Parity Considerations

Due to the extreme field gradients which are key to electron injection, it is imperative to sample these appropriately. Figure 3.6 depicts the E_x field strength at the x -position of the principal injected electron bunch as a function of y , of a simulation with an on-axis beam propagating in x . In order to reduce the number

of transverse cells n_y , we must thus sample the y domain uniformly. When we sample with an odd parity, we see that the on-axis dip in field strength, about the position of the injected electrons, is exactly sampled. However, with even parity of similar resolution, we now significantly over-estimate the on-axis field strength. In preliminary simulations, this overestimate caused the generated electron beam to disappear completely.

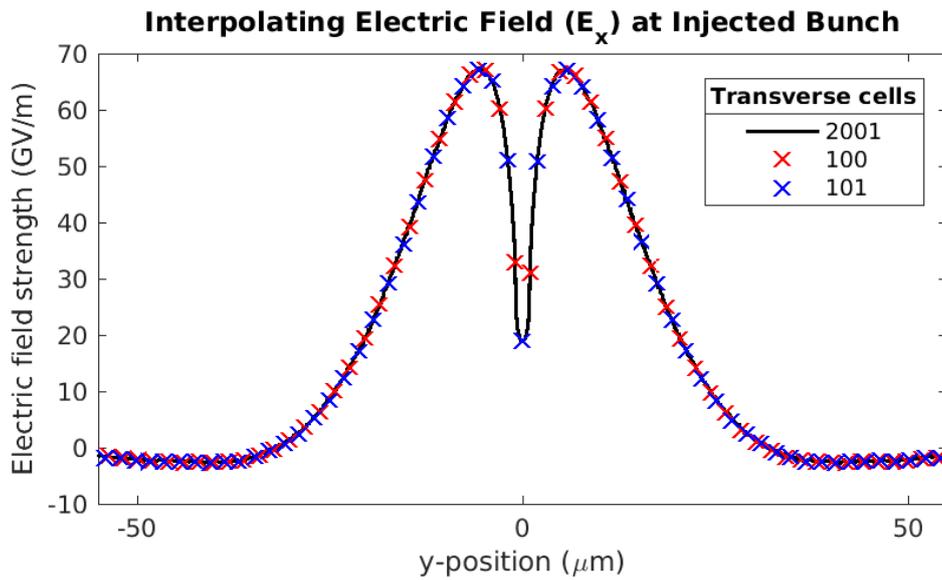


Figure 3.6: Interpolation of the E_x field component at fixed x -position, corresponding to the location of an injected electron bunch. Due to the rapidly varying fields, the on-axis field strength is significantly overestimated when interpolating with even parity, which samples either side, whilst it is exactly sampled with odd parity.

3.3 Variable PPC

An alternative method to reduce the number of simulated particles mid-simulation involves changing the number of particles-per-cell without any NN algorithmic reduction. Whilst most PIC codes, such as EPOCH, do not allow a spatially- or temporally defined variation in PPC, it is possible to effectively restart a simula-

tion with fewer particles-per-cell, thereby corresponding to a step-function profile.

In EPOCH, this can be achieved through the following method:

1. Using your preferred language, load the SDF file corresponding to the timestep in the simulation you wish to reduce.
2. Extract the following data, and write these as binary data files (type `double`): weight w , positions $\{x, y\}$, momenta $P\{x, y, z\}$, electric fields $E\{x, y, z\}$, magnetic fields $B\{x, y, z\}$, x-grid gx , and time t .
3. Determine the left boundary position of the window $x_{\text{left}} = \min(gx)$
4. Make the following modifications to the input deck, based on the original simulation:
 - Offset all time-based parameters by $-t$, and the x -domain by $+x_{\text{left}}$.
 - Change the moving window start time to 0.
 - Delete the laser control block.
 - Change the PPC to the new desired value n'_{ppc} .
 - Add a `particles_from_file` block to load particle data from step 2.
 - Similarly, add a `fields` block to load the field data from step 2.
5. Run the modified input deck

This approach will initialise new cells with a lower PPC when the modified simulation is run, before replacing existing particles with those loaded from file – therefore, the lower PPC effectively is only defined for future cells added to the front of the moving window. As the simulation window moves, pseudo-particles gradually despawn as they leave the simulation window; however, since the in-

jected electrons are travelling with relativistic speed, they continue to move with the window and do not despawn. This therefore preserves exactly the resolution in the injected electrons, whilst gradually decreasing the number of non-injected particles simulated until the window has moved beyond the domain at which the PPC was reduced, resulting in a corresponding speedup in computation time. This approach may also be combined with field interpolation to reduce the grid resolution – thus further decreasing the number of simulated non-injected particles in future simulation cells.

4 | Methodology

4.1 Aim and Objectives

Using the open-source laser-plasma simulation code EPOCH, we aim to quantify the effect of running narrow width simulations (few cells in transverse direction). Using this as a basis, we will analyse the suitability of various reduction schemes – implementing nearest-neighbour algorithms, field reduction, and variable particles-per-cell – with the aim to maintain high detail in the beam characteristics after reduction, whilst decreasing the simulation time. In order to ensure results are self-consistent and representative, we also test to ensure convergence in the initial simulation parameters - specifically, the number of particles-per-cell, and cells per laser wavelength (in the longitudinal direction).

4.2 Simulation Parameters

To maintain consistency across the various simulations, we hold all but the key parameters of interest constant. Whilst there are many schemes to generate quasi-monoenergetic electron beams in LWFA [9, 10, 26, 52], we here use shock-injection. The density profile (Figure 4.1) is a modified version of that used by Swanson *et al.* [9], with some minor modifications; instead of an effective transition width of $100\mu\text{m}$, this was decreased to $25\mu\text{m}$, with the aim of injecting a greater charge. Additionally, the post-step density was held constant to ensure a narrower peak in the energy spectrum, as demonstrated by Baird [12].

Further key simulation parameters are as follows:

- Laser wavelength: $\lambda_L = 0.805\mu\text{m}$

- Laser beam waist: $w(\text{FWHM}) = 18\mu\text{m}$
- Laser pulse duration: $\tau_L(\text{FWHM}) = 47\text{fs}$
- Laser intensity: $I_L = 1.5 \times 10^{19}\text{W}/\text{cm}^2$
- Laser polarization: $\theta_L = 90^\circ$ (out-of-plane)
- Window domain: $\{l_x, l_y\} = \{100\lambda_L, 120\lambda_L\}$
- Number of cells: $\{n_x, n_y\} = \{10000, 301\}$
- Particles-per-cell: $n_{\text{ppc}} = 32$
- End time: $t_{\text{end}} = 10.6\text{ps}$

All simulations will be executed in 2D; whilst comparative data for 3D simulations would be useful, these are significantly more computationally expensive. For these simulations, ions are also not simulated, instead being treated as a neutralizing background field; this is possible due to the large inertia of the ions, with respect to electrons, resulting in negligible contributions over the short time scales simulated.

4.3 Convergence Tests

In order to ensure that the base simulation parameters, around which the reduction and transverse cell simulations are based, are self-consistent, we need to test for convergence in two key parameters: particles-per-cell (n_{ppc}) and cells-per-wavelength (CPW). For the former, we hold all simulation parameters constant as listed in Chapter 4.2, apart from the number of particles-per-cell, which are varied in accordance with Table (4.1).

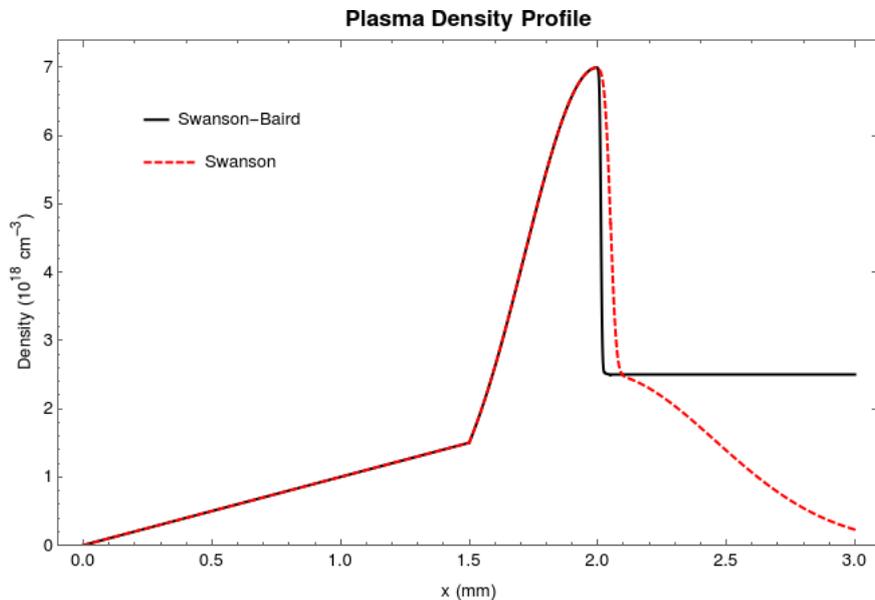


Figure 4.1: The plasma density profile used for simulations (black), which is based on the profile (red, dashed) simulated by Swanson *et al.* [9], but with some modifications. Instead of an effective step width of $100\mu\text{m}$, a width of $25\mu\text{m}$ was used, with a flat post-step density as suggested by Baird [12].

4.4 Transverse Simulation Cells

In order to better understand potential effects of the different reduction schemes, it is important to have a baseline to which these can be compared. By holding the particles-per-cell and simulation domain constant as described in Chapter 4.2, we now vary the number of transverse cells spanning the y-domain (noting that overall window dimensions are constant). Preliminary simulations showed large statistical variance in simulation results at low transverse cell counts, and we thus average over three simulations with identical parameters, but different random seeds (Table 4.2).

The combination of varying number of transverse cells, whilst leaving the overall window dimensions unchanged, results in correspondingly fewer macro-particles being simulated, and a decrease in the electric- and magnetic field resolution (in

No.	Convergence Test Parameters		
	n_{ppc}	Window length	CPW
1	1	100 λ_L	100
2	2	100 λ_L	100
3	4	100 λ_L	100
4	8	100 λ_L	100
5	16	100 λ_L	100
6	32	100 λ_L	100
7	64	100 λ_L	100
8	128	100 λ_L	100
9	32	1,000 λ_L	10
10	32	400 λ_L	25
11	32	200 λ_L	50
12	32	150 λ_L	66.6
13	32	100 λ_L	100
14	32	50 λ_L	200

Table 4.1: Simulation parameters to test convergence in particles-per-cell (n_{ppc}) and cells-per-wavelength (CPW), by varying n_{ppc} and the length of the moving window (l_x), respectively. For both, the number of longitudinal and transverse cells remains constant in accordance with the base parameters laid out in Chapter 4.2. Simulation 13 is identical to 6, corresponding to the base parameters, and is included for completeness although the same results will be used for both.

the transverse direction). We are also careful to select only odd-parity transverse cell numbers, since although simulations initiated with even parity successfully simulate injection, any simulations in which fields have been interpolated require odd parity (see Chapter 3.2). In order to be able to fully compare simulations, we thus opt to only simulate with odd parity.

