Czech Technical University in Prague
Faculty of Nuclear Sciences and Physical Engineering
Department of Physics

Radiation generation during laser and particle beam interactions in particle-in-cell codes

Master's Thesis

Author: Bc. Patrik Puškáš
Supervisor: Ing. Miroslav Krůs, Ph.D.
Consultants: Ing. Dominika Mašlárová
Ing. Róbert Babjak
Year: 2022/2023

České vysoké učení technické v Praze
Fakulta jaderná a fyzikálně inženýrská
Katedra fyziky

Generace záření při interakci laserových a částicových svazků v particle-in-cell kódech

Diplomová práce

Autor: Bc. Patrik Puškáš
Vedoucí práce: Ing. Miroslav Krůs, Ph.D.
Konzultanti: Ing. Dominika Mašlárová
Ing. Róbert Babjak
Rok: 2022/2023

FAKULTA JADERNÁ A FYZIKÁLNĚ INŽENÝRSKÁ
KATEDRA FYZIKY

ZADÁNÍ DIPLOMOVÉ PRÁCE

Akademický rok: 2022/2023

Student: Bc. Patrik Puškáš

Studijní program: Jaderná a částicová fyzika

Název práce: (česky) Generace záření při interakci laserových a částicových svazků v particle-in-cell kódech

Název práce: (anglicky) Radiation generation during laser and particle beam interactions in particle-in-cell codes

Jazyk práce: anglický

Pokyny pro vypracování:

1) Seznamte se s metodou kinetického numerického modelování „particle-in-cell” (PIC).

2) Prostudujte způsoby numerické simulace fotonové emise z vysokoenergetických částic pohybujících se v silných elektromagnetických polí (tzv. radiační reakce) v PIC kódech. Popište klasický a kvantový režim radiační reakce a rozdíl v jejich simulování.

3) Seznamte se s prostředím výpočetního gridu MetaCentrum a vzdáleným zpracováním a vyhodnocováním dat z PIC kódu.

4) Simulujte konkrétní případ režimu klasické a konkrétní případ režimu kvantové radiační reakce ve dvou vhodně vybraných PIC kódech. Případné rozdíly mezi výsledky diskutujte.

Doporučená literatura:

[1] LANDAU, L. D.; LIFSHITZ, E. M. The classical theory of fields. Butter-worth and Heinemann: 1975.

[2] DI PIAZZA, A., et al. Extremely high-intensity laser interactions with fundamental quantum systems. Reviews of Modern Physics 84.3 (2012): 1177.

[3] DEROUILLAT, J., et al. SMILEI: a collaborative, open-source, multi-purpose particle-in-cell code for plasma simulation, Computer Phys. Commun. 222 (2018), 351-373.

[4] VRANIC, M., et al. Classical radiation reaction in particle-in-cell simulations. Computer Physics Communications 204 (2016), 141-151.

[5] ARBER, T. D., et al. Contemporary particle-in-cell approach to laser-plasma modelling. Plasma Physics and Controlled Fusion 57 (2015), 113001.

Jméno a pracoviště vedoucího diplomové práce:
Ing. Miroslav Krůs, Ph.D., Ústav fyziky plazmatu AV ČR, v. v. i.
Katedra fyziky, Fakulta jaderná a fyzikálně inženýrská ČVUT v Praze

Konzultant:
Ing. Dominika Mašlárová, Ústav fyziky plazmatu AV ČR, v. v. i.;
Katedra fyzikální elektroniky, Fakulta jaderná a fyzikálně inženýrská ČVUT v Praze
Ing. Róbert Babjak, Instituto Superior Técnico Lisabon, Portugalsko
Ústav fyziky plazmatu Akademie věd ČR, v. v. i.

Datum zadání diplomové práce: 20.10.2022

Termín odevzdání diplomové práce: 03.05.2023

Doba platnosti zadání je dva roky od data zadání.

- 3 -

....
garant studijního programu

....
vedoucí katedry

....
Děkan

V Praze dne 20.10.2022

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
Fakulta jaderná a fyzikálně inženýrská

Břehová 7
115 19 Praha 1

PROHLÁŠENÍ

Já, níže podepsaný

Jméno a příjmení studenta:	Patrik Puškáš
Osobní číslo:	473407
Název studijního programu (oboru):	Jaderná a částicová fyzika

prohlašuji, že jsem diplomovou práci s názvem:

Generace záření při interakci laserových a částicových svazků v particle-in-cell kódech

vypracoval samostatně a uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze dne 3.5.2023

....
podpis

Acknowledgment

First of all, I would like to thank to the two people most responsible for this thesis seeing the light of day, my supervisor Ing. Miroslav Krůs, PhD. and my consultant Ing. Dominika Mašlárová, for their time, patience, understanding and very valuable leadership. I also want to thank my consultant Ing. Róbert Babjak, for his valuable advice and feedback. Additionally, I would like to appreciate my friends and family, all of the people close to me, for their support.

Computational resources were provided by the e-INFRA CZ project (ID:90140), supported by the Ministry of Education, Youth and Sports of the Czech Republic.

Bc. Patrik Puškáš

Název práce:
Generace záření při interakci laserových a částicových svazků v particle-in-cell kódech

Autor:	Bc. Patrik Puškáš
Studijní program:	Jaderná a částicová fyzika
Druh práce:	Diplomová práce
Vedoucí práce:	Ing. Miroslav Krůs, Ph.D. Ústav fyziky plazmatu Akademie věd ČR, v. v. i. Fakulta jaderná a fyzikálně inženýrská, ČVUT v Praze
Konzultanti:	Ing. Dominika Mašlárová Ústav fyziky plazmatu Akademie věd ČR, v. v. i. Fakulta jaderná a fyzikálně inženýrská, ČVUT v Praze Ing. Róbert Babjak Instituto Superior Técnico Lisabon, Portugalsko Ústav fyziky plazmatu Akademie věd ČR, v. v. i.

Abstrakt: Táto práca sa venuje porovnaniu numerických algoritmov implementovaných na modelovanie účinkov radiačnej reakcie v particle-in-cell (PIC) kódoch pre klasický aj kvantový režim. Pohyb častíc v ultrarelativistickom režime môže byť do značnej miery ovplyvnený vyžiarením vysokoenergetických fotónov v dôsledku účinkov radiačnej reakcie. Tento jav je významný pre laser-plazmové intarakcie a v astrofyzike. V PIC kódoch boli doposiaľ implementované rôzne numerické modely radiačnej reakcie, vrátane Abraham-Lorentzovej sily, rozvoja Fokker-Planckovej rovnice alebo Monte-Carlo generátora. V tejto práci je porovnaná implementácia týchto algoritmov v klasických, semi-kvantových a plne kvantových režimoch pre dva populárne voľne šíriteľné PIC kódy Smilei a EPOCH. Porovnaná je aj implementácia tvorby elektrón-pozitrónových párov Breit-Wheelerovým procesom. V závere práce sú prezentované a diskutované výsledky simulácii pre rôzne režimy radiačnej reakcie.

Klíčová slova: radiačná reakcia, Breit–Wheelerov proces, particle-in-cell, laser-plazmová interakcia, relativistické elektróny

Title:
Radiation generation during laser and particle beam interactions in particle-in-cell codes
Author: Bc. Patrik Puškáš

Abstract: This thesis presents a comparative study of numerical approaches for implementing radiation reaction effects in particle-in-cell (PIC) simulations in both classical and quantum regimes. The motion of particles in the ultra-relativistic regime can be severely affected by high-energy photon emission due to the effects of radiation reaction, which is of interest to the laser-plasma and astrophysics communities. Various theoretical works have been proposed and implemented in PIC codes to deal with radiation reaction effects in classical and quantum regimes, including a radiation friction force, Fokker-Planck expansion, or a Monte Carlo procedure. In this thesis, implementation of such approaches in classical, semi-quantum and fully quantum regimes is compared for two widely-used open source codes, Smilei and EPOCH. The implementation of Breit-Wheeler pair creation is also compared. The results and benchmarks are presented and discussed in detail.

Key words: radiation reaction, Breit-Wheeler pair creation, particle-in-cell, laser-plasma interaction, relativistic electron motion

Contents

Introduction **1**

1 The Particle-In-Cell method for collisionless plasmas **3**
- 1.1 The Maxwell-Vlasov model . . . 4
- 1.2 Macro-particles . . . 6
- 1.3 Mathematical definition of the macro-particle shape function in higher dimensions . . . 7
- 1.4 Initialization of the simulation . . . 8
- 1.5 The PIC loop . . . 9
- 1.6 Interpolation of the electric and magnetic fields . . . 9
- 1.7 The particle pusher . . . 10
- 1.8 The current deposition . . . 12
- 1.9 Maxwell's solver and the Yee grid . . . 13
  - 1.9.1 Yee grid . . . 13
  - 1.9.2 Maxwell's solver . . . 14
  - 1.9.3 CFL condition . . . 15
- 1.10 Popular PIC codes . . . 15
  - 1.10.1 Smilei . . . 15
  - 1.10.2 EPOCH . . . 16
  - 1.10.3 OSIRIS . . . 16
  - 1.10.4 PIConGPU . . . 16
  - 1.10.5 WarpX . . . 17
- 1.11 Grid computing . . . 17
  - 1.11.1 Metacentrum . . . 18

2 The theory of radiation reaction **21**
- 2.1 Classical radiation reaction . . . 21
  - 2.1.1 Covariant form of the Lorentz force . . . 21
  - 2.1.2 Close proximity potentials generated by a particle . . . 23
  - 2.1.3 Abraham-Lorentz equation . . . 24
  - 2.1.4 Lorentz-Abraham-Dirac (LAD) equation . . . 25
  - 2.1.5 Landau - Lifshitz equation . . . 26
- 2.2 The quantum parameter . . . 27
- 2.3 Quantum correction and stochasticity . . . 28
- 2.4 Multiphoton Breit-Wheeler pair creation . . . 31

3 Modelling of the radiation reaction **33**
3.1 The classical regime 33
3.2 Intermediate, semi-quantum regime 34
3.3 Quantum regime 35
3.4 Stochastic scheme of the Breit-Wheeler pair creation 36
3.5 Choosing the model for simulation 37

4 Results and discussion **39**
4.1 Methods: Physics 39
4.1.1 1D case 39
4.1.2 2D case 40
4.1.3 Breit-Wheeler pair creation 40
4.2 Results of 1D simulations 41
4.3 Results of 2D simulations 49
4.4 Performance benchmarks 55

Conclusion **57**

Appendices **59**

A Boris pusher algorithm **61**

B Normalized units used in Smilei **63**

Introduction

The motion of particles in the ultrarelativistic regime under the presence of ultra high intensity lasers or other electromagnetic fields can be severely affected by high-energy photon emission due to the effects of radiation reaction [1]. This phenomena has received immense interest from the laser-plasma and astrophysics communities. For the laser-plasma community, this interest is currently driven by the development of multipetawatt laser facilities, such as ELI [2] or APOLLON [3]. These facilities have been expected to generate light pulses with peak power up to 10 PW with femtosecond duration. Moreover, intensities around 10^{23} W/cm^2 have been reached [4], leading to a new regime of relativistic laser-matter interaction that involves the quantum electrodynamics (QED) effects [5]. High-energy photon emission [6] and electron-positron pair production [7] are among the QED processes that have received significant attention, with some of these processes observed in recent laser-plasma experiments [8, 9]. This area of study is at the forefront of various proposals for experiments on forthcoming multipetawatt laser facilities. Radiation reaction has also been demonstrated to be crucial in various scenarios related to relativistic astrophysics, including interpretation and modeling of gamma-ray flares in the Crab Nebula [10], and pulsars [11].

In the so-called classical regime, where the energy of the emitted photons is small compared to that of the emitting electron, radiation reaction can be treated as a continuous friction force acting on the particles [12]. However, in the quantum regime, photons with energies comparable to that of the emitting electrons can be produced, leading to the QED effects [5]. Various theoretical works have been motivated by recent developments in both classical [13] and QED electrodynamics [14] regarding the treatment of radiation reaction. As a result, several numerical algorithms were proposed to deal with the radiation reaction effects. This includes using a radiation friction force [1], Fokker-Planck expansion of the collision operator [15] or a Monte Carlo procedure [16] to account for the stochastic quantum process of high-energy photon emission. These developments have been implemented in various kinetic simulation codes, particularly in particle-in-cell (PIC) codes [17, 18].

To ensure accurate PIC simulations in the classical and QED radiation reaction regimes, it is necessary to incorporate these radiation effects in the equations of motion for the particles. While various models for radiation reaction have been proposed in the literature, there is currently no universally accepted standard choice for implementing it in PIC codes [1]. In this thesis, we tested numerical approaches implemented in PIC codes used in the classical and quantum radiation reaction regimes. We used two open-source codes: Smilei [19] and EPOCH [17]. This comparative study was performed for intensities in classical, semi-quantum (transition) and full quantum regimes. Furthermore, we compared the implementation of the Breit-Wheeler pair creation in both of the PIC codes.

This thesis is structured as follows. In Chapter 1, the core of the PIC algorithm is described, covering topics such as macro-particles, initialization of the simulation, PIC loop, interpolation of electric and magnetic fields, particle pusher, current deposition, and the Maxwell solver. Also, an overview of some popular PIC codes is presented, along with a short description of grid computing and Metacentrum computer grid. In Chapter 2, the theory of radiation reaction is described in detail, including the Lorentz-Abraham-Dirac equation, Landau-Lifshitz description of radiation reaction, quantum correction, QED regime and multiphoton Breit-Wheeler pair creation. In Chapter 3, the algorithms for classical, semi-quantum and quantum radiation reaction regimes implemented in Smilei and EPOCH PIC codes are outlined. In Chapter 4, results from our comparative study in 1D and 2D geometries are analyzed. The differences in the results are discussed and compared for different radiation reaction regimes. Lastly, the performance benchmarks are presented.

Chapter 1

The Particle-In-Cell method for collisionless plasmas

The theoretical study of plasma is very challenging due to its many-particle nature. Analytical solutions can be found only for limited number of simplified cases. Therefore, it is necessary to use numerical simulations or modeling to investigate the behaviour of real plasma. There are generally two main approaches to plasma simulations. The first one is magneto-hydrodynamics or hydrodynamics approach, where plasma is treated as electrically conducting fluid. The second method is the use of interacting particle approach and kinetic theory, such as Particle-In-Cell (PIC) codes, where plasma is treated as an ensemble of charged particles. Furthermore, there are also hybrid codes allowing the use of the combination of the two methods.

The PIC method was initially developed for the study of fluid dynamics [20]. Today, it is widely used tool in various fields of physics, including plasma physics, accelerator physics and astrophysics. These codes are designed to model the behavior of charged particles in a plasma, where the particle motion is influenced by electromagnetic fields.

PIC codes are highly versatile and can be used to study a wide range of phenomena, including plasma turbulence, wave-particle interactions, plasma heating, and acceleration of charged particles. In accelerator physics, PIC codes are used to model the behavior of charged particles in accelerator components such as radiofrequency cavities. In plasma physics, they can be used to study the interaction of laser light with plasma [21, 22] and also the behaviour of plasma in tokamaks [23]. They are also commonly used to model astrophysics processes, such as the behavior of the solar wind and the interaction of the Earth's magnetic field with the solar wind. In these applications, the PIC code is used to model the behavior of the charged particles in the plasma and the electromagnetic fields that are generated in response to the plasma motion [24, 25].

PIC codes are typically run on high-performance computing systems, as the simulations can be computationally demanding, especially for large simulation domains or for simulations with high temporal and spatial resolution. However, the development of efficient algorithms and the use of parallel computing have made

PIC simulations increasingly accessible and useful for a wide range of applications, as the simulation of plasma in various environments, from basic laboratory setup to astrophysics [26].

Despite its advantages, the PIC models have several weaknesses. One of the main weaknesses is its computational efficiency [27]. Other weaknesses include difficulties in resolving the tail of the distribution and modeling large ranges of timescales and space scales. Additionally, the PIC method requires significant memory and processor resources, and this is unlikely to change in the near future.

In this chapter, the general PIC algorithm is described. Various PIC codes use different algorithms and computational schemes, but the core principle is the same. Particular algorithms introduced in this chapter are implemented both in Smilei [19] and in EPOCH [17] PIC codes, which are compared in this work (see Chapter 3 and 4). Understanding the basic algorithms behind the code is useful not only for a developer but also for a user, since the plasma simulations require a lot of parameters to be taken into account for meaningful results. In the last section, several frequently used PIC codes are described.

1.1 The Maxwell-Vlasov model

The collisionless plasma can be described by the so-called Vlasov-Maxwell system of equations. This description was first proposed by A. Vlasov in 1937. [28] This system describes the time evolution of the particle distribution function $f_\alpha(t, \mathbf{x}, \mathbf{p})$ in the phase space. Here, the α denotes given species of particles with charge q_α and mass m_α. The $\mathbf{x}$ and $\mathbf{p}$ denote the position and momentum of a phase space element, respectively. Distribution function includes the physical information of the system and their time evolution. It has up to 7 independent variables, 3 spatial, 3 for velocity and 1 for time. The evolution of the distribution function in this model is based on the collisionless Boltzmann equation [19] (often called Vlasov equation) in the form

$$\left(\partial_t + \frac{\mathbf{p}}{m_\alpha \gamma} \cdot \nabla + \mathbf{F}_L \cdot \nabla_{\mathbf{p}}\right) f_\alpha(t, \mathbf{x}, \mathbf{p}) = 0, \tag{1.1}$$

where $\gamma = \sqrt{1 + \mathbf{p}^2/(m_\alpha c)^2}$ is the relativistic Lorentz factor, c is the speed of light in vacuum, $\mathbf{F}_L = q_\alpha(\mathbf{E} + \mathbf{v} \times \mathbf{B})$ is the Lorentz force acting on a particle with velocity

$$\mathbf{v} = \frac{\mathbf{p}}{m_\alpha \gamma}, \tag{1.2}$$

and the operators are defined as

$$\nabla = \left(\frac{\partial}{\partial x}, \frac{\partial}{\partial y}, \frac{\partial}{\partial z}\right), \quad \nabla_{\mathbf{p}} = \left(\frac{\partial}{\partial p_x}, \frac{\partial}{\partial p_y}, \frac{\partial}{\partial p_z}\right).$$

Charged particles contained in plasma generate collective electric $\mathbf{E}(t, \mathbf{x})$ and magnetic $\mathbf{B}(t, \mathbf{x})$ fields. These fields in return act on the charged particles via the Lorentz force and satisfy the set of the Maxwell's equations

$$\nabla \cdot \mathbf{E} = \frac{\rho}{\epsilon_0}, \tag{1.3}$$
$$\nabla \cdot \mathbf{B} = 0, \tag{1.4}$$
$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t}, \tag{1.5}$$
$$\nabla \times \mathbf{B} = \mu_0 \mathbf{j} + \mu_0 \epsilon_0 \frac{\partial \mathbf{E}}{\partial t}, \tag{1.6}$$

where ϵ_0 and μ_0 are vacuum permittivity and permeability, respectively. Particles modify the collective electric and magnetic fields via their charge and current densities calculated from their distribution functions as

$$\rho(t, \mathbf{x}) = \sum_\alpha q_\alpha \int f_\alpha(t, \mathbf{x}, \mathbf{p}) \mathrm{d}^3\mathrm{p}, \tag{1.7}$$
$$\mathbf{j}(t, \mathbf{x}) = \sum_\alpha q_\alpha \int \mathbf{v} f_\alpha(t, \mathbf{x}, \mathbf{p}) \mathrm{d}^3\mathrm{p}, \tag{1.8}$$

respectively. The equations (1.1), (1.2) and (1.3) - (1.8) form the Maxwell-Vlasov system of equations.

We can now take the divergence of Maxwell-Ampère's equation (1.6)

$$\nabla \cdot (\mu_0 \epsilon_0 \partial_t \mathbf{E} + \mu_0 \mathbf{j} = \nabla \times \mathbf{B}) \implies \epsilon_0 \partial_t \nabla \cdot \mathbf{E} + \nabla \cdot \mathbf{j} = 0. \tag{1.9}$$

Assuming the charge conservation with the continuity equation

$$\partial_t \rho + \nabla \cdot \mathbf{j} = 0, \tag{1.10}$$

we get the relation

$$\partial_t (\epsilon_0 \nabla \cdot \mathbf{E} - \rho) = 0. \tag{1.11}$$

This means that if the electrostatic Poisson and Gauss equation are satisfied at time $t = 0$ (during the initialization of the simulation), and if our algorithm conserves charge during current deposition, then solving only the Maxwell-Ampère's equation (1.6) ensures that equation (1.3) is satisfied at later time.

We can also take the divergence of Maxwell-Faraday equation (1.5)

$$\frac{\partial (\nabla \cdot \mathbf{B})}{\partial t} = \nabla \cdot \frac{\partial \mathbf{B}}{\partial t} = \nabla \cdot (-\nabla \times \mathbf{E}) = 0. \tag{1.12}$$

Again, if we make sure that the equation (1.4) is satisfied at $t = 0$, then solving Maxwell-Faraday equation (1.5) ensures that the equation (1.4) is satisfied at later time. Both divergence equations act as boundary conditions and are not used during the PIC loop. However, these boundary condition must be included, since theoretical studies [29, 30] and practical experience shown that ignoring Poisson equation leads to incorrect numerical solutions.

1.2 Macro-particles

In the ideal scenario, supercomputers would be able to compute the position and momentum of each particle in a simulation. However, current computers are far from able to compute the systems with electron densities in range of $\approx 10^{14}$ to 10^{20} cm^{-3}, that are typically simulated in plasma physics [31]. One possible simplification can be derived from a fact that we usually care about collective behaviour of the plasma, rather than the behaviour of individual particles. Thus, we can represent a phase space of all particles by a sum of so-called finite size particles, often referred to as macro-particles. A particular set of physical particles is represented by a finite size macro-particle, rather than a point size individual particle. By doing so, we neglect some short range forces, but it allows us to use a much smaller number of particles and obtain results showing the collective plasma phenomena we are interested in. These macro-particles represent a solution of the Maxwell-Vlasov set of equations. In PIC codes, Vlasov's equation 1.1 is integrated along the continuous trajectories of the macro-particles, while Maxwell's equations are solved on a discrete spatial grid in order to simplify and minimize computational demands. The elements of the grid are called cells.

Direct integration of the equation (1.1), used in the so-called Vlasov codes, requires extreme computational resources. The key idea in PIC simulation is to express the distribution function $f_\alpha(t, \mathbf{x}, \mathbf{p})$ as a discrete sum of N_α macro-particles of particle species α (sometimes also referred to as "quasi-particles")

$$f_\alpha(t, \mathbf{x}, \mathbf{p}) = \sum_{p=1}^{N_\alpha} w_p S(\mathbf{x} - \mathbf{x}_p(t)) S_m(\mathbf{p} - \mathbf{p}_p(t)). \tag{1.13}$$

The w_p stands for a weight of the macro-particle, that will be discussed in detail in next sections (Sec. 1.4). The $S(\mathbf{x})$ and $S_m(\mathbf{p})$ are the macro-particle's shape functions in position and momentum space, respectively. These shape functions have several properties, namely, they describe a small portion of the phase space and have zero value outside of this small range, they are symmetric and the integral over their respective domains is unitary [32], e.g. $\int S(\mathbf{x} - \mathbf{x}_p) \mathrm{d}^3\mathrm{x} = 1$, where $\int dx$ represents integral over space. Physically, one macro-particle represents a collective behaviour of a set of particles. For a momentum shape function, the choice is always the Dirac function

$$S_m(\mathbf{p} - \mathbf{p}_p(t)) = \delta(\mathbf{p} - \mathbf{p}_p(t)). \tag{1.14}$$

This choice ensures that the macro-particle has only one value of velocity so it will not expand in the phase space. For the position shape function, early PIC codes also used the Dirac function [32]. Currently, most PIC codes [19, 17, 33] use piece-wise functions, so-called b-splines. For example, in one dimension, the first order b-spline $b_1(x)$ is often defined as the top-hat function

$$b_1(x) = \begin{cases} 1 & \text{if } |x| \leq \frac{1}{2}\Delta x \\ 0 & \text{otherwise,} \end{cases} \tag{1.15}$$

where Δx is the length of a cell. This function occupies two cells during the interpolation phase. This simple b-spline is not used in practice, because it leads to a high level of noise [34]. Higher order spline interpolation is then required. The most often used definition is the "triangle" shape function, also called second order b-spline, defined as

$$b_2(x) = \begin{cases} 1 - |x|/\Delta x & \text{if } |x| \leq \Delta x \\ 0 & \text{otherwise.} \end{cases} \tag{1.16}$$

This function occupies three cells during the interpolation phase. Third order b-spline is defined as

$$b_3(x) = \begin{cases} \frac{3}{4}\left(1 - \frac{4}{3}\left(\frac{x}{\Delta x}\right)^2\right) & \text{if } |x| \leq \Delta x/2 \\ \frac{9}{8}\left(1 - \frac{2}{3}\frac{|x|}{\Delta x}\right)^2 & \text{if } \Delta x/2 < |x| \leq 3\Delta x/2 \\ 0 & \text{otherwise.} \end{cases} \tag{1.17}$$

All three functions, are shown in Fig. 1.1.

