Czech Technical University in Prague

Faculty of Nuclear Sciences and Physical Engineering

Department of Physics

Programme: Applied natural sciences
Branch of Study: Physics and Technology of Nuclear Fusion

Reduction of angular divergence of laser-driven ion beams during their acceleration and transport

RESEARCH PROJECT

Author:	Bc. Martina Žáková
Supervisor:	Jan Pšikal, PhD.
Consultant:	Daniele Margarone, PhD.
Submitted in:	September 2014

Prohlášení

Prohlašuji, že jsem svůj výzkumný úkol vypracovala samostatně a použila jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v přiloženém seznamu.

V Praze dne ....

Bc. Martina Žáková

Acknowledgements

I would like to thank my supervisor Jan Pšikal, PhD. and consultant Daniele Margarone, PhD. for providing expert advices, kind comments, willingness and friendly approach during this work.

Název práce:

Omezení úhlového rozptylu iontových svazků řízených laserem během jejich urychlování a transportu

Autor: Bc. Martina Žáková

Obor: Fyzikální inženýrství

Druh práce: Bakalářská práce

Vedoucí práce: Jan Pšikal, PhD.
ELI-Beamlines Project; Institute of Physics of the Academy of Sciences of the Czech Republic, Czech Technical University in Prague Faculty of Nuclear Sciences and Physical Engineering (department of physical electronics)

Konzultant: Daniele Margarone, PhD.
ELI-Beamlines Project; Institute of Physics of the Academy of Sciences of the Czech Republic

Abstrakt: Cílem této práce je teoretické studium urychlování iontů a rozptylu (divergence) generovaných iontových svazků při interakci ultraintenzivních femtosekundových impulzů s velmi tenkými fóliemi, případně i se sofistikovanějšími terči (např. zahnutou fólií, která svou geometrií příznivě ovlivňuje divergenci generovaného svazku). Ke studiu je využito částicových particle-in-cell simulací, jejichž teoretické pozadí je v textu detailně vysvětleno, a následného propojení s již zpracovaným Matlab kódem pro výpočet trajektorií nabitého částicového svazku při průletu magnetickým solenoidem. Výsledkem je série komentovaných grafů a tabulek shrnujících výsledky provedených simulací. Obsaženy jsou také výpočetní a vizualizační skripty, stejně tak, jako upravený Matlabovský kód. Tato práce je důležitá pro realizaci a optimalizaci budoucích experimentů v rámci projektu ELI-Beamlines, ale i experimentů prováděných v institutu GIST v Gwangju, Korea.

Klíčová slova: omezení svazkové divergence, laserem řízené iontové svazky, Particle-in-cell simulace, urychlovací mechanismy, emitance

Title:

Reduction of angular divergence of laser-driven ion beams during their acceleration and transport

Author: Bc. Martina Žáková

Abstract: The aim of this work is the theoretical study of ion acceleration and scattering (divergence) of ion beams generated by the interaction of ultra-intense femtosecond pulses with very thin foils, or with more sophisticated targets (eg. curved foils, which positively affects generated beam divergence due to their geometry). The study is performed using particle-in-cell simulations, whose theoretical background is explained in detail in the text, and subsequently already processed Matlab code to calculate the trajectories of charged particle beam during its path through a magnetic solenoid. Outputs are series of commented figures and tables summarising the results of done simulations. The papper contains computing and visualising scripts as same as the code of modified Matlab program. This work is important for the implementation and optimization of future experiments at the ELI-Beamlines and also experiments carried out in the Institute of GIST in Gwangju, Korea

Key words: decreasing beam divergence, laser-driven ion beams, Particle-in-cell simulations, acceleration mechanisms, emittance

Contents

Introduction

Laser driven proton acceleration became more and more interesting for future applications – laser triggering, nuclear reactions control, production and probing of warm dense matter, "fast ignition" of Inertial Confinement Fusion targets, cancer treatment etc., [26]. In contrast to electrons and X-rays, protons and light ions deliver the most of their energy at the end of their path (Bragg peak). This is very convenient for applications in hadrontherapy because the particles can pass through healthy tissue without damaging it [42].

Currently, the mechanisms of laser driven acceleration are being improved to obtain higher energies per nucleon up to hundreds MeV. New generation of lasers will allow to achieving much higher intensities than the present ones, thus having the way towards future applications. The recent research is focused on decreasing beam divergence which is crucious parameter not only in medical applications such as hadrontherapy. There are many ways how to achieve more collimated beams such as applying various designs of targets [3], [8], [25], placing magnetic beam devices in the way of particles [13], [28], [29] or using ultra-intense short pulse regime of laser [14] and much more. Those approaches can be studied with aim of computer simulations such as Particle-in-cell method providing valuable information for experiments preparation.

This work builds on my bachelor degree project [42] and expands the knowledge about ion accelerating mechanisms and other related physical phenomenas. Moreover, it describes the theoretical background of particle in cell simulations and combines their results about reduction of divergence with Matlab program already presented in my bachelor thesis [42].

Chapter 1

Laser-driven ion beams

1.1 Basic theoretical background

Definition of plasma

Plasma is a quasi-neutral system of charged or neutral particles which are coupled together by their electric and magnetic fields and show collective behavior (but it may not dominate).

Quasi-neutrality

Quasi-neutral system is a system whose total charge in small volume (at least one Debye length which is a distance over which quasi-neutrality may break down, see 1.1) is equal to zero. Mathematically:

$$n_e = Zn_i, \tag{1.1}$$

where n_e and n_i denote electron and ion density, respectively. Z is the average charge state, i.e. mean ion charge.

Plasma frequency

One cannot deal with quasi-neutrality of very fast phenomenas, because charges are separated for a very short moment. Thus, a simple condition must be taken into account:

$$\tau \gg \omega_{pe}^{-1}, \tag{1.2}$$

where τ is characteristic time. From electron equation of motion and LHO differential equation one can derive electron plasma frequency ω_{pe} [24]:

$$v = \frac{\mathrm{d}\Delta}{\mathrm{d}t} \quad \Rightarrow \quad m_e\frac{\mathrm{d}v}{\mathrm{d}t} = -eE = -e\frac{\sigma}{\varepsilon_0} = -\frac{e^2 n_e \Delta}{\varepsilon_0} \quad \Rightarrow \quad \frac{\mathrm{d}^2\Delta}{\mathrm{d}t^2} + \frac{e^2 n_e}{\varepsilon_0 m_e}\Delta = 0, \tag{1.3}$$

$$\omega_{pe} = \sqrt{\frac{e^2 n_e}{\varepsilon_0 m_e}}, \tag{1.4}$$

where σ is charge per unit area, $E = \sigma/\varepsilon_0$, ε_0 is vacuum permitivity and n_e is electron density satisfying the Boltzmann law:

$$n_e = n_{e0}\text{exp}\left(-e\varphi(x)/(k_B T_e)\right), \tag{1.5}$$

where $\varphi(x)$ is electrostatic potential.
Similarly, ion plasma frequency is defined as:

$$\omega_{pi} = \sqrt{\frac{n_i Z^2 e^2}{m_i \varepsilon_0}} = \sqrt{\frac{Z m_e}{m_i}}\omega_{pe}. \tag{1.6}$$

Critical Density

Depending on whether the incident laser frequency is lower or higher than ω_p, the plasma is so-called overdense or underdense, respectively. The density known as critical denotes the boundary between underdense and overdense plasma and is given by cut-off frequency where the light frequency is equal to the plasma frequency ω_p:

$$\omega = \omega_p \quad \Rightarrow \quad \omega^2 = \frac{e^2 n_c}{\varepsilon_0 m_e}, \tag{1.7}$$

$$n_c = \frac{\varepsilon_0 m_e}{e^2}\omega^2. \tag{1.8}$$

Electromagnetic (EM) waves with frequency higher than plasma frequency ω_p interact with underdense plasmas conductively in contrast to EM waves with frequency lower than ω_p which assign dieletric behavior with overdense plasmas (the inertia of electrons retards their response). As a result, the underdense plasma is rather transparent to the radiation [32].

Macroscopic parameters, Maxwell distribution function

In fact, the plasma is usually described by macroscopic parameters - electron and ion densities and temperatures n_e, n_i and T_e, T_i, respectively. The temperature can be defined for a thermodynamic equilibrium via the Maxwellian distribution function $f(\varepsilon_{kin})$ in kinetic energy [32]:

$$f(\varepsilon_{kin}) = \frac{2}{\sqrt{\pi}(k_B T)^{3/2}}\sqrt{\varepsilon_{kin}}\text{exp}\left(-\frac{\varepsilon_{kin}}{k_B T}\right). \tag{1.9}$$

Kinetic energy can be also expressed by equipartition theorem. Assuming that, the relation for mean thermal velocity of electrons v_{te} moving in one direction is:

$$\varepsilon_{kin} = \frac{1}{2}m_e v_{te}^2 = \frac{1}{2}k_b T_e \quad \Rightarrow \quad v_{te} = \sqrt{\frac{k_B T_e}{m_e}}, \tag{1.10}$$

where k_B is Boltzmann constant and m_e means electron mass.

Collision frequency

The thermal equilibrium is ensured by collisions. The rate of the momentum transfer between particles is expressed by collision frequencies. We divide collisions according to their collision angle into scattering at large angles (bigger than 90°) or small angles which are more frequent in plasma. For both cases the collision frequency could be derived with assumption that electron velocity is bigger than termal velocity $v_0 \gg v_{Te}$ [24]:

$$\nu_{c\ large} = \frac{4\pi n_0}{(4\pi\varepsilon_0)^2}\frac{q^2 q_0^2}{m^2 v_0^3}, \tag{1.11}$$

$$\nu_{c\ small} = \frac{8\pi n_0 e^4}{(4\pi\varepsilon_0)^2 m_e^2 v_0^3}\ln\Lambda, \tag{1.12}$$

where v_0 is electron speed, $\ln\Lambda$ is Coulomb logarithm and n_0 is particle density. These relations are the most frequent ones, but in fact, collision frequnecy can be derived for each action by statistical approach from Fokker-Planck (FP) equation (the solution process is demonstrated in [21]). Collision frequency does not relly on velocity when it is slow, e.g. flow of electric current. On the contrary, for rapid particles the colission frequency ν_c decreases rapidly with increasing velocity and scales as:

$$\nu_c \sim v_0^{-3}, \tag{1.13}$$

which is actually case of collisional frequencies (1.11) and (1.12). Thus, for high speed particles we can work under approximation of collisionless plasma.

Debye length

Charged plasma particles are influenced by generated electric field which can be decomposed into two components – average field and fluctuations over Debye length λ_D, i.e. $\vec{E}_{micro} = \langle\vec{E}\rangle + \delta\vec{E}$. Debye length is length over which the field contribution of single charge is shielded by surrounding electrons [32].

Charges can be spontaneously separated only at a distance allowed by their thermal energy, i.e. the distance, where all the heat energy changes into potential one. Taking Δ as a thickness of electrons layer moving from ion background, the potential energy of electron is equal to its thermal energy, when it moves over one Δ [24]:

$$U_{pot} = -eE\Delta = \frac{e^2 n_e \Delta^2}{\varepsilon_0} = k_B T_e. \tag{1.14}$$

Then, the electron Debye length is given by:

$$\lambda_{De} \equiv \Delta = \sqrt{\frac{\varepsilon_0 k_b T_e}{n_e e^2}}. \tag{1.15}$$

The number of particles within the so-called Debye sphere is called plasmatic parameter N_D and is given by:

$$N_D = \frac{4\pi}{3}\lambda_{De}^3 n_e. \tag{1.16}$$

When $N_D \gg 1$ we are talking about *ideal plasma.*

Collective behavior

The term collective behavior denotes particle interaction by macroscopic electromagnetic fields unlike microscopic ones by which the particles interact with binary collision. Collective action is characterized by electron plasma frequency ω_{pe} which is stronger than the binary operation characterized by collision frequency ν_c, i. e. $\omega_{pe} > \nu_c$ [24].

1.2 Accelerating mechanisms of ion beams in solid targets

Apart from conventional acceleration techniques there is also a possibility to produce particle beams by ion acceleration based on high intensity laser interaction with matter. These beams are comparable with those from conventional accelerators in terms of brightness, pulse duration, emittance and compact source size. Due to their parameters (e.g. energy, divergence, ...) they can be used for various applications including fast ignition, proton radiography, development of compact facilities for laser-driven ion beam radiotherapy, hadrontherapy (a healing of radiation-resistant tumors), nuclear research and so on [29].
The main problem the laser accelerators have to face is producing of divergent proton beams with high energy spread, which caused a big decrease in the number of particles delivered for therapy or other applications. Thus, the main goal is to avoid this divergence.
There are two main acceleration scenarios able to explain the observation of fast ions in a typical experiment on femtosecond laser pulse interaction with solid foil targets. Both of them are demostrated in Fig. 1.1. In the first scenario called Radiation Pressure Acceleration (RPA), the electrons are pushed into the target by ponderomotorive force (i.e. by the radiation pressure of the incident laser beam) and the ions are accelerated from the target front side by the generated electrostatic field. The force is so strong that it can pushes an overdense target inwards, which causes a sharp growth in density profile and changes the shape of its surface. This phenomena is known as Hole boring. RPA could be more efficient with circularly polarized laser pulses [43], [20]. The second scenario which is based on a generation of hot electrons is called Target Normal Sheath Acceleration (TNSA) and is very common in experiments. In nutshell, a laser pulse generates a population of very energetic hot electrons which can pass through the target and cause unbalance in charge on target rear side. The resulting electrostatic field finally leads to ions acceleration.

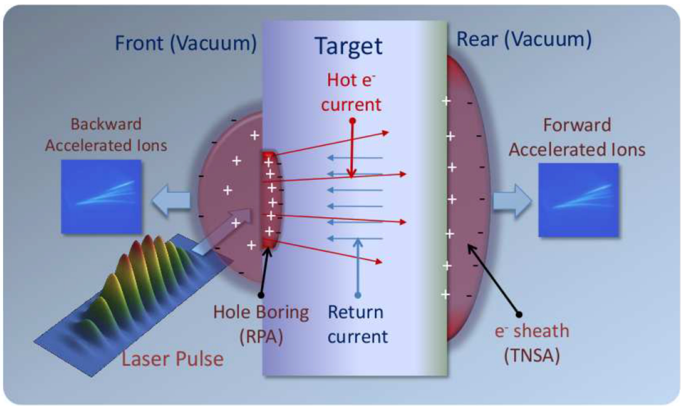


Figure 1.1: A simple sketch of laser-driven ion acceleration from thin foils; [26]

For experiments thin metal or insulator foil targets are usually used due to their easy characterisation and positioning. Ions accelerated in such targets are mainly protons originated from low-Z hydrocarbon or water deposits which can be removed e.g. by target heating or laser ablation. Experiments have demonstrated acceleration of protons to almost 160 MeV, having a range $17,4$ cm in water, which is almost suffucient for proton cancer therapy requiring $150 - 250$ MeV protons. The result was obtained with nano-CH_2 targets using 150 TW laser in the relativistically transparent regime [17]. Moreover, the laser system is capable of generating the full 250 MeV in conformity with the obtained scaling laws [17].

1.2.1 Target Normal Sheath Acceleration (TNSA)

The proton acceleration by an ultraintense laser pulse ($\sim 10^{20}$ W/cm^2) was explained by the Target Normal Sheath Acceleration (TNSA) model [35]. The scenario is demonstrated in Fig. 1.2. Firstly, a very intense current of hot electrons can be generated on the front side of the target. This current may or may not pass through the target. If the hot electrons reach its rear side they cause an unbalance in charge resulting in a strong electrostatic potential. For better understanding, we can imagine this as a double layer of positive (ions) and negative (electrons) charges, which generates an electrostatic field. As soon as electrons reach the rear side and pass the target-vacuum boundary they can be attracted back to the target. In fact, the most energetic ones escape, but the majority of electrons return back to the target (because of acceleration by electrostatic power) or even reach the front side where the TNSA mechanism can be observed as well or they start the cycle again. Due to the electrostatic field the ions on the target rear side will be ionized and accelerated. The direction of the accelerated ions from the rear side is normal to the target (forward acceleration following the direction of the electric field), therefore the mechanism is commonly known as Target normal sheath acceleration (TNSA). The accelerated ions leave the target together with comoving electrons forming a quasineutral plasma cloud. Because the plasma density in this volume drops dramatically after the detachement from the target and temperature stays high, recombination effects are negligible for propagation lengths in the range of several meters [16]. Typically energy of about $1 - 50$ MeV, scalling with $I^{\frac{1}{2}}$, can be obtained [26].

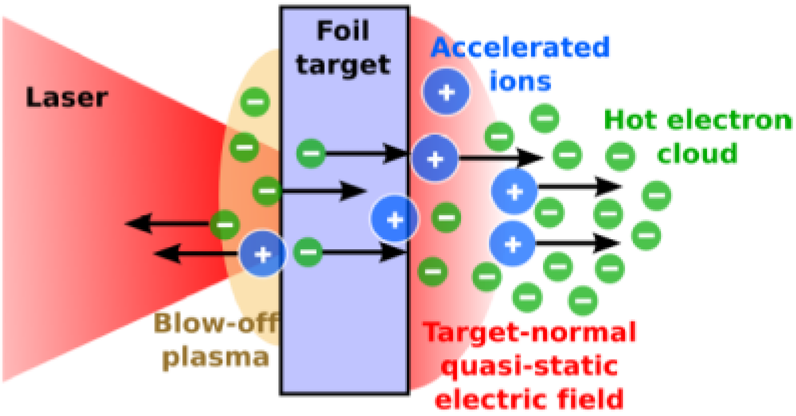


Figure 1.2: Target normal sheath acceleration; [32]

In fact, the species of accelerated ions depends on the purity of the target. If the target contains a water or hydrocarbon contaminants on its surface, protons are accelerated the most because to its the highest charge-to-mass ratio. Thus, for efficient acceleration of heavy ions one must have a very cleaned target which we can obtain by various techniques like target heating or laser ablation [12].

TNSA can be mathematically described by the model of free isothermal expansion into a vacuum of a plasma occupying initially a space in front of the target ($x < 0$). From equation of motion, continuity and Poisson equation (1.18),(1.19), (1.17) with assumption of Boltzmann equilibrium we will get the self-similar solution in the form of rarefaction wave (1.20). This solution is valid on the scale length larger than the Debye radius, where assumption of quasineutrality is valid:

$$\Delta\phi = \frac{\rho}{\epsilon_0} = \frac{e}{\epsilon_0}\left(n_e - Zn_i\right), \tag{1.17}$$

where ϕ is electric potential, e is elementary charge, ϵ_0 permitivity, Z is atomic number, n_e is electron density following the Boltzmann distribution $n_e = n_{e0} \cdot \exp(e\phi/T_e)$ and n_i is ion density following $n_i = 0$ for $x < 0$ or $n_i = n_{i0}$ for $x > 0$. Ion expansion into a vacuum is described via combination of continuity and motion equations:

$$\left(\frac{\partial}{\partial t} + v_i\frac{\partial}{\partial x}\right) n_i = -n_i\frac{\partial v_i}{\partial x}, \tag{1.18}$$

$$\left(\frac{\partial}{\partial t} + v_i\frac{\partial}{\partial x}\right) v_i = -\frac{Ze}{m_i}\frac{\partial\phi}{\partial x}. \tag{1.19}$$

Now, self-similar variable is defined as $\xi = x/t$ and the quasineutrality condition $n_e = Zn_i$ is taken into account. Then, the solution of set of equation (1.17) - (1.19) describes rarefaction wave:

$$Zn_i = n_{e0} \cdot \exp(-\xi/c_s - 1), \qquad v_i = c_s + \xi, \qquad e\phi = -T_e\left(\xi/c_s + 1\right). \tag{1.20}$$

The relation for maximum accelerating electric field is obtained by integrating (1.17) :

$$\vec{E} = \nabla\phi, \tag{1.21}$$

$$E_{ac} = \int_0^\infty \frac{e}{\epsilon_0}\left(n_e - Zn_i\right)\mathrm{d}x = \int_0^\infty \frac{e}{\epsilon_0}\left(n_{e0} \cdot \exp(e\varphi(x)/T_e) - Zn_i\right)\mathrm{d}x, \tag{1.22}$$

$$E_{ac} \approx \frac{T_e}{e\lambda_{D0}} = \sqrt{\frac{n_{e0}T_e}{\epsilon_0}}, \tag{1.23}$$

the variable T_e denotes the product $T_e \equiv T_{el}k_B$, where T_{el} is the electron temperature and k_B is Boltzmann constant, λ_{D0} is Debye length in the unperturbed plasma of electron density n_{e0}.

Because the system acts as a rarefaction wave, its front is moving vith the maximum ion velocity. This maximum speed can be traslated into the cutoff energy of accelerated ions:

$$\varepsilon_{imax} \approx 2ZT_e ln^2\left(\tau + \sqrt{\tau^2 + 1}\right), \tag{1.24}$$

and subsequently into total energy of accelerated ions [32]:

$$W_{itot} = \int \varepsilon_i \frac{\mathrm{d}N}{\mathrm{d}\varepsilon_i}\mathrm{d}\varepsilon_i = ZT_e n_{i0} c_s t_{acc}, \tag{1.25}$$

where t_{acc} is ion acceleration time. Moreover, the laser-to-ion conversion efficiency can be expressed as the ratio of W_{itot} and the laser pulse energy ε_{Ltot}:

$$\eta_i = W_{itot}/\varepsilon_{Ltot}. \tag{1.26}$$

In TNSA regime, the maximum ion/proton energy is determined by the hot electrons (see relation (1.25)) and consequently, the maximum proton energy is dependent on the peak intensity of a laser pulse with scaling $\sim I^{\frac{1}{2}}$.