4.5 Reduction Schemes

As discussed in Chapters 2.3.2 and 3, there are many different approaches one might take when attempting to reduce the number of particles and field calculations in order to increase simulation speed, whilst attempting to preserve accuracy.

No.	Transverse cells (ny)	Window dimensions	Cell width (transverse)
1 – 3	5	(100 λ_L , 120 λ_L)	24.00 λ_L
4 – 6	11		10.91 λ_L
7 – 9	31		3.87 λ_L
10 – 12	61		1.97 λ_L
13 – 15	101		1.19 λ_L
16 – 18	301		0.40 λ_L
19 – 21	601		0.20 λ_L
22 – 24	1001		0.12 λ_L
25 – 26	2001		0.06 λ_L

Table 4.2: Simulation parameters describing the variation in transverse cells, with corresponding cell widths (size in transverse direction), whilst the absolute window dimension remains constant. Runs are grouped since each parameter set is repeated a total of 3 times, each with different random seeds.

Although we shall primarily focus on nearest-neighbour algorithmic reduction (without any averaging in particle momenta or positions) and simulations with variable particles-per-cell, it is important to compare the effects these additional methods have. We shall also examine the effects of different sizes of super-cells (see Chapter 3.1.2), when reducing to different widths. In order to maintain consistency between different reduction schemes, and to avoid difficulties involved with reducing about the transition, all simulation reduction will occur some time after injection, at $t = 8\text{ps}$, and will be based on a simulation matching the parameters listed in Chapter 4.2, but with 2001 transverse cells.

4.5.1 Nearest-Neighbour Particle Averaging

In order to compare the effects of particle attribute (momentum, position) averaging, we here hold constant the following reduction parameters:

```
sampleMethod = 'p_dist'
reduceMethod = 'nearest'
fieldReduce = 'linear'
```

```
superCell = [50, 20]
```

Where `sampleMethod` corresponds to the sampling method for choosing which particles to keep and which to discard, and has here been set to use a structured uniform sampling of the super-cell momentum distribution, using nearest-neighbour algorithmic reduction as specified by `reduceMethod`. Field interpolation is fixed as linear interpolation, as defined by `fieldReduce`. `superCell` defines the number of super-cells that are created in (x, y) - thus corresponding to a grouping clusters of 200 and 100 cells in x and y , respectively.

No.	Attribute Averaging	ny'	n'_{ppc}
1	none	101	32
2	momentum	101	32
3	position	101	32
4	momentum, position	101	32
5	none	2001	1.62

Table 4.3: Simulation parameters for comparing the effects of different particle attribute averaging methods when using nearest-neighbour reduction, where ny' is the reduced number of transverse cells, and n'_{ppc} the reduced PPC. Simulation 5 corresponds to a reduction in PPC only, of equivalent factor - since this is non-integer, this is rounded to 2 in the restarted simulation.

4.5.2 Super-cell size

To compare the nuances of different super-cell sizes, we again fix the reduction parameters as in Chapter 4.5.1, but now vary the `superCell` parameter instead of `averageCombined`, which we leave empty (no averaging):

```
sampleMethod = 'p_dist'
reduceMethod = 'nearest'
fieldReduce = 'linear'
```

```
averageCombined = {}
```

We now look at two different categories of super-cell - one "large" and one "small", and reduce to varying numbers of transverse cells for each. Henceforth, "large" super-cells will refer to 10 super-cell divisions in the x-direction, and 5 in the y-direction, whilst "small" super-cells refer to 100 and 50 divisions respectively. Whilst ideally we would include a simulation with only one super-cell (full frame), this was unfortunately not possible due to memory limitations in the reduction algorithm implementation. Table 4.4 shows the different parameters varied for the "large" and "small" super-cell reductions, as well as a count of the number of original simulation cells that are grouped into a super-cell.

No.	superCell	ny'	Cells per super-cell
1	[10, 5]	5	400,000
2	[10, 5]	11	400,000
3	[10, 5]	31	400,000
4	[10, 5]	61	400,000
5	[10, 5]	101	400,000
6	[10, 5]	301	400,000
7	[100, 50]	5	4,000
8	[100, 50]	11	4,000
9	[100, 50]	31	4,000
10	[100, 50]	61	4,000
11	[100, 50]	101	4,000
12	[100, 50]	301	4,000

Table 4.4: Simulation parameters for comparing reduction to varying numbers of transverse cells ny' , for different "super-cell" resolutions. Simulations 1 – 6 here correspond to "large" super-cells, where each super-cell comprises 400,000 original simulation cells, whilst 7 – 12 correspond to "small" super-cells, being a factor 100 smaller.

4.5.3 Variable PPC

In addition to the nearest-neighbour reduction algorithms, a further scheme to decrease computation time post-injection involves decreasing the number of particles-per-cell. As with the NN reduction, we again make this reduction based on the simulation with 2001 transverse cells and 32 PPC, at a time of 8 picoseconds. In addition to decreasing to various numbers of particles-per-cell, we shall also consider the additional impact of linear field interpolation in conjunction with this method, thereby further reducing the particles simulated as the window moves (see Chapter 3.3). The parameters for this set of simulations are described in Table 4.5, with all other parameters being held constant as defined in Chapter 4.2.

No.	Particles-per-cell	Transverse cells	
		pre-reduction	post-reduction
1	1	2001	2001
2	2	2001	2001
3	4	2001	2001
4	1	2001	31
5	1	2001	101
6	1	2001	301
7	1	301	301

Table 4.5: Simulation parameters for variable PPC simulations; simulations 1 – 3 correspond to reducing to different numbers of particles-per-cell, whilst 4 – 6 reduce to 1 PPC with linear field interpolation. 7 is an additional run without field interpolation, but is based on a 301 cell simulation instead of 2001.

5 | Results

5.1 Convergence

As discussed in Chapter 4.3, we must first ensure that the simulations converge in both particles-per-cell, and cells-per-wavelength in order to enable a meaningful comparison between simulations.

5.1.1 Particles-per-cell

Holding all parameters other than the PPC constant, as described in Table 4.1, the number of particles-per-cell was varied between 1 and 128 in powers of two. Figure 5.1 depicts the position of the beam peak in energy-space, E_{peak} , as a function of PPC, as well as the corresponding upper and lower FWHM energies.

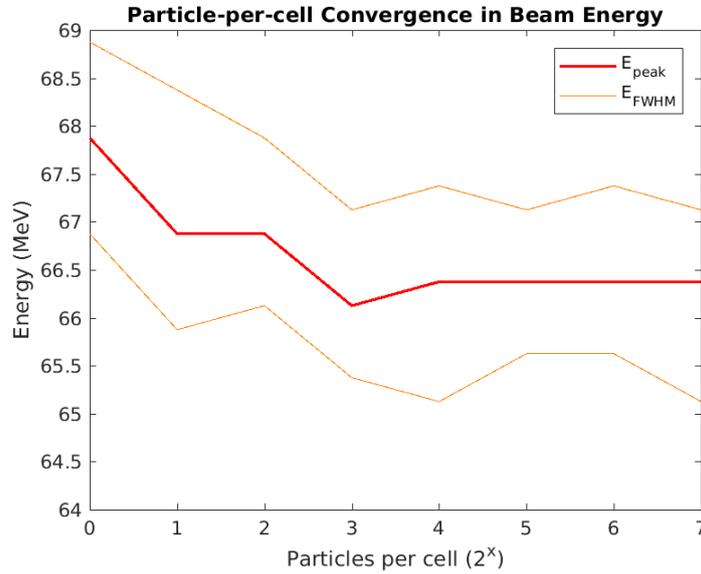


Figure 5.1: The E_{peak} and E_{FWHM} dependence on the number of pseudo-particles per cell. The upper and lower boundaries of the FWHM energy are here plotted in (yellow), about the peak (red).

It is evident that whilst the simulations show relatively little variation even at low PPC, the peak energy converges at 16 PPC, although the FWHM continues to undergo small fluctuations. This is due to small statistical variances in between the different simulations; since the fluctuations are small about an approximately constant value, and the peak energy is exactly constant, this suggests that the selected base simulation value of 32 PPC is well-converged in energy.

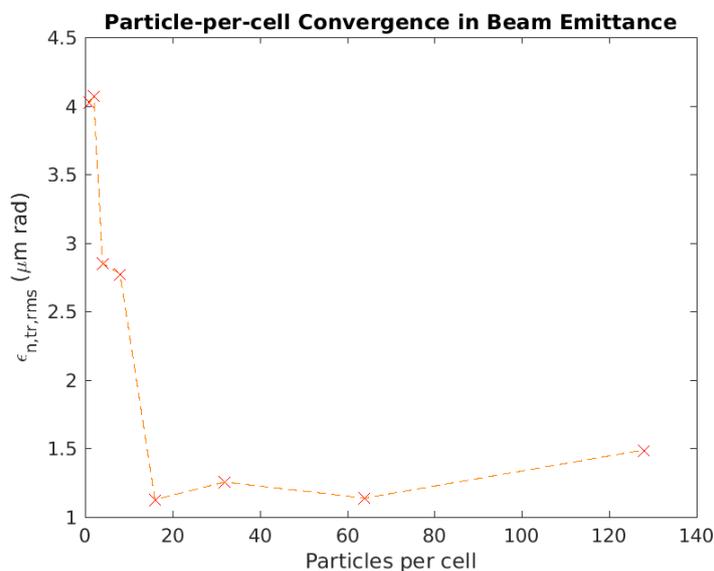


Figure 5.2: The dependence of beam emittance ($\epsilon_{n, tr, rms}$) on the number of pseudo-particles per cell. Although the emittance appears to converge from 16 PPC, it increases again at 128 PPC, although this could be due to statistical variations.

In contrast, the convergence in the normalised trace-space beam emittance is much more drastic, as shown in Figure 5.2. For simulations below 16 PPC, the error relative to higher PPC simulations is $\sim (300 - 400)\%$; beyond this it is approximately constant with minor variations, although 128 PPC suggests a somewhat large beam emittance. Whilst not perfectly converged, 32 particles-per-cell again appears to be sufficiently converged to be well-behaved, whilst requiring significantly less computation time than higher values which provide little improvement.

5.1.2 Cells-per-wavelength

If we now hold all other parameters constant whilst varying cells-per-wavelength (CPW) in the longitudinal direction, as described in Table 4.1, we see that unlike the E_{peak} convergence in PPC, there is significantly more variation in the peak energy as a function of the varied parameter (Figure 5.3). The peak energy now increases non-linearly with the number of cells-per-wavelength, although the absolute width of the FWHM about the peak, E_{FWHM} , remains fairly constant, with minor fluctuations. As discussed in section 2.3.1.5, we notice a large disparity in the beam energy when using fewer than ~ 30 CPW, as a result of numerical dispersion. As expected, we see the beam energy gradually converge as we increase the number of cells per wavelength beyond this.