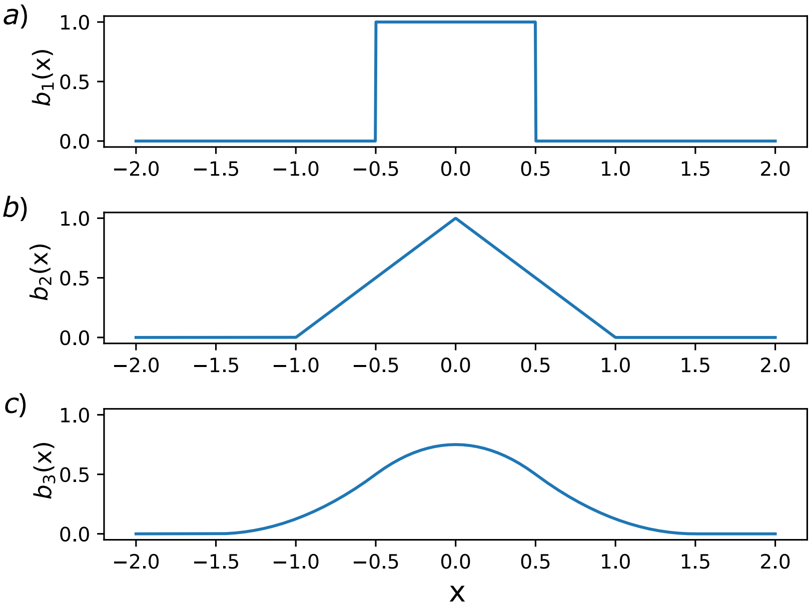


Figure 1.1: The comparison of the a) first $b_1(x)$ b) second $b_2(x)$ and c) third $b_3(x)$ order b-splines. The cell length was assumed to be $\Delta x = 1$.

1.3 Mathematical definition of the macro-particle shape function in higher dimensions

This section describes the macro-particle shape function implemented in Smilei [19]. Nonetheless, this definition is also used in other PIC codes, including EPOCH [17]. We need the macro-particle shape function $S(\mathbf{x})$ with several properties: symmetric with respect to its argument $\mathbf{x}$, non-zero in a center region around $\mathbf{x} = 0$, zero outside of this region and normalized to 1 ($\int d\mathbf{x} S(\mathbf{x}) = 1$) [19]. This particle covers macro-particle volume of $V_p = \Pi_\mu n \Delta x^\mu$, where n is the interpolation order and $\Delta x^\mu = (\Delta x, \Delta y, \Delta z)$ is the size of a cell in μ direction.

We will consider following definition

$$S(\mathbf{x}) = \Pi_{i=1}^{D} \bar{b}_n(x^\mu), \tag{1.18}$$

where D is the simulation dimension and n denotes interpolation order. The one dimensional shape function $\bar{b}_n(x)$ is defined recursively, with the zeroth order as $\bar{b}_0(x) = \delta(x)$, where $\delta(x)$ is the Dirac function. The recursive relationship for higher order functions is

$$\bar{b}_n(x) = \frac{1}{\Delta x}(P(x) * \bar{b}_{n-1}(x)) = \frac{1}{\Delta x}\int_{-\infty}^{\infty} dx' P(x' - x)\bar{b}_{n-1}(x'), \tag{1.19}$$

where the crenel function is defined as

$$P(x) = \begin{cases} 1 & \text{if } |x| \leq \Delta x/2 \\ 0 & \text{otherwise.} \end{cases} \tag{1.20}$$

For $b_n(x) = \Delta x \bar{b}_n(x)$, we get

$$b_0(x) = \Delta x \delta(x). \tag{1.21}$$

Higher order function were already defined in Eq. (1.15) - (1.17).

1.4 Initialization of the simulation

The initialization of the simulation consist of 4 steps:

1. Loading all the particles and computing the macro-particles weight.
2. Calculate the charge and current densities onto the grid.
3. Compute the initial electric and magnetic field.
4. Add any external fields.

For the first step, the user can define multiple simulation parameters for each particle species. This includes spatial profile for the number density n_s, the number of macro-particles per cell N_s, the mean particle velocity $\mathbf{v}_s$ and the mean temperature T_s. Usually, multiple temperature distributions are available, including zero-temperature, Maxwell or Maxwell-Jüttner (Relativistic Maxwell distribution) distribution. After defining these parameters, the particle loading then consist of creating N_s macro-particles in each cell with position $\mathbf{x}_p$ (in regular intervals within the cell or randomly chosen) and with momentum $\mathbf{p}_p$ randomly chosen from requested temperature distribution. What remains is to calculate the particle weight w_p. This is done for each macro-particle and depends on the user defined number density n_s as

$$w_p = \frac{n_s(\mathbf{x}_p(t=0))}{N_s}. \tag{1.22}$$

This parameter can be interpreted as the number of particles that each individual macro-particle represents.

The next step consist of computing the charge and current densities onto the grid for $t = 0$. The charge density $\rho(t = 0, \mathbf{x})$ is computed as [19]

$$\rho(t = 0, \mathbf{x}) = \sum_s q_s \sum_p w_p \int d\mathbf{x} S(\mathbf{x} - \mathbf{x}_p(t = 0)) P_D(\mathbf{x} - \mathbf{x}_i), \tag{1.23}$$

where $P_D(\mathbf{x}) = \Pi_{\mu=1}^{D} P(x^\mu)$, D is the number of spatial dimensions of the simulation and $P(x)$ is the crenel function (1.20) and $S(\mathbf{x})$ the macro-particle shape function (1.18).

The next step is to compute the initial electric and magnetic fields from initial charge and current densities. This is done by solving the Poisson equation. In the last step, we can add any external (divergence-free) electric or magnetic fields, such as the laser fields. Subsequently, the PIC loop can be started. These steps are very similar in both Smilei and EPOCH codes [17, 19].

1.5 The PIC loop

With the mathematical representation of macro-particles and computed initial electric and magnetic fields, we can proceed to the PIC loop. This loop is then repeated N times, where N is either pre-defined number or the codes compute until certain conditions are fulfilled. The core computational cycle of the PIC codes consists of four main steps:

1. Interpolating the electromagnetic fields acting on the macro-particles onto a grid.
2. Computing the new macro-particle velocities and positions.
3. Projecting the new charge and current densities on the grid positions.
4. Computing the new electromagnetic fields on the grid.

These steps are displayed in Fig. 1.2. In the next sections, we will describe each of these four steps in more detail, advancing from the step n to step $n + 1$.

1.6 Interpolation of the electric and magnetic fields

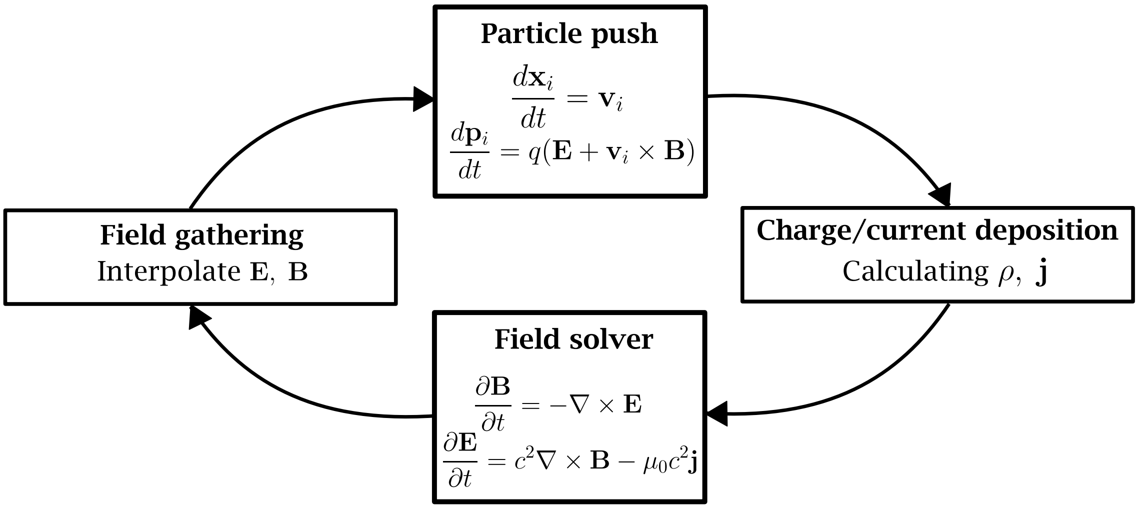
As a first step, we have to calculate the projection of fields from grid points onto the particles. This is referred to as the interpolation of the fields. In one dimensional grid, the field F_p acting on the macro-particle with center at the position x_p can be written in the form

$$F_p = \int dx \bar{b}_n(x - x_p) F(x), \tag{1.24}$$

where $\bar{b}_n(x)$ is one dimensional shape function (1.19) and $F(x)$ can be reconstructed as

$$F(x) = \sum_i F_i P(x - x_i), \tag{1.25}$$

where i denotes the grid point index, F_i the field value at i-th grid point, x_i the location of i-th grid point and $P(x)$ is crenel function (1.20). Visual representation of this process is depicted in Fig. 1.3.



Particle push
$\frac{d\mathbf{x}_i}{dt} = \mathbf{v}_i$
$\frac{d\mathbf{p}_i}{dt} = q(\mathbf{E} + \mathbf{v}_i \times \mathbf{B})$
Charge/current deposition
Calculating ρ, $\mathbf{j}$
Field solver
$\frac{\partial \mathbf{B}}{\partial t} = -\nabla \times \mathbf{E}$
$\frac{\partial \mathbf{E}}{\partial t} = c^2 \nabla \times \mathbf{B} - \mu_0 c^2 \mathbf{j}$
Field gathering
Interpolate $\mathbf{E}$, $\mathbf{B}$

Figure 1.2: The visual representation of PIC loop. We can start by interpolating the values of electromagnetic field values at the macro-particle position. The next step is the particle push, which upgrades the macro-particle position and momentum. Then these upgraded phase space coordinates are used for the computation of charge/current densities on a grid. The Maxwell equations are solved on the grid points in the last step. The loop then starts over.

This can be easily generalized to higher dimensions. Generally, the final set of equations has a form

$$\mathbf{E}_p^{(n)} = \int d\mathbf{x} S(\mathbf{x} - \mathbf{x}_p^{(n)}) \mathbf{E}^{(n)}(\mathbf{x}) \tag{1.26}$$

$$\mathbf{B}_p^{(n)} = \int d\mathbf{x} S(\mathbf{x} - \mathbf{x}_p^{(n)}) \mathbf{B}^{(n)}(\mathbf{x}), \tag{1.27}$$

where (n) denotes the time step and $S(x)$ is the shape function of the macro-particle (1.18). As will be discussed later, the leap frog scheme for advancing fields and macro-particles in time will make the use of a "half-time" step for the magnetic fields $\mathbf{B}^{(n\pm1/2)}(\mathbf{x})$. Effectively, the magnetic field is calculated as $\mathbf{B}^{(n)}(\mathbf{x}) = \frac{1}{2}[\mathbf{B}^{(n+1/2)}(\mathbf{x}) + \mathbf{B}^{(n-1/2)}(\mathbf{x})]$.

1.7 The particle pusher

When we compute the electromagnetic field for each macro-particle, we need to compute the new momentum and position of that macro-particle. There are several possible methods for solving Partial Differential Equations (PDE) by numerical calculations. These include finite difference, finite element and spectral method [35]. PIC codes are using Finite Difference Method (FDM) [36].

We have to solve a set of two equations of motion. The first one can be derived from putting the macro-particle distribution (1.13) into the Vlasov's equation (1.1)

$$\frac{d(\gamma m_p \mathbf{v})}{dt} = q_p(\mathbf{E} + \mathbf{v} \times \mathbf{B}). \tag{1.28}$$

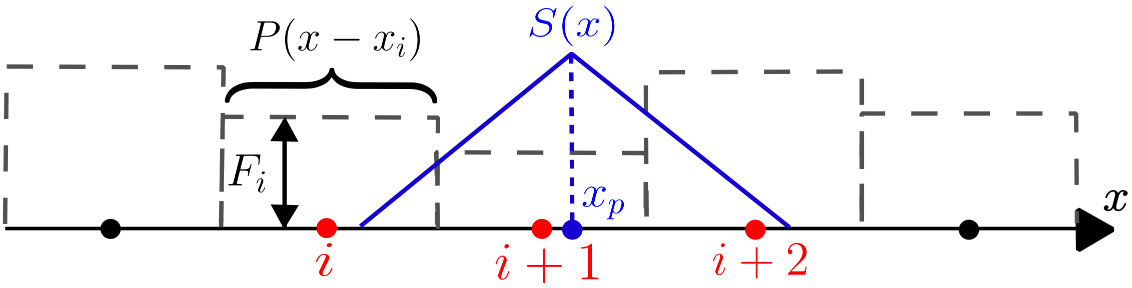


Figure 1.3: The visual representation of field interpolation with the macro-particle shape function $S(x)=\bar{b}_2(x)$. As you can see, this is the shape function of the 2nd order, which uses 3 field values F_i, representing the field's value at the i-th grid position. The $P(x-x_i)$ represents the crenel function for the i-th grid position.

The second one is just the definition of the velocity used to advance the particles

$$\frac{d\mathbf{x}}{dt}=\mathbf{v}, \tag{1.29}$$

where γ is the Lorentz factor. By introducing the normalized velocity $\mathbf{u}=\gamma\mathbf{v}$, we can write the Lorentz factor as $\gamma=\sqrt{1+\mathbf{u}^2/c^2}$.

We want to compute the evolution of the particle in time, evolving with the time-step Δt. Most PIC codes [19, 17, 33] use the second-order method with centered discretization, known as leap-frog method. This method is often implemented in both particle pusher and Maxwell's solver steps. Here, the particle velocity is calculated at half-step intervals, similar to the magnetic field $\mathbf{B}$ in Maxwell's solver. The equations (1.28) and (1.29) can be discretized in the leap-frog method as

$$\frac{\mathbf{u}_p^{(n+1/2)}-\mathbf{u}_p^{(n-1/2)}}{\Delta t}=\frac{q_p}{m_p}\left[\mathbf{E}_p^{(n)}+\frac{\mathbf{u}_p^{(n+1/2)}+\mathbf{u}_p^{(n-1/2)}}{2\gamma^{(n)}}\times\mathbf{B}_p^{(n)}\right], \tag{1.30}$$

$$\frac{\mathbf{x}^{(n+1)}-\mathbf{x}^{(n)}}{\Delta t}=\frac{\mathbf{u}^{(n+1/2)}}{\gamma^{(n+1/2)}}, \tag{1.31}$$

where q_p, m_p are charge and mass of the macro-particle and Lorentz factor $\gamma^{(n)}$ is taken as an average from two half-step values

$$\gamma^{(n)}=\frac{\gamma^{(n+1/2)}+\gamma^{(n-1/2)}}{2}. \tag{1.32}$$

The process is shown at Fig. 1.4. This leap-frog scheme depends on old forces from the previous time-step n, as is typical for explicit solvers. This method is computationally fast and can be stable [37].

There are several methods available for computing the velocity $\mathbf{u}_p^{n+1/2}$ in (1.30). This can be solved either as three scalar equations, one for each component. However, more sophisticated approaches have been developed. The most used method in PIC codes is the Boris pusher [38]. The details of this algorithm are introduced in Appendix A. However, this algorithm introduces errors when calculating the orbits of relativistic particles in certain electromagnetic field configurations,

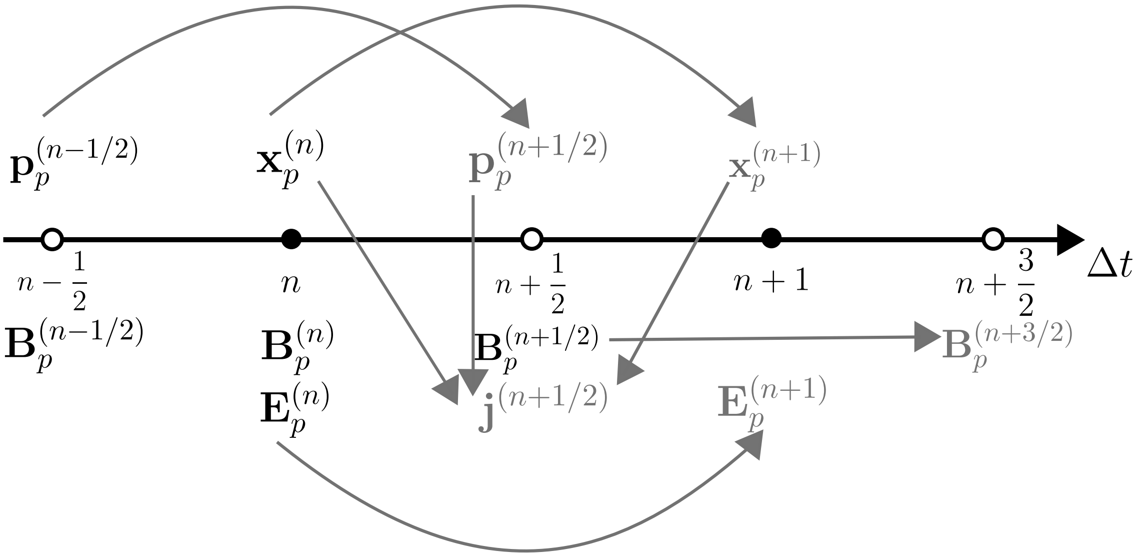


Figure 1.4: The visualization of the process of pushing particles. The horizontal axis shows the time-step index. The greyed-out symbols are values we want to determine during the particle push, current deposition and by Maxwell's solver from the time-step n to $n+1$. The values for electric $\mathbf{E}_p$ and magnetic $\mathbf{B}_p$ fields are also shown, along with the current deposition $\mathbf{j}$. The arrows point the evolution of the values in one PIC cycle.

e.g. when the electric and magnetic contributions cancel each other in the Lorentz force. In these scenarios, the Boris pusher introduces significant errors in particle trajectories [39]. Alternative methods for dealing with this problem were proposed, including Vay pusher [40], which deals with the force balance by splitting the integrator into the explicit first half and implicit second half. This leads to better handling of the $\mathbf{E}\times\mathbf{B}$ drift at relativistic speeds and is therefore interesting for laser physics and astrophysics. Another alternative method was proposed by Higuera and Cary [41]. This relativistic volume-preserving pusher employs Vay's characteristic velocity in the Lorentz-force part of the Boris pusher. It's numerically slower but gives the correct $\mathbf{E}\times\mathbf{B}$ velocity, giving more precise results in highly relativistic simulations. In Smilei Boris and Vay pushers are implemented. In EPOCH, Boris and Higuera & Cary pushers are implemented.

Lastly, we mention the numerical stability of the leap-frog scheme. The scheme is unstable for [42]

$$\omega_p \Delta t \geq 2, \tag{1.33}$$

where ω_p is the plasma frequency and Δt the time-step used in the simulation. In PIC codes, more strict condition $\omega_p \Delta t \geq 0.2$ is used for more accurate results [37].

1.8 The current deposition

The velocity and the the position of all macro-particles is now computed. For the evolution of the electromagnetic fields at the grid, we need to calculate the current $\mathbf{j}^{(n+1/2)}$ (at the half-step) and the charge density $\rho^{(n+1)}$ onto the grid. As discussed earlier, the continuity equation (1.10) needs to be satisfied, in order for the diverge terms in Maxwell's equations ((1.3), (1.4)) to be omitted. This means our current deposition scheme needs to satisfy the equation [43]

$$\frac{\rho^{(n+1)}_{i,j,k} - \rho^{(n+1)}_{i,j,k}}{\Delta t} + \frac{(j_x)^{(n+1/2)}_{i+1/2,j,k} - (j_x)^{(n+1/2)}_{i-1/2,j,k}}{\Delta x} +$$

$$\frac{(j_y)^{(n+1/2)}_{i,j+1/2,k} - (j_y)^{(n+1/2)}_{i,j-1/2,k}}{\Delta y} + \frac{(j_z)^{(n+1/2)}_{i,j,k+1/2} - (j_z)^{(n+1/2)}_{i,j,k-1/2}}{\Delta z} = 0. \tag{1.34}$$

Here, the subscript refers to the place on the so called Yee grid [44] in the Finite Difference Time Domain (FDTD) scheme, as will be discussed in the next section.

In 1992, Villasenor and Buneman [45] proposed a current deposition scheme that satisfies the continuity equation (1.34) rigorously (within numerical precision) in Cartesian coordinates. This is achieved by using the mentioned FDTD field solvers on Yee grid. In 2001, Esirkepov proposed a new charge conserving scheme [43] which generalized Villasenor and Buneman's method to an arbitrary particle form factor. In Smilei and EPOCH codes, the charge-conserving algorithm proposed by Esirkepov is implemented.

In the three dimensional case, the current densities in x, y, z direction from a particle with charge q_p are computed as

$$\begin{aligned}
(j_x)^{(n+1/2)}_{i+1/2,j,k} - (j_x)^{(n+1/2)}_{i-1/2,j,k} &= q_p w_p \frac{\Delta x}{\Delta t} (W_x)^{(n+1/2)}_{i+1/2,j,k} \\
(j_y)^{(n+1/2)}_{i,j+1/2,k} - (j_y)^{(n+1/2)}_{i,j-1/2,k} &= q_p w_p \frac{\Delta y}{\Delta t} (W_y)^{(n+1/2)}_{i,j+1/2,k} \\
(j_z)^{(n+1/2)}_{i,j,k+1/2} - (j_z)^{(n+1/2)}_{i,j,k-1/2} &= q_p w_p \frac{\Delta z}{\Delta t} (W_z)^{(n+1/2)}_{i,j,k+1/2},
\end{aligned} \tag{1.35}$$

where W_x, W_y, W_z are computed using the method developed by Esirkepov [43]. These values depend on the macro-particle present and former position $\mathbf{x}^{(n+1)}, \mathbf{x}^{(n)}$ and on the particle shape function. The computation of the charge density is redundant in our PIC cycle and is usually calculated only for the diagnostic reasons.

We have now calculated the current on the grid. With this, we can calculate the curl Maxwell's equations (1.5) and (1.6).

1.9 Maxwell's solver and the Yee grid

1.9.1 Yee grid

For solving the Maxwell equations, we need to define the positions of the electric and magnetic fields on the grid. A straightforward approach would be to locate all fields within the cell in the origin of the cell. This approach is referred to as unstaggered grid. In PIC simulation, the most commonly used configurations is the so called Yee grid proposed in 1966. On the Yee grid, the position of every field component is staggered in space, where the distance between them equals one half of the cell length ($\Delta x/2$). In this configuration, the magnetic fields are perpendicular to electric fields, which is useful for numerically calculating the curl operator in Maxwell's equations. The illustration of this grid is shown in Fig. 1.5. Note that **E** and **j** are located at the same grid position.