1.2.2 Radiation pressure acceleration

There is also a possibility of developing a significant contribution to acceleration at the front surface of the target, where the mechanisms act on ions at the front side in the vicinity of the laser focus. The main idea is that the intense radiation pressure of the laser pulse, represented by ponderomotive force, pushes an overdense target inwards, which causes a sharp growth in density profile and changes the shape of its surface. This process is known as "hole boring". In other words, when the laser pulse reaches the relativistic critical surface it forces the target electrons inwards and electrostatic field is growing as a result of charge separation balancing the ponderomotorive force. In fact, this can be expressed by balance between total radiation and electrostatic pressure as follows:

$$\frac{1}{2}\epsilon_0 E_{es}^2 = \frac{1+R}{c} I_L \approx \frac{2}{c} I_L, \tag{1.27}$$

where E_{es} is electric field caused by charge separation, I_L is laser pulse intensity, R is target reflectivity (usually taken as $R \approx 1$) , ϵ_0 is vacuum permeability and c is speed of light.
When the electrons are pushed into the target, the ions due to the charge separation on the front side are free to be accelerated by electrostatic field E_{es} and their maximum energy is:

$$\varepsilon_{imax} = \frac{Z m_e c^2 a_0^2}{m_i \gamma_L}, \tag{1.28}$$

where Z is atomic number, m_e is mass of electron, m_i is mass of ion, c is speed of light, a_0 is parametr expressed as $a_0 = eE/m_e c\omega$ and γ_L is a gamma factor of the laser [32].

In contrast with TNSA we get energy scaling with laser intensity I instead of $I^{\frac{1}{2}}$, which is clearly a great advantage. On the other hand, the front surface acceleration is expected to produce a large-divergent ion beams due to the curved critical density interface where the charge separation is present (hole boring). The "hole boring" acceleration is considered as a particular regime of the radiation pressure acceleration (RPA). Another aspect which have to be taken into account is polarization of the laser pulse. If the laser pulse is linearly polarized, many experiments demonstrate that TNSA produces higher energy particles with smaller divergence and a higher efficiency than RPA [32]. Nevertheless, radiation pressure acceleration mechanism preveals with circularly polarized laser beams at normal incidence on a foil, which can suppress most electron heating mechanisms such as resonance absorption, vacuum heating, $\vec{j} \times \vec{B}$ heating etc.
RPA regime is suitable for accelerating ions by a laser pulse with peak intesity $>\sim 10^{23}$ W/cm^2. In addition this intensity can be lowered to $\sim 10^{21}$ W/cm^2 by focusing a circularly polarized laser pulse on a nanometer-thick target. RPA-dominated regime is suggested for the monoenergetic proton/ion generation. The main goal is to obtain a monoenergetic proton beams at a lower intensity ($\sim 10^{20} - 10^{21}$ W/cm^2) even with a linear polarization [35].

1.3 Electron acceleration mechanisms at the critical surface

Target normal sheath acceleration is mostly observed ion acceleration mechanism in experiments and hot electrons play a crucial role in it. Thus, studying electron accelerating mechanisms have a great importance.
The heating mechanisms can be divided into two groups depending on the laser intensity:

- If the laser intensity is below 10^{15} W/cm^2, the plasma is heated by *electron-ion collisions* and absorption mechanisms such as collision absorption, normal and anomalous skin effect cause heating of all electrons. Relatively long time is needed for efficient heating, thus, the long laser pulse times ($\sim$ ns) are applied. Then, more than 80% of the laser pulse energy can be delivered into plasma.
- If the laser intensity is higher than 10^{16} W/cm^2, the plasma is heated by *collisionless absorption mechanisms.* If the plasma has step-like or very steep density profile, the absorption of laser energy takes place due to Brunel vacuum heating or $\vec{j} \times \vec{B}$ heating. Naturally, if the plasma has not step-like or very steep density profile, the resonant absorption plays a crucial role. In contrast to collisional heating mechanisms, in colisionless ones only a part of electrons gain most of absorbed energy; such population of fact electrons is called *hot.*

Hot electrons

Relaxation time of hot electrons is large compared to the plasma expansion time. Thus, after interaction two temperatures T_h, T_c (hot and cold electrons) are considered with Boltzmann distribution for electron density:

$$n_e = n_h + n_c = n_{h0}\text{exp}(e\varphi/T_h) + n_{c0}\text{exp}(e\varphi/T_c), \tag{1.29}$$

where φ is electrostatic potential and T_h can be found experimentally or can be estimated from ponderomotive potential [32].

1.3.1 Brunel vacuum heating

Laser pulse is absorbed for instance by Brunel vacuum heating. In this case, the p-polarized laser pulse (polarized light with its electric field along the plane of incidence) is obliquely incident on the surface, thus, the electric field can pull electrons out of the steep plasma. This mechanism works for high laser intensities, because amplitude of oscillating electrons driven by electric laser field is larger than the density scale length, so electrons will bring away and transform the energy of laser pulse to the kinetic energy of the plasma when they reach the overdense plasma region [9], [32].
Assuming that the overdense region is located $z < 0$, the electric field of obliquely incident laser wave has two components – oscillating perpendicular or parallel to the surface. Then, electric field is given by:

$$E_z(t) = E_p \sin(\omega t + \phi) = E_0 \sin(\alpha) \sin(\omega t + \phi), \tag{1.30}$$

where E_p is the oscillating electric field perpendicular to the plasma surface, E_0 is a uniform static electric field and α is the incidence angle of the wave.

The energy deposed by Brunel vacuum heating is transported by hot electrons in the bunches ejected once per laser period. Firstly, the electrons are pushed inside the plasma, where they obtain only a small amount of energy because of the electric field which is strongly attenuated in plasma (case $(\omega t + \phi) \in (0; \pi)$ and $E_z > 0$). Contrarily, electrons gain very high energy in the second half laser period when they are ejected into vacuum (case $(\omega t + \phi) \in (\pi; 2\pi)$ and $E_z < 0$). Time of electron expulsion influences electron trajectory. Moreover the self-consistent electric field is created, when many electrons are ejected simultaneously. As a consequence of oscillating laser field and self-consistent electric field, the most electrons turn back into the plasma without restoring forces behind the skin layer.
The average energy of electrons is proportional to ponderomotive potential $\varepsilon_e \sim U_p$, where $U_p = m_e c^2 \left(\sqrt{1 + a_p^2} - 1 \right)$. In the most cases, the distribution of such electrons is considered to be maxwellian because the electrons are accelerated in different phases of the laser field [32].
When electric field $\sim E_p$ and energy $\sim U_p$ one can write relations for the number of ejected electrons δN_e and the thickness of the ejected layer d as following:

$$E_p \simeq e\delta N_e d/\varepsilon_0, \qquad eE_p d \simeq U_p. \tag{1.31}$$

In Fig. 1.3 angular dependence of vacuum heating predicted by Brunel model is shown. Curves depend on the dimensionless amplitude (non-dimensionless is A_0) of the vector potential $\vec{A}$ of the electromagnetic laser wave given by $a_0 = \frac{aA_0}{m_e c} = \frac{eE_0}{m_e c\omega}$.

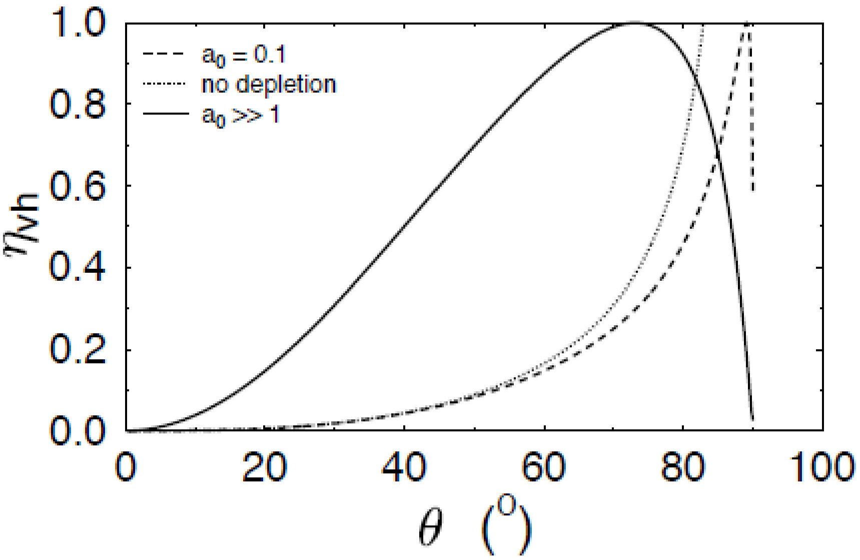


Figure 1.3: Angular dependence of vacuum heating predicted by Brunel model; η_{vh} is efficiency, θ is incidence angle of the wave, a_0 is dimensionless amplitude of the vector potential; [18]

1.3.2 Relativistic $\vec{j} \times \vec{B}$ heating

Contrary to Brunel vacuum heating, $\vec{j} \times \vec{B}$ heating is the electron heating scenario for the normal incidence of laser pulse onto the target, moreover, the oscillating part of the electric

field perpendicular to the plasma surface E_p is zero. This scheme becomes important for relativistic laser pulse intensities i.e. when $a_0 > 1$ (where a_0 is given by $I\lambda^2 = 1,38 \cdot 10^{18} \cdot a_0^2$). In this case, the $\vec{v} \times \vec{B}$ component of the Lorentz force can heat electrons.
Another difference between Brunel and $\vec{j} \times \vec{B}$ heating is that in $\vec{j} \times \vec{B}$ scenario bunches of hot electrons are ejected twice per laser period. In fact, this is the point how distinguish between Brunel and $\vec{j} \times \vec{B}$ heating in numerical simulations.
The force which ejects electrons twice per laser period in the normal direction, acts as follows [32]:

$$F_p \sim \frac{m_e \omega c a_0^2}{\sqrt{1+a_0^2}}. \tag{1.32}$$

Ejected electrons create self-consistent field which, together with the oscillating force of the order of F_p, returns electrons back into the plasma, similarly to Brunel vacuum heating. Moreover, same as in Brunel model, those electrons feel no restoring forces after passing skin layer. For number of ejected electrons δN_e and the thickness of ejected layer d, the following relations are valid:

$$F_p \simeq e^2 \delta N_e d/\varepsilon_0, \qquad F_p d \simeq U_p, \tag{1.33}$$

where U_p is characteristic electron energy; $U_p = m_e c^2 \left(\sqrt{1+a_0^2} - 1\right)$. Similarly to Brunel heating, the average energy of electrons is $\varepsilon_p \simeq U_p$.

1.3.3 Resonance absorption

When a p-polarized laser pulse is obliquely incident on a continuously increasing plasma density, then resonant absorption will dominate as a collisionless absorption mechanism.
Firstly, the laser wave with incidence angle α gets through the underdense plasma. Naturaly, laser wave reach the critical surface, where the electron density can be expressed as $n_e = n_{crit}\cos^2\alpha$, and will be reflected there. Electron plasma wave is resonantly excited by laser field at critical density surface. Subsequently, this wave is damped by various mechanisms – for lower intensities by collisions and Landau damping, for higher intensities by particle trapping and wave breaking [32].

1.4 Divergence of laser-driven ion beams

The ion beams produced form flat taget are typically divergent with half angle $0° - 25°$, depending on proton energy, since the expanding field front on the target rear side is Gaussian in shape [8], [33].

Improvements in the quality of the laser-driven proton beams produced from flat foil target, thus, decreasing their divergence, have been demostrated by small devices installed close to the target [28], [42] or by using various targets. For instance *double layer targets* [11] can produce collimated and monoenergetic proton beams, *curved* [30] or *(micro)structured targets* [25] focuses a divergent beam, where focal legth is determined by the target curvature. Monochromatic collimated laser-driven proton beam can be obtained by using *microlense devices with two synchronized carefully aligned high-intensity laser beams* [36]. Moreover, the *assembled target* — *flat-target* [3] and *microlense attached bellow* [19] are making the proton beam divergence smaller without any additional laser beam. In addition, a *conical cavity target holder* is used to produce quasi-monochromatic pencil beam [29]. Some of these methods will be discussed in next subchapters.

1.4.1 Decreasing divergence by various targets

Decreasing divergence by ultra-thin targets

In comparison to μm targets, more than 10 times reduction in the divergence, when using the nanometer thick foils, was demostrated. Proton beams with extremely small divergence about half angle only 2° was observed. These beams demostrate surprising collimation over the whole energy range and reach 6 MeV where, moreover, typical increasing of divergence with decreasing energy is not observed. Furthermore, the similar results were obtained form 2D particle-in-cell simulations which also establish that the small divergence θ is a result of a steep longitudinal electron density gradient [3]:

$$\theta = \arctan\left\langle \frac{\partial n_e}{\partial y} \middle/ \frac{\partial n_e}{\partial z} \right\rangle, \tag{1.34}$$

where the angle brackets denote the average along the trajectory and y and z are the transversal and the longitudinal dimension, respectively. The laser pulse propagates along z-axis with target normal incidence.

In general, experimental results show that reduction in foil thickness implies beams with smaller divergence. The most common experimental approaches are based on TNSA mechanism with μm targets. In addition divergence of such produced ion bunches depends on the electron density and phase space distribution behind the target. Firstly, those parameters are influeced by the laser profile and then, they are changed during the path through the target.

2D PIC simulations confirm experimental results. For instance, divergence is comparably small $(2-4,6^\circ)$ and indicates decreasing value for decreasing target thickness. Moreover, none noticeable divergence dependence on energy was measured. On the other hand, the maximum proton energy slightly depends on the position of the target. In addition, 2D simulations show interisteng scaling laws. The first for divergence and the laser FWHM diameter $\theta \sim (D_L)^{-1/2}$ and the second for divergence and laser intensity with other parameters unchanged $\theta \sim (I_0)^{1/4}$ [3].

Decreasing divergence by curved targets

There is another way how to compensate divergence. Curved back surface of the target causes that the accelerated proton beam will converge. In Ref. [8] ion beam focusing dynamics is studied with picosecond and micrometer time/space resolution. Curved targets are irradiated by a high intensity short pulse laser. Moreover the experiments are coupled with 2D PIC simulations and the main results are [8]:

- $\sim 30\mu$m convergence diameter is observed for ion beam accelerated from 800μm diameter curved target

- significant fillamentation occurs in the converging plasma
- focus location depends on proton energy; most of protons focus at the geometric target center
- the exact location of laser illumination on the curved target modifies the directionality of the ion beam, but not affects the ability to focus

Moreover, 2D PIC simulations suggest that with higher laser intensities, focusing of the proton beam will improve. In addition, experiment as well as simulations demonstrate that lower energy protons have a larger divergence, which is consistent with previous results [8].

Decreasing divergence by targets with micro-structures on their rear side

Using microstructures on the target rear side was considered to be a way how to decrease divergence and, thus, improve the quality of ion beams for applications. Two cases of the laser impact on the rear surface of the target are depicted in Fig. 1.4. Naturaly, when the laser focus is located between the walls, the walls create a shilding electric field which finally leads to much lower divergence typically about $\sim 2°$ (case rear1). In contrary, when the laser focus is located opossite of the wall (thus, lying in a line, corresponding to the case rear2) shielding of the normal direction is formed and protons cannot be accelerated in this direction. This leads to large divergences with half angles about $\sim 10°$. The interpretation of previous two cases gives an important rule: a proton angular distribution is strongly dependent on the relative position of the laser focus and the microstructure on the rear surface. Thus, for larger focal diameter than the microstructure period, the relative position of the laser focus do not affect the angular divergence [25].

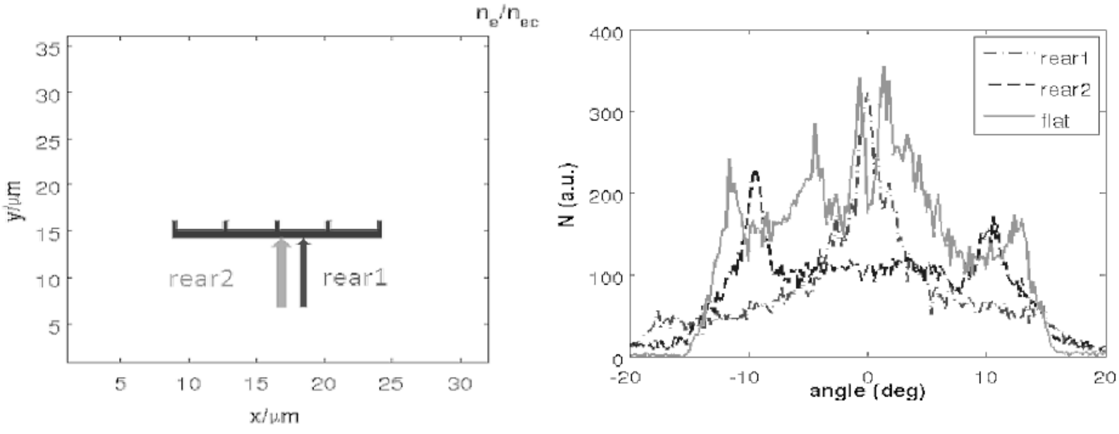


Figure 1.4: (a) Schematic of the foil target with the structure at the rear side (arrows denote focus center position in 2 different simulations); (b) angular distribution of protons accelerated to energy > 30 MeV; [25]

Microstructures at the foil front surface may affect the obtainable ion energies and ion number, more details in Ref. [25].

1.4.2 Decreasing divergence by beam devices

Magnetic Solenoids

Improvements in the quality of the laser-driven proton beams produced from flat-foil target, thus, decreasing their divergence, have been demonstrated by small beam optic devices installed close to the target. For instance, magnetic solenoid can be used in transport system of the beamline. The crucial for understanding this focusing phenomena is to understand the beam dynamics in solenoid magnetic field. A detailed description can be found in Ref. [42].

Focusing of charged beams in a solenoid magnetic field can be described by a simple geometry:
We will assume the charged particle beam, where only four particles on the beam surface (A, B, C, D) will be taken into account for studying beam behavior. Each particle moves on a circular trajectory in the $x - y$ plane which is depicted by dotted lines in Fig. 1.5. Moreover, each particle touches the solenoid axis just once and returns to its circular trajectory. After some time and a certain distance the particles move on their trajectories to new spots. The new positions of the particles are marked as A', B', C', D'. For better understanding there is also a particle E which is not situated on the beam surface at the beginning and its new position E' is in the volume of the new *focused* beam shown as a dashed circle. The radius of the original beam decreases from OA to OA'. This is the main process of periodic focusing in a uniform magnetic field.

In my previous work [42], the using of magnetic solenoid is demonstrated with the help of numerical simulations. A simple matlab program modelling a charged particle beam trajectories and computing emittances in the solenoid was developed. In this matlab code, the trajectories are calculated by equations of motion of charged particles in the magnetic field of solenoid.
Based on the knowledge of initial particle beam emittance, a set of particles in the phase space was generated by a random function. Then, the system of three differential equations of motion in cylindrical magnetic field are solved for initialized particles:

$$m(\ddot{r} - r\dot{\theta}^2) = er\dot{\theta}B_z \qquad \text{focusing,} \tag{1.35}$$

$$m(2\dot{r}\dot{\theta} + r\ddot{\theta}) = e(\dot{z}B_r - \dot{r}B_z) \qquad \text{rotation,} \tag{1.36}$$

$$m\ddot{z} = -er\dot{\theta}B_r \qquad \text{acceleration.} \tag{1.37}$$

The results of the simulation are shown in Fig. 1.6. In this case, the beam of 100 protons is initialized with emittance 0.5 π·mm·mrad. Then, the beam is moving in the magnetic field of solenoid along z-axis. The length of solenoid is 15 cm and the magnetic field in the centre of solenoid is set to 17.2 T.

In the simulation, a focused beam of protons was observed at z-distance about 35 cm from the front edge of the solenoid. At later stage of the beam propagation in solenoid, the beam is again divergent. However, if the magnetic field and the length of the solenoid would be setup properly, divergence of initial proton beam could be decreased. This will be one of the tasks for our future research.

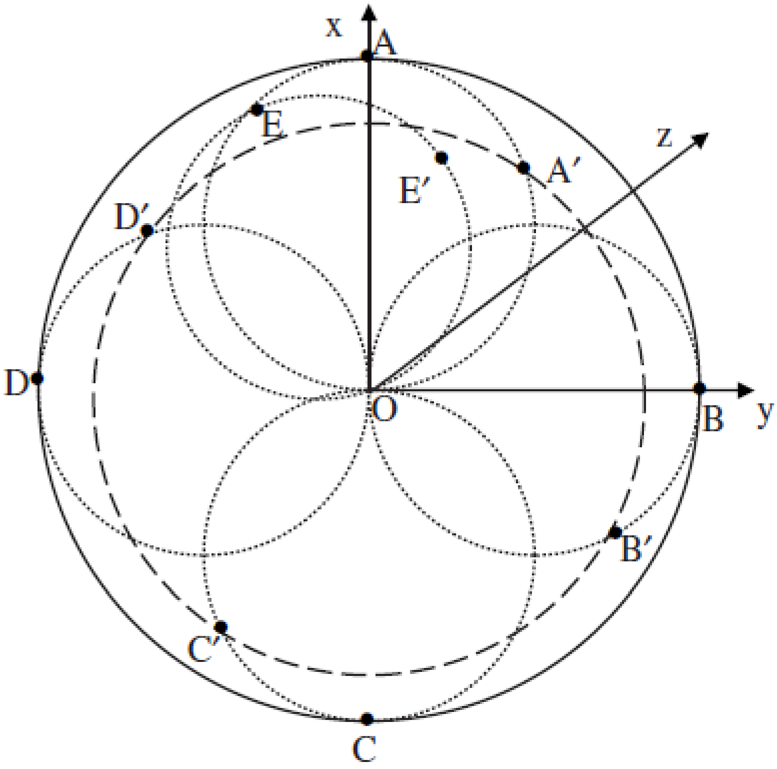


Figure 1.5: Focusing in the region with a uniform magnetic field; A, B, C, D, E – particles' positions at the beginning, A', B', C', D', E' – particles' positions after focusing, O – the centre of the beam; [22]

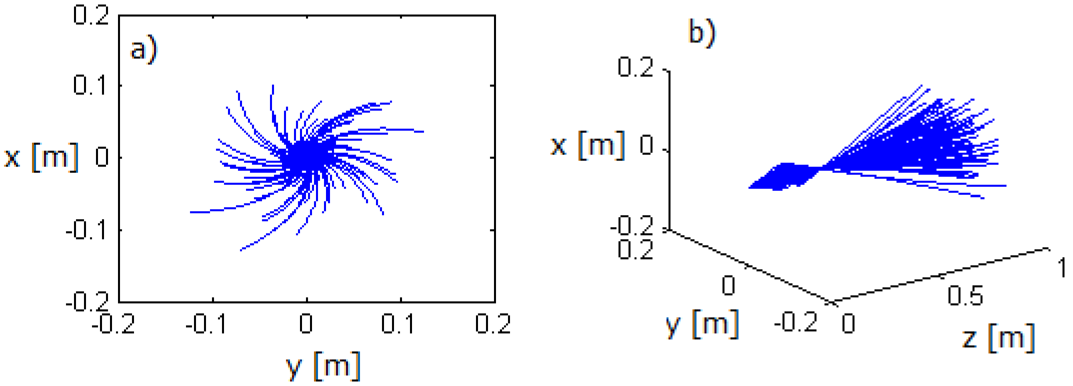


Figure 1.6: Examples of spatial views of charged particle beam trajectories in magnetic field of solenoid; a) y-x plane, b) 3D graph; Parameters of the beam: 100 protons, initial emittance 0.5 π·mm·mrad initial magnetic field in the solenoid centre 17.2 T, focus at 0.35 m

Quadrupole magnets

Quadrupole magnets (PQMs) are very strong focusing magnets arranged in quadrature, i.e. they are rotated 90° from each other and spaced-apart by electrical insulators, Fig. 1.7. In fact, PQMs are used for making particle beams more convergent when they move through.