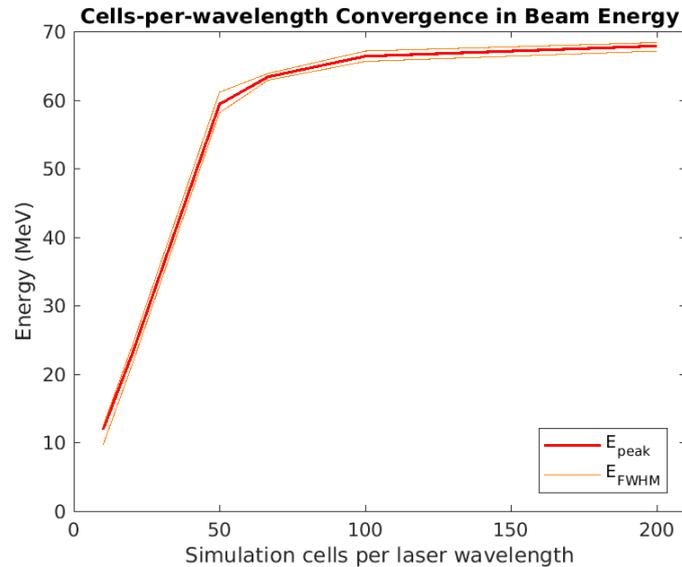


Figure 5.3: The E_{peak} and E_{FWHM} dependence on the number of simulation cells (in the longitudinal direction) per laser wavelength.

The emittance of the generated beam decreases as the CPW is increased, with an order of magnitude difference in emittance between 200 CPW and below 100 CPW (Figure 5.4). Whilst it is clear for both peak energy and emittance that the

simulation is not fully converged, the resultant error when simulating with 100 CPW at the given parameters is sufficiently small that it is a good compromise between computation time and accuracy. This is especially true since the absolute accuracy of the simulated beam compared to 3D simulations or experimental data is not vital; rather, it is the resultant change to the beam as a result of reduction methods that is the focus. As such we must ensure that statistical aberrations are sufficiently small that they do not obscure this comparison, and that simulations are reasonably representative, which the combination of 32 PPC and 100 CPW achieves.

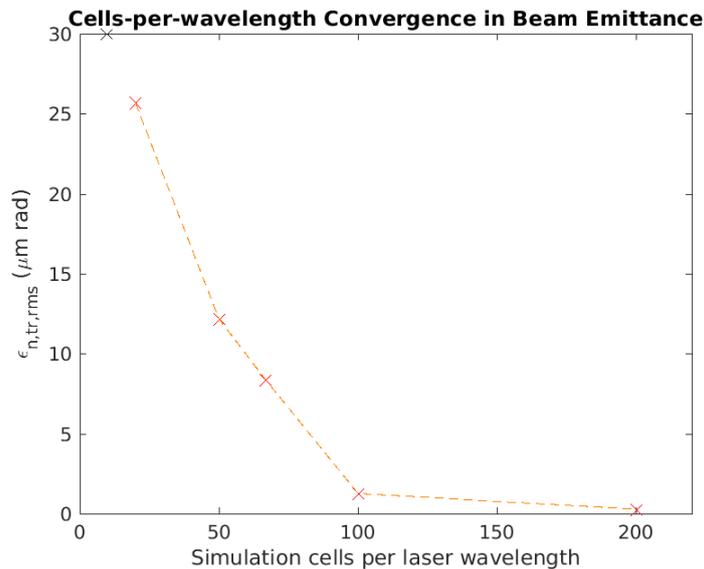


Figure 5.4: The dependence of beam emittance ($\epsilon_{n, tr, rms}$) on the number of simulation cells (in the longitudinal direction) per laser wavelength. The datapoint for 10 cells per wavelength has been distinguished since the beam emittance was in calculable (NaN).

5.2 Transverse Simulation Cells

To establish a baseline to which reduced simulations can be compared, it is important to run simulations with small numbers of transverse cells throughout the

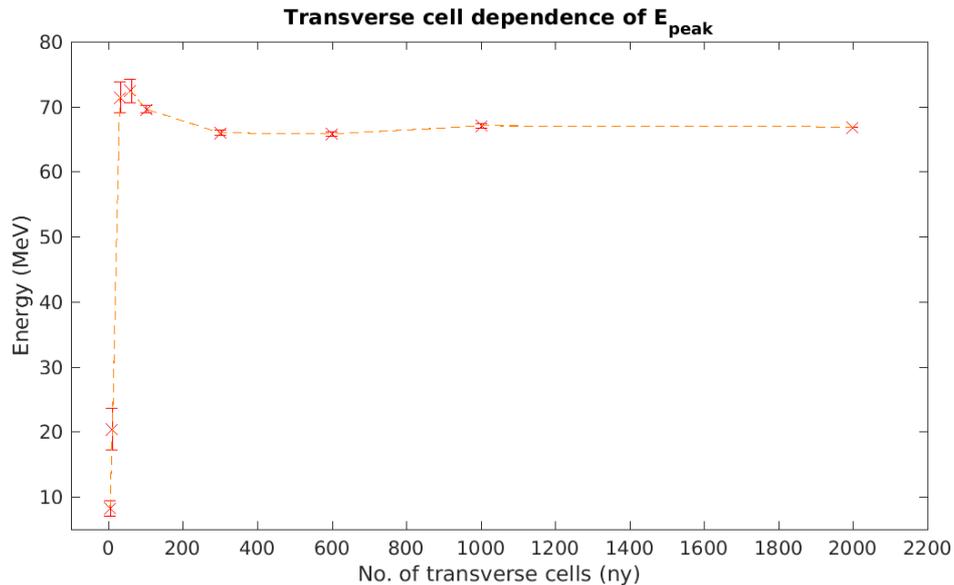


Figure 5.5: The E_{peak} dependence on number of transverse simulation cells. The datapoints depicted are averaged over 3 simulations with different random seeds, with the standard deviation of these values being represented by an error bar.

simulation (see Chapter 4.4, Table 4.2). In order to reduce the statistical noise in these simulations, we thus average over three simulations run with identical parameters, which we showed to converged as in Chapter 5.1, but with different random seeds.

Figure 5.5 depicts the resultant distribution of peak energies for the different number of transverse cells; the errorbars here depict the standard deviation over the values calculated from the simulations, vanishes for 2001 cells. Whilst both 5 and 11 cells significantly underestimate the peak energy, there is little variation from 301 cells onwards. In a sense, this shows that the chosen value of 301 transverse cells as a basis for the convergence tests is a sensible value, and is also well converged.

The corresponding beam emittances are depicted in Figure 5.6. Whilst the beam emittance for 5 and 11 transverse cells could not be determined, the emittance rapidly converged from 61 cells onwards, with only small fluctuations within er-

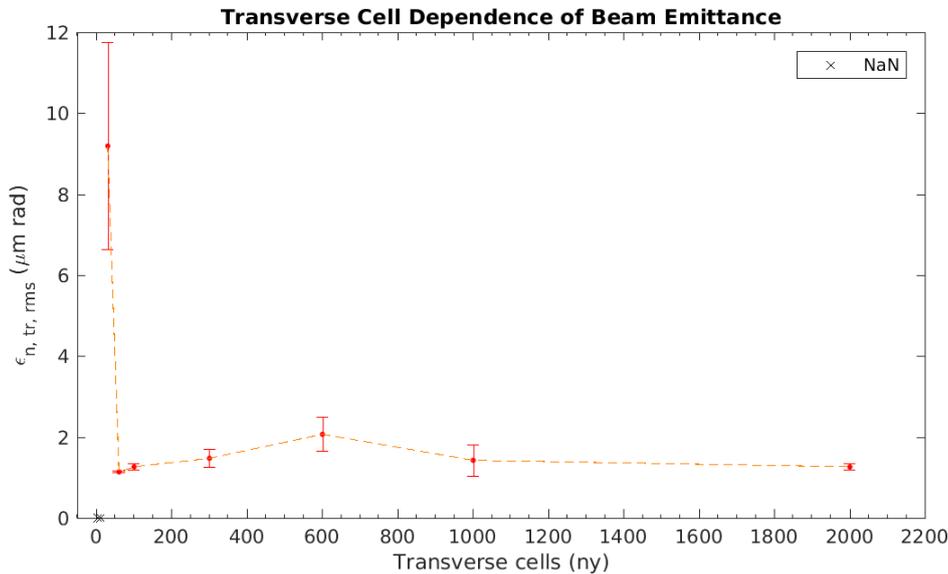


Figure 5.6: The dependence of beam emittance ($\epsilon_{n, \text{tr}, \text{rms}}$) on the number of simulation cells in the transverse direction. Data has been averaged over 3 simulations with different random seeds, depicting the standard deviation as in the form of an error bar. For 5 and 11 transverse cells, the beam was incalculable (NaN).

ror. Although this is somewhat surprising given that preliminary simulations suggested that the beam emittance should converge slower than the peak energy as a function of number of transverse cells, it may be that this was affected by the use of a different density profile and laser parameters. We nevertheless show that, again, 301 transverse cells is sufficient to accurately determine the beam emittance, as well as peak energy in these simulations.

5.3 Nearest-Neighbour Simulation Reduction

Having established the comparison data for full simulations run with low transverse resolution, we now move to the impact of reduction algorithms, and how these compare. As detailed in Chapter 4.5.1, we first test the different nearest-neighbour particle averaging approaches.

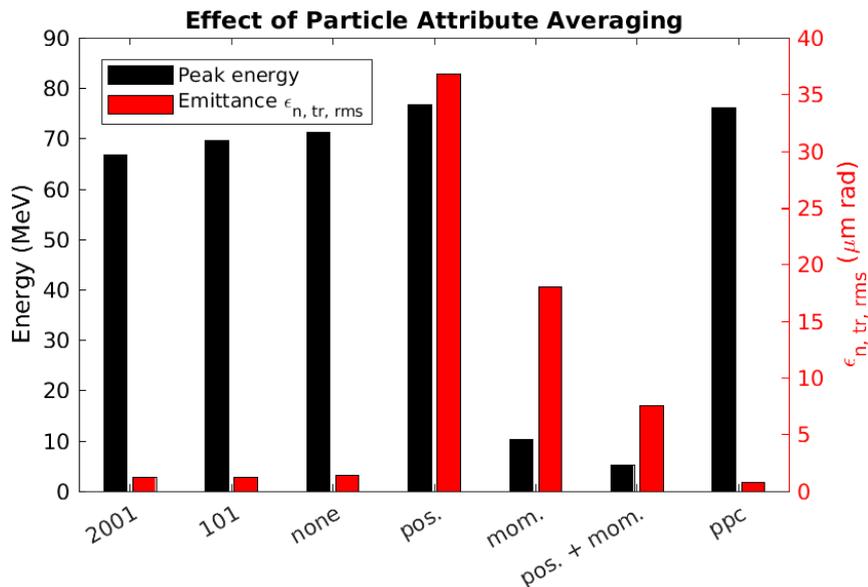


Figure 5.7: Comparison of the different particle attribute averaging (momentum and position) methods, relative to unreduced simulations of full width. **2001** and **101** are the full width simulations with respective numbers of transverse cells. **PPC** is a simulation reduced in PPC only using the NN algorithm, but maintains the field resolution. The others (**none**, **pos.**, **mom.**, **pos. + mom.**) correspond to: no averaging, position averaging, momentum averaging, and the averaging over both position and momentum, respectively.