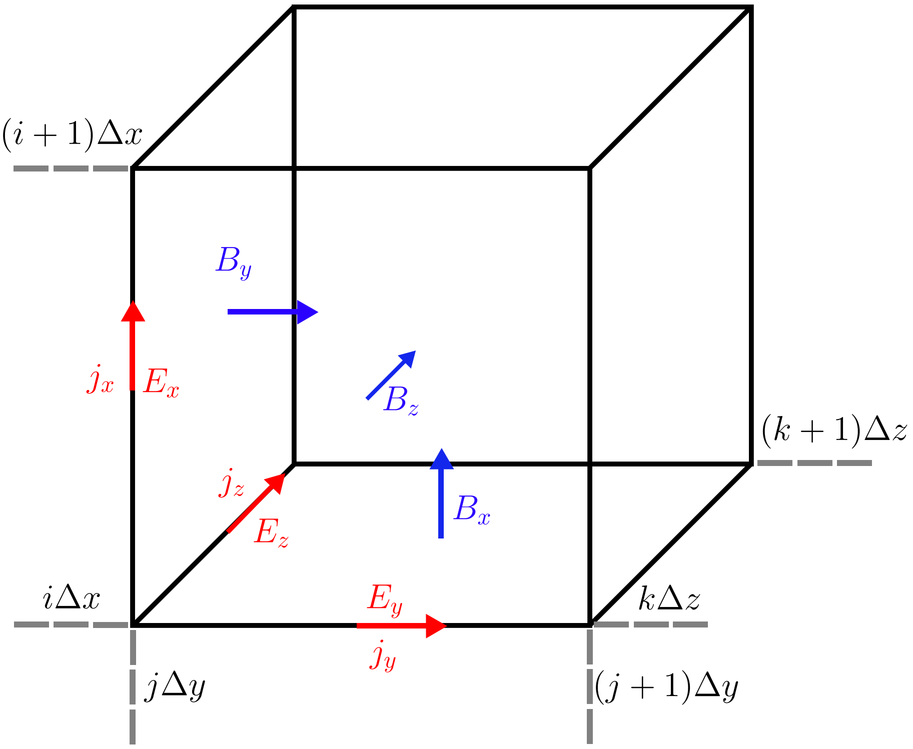


Figure 1.5: Representation of the 3D staggered Yee grid around the cell node (i, j, k).

1.9.2 Maxwell's solver

As mentioned before, for the computation of Maxwell's equations, for the current deposition algorithm that fulfills the charge conservation equation (1.10), we only need to consider the equations with curl operator (1.5) and (1.6). In PIC codes, including Smilei and EPOCH, these equations are usually solved by the FDTD technique. This technique was formulated by Yee [44] and has been applied extensively for solving EM equations [36]. In the leap-frog scheme, equations (1.5) and (1.6) can be written as

$$\mathbf{E}^{(n+1)} = \mathbf{E}^{(n)} + \Delta t\Big[c^2(\nabla\times\mathbf{B})^{(n+1/2)} - \frac{\mathbf{j}^{(n+1/2)}}{\varepsilon_0}\Big], \tag{1.36}$$

$$\mathbf{B}^{(n+3/2)} = \mathbf{B}^{(n+1/2)} - \Delta t(\nabla\times\mathbf{E})^{(n+1)}. \tag{1.37}$$

Now we only have to write the curl operator numerically on the Yee grid. The x-component of equations (1.36) and (1.37) can be discretized on the Yee grid as [19]

$$\frac{(E_x)^{(n+1)}_{i+1/2,j,k} - (E_x)^{(n)}_{i+1/2,j,k}}{\Delta t} = (j_x)^{(n+1/2)}_{i+1/2,j,k} + (\partial_y B_z)^{(n+1/2)}_{i+1/2,j,k} - (\partial_z B_y)^{(n+1/2)}_{i+1/2,j,k}, \tag{1.38}$$

$$\frac{(B_x)^{(n+3/2)}_{i,j+1/2,k+1/2} - (E_x)^{(n+1/2)}_{i,j+1/2,k+1/2}}{\Delta t} = (\partial_z E_y)^{(n+1)}_{i,j+1/2,k+1/2} - (\partial_y E_z)^{(n+1)}_{i,j+1/2,k+1/2}. \quad (1.39)$$

The equations are similar in the y and z direction. The discretization of the partial derivative operator acting on field F in the FDTD method reads

$$(\partial_x F)_{i,j,k} = \frac{F_{i+1/2,j,k} - F_{i-1/2,j,k}}{\Delta x}. \quad (1.40)$$

The Yee solver is known to cause numerical errors [46]. To overcome this issue, a number of alternative field solvers have been proposed, such as the M4 [47] and Lehe [48] solver.

1.9.3 CFL condition

Note that FDTD solvers are subjected to the Courant-Friedrich-Lewy (CFL) condition. [49] The CFL condition requires that the time-step Δt used in numerical simulations for solving a hyperbolic PDE (such as the wave equation) is small enough to ensure that the solution does not become unstable. Specifically, the CFL condition states that the time step size must be less than or equal to a certain multiple of the spatial grid cell size and the maximum velocity of the wave being modeled. This multiple factor is often referred to as the Courant number and is typically less than or equal to 1. In Smilei and EPOCH codes, the default value of the Courant number is 0.95.

In other words, the CFL condition ensures that information cannot propagate too far in a single time step, which is necessary for accurate and stable numerical simulations. Violating the CFL condition can result in numerical instabilities, such as oscillations and overshoots, that can make the solution meaningless. For the electromagnetic wave in plasma, this condition requires the time-step Δt to be smaller than (in 3D case)

$$\Delta t \leq \frac{C}{c\sqrt{\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2} + \frac{1}{\Delta z^2}}}. \quad (1.41)$$

where C is a dimensionless Courant number and $\Delta\mu$ is the cell length in the corresponding direction. In our case, we want to study the electromagnetic wave in a plasma, which cannot travel more than one cell per time step with a velocity of c.

1.10 Popular PIC codes

1.10.1 Smilei

Smilei [19] is an open-source, PIC code written in C++. It is developed mainly by Maison de la Simulation, Université Paris-Saclay, Laboratoire Leprince-Ringuet, École polytechnique and Sorbonne Universités, France. The input file and post-processing is handled via a user-friendly Python interface. It is highly flexible and can be easily extended with user-defined modules and scripts, allowing researchers to implement new physical models and algorithms. Smilei includes a number of built-in physics modules, including electrical field and current density filter, antennas, Monte-Carlo treatment of the binary collisions, reactions and field ionization, particle injection, radiation reaction and electron-positron pair creation. It can simulate a wide range of phenomena, including laser-plasma interactions (e.g. electron acceleration from intense femtosecond laser interaction with dense plasmas, short laser pulse amplification by stimulated Brillouin scattering), relativistic astrophysics (e.g. magnetic reconnection at the Earth magnetopause), and fusion energy.

1.10.2 EPOCH

EPOCH [17] is a fully-featured, open-source particle-in-cell code written in Fortran 90. It is developed by the Plasma Physics Group at the University of Warwick, UK. The postprocessing is handled through LLNL Visit software or through Python interface. It uses advanced numerical methods and modeling techniques, including adaptive mesh refinement and plasma radiation models. EPOCH includes a wide range of physics modules, including bremsstrahlung radiation, particle injectors, antennas, binary collision models and extensive quantum electrodynamics (QED) module, including radiation reaction and electron positron pair production. It can simulate a variety of phenomena, including Raman scattering, short-pulse, laser-solid interaction or fast-electron transport.

1.10.3 OSIRIS

OSIRIS [33] (Open Simulation Infrastructure for Research in Science) is a PIC simulation code used to model plasmas. It is developed by the Osiris Consortium, consisting of the Group of Lasers and Plasmas of the Instituto de Plasmas e Fusão Nuclear at IST, Lisbon, Portugal and the Plasma Theory and Simulation Group at the UCLA, Los Angeles, California, USA . It is written in Fortran 90 and C programming language. It is fully relativistic, massively parallel PIC code with dynamic load balancing. Several physics modules are implemented, including ADK tunneling/impact ionization, handling of binary collisions, radiation cooling and QED module.

1.10.4 PIConGPU

PIConGPU [50] is a PIC code designed to model plasma behavior. The code is optimized to run on graphics processing units (GPUs), which enables it to achieve high performance and fast simulation times. It is developed at the Helmholtz-Zentrum Dresden-Rossendorf (HZDR) in Germany. It is an open source software. The programming language used to implement PIConGPU is C++ with CUDA extensions. The code is designed to run on NVIDIA GPUs and is optimized to take full advantage of the parallel processing capabilities of these devices. CUDA allows developers to write code that can execute on the GPU, enabling high-performance parallel computing. One of the main advantages of using PIConGPU is its speed. By utilizing the computational power of GPUs, the code can simulate complex plasma behavior in a fraction of the time it would take using traditional CPU-based simulations. Physics models implemented range from classical radiation reaction, QED synchrotron radiation or advanced field ionization methods.

1.10.5 WarpX

WarpX [51] is an open-source PIC code used to simulate high-energy plasma physics. It is written in C++ and uses CUDA and OpenMP for parallelization to achieve high performance on both CPUs and GPUs. WarpX implements a fully relativistic approach to simulate plasma physics, which takes into account the relativistic effects of particle motion and electromagnetic fields. WarpX includes advanced features like adaptive mesh refinement, high-order time integration schemes, moving window boundary conditions, and particle tracking and diagnostics. ARTEMIS (Adaptive mesh Refinement Time-domain ElectrodynaMics Solver), a time-domain electrodynamics solver developed in CCSE is based ont the WarpX code.

1.11 Grid computing

The PIC simulations are often computationally demanding and therefore rely on the computational grids or supercomputers. Grid computing is a type of distributed computing that enables multiple computers to work together on a common task. It is a powerful solution for tackling computationally extensive problems that require significant processing power and memory resources. The concept of grid computing originated in the early 1990s [52] when researchers began to explore ways to harness the collective computing power of multiple machines connected by a network. Grid computing enables tasks that would be impossible for a single machine to handle to be divided into smaller sub-tasks and distributed across multiple machines. This makes it possible to complete complex tasks more quickly and efficiently.

The resources in grid computing include the individual computers that create a grid, as well as other resources such as storage devices. These resources are typically distributed across a wide geographic area and connected by a high-speed network. The resources available on the grid are typically managed by a control node or server, which monitors the available devices and allocates them to various tasks.

The middleware is the software that allows the individual computers to communicate with each other and with the control node. This typically includes tools for data transfer, resource discovery, security, and monitoring. It enables the individual computers to work together as a single, virtual machine.

The architecture of grid computing typically consists of a control node or server that manages the resources and coordinates the distribution of tasks across the network. The control node is responsible for job scheduling, resource allocation, and load balancing. The individual computers on the grid are typically referred to as nodes or clients and are responsible for executing the tasks assigned to them. The computational jobs are requested by the user from the so called front-ends. These are specific computers reserved primarily for user activity (preparing jobs, analyzing the data etc.).

On the other hand, supercomputers are typically centralized computing systems that consist of a single large-scale machine with a high processing speed and a large amount of memory and storage. They are designed to solve complex computational problems that require massive parallel processing capabilities. Supercomputers are expensive to build and maintain, and typically used by large research institutions or corporations.

One of its key features of PIC codes is the ability to efficiently perform large-scale simulations through the use of parallelization. To achieve this, PIC codes utilizes multi-processing application program interface, e.g. OpenMP and MPI. OpenMP is a programming interface that allows for shared-memory parallelism, while MPI (Message Passing Interface) is used for distributed-memory parallelism. Combining the two allows PIC codes to efficiently distribute the workload across multiple processors or nodes in a computational grid or supercomputer. This results in significant speedups and allows for the simulation of larger and more complex systems than would not be possible with a single processor. Some PIC codes, e.g. Smilei and EPOCH, also employs dynamic load balancing, which helps to evenly distribute the simulation workload among the MPI processes. This is done by periodically exchanging segments of the simulation box between processes, ensuring that each process has a fair amount of the overall simulation load. In this thesis, we are using the Czech National Grid Infrastructure Metacentrum.

1.11.1 Metacentrum

Metacentrum [53] is a computational grid infrastructure that provides high-performance computing resources to academic and research institutions in the Czech Republic. It was established in 2007 by a consortium of Czech universities and research institutions, and it has since grown to become the largest academic computing facility in the country. The main goal of Metacentrum is to support scientific research by providing access to advanced computing resources, such as clusters, grids. These resources are used to run simulations, model complex systems, and analyze large datasets. Metacentrum also provides storage solutions and data management services to support scientific workflows.

Metacentrum consists of multiple computing clusters located across different sites in the Czech Republic. The infrastructure is interconnected by high-speed networks, allowing users to access computing resources from any location. The system is also designed to be scalable, meaning that it can easily adapt to changing user needs and resource requirements. One of the key features of Metacentrum is its user-friendly interface, which allows researchers to easily submit and manage their computing jobs.

For this thesis, we utilized the so-called non-interactive batch jobs. Batch jobs are a common use case for grid computing, where the user prepares input and instructions at the beginning, and the calculation then runs independently on the user. The batch job process involves the user preparing data and instructions (input files + batch script), submitting them to a job planner (also called PBS server), which stages the job until resources are available. The job then runs on a computational node with the required applications loaded. When the job finishes, the results are copied back to the user's specified directory.

Chapter 2

The theory of radiation reaction

In this chapter, the radiation reaction in the classical, semi-quantum and quantum descriptions is discussed. In particular, the covariant form of the Lorentz force, retarded and advanced potentials generated by a moving charges, Abraham-Lorentz force and the covariant generalization of this equation as Lorentz-Dirac force are derived. The formulation and advantages of Landa-Lifshitz equation are described. The quantum correction and stochastic effects of radiation reaction are discussed. The Breit-Wheeler pair generation is also explained at the end of this chapter. This provides a theoretical basis for the implementation of the radiation reaction in various PIC codes.

2.1 Classical radiation reaction

2.1.1 Covariant form of the Lorentz force

The equation of motion for a charged particle can be derived from the standard relativistic Lagrangian

$$L = -mc^2\sqrt{1-\left(\frac{\dot{\mathbf{x}}}{c}\right)^2} + q\mathbf{A}(\mathbf{x})\cdot\dot{\mathbf{x}} - q\phi(\mathbf{x}), \tag{2.1}$$

where $\mathbf{x} = (x, y, z)$ is the position of the particle, $\dot{\mathbf{x}}$ is the derivation of the position with respect to time, $\mathbf{A}(\mathbf{x})$ is the magnetic vector potential and $\phi(\mathbf{x})$ is the electric potential. By using the Lagrange-Euler equation (so by minimizing the action) we can get the equation of motion in the form

$$\frac{d}{dt}(m_0\gamma\mathbf{v}) = q\mathbf{E} + q\mathbf{v}\times\mathbf{B}, \tag{2.2}$$

where γ is the Lorentz factor and m_0 the rest mass of the particle and velocity $\mathbf{v} = \dot{\mathbf{x}}$. For numerical applications, we can use the substitution $\mathbf{u} = \gamma\mathbf{v}$, similarly to (1.28).

For the derivation of Lorentz force in covariant form, we need to introduce the 4-velocity as

$$u^\alpha = \frac{dx^\alpha}{d\tau}, \tag{2.3}$$

where x^α denotes the space-time coordinates and τ is the proper time. By using the relationship between the proper and general space-time interval

$$-c^2 d\tau^2 = -c^2 dt^2 + d\mathbf{x}^2 \implies d\tau = \sqrt{1 - \mathbf{v}^2/c^2} dt = \frac{dt}{\gamma}, \tag{2.4}$$

we can derive the 4-velocity form as

$$u^\alpha = \frac{dx^\alpha}{d\tau} = \frac{dx^\alpha}{dt}\frac{dt}{d\tau} = \begin{pmatrix} \gamma c \\ \gamma \mathbf{v} \end{pmatrix}. \tag{2.5}$$

Also, the scalar product of the 4-velocity is the space-time invariant, since $u^\alpha u_\alpha = -c^2$. We can also define the 4-acceleration as

$$a^\alpha = \frac{du^\alpha}{d\tau}. \tag{2.6}$$

The 4-acceleration is perpendicular to velocity

$$\frac{d}{dt}(u^\alpha u_\alpha) = a^\alpha u_\alpha + u^\alpha a_\alpha = 2a^\alpha u_\alpha = 0 \implies a^\alpha u_\alpha = 0. \tag{2.7}$$

Finally, the 4-momentum can be defined as

$$p^\alpha = m_0 u^\alpha = \begin{pmatrix} m_0 \gamma c \\ m_0 \gamma \mathbf{v} \end{pmatrix}. \tag{2.8}$$

The time-part of 4-momentum must be in the units of energy and the space-part must be the classical momentum vector

$$\begin{pmatrix} m_0 \gamma c \\ m_0 \gamma \mathbf{v} \end{pmatrix} = \begin{pmatrix} \varepsilon / c \\ \mathbf{p} \end{pmatrix}. \tag{2.9}$$

This allows us to identify

$$\varepsilon = \gamma m_0 c^2 = mc^2, \tag{2.10}$$

$$\mathbf{p} = \gamma m_0 \mathbf{v} = m\mathbf{v}, \tag{2.11}$$

where $m = m_0 \gamma$ is the relativistic mass. At the end, we need to introduce the electromagnetic field tensor [54]

$$F^{\mu\nu} = \partial^\mu A^\nu - \partial^\nu A^\mu = \begin{pmatrix} 0 & E^x/c & E^y/c & E^z/c \\ -E^x/c & 0 & B^z & -B^y \\ -E^y/c & -B^z & 0 & B^x \\ -E^z/c & B^y & -B^x & 0 \end{pmatrix}, \tag{2.12}$$

which is antisymmetric tensor of the second order. By combining the 4-momentum (2.8) with (2.12), the equation (2.2) can be cast in the form of

$$\frac{dp^\alpha}{d\tau} = qF^{\alpha\beta} u_\beta. \tag{2.13}$$

By using the relation from (2.4) on the space components, the Lorentz equation of motion is recovered. For the time component, the equation for energy balance can be written as

$$\frac{d\varepsilon}{dt} = q\mathbf{E} \cdot \mathbf{v}. \tag{2.14}$$

This shows that the power from the electric field is "transformed" into particle energy and vice-versa.

2.1.2 Close proximity potentials generated by a particle

The time dependent solution for Maxwell's equations can be written in a form of the 4-potential [55]

$$A^{\alpha}_{\mp}(t,\mathbf{x}) = \frac{\mu_0}{4\pi}\int \frac{J^{\alpha}(t \mp \Delta t,\mathbf{x}')}{|\mathbf{x}-\mathbf{x}'|} d^3\mathbf{x}', \tag{2.15}$$

where 4-current is defined as

$$J^{\alpha}(t,\mathbf{x}) = \begin{pmatrix} \rho(t,\mathbf{x}) \\ \mathbf{j}(t,\mathbf{x}) \end{pmatrix}. \tag{2.16}$$

Here, ρ is the charge density and $\mathbf{j}$ the current density. The $A^{\alpha}_{-}(t,\mathbf{x})$ denotes the retarded potential generated by a charged particle, $A^{\alpha}_{+}(t,\mathbf{x})$ denotes the advanced potential, which is unphysical since it implies the violation of causality, as the potential is indicated "ahead" of time. The expression $\Delta t = |\mathbf{x}-\mathbf{x}'|/c$ is time of signal spreading from a point source.

In a close proximity to the charged particle, we can assume that $\Delta t << 1$. This allows to approximate the 4-current with the Taylor expansion in the following way

$$A^{\alpha}_{\mp}(t,\mathbf{x}) = \frac{\mu_0}{4\pi}\int \left(\sum_{k=0}^{\infty} \frac{(\mp\Delta t)^k}{k!}\frac{\partial^k J^{\alpha}(t,\mathbf{x}')}{\partial t^k}\right)\frac{d^3\mathbf{x}'}{|\mathbf{x}-\mathbf{x}'|}. \tag{2.17}$$

After using $\Delta t = |\mathbf{x}-\mathbf{x}'|/c$, we can get the following expression (we assume that the sum and integral can be exchanged)

$$A^{\alpha}_{\mp}(t,\mathbf{x}) = \frac{\mu_0}{4\pi}\sum_{k=0}^{\infty}\left[\frac{(\mp 1)^k}{c^k k!}\frac{\partial^k}{\partial t^k}\int |\mathbf{x}-\mathbf{x}'|^{k-1} J^{\alpha}(t,\mathbf{x}') d^3\mathbf{x}'\right]. \tag{2.18}$$

We can separate this expression into even and odd parts of the Taylor expansion

$$\begin{aligned} A^{\alpha}_{\mp}(t,\mathbf{x}) = \frac{\mu_0}{4\pi}\sum_{k=0,2,4,\dots}^{\infty}\left[\frac{(\mp 1)^k}{c^k k!}\frac{\partial^k}{\partial t^k}\int |\mathbf{x}-\mathbf{x}'|^{k-1} J^{\alpha}(t,\mathbf{x}') d^3\mathbf{x}'\right] \mp \\ \mp\frac{\mu_0}{4\pi}\sum_{k=1,3,5,\dots}^{\infty}\left[\frac{(\mp 1)^k}{c^k k!}\frac{\partial^k}{\partial t^k}\int |\mathbf{x}-\mathbf{x}'|^{k-1} J^{\alpha}(t,\mathbf{x}') d^3\mathbf{x}'\right]. \end{aligned} \tag{2.19}$$

The first part is the same for retarded and advanced potentials and is divergent for $k = 0$ on the particle worldline [1]. However, this divergence cancels out in the corresponding force calculations. It can, for example, correspond to a Coulomb field. The second part of the formula has a different sign, so the solution is different depending on the direction of the wave (in the direction of the particle or away from it). This part is non-divergent, the force acting on a particle is non-zero and corresponds to a reaction of the particle to the field generated by that particle. The retarded potential can formally be written in terms of symetric and antisymmetric parts

$$A^{\alpha} = A^{\alpha}_{ret} = \frac{1}{2}(A^{\alpha}_{sym} + A^{\alpha}_{ant}) = \frac{1}{2}(A^{\alpha}_{-} + A^{\alpha}_{+}) + \frac{1}{2}(A^{\alpha}_{-} - A^{\alpha}_{+}). \tag{2.20}$$

---

[1] Worldline is the path that an object traces in 4-dimensional spacetime.

In the symmetric part, all odd $k = 1, 3, ...$ members of the expansion are cancelled and only even parts remain

$$A^\alpha_{sym} = \frac{1}{2}(A^\alpha_- + A^\alpha_+) = \frac{\mu_0}{4\pi}\sum_{k=0,2,4,...}^{\infty}\left[\frac{1^k}{c^k k!}\frac{\partial^k}{\partial t^k}\int |\mathbf{x}-\mathbf{x}'|^{k-1}J^\alpha(t,\mathbf{x}')d^3\mathbf{x}'\right].$$

As argued before, this part do not act on the particle, as the contributions cancels out. On the other hand, for the antisymmetric part, only the odd members of the expansion remain

$$A^\alpha_{ant} = \frac{1}{2}(A^\alpha_- - A^\alpha_+) = -\frac{\mu_0}{4\pi}\sum_{k=1,3,5,...}^{\infty}\left[\frac{1}{c^k k!}\frac{\partial^k}{\partial t^k}\int |\mathbf{x}-\mathbf{x}'|^{k-1}J^\alpha(t,\mathbf{x}')d^3\mathbf{x}'\right].$$

This correspond to a self-field that acts on the particle.