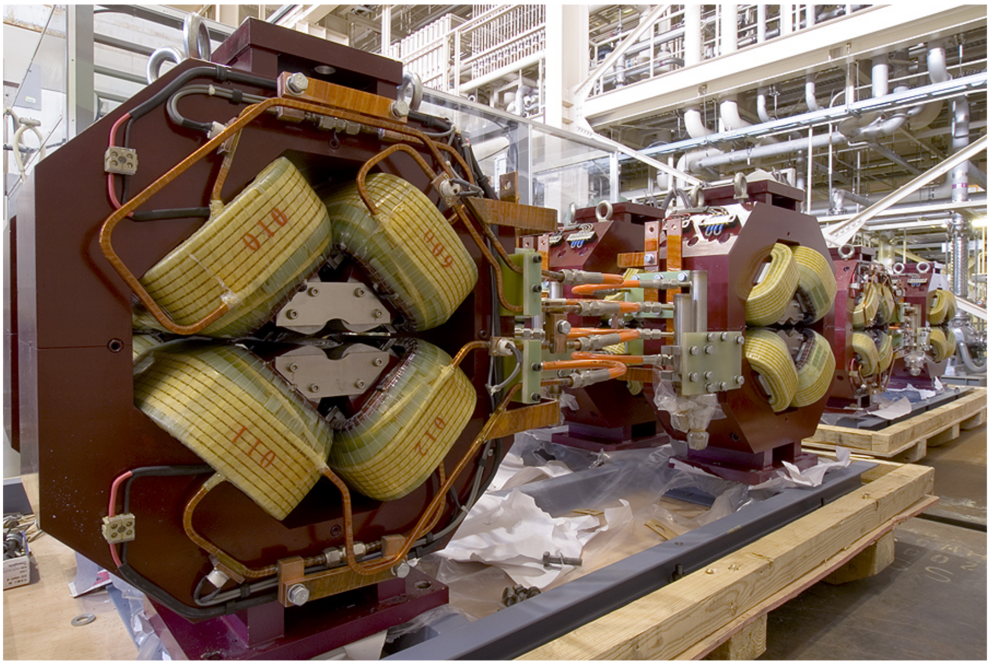


Figure 1.7: A batch of quadrupole magnets built for ATF2 by collaborators at IHEP, Beijing; [13]

Unlike curved targets or microspheres etc., the miniature permanent quadrupole magnets with small aceptance angles is divergence-decreasing components which could be used for high repetation rate operations. Focusing of a stable, at 1 Hz repetition-rated, laser-driven 2.4 MeV proton source using Ti:sapphire laser system and conventional PMQ pair is demonstrated in Ref. [28]. The proton beam is focused at the distance of 650 mm from source with focal spot of $\sim 3 \times 8$ mm^2. Moreover, the results are confirmed with Monte Carlo particle trajectory simulation. The exprimintal setup is demostrated in Fig. 1.8, further informations can be found in Ref. [28].

Conical cavity target holder

A divergence decreasing method using a target holder with an array of conical cavities is reported in Ref. [29]. The cavity is used for inducing an electric field. Thus, protons emitted from the target are collimated when they are traveling through the cavity. The holder used in experiment on J-KAREN Ti:sapphire laser system at Japan Atomic Energy Agency is made from alluminium and contains 16 conical cavities by 4 mm spacing arranged in a row. Moreover, the results obtained with conical target holder not significantly depend on the target material or thickness. In addition, its great advantage is that the conical cavities are not destroyed by the laser pulse, so only the target foil have to be replaced. This target holder and whole experimental set-up can be seen in Fig. 1.9.

Numerous of hot electrons are emitted from the surface of the target holder during the laser pulse-holder interaction and leave the holder charged. A strong electric potential grows and enlarges over the surface with relativistic velocity. Then, created electric field has a component normal to the surface which finally leads to collimated protons with certain energies. The

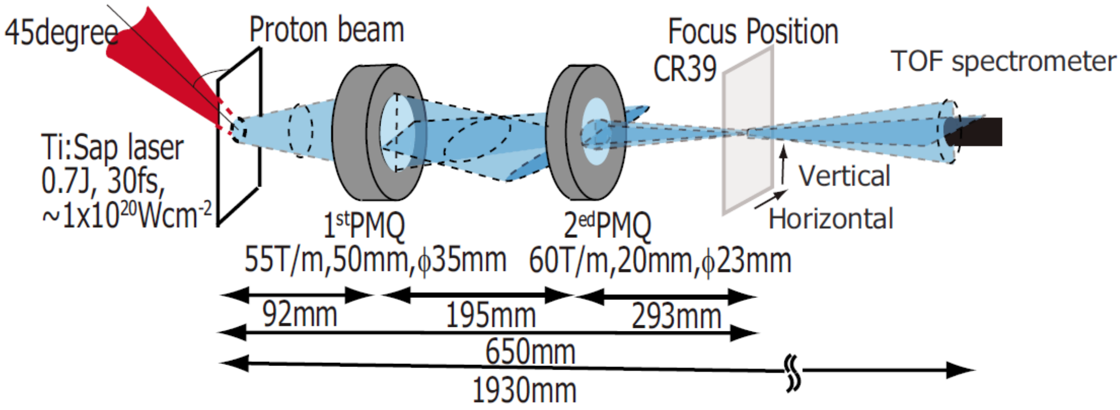


Figure 1.8: Schematic view of the experimental setup. The proton beam produced by the interaction between the short pulse Ti:sapphire laser and the 12.5μm polyimide tape target propagate into the PMQ devices. Those PMQs manipulate the orbit of the proton beam. The intensity profile of the focused proton beam is measured by the CR-39 nuclear track detector covered with the Al range filter. The proton spectrum is measured by the TOF spectrometer.; [28]

focusing strength of an electrostatic lens changes substantially with beam energy.
As a result, a 7 MeV proton beam of more than 10^{16} particles have been collimated to 16 mrad by focusing 2 J, 60 fs Ti:sapphire laser onto targets from different materials (Al, Si_3N_4), [29]. Unfortunately, the obtained values are too low for hadron therapy treatment which requires a well collimated $\sim$ 200 MeV proton beam. In theory, if ten times stronger collimating electric field will be applied within the volume of the conical cavity with the same size of the target but with a three times greater height, the well colimated 200 MeV proton beam will be obtain at 5 cm away from the target [29].

1.4.3 Decreasing beam divergence in the ultra-intense short pulse regime

It was shown experimentally that divergence of proton beams accelerated during ultra-intense laser irradiation of thin foils can be influenced by varying laser contrast. The laser contrast is the ratio of the main pulse intensity to that of the nanosecond amplified spontaneous emission (pre-pulse). Thus, the careful control of the laser contrast is crucial in laser-driven ion applications. A surprising decrease from 20° to 10° is observed when the increasing laser contrast using a double plasma mirror system is used (contrast was increased by a factor of $\sim 10^3$). Moreover, the improved collimation persisted over a various thickness of targets (from 50 nm to 6μm); the increased flux was measured with thinner targets [14].
Naturaly, the properties of laser-driven ion beams are driven by the parameters of the laser such as pulse duration, intensity or energy. With increasing laser intensity, the effect of pre-pulses and amplified spontaneous emission become more significant. The aim of experiments is to have the pre-pulse as low as it is possible in contrast to peak of laser intensity, because the pre-pulse may form preformed-plasma on the front surface of the target and, thus, ionize the target before the main pulse will come. This causes serious problems – for instance, the sheath acceleration mechanism is suppressed by reducing the peak accelerating electric field due to the pre-plasma existence.

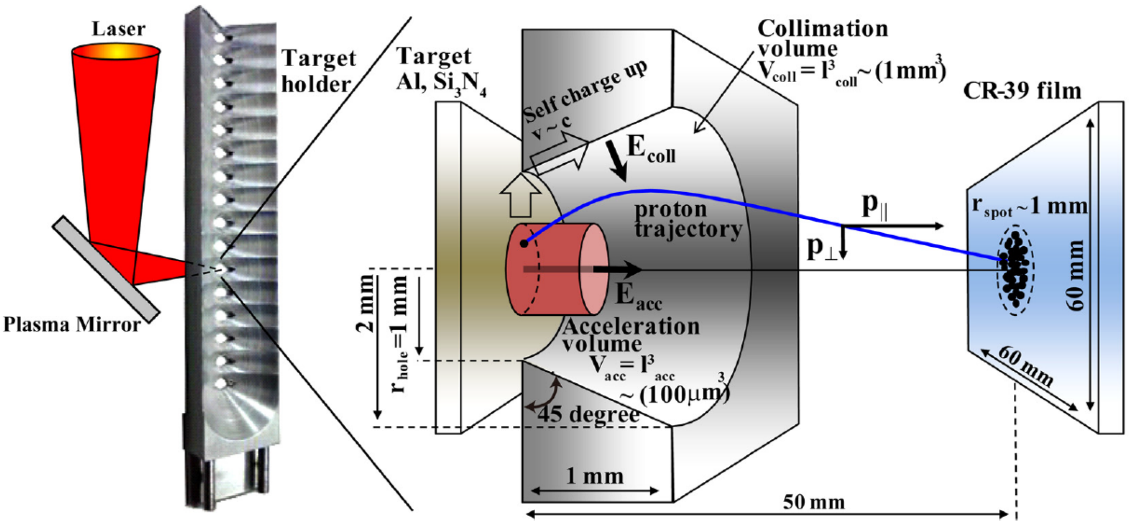


Figure 1.9: Experimental setup. Whole view of the target holder and cut away view of one of the conical cavities in the target holder arranged in a row are shown. The distribution of the static electric fields for acceleration (E_{acc}) and collimating (E_{coll}); [29]

Hydrodynamics and 2D PIC simulations demonstrate that the key factor in altering the proton beam emission profile is a change in the fast electron generation process due to the pre-plasma presence. In the case of low contrast (i.e. short scale length $\sim 1/4\lambda$), fluctuations in divergence is observed with varying target thickness ($6 - 20\mu$m), however, in high contrast (i.e. long scale length $\sim 5\lambda$) case the divergence stays constant over two orders of magnitude in target thiskness. In Fig. 1.10 the spatial profile of the accelerating longitudinal field along the rear surface after the peak of the acceleration is demonstrated. The profile is clearly different in the case of high and low laser contrast. For the low contrast case, the curvature is significantly lower and the protons from a significantly larger transverse region on the rear surface are accelerated by the sheath. In the case of the high laser contrast, at the beginning there is a high electron density which, as time goes on, results in a on axis peaked field where only protons within a few focal spots are accelerated to the highest energies [14].

1.5 Applications

Laser-driven ion beams have a great importance in a number of technological applications since the high internsity pulsed lasers were developed. Nowdays, the research is focused on decreasing divergence and energy spread of such beams. They can be a very promising source in medicine, for example for short-lived isotope production (positron emission tomography, PET), radiotherapy or hadrontherapy for treating malign tumors. Nevertheless, applications do not only cover biophysics, but also ion-driven fast ignition inertial confinement fusion, accelerators injectors or isochoric heating of solid-dense matter etc.

This work is taking into accound the ion cancer therapy (hadrontherapy), but there is a long way to go. Currently, hadrontherapy systems does not fit with the strict radiological requirements (ion energy, ion dose, etc.). Moreover, beside technical aspects, the biological

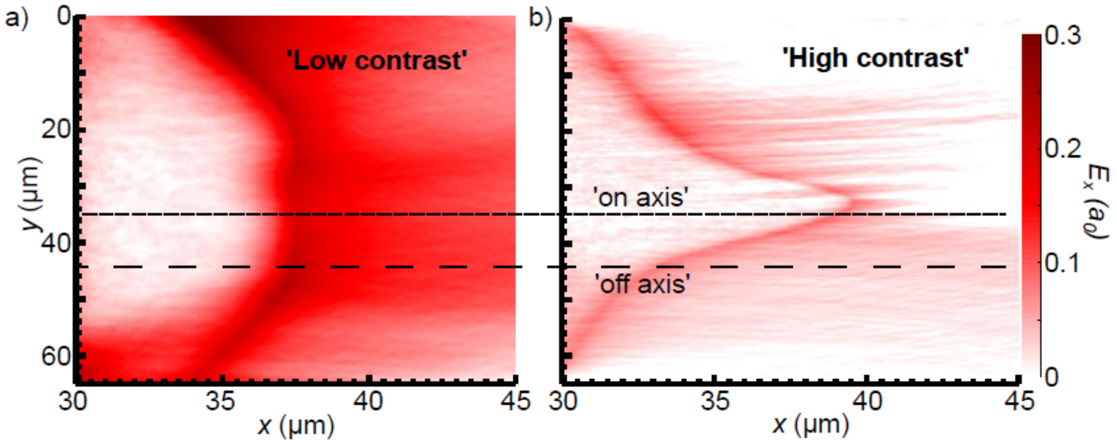


Figure 1.10: a) and b) E_x field spatial profiles along the rear surface showing the difference in sheath evolution between high and low contrast cases at ($t_{sim} = 320$ fs); [14]

consequences of short ion bunches associated with ultrashort laser pulses have to be better understood [42]. One of the current limitations of laser driven ion sources is the characteristic large divergence of such beams. Thus, this is the reason why the various method for avoiding divergence are discussed in this paper.

Chapter 2

Particle in cell Simulations (PIC)

2.1 Plasma modelling

There are three main possible descriptions in plasma modelling:

- **Kinetic description** – Kinetic model deals with general distribution function $f_\alpha(t, x_\alpha, v_\alpha)$, where index α means particle species. Boltzmann transport equation (BTR) is the equation describing a time evolution of the general distribution function $f_\alpha(t, \vec{x}, \vec{v_\alpha})$. It can be derived with assumption of α-particles colissions with target from β-particles [21]:

  $$\frac{\mathrm{d}f_\alpha(t, \vec{x}, \vec{v_\alpha})}{\mathrm{d}t} = \sum_\beta S_{\alpha\beta}, \tag{2.1}$$

  where right side is called Boltzmann collision integral. The commonly used BTR equation has the form:

  $$\frac{\partial f_\alpha}{\partial t} + \left(\vec{v_\alpha} \cdot \vec{\nabla}_x\right) f_\alpha + \frac{1}{m_\alpha}\left(\vec{F_\alpha} \cdot \vec{\nabla}_v\right) f_\alpha = \sum_\beta S_{\alpha\beta}. \tag{2.2}$$

  The right side express the collisional term which vary for different approaches to collisions. The resulting equations are called Fokker-Planck, Landau, Boltzman, BGK (Bhatnagar-Gross-Krook) or Vlasov equation [21]. The last case is suitable for collisionless plasma, i.e. the low temperature plasma where collisions are negliable. Thus, the collision term in BGK is zero:

  $$\frac{\partial f_\alpha}{\partial t} + \left(\vec{v_\alpha} \cdot \vec{\nabla}_x\right) f_\alpha + \frac{1}{m_\alpha}\left(\vec{F_\alpha} \cdot \vec{\nabla}_v\right) f_\alpha = 0. \tag{2.3}$$

- **Particle description** – Plasma is a system of single charged and neutral particles in vacuum. The plasma is described by electrons and ions moving under the influence of electric and magnetic field (due to their own charge) and of laser field [32]. Electromagnetic field is described via Maxwell equations for electric field $\vec{E}$ and magnetic field $\vec{B}$:

  $$\nabla \times \vec{E} + \frac{\partial \vec{B}}{\partial t} = 0, \qquad \nabla \cdot \vec{B} = 0, \tag{2.4}$$

  $$\nabla \times \vec{B} - \varepsilon_0\mu_0\frac{\partial \vec{E}}{\partial t} = \mu_0\vec{j}, \qquad \nabla \cdot \vec{E} = \frac{\rho}{\varepsilon_0}, \tag{2.5}$$

where $\vec{j}$ is current density, ε_0 is permitivity, μ_0 is permeability and ρ is charge density. For practical reasons, computer simulations of plasma using particle codes are limited to $N \approx 10^8$ particles, but the typical value for laboratory laser-plasma is $N \approx 10^{15}$ [32]. Thus, something as "simulation particle" called *macroparticle* containing a large number of real particles is presented. However, the decreasing number of particles means increasing noise.
Particle-in-cell is an example of such particle plasma modelling.

- **Hydrodynamic description** – For hydrodynamic description not only Maxwell equations, conservation laws of mass, momentum and energy, but also equations of state are required. Moreover, a state of local thermodynamic equilibrium is assumed. For system with relatively low intensities ($\approx 10^{15}$ W/cm^2) and relatively long laser pulses (ns) is taking plasma as a fluid a good approximation. Nevertheless, the model is not always valid because of an assumption of local thermodynamic equilibrium LTE (for instance, temperature and pressure can be defined only in LTE) [32].

2.2 Computational Approach difference between Vlasov equation and PIC code

Assume the Vlasov equation (2.3) with single-particle density $f(\vec{r}, \vec{p})$, where $\vec{r}$ and $\vec{p}$ are position vectors of the particle in phase space. Naturaly, each of them has three components, thus, we have to solve Vlasov equation in six-dimensional phase space, which is very computational-demanding even for 1D. Usually, finite differences on the Eulerian grid is used for solving the partial differential eqauation of Vlasov.

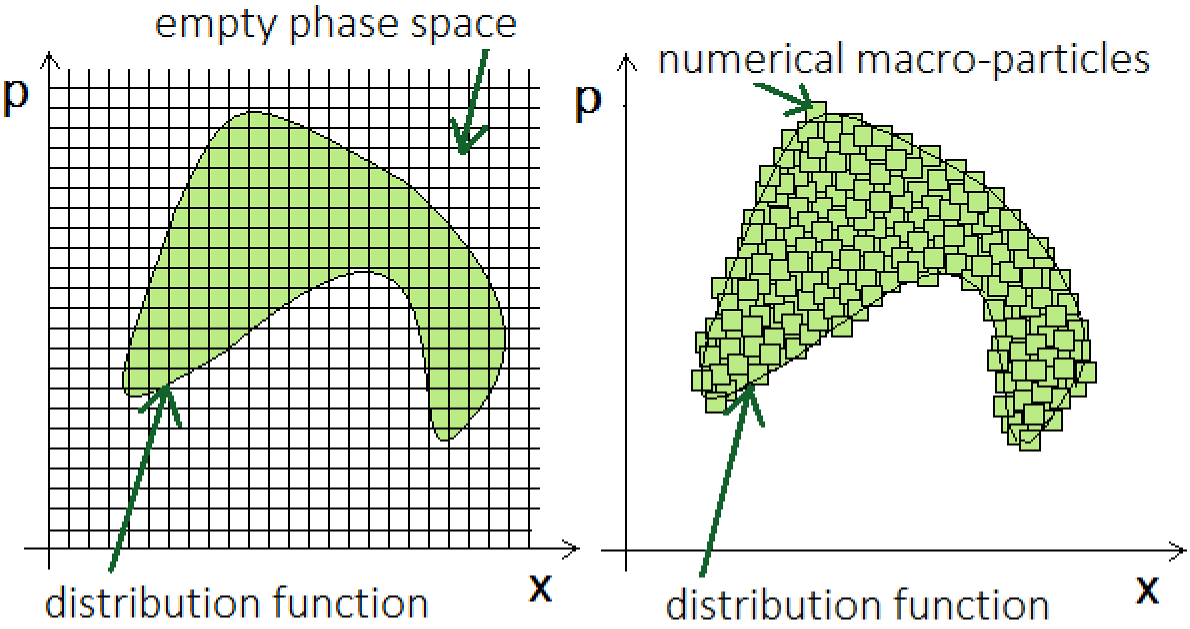


Figure 2.1: (left) Vlasov method: distribution function on Eulerian grid in 2D phase space; (right) PIC method: numerical macroparticles sample the distribution function.

The reason of wasting so much computational effort and cost is clear from the Fig. 2.1, where one plane of phase space with distribution function is shown. This distribution function is non-zero in dashed cloud, which represents the amount of phase space occupied by plasma particles and conversely zero in clear white surrounding space, where any particle is present. Regardless of whether the particles are in the region or not, all cells of the grid are included in numerical solving, which explains so high demands to computational power. Moreover, the impropriety of this computational approach rises with each dimension, because the empty space become bigger and bigger. Thus, solving kinetic collisionless Vlasov equation in six-dimensional phase space is almost impossible. On the other hand, there is one indisputable advantage od Vlasov codes and so the possibility of producing smooth results. This is mostly the reason why this codes are still studied.