5.3.1 Nearest-Neighbour Attribute Averaging

The coalescing of particles with an attempt to maintain the beam attributes is non-trivial – since it inherently loses information, there are many possible approaches which attempt to reduce the number of particles simulated without negatively impacting the beam (see Chapter 3). In particular, we now consider the impact of averaging the coalesced particle positions and momenta, compared to no averaging, whilst implementing the nearest-neighbour as described in 4.3.

The results for both peak energy and emittance have been combined in Figure 5.7. Since the post-reduction number of transverse cells is 101, we have included the corresponding full simulation from Chapter 5.2, as well as the full simulation

which was reduced (2001 cells).

It is immediately evident that, as expected, averaging over either momentum or position when coalescing particles using this approach (or a combination of the two) performs significantly worse than both no averaging, and full-width low transverse resolution simulations. Whilst position averaging only slightly over-estimates the peak energy, the calculated emittance is almost 2 orders of magnitude greater. Meanwhile, momentum averaging under-estimates the peak energy by a factor of 7, whilst still overestimating beam emittance by a factor of 40. It is hence unsurprising that the combination of the two does not fare significantly better.

However, both simulations subject to nearest-neighbour reduction which did not involve the averaging of particle momenta or positions performed well (simulations (1) and (5) from Table 4.3). Simulation (1), here represented as 'none', resulted in a peak energy of 71.25 MeV, compared with 66.79 MeV and 69.62 MeV for the 2001 and 101 cell simulations. This corresponds to an increase of 4.46 MeV (6.7%) and 1.63 MeV (2.3%) respectively. The emittance also evaluated slightly higher (1.397 $\mu\text{m rad}$ compared to 1.261 $\mu\text{m rad}$ for both full width simulations), corresponding to an increase of 10.8%. Simulation (5) meanwhile, maintained field resolution and instead decreased the PPC by an equivalent amount using the NN algorithm. This produced a beam with peak energy 76.25 MeV (+14.2%) and emittance 0.780 $\mu\text{m rad}$ (-38.1%), with differences relative to the 2001 cell simulation.

These results suggest that when using the nearest-neighbour algorithm to reduce the number of particles simulated, using a structured uniform sample from the momentum distribution within each super-cell, that it is best to reduce in the number of transverse cells, without any particle averaging.

5.3.2 Large Super-cells

We now analyse the importance in choosing a suitable size for the super-cells. Corresponding to the simulations (1) – (6) in Table 4.4, Figure 5.8 depicts the final beam energies when reducing with "large" super-cells.

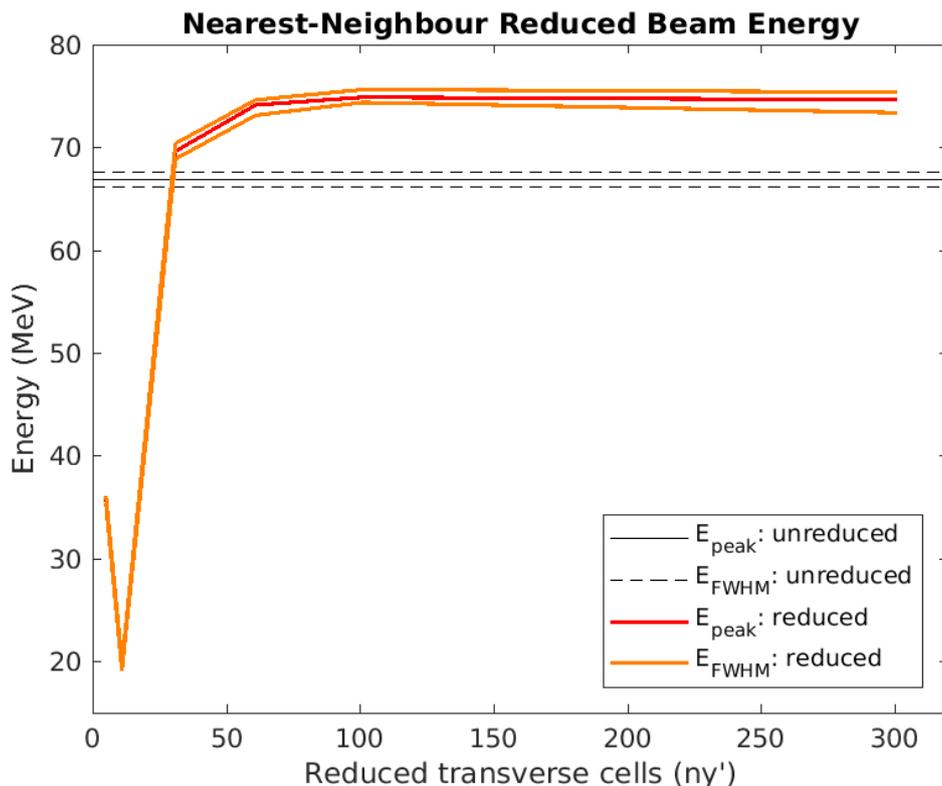


Figure 5.8: Here shown are the positions of the peak energies (E_{peak}), and positions of the upper- and lower FWHM energies (E_{FWHM}), for reduced simulations with "large" super-cells, as compared to the full-width control.

Whilst reducing to 5 and 11 transverse cells caused the peak energy to be significantly underestimated, reducing to 31 cells only overestimates E_{peak} by +2.83 MeV (69.62 MeV, identical to full-width 101 transverse cells), this overestimation increases until it stabilizes at ~ 75 MeV for $\text{ny}' > 101$.

Regarding beam emittance, unlike in Chapter 5.2, this was calculable for both 5 and 11 reduced transverse cells, although it overestimated by a factor 20. The

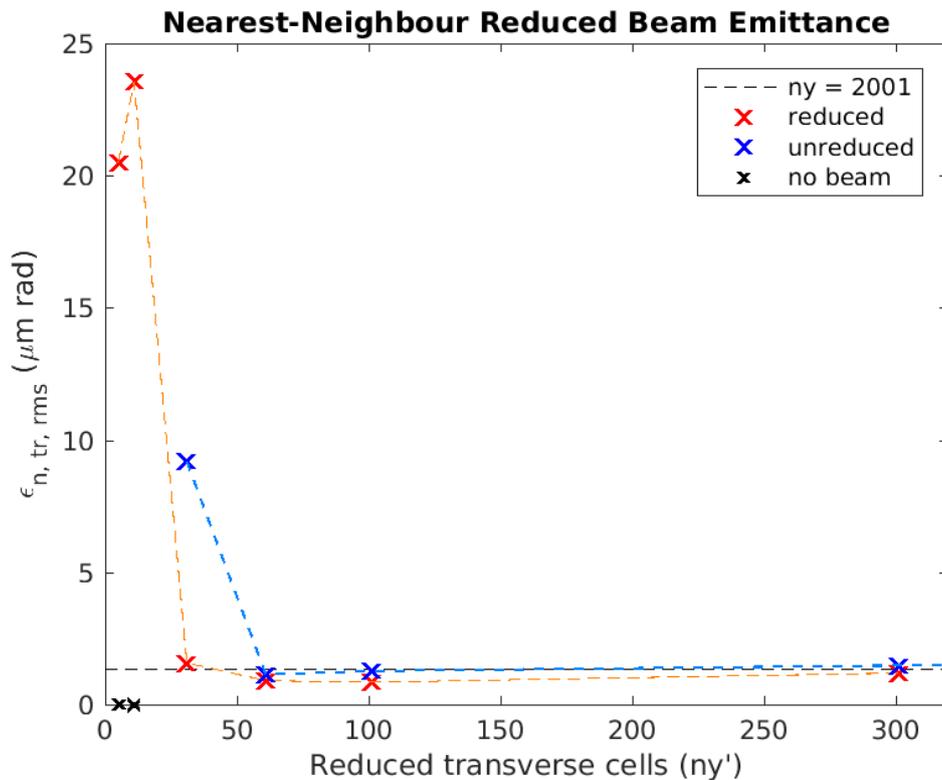


Figure 5.9: Here shown is the calculated beam emittance ($\epsilon_{n, tr, rms}$), for the full-width control (black) and reduced (red) simulations with "large" super-cells. Again, the values for unreduced narrow-width simulations have been included for comparison, and the emittance for 5 and 11 transverse cells was incalculable for the unreduced simulation.

emittance then approached that of the 2001 cell simulation, with only 31 transverse cells, thereby outperforming the unreduced simulations with low numbers of transverse cells. For 61, 101 and 301 reduced cells, however, the emittance converged to a slightly lower value, underestimating this by -12%, compared to +10% for the unreduced 301 cell simulation.

5.3.3 Small Super-cells

Moving to the "small" super-cell simulations (7 – 12, Table 4.4), Figure 5.10 shows the corresponding peak energies.

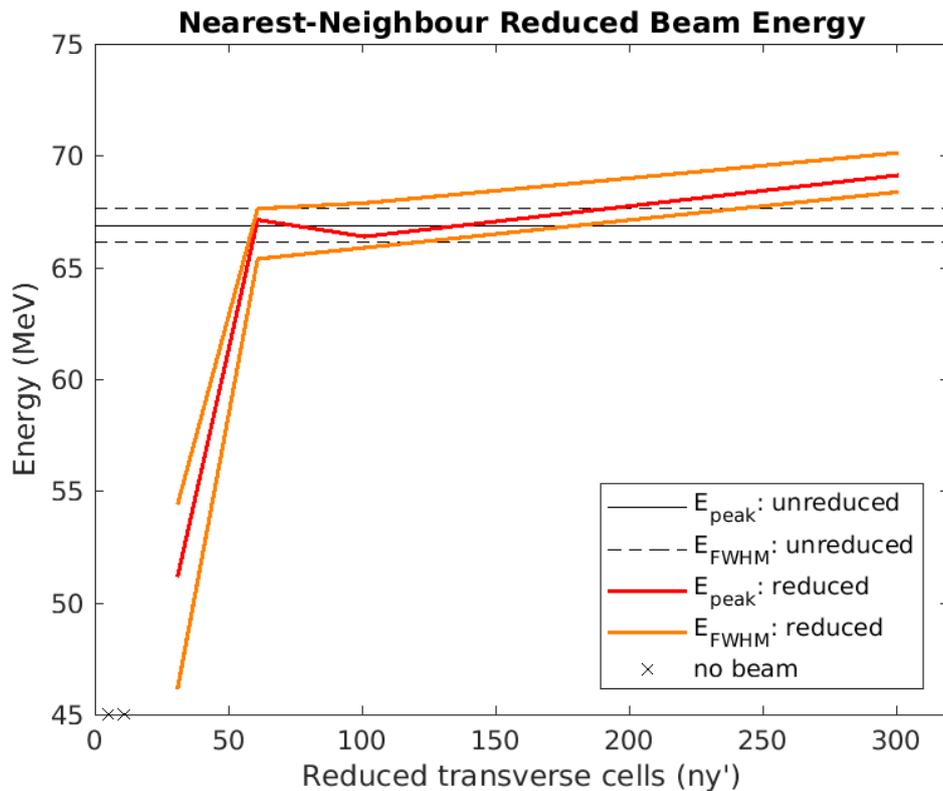


Figure 5.10: Here shown are the positions of the peak energies (E_{peak}), and positions of the upper- and lower FWHM energies (E_{FWHM}), for "small" super-cell reduced simulations as compared to the full-width control. For reduced widths of 5 and 11 cells, no beam was present.