We can find the analytical solution to this self-force in the local inertial frame (LIF). This can be done non-relativistically and only for the first non-zero part [56]. For the scalar potential ($\alpha = 0$), the $k = 1$ gives us the derivation of the total charge, which is zero. The first non-zero term is for $k = 3$ and reads

$$\frac{\varphi}{c} = -\frac{\mu_0}{4\pi}\frac{1}{c^3 3!}\frac{\partial^3}{\partial t^3}\int |\mathbf{x}-\mathbf{x}'|^2\rho(t,\mathbf{x}')c d^3\mathbf{x}'. \tag{2.21}$$

For the vector potential, the first non-zero term is for $k = 1$

$$\mathbf{A} = -\frac{\mu_0}{4\pi c}\frac{\partial}{\partial t}\int \mathbf{j}(t,\mathbf{x}')d^3\mathbf{x}'. \tag{2.22}$$

For the point-like particle with a position at $\mathbf{x}_0$

$$\rho(t,\mathbf{x}) = q\delta(\mathbf{x}-\mathbf{x}_0), \tag{2.23}$$

$$\mathbf{j}(t,\mathbf{x}) = q\mathbf{v}\delta(\mathbf{x}-\mathbf{x}_0). \tag{2.24}$$

After integration, the potentials have a form

$$\varphi = -\frac{\mu_0 q}{24\pi c}\frac{\partial^3}{\partial t^3}(\mathbf{x}-\mathbf{x}_0(t))^2, \tag{2.25}$$

$$\mathbf{A} = -\frac{\mu_0 q}{4\pi c}\frac{d}{dt}\mathbf{v}(t). \tag{2.26}$$

2.1.3 Abraham-Lorentz equation

We can now express the corresponding electric and magnetic fields

$$\mathbf{B} = \boldsymbol{\nabla}\times\mathbf{A} = 0, \tag{2.27}$$

$$\begin{aligned}\mathbf{E} = -\nabla\varphi - \frac{\partial\mathbf{A}}{\partial t} &= \frac{\mu_0 q}{24\pi c}\frac{\partial^3}{\partial t^3}\nabla(\mathbf{x}-\mathbf{x}_0(t))^2 + \frac{\partial}{\partial t}\frac{\mu_0 q}{4\pi c}\frac{d}{dt}\mathbf{v}(t) = \\ = \frac{\mu_0 q}{24\pi c}\frac{\partial^3}{\partial t^3}2(\mathbf{x}-\mathbf{x}_0(t)) + \frac{\mu_0 q}{4\pi c}\frac{d\mathbf{a}}{dt} &= -\frac{\mu_0 q}{12\pi c}\frac{d\mathbf{a}}{dt} + \frac{\mu_0 q}{4\pi c}\frac{d\mathbf{a}}{dt} = \frac{\mu_0 q}{6\pi c}\dot{\mathbf{a}}.\end{aligned} \tag{2.28}$$

Here, the $\mathbf{v}(t)$ corresponds to the particle velocity, $\mathbf{a}(t)$ to particle acceleration vector. The "self-force" force acting on the particle is then

$$\mathbf{F}_{rad} = q\mathbf{E} = \frac{\mu_0 q^2}{6\pi c}\dot{\mathbf{a}}. \tag{2.29}$$

This is often called radiation reaction force, or Abraham-Lorentz force, derived by M. Abraham in 1905 [57]. The equation of motion then has a form

$$m\mathbf{a} = \mathbf{F}_{ext} + \mathbf{F}_{rad} = \mathbf{F}_{ext} + \frac{\mu_0 q^2}{6\pi c}\dot{\mathbf{a}}, \tag{2.30}$$

where $\mathbf{F}_{ext}$ are the external fields acting on a particle. This can be written as

$$m(\ddot{\mathbf{x}} - \tau_0\,\dddot{\mathbf{x}}) = \mathbf{F}_{ext}, \tag{2.31}$$

where

$$\tau_0 = \frac{\mu_0 q^2}{6\pi m c}. \tag{2.32}$$

This equation has several problems. First, it requires the third integration constant in the solutions, because we have to deal with the third derivative of the position. This is usually redundant, since we need to know at least 3 parameters to find 3 integration constant. However, in classical physics, we usually deal with only the particle position and momentum as the boundary condition. Second, some solutions of this equation violate the causality. Third, it has the so-called "runaway" solution. This can be seen by setting $\mathbf{F}_{ext} = 0$ and by integrating the (2.31) two times. One possible solution is then

$$\mathbf{x} = \mathbf{x}_0 e^{t/\tau_0}. \tag{2.33}$$

This solution is not physical, since without the external force, it assumes that particles are exponentionaly accelerated. General solutions can be cast as

$$\mathbf{r}(t) = \mathbf{c}_0 e^{t/\tau_0} + \mathbf{c}_1 t + \mathbf{c}_2. \tag{2.34}$$

By choosing $\mathbf{c}_0 = 0$, we can remove the "runaway" solution.

2.1.4 Lorentz-Abraham-Dirac (LAD) equation

The equation (2.31) can be generalized to any coordinate system in the covariant form [58]

$$m\frac{du^\alpha}{d\tau} = F^\alpha_{ext} + m\tau_0\left(\frac{da^\alpha}{d\tau} - \frac{a^2}{c^2}u^\alpha\right), \tag{2.35}$$

where $a^2 = a^\alpha a_\alpha$. The additional second term in the bracket is there to ensure the equation is Lorentz covariant. We can see this by multiplying (2.35) by u^α and using $F^\alpha_{ext} = qF^{\alpha\beta}u_\beta$,

$$ma^\alpha u_\alpha = qF^{\alpha\beta}u_\beta u_\alpha + m\tau_0\left(\frac{da^\alpha}{d\tau}u_\alpha - \frac{a^2}{c^2}u^\alpha u_\alpha\right). \tag{2.36}$$

Using the (2.7), $u^\alpha u_\alpha = -c^2$ and the fact that the contraction of antisymmetric ($F^{\alpha\beta}$) and symmetric ($u_\beta u_\alpha$) tensor is zero, we get

$$0 = 0 + m\tau_0 \left(\frac{d(a^\alpha u_\alpha)}{d\tau} - a^\alpha a_\alpha - \frac{a^2}{c^2}(-c^2) \right) = m\tau_0 \left(0 - a^2 + a^2 \right) = 0. \tag{2.37}$$

The added term can be alternatively written as

$$a^2 u^\alpha = a^\beta a_\beta u^\alpha = a^\beta \frac{du_\beta}{d\tau} u^\alpha = \left(\frac{d(a^\beta u_\beta)}{d\tau} - \frac{da^\beta}{d\tau} u_\beta \right) u^\alpha = -\frac{da^\beta}{d\tau} u^\alpha u_\beta. \tag{2.38}$$

This leads to an alternative formulation of (2.35), where F_{rad} is directly proportional to the derivation of 4-acceleration, similar to (2.29)

$$m\frac{du^\alpha}{d\tau} = F^\alpha_{ext} + m\tau_0 \left(g^\alpha_\beta + \frac{u^\alpha u_\beta}{d\tau} \right) \frac{da^\beta}{d\tau} = F^\alpha_{ext} + F^\alpha_{rad}. \tag{2.39}$$

The steps of the LAD derivation were described in the same manner as in the textbook of Kulhanek [59].

2.1.5 Landau - Lifshitz equation

The most widely used classical theory for the treatment of radiation reaction was proposed by Landau and Lifshitz [12]. They considered the following two conditions:

1. the magnitude of the external field E is in the instantaneous rest frame much smaller than the magnitude of the Schwinger field $E_s = m_e^2 c^3/e\hbar$ [60, 61], divided by the fine structure constant $\alpha \approx 1/137$. The Schwinger (or critical) field is a threshold above which the electromagnetic field is expected to become nonlinear. In the perfect vacuum, it corresponds to the upper limit of laser field strength. This condition can be expressed as $E << E_s/\alpha$. This ensures that the recoil at the level of individual photon is negligible.
2. The characteristic length scale over which the external field varies L is in the instantaneous rest frame varies is much larger than the Compton length $\lambda_c = h/m_e c$ [62], where h is the Planck length. This can be expressed as $L >> \lambda_c$. This ensures that the electron wavefunction is well-localized.

Both of these conditions are automatically fulfilled in the realm of classical electrodynamics [5]. Landau and Lifshitz showed that under these conditions, the second (radiation reaction) term in (2.35) were much smaller than the first term in the instantaneous rest frame of the charge. This allows to reduce the order of LAD equation by substituting $a^\alpha = \frac{du^\alpha}{d\tau} \to -\frac{e}{m} F^{\mu\nu} u_\nu$. The result is called Landau-Lifshitz (LL) equation [12]

$$\frac{du^\mu}{d\tau} = -\frac{e}{m} F^{\mu\nu} u_\nu + \tau_0 \left(-\frac{m}{e} (\partial_\alpha F^{\mu\nu}) u_\nu u^\alpha + F^{\mu\nu} F^{\nu\alpha} u^\alpha + (F^{\nu\alpha} u_\alpha)^2 u^\mu \right). \tag{2.40}$$

This equation can be then used as the equation of motion in the electromagnetic field. It leads to a continuous loss of energy for the radiating particle. After determining the trajectories, the self-consistent radiation can be obtained from the Liénard-Wiechert potentials [63, 64]. These potentials provide information about the electric and magnetic fields of a charge in any type of motion. In the far field, the spectral intensity of radiation from a group of N_e electrons and the energy radiated per unit frequency ω and solid angle Ω can be calculated as [54]

$$\frac{d^2\varepsilon}{d\omega d\Omega} = \frac{\alpha\omega^2}{4\pi^2}\left|\sum_{k=1}^{N_e}\int \mathbf{n}\times(\mathbf{n}\times\mathbf{v}_k)e^{i\omega(t-\mathbf{n}\cdot\mathbf{r}_k)}dt\right|^2, \tag{2.41}$$

where $\mathbf{n}$ is the observation direction, $\mathbf{r}_k$ and $\mathbf{v}_k$ are the position and velocity of the k-th particle.

2.2 The quantum parameter

The dominance of classical or quantum processes in the laser-electron interaction is determined by the laser intensity, electron energy, and the angle of incidence relative to each other. An effective method of distinguishing between these two domains is by using a Lorentz-invariant dimensionless parameter χ, which is in the general electromagnetic field defined as [65]

$$\chi = \frac{\sqrt{-(F_{\mu\nu}p^\nu)^2}}{mE_S} = \frac{\gamma}{E_S}\sqrt{(\mathbf{E}+\mathbf{v}\times\mathbf{B})^2-(\mathbf{v}\cdot\mathbf{E})^2}, \tag{2.42}$$

where p is the particle 4-momentum (2.8), γ is the particle Lorentz factor and E_s is the Schwinger field. In terms of physical interpretation, the parameter χ denotes the ratio between the external electromagnetic field and E_S, observed from the rest frame of the particle. When $\chi << 1$, the particle interaction with the field is purely classical. However, when $\chi \geq 1$, the interaction is quantum-dominated, and a two-step Breit-Wheeler process is expected to generate an extensive quantity of electron-positron pairs [66].

Within a plane electromagnetic wave, where both $\mathbf{E}$ and $\mathbf{B}$ have equal magnitudes and are mutually perpendicular, quantum parameter has a form $\chi = \gamma|\mathbf{E}|(1-\cos\theta)/E_S$. Here, θ refers to the angle between the electron momentum and the wavevector of the electric field. In this case, χ reaches maximum when electromagnetic field counter-propagates with respect to the particle momentum. An interesting outcome of (2.42) is the presence of a radiation-free direction. For any $\mathbf{E},\mathbf{B}$ configuration, there exist a specific $\mathbf{v}$, for which the χ vanishes. Under the influence of a strong electromagnetic field, electrons have a tendency to orient themselves along this particular direction [67]. Any transverse momentum they possess is swiftly lost through radiation process.

For the laser-electron interactions, we will refer to χ as the electron quantum parameter. The optimal arrangement for achieving high values of χ involves scattering in which the laser and electron beam are oriented in opposite directions. A simplified version of (2.42) applies to an electron moving in the opposite direction to the laser beam and has a form [66]

$$\chi = \frac{2\gamma a_0\hbar\omega_0}{mc^2}, \tag{2.43}$$

where $\hbar$ is the reduced Planck constant, ω_0 the laser frequency, m the electron mass and a_0 is normalized vector potential of the laser. The normalized vector potential can be expressed as

$$a_0 = 0.85\lambda_0[\mu\mathrm{m}]\sqrt{I[10^{18}\mathrm{W}\cdot\mathrm{cm}^{-2}]}, \tag{2.44}$$

where λ_0 is the laser wavelength and I laser intensity. For a co-propagating particle, quantum parameter can be approximated as

$$\chi = \frac{a_0\hbar\omega_0}{2\gamma mc^2}, \tag{2.45}$$

and for the interaction at 90 degrees as

$$\chi = \frac{a_0\gamma\hbar\omega_0}{mc^2}. \tag{2.46}$$

2.3 Quantum correction and stochasticity

Reaching the regime where radiation reaction becomes significant also implies the importance of quantum effects. These regimes are of great interest for the new generation of multi-petawatt laser systems, since at the expected intensities$> 10^{23}$ Wcm^{-2}, the radiation reaction will be comparable in magnitude to the Lorentz force, rather than being a small correction, as is familiar from synchrotrons [68]. In both classical and quantum description of radiation reaction, the direction of the radiation reaction force is opposite to the electron instantaneous momentum, and its strength is determined by the χ parameter. As the value of χ increases, the differences between the classical and quantum models of radiation reaction are more significant. In the classical approach, there is no upper limit on the frequency range, while the quantum theory has a cutoff that ensures $\omega < \gamma m$.

First, let's consider the radiation emission of an ultra-relativistic electron in any external field within the framework of classical electrodynamics. When an electron moves at a very high speed, its radiation emission is almost equivalent to that of an electron moving instantaneously in a circular path, regardless of its original motion. This is commonly referred to as synchrotron radiation emission and can be used as a good approximation. The power distribution of the emitted radiation with respect to the frequency ω of the photons emitted can be expressed as follows [54]

$$\frac{dP}{d\omega} = \frac{9\sqrt{3}}{8\pi}\frac{P_0\alpha^2\chi^2}{\omega_c}\frac{\omega}{\omega_c}\int_{\omega/\omega_c}^{\infty} K_{5/3}(y)dy, \tag{2.47}$$

where $K_{5/3}(y)$ is the modified Bessel function of the second kind, $\omega_c = 3\gamma\alpha\chi/2\tau_e$ is the critical frequency for synchrotron emission (half of the radiated power is emitted above this frequency, the other half below), $\tau_e = r_e/c$ is the time for light to travel across the classical radius of the electron $r_e = e^2/4\pi\epsilon_0 mc^2$ and $P_0 = 2mc^2/3\tau_e$. The classical approach requires the emitted photon energy $\hbar\omega$ to be much smaller than the energy of emitting electron. This is satisfied if $\chi << 1$.

The derivation of the spectral properties of the radiated emission in regimes where quantum effects are non-negligible is simplified under the following conditions: [15]

1. Ultra-relativistic electrons emit radiation in a presence of a slowly varying field compared to the formation time of radiated photon (so-called local constant field approximation). This ensures that the coherent emission contribution are suppressed. Condition is fulfilled when the electromagnetic field has a relativistic strength $a_0 >> 1$.

2. The external field is under-critical. This requires that the following Lorentz invariants of the electromagnetic field are small compared to the critical field of QED $E_S \approx 1.3 \times 10^{18}$ V/m:

   $$\zeta_1 = F^{\mu\nu}F_{\mu\nu}/E_S^2 = (c^2\mathbf{B}^2 - \mathbf{E}^2)/E_S^2 << 1, \tag{2.48}$$

   $$\zeta_2 = \varepsilon^{\mu\nu\alpha\beta}F^{\mu\nu}F_{\alpha\beta}/E_S^2 = (c^2\mathbf{E}\cdot\mathbf{B})/E_S^2 << 1, \tag{2.49}$$

   where $\varepsilon^{\mu\nu\alpha\beta}$ is anti-symmetric unit tensor.

3. Here, we restrict our study to non-linear moderately quantum regime corresponding to $\chi \leq 1$ and $a_0 >> 1$. In this regime, the radiation reaction has been identified as the overall electron energy and momentum loss due to consecutive emission of many photons [69] and pair production and higher order QED processes are neglected.

The production rate of high-energy photons emitted by the electron, which is invariant under Lorentz transformations, can be expressed using the given assumptions as follows [65]

$$\frac{d^2N}{d\tau d\chi_\gamma} = \frac{2\alpha^2}{3\tau_e}\frac{G(\chi,\chi_\gamma)}{\chi_\gamma}, \tag{2.50}$$

where

$$G(\chi,\chi_\gamma) = \frac{\sqrt{3}\chi_\gamma}{2\pi\chi}\left(\int_\nu^\infty K_{5/3}(y)dy + \frac{3}{2}\chi_\gamma\nu K_{2/3}(\nu)\right) \tag{2.51}$$

is the quantum emissivity and $\nu = 2\chi_\gamma/3\chi(\chi-\chi_\gamma)$. The production rate (2.50) depends on the electron quantum parameter χ and on the quantum parameter of the emitted photon

$$\chi_\gamma = \frac{|F^{\mu\nu}\hbar k_\nu|}{E_S mc}, \tag{2.52}$$

where $k^\nu = (\hbar\omega, \hbar\mathbf{k})$ is the four-momentum of emitted photon. This can be also written in a form

$$\chi_\gamma = \frac{\gamma_\gamma}{E_S}\sqrt{(\mathbf{E}_\perp + \mathbf{c}\times\mathbf{B})^2}, \tag{2.53}$$

where $\gamma_\gamma = \varepsilon_\gamma/mc^2 = \hbar\omega/mc^2$ is the photon normalized energy, m refers to electron mass, $\mathbf{c}$ denotes velocity vector with $||\mathbf{c}|| = c$ and $\mathbf{E}_\perp$ denotes the electric field orthogonal to the propagation direction of the photon.

Let us consider a laboratory frame where the electron is ultra-relativistic ($\gamma >> 1$). The photon quantum parameter can be linked to electron quantum parameters as $\chi_\gamma = \xi\chi$ with $\xi = \gamma_\gamma/\gamma$ as the ratio of photon normalized energy γ_γ and electron Lorentz factor. In this frame, the instantaneous power spectrum is obtained from (2.50) as [15]

$$\frac{dP_{rad}}{d\gamma_\gamma} = \frac{\sqrt{3}}{2\pi}\frac{2\alpha^2 mc^2}{3\tau_E}\frac{\xi}{\gamma}\left(\int_\nu^\infty K_{5/3}(y)dy + \frac{\xi^2}{1-\xi}K_{2/3}(\nu)\right). \tag{2.54}$$

This reduces for $\chi << 1$ to (2.47).

From integration of (2.54) over all photon energies we can derive the another Lorentz invariant quantity - instantaneous power radiated away by the electron [70]. It has a form

$$P_{rad} = P_0 \alpha^2 \chi^2 g(\chi), \tag{2.55}$$

where

$$g(\chi) = \frac{9\sqrt{3}}{8\pi} \int_0^\infty d\nu \left(\frac{2\nu^2 K_{5/3}(\nu)}{(2+3\nu\chi)^2} + \frac{4\nu(3\nu\chi)}{(2+3\nu\chi)^4} K_{2/3}(\nu) \right). \tag{2.56}$$

The $g(\chi)$ is often called quantum correction. The radiated power P_{rad} is only classical (Larmor) radiated power $P_{cl} = P_0 \alpha^2 \chi^2$ multiplied by the value of the quantum correction. The classical limit can be retained in a limit $\chi \to 0 \Rightarrow g(\chi) \to 1$. This approach of replacing P_{cl} for P_{rad} may seem as heuristic, but it can be shown that statistical average of the quantum effects provide the quantum correction naturally [70].

One effect of this correction is that the emission of photons with higher energies than the emitting particle ($\chi_\gamma > \chi$) is prevented. However, this is not the only quantum effect. When χ approaches unity, even a single radiated photon can carry a large portion of electron momentum. At this regime, the concept of continuous radiation breaks down. Electron lose energy probabilistically, in discrete emissions. The example of modelling of this stochastic emission is described in more details in the next section. Now we can interpret the emission spectrum (2.54) as the probability function of the photon energy at specific moment in time. Although two electrons may possess identical values for γ and χ, they can still emit photons with different energies, or none at all, resulting in distinct recoil. This is in contrast to the classical notion, where the continuous loss of energy is attributed to the emission of numerous photons, each having infinitesimal energy values. This is illustrated in Fig. 2.1.

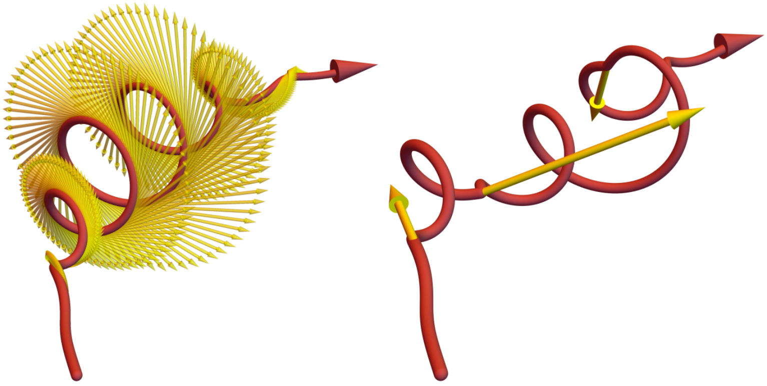


Figure 2.1: In the classical case (left), electron (with the trajectory depicted by the red arrow) is under continuous loss of energy to radiation (yellow) under Landau-Lifshitz force. In the quantum regime (right), the energy loss is discontinuous and probabilistic. Adapted from T. Blackburn (2015) [71].

2.4 Multiphoton Breit-Wheeler pair creation

The nonlinear Breit-Wheeler process ($\gamma + n\omega \rightarrow e^- + e^+$), also known as the multiphoton Breit-Wheeler process, involves the conversion of a high-energy photon into an electron-positron pair through its interaction with a strong electromagnetic field. For this process, the photon normalized energy γ_γ must be greater than 2. This effect is significant for $\chi \gtrsim 1$ [72].

The Ritus formulae [65] provide the energy distribution for the production rate of pairs by a high-energy photon

$$\frac{d^2N_{BW}}{d\chi_\pm dt} = \frac{\alpha m^2c^4}{\pi\sqrt{3}\hbar\varepsilon_\gamma\chi_\gamma}\int_x^\infty \sqrt{s}K_{1/3}\left(\frac{2}{3}s^{3/2}ds\right) - \left(2 - \chi_\gamma x^{3/2}\right)K_{2/3}\left(\frac{2}{3}x^{3/2}\right), \quad (2.57)$$

where $x = (\chi_\gamma/\chi_-\chi_+)^{2/3}$. The χ_- and χ_+ are the respective quantum parameters for created electron and the positron after pair creation with $\chi_- = \chi_\gamma - \chi_+$. Total production rate can be written as

$$\frac{dN_{BW}}{dt} = \frac{\alpha m^2c^4}{\hbar\varepsilon_\gamma}\chi_\gamma T(\chi_\gamma), \quad (2.58)$$

where

$$T(\chi_\gamma) = \frac{1}{\pi\sqrt{3}\chi_\gamma^2}\int_0^\infty \left[\int_x^\infty \sqrt{s}K_{1/3}\left(\frac{2}{3}s^{3/2}ds\right) - \left(2 - \chi_\gamma x^{3/2}\right)K_{2/3}\left(\frac{2}{3}x^{3/2}\right)\right]d\chi_-. \quad (2.59)$$

Chapter 3

Modelling of the radiation reaction

In this chapter, we focus on modelling of the radiation reaction force in different regimes from classical, through semi-quantum to full quantum regime. Implementation of various algorithms in Smilei and Monte-Carlo algorithm implemented in the EPOCH code are discussed in more details within this chapter.

3.1 The classical regime

In the classical regime, where $\chi << 1$, the QED stochastic effects can be neglected. The radiation reaction behaves as a continuous friction force acting on particles. Relying solely on the Lorentz force to determine the particle motion and subsequent emission results in an energy balance inconsistency. To resolve this, equations (2.35) or (2.40) can be used as the equation of motion. This approach ensures that the energy lost by the electron is matched by the energy carried away in radiation [68]. Various models for the radiation reaction force in this regime have been proposed [5]. Classical radiation reaction implementations in plasma simulation codes have predominantly utilized the Landau-Lifshitz equation due to its first-order momentum and relatively low computational cost. This approach has been employed in several studies investigating radiation reaction effects in laser-plasma interactions [6, 73]. These codes have been used to study radiation reaction effects in laser–plasma interactions and also to study differences among different implementations [1, 74].

Smilei employs models based on the Landau-Lifshitz (LL) equation adapted for high Lorentz factors $\gamma >> 1$ [19]. It has been demonstrated that the LL force combined with quantum corrections naturally arises from the complete quantum description [70]. The implemented equation of motion has a form [18]

$$\frac{d\mathbf{p}}{dt} = \mathbf{F}_L + \mathbf{F}_{rad}, \tag{3.1}$$

where $\mathbf{F}_L$ is the Lorentz force and

$$\mathbf{F}_{rad} = P_0 \alpha^2 \chi^2 g(\chi) \frac{\mathbf{u}}{\mathbf{u}^2 c}, \tag{3.2}$$

where $\mathbf{u} = \mathbf{p}/mc\gamma$. The quantum correction (2.56) is costly to compute for every timestep. Smilei uses the fit function of $g(\chi)$ given by

$$g(\chi) = [1 + 4.8(1 + \chi)\log(1 + 1.7\chi) + 2.44\chi^2]^{-2/3}. \tag{3.3}$$

This model is called "corrected Landau-Lifshitz" or "CLL". This equation implementation is based on the leap-frog technique. In the Smilei input file, this purely classical model is called "Landau-Lifshitz" or "LL", and uses the case for $g(\chi) = 1$. The spectrum (2.54) reduces to classical spectrum of synchrotron radiation. Classical radiation model is not included in EPOCH.