Fortunately, there is Particle-in-cell (PIC) method which can solve Vlasov equation more elegant. To clarifying, when we use the Fig. 2.1 again, we can see that the greatest advantage is approximation of distrubution function by a set of finite phase-fluid elements (FPEE). Then the equation for distribution function is:

$$f_s(\vec{r},\vec{p}) = \sum_k W_k S(\vec{r}-\vec{r}_k,\vec{p}-\vec{p}_k), \tag{2.6}$$

where W_k is the weight of k-th particle of species s and $S(\vec{r}-\vec{r}_k,\vec{p}-\vec{p}_k)$ is the shape of particle in the phase space or the support function; $\vec{r}$ is coordinate of the particle and $\vec{r}_k$ is an observation point in phase space; for $\vec{p}$ the same notation is valid. In fact, the shape function is introduced, because real particles can occur anywhere in the space, but PIC simulation can compute the values of macro-quantities (i.e. particle density, current density) only in the node of the grid. Consequently, the real particles are assigned to simulation as macroparticles with specific *shape* (small squares in Fig. 2.1(b)). Moreover, the shape function has to satisfy the following conditions [37]:

- space isotropy,
- charge conservation,
- increasing accuracy (convergence) for higher-order terms.

Now, the evaluation of the shape function in 2D phase space is given by:

$$S(x,p_x) = 1 \qquad \text{for} \qquad |x-x_k| < \frac{\Delta x}{2}, \qquad |p-(p_x)_k| < \frac{\Delta p_x}{2}, \tag{2.7}$$

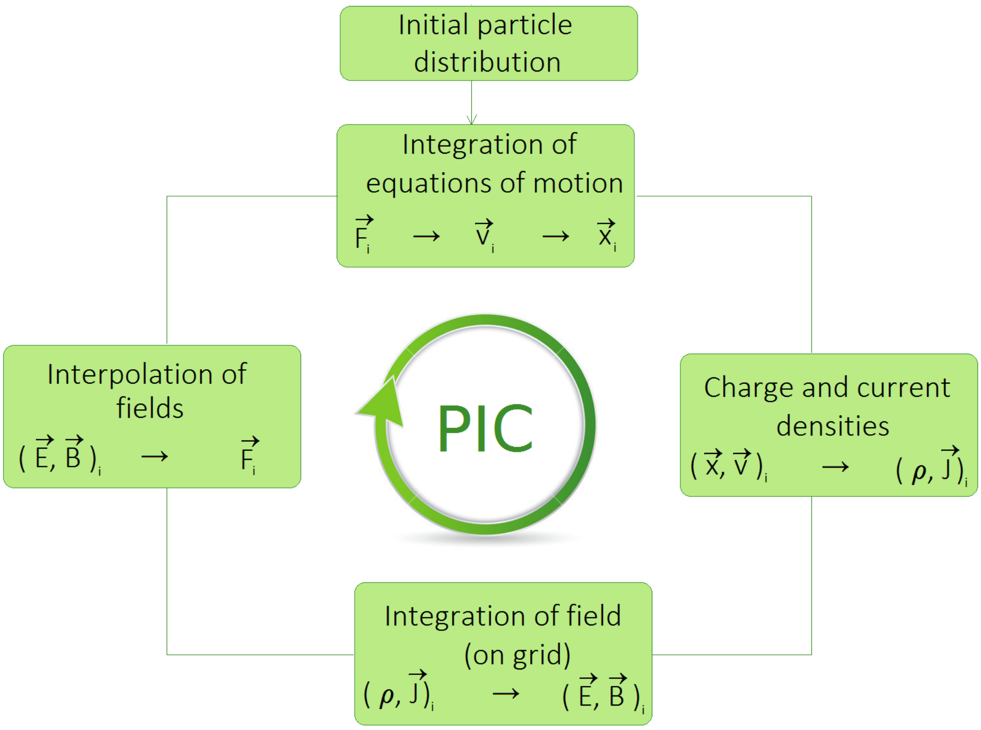
where Δx is *width* along x axis and Δp is *width* along p axis. In nutshell, when the real particle is inside examining cell with proportions $\Delta x \times \Delta t$, the shape function is equal to 1. Vlasov equation can be substituted by the following set of relativistic equations of motion for macroparticles [32]:

$$\frac{\partial \vec{r}_k}{\partial \vec{t}} = \frac{\vec{p}_k}{\gamma m_k}, \qquad \frac{\partial \vec{p}_k}{\partial \vec{t}} = q_k\left(\vec{E}_k - \frac{\vec{p}_k}{\gamma m_k} \times \vec{B}_k\right), \qquad \gamma = \sqrt{1+\left(\frac{\vec{p}_k}{m_k c}\right)^2}, \tag{2.8}$$

where $\vec{B}_k$ is magnetic field at the position of k-th particle, similarly $\vec{E}_k$ is electric field in the same place, γ is relativistic factor and m_k is rest mass of k-th particle. In general, the greatest advantage of PIC code is that is not computed in the regions where any particle is present.

2.3 Basic four-step Particle-in-cell scheme (PIC)

For solving a certain class of partial differential equations a four-step PIC scheme can be used for each dimension cases. The former, a list of individual steps is presented here and the latter, we will give a detailed explanation of each of them in 2D. A brief cycled plan of 4-step PIC is demostrated in Fig. 2.2.



Initial particle distribution
Integration of equations of motion
$\vec{F}_i \rightarrow \vec{v}_i \rightarrow \vec{x}_i$
Charge and current densities
$(\vec{x}, \vec{v})_i \rightarrow (\rho, \vec{J})_i$
Integration of field (on grid)
$(\rho, \vec{J})_i \rightarrow (\vec{E}, \vec{B})_i$
Interpolation of fields
$(\vec{E}, \vec{B})_i \rightarrow \vec{F}_i$
PIC

Figure 2.2: Basic cycled four-step scheme of Particle-in-cell method

1. Integration of equations of motion

An extraction of i-th particle velocity $\vec{v}_i$ and position $\vec{x}_i$ from the equation of motion is based on Lorentz force $\vec{F}_i$:

$$\vec{F}_i = \frac{\mathrm{d}(\gamma \vec{v}_i)}{\mathrm{d}t} = \frac{q}{m_i}\left(\vec{E}_i + \vec{v}_i \times \vec{B}_i\right). \tag{2.9}$$

Firstly, we will get velocity $\vec{v} = \frac{\mathrm{d}\vec{x}}{\mathrm{d}t}$ and then by integration with respect to time position $\vec{x}$. For better accurancy the *leap-frog scheme* [5] instead of Forward Difference method [5] is usually used to perform the integration. Therefore, velocities are evaluated at time $t = (n + 1/2)\Delta t$ and positions at time $t = n\Delta t$.

Leap-frog scheme

The Leap-frog scheme is an improvement of numeric Euler method, which can be derived by replacing the forward difference in Euler method by central difference [5]. Its advantage is mainly better accurancy, however it is unstable.

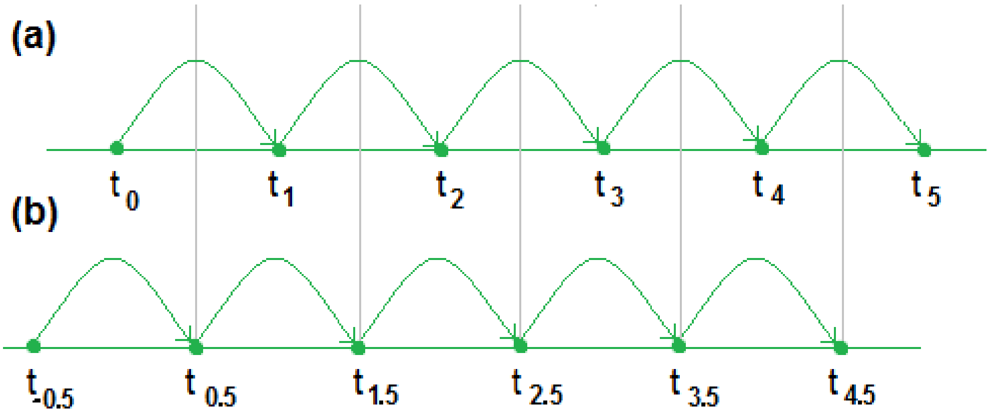


Figure 2.3: Leap-frog scheme; (a) positions are calculated at time $t = n\Delta t$; (b) velocities are calculated at time $t = (n + 1/2)\Delta t$

2. Charge and current densities

Secondly, charge and current densities ρ, $\vec{J}$ are computed in the node of the grid, where continuity equation must be satisfied:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \vec{J} = 0. \tag{2.10}$$

3. Integration of field (on grid)

From the previous step we obtain charge and current density. Subsequently, it is possible to solve the Maxwell equations (2.11) - (2.14) and get electric and magnetic field $\vec{E}$, $\vec{B}$ on the grid.

$$\nabla \times \vec{E} = -\frac{\partial \vec{B}}{t}, \tag{2.11}$$

$$\nabla \times \vec{B} = \mu_0 \vec{J} + \frac{1}{c^2}\frac{\partial \vec{E}}{\partial t}, \tag{2.12}$$

$$\nabla \cdot \vec{E} = \frac{\rho}{\epsilon_0}, \tag{2.13}$$

$$\nabla \cdot \vec{B} = 0. \tag{2.14}$$

If the fields do not change much during one time step, they can be computed on the grid only from Poisson equation for potentials [21].

4. Interpolation of fields

In last step electric and magnetic fields are interpolated from the mesh to the particle positions.

2.4 Computational steps of 2D PIC code

2.4.1 STEP ONE: Integration of equations of motion

In PIC particle solver the leap-frog scheme is used, thus similarly as velocities are evaluated at time $t = (n + 1/2)\Delta t$ and positions at time $t = n\Delta t$, magnetic filed is evaluated at $t = (n + 1/2)\Delta t$ and electric field at time $t = n\Delta t$. Firstly, the second Newton law is used:

$$\vec{F} = \frac{\mathrm{d}\vec{p}}{\mathrm{d}t} = m \cdot \frac{\mathrm{d}\vec{v}}{\mathrm{d}t} = q\left(\vec{E} + \vec{v} \times \vec{B}\right), \tag{2.15}$$

where Lorentz force $\vec{F} = \mathrm{d}\vec{p}/\mathrm{d}t$ is substituted by time central difference with dimensionless momentum of the particle $\vec{u} = \vec{p}/(mc)$:

$$\frac{\vec{u}^{n+1/2} - \vec{u}^{n-1/2}}{\Delta t} = \frac{q}{m}\left(\frac{\vec{E}^n}{c} + \frac{\vec{u}^n}{\gamma^n} \times \vec{B}^n\right), \tag{2.16}$$

$$\frac{\vec{x}^{n+1} - \vec{x}^n}{\Delta t} = \frac{\vec{u}^{n+1/2}}{\gamma^{n+1/2}}, \tag{2.17}$$

where $\vec{B}^n$ is approximated by arithmetic mean $\vec{B}^n = 1/2 \cdot \left(\vec{B}^{n-1/2} + \vec{B}^{n+1/2}\right)$ and γ^n is relastivistic factor satisfying an equation:

$$\gamma = \sqrt{(1 + (\vec{u})^2)}. \tag{2.18}$$

Equations (2.16), (2.17) and (2.18) are three main relations for moving particle in relativistic plasma [5], [32].

Now, approximations for dimensionless momentum $\vec{u}^n$ (2.25) and magnetic field $\vec{B}^n$ (2.24) will be prepared, because $\vec{u}^n, \vec{B}^n$ are not known yet. From Maxwell equation for $\nabla \times \vec{E}$ (2.11), central time difference and $\vec{B}^n$ we obtain:

$$\nabla \times \vec{E} = -\frac{\partial \vec{B}}{\partial t}, \tag{2.19}$$

$$\nabla \times \vec{E}^n = -\frac{l\vec{B}^{n+1/2} - \vec{B}^{n-1/2}}{\Delta t}, \tag{2.20}$$

$$\vec{B}^{n-1/2} - \Delta t \nabla \times \vec{E}^n = \vec{B}^{n+1/2}, \tag{2.21}$$

$$\vec{B}^n = \frac{\vec{B}^{n+1/2} - \vec{B}^{n-1/2}}{2} \rightarrow \vec{B}^{n+1/2} = 2\vec{B}^n - \vec{B}^{n-1/2} \tag{2.22}$$

$$\vec{B}^{n-1/2} - \Delta t \nabla \times \vec{E}^n = 2\vec{B}^n - \vec{B}^{n-1/2} \tag{2.23}$$

$$\vec{B}^n = \vec{B}^{n-1/2} - \frac{\Delta t}{2} \nabla \times \vec{E}^n. \tag{2.24}$$

Approximation for dimensionless momentum is a simple arithemetic mean:

$$\vec{u}^n = \frac{\vec{u}^{n+1/2} + \vec{u}^{n-1/2}}{2}, \tag{2.25}$$

where $\vec{u}^{n+1/2}$ and $\vec{u}^{n-1/2}$ are substituted in agreement with *Boris Method* [5]. In fact, it is the standard for particle pushing in plasma simulation codes:

$$\vec{u}^{n-1/2} = \vec{u^-} - \frac{q\vec{E}^n \Delta t}{2mc}, \qquad \vec{u}^{n+1/2} = \vec{u^+} + \frac{q\vec{E}^n \Delta t}{2mc}. \tag{2.26}$$

Now, by substituting (2.25), (2.24) and (2.26) into (2.16):

$$\frac{\vec{u}^{n+1/2}-\vec{u}^{n-1/2}}{\Delta t}=\frac{\vec{u}^{+}+\frac{q\vec{E}^{n}\Delta t}{2mc}-\vec{u}^{-}+\frac{q\vec{E}^{n}\Delta t}{2mc}}{\Delta t}=\frac{q\vec{E}^{n}}{mc}+q\frac{\vec{u}^{n+1/2}+\vec{u}^{n-1/2}}{2m\gamma^{n}}\times\vec{B}^{n}, \quad (2.27)$$

$$\frac{\vec{u}^{+}-\vec{u}^{-}}{\Delta t}+\frac{q\vec{E}^{n}}{mc}=\frac{q\vec{E}^{n}}{mc}+q\frac{\vec{u}^{+}+\vec{u}^{-}}{2m\gamma^{n}}\times\vec{B}^{n}, \quad (2.28)$$

the following relation is derived:

$$\frac{\vec{u}^{+}-\vec{u}^{-}}{\Delta t}=\frac{q}{2m\gamma^{n}}\left(\vec{u}^{+}+\vec{u}^{-}\right)\times\vec{B}^{n}. \quad (2.29)$$

The factor γ^n may be used to calculate a time-centered kinetic energy where the identity $(\gamma-1)mc^2=mu^2(\gamma+1)^{-1}$ is used.
Furthermore, the basic geometry in Fig. 2.4 is utilized to derive the expression for performing the rotation. Firstly, the vector $\vec{u}'$ is find by bisecting the angle θ between $\vec{u}^-$ and $\vec{u}^+$ (pre and post rotation relativistic momentum; where $\vec{u}=\vec{v}\gamma$). Then, from geometry:

$$\tan\frac{\theta}{2}=-\frac{qB\Delta t}{2\gamma^{n}mc}. \quad (2.30)$$

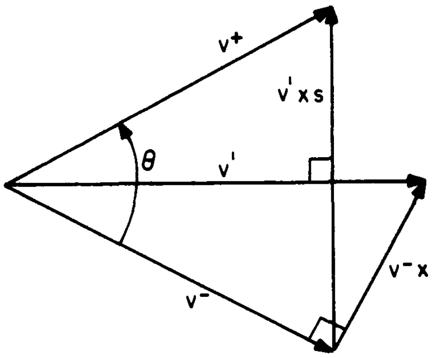


Figure 2.4: Boris method; velocity space showing the rotation from v^- to v^+. The velocities shown a projections of the total velocities onto the plane perpendicular to $\vec{B}$, dimensionless momentum $\vec{u}$ is used in text assuming the relation $\vec{u}=\gamma\vec{v}$; [5]

Because $\vec{B}$ is fixed and θ is not too large [5], approximationtan $\theta\simeq\theta$ can be used. Then, the following equation can be written:

$$\tan\frac{\theta}{2}=-\tan\left(\frac{qB\Delta t}{2\gamma^{n}mc}\right). \quad (2.31)$$

Moreover, assuming the substitution:

$$t=-\tan\frac{\theta}{2}, \quad (2.32)$$

the relations for s and r defined by $\sin\theta$ and $\cos\theta$ are:

$$s \equiv -\sin\theta = -2\sin\frac{\theta}{2}\cos\frac{\theta}{2} = \frac{-2\tan\frac{\theta}{2}}{1+\tan^2\frac{\theta}{2}} = \frac{-2t}{1+t^2}, \tag{2.33}$$

$$c \equiv \cos\theta = \cos^2\frac{\theta}{2} - \sin^2\frac{\theta}{2} = \frac{\frac{\cos^2\frac{\theta}{2}-\sin^2\frac{\theta}{2}}{\cos^2\frac{\theta}{2}}}{\frac{\cos^2\frac{\theta}{2}+\sin^2\frac{\theta}{2}}{\cos^2\frac{\theta}{2}}} = \frac{1-t^2}{1+t^2}. \tag{2.34}$$

Rotation in 2D is given by the relations:

$$x^{'} = x\cos\theta - y\sin\theta, \tag{2.35}$$

$$y^{'} = x\sin\theta + y\cos\theta, \tag{2.36}$$

then the rotation from (2.29) could be estimated as:

$$u_x^+ = cu_x^- + su_y^-, \tag{2.37}$$

$$u_y^+ = -su_x^- + cu_y^-. \tag{2.38}$$

In 1973 *Buneman* demonstrated that equations (2.33), (2.34), (2.37) and (2.38) can be reduced to [5]:

$$u_x^{'} = u_x^- + u_y^- t, \tag{2.39}$$

$$u_y^+ = u_y^- - u_x^{'} s, \tag{2.40}$$

$$u_x^+ = u_x^{'} + u_y^+ t. \tag{2.41}$$

From Fig. 2.4 (relations (2.39) and (2.41)):

$$\vec{u}' = \vec{u^-} + \vec{u^-} \times t, \tag{2.42}$$

where $\vec{u}' \perp \vec{B}$ and $\vec{u}' \perp (\vec{u^+} - \vec{u^-})$, as shown in Fig. 2.4. In addition, the directions of $\vec{B}$ and $\vec{v}$ are random and the angle between $\vec{u^-}$ and $\vec{u}'$ is $\theta/2$ (as well as between $\vec{u^+}$ and $\vec{u}'$ is $\theta/2$). By rewriting (2.30):

$$t \equiv -\hat{\vec{e}}\tan\frac{\theta}{2} = -\frac{q\vec{B}\Delta t}{2\gamma^n mc}, \tag{2.43}$$

where $\hat{\vec{e}}$ is unit vector in the direction of $\vec{B}$.
And again from the Fig. 2.4 and (2.40):

$$\vec{u^+} = \vec{u^-} + \vec{u}' \times s, \tag{2.44}$$

Thus, finally for dimensionless momentum:

$$\vec{u}^{'} = \vec{u^-} + \vec{u^-} \times t, \tag{2.45}$$

$$\vec{u^+} = \vec{u^-} + \vec{u}' \times s. \tag{2.46}$$

And new position of the particle is given from (2.17) by:

$$\vec{x}^{n+1} = \vec{x}^n + \vec{v}^{n+1}\Delta t = \vec{x}^n + \frac{\vec{u}^{n+1/2}\Delta t}{\gamma^{n+1/2}}, \tag{2.47}$$

with $\gamma^{n+1/2} = \sqrt{1+(u^{n+1/2}/c)^2}$.

2.4.2 STEP TWO: Interpolation of charge and current source terms to the field mesh

At the beginning, a proof of following statement will be shown:
Firstly, equations (2.13) *and* (2.14) *have to be fullfiled at the start of simulation and then if a conservation algorithm is used, the divergence is still valid. The current densities $\vec{J}$ have to satisfy the charge continuity equation.*
A local Maxwell solver equivalent to Finite Difference Time Domain Method (FDTDM) [40] is considered:

$$\frac{1}{c^2}\frac{\vec{E}^{n+1}-\vec{E}^{n}}{\Delta t} = \nabla^{+}\times\vec{B}^{n+\frac{1}{2}} - \mu_0\vec{J}^{n+\frac{1}{2}}, \tag{2.48}$$

$$\frac{\vec{B}^{n+\vec{1}2}-\vec{B}^{n-\frac{1}{2}}}{\Delta t} = -\nabla^{-}\times\vec{E}^{n}, \tag{2.49}$$

$$\nabla^{+}\cdot\vec{E}^{n} = \frac{\rho^n}{\epsilon_0}, \tag{2.50}$$

$$\nabla^{-}\cdot\vec{B}^{n+\frac{1}{2}} = 0, \tag{2.51}$$

where index n means value of a quantity in time $n\Delta t$. Symbols ∇^{+}, ∇^{-} are shortened notations for $\nabla^{+}f_{i,j,k}, \nabla^{-}f_{i,j,k}$ (2.52), (2.53) and we will call them *forward* and *backward gradient*, respectively (similar to forward and backward difference in numerical schemes). Its definitions are:

$$\nabla^{+}f_{i,j,k} = \left(\frac{f_{i+1,j,k}-f_{i,j,k}}{\Delta x}, \frac{f_{i,j+1,k}-f_{i,j,k}}{\Delta y}, \frac{f_{i,j,k+1}-f_{i,j,k}}{\Delta z}\right), \tag{2.52}$$

$$\nabla^{-}f_{i,j,k} = \left(\frac{f_{i,j,k}-f_{i-1,j,k}}{\Delta x}, \frac{f_{i,j,k}-f_{i,j-1,k}}{\Delta y}, \frac{f_{i,j,k}-f_{i,j,k-1}}{\Delta z}\right), \tag{2.53}$$

where indexes i, j, k mean grid points $x = i\Delta x, y = j\Delta y, z = k\Delta z$.