Interestingly, whilst beams were observed in the large super-cell simulations reduced to 5 and 11 cells, this was not the case with small super-cells. Again, 31 reduced cells was insufficient to accurately reproduce the peak energy, being around 20 MeV too low, but reducing to higher cell counts proved more effective than when using larger super-cells. Here, 61 and 101 reduced cells both predicted peak energies within 0.5 MeV of the 2001 cell full simulation, thereby also outperforming the 101 cell unreduced simulation. 301 reduced cells proved slightly worse, perhaps, although it still resulted in a lower deviation than the large super-cells, at +2.24 MeV (+3.4%).

With regards to beam emittance, we now see that the small super-cells perform

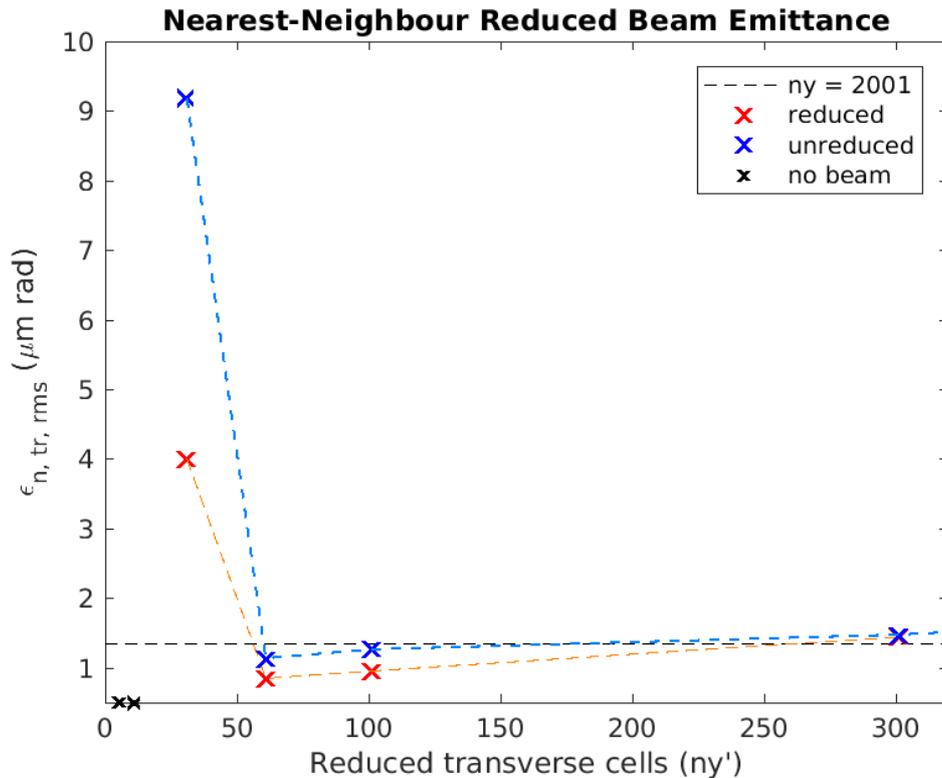


Figure 5.11: Here shown is the calculated beam emittance ($\epsilon_{n, tr, rms}$), for the full-width control (black) and reduced (red) simulations with "small" super-cells. Additionally, the values for unreduced narrow-width simulations have been included for comparison. Once again, the emittance for 5 and 11 transverse cells was incalculable for both reduced and unreduced simulations.

slightly worse for low reduced transverse cells (31, 61, 101) than both unreduced simulations, and large super-cells. However, in the case of 301 reduced transverse cells, this now overlaps exactly with that of the unreduced 301 cell simulation.

5.4 Variable PPC Simulation Reduction

An alternative method to reduce the number of particles being simulated, without algorithmic reduction, involves a spatially-varying particle-per-cell definition – in this case, a step function (see Chapter 3.3). Figure 5.12 shows the variation in

peak energy as a function of number of transverse cells (post-reduction), corresponding to the parameters in Table 4.5.

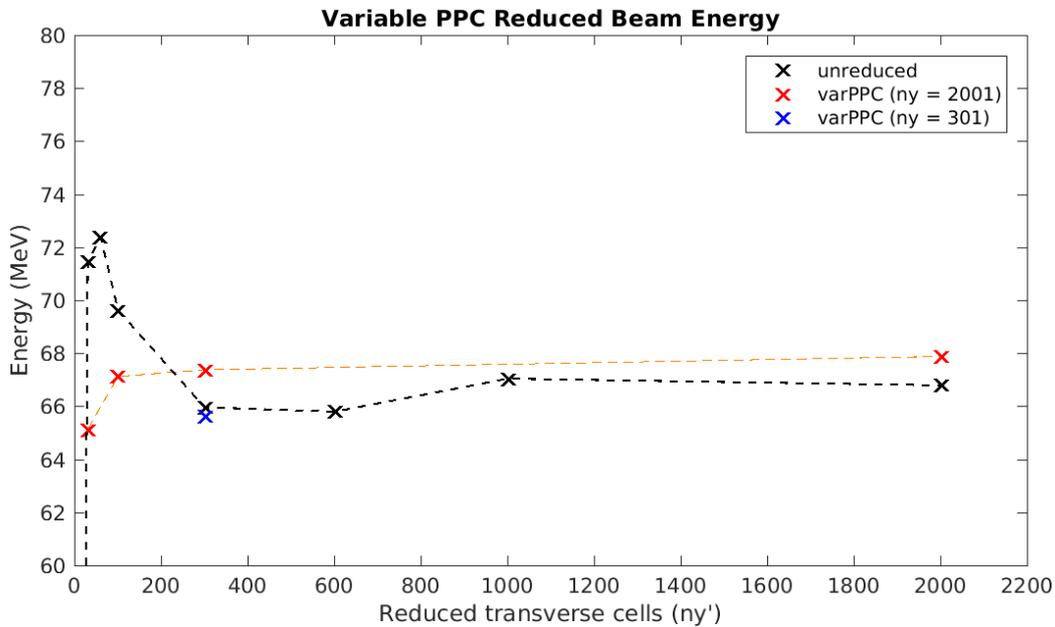


Figure 5.12: Depicted are the peak energies E_{peak} for unreduced (black) and variable PPC simulations (red), where the number of particles per cell is reduced from 32 to 1, and fields are interpolated linearly from the original 2001 transverse cell simulation. Additionally, the energy when reducing from a narrower simulation ($n_y = 301$ cells), with identical decrease in PPC, is depicted in blue.

It is notable that simulations (2) and (3) – those decreasing to 2 and 4 particles-per-cell – are not depicted; this is due to both beam energy and emittance evaluating to the same value as simulation (1). With regards to beam peak energy, it is evident that when using the variable PPC method, this converges rapidly, with little variation beyond 101 transverse cells. Whilst performing significantly better at the lower transverse resolutions as compared to the unreduced simulation, the final energy converges to a slightly higher value: 67.88 MeV as opposed to 66.79 MeV (an increase of 1.6%). Simulation (7), which was instead based on a simulation with $n_y = 301$, only deviated by 0.34 MeV (-0.5%).

Similarly, when examining the beam emittance (Figure 5.13), this again converges

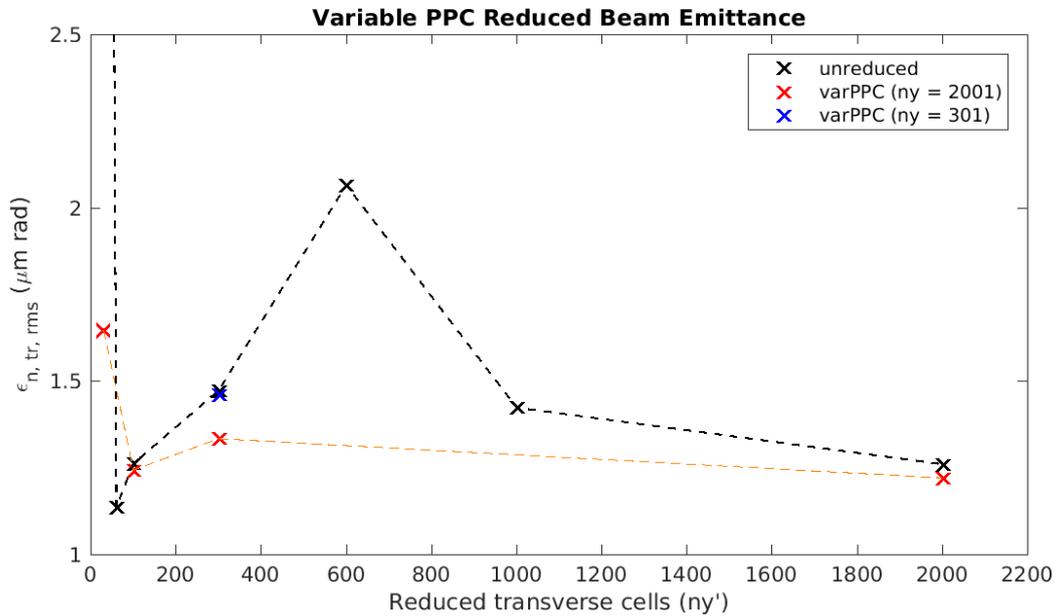


Figure 5.13: Depicted are the calculated beam emittances $\varepsilon_{n, tr, rms}$ for unreduced (black) and variable PPC simulations (red), where the number of particles per cell is reduced from 32 to 1, and fields are interpolated linearly from the original 2001 transverse cell simulation. Additionally, the emittance when reducing from a narrower simulation ($ny = 301$ cells), with identical decrease in PPC, is depicted in blue.

much more rapidly than the unreduced simulation, with a significantly closer value at 31 transverse cells, and little variation above 101 cells. Whilst the emittance calculated for the full-width variable PPC simulations (1 – 3) is slightly lower than the equivalent unreduced simulation, this only represents a difference of $0.04 \mu\text{m rad}$ (-3.2%) – less than half a standard deviation as calculated in Figure 5.6. Simulation (7) performed even better, with a difference from the 301 transverse cell unreduced simulation of merely $0.01 \mu\text{m rad}$ (-0.9%).

5.5 Reduction Spectra

Whilst the macroscopic quantities such as the peak energy E_{peak} and normalised trace-space beam emittance $\varepsilon_{n,\text{tr,rms}}$ (Chapter 2.2) are useful in allowing a direct quantitative comparison between simulations, this of course fails to capture the full detail of the simulation. Figure 5.14 thus depicts the electron energy spectra for some key simulations spanning the previous results sections.