3.2 Intermediate, semi-quantum regime

During the intermediate regime, the energy of the emitted photons remains significantly lower than that of the electrons emitting them. Despite this, the random nature of photon emission cannot be disregarded. Consequently, a stochastic differential equation, obtained from a Fokker-Planck expansion of the complete quantum (Monte-Carlo) model, can be used to describe the electron dynamics [70].

Smilei employs this semi-quantum regime as a change in electron momentum during a time interval dt in a form [18]

$$d\mathbf{p} = \mathbf{F}_L dt + \mathbf{F}_{rad} dt + mc^2\sqrt{R(\chi,\gamma)}dW\mathbf{u}/\mathbf{u}^2 c, \tag{3.4}$$

where F_L and F_{rad} have the same definition as the CLL regime (with quantum correction) and the stochastic force term (also called diffusion term) is proportional to dW, a Wiener process of variance dt. This stochastic term allows to account for the stochastic nature of high energy photon emission and also depends on the function

$$R(\chi,\gamma) = \frac{2\alpha^2}{3\tau_e}\gamma h(\chi), \tag{3.5}$$

where

$$h(\chi) = \frac{9\sqrt{3}}{4\pi}\int_0^\infty \left(\frac{2\chi^3\nu^3}{(2+3\nu\chi)^3}K_{5/3}(\nu) + \frac{54\chi^5\nu^4}{(2+3\nu\chi)^5}K_{2/3}(\nu)\right). \tag{3.6}$$

For the numerical implementation, dW is a random number generated using a normal distribution of variance equal to the simulation timestep Δt. Smilei avoids the computationally demanding direct computation of (3.6) during the emission process by using tabulated values or fit functions for performance reasons. In terms of tabulation, Smilei first checks for an external table at the specified path. If the table is not present, it is computed at initialization and output to the current simulation directory. To obtain polynomial fits of this integral, Smilei uses log-log high-order polynomials of order 5 and 10, valid for quantum parameters between 0.1 and 10. Used order 5 log polynomial has a form (here rounded up to 4 decimal places for brevity)

$$h(\chi) = \exp(1.3999 \cdot 10^{-4}\log(\chi)^5 + 3.1237 \cdot 10^{-3}\log(\chi)^4 + 1.0966 \cdot 10^{-2}\log(\chi)^3 - \\ -1.7340 \cdot 10^{-1}\log(\chi)^2 + 1.4927 \cdot \log(\chi) - 2.7490). \tag{3.7}$$

This model is available in Smilei under the name "Niel". Similarly as the classical methods described in Sec. 3.1, the Niel method is not implemented in EPOCH.

3.3 Quantum regime

In order to simulate fully quantum regime in the PIC codes, the Monte-Carlo method is standardly used. The Monte-Carlo model of high-energy photon emission is applicable for any value of the electron quantum parameter, provided that the assumptions mentioned in Sec. 2.3 hold. It accurately describes high-energy photon emission and its back-reaction on particle dynamics in the quantum regime [75]. In this regime, photons with energies similar to those of the emitting particle can be generated, causing abrupt jumps in particle energy/velocity. The stochastic nature of high-energy photon emission is also significant. Consequently, the Monte-Carlo approach is typically used to model discrete high-energy photon emission and its impact on radiating particle dynamics.

The Monte-Carlo process implemented in Smilei [18] and EPOCH [76] PIC codes follows the algorithms described by Duclous, R. in 2010 [16]. This algorithm is complex and requires several steps:

1. To each particle, an incremental optical depth τ, which starts at 0, is assigned.. Once the particle reaches the final optical depth τ_f, the emission of the photon occurs. The value of final optical depth is sampled as $\tau_f = \log(\eta)$, where η is a random number from the interval $(0, 1)$. The optical depth τ evolves with the field and particle energy according to

$$\frac{d\tau}{dt} = \int_0^{\chi_\pm} \frac{d^2N}{d\chi' dt} d\chi' = \frac{2\alpha^2}{3\tau_e} \int_0^{\chi_\pm} \frac{G(\chi_\pm, \chi'/\chi_\pm)}{\chi'} d\chi' = \frac{2\alpha^2}{3\tau_e} K(\chi_\pm). \tag{3.8}$$

   Here, the $\chi_\pm$ refers to quantum parameter for the emitting electron (χ_-) or positron (χ_+). Due to the high computational cost of computing (3.8) for each particle individually, the integral $K(\chi_\pm)$ is tabulated. Smilei and EPOCH allows to input custom table or used the default one.

2. In order to determine the quantum parameter of the emitted photon χ_γ, this cumulative distribution function is inverted

$$\xi = P(\chi_\pm, \chi_\gamma) = \frac{\displaystyle\int_0^{\chi_\gamma} G(\chi_\pm, \chi'/\chi_\pm) d\chi'}{\displaystyle\int_0^{\chi_\pm} G(\chi_\pm, \chi'/\chi_\pm) d\chi'}. \tag{3.9}$$

   A random number $\phi \in [0, 1]$ is generated and the value of χ_γ is found as

$$\phi = \xi^{-1} = P(\chi_\pm, \chi_\gamma)^{-1} \tag{3.10}$$

   Similarly to (3.8), the values of $P(\chi_\pm, \chi_\gamma)$ are tabulated in $\chi_\pm$ and χ_γ directions to save the computational time. The external tables can be also provided in both Smilei and EPOCH codes.

3. The energy of the emitting photon is computed as

$$\varepsilon_\gamma = \hbar\omega = \frac{mc^2 \gamma_\pm \chi_\gamma}{\chi_\pm}, \tag{3.11}$$

   where $\gamma_\pm$ refers to the Lorentz factor of emitting electron/positron. The photon is then emitted in the forward direction. If a macro-photon is created, its initial position is at the position of the emitting particle with the same numerical weight. The particle momentum is then updated as

$$d\mathbf{p} = -\frac{\varepsilon_\gamma}{c}\frac{\mathbf{p}_\pm}{||\mathbf{p}_\pm||}, \tag{3.12}$$

where $\mathbf{p}_\pm$ refers to the momentum of the emitting positron/electron.

The resulting force follows from the recoil induced by the photon emission. Radiation reaction is therefore a random and discrete process capturing the stochastic nature of the radiation reaction. It should be noted that although momentum conservation is maintained, energy conservation is not exact. However, it has been demonstrated [77] that as the energy of the particle increases to infinity, the error approaches zero, and for $\varepsilon_\pm >> 1$ and $\varepsilon_\gamma << \varepsilon_\pm$, the error is negligible [77]. The motion of the electron is governed by the Lorentz force during the periods between radiation reaction emission events.

This regime can be specified in Smilei input file by the name "Monte-Carlo" and in EPOCH, it can be enabled by including QED block in the input file.

3.4 Stochastic scheme of the Breit-Wheeler pair creation

In both EPOCH [76] and Smilei [78], the Breit-Wheeler (B-W) process is treated as a Monte-Carlo process similar to the radiation reaction described in the previous section. Again, the implementations follow the algorithm described by Duclous, R. in 2010 [16].

When this algorithm is enabled, a macrophoton is emitted from a macroparticle. These macrophotons are defined by mass and charge equal to 0. The momentum for macrophoton is defined as $\mathbf{p}_\gamma = \hbar\mathbf{k} = \gamma_\gamma m\mathbf{c}$, where $\mathbf{k}$ is the wave factor, γ_γ is the photon Lorentz factor, m refers to electron mass and $\mathbf{c}$ is the photon velocity vector with $||\mathbf{c}|| = c$. The B-W algorithm consists of these steps:

1. Incremental optical depth τ is assigned to a photon. The starting value is $\tau = 0$. Once the photon reaches the final optical depth τ_f, the creation of the pair occurs. The value of final optical depth is sampled as $\tau_f = \log(\eta)$, where η is a random number from the interval $(0, 1)$. The optical depth τ evolves with the field and particle energy according to

$$\frac{d\tau}{dt} = \frac{dN_{BW}}{dt}(\chi_\gamma). \tag{3.13}$$

   This is the production rate of pairs (2.58). This integral is tabulated in order to save computational time.

2. In order to determine the quantum parameter of the created electron χ_-, the

following cumulative distribution function is inverted

$$P(\chi_-, \chi_\gamma) = \frac{\displaystyle\int_0^{\chi_-} \frac{d^2 N_{BW}}{d\chi' dt} d\chi'}{\displaystyle\int_0^{\chi_\gamma} \frac{d^2 N_{BW}}{d\chi' dt} d\chi'}. \tag{3.14}$$

A random number $\phi \in [0, 1]$ is generated and the value of χ_- is found as

$$\phi = P(\chi_-, \chi_\gamma)^{-1}. \tag{3.15}$$

Similarly to (3.13), the values of $P(\chi_-, \chi_\gamma)$ are tabulated to save the computational time. The value of positron quantum parameter is computed as $\chi_+ = \chi_\gamma - \chi_-$. The external tables can be provided in both Smilei and EPOCH codes.

3. The energy of the created electron is computed as

$$\varepsilon_- = mc^2\gamma_- = mc^2(1 + (\gamma_\gamma - 2)\chi_-/\chi_\gamma), \tag{3.16}$$

where γ_- refers to the Lorentz factor of created electron. If the photon Lorentz factor $\gamma_\gamma < 2$ the pair creation is not possible since the photon energy is below the rest mass of the particles. After updating the momentum of the photon, pairs are generated at its position with the same propagation direction. The weight of the macro-photon is maintained in this process.

This algorithm is enabled in Smilei by specifying the "Monte-Carlo" radiation model and adding a "MultiphotonBreitWheeler" block in the input file. In EPOCH, this is enabled by specifying "produce_pairs = True" inside the QED block in the input file.

3.5 Choosing the model for simulation

This concise section presents a table that provides information on multiple radiation reaction regimes, their corresponding χ value, and the recommended radiation reaction model to use for each regime. It should be noted that this table has been directly sourced from the official Smilei documentation [18]. These values are based on the previous scientific works [70].

Regime	**χ value**	**Recommended models**
Classical radiation emission	$\chi \lesssim 10^{-3}$	Landau-Lifshitz
Semi-quantum radiation emission	$\chi \lesssim 10^{-2}$	Corrected Landau-Lifshitz
Weak quantum regime	$\chi \lesssim 10^{-1}$	Stochastic model of Niel / Monte-Carlo
Quantum regime	$\chi \sim 1$	Monte-Carlo

Chapter 4

Results and discussion

In this chapter, we present the findings of a comparative study that was conducted to investigate the implementation of radiation reaction in Smilei and EPOCH PIC codes. The aim of the study was to evaluate the performance of these two codes by running physically identical benchmarks in one and two dimensions. Also, the implementation of the Breit-Wheeler pair creation was tested.

The simulation parameters were carefully selected to ensure that they were as similar to each other as possible. The simulations were carried out for various values of the laser intensities. By comparing the results obtained from the two codes, we were able to gain valuable insights into the differences between the algorithms implemented in the codes, and how they impact the accuracy of the simulations.

The details of the simulation setup are described in this chapter, including the input parameters such as the number of particles, the size of the simulation domain, and the time step used in the numerical solver. The simulations were conducted using the Metacentrum computational grid.

4.1 Methods: Physics

Firstly, the simulations were conducted for one-dimensional (1D) and two-dimesional (2D) case for radiation reaction effects only. Next, 1D and 2D case with enabled Breit-Wheeler pair creation was also tested. Here, each case is described separately in SI units. The EPOCH uses SI units as the parameter input, however Smilei uses normalized units. The definition of these units is provided in Appendix B. Summary of the simulation parameters is provided in Tab. 4.1 and 4.2.

4.1.1 1D case

In the 1D case, a 1 GeV monoenergetic electron beam with the uniform density of 1.1×10^{16} cm^{-3} and the beam duration of 2 fs propagates from the right boundary towards the left boundary. From the left boundary, a counter-propagating circularly polarized wave with a wavelength of $\lambda = 1$ µm is injected. For the study of different behaviour of all studied algorithms under different quantum parameters, the intensity of the injected wave was gradually changed in each of the simulations. The intensity values were $I = 1.4 \times 10^{18} (a_0 = 1), 1.4 \times 10^{20} (a_0 = 10), 1.4 \times 10^{22} (a_0 = 100), 9.9 \times 10^{22} (a_0 = 270)$ W/cm^2. The intensity $I \sim 10^{23}$ W/cm^2 corresponds to the current world record [4]. The laser has a super-Gaussian temporal profile of the 4th order with the full width at half maximum (FWHM) of approximately 33 fs (10 laser periods) with the initial center at $t = 100$ fs. This configuration of an electron beam with a counter-propagating electromagnetic wave maximizes the value of the quantum parameter χ for the given electron energy and field strength and is the most effective to trigger QED effects [66].

Length of the simulation window is 30 µm. Total simulation time is 200 fs. The value of CFL constant is 0.95. Injected wave collides with the electron beam in the middle of the simulation window. This is accomplished by freezing the particles in time from $t = 0$ to $t = 116$ fs. The theoretical maximum value of the quantum parameter χ_{max}^{theory} was calculated from (2.43).

At the beginning of the simulation, 64 macro-particles per cell were created. These have a 5 point stencil as their shape function (1.17). There are 3840 cells in x direction. The Maxwell solver is "Yee". A Boris pusher is used as a particle pusher. Particles are initialized with temperature $T = 0$ K to simulate mono-energetic beam. At both ends, open-boundary conditions are used, "silver-muller" in Smilei and "Simple Laser" in EPOCH. Radiation models used in Smilei are "Landau-Lifshitz", "corrected-Landau-Lifshitz", "Niel" and "Monte-Carlo". The parameters for minimum χ value to produce a photon was set to be 10^{-5}. In EPOCH, a QED radiation reaction module was enebled with Monte-Carlo generator. The limit for minimal photon energy was set to 0. The input file for these simulations are provided in the Github repository [79] with filenames '`1d_qed_smilei.py`' (Smilei) and '`1d_qed_epoch.deck`' (EPOCH). These input file present an example case for $a_0 = 100$ and $\chi_{max}^{theory} = 1$.

4.1.2 2D case

In two dimesions, the simulation window is a box size 30 µm × 4 µm in x and y directions, respectively. Cell count is 1920 × 256. The bunch duration is 2 fs and bunch width is 0.16 µm. The wave is linearly polarized in the y direction. The injected electromagnetic wave was chosen as a planar wave for simplicity. This is approximated by a Gaussian spatial profile in the y direction, with the beam waist of 160 m. The wave is focused in the middle of the simulation window. For the 2D case, 32 particles per cell are generated. The boundary conditions in the new y direction are defined as periodic. Every other simulation parameter is identical to the previous case. The input file for these simulations is provided in Github repository [79] with filenames '`2d_qed_smilei.py`' (Smilei) and '`2d_qed_epoch.deck`' (EPOCH). These input file present an example case for $a_0 = 100$ and $\chi_{max}^{theory} = 1$.

4.1.3 Breit-Wheeler pair creation

For the comparative study of Breit-Wheeler (BW) pair creation, the simulation parameters in 1D and 2D are identical to the previous respective cases, with two exceptions. The electron beam kinetic energy is 4 GeV, and only Monte-Carlo radiation reaction model is studied in Smilei. The studied intensity is $I = 1.4 \times 10^{22}$ W/cm^2 ($a_0 = 100$). These parameters were selected to maximize the pair creation, with expected $\chi_{max}^{theory} = 3.8$. The Breit-Wheeler pair creation with the respective parameters is enabled in the input files. The input file for these simulations is provided in Github repository [79] with filenames '`1d_BW_smilei.py`', '`2d_BW_smilei.py`' (Smilei) and '`1d_BW_epoch.deck`', '`2d_BW_epoch.deck`' (EPOCH).

Parameter	**Value**
Laser wavelength	1.0 µm
Laser pulse duration	33 fs FWHM
Minimum examined intensity	1.4×10^{18} W/cm^2 ($a_0 = 1$)
Maximum examined intensity	9.9×10^{22} W/cm^2 ($a_0 = 270$)
Electron energy	1 GeV (no B-W); 4 GeV (BW)
Electron density	1.1×10^{16} cm^{-3}
Electron bunch duration	2 fs
Electron bunch width	0.16 µm

Table 4.1: Physics parameters used in the simulations.

Dimensions	30 µm (1D); 30 µm × 4 µm (2D)
Time	200 fs
CFL constant	0.95
Pusher	Boris algorithm
Maxwell solver	Yee
Particles per cell	64 (1D); 32 (2D)

Table 4.2: Used simulation parameters.

4.2 Results of 1D simulations

In this section, we present the outcomes of 1D simulations. Each case is accompanied by 3 graphs. The first graph depicts the total kinetic energy of electron beam as a function of time, while the second graph represents the total radiated energy as a function of time. The left y-axis of these graphs displays the corresponding energies ε relative to the initial total energy in the simulation $\varepsilon_{tot}(t = 0)$. The right y-axis displays the energies in Joules. The third graph displays the comparison of total kinetic energy of electrons in Smilei $\varepsilon_{Smilei}^{MC}$ over the total kinetic energy from EPOCH ε_{EPOCH}^{MC} for the MC algorithm expressed in relative units as a function of time. We present the results for four different scenarios, namely, $\chi_{max}^{theory} = 0.01$ (Fig. 4.1), $\chi_{max}^{theory} = 0.1$ (Fig. 4.2), $\chi_{max}^{theory} = 1$ (Fig. 4.3), and $\chi_{max}^{theory} = 2.6$ (Fig. 4.4).

For the case with enabled pair creation via BW algorithm with $\chi_{max}^{theory} = 3.8$ (Fig. 4.5), the first graph shows the comparison of total kinetic energy in electrons, positrons and total radiated energy over time. This is again shown in relative (left y axis) and SI units (right y axis). The second graph is again the comparison of total electron kinetic energy ε from both PIC codes over ε_{EPOCH}. In Smilei, only MC radiation reaction model was considered.

In the instance, where $\chi_{max}^{theory} = 0.01$ (Fig. 4.1), the findings indicate that the radiated energy derived from the Monte Carlo (MC) algorithm in Smilei is lower than that of other algorithms. This difference is very minor in relative terms. Results from other algorithms are comparable to each other. This is expected results, since this χ range is inside all considered domains of applicability.

For $\chi_{max}^{theory} = 0.1$ (Fig. 4.2), the LL algorithm started to show higher rates of emission compared to other algorithms. This was expected, as reached χ values are outside of the domain of applicability for this algorithm. Other algorithms reached comparable results, including both MC algorithms, where the difference was minimal.

For $\chi_{max}^{theory} = 1$ (Fig. 4.3), the classical LL algorithm deviated even more from other algorithms. Interestingly, the CLL is still comparable to Niel and MC algorithms even outside the domain of applicability. This can be caused by a fact that χ_{max}^{theory} is only theoretical maximum value expected during the simulation. Reached values of χ are usually lower and might be closer to the CLL domain of applicability. Niel algorithm was also comparable to both MC algorithms. Smilei MC algorithm showed higher electron kinetic energy (by as high as 6%) and started to radiate later. However, the final kinetic energy values are comparable.

For $\chi_{max}^{theory} = 2.6$ (Fig. 4.4), the LL algorithm overestimated the radiated energy. The CLL and Niel algorithms estimations were comparable to Smilei MC, however the radiated energy was lower during the interaction with the laser. These are remarkable results, as the reached intensities were far outside their respective domain of applicability. Furthermore, the Smilei MC algorithms again exhibited a delayed onset of radiation and a higher electron kinetic energy compared to the EPOCH simulation due to different stochastic process. The electron kinetic energy reached values up to 10% greater compared to EPOCH during the interaction. However, the final energy values were again comparable for both codes. Total radiated energy during the simulation was greater than one. This could possibly be a result of electrons oscillating in the electromagnetic field and radiating continuously.

In the BW case with $\chi_{max}^{theory} = 3.8$ (Fig. 4.5), the results obtained from both PIC codes are comparable. Specifically, the Smilei algorithm again exhibited a delayed start of radiation than EPOCH, yet the total radiated energy was ultimately greater in Smilei at the end of the simulation. This can be caused by the stochastic nature of the MC algorithm. Overall, the results from both codes were similar, including the number of generated positrons.

The LL algorithm deviated from other algorithms for $\chi_{max}^{theory} = 0.1$ and higher. This was expected result. The CLL algorithm was reasonably accurate in each case, even for $\chi_{max}^{theory} = 2.6$ with slight deviation. However, this algorithm does not account for the stochastic effects on particle motion. Niel algorithm was comparable to MC in each case, with minor underestimating of the radiation for $\chi_{max}^{theory} = 2.6$ compared to MC. For stochastic MC algorithms, Smilei showed delayed onset of radiation in each case compared to EPOCH. The difference in electron kinetic energy during the interaction was up to 10%. However, the energy balances of MC algorithms in both PIC codes were comparable and didn't show any significant deviation to one another.

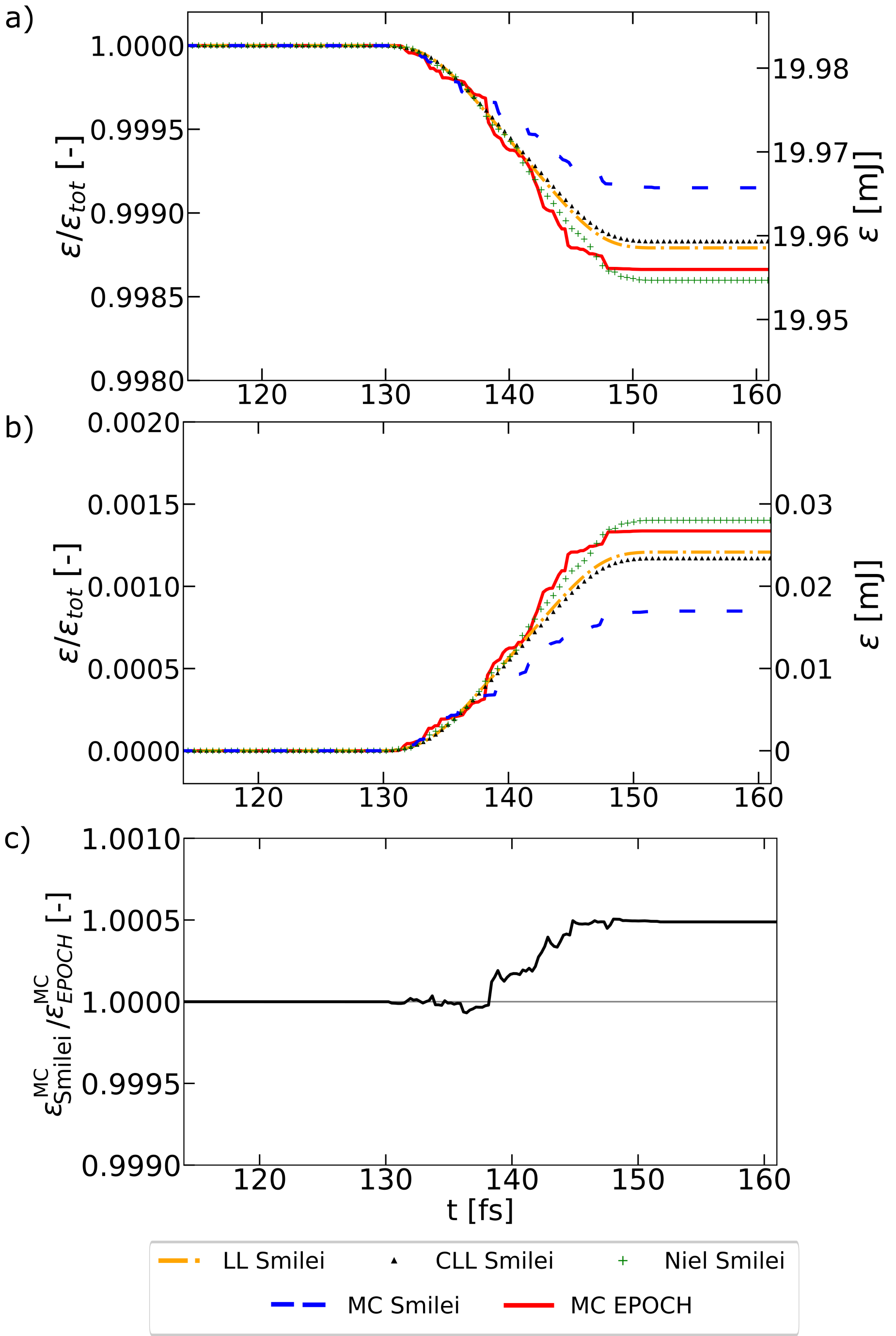


Figure 4.1: a) The comparison of total kinetic energy of electron ε over total initial kinetic energy in the simulation ε_{tot} for $\chi_{max}^{theory} \approx 0.01$. b) The comparison of total radiated energy ε over total initial kinetic energy in the simulation ε_{tot} for $\chi_{max}^{theory} \approx 0.01$. The axis on the right side shows the energy in Joules for comparison. c) The comparison of total kinetic energy of electrons in Smilei $\varepsilon^{MC}_{Smilei}$ over the total kinetic energy from EPOCH ε^{MC}_{EPOCH} for the MC algorithm (black line).