Now, the operator ∇^{+} acts on (2.48) and the operator ∇^{-} acts on (2.49); $\mu_0 = (c^2\epsilon_0)^{-1}$; moreover the identities $\nabla^{-}\cdot\nabla^{-}\times\vec{f} = \nabla^{+}\cdot\nabla^{+}\times\vec{f} = \vec{0}$ and $\nabla^{+}\cdot(\nabla^{+}\times\vec{B}) = 0$ are used:

$$\frac{1}{c^2}\frac{\nabla^{+}\vec{E}^{n+1}-\nabla^{+}\vec{E}^{n}}{\Delta t} = \nabla^{+}\left(\nabla^{+}\times\vec{B}^{n+\frac{1}{2}}\right) - \mu_0\nabla^{+}\cdot\vec{J}^{n+\frac{1}{2}}, \tag{2.54}$$

$$\frac{\rho^{n+1}-\rho^{n}}{\Delta t} + \nabla^{+}\cdot\vec{J}^{n+\frac{1}{2}} = 0, \tag{2.55}$$

$$\frac{\nabla^{-}\cdot\vec{B}^{n+\frac{1}{2}}-\nabla^{-}\cdot\vec{B}^{n-\frac{1}{2}}}{\Delta t} = 0, \tag{2.56}$$

where the equations (2.55) and (2.56) are the proof: when (2.55) is fullfiled, then the $\nabla\cdot\vec{E}$ is always equal to charge density and in addition if initial $\nabla\cdot\vec{B}$ is zero, then, according to (2.56), remains zero.

Zigzag scheme

There are a several methods which assume the trajectory of a particle as a straight line (Villasenor-Buneman method [38] and Esirkepov method [10]), opossite of these, the Zigzag scheme assume this trajectory as a zigzag line. In fact, this approach has many advantages,

such as a computational speed and no substantial distortion of physics.
In this paper, Zigzag scheme for the first and the second order spline interpolation is derived and results for the third order spline interpolation is presented [41].

Zigzag scheme for the first-order spline interpolation
The particle starts movement in point (x_1, y_1) and travels to (x_2, y_2) during one time step Δt. These initial and final coordinates can be expressed as follows:

$$x_1 \equiv x^n, \qquad y_1 \equiv y^n, \tag{2.57}$$

$$x_2 = x_1 + \Delta t = x^{n+1} = x^n + v_x^{n+\frac{1}{2}} \Delta t, \qquad y_2 = y_1 + \Delta t = y^{n+1} = y^n + v_y^{n+\frac{1}{2}} \Delta t. \tag{2.58}$$

Grid point indexes (that we will use in description of movement) are given by *floor function* with following relations:

$$i_1 = \left\lfloor \frac{x_1}{\Delta x} \right\rfloor, \qquad i_2 = \left\lfloor \frac{x_2}{\Delta x} \right\rfloor, \qquad j_1 = \left\lfloor \frac{y_1}{\Delta y} \right\rfloor, \qquad j_2 = \left\lfloor \frac{y_2}{\Delta y} \right\rfloor. \tag{2.59}$$

Moreover, very important *Courant condition* [5] have to be satisfied:

$$1 > (c\Delta t)^2 \left(\frac{1}{(\Delta x)^2} + \frac{1}{(\Delta y)^2} \right). \tag{2.60}$$

In short, this condition ensures that no particle moves more than a cell proportion (Δx and Δy) during one time step. In other words $v_x \Delta t < \Delta x, v_y \Delta t < \Delta y$. Courant condition is a condition on numerical calculations, having its importance in hydrodynamics, and saying that if the wave or hydrograph can travel throught the subreach (of known length) in a time less that the computational interval Δt, then computational instabilities may envolve.

The basic cases of particle motion during one time step are demonstrated in Fig. 2.5 and described in Tab. 2.1.

relation between init. *and final coordinate*	$i_1 = i_2$ $j_1 = j_2$	$i_1 \neq i_2$ $j_1 \neq j_2$	$i_1 \neq i_2$ or $i_1 = i_2$ $j_1 = j_2$ $\quad$ $j_1 \neq j_2$
occurence of motion	the same cell	across 2 cell boundaries	across 1 cell boundary
shape of trajectory	straight line	zigzag line	zigzag line
# steps of movement *+ description* *of the movement*	2 steps $(x_1, y_1) \to (\frac{x_1+x_2}{2})$ $(\frac{x_1+x_2}{2}) \to (x_2, y_2)$	2 steps $(x_1, y_1) \to (i_2\Delta x, j_2\Delta y)$ $(i_2\Delta x, j_2\Delta y) \to (x_2, y_2)$	2 steps $(x_1, y_1) \to (i_2\Delta x, \frac{y_1+y_2}{2})$ $(i_2\Delta x, \frac{y_1+y_2}{2}) \to (x_2, y_2)$ or $(x_1, y_1) \to (\frac{x_1+x_2}{2}, j_2\Delta y)$ $(\frac{x_1+x_2}{2}, j_2\Delta y) \to (x_2, y_2)$

Table 2.1: Zigzag scheme; Four basic cases of particle motion during one time step.

In the simplest case of movement existing in the same cell (see Tab. 2.1), a movement decomposition is only formal. Firstly, charge fluxes F_x, F_y and the first-order shape factor W are

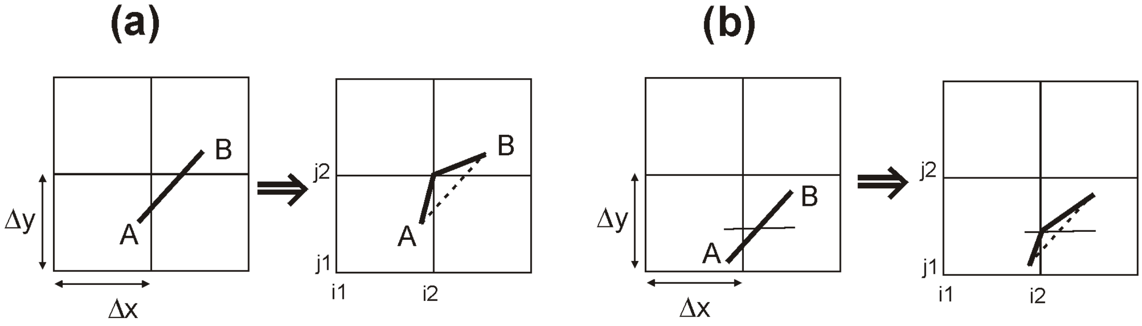


Figure 2.5: (a) When $i_1 \neq i_2$ and simultaneously $j_1 \neq j_2$, the movement of particle from A to B is decomposed in two movements. (b) The decomposition of particle trajectory from A to B in the case when $i_1 \neq i_2$ and $j_1 = j_2$; [32]

defined. The shape factor corresponds to the linear weighting function, which defined in the midpoint between (x_1, y_1) and (x_2, y_2):

$$F_x = qv_x = q\frac{x_2 - x_1}{\Delta t}, \qquad F_y = qv_y = q\frac{y_2 - y_1}{\Delta t}, \tag{2.61}$$

$$W_x = \frac{x_1 + x_2}{2\Delta x} - i_1, \qquad W_y = \frac{y_1 + y_2}{2\Delta y} - j_1. \tag{2.62}$$

For dimensionless current densities J_x, J_y (normalized by $en_c c$, where n_c is critical density) [32]:

$$J_x(i_1 + \frac{1}{2}, j_1) = \frac{1}{ppc}F_x(1 - W_y) \qquad J_x(i_1 + \frac{1}{2}, j_1 + 1) = \frac{1}{ppc}F_x W_y, \tag{2.63}$$

$$J_y(i_1, j_1 - \frac{1}{2}) = \frac{1}{ppc}F_y(1 - W_x) \qquad J_y(i_1 + 1, j_1 - \frac{1}{2}) = \frac{1}{ppc}F_y W_x, \tag{2.64}$$

where *ppc* means the number of electrons per cell at critical density.
Now the charge fluxes F_x, F_y given by (2.61) and weighting factors W_x, W_y given by (2.62) are decomposed into $F_{x1}, F_{x2}, F_{y1}, F_{y2}$ and $W_{x1}, W_{x2}, W_{y1}, W_{y2}$ by using new point (x_r, y_r) defined as:

$$x_r = \min(\min(i_1\Delta x, i_2\Delta x) + \Delta x), \max(\max(i_1\Delta x, i_2\Delta x), \frac{x_1 + x_2}{2}) \tag{2.65}$$

$$y_r = \min(\min(j_1\Delta y, j_2\Delta y) + \Delta y), \max(\max(j_1\Delta y, j_2\Delta y), \frac{y_1 + y_2}{2}) \tag{2.66}$$

Clearly, new point (x_r, y_r) is situated between points (x_1, y_1), i.e. initial position, and (x_2, y_2), i.e. final position, during one time step Δt. The definition of (x_r, y_r) follows the demand on the shortest possible distance from (x_1, y_1) or (x_2, y_2) to the border of considered cell [41]. Thus, charge fluxes and the first order-shape factors:

$$F_{x1} = q\frac{x_r - x_1}{\Delta t}, \qquad F_{y1} = q\frac{y_r - y_1}{\Delta t}, \tag{2.67}$$

$$F_{x2} = q\frac{x_2 - x_r}{\Delta t}, \qquad F_{y2} = q\frac{y_2 - y_r}{\Delta t}, \tag{2.68}$$

$$W_{x1} = \frac{x_1 + x_r}{2\Delta x} - i_1, \qquad W_{y1} = \frac{y_1 + y_r}{2\Delta y} - j_1, \tag{2.69}$$

$$W_{x2} = \frac{x_r + x_2}{2\Delta x} - i_2, \qquad W_{y2} = \frac{y_r + y_2}{2\Delta y} - j_2. \tag{2.70}$$

And finally, charge fluxes assigned to 8 grid points are defined as follows:

$$J_x(i_1 + \frac{1}{2}, j_1) = \frac{1}{ppc} F_{x1}(1 - W_{y1}), \qquad J_x(i_1 + \frac{1}{2}, j_1 + 1) = \frac{1}{ppc} F_{x1} W_{y1}, \tag{2.71}$$

$$J_y(i_1, j_1 + \frac{1}{2}) = \frac{1}{ppc} F_{y1}(1 - W_{x1}), \qquad J_y(i_1 + 1, j_1 + \frac{1}{2}) = \frac{1}{ppc} F_{y1} W_{x1}, \tag{2.72}$$

$$J_x(i_2 + \frac{1}{2}, j_2) = \frac{1}{ppc} F_{x2}(1 - W_{y2}), \qquad J_x(i_2 + \frac{1}{2}, j_2 + 1) = \frac{1}{ppc} F_{x2} W_{y2}, \tag{2.73}$$

$$J_y(i_2, j_2 + \frac{1}{2}) = \frac{1}{ppc} F_{y2}(1 - W_{x2}), \qquad J_y(i_2 + 1, j_2 + \frac{1}{2}) = \frac{1}{ppc} F_{y2} W_{x2}. \tag{2.74}$$

Zigzag scheme for the second-order spline interpolation
Here a the second-order spline interpolation of Zigzag scheme will be derived; article [41] is mainly followed.
Firstly, the continuity equation is itemized:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \vec{J} = 0, \tag{2.75}$$

$$\begin{aligned} &\frac{\rho^{t+\Delta t}(i,j) + \rho^{t}(i,j)}{\Delta t} + \frac{J_x^{t+\frac{\Delta t}{2}}(i + \frac{1}{2}, j) + J_x^{t+\frac{\Delta t}{2}(i-\frac{1}{2},j)}}{\Delta x} + \\ &+ \frac{J_y^{t+\frac{\Delta t}{2}}(i, j + \frac{1}{2}) + J_y^{t+\frac{\Delta t}{2}(i,j-\frac{1}{2})}}{\Delta y} = 0, \end{aligned} \tag{2.76}$$

where dx and dy is grid proportions (divergence in 2D case) and dt means one time step; charge density ρ can be expressed by charge q_m and two-order shape-factor of m-th particle $S_{i,j}(x_m, y_m)$ as following:

$$\rho(i,j) = \sum_m q_m S_{i,j}(x_m, y_m). \tag{2.77}$$

The initial and final particle coordinates of movement are expressed by (2.58), same as in the first-order spline interpolation case. Then, x_1, x_2, y_1, y_2 are normalized by Δx and Δy and denoted as $x_{old}, x_{new}, y_{old}$ and y_{new}, respectively. New grid points i_1, i_2, j_1, j_2, similarly to (2.59), are defined as follows:

$$i_1 = \left\lfloor \frac{x_1}{\Delta x} + 1{,}5 \right\rfloor = \left\lfloor x_{old} + 1{,}5 \right\rfloor, \qquad j_1 = \left\lfloor \frac{y_1}{\Delta y} + 1{,}5 \right\rfloor = \left\lfloor y_{old} + 1{,}5 \right\rfloor, \tag{2.78}$$

$$i_2 = \left\lfloor \frac{x_2}{\Delta x} + 1{,}5 \right\rfloor = \left\lfloor x_{new} + 1{,}5 \right\rfloor, \qquad j_2 = \left\lfloor \frac{y_2}{\Delta y} + 1{,}5 \right\rfloor = \left\lfloor y_{new} + 1{,}5 \right\rfloor. \tag{2.79}$$

In addition, the Courant condition (2.60) have to be satisfied. The motion progress can be divided into the moves in x-direction and y-direction. Here, we will discuss only movement in x-direction, because for y-direction is derivation analogous.

Likewise in the first-order case, the point (x_r, y_r) is defined as a point locating between (x_{old}, y_{old}) and (x_{new}, y_{new}):

$$x_r = \begin{cases} \frac{x_1+x_2}{2}, & (i_1 = i_2) \\ \\ \frac{(i_1+i_2)\Delta x}{2}, & (i_1 \neq i_2) \end{cases} \tag{2.80}$$

$$y_r = \begin{cases} \frac{y_1+y_2}{2}, & (j_1 = j_2) \\ \\ \frac{(j_1+j_2)\Delta y}{2}, & (j_1 \neq j_2) \end{cases} \tag{2.81}$$

For computing (x_r, y_r) following equations are used:

$$x_r = \min[\min(i_1 - 0,5, i_2 - 0,5), \max(\max(i_1 - 1,5, i_2 - 1,5), \frac{x_{old} + x_{new}}{2})], \tag{2.82}$$

$$y_r = \min[\min(j_1 - 0,5, j_2 - 0,5), \max(\max(j_1 - 1,5, j_2 - 1,5), \frac{y_{old} + y_{new}}{2})]. \tag{2.83}$$

Definitions (2.78), (2.79), (2.80), (2.81) are demonstrated in Fig. 2.6:

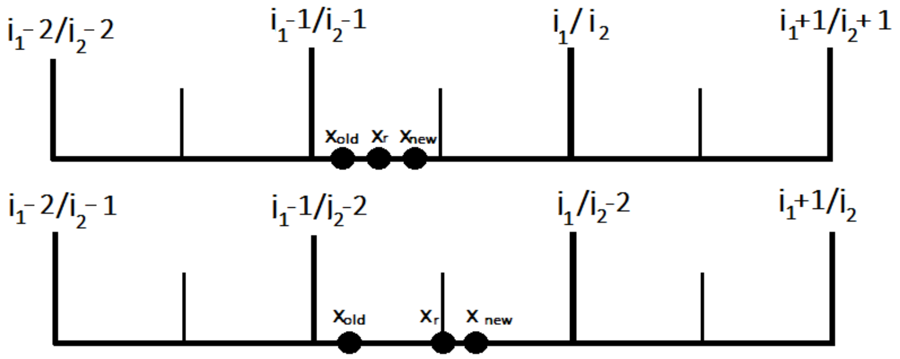


Figure 2.6: Location of particle at times t and $t+\Delta t$; ilustration of x_{new}, x_{old}, x_r and definition of i_1, i_2.

The movement of the particle is decomposed again into two parts and follows the same rule as described in the first-order Zigzag scheme (see picture Fig. 2.5, but do not forget that notation of grid points is different).

After getting point (x_r, y_r), a charge flux can be separated into $F_1 = (F_{x1}, F_{y1}), F_2 = (F_{x2}, F_{y2})$ (analog to (2.67) and (2.68)) as follows:

$$F_{x1} = q_n(x_r - x_{old})\frac{\Delta x}{\Delta t}, \tag{2.84}$$

$$F_{y1} = q_n(y_r - y_{old})\frac{\Delta y}{\Delta t}, \tag{2.85}$$

$$F_{x2} = q_n(x_{new} - x_r)\frac{\Delta x}{\Delta t} = q_n(x_{new} - x_{old})\frac{\Delta x}{\Delta t} - q_n(x_r - x_{old})\frac{\Delta x}{\Delta t} = q_n v_x - F_{x1}, \tag{2.86}$$

$$F_{y2} = q_n(y_{new} - y_r)\frac{\Delta y}{\Delta t} = q_n(y_{new} - y_{old})\frac{\Delta y}{\Delta t} - q_n(y_r - y_{old})\frac{\Delta y}{\Delta t} = q_n v_y - F_{y1}. \tag{2.87}$$

For obtaining the final results weighting functions $w_{x1}, w_{x2}, w_{x3}, w_{y1}, w_{y2}, w_{y3}$ (for the midpoint of (x_{old}, y_{old})) are required. Similarly, (x_r, y_r) and $w_{x4}, w_{x5}, w_{x6}, w_{y4}, w_{y5}, w_{y6}$ (for the midpoint of (x_r, y_r) and (x_{new}, y_{new})) are required to getting the results. More detailed definition of these functions can be found in Ref. [41]. Further, we define:

$$i_{11} = [x_{old} + 1], \qquad i_{22} = [x_{new} + 1], \tag{2.88}$$

$$j_{11} = [y_{old} + 1], \qquad j_{22} = [y_{new} + 1], \tag{2.89}$$

and we are finally obtaining:

$$j_x(j_1 - 1, i_{11} + \frac{1}{2}) = \frac{1}{\Delta x \Delta y} F_{x1} W_{y1}, \qquad j_x(j_2 - 1, i_{22} + \frac{1}{2}) = \frac{1}{\Delta x \Delta y} F_{x2} W_{y4}, \tag{2.90}$$

$$j_x(j_1, i_{11} + \frac{1}{2}) = \frac{1}{\Delta x \Delta y} F_{x1} W_{y2}, \qquad j_x(j_2, i_{22} + \frac{1}{2}) = \frac{1}{\Delta x \Delta y} F_{x2} W_{y5}, \tag{2.91}$$

$$j_x(j_1 + 1, i_{11} + \frac{1}{2}) = \frac{1}{\Delta x \Delta y} F_{x1} W_{y3}, \qquad j_x(j_2 + 1, i_{22} + \frac{1}{2}) = \frac{1}{\Delta x \Delta y} F_{x2} W_{y6}, \tag{2.92}$$

$$j_y(j_{11} + \frac{1}{2}, i_1 - 1) = \frac{1}{\Delta x \Delta y} F_{y1} W_{x1}, \qquad j_y(j_{22} + \frac{1}{2}, i_2 - 1) = \frac{1}{\Delta x \Delta y} F_{y2} W_{x4}, \tag{2.93}$$

$$j_y(j_{11} + \frac{1}{2}, i_1) = \frac{1}{\Delta x \Delta y} F_{y1} W_{x2}, \qquad j_y(j_{22} + \frac{1}{2}, i_2) = \frac{1}{\Delta x \Delta y} F_{y2} W_{x5}, \tag{2.94}$$

$$j_y(j_{11} + \frac{1}{2}, i_1 + 1) = \frac{1}{\Delta x \Delta y} F_{y1} W_{x3}, \qquad j_y(j_{22} + \frac{1}{2}, i_2 + 1) = \frac{1}{\Delta x \Delta y} F_{y2} W_{x6}. \tag{2.95}$$

Zigzag scheme for the third-order spline interpolation
Only a result is presented here. For more details see [41].

$$j_x(j_1 - 1, i_1 + \frac{1}{2}) = \frac{1}{\Delta x \Delta y} F_{x1} W_{y1}, \qquad j_x(j_2 - 1, i_2 + \frac{1}{2}) = \frac{1}{\Delta x \Delta y} F_{x2} W_{y5}, \tag{2.96}$$

$$j_y(j_1, i_1 + \frac{1}{2}) = \frac{1}{\Delta x \Delta y} F_{x1} W_{x2}, \qquad j_x(j_2, i_2 + \frac{1}{2}) = \frac{1}{\Delta x \Delta y} F_{x2} W_{x6}, \tag{2.97}$$

$$j_x(j_1 + 1, i_1 + \frac{3}{2}) = \frac{1}{\Delta x \Delta y} F_{x1} W_{y3}, \qquad j_x(j_2 + 1, i_2 + \frac{3}{2}) = \frac{1}{\Delta x \Delta y} F_{x2} W_{y7}, \tag{2.98}$$

$$j_x(j_1 + 1, i_1 + \frac{3}{2}) = \frac{1}{\Delta x \Delta y} F_{x1} W_{y4}, \qquad j_x(j_2 + 2, i_2 + \frac{3}{2}) = \frac{1}{\Delta x \Delta y} F_{x2} W_{y8}, \tag{2.99}$$

$$j_y(j_1 + \frac{3}{2}, i_1 - 1) = \frac{1}{\Delta x \Delta y} F_{y1} W_{x1}, \qquad j_y(j_2 + \frac{3}{2}, i_2 - 1) = \frac{1}{\Delta x \Delta y} F_{y2} W_{x5}, \tag{2.100}$$

$$j_y(j_1 + \frac{3}{2}, i_1) = \frac{1}{\Delta x \Delta y} F_{y1} W_{x2}, \qquad j_y(j_2 + \frac{3}{2}, i_2) = \frac{1}{\Delta x \Delta y} F_{y2} W_{x6}, \tag{2.101}$$

$$j_y(j_1 + \frac{3}{2}, i_1 + 1) = \frac{1}{\Delta x \Delta y} F_{y1} W_{x3}, \qquad j_y(j_2 + \frac{3}{2}, i_2 + 1) = \frac{1}{\Delta x \Delta y} F_{y2} W_{x7}, \tag{2.102}$$

$$j_y(j_1 + \frac{3}{2}, i_1 + 2) = \frac{1}{\Delta x \Delta y} F_{y1} W_{x4}, \qquad j_y(j_2 + \frac{3}{2}, i_2 + 2) = \frac{1}{\Delta x \Delta y} F_{y2} W_{x8}. \tag{2.103}$$

2.4.3 STEP THREE: Integration of electric and magnetic field on mesh points

In short, the goal in this step is to compute the electric and magnetic field on mesh points. For better clarity picture of one cell ($\Delta x \times \Delta y$) is shown in Fig. 2.7, where arrangement of defined components of $\vec{J}, \vec{E}$ and $\vec{B}$ is clearly organized.