Subfigure (A) displays the spectrum for unreduced full-width simulations with 2001 transverse cells, and 32 PPC. Whilst there are minor statistical fluctuations due to different random seeds, the resultant spectra are near identical. (B) depicts three select small super-cell runs – with 31, 101 and 301 reduced transverse cells. It is evident that although the latter two simulations showed reasonable agreement with the emittance and peak energy, the nearest-neighbour reduction algorithm fails to preserve the spectrum at lower energies, whilst also splitting the main peak into multiple. However, the total injected charge in the principal peak appears to be approximately conserved, unlike in the large super-cell simulations (C). Whilst these similarly fail to preserve the lower energy distribution, the principal peaks are shifted significantly to higher energies, and inject a significantly lower charge.

(D) depicts the spectra for the variable PPC simulations without field interpolation, showing excellent agreement both between different reduced PPC and the full resolution simulations in (A) – not just regarding the principal beam, but across the full energy spectrum. (E) adds additional field interpolation, which causes increasing deterioration in the overall spectrum accuracy as the field resolution is decreased. However, it still performs significantly better than either of the nearest-neighbour reduced simulations.

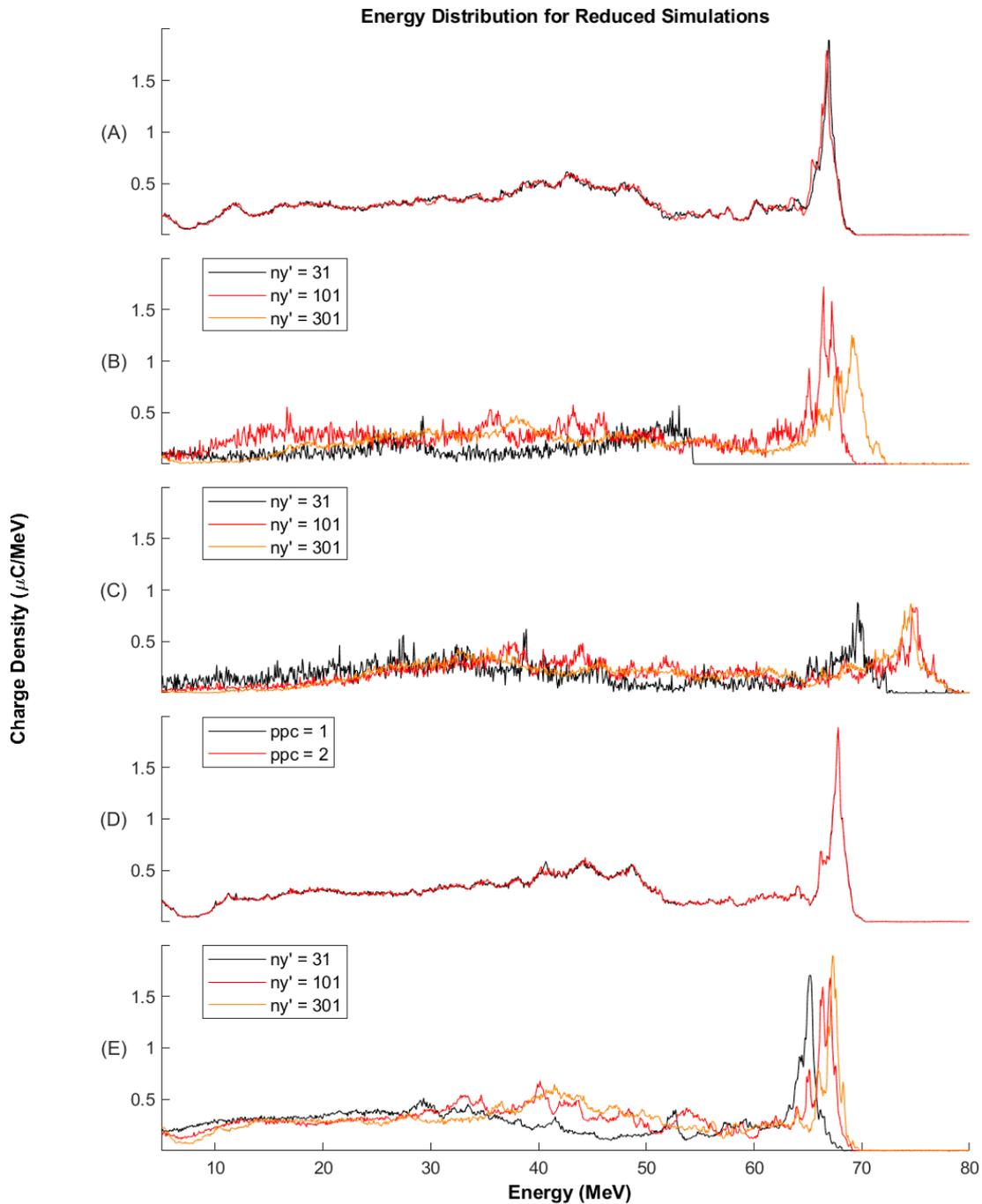


Figure 5.14: Depicted are the full energy spectra for various reduction methods. **(A)** full simulation with 2001 transverse cells, and 32 PPC. **(B)** NN reduction, using the "small" super-cells. **(C)** NN reduction, using the "large" super-cells. **(D)** variable PPC. **(E)** variable PPC, with additional field interpolation.

5.6 Computation Time

Of course, since the principal aim of the various reduction algorithms is a decrease in computation time, it is important to compare these. Figure 5.15 shows the post-reduction CPU-normalised computation time, per simulation femtosecond, as a function of the "reduction factor" – here defined as the inverse product of relative PPC with relative number of post-reduction transverse cells:

$$\text{Reduction Factor} = \frac{n_y}{n_{y'}} \cdot \frac{n_{\text{ppc}}}{n'_{\text{ppc}}}$$

In dotted black, we depict the computation time of the unreduced simulations with varying transverse resolution; since these do not undergo reduction, we instead use a "virtual" reduction factor relative to the unreduced 2001 cell width simulation. Unsurprisingly, both nearest-neighbour reduced simulations (orange, red) are approximately colinear with the unreduced simulations, as these contain an identical number of macro-particles and grid cells post-reduction (with some statistical fluctuation).

The blue lines correspond to simulations reduced using the variable PPC approach; since these preserve the injected macro-particles directly, this results in more particles in a given simulation frame after reduction, compared with the unreduced simulations. Correspondingly, the computation time for equivalent reduction factors is ~ 2 times greater. Nevertheless, the variable PPC simulation reducing to 1 PPC attained a post-reduction speed-up of factor 11.3, whilst preserving the beam almost exactly. Allowing minor decay in the beam profile, further adding field interpolation in combination with a variable PPC reduction to 1 PPC reduced computation time by factors of 37.9 and 90.3, for 301 and 101 transverse post-reduction cells respectively.

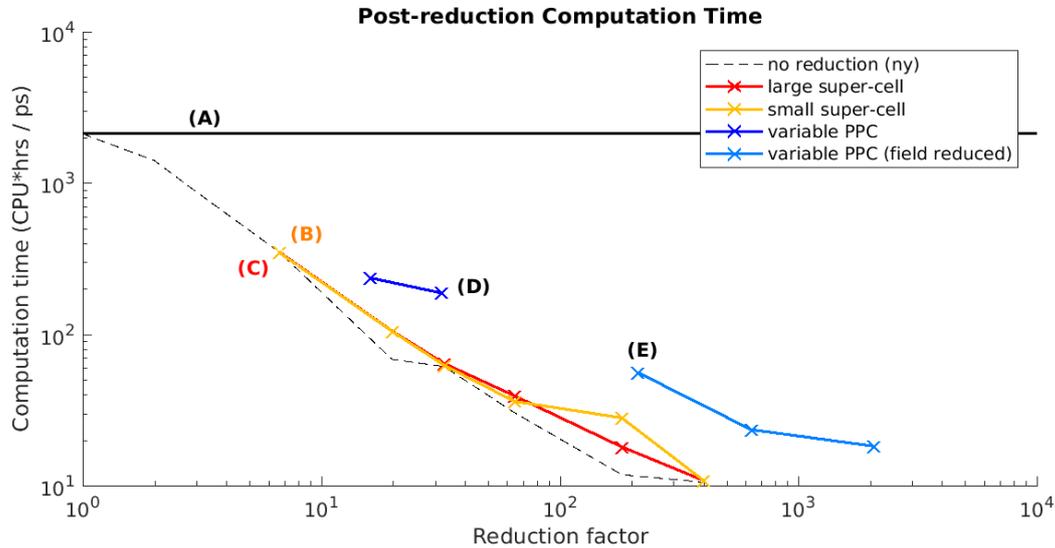


Figure 5.15: We here show the relationship between the (CPU-independent) computation time per simulated picosecond, and the reduction factor (the inverse product of relative number of cells, and relative PPC), for various reduction schemes corresponding to (A, ..., E) in Figure 5.14. In solid black (A), we represent the full-width (2001 transverse cells, 32 PPC) simulation, used as a basis for reduction. In dotted black, we represent unreduced simulations with fewer transverse cells, assigning a virtual reduction factor of $(2001/ny)$. Red and orange represent nearest-neighbour approaches with large- and small super-cells, respectively, whilst the variable PPC method is depicted in blue. (B) and (C) correspond to 301 reduced cell simulations using small- and large super-cell NN reduction respectively. Variable PPC reduction to 1 PPC corresponds to (D), whereas (E) further includes field interpolation to 301 transverse cells.

6 | Discussion

Having examined the different possible methods when reducing simulations using the nearest-neighbour algorithm, it is evident that coalesced macro-particles should retain the momentum and position of the sampled particle, as opposed to averaging over coalesced particles (Chapter 5.3.1). Regarding the optimal size of the super-cells to use, we show that with the given simulation parameters, smaller super-cells were more substantially more effective than larger cells (Chapters 5.3.2, 5.3.3); this likely due to a better local preservation of the particle-momentum distribution. Since a relatively small proportion of electrons simulated are injected, this would hence result in a slightly higher number of high-energy macro-particles in the reduced simulation, as compared to larger cells, which better preserve the global distribution.

When compared with low transverse resolution simulations which are not reduced (Chapter 5.2), however, it is clear nearest-neighbour reduced simulations do not better preserve the electron beam quality, except at very low transverse resolution. When coupled with the high computation time for the initial high-resolution simulation, and no comparative speed-up post-reduction (Chapter 5.6), it rather obvious that nearest-neighbour algorithms do not provide a benefit over unreduced simulations with a relatively low transverse resolution. We thereby also note that simulations with few cells in the transverse direction are viable, and can produce good results, but are more prone to statistical fluctuations; depending on the desired accuracy and quantities of interest, it may be useful to simply average over multiple low transverse resolution simulations, which have a significantly shorter computation time than higher resolution simulations.