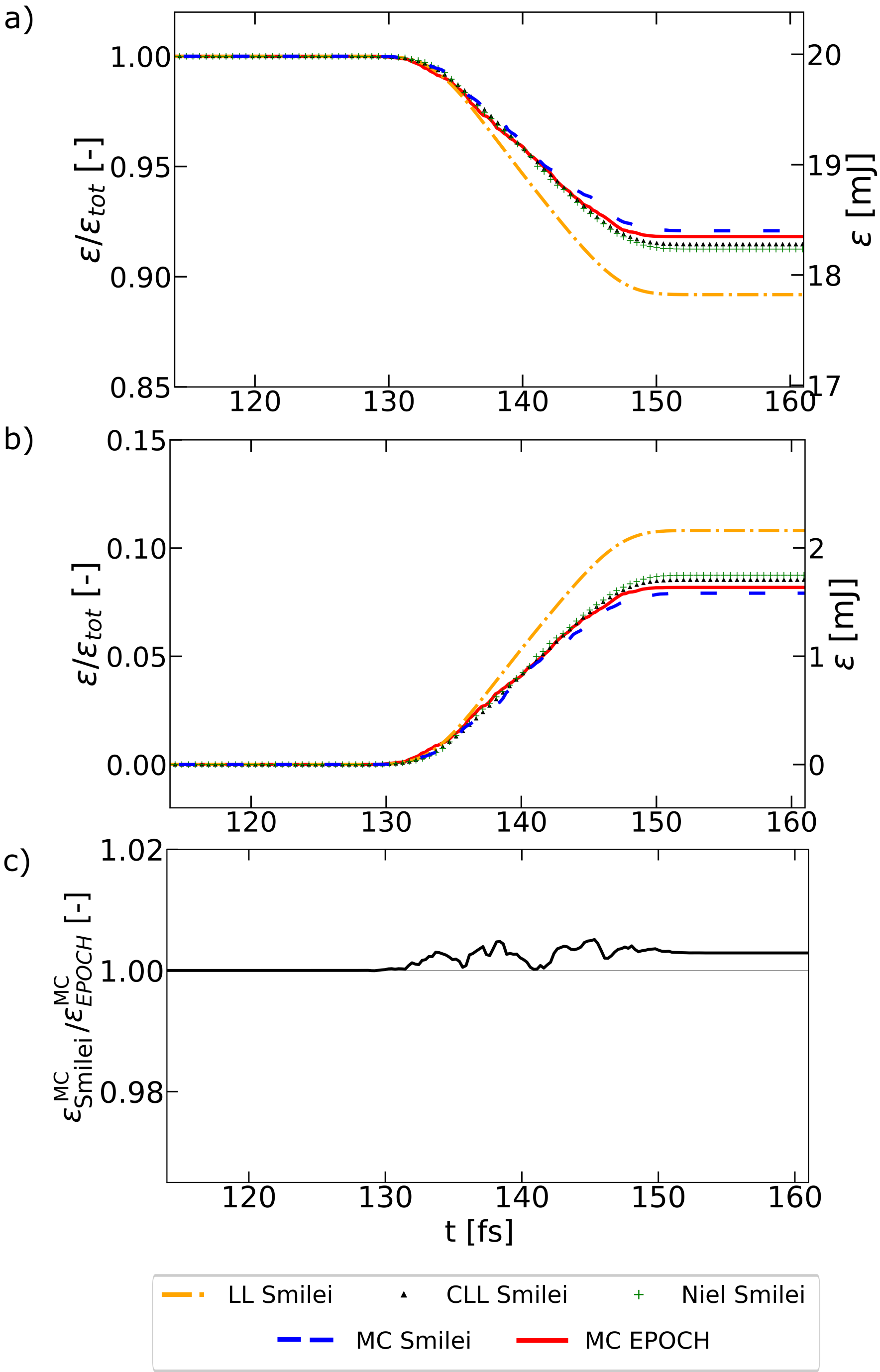


Figure 4.2: a) The comparison of total kinetic energy of electron ε over total initial kinetic energy in the simulation ε_{tot} for $\chi_{max}^{theory} \approx 0.1$. b) The comparison of total radiated energy ε over total initial kinetic energy in the simulation ε_{tot} for $\chi_{max}^{theory} \approx 0.1$. The axis on the right side shows the energy in Joules for comparison. c) The comparison of total kinetic energy of electrons in Smilei $\varepsilon_{Smilei}^{MC}$ over the total kinetic energy from EPOCH ε_{EPOCH}^{MC} for the MC algorithm (black line).

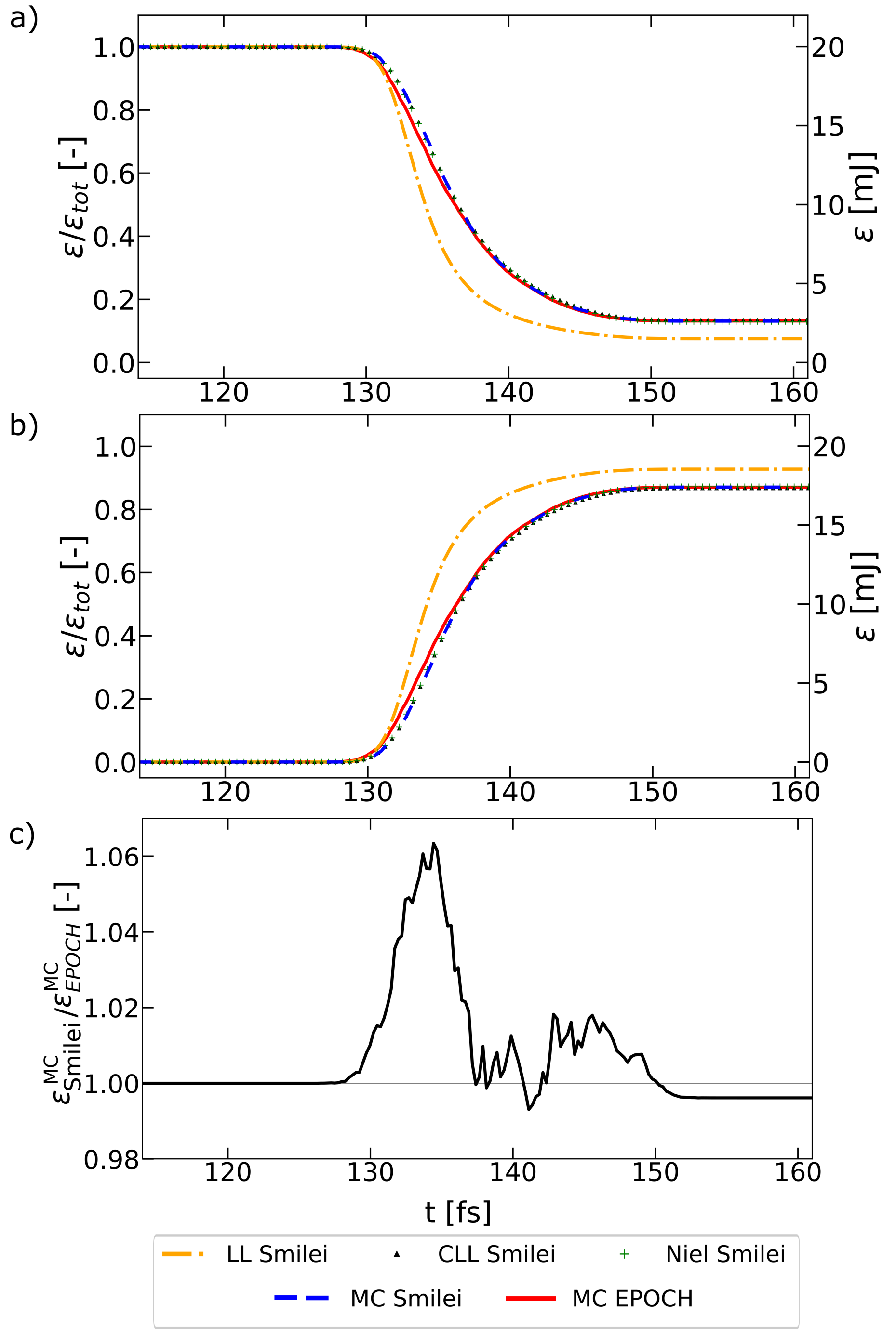


Figure 4.3: a) The comparison of total kinetic energy of electron ε over total initial kinetic energy in the simulation ε_{tot} for $\chi^{theory}_{max} \approx 1$. b) The comparison of total radiated energy ε over total initial kinetic energy in the simulation ε_{tot} for $\chi^{theory}_{max} \approx 1$. The axis on the right side shows the energy in Joules for comparison. c) The comparison of total kinetic energy of electrons in Smilei $\varepsilon^{MC}_{Smilei}$ over the total kinetic energy from EPOCH ε^{MC}_{EPOCH} for the MC algorithm (black line).

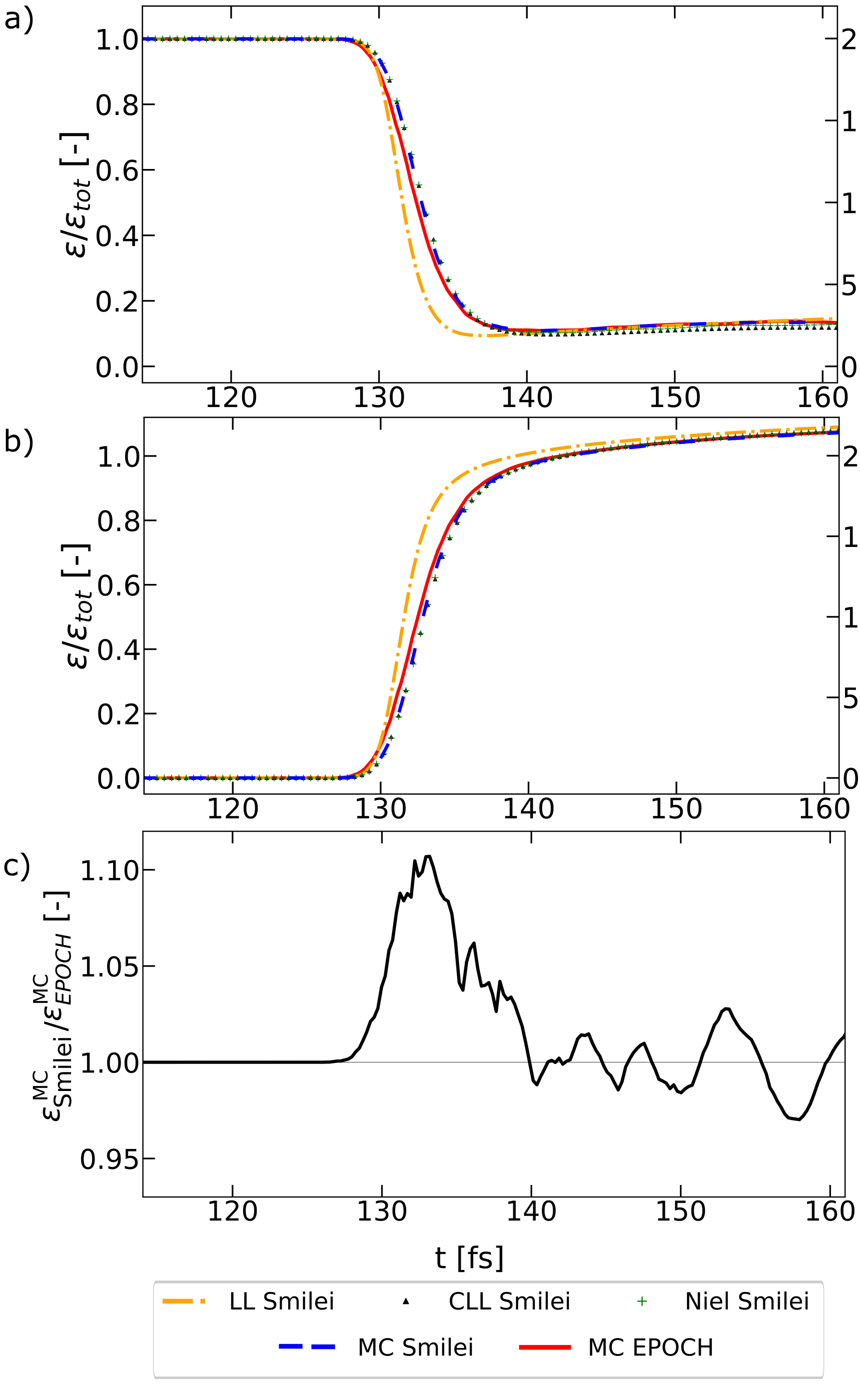


Figure 4.4: a) The comparison of total kinetic energy of electron ε over total initial kinetic energy in the simulation ε_{tot} for $\chi^{theory}_{max} \approx 2.6$. b) The comparison of total radiated energy ε over total initial kinetic energy in the simulation ε_{tot} for $\chi^{theory}_{max} \approx 2.6$. The axis on the right side shows the energy in Joules for comparison. c) The comparison of total kinetic energy of electrons in Smilei $\varepsilon^{MC}_{Smilei}$ over the total kinetic energy from EPOCH ε^{MC}_{EPOCH} for the MC algorithm (black line).

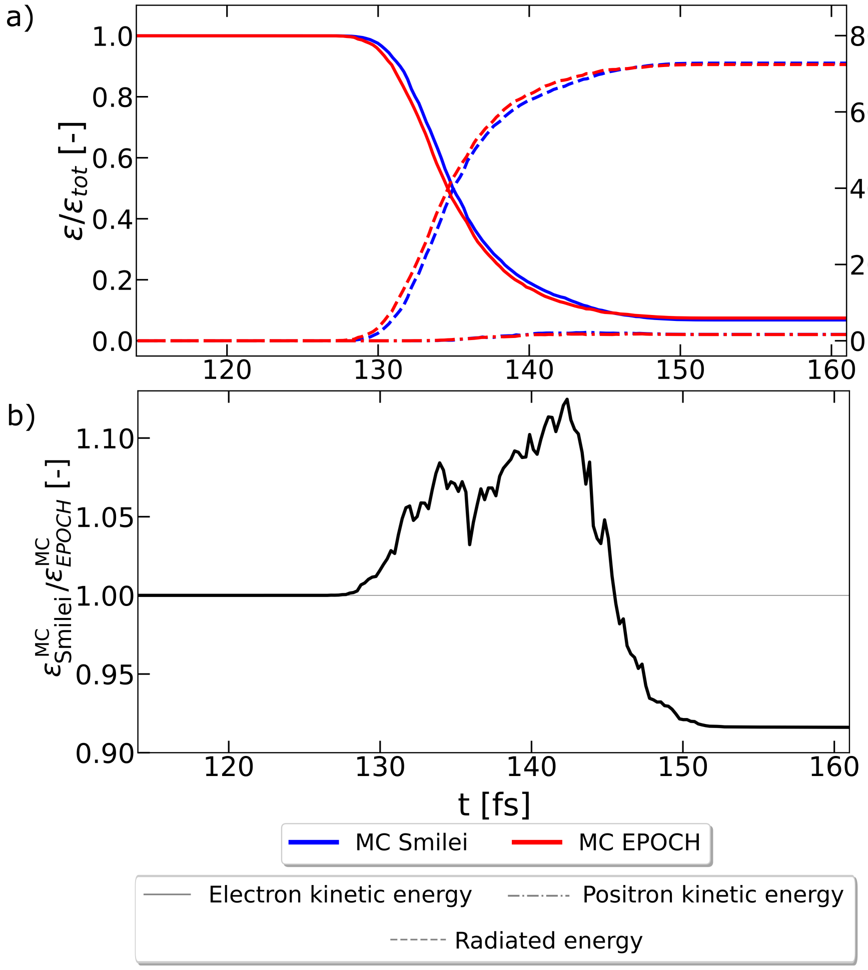


Figure 4.5: a) The comparison of total kinetic energy of electron/positron and total radiated energy ε over total initial kinetic energy in the simulation ε_{tot} for enabled BW algorithm with $\chi_{max}^{theory} \approx 3.8$. b) The comparison of total kinetic energy of electrons in Smilei $\varepsilon^{MC}_{Smilei}$ over the total kinetic energy from EPOCH ε^{MC}_{EPOCH} for the MC algorithm (black line).

4.3 Results of 2D simulations

In this section, we present the outcomes of 2D simulations. Again, each case is accompanied by 3 graphs similar to previous section. We present the results for four different scenarios, namely, $\chi_{max}^{theory} = 0.01$ (Fig. 4.6), $\chi_{max}^{theory} = 0.1$ (Fig. 4.7), $\chi_{max}^{theory} = 1$ (Fig. 4.8), and $\chi_{max}^{theory} = 2.6$ (Fig. 4.9). The comparison of BW pair production algorithm for $\chi_{max}^{theory} = 3.8$ is shown in Fig. 4.10.

The conclusions from simulations $\chi_{max}^{theory} = 0.01,\ 0.1,\ 1$ are identical to the previous section for 1D case. For $\chi_{max}^{theory} = 0.1$ and $\chi_{max}^{theory} = 1$, the LL algorithm showed higher rates of emission compared to other algorithms, while the CLL algorithm remained comparable to Niel and MC algorithms even outside the domain of applicability. The Smilei MC algorithm showed higher electron kinetic energy during the interaction and started to radiate later, but the final kinetic energy values are comparable to EPOCH. For $\chi_{max}^{theory} = 2.6$, total kinetic energies for LL and CLL algorithms oscillate after the photon emission. Again, this could possibly be a result of electrons oscillating in the electromagnetic field and radiating continuously. Also, the difference between total kinetic energy after the interaction in Smilei and EPOCH MC algorithms grew slightly beyond 10%. In the BW case, the results from both PIC codes are almost identical in terms of the energy balance.

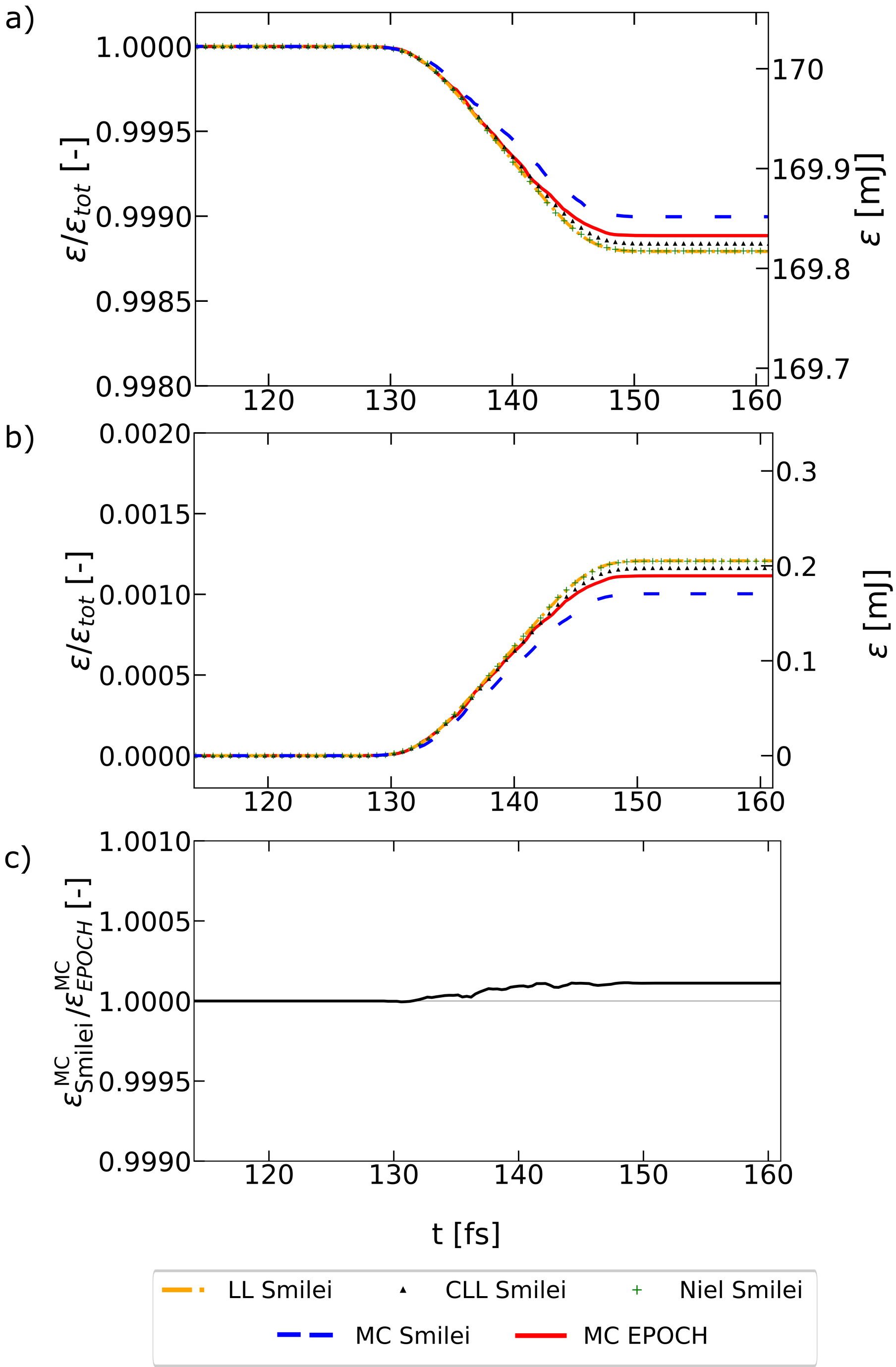


Figure 4.6: a) The comparison of total kinetic energy of electron ε over total initial kinetic energy in the simulation ε_{tot} for $\chi_{max}^{theory} \approx 0.01$. b) The comparison of total radiated energy ε over total initial kinetic energy in the simulation ε_{tot} for $\chi_{max}^{theory} \approx 0.01$. The axis on the right side shows the energy in Joules for comparison. c) The comparison of total kinetic energy of electrons in Smilei $\varepsilon_{Smilei}^{MC}$ over the total kinetic energy from EPOCH ε_{EPOCH}^{MC} for the MC algorithm (black line).