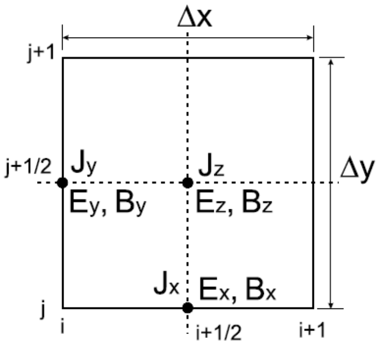


Figure 2.7: Positions of x, y and z component of $\vec{J}, \vec{E}, \vec{B}$ on the grid; [32]

Firstly, the assumption of *trans-magnetic* case is used, meaning that there is no magnetic field in the direction of propagation (which can be along x or y axis), thus the only non-zero component of magnetic field $\vec{B}$ is B_z. Secondly, the Maxwell equation for rotations (2.11) and (2.12) are taken and modified according to the components position oĂŻn Fig. 2.7 and relation for rotation:

$$\mathrm{rot}\vec{A} = \nabla \times \vec{A} = \left(\frac{\partial A_z}{\partial y} - \frac{\partial A_y}{\partial z}, \frac{\partial A_x}{\partial z} - \frac{\partial A_z}{\partial x}, \frac{\partial A_y}{\partial x} - \frac{\partial A_x}{\partial y} \right) \tag{2.104}$$

where $\vec{A} = (A_x, A_y, A_z)$ is a vector having three components.
Now, the equation (2.12) will be modified to get a basic for deriving equations (2.113) and (2.114):

$$\nabla \times \vec{B} = \mu_0 \vec{J} + \frac{1}{c^2}\frac{\partial \vec{E}}{\partial t} \tag{2.105}$$

$$\nabla \times \vec{B} = \left(\frac{\partial B_z}{\partial y} - \frac{\partial B_y}{\partial z}, \frac{\partial B_x}{\partial z} - \frac{\partial B_z}{\partial x}, \frac{\partial B_x}{\partial y} - \frac{\partial B_y}{\partial x} \right) = \left(\frac{\partial B_z}{\partial y}, -\frac{\partial B_z}{\partial x}, 0 \right) \tag{2.106}$$

$$\frac{1}{c^2}\frac{\Delta E_x}{\Delta t} = \frac{\Delta B_z}{\Delta y} - \mu_0 J_x \tag{2.107}$$

$$\frac{1}{c^2}\frac{\Delta E_y}{\Delta t} = -\frac{\Delta B_z}{\Delta x} - \mu_0 J_y \tag{2.108}$$

Similarly, the equation (2.11) is used for deriving the relation needed for (2.112):

$$\nabla \times \vec{E} = -\frac{\partial \vec{B}}{\partial t}, \tag{2.109}$$

because only ΔB_z is non-zero, we will get only one equation with z-component of rotation $\vec{E}$ on the left side:

$$\frac{\partial E_y}{\partial x} - \frac{\partial E_y}{\partial x} = -\frac{\partial B_z}{\partial t} \tag{2.110}$$

$$\frac{\Delta E_y}{\Delta x} - \frac{\Delta E_y}{\Delta x} = -\frac{\Delta B_z}{\Delta t}. \tag{2.111}$$

Finally, from (2.111), (2.107) and (2.108) according to the components position in Fig. 2.7 these three equations are obtained:

$$\frac{(B_z)^{n+\frac{1}{2}}_{i+\frac{1}{2},j+\frac{1}{2}} - (B_z)^{n-\frac{1}{2}}_{i+\frac{1}{2},j+\frac{1}{2}}}{\Delta t} = -\frac{(E_y)^n_{i+1,j+\frac{1}{2}} - (E_y)^n_{i,j+\frac{1}{2}}}{\Delta x} + \frac{(E_x)^n_{i+\frac{1}{2},j+1} - (E_x)^n_{i+\frac{1}{2},j}}{\Delta y}, \tag{2.112}$$

$$\frac{(E_x)^{n+1}_{i+\frac{1}{2},j} - (E_x)^n_{i+\frac{1}{2},j}}{\Delta t} = \frac{(B_z)^{n+\frac{1}{2}}_{i+\frac{1}{2},j+\frac{1}{2}} - (B_z)^{n+\frac{1}{2}}_{i+\frac{1}{2},j-\frac{1}{2}}}{\Delta y} - 2\pi(J_x)^{n+\frac{1}{2}}_{i+\frac{1}{2},j}, \tag{2.113}$$

$$\frac{(E_y)^{n+1}_{i,j+\frac{1}{2}} - (E_y)^n_{i,j+\frac{1}{2}}}{\Delta t} = -\frac{(B_z)^{n+\frac{1}{2}}_{i+\frac{1}{2},j+\frac{1}{2}} - (B_z)^{n+\frac{1}{2}}_{i-\frac{1}{2},j+\frac{1}{2}}}{\Delta x} - 2\pi(J_y)^{n+\frac{1}{2}}_{i,j+\frac{1}{2}}, \tag{2.114}$$

where 2π is dimensionless normalisation.

2.4.4 STEP FOUR: Interpolation of the fields from the mesh to the particle locations

Finally, the last step of PIC scheme is interpolation of fields. In nutshell, each field is weighted from the grid point to theparticles positions. For achieving it the first-order weighting represented by the first-order shape factor $S(\xi, \eta)$ is used. These shape factors can be evaluated according to if or if not ξ or η is located in the cell containing the particle (see also Fig. 2.8):

$$S_i(\xi) = \begin{cases} 1 - |\xi - i| & \text{for } |\xi - i| \le 1 \\ \\ 0 & \text{for } |\xi - i| > 1 \end{cases} \tag{2.115}$$

where $\xi = \frac{x_p}{\Delta x}$ and p means index of the particle;

$$S_j(\eta) = \begin{cases} 1 - |\eta - j| & \text{for } |\eta - j| \le 1 \\ \\ 0 & \text{for } |\eta - j| > 1 \end{cases} \tag{2.116}$$

where $\eta = \frac{y_p}{\Delta y}$ and p means index of particle.

Now, the four particle closest apperances of E_x, E_y and B_z in Fig. 2.8 is summed into $(E_x)_p, (E_y)_p, (B_z)_p$, respectively:

$$\begin{aligned}(E_x)_p = {} & S_{i+\frac{1}{2}}(\xi)S_j(\eta)(E_x)_{i+\frac{1}{2},j} + S_{i+\frac{1}{2}}(\xi)S_{j+1}(\eta)(E_x)_{i+\frac{1}{2},j+1} \\ & + S_{i+\frac{3}{2}}(\xi)S_j(\eta)(E_x)_{i+\frac{3}{2},j} + S_{i+\frac{3}{2}}(\xi)S_{j+1}(\eta)(E_x)_{i+\frac{3}{2},j+1}\end{aligned} \tag{2.117}$$

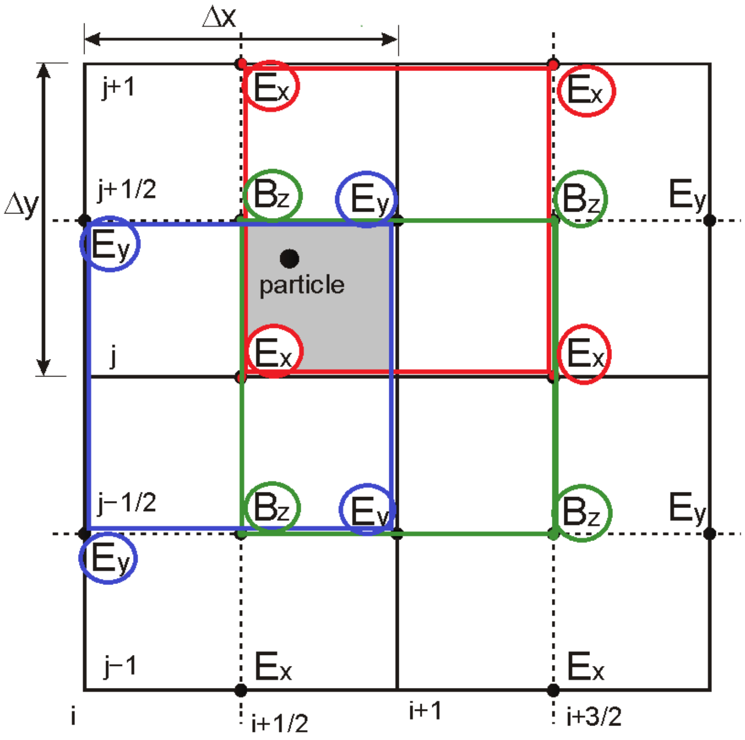


Figure 2.8: Positions of components of $\vec{E}, \vec{B}$ and position of the particle. Four particle closest appearances of E_x, E_y and B_z are depicted by red, blue and green color, respectively.

$$\begin{aligned}(E_y)_p = {} & S_i(\xi)S_{j-\frac{1}{2}}(\eta)(E_y)_{i,j-\frac{1}{2}} + S_{i+1}(\xi)S_{j-\frac{1}{2}}(\eta)(E_y)_{i+1,j-\frac{1}{2}} \\ & + S_i(\xi)S_{j+\frac{1}{2}}(\eta)(E_y)_{i,j+\frac{1}{2}} + S_{i+1}(\xi)S_{j+\frac{1}{2}}(\eta)(E_y)_{i+1,j+\frac{1}{2}}\end{aligned} \tag{2.118}$$

$$\begin{aligned}(B_z)_p = {} & S_{i+\frac{1}{2}}(\xi)S_{j-\frac{1}{2}}(\eta)(B_z)_{i+\frac{1}{2},j-\frac{1}{2}} + S_{i+\frac{1}{2}}(\xi)S_{j+\frac{1}{2}}(\eta)(B_z)_{i+\frac{1}{2},j+\frac{1}{2}} \\ & + S_{i+\frac{3}{2}}(\xi)S_{j-\frac{1}{2}}(\eta)(B_z)_{i+\frac{3}{2},j-\frac{1}{2}} + S_{i+\frac{3}{2}}(\xi)S_{j+\frac{1}{2}}(\eta)(B_z)_{i+\frac{3}{2},j+\frac{1}{2}}\end{aligned} \tag{2.119}$$

Zero, the first and higher order weighting for field and density

Zero order weighting

Common name for this weighting is *nearest grid point* (NGP) [4]. The main idea is that the electric field is same as in x_i for all particles in the i^{th} cell; the total number of particles in one cell width Δx is simply counted and then assigned to the i^{th} grid point. Charge density is obtained by putting the charge and mass of a particle at the nearest i^{th} grid point. These choices produce zero self-force, as desired. Zero order weighting produce *rectangular* particle shape S with width Δx, thus we have finite-size particles. However this atitude does not need so much computational power, we have to choose better weighting, because zero-order weighting hardly alters the basic plasma effects which have to be studied [5]. Moreover the shape jumps up and down as the particle passes through the cell boundaries, so resulting field and density are too noisy.

The first order weighting

The first-order weighting smooths the density and field fluctuations, thus, relative to zero-order weighting, reduces noise but needs more computational power because of accesing two grid points for each particle twice per step. Used model is called *cloud-in-cell* (CIC) [4]. For variable i.e. electric field[5]:

$$E(x) = \left[\frac{X_{i+1} - x}{\Delta x}\right] E_j + \left[\frac{x - X_j}{\Delta x}\right] E_{j+1}, \tag{2.120}$$

the part assigned to i is $\left[\frac{X_{i+1}-x}{\Delta x}\right] E_j$ and the part assigned to $i+1$ is $\left[\frac{x-X_j}{\Delta x}\right] E_{j+1}$. Moreover terms $\left[\frac{X_{i+1}-x}{\Delta x}\right]$ and $\left[\frac{x-X_j}{\Delta x}\right]$ is more/less the shape-factor $S(\xi)$ given by (2.115). The first order weighting produces *triangular* particle shape S with width $2\Delta x$.
For charge density is process similar. For total cloud charge q_c, the part assigned to j is:

$$q_j = q_c \left[\frac{\Delta x - (x - X_j)}{\Delta x}\right] = q_c \left[\frac{X_{j+1} - x}{\Delta x}\right] \tag{2.121}$$

and part assignet to $j+1$ is:

$$q_j = q_c \left[\frac{\Delta x - (X_{j+1} - x)}{\Delta x}\right] = q_c \left[\frac{x - X_j}{\Delta x}\right] \tag{2.122}$$

In this case CIC can be the same as Particle-in-cell simulation, because assignment of a field at x to its nearest points by linear interpolation would produce the same result.

The second and higher order weighting
Apparently, quadratic, cubic and higher order weighting leads to bigger roughness in particle shape and reduce density and field noise, but costs a lot of computation. In addition, higher-order weighting and smoothing tent to reduce non-physical effects [5].

Finally, it is really important to use the same weighting in both density and force calculations in order to avoid a self-forces (i.e. particles accelerate themselves) [5].

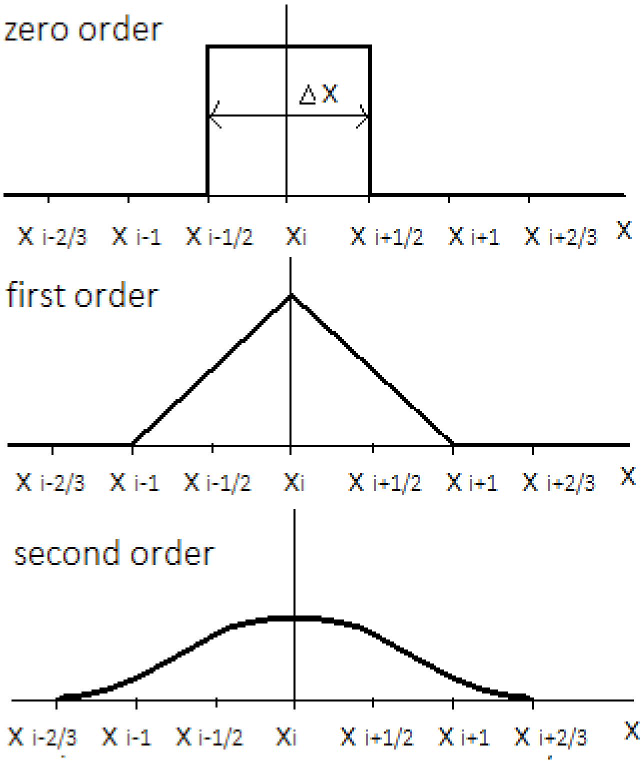


Figure 2.9: Zero, the first and the second order weighting functions.

2.5 Computational background

2.5.1 Extendable PIC Open Collaboration (EPOCH)

Figure 2.10: EPOCH logo

EPOCH is a project to develop a UK community advance relativistic EM PIC code written in Fortran. The code was published under the University of Warwick as a free full source code available for students and academic workers as CCP-project `http://ccpforge.cse.rl.ac.uk/gf/project/epoch/` (signup is required). Moreover, there is a possibility to download well-written users or developers manual or share with others your helping comments & problems via discussion forum, which is actually one of the reasons why we decided to use EPOCH.

The second reason was very user-friendly platform for setting input parameters. EPOCH is able to provide not only 1D and 2D simulations, but also 3D simulations as well. For parallel computing code needs MPI library (e.g. OpenMPI).

Development of the code

The core algorithm is developed by Dr. Chris Brandy and Dr. Keith Bennett at University of Warwick and based on the particle pusher and field update from Hartmut Ruhl's PSC code. The QED/semi-classical photon emission and pair production routines were developed by Dr. Christopher Ridgers and Dr. Roland Duclous and incorporated into EPOCH by Dr. Ridgers [6].

Input of parameters

Controlling of EPOCH code is user-friendly. With each new simulation a new `input.deck` file is created. In this file the parameters of simulation are inserted and organized into different blocks. Most of them use The International System of Units (SI) (the exception is particle charge or mass which is set in multiples of electron charge or mass, respectively). Each block serves its own purpose – proportions of simulation area, laser parameters, particle species, fields, boundary conditions, output options, etc. In EPOCH user manual a very detailed interpretation of different parameters and their possibilities are reported. Moreover, together with starting package (code + manual) also a few examples are downloaded for better understanding.

Output files and Visualization

Output data are saved in every time-step (which can be set as a parameter in `input.deck` file) of simulation into `.sdf` files (self-describing file). Then, these files can be loaded/plotted/visualized in Matlab or VisIt studio. VisIT studio allows to create time evolution (moving) images very easily, but works badly with big data files. Thus, it would be appropriate for 1D simulations only. In distribution of EPOCH code there are also the .sdf Matlab functions for working user's own scripts correctly.

2.5.2 MetaCentrum

Figure 2.11: MetaCentrum logo

2D or 3D PIC simulations is too computational-demanded for PCs. Thus, the simulation is prepared in personal computer and sent to the computing infrastructure (in our case called MetaCentrum) when it is accesed. Then, users are able to download computed data to their own computer.

Catch-all MetaCentrum Virtual Organization operates and manages distributed computing infrastructure consisting of computing and storage resources owned by CESNET as well as those of co-operative academic centers within the Czech Republic. The project tries to construct a virtual super-computer on which the tasks whose memory and/or CPU requirement are too severe could be solved. MetaCentrum membership is free for researchers and students of academic institutions in the Czech Republic, the members of the CESNET association, but including of the official appreciation formula to theirs publications is obliged [7].
Metacentrum is used by different users doing their reseach in various areas such as computational chemistry, material and structural simulation, simulation of flow of gases and liquids, recognition and speech generation, physical geodesy, ecological modelling, video processing, data mining or analysis of medical images [7].

How it works in few steps:

1. Use your name and password to login via .ssh client into MetaCentrum
2. Copy your simulation files that you prepared to your MetaCentrum account
3. Plan your job using the specification of requirements for computational resources
4. Wait until the job will not be done
5. Download the computed results into your computer
6. Visualize them by any tool, for instance Matlab or VisIt

To run simulations in MetaCentrum we must create a job and run it through the planning system PBS (Portable Batch System) (point (3)), which classifies jobs into queues according to the expected run time and to release the necessary computing resources for running the job. Simplified task could be defined in `.sh` script, e.g. `soubor.sh` as follows:

Listing 2.1: soubor.sh

```
1  #!/bin/sh
2  #PBS -N curvedfoil
3  #PBS -q short
4  #PBS -l nodes=1:ppn=12:cl_minos
5  #PBS -l mem=1gb
6  #PBS -j oe
7
8  cd ./curvedfoil
9  module add openmpi-1.6-intel
10 echo Data | mpirun -np 12 ./bin/epoch2d
```

On the second line we set name of the job, on the third row we choose the queue (`-short` specifies maximum run time in the length of 1 hour), the fourth line is requesting 1 node (computer) having 12 processors (Minos cluster) and 1GB of memory (row 5). Then, we have to specify what next have to happen. We move to the folder curvedfoil (line 8), which contains the file with epoch2d code and our input parameters file. Library for parallel computing is loaded on line 9. The job is added into MetaCentrum planning system by `-qsub` command in linux terminal.

Through specifying requirements for computational resources we can set many parameters such as [1]:

- maximal amount of CPU time exhausted by all processes belonging to a job
- maximal amount of real time during which a job can be running
- number and/or typ of nodes that should be reserved for a job
- used amount of physical memory

Detailed description not only for setting these parameters can be found on official MetaCentrum wiki page:
`https://wiki.metacentrum.cz/wiki/Scheduling_system_-_detailed_description`.

Chapter 3

Obtained results from PIC simulations

In this work two types of targets – flat foil and curved foil section – are studied in order to decrease divergence of accelerated proton beam. For this purpose 2D Particle-in-cell EPOCH code is used and obtained results are visualized by Matlab. Specifically, we are interested primarily in energy and angular distribution of accelerated ions and energy balance. For simplicity and potentional control we decided to use simulation parameters used in Ref. [31]. Moreover, the output data are used in Matlab program for computing particle trajectories in magnetic solenoid and calculating emittances of the beam in chapter 4.

3.1 Implementation of energy balance

In current version of EPOCH code (4.3.4), same as in the previous ones, energy balance and computing of absorption coefficient do not work well. Actually, energy balance is essential for initial check-up of simulation progress and both for results interpretation. Thus, the procedure how to get dimensionless energy balance [23] is described bellow in section 3.1.1.

3.1.1 Dimensionless energy balance

To obtain dimensionless equivalents $\vec{E'}, \vec{B'}$ (3.3) of electric and magnetic field $\vec{E}, \vec{B}$ we have to specify its units and then multiply them by appropriate variables (electron charge e, electron mass m_e, laser angular frequency ω and speed of light c):

$$[E] = \frac{kg \cdot m}{A \cdot s^3}, \qquad [B] = \frac{kg}{A \cdot s^2}, \tag{3.1}$$

$$[e] = A \cdot s, \qquad [m_e] = kg, \qquad [\omega] = s^{-1}, \qquad [c] = m \cdot s^{-1}, \tag{3.2}$$

$$\frac{e\vec{E}}{m_e \omega c} =: \vec{E'}, \qquad \frac{e\vec{B}}{m_e \omega} =: \vec{B'}, \tag{3.3}$$

$$[E'] = [-], \qquad [B'] = [-]. \tag{3.4}$$

Then, dimensionless mass and charge is given by:

$$\frac{m}{m_e} =: m', \qquad \frac{n}{n_c} =: n', \tag{3.5}$$

where n_c is critical density.