In contrast, the variable PPC method reproduced the beam almost perfectly when applied without field interpolation, and with only slight deterioration in the precise

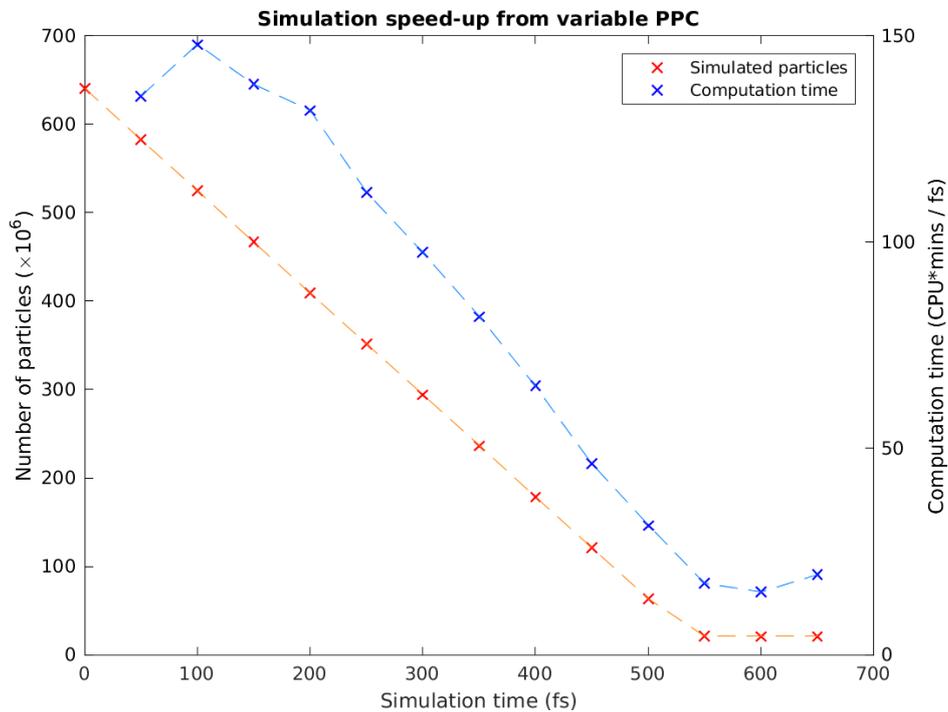


Figure 6.1: For a variable PPC simulation, the number of simulated macro-particles is shown (red) as function of simulation time post-reduction. In blue, the equivalent CPU-normalised computation time per simulated femtosecond is plotted, showing strong correlation with the number of particles as expected.

energy distribution with interpolation (Chapters 5.4, 5.5). Showing significant decrease in computation time post-reduction, this is of course dependent on the ratio of reduced PPC and number of transverse cells, to that in the initial section of the simulation. Whilst not offering an immediate speed-up, since it takes time for non-injected particles to leave the simulation window, we observe a linear speed-up, up to some maximum (Figure 6.1). This method is thus very useful especially when a very high beam resolution and accuracy is desired, and there is a significant post-injection acceleration phase.

It would also be useful to characterise at which point the simulation would best be reduced, using the variable PPC technique, and the effect of various field interpolation methods. Furthermore, the interaction of this method with other existing

schemes aimed to decrease computation time – such as boosted frame simulations – could be considered. This we leave as further work.

7 | Conclusion

Reduction algorithms can be used to increase simulation speed whilst preserving injected electron principal beam spectra and beam emittance, although care needs to be taken regarding specific reduction methods and parameters. We show that for LWFA in particular, greater accuracy is achieved using the nearest-neighbour reduction algorithm without additional averaging of momenta and position, when creating the coalesced particle. The NN algorithm does not show significant improvement over full simulations at low transverse resolutions, however, whilst being more computationally expensive.

The most accurate method which increases simulation speed with negligible loss in accuracy involves changing the number of particles-per-cell count some distance after the step transition (at a position where no more electrons are injected). A further increase based on this method is viable, through the interpolation of the electric- and magnetic fields, thereby further decreasing the number of particles post-reduction with slight deterioration in the precise beam spectrum.

Using the variable PPC method, we observe an order-of-magnitude decrease in the post-reduction computation time when reducing a 2001 transverse cell simulation with 32 PPC to 1 PPC; when incorporating field reduction to 301 and 101 transverse cells, we observe decreases factor 38 and 90 decreases in computation time, respectively. We thus suggest modifying existing PIC codes – such as EPOCH – to allow the definition of a spatially- or temporally varying particle-per-cell distribution; further changes to allow a variation in the grid resolution, to further increase speeds at the cost of some precision, would be recommended but may be more difficult to implement.

Bibliography

- [1] T. Tajima and J. M. Dawson, *Laser electron accelerator*, [Phys. Rev. Lett. **43**, 267 \(1979\)](#).
- [2] W. P. Leemans, D. Rodgers, P. E. Catravas, C. G. R. Geddes, G. Fubiani, E. Esarey, B. A. Shadwick, R. Donahue, and A. Smith, *Gamma-neutron activation experiments using laser wakefield accelerators*, [Physics of Plasmas **8**, 2510 \(2001\)](#).
- [3] M. Jong, *Producing medical isotopes using x-rays*, (2012) pp. 3177–3179.
- [4] F. Massimo, A. F. Lifschitz, C. Thaury, and V. Malka, *Numerical studies of density transition injection in laser wakefield acceleration*, [Plasma Physics and Controlled Fusion **59**, 085004 \(2017\)](#).
- [5] C. Schroeder, E. Esarey, W. Leemans, J. van Tilborg, F. J. Grüner, A. R. Maier, Y. Ding, and Z. Huang, *Free-electron lasers driven by laser-plasma accelerators using decompression or dispersion*, in [Proc. of 2013 Free-Electron Laser Conference, New York, NY, USA \(2013\)](#) pp. 117–121.
- [6] A. R. Maier, A. Meseck, S. Reiche, C. B. Schroeder, T. Seggebrock, and F. Grüner, *Demonstration scheme for a laser-plasma-driven free-electron laser*, [Phys. Rev. X **2**, 031019 \(2012\)](#).
- [7] S. Chen, N. D. Powers, I. Ghebregziabher, C. M. Maharjan, C. Liu, G. Golovin, S. Banerjee, J. Zhang, N. Cunningham, A. Moorti, S. Clarke, S. Pozzi, and D. P. Umstadter, *Mev-energy x rays from inverse compton scattering with laser-wakefield accelerated electrons*, [Phys. Rev. Lett. **110**, 155003 \(2013\)](#).
- [8] A. Buck, J. Wenz, J. Xu, K. Khrennikov, K. Schmid, M. Heigoldt, J. M.

- Mikhailova, M. Geissler, B. Shen, F. Krausz, S. Karsch, and L. Veisz, *Shock-front injector for high-quality laser-plasma acceleration*, [Phys. Rev. Lett. **110**, 185006 \(2013\)](#).
- [9] K. K. Swanson, H.-E. Tsai, S. K. Barber, R. Lehe, H.-S. Mao, S. Steinke, J. van Tilborg, K. Nakamura, C. G. R. Geddes, C. B. Schroeder, E. Esarey, and W. P. Leemans, *Control of tunable, monoenergetic laser-plasma-accelerated electron beams using a shock-induced density downramp injector*, [Phys. Rev. Accel. Beams **20**, 051301 \(2017\)](#).
- [10] V. Malka, *Plasma wake accelerators: Introduction and historical overview*, [CERN Yellow Reports **1**, 1 \(2016\)](#).
- [11] G. Golovin, S. Banerjee, S. Chen, N. Powers, C. Liu, W. Yan, J. Zhang, P. Zhang, B. Zhao, and D. Umstadter, *Control and optimization of a staged laser-wakefield accelerator*, [Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment **830**, 375 \(2016\)](#).
- [12] C. Baird, *Simulation of laser wakefield accelerated electrons using epoch code*, MSc Thesis, University of York (2015).
- [13] W. P. Leemans, J. van Tilborg, J. Faure, C. G. R. Geddes, C. Tóth, C. B. Schroeder, E. Esarey, G. Fubiani, and G. Dugan, *Terahertz radiation from laser accelerated electron bunches*, [Physics of Plasmas **11**, 2899 \(2004\)](#).
- [14] J. van Tilborg, C. B. Schroeder, C. V. Filip, C. Tóth, C. G. R. Geddes, G. Fubiani, R. Huber, R. A. Kaindl, E. Esarey, and W. P. Leemans, *Temporal characterization of femtosecond laser-plasma-accelerated electron bunches using terahertz radiation*, [Phys. Rev. Lett. **96**, 014801 \(2006\)](#).

- [15] E. Esarey, C. B. Schroeder, and W. P. Leemans, *Physics of laser-driven plasma-based electron accelerators*, *Rev. Mod. Phys.* **81**, 1229 (2009).
- [16] G. Mourou and D. Umstadter, *Development and applications of compact high-intensity lasers*, *Physics of Fluids B: Plasma Physics* **4**, 2315 (1992).
- [17] V. Malka, *Laser Plasma Accelerators: Towards High Quality Electron Beam* (Intech, 2010).
- [18] V. Malka, S. Fritzler, E. Lefebvre, M.-M. Aeonard, F. Burgy, J.-P. Chambaret, J.-F. Chemin, K. Krushelnick, G. Malka, S. P. D. Mangles, Z. Najmudin, M. Pittman, J.-P. Rousseau, J.-N. Scheurer, B. Walton, and A. E. Dangor, *Electron acceleration by a wake field forced by an intense ultrashort laser pulse*, *Science* **298**, 1596 (2002).
- [19] C. E. Clayton, C. Joshi, C. Darrow, and D. Umstadter, *Relativistic plasma-wave excitation by collinear optical mixing*, *Phys. Rev. Lett.* **54**, 2343 (1985).
- [20] X. Davoine, E. Lefebvre, C. Rechatin, J. Faure, and V. Malka, *Cold optical injection producing monoenergetic, multi-gev electron bunches*, *Phys. Rev. Lett.* **102**, 065001 (2009).
- [21] J. Faure, Y. Glinec, A. Pukhov, S. Kiselev, S. Gordienko, E. Lefebvre, J.-P. Rousseau, F. Burgy, and V. Malka, *A laser-plasma accelerator producing monoenergetic electron beams*, *Nature* **431**, 541 (2004).
- [22] M. N. Rosenbluth and C. S. Liu, *Excitation of plasma waves by two laser beams*, *Phys. Rev. Lett.* **29**, 701 (1972).
- [23] D. Strickland and G. Mourou, *Compression of amplified chirped optical pulses*, *Optics Communications* **55**, 447 (1985).