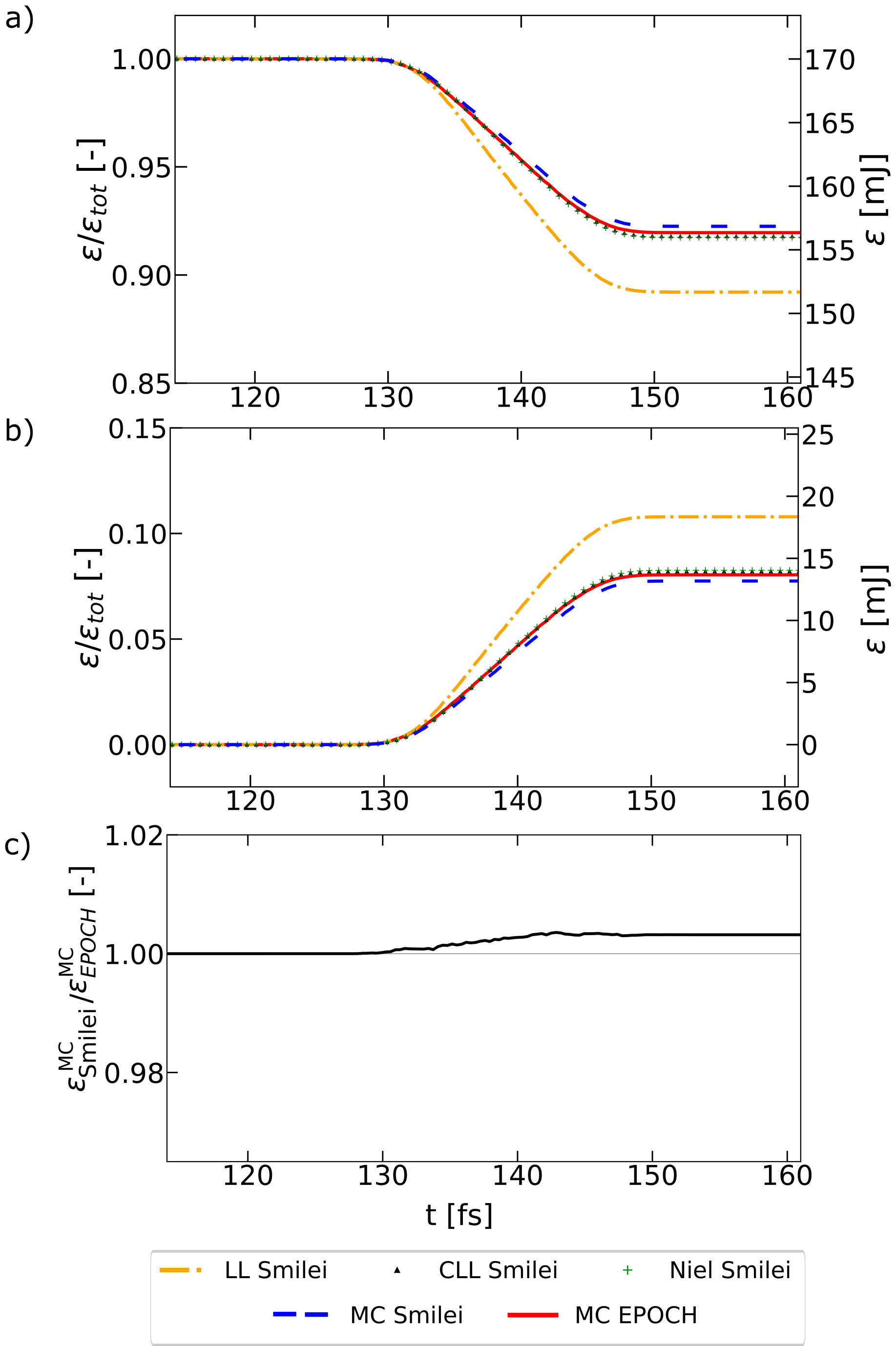


Figure 4.7: a) The comparison of total kinetic energy of electron ε over total initial kinetic energy in the simulation ε_{tot} for $\chi^{theory}_{max} \approx 0.1$. b) The comparison of total radiated energy ε over total initial kinetic energy in the simulation ε_{tot} for $\chi^{theory}_{max} \approx 0.1$. The axis on the right side shows the energy in Joules for comparison. c) The comparison of total kinetic energy of electrons in Smilei $\varepsilon^{MC}_{Smilei}$ over the total kinetic energy from EPOCH ε^{MC}_{EPOCH} for the MC algorithm (black line).

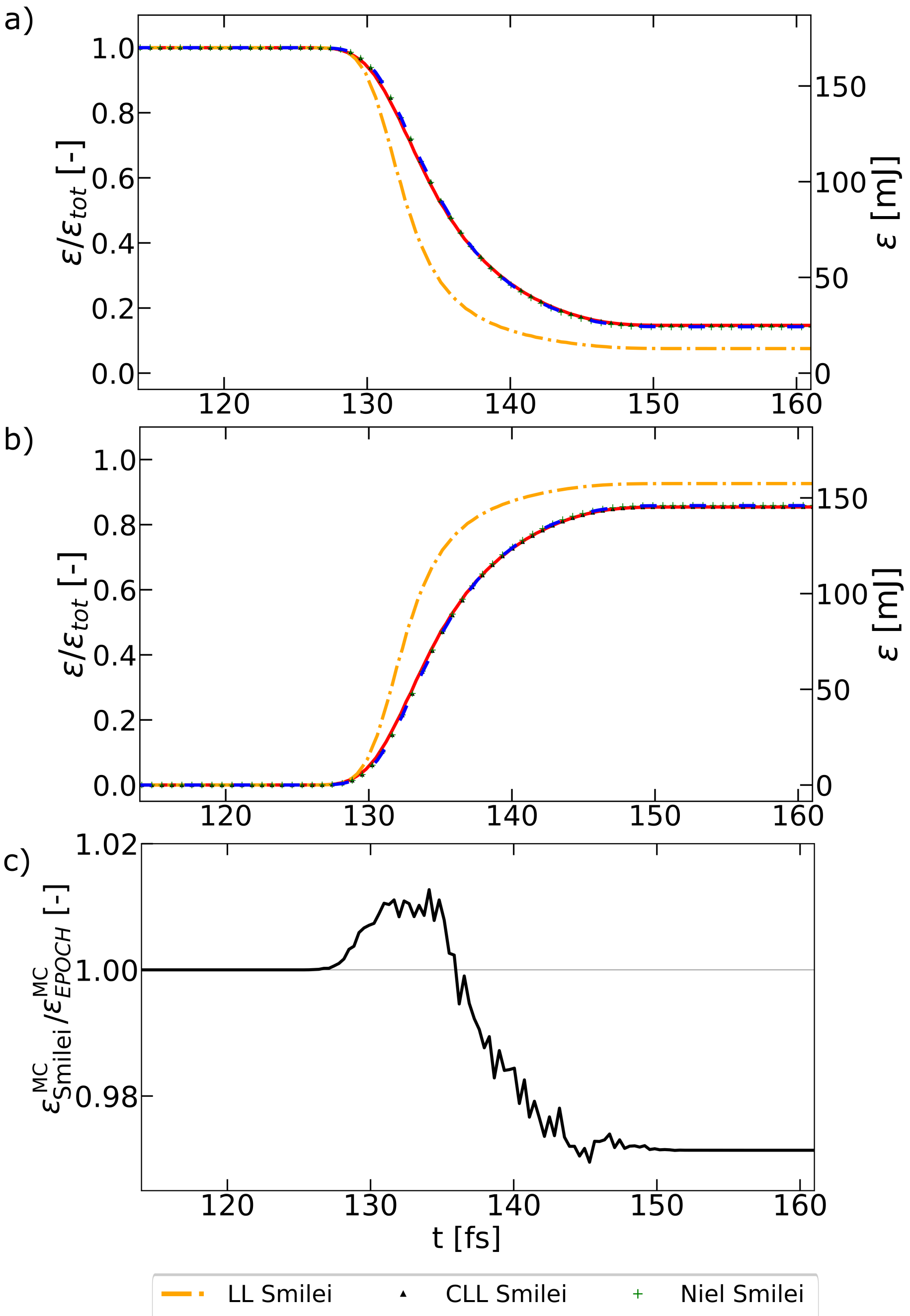


Figure 4.8: a) The comparison of total kinetic energy of electron ε over total initial kinetic energy in the simulation ε_{tot} for $\chi^{theory}_{max} \approx 1$. b) The comparison of total radiated energy ε over total initial kinetic energy in the simulation ε_{tot} for $\chi^{theory}_{max} \approx 1$. The axis on the right side shows the energy in Joules for comparison. c) The comparison of total kinetic energy of electrons in Smilei $\varepsilon^{MC}_{Smilei}$ over the total kinetic energy from EPOCH ε^{MC}_{EPOCH} for the MC algorithm (black line).

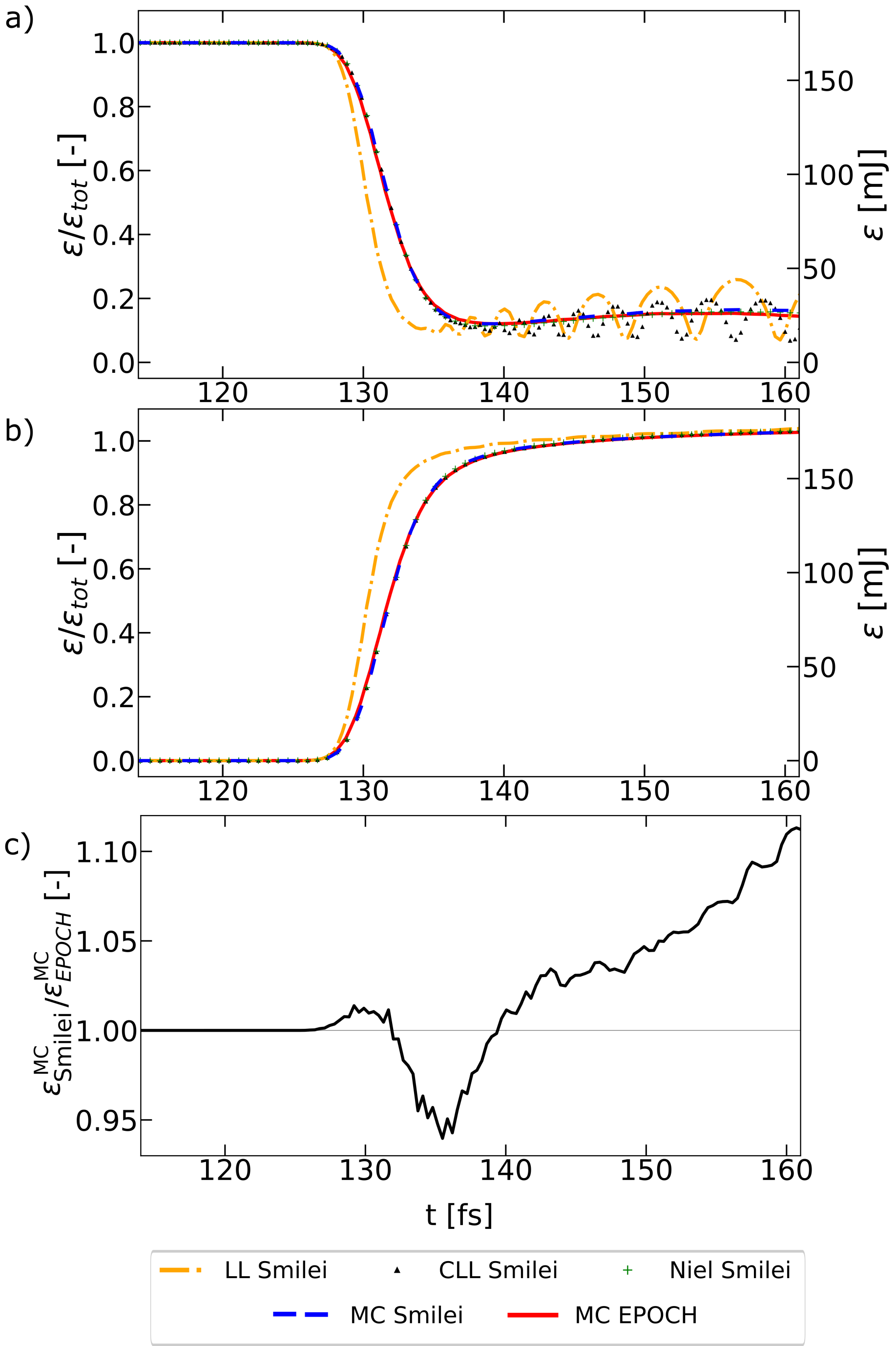


Figure 4.9: a) The comparison of total kinetic energy of electron ε over total initial kinetic energy in the simulation ε_{tot} for $\chi^{theory}_{max} \approx 2.6$. b) The comparison of total radiated energy ε over total initial kinetic energy in the simulation ε_{tot} for $\chi^{theory}_{max} \approx 2.6$. The axis on the right side shows the energy in Joules for comparison. c) The comparison of total kinetic energy of electrons in Smilei $\varepsilon^{MC}_{Smilei}$ over the total kinetic energy from EPOCH ε^{MC}_{EPOCH} for the MC algorithm (black line).

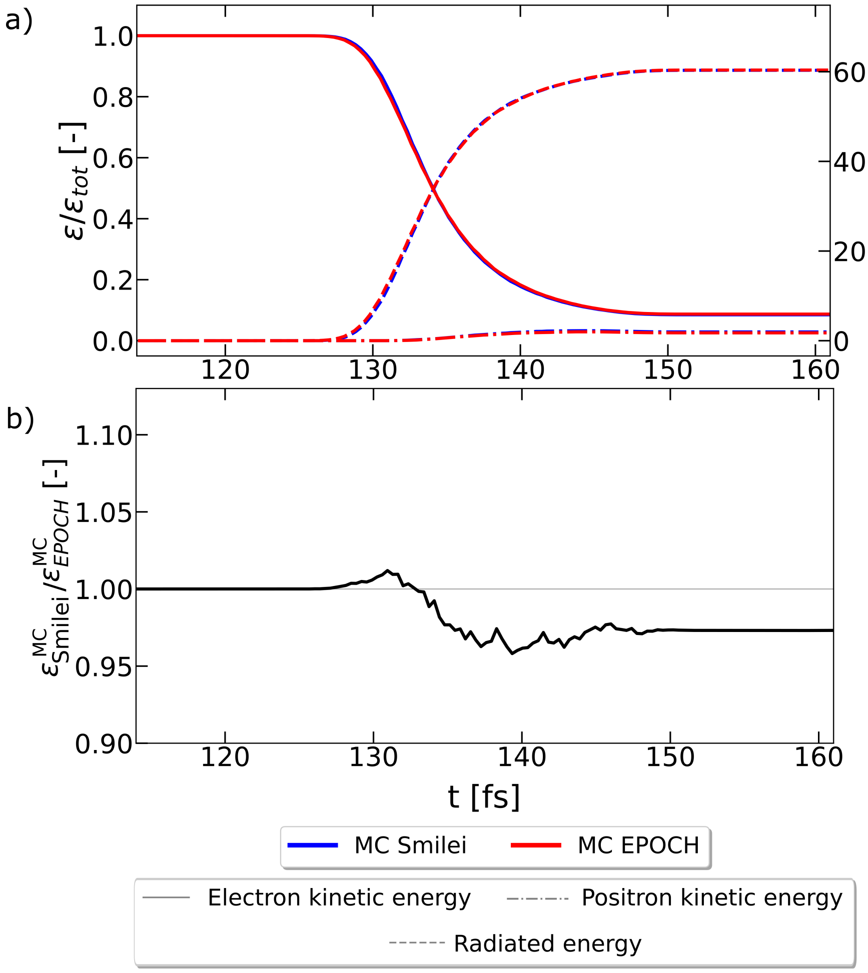


Figure 4.10: a) The comparison of total kinetic energy of electron/positron and total radiated energy ε over total initial kinetic energy in the simulation ε_{tot} for enabled BW algorithm with $\chi_{max}^{theory} \approx 3.8$. b) The comparison of total kinetic energy of electrons in Smilei $\varepsilon^{MC}_{Smilei}$ over the total kinetic energy from EPOCH ε^{MC}_{EPOCH} for the MC algorithm (black line).

4.4 Performance benchmarks

For completeness, we include the performance benchmarks of EPOCH and Smilei PIC codes for both one-dimensional and two-dimensional simulations. These benchmarks are illustrated in Figure 4.11 and Figure 4.12, respectively. The reported times represent the duration of the core algorithm computation without any data output, except a text file containing the main parameters of the computation. In both cases, the Monte Carlo (MC) algorithm was employed to handle radiation reaction. The benchmarks were run on one machine with 4 processors on the Metacentrum computational grid. Parallelization was handled by the MPI protocol.

When the BW algorithm was disabled, the benchmarked scenario was the case when $\chi_{max}^{theory} = 1$, for both 1D and 2D simulations. When the BW algorithm was enabled, the simulation corresponded to $\chi_{max}^{theory} = 3.8$. We conducted five identical simulations for each case. The computed time value represents the statistical mean of the computation time, denoted as $\overline{x} = \sum_1^5 x_i/5$, where x_i denotes the recorded computation time. The standard deviation was computed as $\sigma = \sqrt{\sum_1^5 (x_i - \overline{x})^2/5}$.

In both the 1D and 2D benchmarks, we observed that the computation time for both the EPOCH and Smilei particle-in-cell (PIC) codes was quite similar when BW algorithm was disabled, with Smilei demonstrating slightly faster performance. However, in the case where the BW algorithm was enabled, we noted that EPOCH outperformed Smilei by almost 30% in the 1D scenario and by 10% in the 2D scenario.

Several factors could be responsible for this difference, including inefficient parallelization for the utilized computational grid (Metacentrum), differencies in the implemented optimization settings between the two codes, and other underlying differences in their respective algorithms. Further optimization of the parallelization in the Smilei code by using the OpenMP protocol could also potentially improve the Smilei performance.

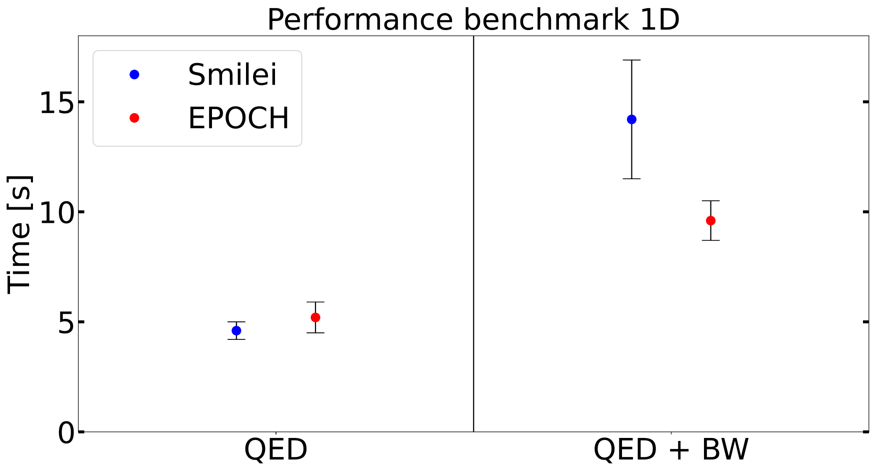


Figure 4.11: Performance benchmarks for 1D simulations with disabled and enabled BW algorithms.

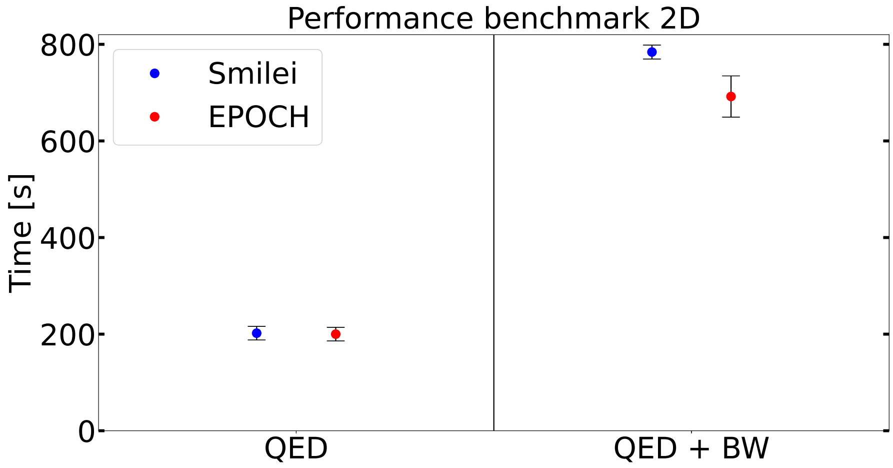


Figure 4.12: Performance benchmarks for 2D simulations with disabled and enabled BW algorithms.

Conclusion

This thesis is focused on the generation of radiation during laser and particle beam interaction in particle-in-cell codes. These radiation reaction recoil effects will have a great importance in laser-plasma interactions in the next generation of PW laser facilities.

The first chapter describes the core of the PIC algorithm. This theory starts with Maxwell-Vlasov model of plasma description, continues with the definitions of macro-particles and explains all steps used in the basic PIC loop, including the interpolation of electromagnetic field, the particle pusher, the current deposition algorithm and Maxwell solvers based on the Yee grid. A brief description of grid computing and a summary of popular PIC codes is also provided. The second chapter contains the theory of radiation reaction. This includes the derivation of the LAD equation from the Lorentz force, the solution of the classical approach in terms of the Landau-Lifshitz equation, the quantum parameter and quantum correction. Also, a short description of the process for Breit-Wheeler pair creation is provided. In the third chapter, various regimes of radiation reaction are described. The corresponding algorithms implemented in PIC codes for these regimes are illustrated. Furthermore, the model for implementing Breit-Wheeler pair creation is explained.

In the 4th chapter, we compared different algorithms for the modelling of radiation reaction implemented in Smilei (LL, CLL, Niel, MC) and EPOCH (MC) PIC codes. In our simulations, an ultra-relativistic electron beam is made to collide with a counter-propagating planar electromagnetic wave. This setup is very effective for triggering the radiation reaction effects. To simulate different radiation reaction regimes, four different scenarios were simulated, with the maximum possible values of the quantum parameter ranging from $\chi_{max}^{theory} = 0.01$ to 2.6. This was done to simulate classical, semi-quantum and full quantum regime. The simulation outcomes are presented in three graphs for each case: one for the evolution of total kinetic energy of electrons, one for the evolution of total radiated energy, and for the comparison of the electron kinetic energy from Smilei MC algorithm relative to that of EPOCH. Lastly, the comparison of the implementation of the Breit-Wheeler pair production is presented, with the evolution of total electron and positron kinetic energy, total radiated energy in the first graph and the comparison of the electron kinetic energy from Smilei and EPOCH MC algorithm in the second graph. The results for 1D simulations are presented first in Fig. 4.1 - 4.5, followed by the results from 2D simulations in Fig. 4.6 - 4.10.

The comparison reveals that the LL algorithm is suitable for lower intensity application with maximum quantum parameter value $\chi_{max}^{theory} \approx 0.01$, according to expectations. However, it is not applicable for high-intensity interactions with $\chi_{max}^{theory} \gtrsim 0.1$, since this algorithm overestimates the radiated energy. This was true in both 1D and 2D case. The CLL algorithm show a good agreement in terms of total radiated energy for $\chi_{max}^{theory} \approx 0.01 - 1$ with more accurate algorithms, even without accounting for the stochastic effects that are significant for higher intensities. These results are likewise similar for 1D and 2D. However, for the 2D with $\chi_{max}^{theory} \approx 2.6$, CLL (along with LL) started to show rapid oscillations in total kinetic energy of electrons. We suspect that this effect is due to the fact that electrons started to to oscillate in the laser field during the interaction. However, further analysis would be required to test this hypothesis.

Surprisingly, the Niel algorithm is comparable to the MC algorithms in the whole χ_{max}^{theory} range studied here, capturing stochastic effects of the radiation reaction. This is true in both 1D and 2D. This algorithm can provide a good alternative to computationally demanding MC algorithm, nevertheless more research would be needed to verify this results for higher values of χ_{max}^{theory}.

Outcomes from the MC algorithm implemented in both Smilei and EPOCH were qualitatively comparable in the whole range of χ_{max}^{theory} and for both 1D and 2D geometries. This is not surprising, since both PIC codes implemented the same MC routine. The MC algorithms are well sufficient for all the χ_{max}^{theory} studied here, fully capturing the stochastic effects of radiation reaction. The results of energy balance from both PIC codes are also comparable when pair creation via BW algorithm was enabled. In general, only notable difference between the codes is that the Smilei MC algorithm exhibits a slightly delayed onset of radiation in most cases compared to EPOCH. Performance of the codes was also comparable, with EPOCH showing slightly better performance for the BW pair creation algorithm.

Overall, the results confirm that the MC and Neil algorithms are more accurate in high-intensity scenarios, while the LL and CLL algorithms are suitable for lower intensities. Results also indicate that the choice of the algorithm and its domain of applicability are critical factors in accurately predicting radiation emission in particle-in-cell simulations. Both Smilei and EPOCH implement Monte Carlo algorithms that yield comparable results with high accuracy. However, in contrast to EPOCH, Smilei also incorporates computationally less intensive algorithms (LL, CLL, Niel), which can produce accurate results for lower quantum parameters χ. This allows the user to have a wider range of algorithmic options to choose from, based on the specific needs.