Total energy can be obtained by the sum of its particles and field part:

$$W_{field} = \frac{1}{2}\sum_{cell}\left((E')^2 + (B')^2\right), \tag{3.6}$$

$$W^i_{particles} = \sum_{particles} m'_i \cdot n'_i \cdot (\gamma - 1), \tag{3.7}$$

$$W_{particles} = \sum_i W^i_{particles}, \tag{3.8}$$

$$W_{total} = W_{field} + W_{particles}, \tag{3.9}$$

where γ is relativistic factor given by:

$$\gamma = \sqrt{1 + \frac{p_i}{m_i c}}. \tag{3.10}$$

Transformation to dimensionless equivalents of $\vec{E}, \vec{B}$ can be implemented in Matlab very easily, for example:

Listing 3.1: Dimensionless energy balance

```
e=-1.6021766e-19;
m_e=9.10938e-31;
c=299792458;
omega=2*pi*c/(lambda*1e-9);

l=GetDataSDF(number.SDF);
Ex=l.Electric_Field.Ex.data.*e./(m_e*omega*c);
Ey=l.Electric_Field.Ey.data.*e./(m_e*omega*c);
Ez=l.Electric_Field.Ez.data.*e./(m_e*omega*c);
Bx=l.Magnetic_Field.Bx.data.*e./(m_e*omega);
By=l.Magnetic_Field.By.data.*e./(m_e*omega);
Bz=l.Magnetic_Field.Bz.data.*e./(m_e*omega);
```

This approach is used in energy balance script (see section 3.4) which allows us to get time evolution of energy (e.g. W_{field} (3.6), $W_{particles}$ (3.8) and W_{total} (3.9)).

3.2 Flat foil section

3.2.1 Setting input.deck file

Firstly, the simulation for a flat foil section was run. I have used 2D version of EPOCH PIC code. Here, the `input.deck` file is demonstrated and explained in detail.

Blocks

As it is explained in the EPOCH section 2.5.1, the `input.deck` file is composed from different "blocks". The parameters essential for the simulation are explained here. Meaning of other aspects (usually related to properties of Gaussian beam) which are not discussed here could be found in Ref. [2].

Listing 3.2: Control block

```
1  begin:control
2    nx = 4400
3    ny = 4000
4    t_end = 300 * femto
5    #t_end = 6 * femto  # short test simulation
6    x_min = -12.8 * micron
7    x_max = 22.4 * micron
8    y_min = -16 * micron
9    y_max = -y_min
10   stdout_frequency = 10
11 end:control
```

In *control block* the global number of gridpoits in x and y direction (`nx` and `ny`, respectively) and size of the domain (line $6-9$) were set. The reason of non-symetric area around $x = 0$ is that more space behind the target is needed to study the physical phenomenas of proton acceleration from foil rear side (which is situated in the centre of the coordinate system). The simulation were running for 300 fs (row 4).

Listing 3.3: Boundaries block

```
begin:boundaries
  cpml_thickness = 16
  cpml_kappa_max = 20
  cpml_a_max = 0.2
  cpml_sigma_max = 0.7
  bc_x_min = cpml_laser
  bc_x_max = cpml_outflow
  bc_y_min = cpml_outflow
  bc_y_max = cpml_outflow
end:boundaries
```

CPML is Convolutional Perfectly Matched Layer boundary conditions and they are set as default values from EPOCH. For more information see [2].

Listing 3.4: Constant block

```
begin:constant
  las_lambda0 = 800 * nano        // laser wavelength
  foil_thickness = 3 *las_lambda0  // thickness of the foil
  foil_width     = 4 *las_lambda0   // foil width
  cell_xsize = las_lambda0/100
  nx_foil  = foil_thickness / cell_xsize  // number of cells in foil
  cell_ysize = las_lambda0/100
  ny_foil  = foil_width / cell_ysize  // number of cells in foil
  las_omega = 2.0 * pi * c / las_lambda0  // angular laser frequency
  las_time = 2.0 * pi / las_omega   // laser wave period
  theta = 0.0            // incidence angle (in rad)
  fwhm0 = 2.5* las_lambda0         // laser beam width (FWHM)
  w0 = fwhm0 / (2.0*sqrt(loge(2.0)))  // laser beam waist
  f_duration= 30 * femto           // full laser pulse duration
  xfocus = 12.8 * micron      // distance of laser focal spot along x-axis
  yc = -xfocus * tan(theta)  // distance of laser beam axis from central axis
  zR = pi * w0^2 / las_lambda0    // rayleigh range
  lfocus = xfocus / cos(theta)     // along laser propagation direction
  wl = w0 * sqrt(1.0+(lfocus^2)/(zR^2)) // spot size at distance lfocus
  w_y = wl / cos(theta)     // spot size in the projection on y-axis
```

```
21   r_curv = lfocus * (1.0+(zR^2)/(lfocus^2))  // radius of beam curvature
22   intensity_fac2d = 1.0 / sqrt(1.0+lfocus^2/zR^2) * cos(theta)
23   n_crit = critical(las_omega)  // electron critical density
24  n_max  = 20 * n_crit          // maximum initial electron density
25 end:constant
```

In *Constant block* a various constants and parameters needed in others blocks are prepared. The foil is 2400 nm thick and 3200 nm wide, these dimensions are specified on lines 3, 4. Then, number of cells occupying by plasma created from the ionized foil are computed on lines 6, 8. For setting intensity in *laser block* the factor `intensity_fac2d` is needed and its meaning is demonstrated in Fig. 3.1. Critical density and maximum initial electron density are set on lines 25 and 26, respectively. Other constants are shortly described in the code and usually are related to the properties of Gaussian beam.

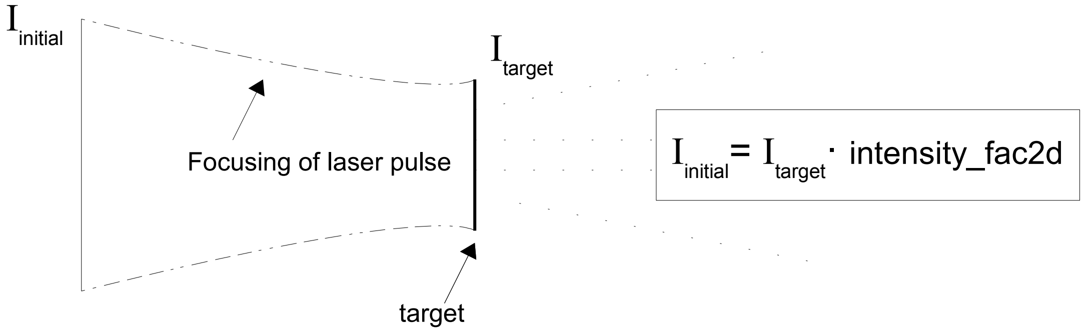


Figure 3.1: Graphically demonstrated meaning of *intensity_fac2d* parameter used in *Constant block*.

Listing 3.5: Particle species block

```
begin:species
  name = electron
  charge = -1.0
  mass = 1.0
  npart = 150 * nx_foil * ny_foil
  density = if((x lt foil_thickness) and (x gt 0), n_max, 0.0)
  density = if((abs(y) lt (foil_width/2.0)), density(electron), 0.0)
  temp_ev = 1000
  identify:electron
end:species

begin:species
  name = proton
  charge = 1.0
  mass = 1836.0
  npart = 30 * nx_foil * ny_foil
  density = density(electron) * 1.0/5.0
  temp_ev = 1000
end:species

begin:species
  name = carbon
  charge = 4.0
```

```
24   mass = 22032.0
25   npart = 30 * nx_foil * ny_foil
26   density = density(electron) * 1.0/5.0
27   temp_ev = 1000
28 end:species
```

Species block defines all particles participating in the simulation. Here, electrons are taken as a base (with mass 1) and other species are formulated as its multiple. Actually, it is the exception in SI units in EPOCH code. Density of electrons is defined only in the space where the foil is present (see *"if"* conditions on lines 6 and 7). C^{4+} ions are at ratio to protons 1 : 1. Because of charge-neutrality of the system, carbon ions as same as protons have to be one fifth of electrons amount (defined as `density` on lines 17 and 26). This is also the reason of coefficients 150, 30, 30 in `npart` on lines 5, 16 and 25, respectively.

Listing 3.6: Laser block

```
1  begin:laser
2    boundary = x_min
3    intensity_w_cm2 = 4.5e19 * intensity_fac2d
4    lambda = las_lambda0 * cos(theta)
5    pol_angle = 0    // normal incidence
6    phase = -2.0*pi*(y-yc)*sin(theta)/las_lambda0 + pi/(las_lambda0*cos(theta))
       *((y-yc)/cos(theta))^2*1.0/r_curv - atan(((y-yc)*sin(theta)+lfocus)/zR)
7    profile = gauss(y,yc,sqrt(2)*w_y)
8    t_profile = (sin(time/(f_duration)*pi))^2
9    t_start = 0.0
10   t_end   = 30*femto
11 end:laser
```

In the simulation, the laser of wavelegth $\lambda = 800$ nm, full duration of the laser pulse 30 fs and intensity $\sim 10^{19}$ W$\cdot$cm^{-2} (computed on line 3) was used. Laser beam has normal incidence to the target and comes along the x-axis in the positive direction. Spatial profile of the laser pulse is Gaussian and time-profile has sinusoidal shape.

Listing 3.7: Output block

```
begin:output
  //  number of timesteps between output dumps
  dt_snapshot = 15* femto
  restart_dump_every = 8
  // properties at particle positions
  particles = always + species + single
  px = always + single
  py = always + single
  pz = always + single
  // properties on grid
  grid = always + single
  ex = always + single
  ey = always + single
  ez = always + single
  bx = always + single
  by = always + single
  bz = always + single
  number_density = always + species + single
end:output
```

In *Output* settings we want to save data about all spatial components of momentum (at particle positions), magnetic and electric field (on grid) every 15 fs. Moreover particle positions and densities are written down for each species separately. We get 20 `.sdf` files with needed data for visualisation, as the simulation calculates 300 fs of plasma evolution.

3.3 Curved Foil

Curved foil, as already discussed in theoretical part 1.4.1, could be very effective in the question of decreasing beam divergence because of its geometry, which is exactly the reason why this shape was picked for further examination.

In the *curved foil* simulation, the *Control, Boundaries, Laser* and *Output block* stay the same as in *flat foil* 3.2 case. Thus, only *Constant* (most of the parameters were already described) and *Species* block are discussed bellow. Demonstrated are only those parameters which are different from flat foil section. Continuation of the block in same way as in flat foil `input.deck` is indicated by dots.
The curved foil was described in *Species block* via non-zero density inside the blue target (see Fig. 3.2) created by two same circles with two common intersection points. As demonstrated in Fig. 3.2, the final $3\lambda \times 4\lambda$ target curvature radius is set to 4λ and his middle axis lies on x-axis, thus, coordinates $y1, y2$ are equal to zero.

Listing 3.8: Constant block

```
begin:constant
  las_lambda0 = 800 * nano
  x1= 4 *las_lambda0
  R1= 4 *las_lambda0
  x2= 7 *las_lambda0
  R2= 4 *las_lambda0
  .
  .
end:constant
```

Constant block is nearly the same like that in *flat foil case.* In addition to the parameters defined for flat foil case (see listing 3.4), we have to determine parameters $x1, x2, R1, R2$ because they are necessary for the definiton of the shape of curved foil in *Species block* according to the Fig. 3.2.

Listing 3.9: Species block

```
begin:species
  name = electron
  charge = -1.0
  mass = 1.0
  npart = 150 * nx_foil * ny_foil
  density=if((x-x1)^2+y^2 lt R1^2, n_max, 0.0)
  density=if((x-x2)^2+y^2 lt R2^2,0.0, density(electron))
  density=if(abs(y) lt 2*las_lambda0, density(electron))
  temp_ev = 1000
  identify:electron
end:species
.
.
```

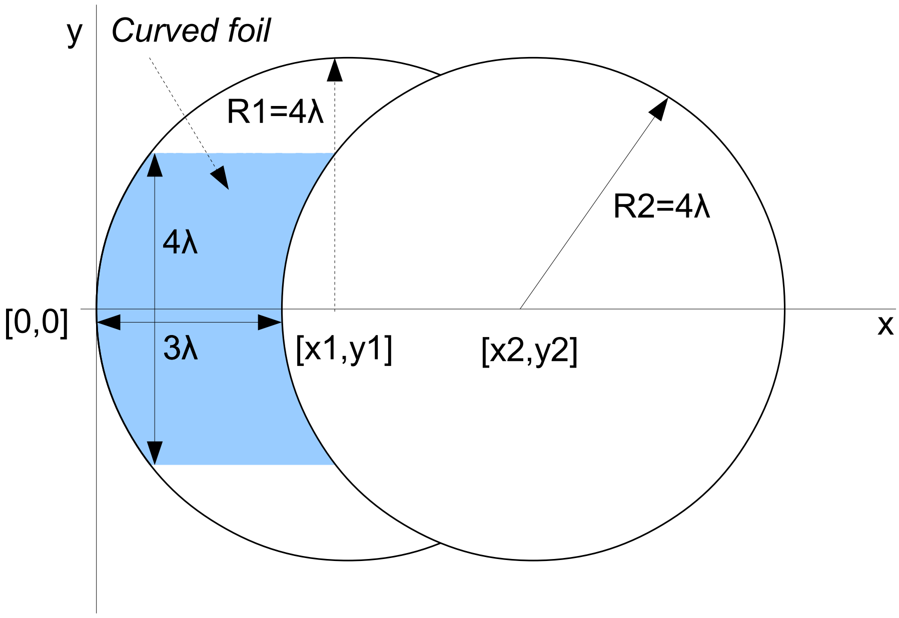


Figure 3.2: putting parameters of the foil in PIC input.deck file: Creating curved foil by two circles and its parameters

3.4 Results from PIC simulations

Energy spectrum

Comparison between energy spectrum of protons and C^{4+} ions for flat foil and curved foil section is shown in Fig. 3.3. Energy spectrum is created for last `.sdf` files, i.e. at the end of the simulations at the time 300 fs. Maximum energies per atomic mass unit u of each species (protons and carbon ions) are following:

Flat foil section:

$$E_{max}^{p_f+}/u = 3.4495 \text{ MeV/u} \tag{3.11}$$

$$E_{max}^{C_f^{4+}}/u = 0.5369 \text{ MeV/u} \tag{3.12}$$

Curved foil section:

$$E_{max}^{p_c+}/u = 4.0471 \text{ MeV/u}, \tag{3.13}$$

$$E_{max}^{C_c^{4+}}/u = 0.5519 \text{ MeV/u}. \tag{3.14}$$

Higher maximum proton energy (as well as maximum energy of carbon ions) E_{max}^{p+} were obtained in the case of curved foil section simulation. In explanation, we have to start with absorption. For curved foil case, laser pulse is incident on target surface at various angles in the contrast to flat foil case in which laser pulse is incident perpendicularly to the target surface, e.g. comes to the front side at zero angle. Electrons heating mechanisms, such as Brunel vacuum heating (described in section 1.3.1), are often inefficient when laser pulse is incident at zero angle; see Fig. 1.3. On the other hand, some of heating mechanisms (for example $\vec{j} \times \vec{B}$ heating described in section 1.3.2) apply also for normal incidence. In other words, curved foil case has bigger absorption coefficient κ (because angles of incidence are varying), which means that more energy from incident laser pulse is transformed into kinetic energy of particles. Obviously, the highest theoretical value of total energy which particles can gain is equal to the energy of electromagnetic field given by the laser pulse. Absorption coefficients for both targets are evaluated in Energy balance results (section 3.4), Tab. 3.3.

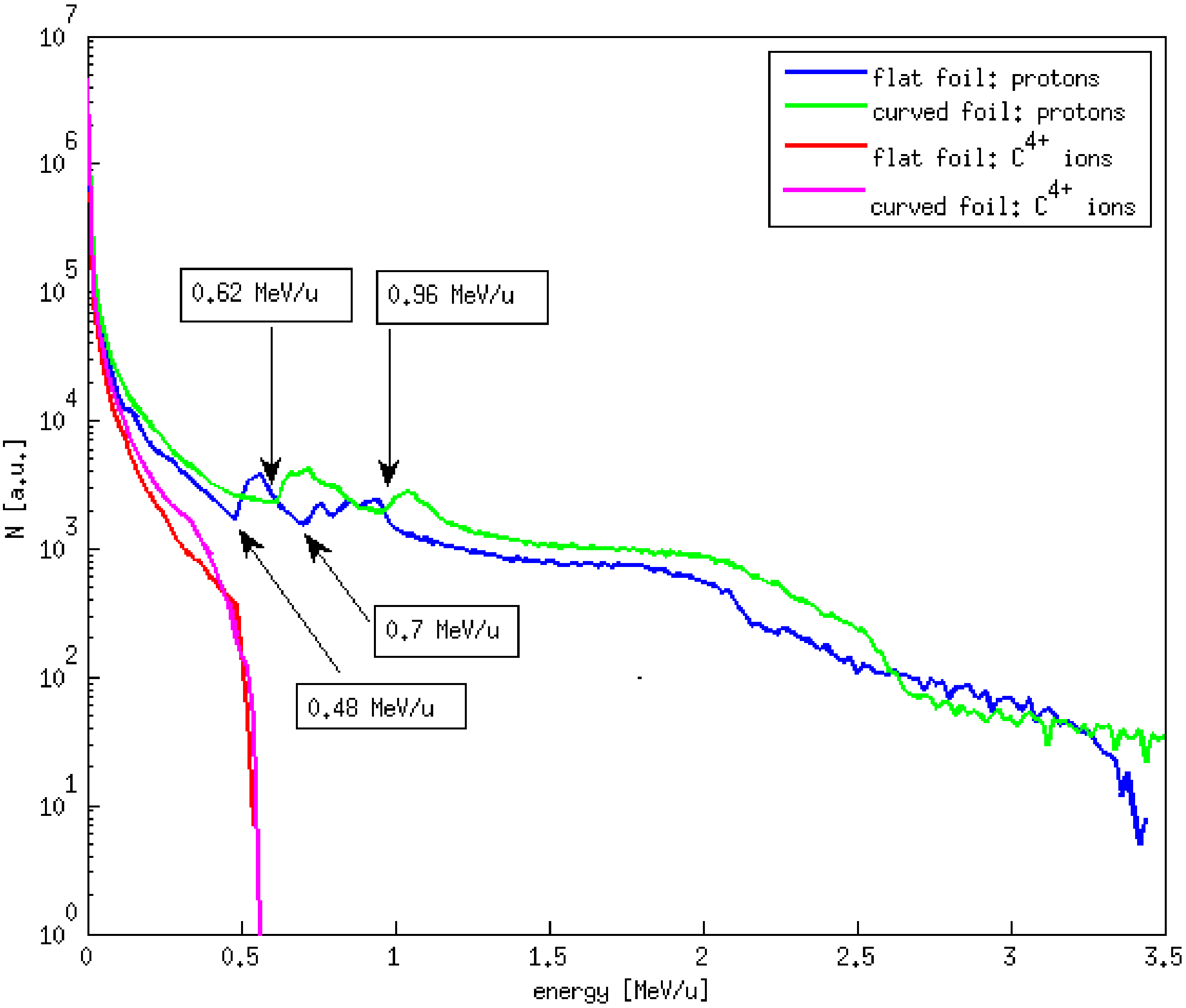


Figure 3.3: Energy spectrum of protons and carbon ions C^{4+} for the cases of flat foil section and curved foil section at the end of simulation at $t = 300$ fs.

Breakpoints where the number of protons is suddenly steeply increased are equal to 0.48 MeV/u and 0.7 MeV/u (flat foil section), 0.62 MeV/u and 0.96 MeV/u (curved foil section); see Fig.

3.3. This thresholds are given by transverse dimensions of the target relative to the direction of propagation of the incident laser pulse. If the target had infinite transverse dimension (a situation similar to experiments), the increase in the number of protons would be never appeared. Breakpoints in energies of protons are created on high-energy carbon impurities locations as demonstrated of Fig. 3.4.

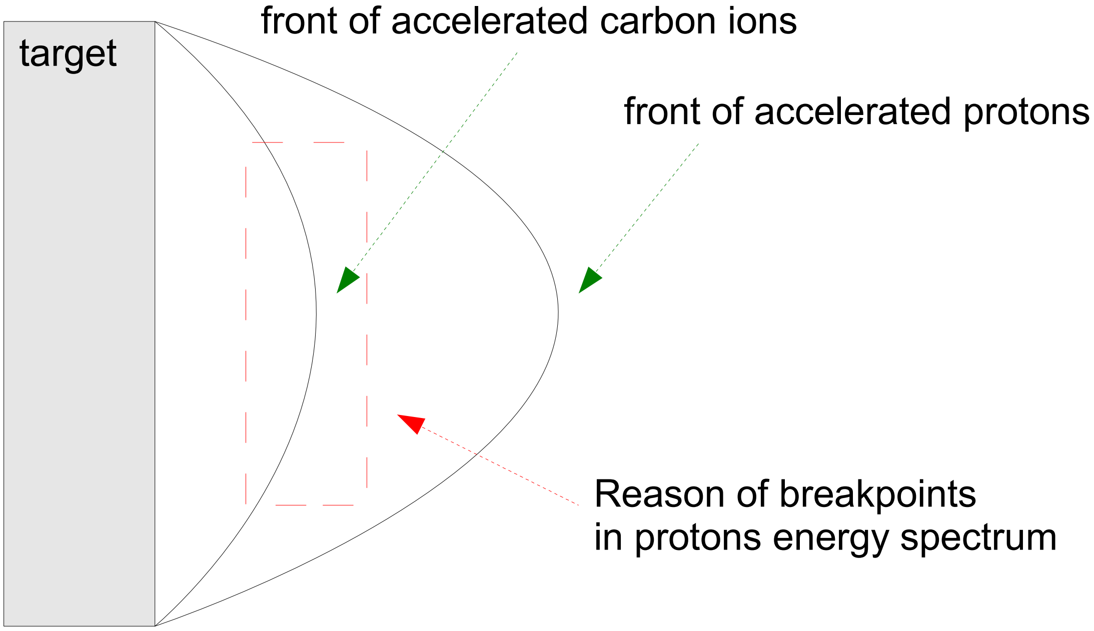

target
front of accelerated carbon ions
front of accelerated protons
Reason of breakpoints
in protons energy spectrum

Figure 3.4: Accelerated protons and C^{4+} ions; the most accelerated carbon ions are reason why breakpoints in proton energy spectrum are present.