-
- [24] S. Y. Kalmykov, A. Beck, S. A. Yi, V. N. Khudik, M. C. Downer, E. Lefebvre, B. A. Shadwick, and D. P. Umstadter, *Electron self-injection into an evolving plasma bubble: Quasi-monoenergetic laser-plasma acceleration in the blowout regime*, [Physics of Plasmas](#) **18**, 056704 (2011).
- [25] P. Sprangle, E. Esarey, A. Ting, and G. Joyce, *Laser wakefield acceleration and relativistic optical guiding*, [Appl. Phys. Lett.](#) **53**, 2146 (1988).
- [26] K. Schmid, A. Buck, C. M. S. Sears, J. M. Mikhailova, R. Tautz, D. Herrmann, M. Geissler, F. Krausz, and L. Veisz, *Density-transition based electron injector for laser driven wakefield accelerators*, [Phys. Rev. ST Accel. Beams](#) **13**, 091301 (2010).
- [27] L. M. Gorbunov and V. I. Kirsanov, *Excitation of plasma waves by an electromagnetic wave packet*, [Sov. Phys. JETP](#) **66**, 290 (1986).
- [28] P. Sprangle, E. Esarey, J. Krall, and G. Joyce, *Propagation and guiding of intense laser pulses in plasmas*, [Phys. Rev. Lett.](#) **69**, 2200 (1992).
- [29] N. Andreev, L. Gorbunov, V. Kirsanov, A. Pogosova, and R. Ramazashvili, *Resonant excitation of wakefields by a laser pulse in a plasma*, *Jetp Letters - JETP LETT-ENGL TR* **55** (1992).
- [30] X. F. Li, Q. Yu, Y. J. Gu, S. Huang, Q. Kong, and S. Kawata, *Bubble shape and electromagnetic field in the nonlinear regime for laser wakefield acceleration*, [Physics of Plasmas](#) **22**, 083112 (2015).
- [31] C. Joshi and W. B. Mori, *The status and evolution of plasma wakefield particle accelerators*, [Phil. Trans. R. Soc. A](#) **364**, 577 (2005).
- [32] A. Pukhov and J. Meyer-ter Vehn, *Laser wake field acceleration: the highly non-linear broken-wave regime*, [Appl. Phys. B](#) **74**, 355 (2002).

- [33] S. Mangles, C. Murphy, and Z. e. a. Najmudin, *Monoenergetic beams of relativistic electrons from intense laser-plasma interactions*, [Nature](#) , 535 (2004).
- [34] R. Bingham, *Basic concepts in plasma accelerators*, [Phil. Trans. R. Soc. A](#) **364**, 559 (2006).
- [35] S. M. Hooker, *Developments in laser-driven plasma accelerators*, [Nature](#) **7**, 775 (2013).
- [36] E. W. Weisstein, *Convective derivative*, from [Wolfram MathWorld](#) (accessed: 23/10/19).
- [37] W. L. Kruer, *The physics of laser plasma interactions*, *Frontiers in physics* ; v. 73 (Addison-Wesley, 1988).
- [38] G. Dattoli, A. Doria, E. Sabia, and M. Artioli, *Charged Beam Dynamics, Particle Accelerators and Free Electron Lasers*, 2053-2563 (IOP Publishing, 2017).
- [39] C. Benedetti, C. B. Schroeder, E. Esarey, F. Rossi, and W. P. Leemans, *Numerical investigation of electron self-injection in the nonlinear bubble regime*, [Phys. Plasmas](#) **20**, 103108 (2013).
- [40] J. M. Dawson, *Nonlinear electron oscillations in a cold plasma*, [Phys. Rev.](#) **113**, 383 (1959).
- [41] A. I. Akhiezer and R. V. Polovin, *Theory of wave motion of an electron plasma*, [Sov. Phys. JETP](#) **3**, 696 (1956).
- [42] K. Nakamura, B. Nagler, C. Tóth, C. G. R. Geddes, C. B. Schroeder, E. Esarey, W. P. Leemans, A. J. Gonsalves, and S. M. Hooker, *GeV electron*

- beams from a centimeter-scale channel guided laser wakefield accelerator*, [Physics of Plasmas](#) **14**, 056708 (2007).
- [43] A. Döpp, C. Thaury, E. Guillaume, F. Massimo, A. Lifschitz, I. Andriyash, J.-P. Goddet, A. Tazfi, K. Ta Phuoc, and V. Malka, *Energy-chirp compensation in a laser wakefield accelerator*, [Phys. Rev. Lett.](#) **121**, 074802 (2018).
- [44] S. Bulanov, N. Naumova, F. Pegoraro, and J. Sakai, *Particle injection into the wave acceleration phase due to nonlinear wake wave breaking*, [Phys. Rev. E](#) **58**, R5257 (1998).
- [45] C. G. R. Geddes, K. Nakamura, G. R. Plateau, C. Toth, E. Cormier-Michel, E. Esarey, C. B. Schroeder, J. R. Cary, and W. P. Leemans, *Plasma-density-gradient injection of low absolute-momentum-spread electron bunches*, [Physical review letters](#) **100**, 215004 (2008).
- [46] A. Deng, O. Karger, T. Heinemann, A. Knetsch, P. Scherkl, G. Manahan, A. Beaton, D. Ullmann, G. Wittig, A. Habib, M. Litos, S. Gessner, C. Clarke, S. Green, C. Lindström, E. Adli, R. Zgadzaj, M. Downer, G. Andonian, A. Murokh, D. Bruhwiler, J. Cary, M. Hogan, V. Yakimenko, J. Rosenzweig, and B. Hidding, *Generation and acceleration of electron bunches from a plasma photocathode*, [Nature Physics](#) **15**, 1156 (2019).
- [47] A. Gonsalves, K. Nakamura, C. Lin, D. Panasencko, S. Shiraishi, T. Sokollik, C. Benedetti, C. Schroeder, C. G. Geddes, J. van Tilborg, J. Osterhoff, E. Esarey, C. Toth, and W. Leemans, *Tunable laser plasma accelerator based on longitudinal density tailoring*, [Nature Physics](#) **7**, 862 (2011).
- [48] E. Thaury, C. Guillaume, A. Lifschitz, K. Ta Phuoc, M. Hansson, G. Grittani, J. Gautier, J.-P. Goddet, A. Tafzi, O. Lundh, and V. Malka, *Shock*

- assisted ionization injection in laser-plasma accelerators*, [Nature](#) **5** (2015), [10.1038/srep16310](#).
- [49] J. Xu, A. Buck, S.-W. Chou, K. Schmid, B. Shen, T. Tajima, M. C. Kaluza, and L. Veisz, *Dynamics of electron injection in a laser-wakefield accelerator*, [Phys. Plasmas](#) **24**, 083106 (2017).
- [50] K. Floettmann, *Some basic features of the beam emittance*, [Phys. Rev. ST Accel. Beams](#) **6**, 034202 (2003).
- [51] T. Mehrling, J. Grebenyuk, F. S. Tsung, K. Floettmann, and J. Osterhoff, *Transverse emittance growth in staged laser-wakefield acceleration*, [Phys. Rev. ST Accel. Beams](#) **15**, 111303 (2012).
- [52] S. Balascuta, *Numerical calculations of the electron beam emittance for laser acceleration experiments*, [AIP Conference Proceedings](#) **1852**, 070004 (2017).
- [53] J. M. Dawson, *Particle simulation of plasmas*, [Rev. Mod. Phys.](#) **55**, 403 (1983).
- [54] G. Holt, *Laser wakefield acceleration of muons*, MSc Thesis, University of York (2018).
- [55] R. W. Hockney and J. W. Eastwood, *Computer simulation using particles* (IOP Publishing, 1988).
- [56] C. Ren, (*HEDP summer school lecture*), (2009).
- [57] C. Birdsall and A. Langdon, *Plasma Physics via Computer Simulation*, Series in Plasma Physics (Taylor & Francis, 2004).
- [58] M. Vranic, J. L. Martins, R. A. Fonseca, and L. O. Silva, *Classical radia-*

- tion reaction in particle-in-cell simulations*, [Comp. Phys. Comm. **204**, 141 \(2016\)](#).
- [59] T. D. Arber, K. Bennett, C. S. Brady, A. Lawrence-Douglas, M. G. Ramsay, N. J. Sircombe, P. Gillies, R. G. Evans, H. Schmitz, A. R. Bell, and C. P. Ridgers, *Contemporary particle-in-cell approach to laser-plasma modelling*, [Plasma Physics and Controlled Fusion **57**, 113001 \(2015\)](#).
- [60] X. S. Wei, Y. Xiao, A. Kuley, and Z. Lin, *Method to integrate full particle orbit in toroidal plasmas*, [Physics of Plasmas **22**, 092502 \(2015\)](#).
- [61] J. P. Boris, *Relativistic plasma simulation-optimization of a hybrid code*, in *Proceedings of the 4th Conference on Numerical Simulation of Plasmas, Naval Res. Lab., Washington, D.C., Nov 1970*, Conference on Numerical Simulation of Plasmas (1970) pp. 3–67.
- [62] K. Yee, *Numerical solution of initial boundary value problems involving maxwell's equations in isotropic media*, [IEEE Transactions on Antennas and Propagation **14**, 302 \(1966\)](#).
- [63] F. S. Tsung, W. Lu, M. Tzoufras, W. B. Mori, C. Joshi, J. M. Vieira, L. O. Silva, and R. A. Fonseca, *Simulation of monoenergetic electron generation via laser wakefield accelerators for 5-25tw lasers*, [Physics of Plasmas **13**, 056708 \(2006\)](#).
- [64] D. F. Gordon, W. B. Mori, and T. M. Antonsen, *A ponderomotive guiding center particle-in-cell code for efficient modeling of laser-plasma interactions*, [IEEE Transactions on Plasma Science **28**, 1135 \(2000\)](#).
- [65] P. Yu, *Lorentz boosted frame simulation technique in Particle-in-cell methods*, [Ph.D. thesis, UCLA \(2016\)](#).
- [66] W. B. Mori, V. K. Decyk, T. C. Katsouleas, D. W. Forslund, R. D. Ferraro,

- and E. Horowitz, *Full-scale numerical experiment to test a future plasma-based accelerator*, Submitted to NSF as part of the U.S. High Performance Computing and Communications Program (1992).
- [67] J.-L. Vay, *Noninvariance of space- and time-scale ranges under a lorentz transformation and the implications for the study of relativistic interactions*, [Phys. Rev. Lett. **98**, 130405 \(2007\)](#).
- [68] C. D. Decker and W. B. Mori, *Group velocity of large amplitude electromagnetic waves in a plasma*, [Phys. Rev. Lett. **72**, 490 \(1994\)](#).
- [69] P. Yu *et al.*, *Lorentz boosted frame simulation of laser wakefield acceleration using hybrid yee-fft solver in quasi-3d geometry*, in [Proc. 6th International Particle Accelerator Conference \(IPAC'15\), Richmond, VA, USA, May 3-8, 2015](#), International Particle Accelerator Conference No. 6 (JACoW, Geneva, Switzerland, 2015) pp. 691–693.
- [70] C. H. Shon, H. J. Lee, and J. K. Lee, *Method to increase the simulation speed of particle-in-cell (pic) code*, [Comput. Phys. Commun. **141**, 322 \(2001\)](#).
- [71] J. Teunissen and U. Ebert, *Controlling the weights of simulation particles: adaptive particle management using k-d trees*, [J. Comput. Physics **259**, 318 \(2014\)](#).
- [72] H. Wang, X. Fu, R. Wang, and Y. Li, *New macroparticle coalescing models that conserve particle's phase-space distribution in 3-d particle-in-cell simulations of plasmas*, [IEEE Trans. Plasma Sci. **44**, 2638 \(2016\)](#).
- [73] D. Faghihi, V. Carey, C. Michoski, R. Hager, S. Janhunen, C.-S. Chang, and R. Moser, *Moment preserving constrained resampling with applications to particle-in-cell methods*, (2017), [arXiv:1702.05198 \[physics.plasm-ph\]](#) .