Appendices

Appendix A

Boris pusher algorithm

The Boris algorithm was proposed in 1970 [38]. It uses following substitutions in (1.30)

$$\mathbf{u}^{(n-1/2)} = \mathbf{u}^{-} - \frac{q_p \Delta t}{2m_p}\mathbf{E}_p^{(n)}, \tag{A.1}$$

$$\mathbf{u}^{(n+1/2)} = \mathbf{u}^{+} + \frac{q_p \Delta t}{2m_p}\mathbf{E}_p^{(n)}. \tag{A.2}$$

Inserting this into (1.30), we get

$$\frac{\mathbf{u}^{+} + \frac{q_p\Delta t}{2m_p}\mathbf{E}_p^{(n)} - \mathbf{u}^{-} + \frac{q_p\Delta t}{2m_p}\mathbf{E}_p^{(n)}}{\Delta t} = \frac{q_p}{m_p}\Bigg[\mathbf{E}_p^{(n)} + \frac{\mathbf{u}^{+} + \frac{q_p\Delta t}{2m_p}\mathbf{E}_p^{(n)} + \mathbf{u}^{-} - \frac{q_p\Delta t}{2m_p}\mathbf{E}_p^{(n)}}{2\gamma^{(n)}} \times \mathbf{B}_p^{(n)}\Bigg]. \tag{A.3}$$

Here, the contributions from the electric field $\mathbf{E}_p^{(n)}$ are cancelled. The result is

$$\frac{\mathbf{u}^{+} - \mathbf{u}^{-}}{\Delta t} = \frac{q_p}{m_p\gamma^{(n)}}\mathbf{u}^{(\mathbf{n})} \times \mathbf{B}_p^{(n)}, \tag{A.4}$$

where $\mathbf{u}^{(\mathbf{n})} = (\mathbf{u}^{(\mathbf{n+1/2})} + \mathbf{u}^{(\mathbf{n-1/2})})/2 = (\mathbf{u}^{+} + \mathbf{u}^{-})/2$. The Lorentz factor is set to be constant during the operation, so $\gamma^{+} = \gamma^{-}$. We know the velocity $\mathbf{u}^{(n-1/2)}$, so we also know $\mathbf{u}^{-}$. The implicit equation (A.4) can be solved by inverting a 3×3 matrix. The phase angle in the rotation (A.4) is given by

$$\theta = \frac{q_p \Delta t}{m_p \gamma^{(n)}}\mathrm{B}, \tag{A.5}$$

where $\mathrm{B} = |\mathbf{B}|$.

There are two choices for computing the $\mathbf{u}^{(n+1/2)}$. The one proposed by Boris in the original paper with

$$\mathbf{t} = \tan\frac{\theta}{2}\mathbf{b}_p^{(n)}, \tag{A.6}$$

where $\mathbf{b}_p^{(n)} = \mathbf{B}_p^{(n)}/|\mathrm{B}_p^{(n)}|$ is a unit vector. The subsequent textbooks used the following simplified approximation [26, 39]

$$\mathbf{t}_p^{(n)} = \frac{\theta}{2}\mathbf{b}_p^{(n)} = \frac{q_p\Delta t}{2m_p\gamma^{(n)}}\mathbf{B}_p^{(n)}. \tag{A.7}$$

We call these two procedures the Boris-A and the Boris-B pusher, respectively. We can compute

$$\mathbf{u}^{(n)} = \mathbf{u}^{-} + \mathbf{u}^{-} \times \mathbf{t}^{(n)}. \tag{A.8}$$

Afterwards

$$\mathbf{u}^{+} = \mathbf{u}^{-} + \frac{2}{1 + |\mathbf{t}^{(n)}|^2} \mathbf{u}^{(n)} \times \mathbf{t}^{(n)}. \tag{A.9}$$

This is then used in (A.2) and $\mathbf{u}^{(n+1/2)}$ is computed. The Boris-A pusher handles the rotation accurately. The Boris-B pusher is an approximate form of the original Boris-A pusher and is wildly used in PIC codes, because of its simplicity and lower computational cost [39]. The accuracy of this algorithm was proven in theoretical studies, as well as in practice [80].

Appendix B

Normalized units used in Smilei

For completeness, the Tab. B.1 with dimension-less reference units used in Smilei is included here. The basis of this system is the unit value for the speed of light, elementary charge and electron mass [81].

Reference unit	**SI value**
Electric charge Q_r	e
Electron mass M_r	m_e
Velocity V_r	c
Energy K_r	$m_e c^2$
Momentum P_r	$m_e c$
Time T_r	$1/\omega_r$
Length L_r	c/ω_r
Electric field E_r	$m_e c \omega_r / e$
Magnetic field B_r	$m_e \omega_r / e$
Particle density N_r	$\varepsilon_0 m_e \omega_r^2 / e^2$
Current J_r	ceN_r

Table B.1: The table with the reference units used in Smilei and their correspong SI units. The value of reference angular frequency ω_r is used to give physical dimention to the simulation and can be changed in the postprocess. However, it must be specified in the input file for radiation reaction module to work. Usually, ω_r corresponds to the laser frequency.

Bibliography

[1] M. Vranic et al. “Classical radiation reaction in particle-in-cell simulations”. In: *Computer Physics Communications* 204 (2016), p. 141–151. ISSN: 0010-4655. DOI: `10.1016/j.cpc.2016.04.002`.

[2] *Extreme Light Infrastructure (ELI)*. `https://eli-laser.eu/`. Accessed: 20.04.2023.

[3] B. Cros, B.S. Paradkar, X. Davoine, et al. “Laser plasma acceleration of electrons with multi-PW laser beams in the frame of CILEX”. In: *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 740 (2014). Proceedings of the first European Advanced Accelerator Concepts Workshop 2013, p. 27–33. ISSN: 0168-9002. DOI: `10.1016/j.nima.2013.10.090`.

[4] J. W. Yoon, Y. G. Kim, I. W. Choi, et al. “Realization of laser intensity over 10^{23} W/cm^2”. In: *Optica* 8.5 (2021), p. 630–635. DOI: `10.1364/OPTICA.420520`.

[5] A. Di Piazza et al. “Extremely high-intensity laser interactions with fundamental quantum systems”. In: *Rev. Mod. Phys.* 84 (3 2012), p. 1177–1228. DOI: `10.1103/RevModPhys.84.1177`.

[6] M. Tamburini, F. Pegoraro, A. Di Piazza, et al. “Radiation reaction effects on radiation pressure acceleration”. In: *New Journal of Physics* 12.12 (2010), p. 123005. DOI: `10.1088/1367-2630/12/12/123005`.

[7] E. N. Nerush, I. Yu. Kostyukov, A. M. Fedotov, et al. “Laser Field Absorption in Self-Generated Electron-Positron Pair Plasma”. In: *Phys. Rev. Lett.* 106 (3 2011), p. 035001. DOI: `10.1103/PhysRevLett.106.035001`.

[8] G. Sarri, K. Poder, J. M. Cole, et al. “Generation of neutral and high-density electron–positron pair plasmas in the laboratory”. In: *Nature Communications* 6.1 (2015), p. 6747. ISSN: 2041-1723. DOI: `10.1038/ncomms7747`.

[9] J. M. Cole, K. T. Behm, E. Gerstmayr, et al. “Experimental Evidence of Radiation Reaction in the Collision of a High-Intensity Laser Pulse with a Laser-Wakefield Accelerated Electron Beam”. In: *Phys. Rev. X* 8 (1 2018), p. 011020. DOI: `10.1103/PhysRevX.8.011020`.

[10] B. Cerutti et al. “Gamma-ray flares in the Crab Nebula: A case of relativistic reconnection?a)”. In: *Physics of Plasmas* 21.5 (2014). ISSN: 1070-664X. DOI: `10.1063/1.4872024`.

[11] B. Cerutti, A. A. Philippov, and A. Spitkovsky. “Modelling high-energy pulsar light curves from first principles”. In: *Monthly Notices of the Royal Astronomical Society* 457.3 (2016), p. 2401–2414. ISSN: 0035-8711. DOI: `10.1093/mnras/stw124`.

[12] L.D. Landau, E.M. Lifshitz, and M. Hamermesh. *The Classical Theory of Fields: Volume 2*. Course of theoretical physics. Elsevier Science, 2000. ISBN: 9780750627689.

[13] S. V. Bulanov, T. Zh. Esirkepov, M. Kando, et al. “Lorentz-Abraham-Dirac versus Landau-Lifshitz radiation friction force in the ultrarelativistic electron interaction with electromagnetic wave (exact solutions)”. In: *Phys. Rev. E* 84 (5 2011), p. 056605. DOI: `10.1103/PhysRevE.84.056605`.

[14] V. S. Krivitskiĭ and V. N. Tsytovich. “Average radiation-reaction force in quantum electrodynamics”. In: *Soviet Physics Uspekhi* 34.3 (1991), p. 250. DOI: `10.1070/PU1991v034n03ABEH002352`.

[15] F. Niel, C. Riconda, F. Amiranoff, et al. “From quantum to classical modeling of radiation reaction: a focus on the radiation spectrum”. In: *Plasma Physics and Controlled Fusion* 60.9 (2018), p. 094002. DOI: `10.1088/1361-6587/aace22`.

[16] R. Duclous, J. G. Kirk, and A. R. Bell. “Monte Carlo calculations of pair production in high-intensity laser–plasma interactions”. In: *Plasma Physics and Controlled Fusion* 53.1 (2010), p. 015009. DOI: `10.1088/0741-3335/53/1/015009`.

[17] T. D. Arber, K. Bennett, C. S. Brady, et al. “Contemporary particle-in-cell approach to laser-plasma modelling”. In: *Plasma Physics and Controlled Fusion* 57.11 (2015), p. 1–26. DOI: `10.1088/0741-3335/57/11/113001`.

[18] *Smilei documentation: High-energy photon emission and radiation reaction.* `https://smileipic.github.io/Smilei/Understand/radiation_loss.html`. Accessed: 05.04.2023.

[19] J. Derouillat, A. Beck, F. Pérez, et al. “Smilei : A collaborative, open-source, multi-purpose particle-in-cell code for plasma simulation”. In: *Computer Physics Communications* 222 (2018), p. 351–373. ISSN: 0010-4655. DOI: `10.1016/j.cpc.2017.09.024`.

[20] F.H. Harlow. *A machine calculation for hydrodynamic problems.* Tech. rep. Los Alamos Scientific Laboratory report LAMS-1956, 1956.

[21] J.-L. Vay et al. “Novel methods in the Particle-In-Cell accelerator Code-Framework Warp”. In: *Computational Science & Discovery* 5.1 (2012), p. 014019. DOI: `10.1088/1749-4699/5/1/014019`.

[22] A. Pukhov. “Particle-In-Cell Codes for Plasma-based Particle Acceleration”. In: *CAS - CERN Accelerator School: Plasma Wake Acceleration.* Geneva: CERN, 2016, p. 181–206. DOI: `10.5170/CERN-2016-001.181`.

[23] Y. Min-Gu et al. "Development of 2D implicit particle simulation code for ohmic breakdown physics in a tokamak". In: *Computer Physics Communications* 221 (2017), p. 143–159. ISSN: 0010-4655. DOI: `10.1016/j.cpc.2017.08.009`.

[24] A. R. Poppe, J. S. Halekas, G. T. Delory, et al. "Particle-in-cell simulations of the solar wind interaction with lunar crustal magnetic anomalies: Magnetic cusp regions". In: *Journal of Geophysical Research: Space Physics* 117.A9 (2012). DOI: `10.1029/2012JA017844`.

[25] L. Quanming, Y. Zhongwei, W. Huanyu, et al. "Two-dimensional Particle-in-cell Simulation of Magnetic Reconnection in the Downstream of a Quasi-perpendicular Shock". In: *The Astrophysical Journal* 919.1 (2021), p. 28. DOI: `10.3847/1538-4357/ac18c0`.

[26] C.K. Birdsall and A.B. Langdon. *Plasma Physics via Computer Simulation (1st ed.)* CRC Press, 1991. DOI: `10.1201/9781315275048`.

[27] J. P. Verboncoeur. "Particle simulation of plasmas: review and advances". In: *Plasma Physics and Controlled Fusion* 47.5A (2005), A231. DOI: `10.1088/0741-3335/47/5A/017`.

[28] A. A. Vlasov. "The vibrational properties of an electron gas". In: *Usp. Fiz. Nauk* 93.11 (1967), p. 444–470. DOI: `10.3367/UFNr.0093.196711f.0444`.

[29] B. Jiang, J. Wu, and L.A. Povinelli. "The Origin of Spurious Solutions in Computational Electromagnetics". In: *Journal of Computational Physics* 125.1 (1996), p. 104–123. ISSN: 0021-9991. DOI: `10.1006/jcph.1996.0082`.

[30] P. Hillion. "Beware of Maxwell's divergence equations". In: *Journal of Computational Physics* 132.1 (1997), p. 154–155.

[31] C. Schroeder. *Physics and applications of plasma-based electron accelerators.* `https://conf-slac.stanford.edu/sssepb-2013/sites/conf-slac.stanford.edu.sssepb-2013/files/S3EPB-Schroeder.pdf`. Accessed: 07.04.2023. 2013.

[32] G. Lapenta. "Kinetic Plasma Simulation: Particle In Cell Method". In: 2015. DOI: `10.13140/RG.2.1.3319.2801`.

[33] R. A. Fonseca, L. O. Silva, F. S. Tsung, et al. "OSIRIS: A Three-Dimensional, Fully Relativistic Particle in Cell Code for Modeling Plasma Based Accelerators". In: *Computational Science — ICCS 2002*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, p. 342–351. ISBN: 978-3-540-47789-1.

[34] E. Cormier-Michel, B. A. Shadwick, C. G. R. Geddes, et al. "Unphysical kinetic effects in particle-in-cell modeling of laser wakefield accelerators". In: *Phys. Rev. E* 78 (1 2008), p. 016404. DOI: `10.1103/PhysRevE.78.016404`.

[35] S. Jardin. *Computational Methods in Plasma Physics.* 1st. USA: CRC Press, Inc., 2010. ISBN: 1439810214.

[36] A. Taflove, S. C. Hagness, and M. Piket-May. "Computational Electromagnetics: The Finite-Difference Time-Domain Method". In: *The Electrical Engineering Handbook*. Academic Press, 2005, p. 629–670. ISBN: 978-0-12-170960-0. DOI: `10.1016/B978-012170960-0/50046-3`.

[37] D. Tskhakaya et al. “The Particle-In-Cell Method”. In: *Contributions to Plasma Physics* 47.8-9 (2007), p. 563–594. DOI: `10.1002/ctpp.200710072`.

[38] J. P. Boris. “Relativistic plasma simulation-optimization of a hybrid code”. In: *Proceeding of Fourth Conference on Numerical Simulations of Plasmas* (1970), p. 3–67.

[39] S. Zenitani and T. Umeda. “On the Boris solver in particle-in-cell simulation”. In: *Physics of Plasmas* 25.11 (2018), p. 112110. DOI: `10.1063/1.5051077`.

[40] J.-L. Vay. “Simulation of beams or plasmas crossing at relativistic velocity”. In: *Physics of Plasmas* 15.5 (2008), p. 056701. DOI: `10.1063/1.2837054`.

[41] A. V. Higuera and J. R. Cary. “Structure-preserving second-order integration of relativistic charged particle trajectories in electromagnetic fields”. In: *Physics of Plasmas* 24.5 (2017), p. 052104. DOI: `10.1063/1.4979989`.

[42] R. W. Hockney and J. W. Eastwood. *Computer Simulation Using Particles (1st ed.)* CRC Press, 1988. DOI: `10.1201/9780367806934`.

[43] T.Zh. Esirkepov. “Exact charge conservation scheme for Particle-in-Cell simulation with an arbitrary form-factor”. In: *Computer Physics Communications* 135.2 (2001), p. 144–153. ISSN: 0010-4655. DOI: `10.1016/S0010-4655(00)00228-9`.

[44] K. Yee. “Numerical solution of initial boundary value problems involving Maxwell’s equations in isotropic media”. In: *IEEE Transactions on Antennas and Propagation* 14.3 (1966), p. 302–307. DOI: `10.1109/TAP.1966.1138693`.

[45] J. Villasenor and O. Buneman. “Rigorous charge conservation for local electromagnetic field solvers”. In: *Computer Physics Communications* 69.2 (1992), p. 306–316. ISSN: 0010-4655. DOI: `10.1016/0010-4655(92)90169-Y`.

[46] A. Blinne, D. Schinkel, S. Kuschel, et al. “A systematic approach to numerical dispersion in Maxwell solvers”. In: *Computer Physics Communications* 224 (2018), p. 273–281. ISSN: 0010-4655. DOI: `10.1016/j.cpc.2017.10.010`.

[47] Y. Lu et al. “Time-step dependent force interpolation scheme for suppressing numerical Cherenkov instability in relativistic particle-in-cell simulations”. In: *Journal of Computational Physics* 413 (2020), p. 109388. ISSN: 0021-9991. DOI: `10.1016/j.jcp.2020.109388`.

[48] R. Lehe et al. “Numerical growth of emittance in simulations of laser-wakefield acceleration”. In: *Phys. Rev. ST Accel. Beams* 16 (2 2013), p. 021301. DOI: `10.1103/PhysRevSTAB.16.021301`.

[49] R. Courant, K. Friedrichs, and H. Lewy. “Über die partiellen Differenzengleichungen der mathematischen Physik”. In: *Mathematische Annalen* 100.1 (1928), p. 32–74. ISSN: 1432-1807. DOI: `10.1007/BF01448839`.

[50] H. Burau, R. Widera, W. Hoenig, et al. “PIConGPU: A fully relativistic particle-in-cell code for a GPU cluster”. In: *Plasma Science, IEEE Transactions on* 38 (2010), p. 2831 –2839. DOI: `10.1109/TPS.2010.2064310`.

[51] L. Fedeli, A. Huebl, F. Boillod-Cerneux, et al. "Pushing the Frontier in the Design of Laser-Based Electron Accelerators with Groundbreaking Mesh-Refined Particle-In-Cell Simulations on Exascale-Class Supercomputers". In: Los Alamitos, CA, USA: IEEE Computer Society, 2022, p. 25–36.

[52] I. Foster and C. Kesselman. *The Grid: Blueprint for a New Computing Infrastructure.* Morgan Kaufmann Publishers, 1999. ISBN: 9781558604759.

[53] *Metacentrum documentation: Computing.* `https://docs.e-infra.cz/compute/grid/basics/jobs/`. Accessed: 20.04.2023.

[54] J. D. Jackson. *Classical electrodynamics.* 3rd ed. John Wiley & Sons, 2021. ISBN: 9780471309321.

[55] I. S. Grant and W. R. Phillips. *Electromagnetism.* Manchester Physics Series. Wiley, 1990. ISBN: 9780471927129.

[56] E. Poisson. "An introduction to the Lorentz-Dirac equation". In: *arXiv: General Relativity and Quantum Cosmology* (1999). DOI: `10.48550/arXiv.gr-qc/9912045`.

[57] M. Abraham. "Theorie der Elektrizität. Zweiter Band: Elektromagnetische Theorie der Strahlung". In: *Monatshefte für Mathematik und Physik* 17.1 (1906), A39–A39. ISSN: 1436-5081. DOI: `10.1007/BF01697706`.

[58] P. A. M. Dirac. "Classical theory of radiating electrons". In: *Proc. R. Soc. Lond.* 167.1 (1938), p. 148–169. DOI: `10.1098/rspa.1938.0124`.

[59] P. Kulhánek. "Vybrané kapitoly z fyziky plazmatu; studijní text pro FJFI ČVUT". In: 2017, [cited 23.2.2023]. URL: `https://www.aldebaran.cz/studium/vkpl.pdf`.

[60] Fritz Sauter. "Über das Verhalten eines Elektrons im homogenen elektrischen Feld nach der relativistischen Theorie Diracs". In: *Zeitschrift für Physik* 69.11 (1931), p. 742–764. ISSN: 0044-3328. DOI: `10.1007/BF01339461`.

[61] J. Schwinger. "On Gauge Invariance and Vacuum Polarization". In: *Phys. Rev.* 82 (5 1951), p. 664–679. DOI: `10.1103/PhysRev.82.664`.

[62] A. H. Compton. "A Quantum Theory of the Scattering of X-rays by Light Elements". In: *Phys. Rev.* 21 (5 1923), p. 483–502. DOI: `10.1103/PhysRev.21.483`.

[63] A. Liénard. "Champ électrique et magnétique produit par une charge concentrée en un point et animée d'un mouvement quelconque". In: *L'Éclairage Électrique.* 16 (27 1898), p. 5–14. URL: `https://cnum.cnam.fr/CGI/fpage.cgi?P91.16/0006/80/694/0/0`.

[64] E. Wiechert. "Elektrodynamische Elementargesetze". In: *Annalen der Physik* 309.4 (1901), p. 667–689. DOI: `10.1002/andp.19013090403`.

[65] V. I. Ritus. "Quantum effects of the interaction of elementary particles with an intense electromagnetic field". In: *Journal of Soviet Laser Research* 6.5 (1985), p. 497–617. ISSN: 1573-8760. DOI: `10.1007/BF01120220`.

[66] M. Vranic et al. "Multi-GeV electron-positron beam generation from laser-electron scattering". In: *Scientific Reports* 8.1 (2018), p. 4702. ISSN: 2045-2322. DOI: `10.1038/s41598-018-23126-7`.

[67] Y. B. Zeldovich. “Interaction of free electrons with electromagnetic radiation”. In: *Soviet Physics Uspekhi* 18.2 (1975), p. 79. DOI: `10.1070/PU1975v018n02ABEH001947`.

[68] T. Blackburn. “Radiation reaction in electron–beam interactions with high-intensity lasers”. In: *Reviews of Modern Plasma Physics* 4 (2020). DOI: `10.1007/s41614-020-0042-0`.

[69] A. D. Piazza, K. Z. Hatsagortsyan, and C. H. Keitel. “Quantum Radiation Reaction Effects in Multiphoton Compton Scattering”. In: *Phys. Rev. Lett.* 105 (22 2010), p. 220403. DOI: `10.1103/PhysRevLett.105.220403`.

[70] F. Niel, C. Riconda, F. Amiranoff, et al. “From quantum to classical modeling of radiation reaction: A focus on stochasticity effects”. In: *Phys. Rev. E* 97 (4 2018), p. 043209. DOI: `10.1103/PhysRevE.97.043209`.

[71] T. Blackburn. “QED effects in laser-plasma interactions”. PhD thesis. Oxford University, UK., 2015.

[72] A. R. Bell and J. G. Kirk. “Possibility of Prolific Pair Production with High-Power Lasers”. In: *Phys. Rev. Lett.* 101 (20 2008), p. 200403. DOI: `10.1103/PhysRevLett.101.200403`.

[73] Ch. Harvey et al. “Covariant worldline numerics for charge motion with radiation reaction”. In: *Physical Review D* 83.7 (2011). DOI: `10.1103/physrevd.83.076013`.

[74] M. Chen et al. “Radiation reaction effects on ion acceleration in laser foil interaction”. In: *Plasma Physics and Controlled Fusion* 53.1 (2010), p. 014004. DOI: `10.1088/0741-3335/53/1/014004`.

[75] C.P. Ridgers, J.G. Kirk, R. Duclous, et al. “Modelling gamma-ray photon emission and pair production in high-intensity laser–matter interactions”. In: *Journal of Computational Physics* 260 (2014), p. 273–285. ISSN: 0021-9991. DOI: `10.1016/j.jcp.2013.12.007`.

[76] *Epoch documentation: QED block.* `https://epochpic.github.io/documentation/input_deck/input_deck_qed.html#fn:2`. Accessed: 07.04.2023.

[77] M. Lobet. “Effets radiatifs et d’électrodynamique quantique dans l’interaction laser-matière ultra-relativiste”. PhD thesis. University of Bordeaux, FR., 2015.

[78] *Smilei documentation: Multiphoton Breit-Wheeler pair creation.* `https://smileipic.github.io/Smilei/Understand/multiphoton_Breit_Wheeler.html`. Accessed: 07.04.2023.

[79] *Input files (Github).* `https://github.com/rikoslav/DP-Input-files-Smilei-EPOCH`. Created: 27.04.2023.

[80] H. Qin et al. “Why is Boris algorithm so good?” In: *Physics of Plasmas* 20.8 (2013), p. 084503. DOI: `10.1063/1.4818428`.

[81] *Smilei documentation: Units.* `https://smileipic.github.io/Smilei/Understand/units.html`. Accessed: 15.04.2023.