Fig. 3.4 also demonstrates that protons are accelerated more than C^{4+} ions because of their more favorable q/m ratio. Furthermore, protons are shielding the field before carbon ions, thus, these ions cannot be accelerated to such high energies (per atomic mass unit) as the protons. Actually, the possibility of easy comparison of accelerating energies is the reason why we use division by atomic mass unit u. The resulting energy we give in [MeV/u].

(Note: Visualisation matlab script is shown in Appendix 5.)

Angular spectrum

Comparison between angular distributions of accelerated protons from flat and curved foil section is plotted in Fig. 3.5. Both distributions are nearly Gaussian and they are plotted at time 300 fs, i.e. at the end of simulations. In both cases, only protons with energies above 0.5 MeV are taken into account.

In Fig. 3.5 we can observe that protons accelerated from curved foil have less divergence, only $-4.2°$ to $4.2°$ in FWHM than protons accelerated from flat foil collimated between $-5.1°$ to $5.1°$ in FWHM. All half angle divergences obtained in each energy interval are summarized in Tab. 3.1 and Tab. 3.2 for flat foil and curved foil section, respectively. In reality, divergence decreases with increasing particle energy [15]. In Tab. 3.1, this statement is broken for the second and the fourth interval, i.e. intervals where breakpoints in energy spectrum (see Fig. 3.3) are present. Similarly for curved foil (see Tab. 3.2), divergence statement is broken for the second interval and in the fourth interval the drop in half angle is smaller than usual. For both target shapes divergence values for wide energy interval are close to typical thin foil divergence $(2 - 4.6°)$ [3] obtained from 2D modeling; see section 1.4.1. In fact, standart values of divergence for flat foils acquire values from 0° to 25° in half angle as discussed in chapter 1.4. In shortl, the protons are accelerated perpendicularly to the target from its rear side which is exactly the reason why the more collimated protons are observed in the case of curved foil (see chapter. 1.4.1).

(Note: Matlab code for obtaining these distributions is presented in Appendix 5.)

E_{min} [MeV/u]	E_{max} [MeV/u]	half angle divergence [°]
0.00	0.69	40.2
0.69	1.38	4.7
1.38	2.07	6.6
2.07	2.76	9.5
2.76	3.45	5.9
0.50	3.45	5.1

Table 3.1: Summary: half angle divergence in each energy interval for flat foil section (values of energy are rounded, but they were taken precisely in the calculations)

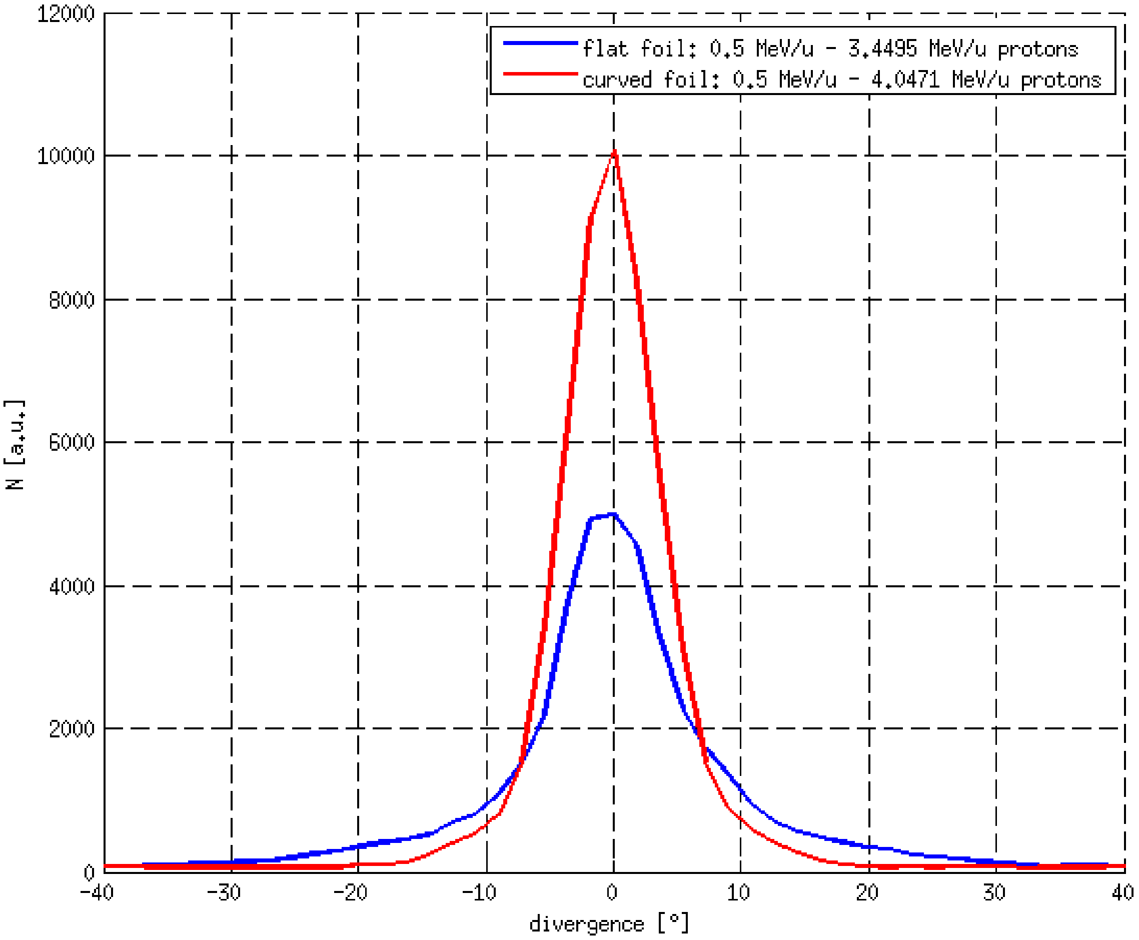


Figure 3.5: Angular distributions of accelerated protons at 300 fs for flat foil section (blue) and curved foil section (red); Only protons accelerated to energies above 0.5 MeV/u are taken into account.

E_{min} [MeV/u]	E_{max} [MeV/u]	half angle divergence [°]
0.00	0.81	45.6
0.81	1.62	3.8
1.62	2.43	4.1
2.43	3.24	4.0
3.24	4.05	2.7
0.50	4.05	4.2

Table 3.2: Sumarry: half angle divergence in each energy interval for curved foil section (values of energy are rounded, but they were taken precisely in the calculations)

Energy balance

Absorption coefficient is equal to maximum kinetic energy after the interaction divided by the maximum energy of the electromagnetic field in the simulation domain before the interaction. It is important to take the value of maximum field energy at time when the whole laser pulse is present in the simulation area. Our conditions are satisfied for times $t_{before} = 30fs$ and $t_{after} = 300$ fs. Finally, the relation for absorption coefficient κ is then:

$$\kappa = \frac{W_{max}^{kin}(t_{after})}{W_{max}^{field}(t_{before})}, \tag{3.15}$$

where W_{max}^{kin} is maximum kinetic energy and W_{max}^{field} is maximum energy of laser (EM) field, same notations are used in Energy balance section 3.1.

Values of absorption coefficient κ for both targets are listed in Tab. 3.3. Curved foil has bigger absorption coefficient (more energy from incident laser pulse is transformed into kinetic energy of particles), because incident angle of laser pulse hitting the target is varying (target is curved), thus, more electron heating mechanisms are efficient. Further explanation can be found in section 3.4.

	curved foil	flat foil
κ [%]	21.4	15.6

Table 3.3: Values of absorption coefficient κ (3.15) for flat foil and curved foil cases.

Moreover, energy balance displaying progress of total energy W_{tot}, total kinetic energy W_{kin}, field energy W_{field} and kinetic energies of protons W_{kin}^{p+}, carbon ions $W_{kin}^{C^{4+}}$ and electrons W_{kin}^{e-} in time is depicted in Fig. 3.6 for the flat foil section. Due to the fact that the energy balance graph for curved foil section is very similar, it is not shown here, only absorption coefficient is computed in Tab. 3.3. Energy variables are computed according to equations in section 3.1 and all of them use dimensionless magnetic and electric field $\vec{B}'$, $\vec{E}'$ (3.3).

In Fig. 3.6, we can observe that laser energy W_field starts decreasing from 50 fs to 100 fs when energy of laser pulse is absorbed by electrons (W_kin_e rises). The most of the pulse is reflected from the target and from the time 100 fs is absorbed on the edge of simulation area. Then, kinetic energy of electrons W_kin_e is gradually transfered into ion energy W_kin_p and W_kin_p till the end of the simulation.

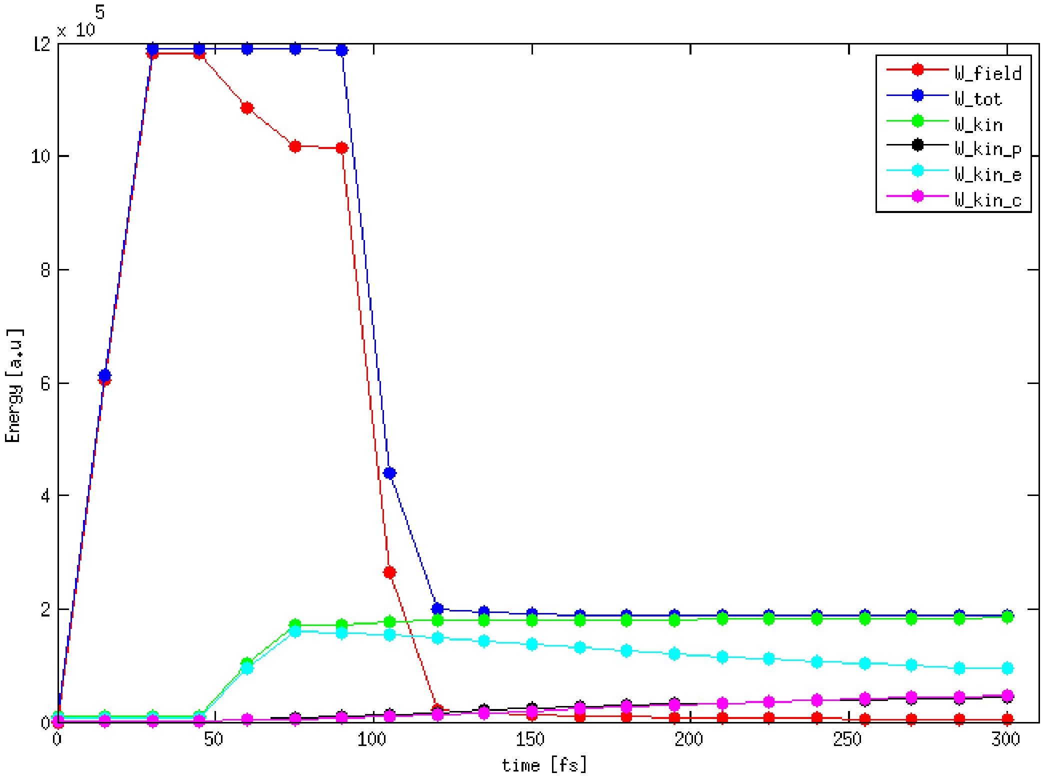


Figure 3.6: Energy balance for flat foil section; total energy W_tot, kinetic energy of all particles W_kin, field energy W_field, kinetic energy of protons W_kin_p, kinetic energy of electrons W_kin_e and kinetic energy of C^{4+} ions W_kin_c.

Chapter 4

Input for BT program (Matlab)

The goal was to connect data from PIC simulations with the Matlab program created for the purpose of my bachelor thesis [42]. The code I have developed in Matlab can compute the trajectory coordinates of a charged particle beam in a solenoid with given starting conditions for the position and the velocity coordinates. The charged particle motion through the solenoidal field was written in a cylindrical coordinate system, thus, the transformation into Cartesian coordinates is used only for plotting. Moreover, the program contains also the file for solving differential equation of motion and the file for generating the particle beam emittance. By changing parameters I have studied the behaviour of charged particles (or a single particle) in the solenoidal field – for example a fringing field effect. In this work, our goal was to take the output from PIC simulations as input to the Matlab code and see if our results correspond to reality.
There were a few problems:

- Firstly, the PIC simulations were computed using 2D EPOCH code in contrast to Matlab program which computes particles trajectories in 3D. As a solution of this problem, the θ-component of velocity v_θ was set to 0 and angular position coordinate θ we obtain from PIC simulation as an angle between velocities v_x and v_y (v_r and v_z, respectively); directions can be seen in Fig. 3.2.

- Secondly, to obtain emittance the equation for the RMS emittance (4.1) from [15] is used instead of the equation of emittance ellipse in phase space characterized by Twiss parameters α, β, γ (this approach is applied in Ref. [42]).

$$\varepsilon_{RMS} = 4\sqrt{\sum_{i=1}^{N} \frac{1}{N} \left(y_i - \langle y \rangle\right)^2} \times \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left(\theta_i - \langle \theta \rangle\right)^2} \tag{4.1}$$

Actually, the Twiss parameters β, γ can be computed as follows [42]:

$$\sigma_y = \sqrt{\frac{1}{N} \cdot \sum_{i=1}^{N} \left(y_i - \langle y \rangle\right)^2} = \sqrt{\varepsilon_{RMS} \cdot \beta} \qquad \rightarrow \qquad \beta = \frac{\frac{1}{N} \cdot \sum_{i=1}^{N} \left(y_i - \langle y \rangle\right)^2}{\varepsilon_{RMS}} \tag{4.2}$$

$$\sigma_\theta = \sqrt{\frac{1}{N} \cdot \sum_{i=1}^{N} \left(\theta_i - \langle \theta \rangle\right)^2} = \sqrt{\varepsilon_{RMS} \cdot \gamma} \qquad \rightarrow \qquad \gamma = \frac{\frac{1}{N} \cdot \sum_{i=1}^{N} \left(\theta_i - \langle \theta \rangle\right)^2}{\varepsilon_{RMS}} \tag{4.3}$$

More detailed description can be found in my bachelor thesis [42].

- Then, the kinetic proton energy was set to a constant in a matlab program. It has to be changed manually, if needed.
- Non-solved problem is also the diferent numbers of particles – thousands or milions in PIC and only hundreds in matlab, which is computationaly very limited.

4.1 Beam Emittance correction (demonstrated in flat foil section case)

In the case of flat foil we have used as a data source the last .sdf file (`20.sdf`). This file contains information about accelerated protons at the end of the simulation, i.e. at 300 fs from the start. Firstly, we have to check that the accelerated protons are still in the simulation area. Thus, the plotted protons density at the end of simulation is shown as an example in Fig. 4.1. None of particles come close to the edge of the simulation area. This process were aplied also on curved foil section with the same result (at the end of simulation none of particles come close to the edge of simulation area).

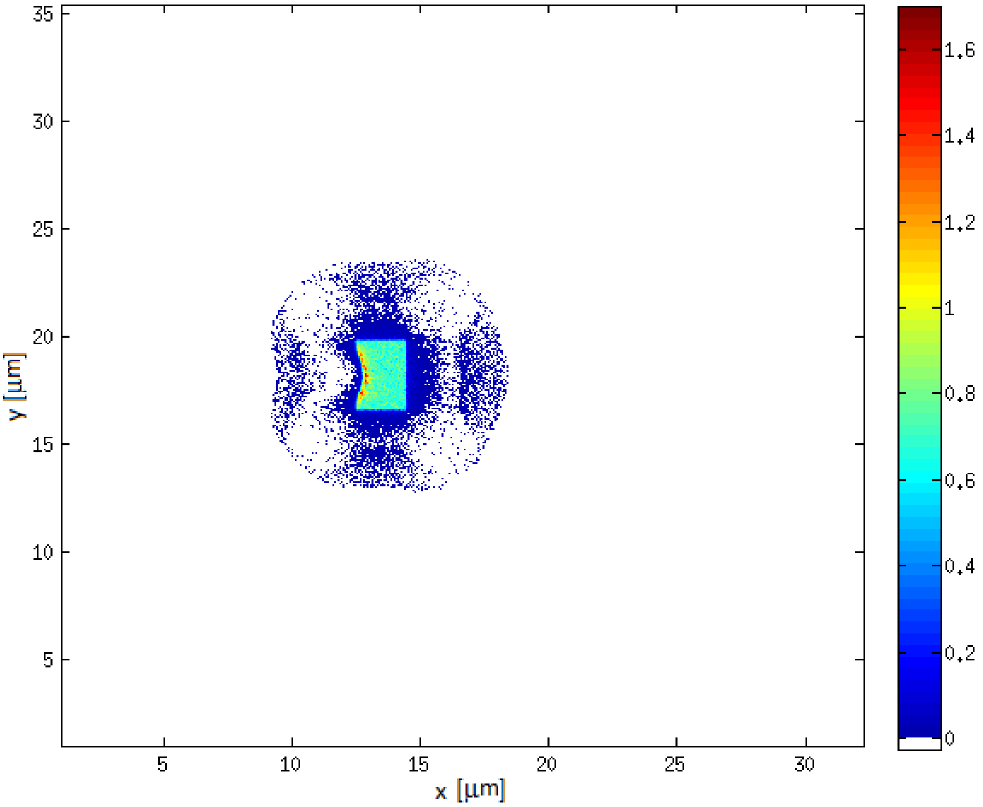


Figure 4.1: Expansion of the target at $t = 300$ fs; none of the protons come close to the edge of the simulation area; color scale shows protons density

4.1.1 Calculation of Emittance and Twiss parameters

Based on the data from PIC simulations, we determine emittance (see (4.1)) and Twiss parameters β, γ (obtained from (4.2), (4.3)). The spectrum of accelerated protons in time 300 fs was divided in 5 energy intervals between $E^{p+}_{min} = 0$ MeV and $E^{p+}_{max} = 3.4495$ MeV (3.11) for flat foil section and between $E^{p+}_{min} = 0$ MeV and $E^{p+}_{max} = 4.0471$ MeV 3.13, thus, 5 sets of emittance plus Twiss parameters for each foil are received. Data for flat foil section are summarized in Tab. 4.1 and for curved foil section in Tab. 4.2, where `eps_y` and `eps_theta` are square roots in relation (4.1).

E_{min} [MeV]	E_{max} [MeV]	ε [mm×mrad]	eps_y [mm]	eps_θ [rad]	β [m/rad]	γ [rad/m]
0.00	0.69	1.54	8.36E-007	0.46	4.54E-007	1.38E+005
0.69	1.38	0.79	1.09E-003	0.18	1.50E-006	4.16E+004
1.38	2.07	2.17	1.81E-003	0.30	1.52E-006	4.13E+004
2.07	2.76	1.23	1.53E-003	0.20	1.92E-006	3.26E+004
2.76	3.45	0.27	8.22E-004	0.08	2.54E-006	2.46E+004

Table 4.1: Flat foil section: Results of computed emittances and Twiss parameters β (4.2) and γ (4.3) from last output .sdf file of PIC simulation for flat foil section in time $t = 300$ fs; results are divided between five energy intervals (see E_{min} and E_{max}); *eps_ y* and *eps_ theta* are square roots in equation (4.1).

E_{min} [MeV]	E_{max} [MeV]	ε [mm×mrad]	eps_y [mm]	eps_θ [rad]	β [m/rad]	γ [rad/m]
0.00	0.81	1.48	8,10E-004	0.46	4,43E-007	1.41E+005
0.81	1.62	0.46	8.27E-004	0.14	1.47E-006	4.24E+004
1.62	2.43	1.45	1.54E-003	0.23	1.64E-006	3.81E+004
2.43	3.24	0.06	3.27E-004	0.05	1.66E-006	3.76E+004
3.24	4.05	0.02	1.90E-004	0.03	1.58E-006	3.96E+004

Table 4.2: Curved foil section: Results of computed emittances and Twiss parameters β (4.2) and γ (4.3) from last output .sdf file of PIC simulation for curved foil section in time $t = 300$ fs; results are divided between five energy intervals (see E_{min} and E_{max}); *eps_ y* and *eps_ theta* are square roots in equation (4.1).

The emittance `eps` (in tables noted as ε) reaches ones or tenths of [mm x mrad] for flat foil section (see Tab. 4.1) which is in agreement with theoretical articles based on particle-simulations, e.g. [15]. Emittance and standard deviation of divergence is lower for the curved foil section than for the flat foil case (see Tab. 4.2). Moreover, the angular deviation `eps_theta` for both cases has the lowest value for the higher energy intervals which is in agreement with statement that the most energetic particles have the less angular divergence [15]. In fact, `eps_theta` has a a decreasing trend except the second interval for flat foil as same as for curved foil. This mis-values may be explained by two breakpoints in proton energy spectrum (see Fig.3.3) given by transverse dimension of target (more detailed explanation in section 3.4).

4.1.2 Modification in the Matlab code (from bachelor thesis) for the purposes of the new input data from PIC simulations (flat foil section only)

In previous section 4.1.1 we obtain Twiss parameters β, γ which are used as input in *beam emittance* Matlab file. All needed Matlab files can be found in Appendix 5. The corrected files are following:

1. *Main code* – new saving of velocity coordinates; input of new starting energy (2.5 MeV); moving of the solenoid closer to the particle-source (`zini` is 0.001 m) to catch more particles (when divergence is huge).

2. *Beam emittance* – input of new "typical" values of Twiss parameters (see Tab. 4.1):

   - $\beta = 1.5 \cdot 10^{-6}$ m/rad,
   - $\gamma = 3 \cdot 10^{4}$ rad/m;

   Emittance is not calcaluted from emittance elipse equation [42].

3. *Plot emittances* – nearly all new file.

New ***Plot emittances*** Matlab file in the contrast to the old one uses only one 2D plane in phase space as an input, because of problematic transition between 3D (the code in Ref. [42]) and 2D (PIC simulations). As we can observe in Fig. 4.2 b), d) the second plane of phase space is set to zero at the beginning of the simulation, but does not remain zero at the end. In other words, particles do not stay in one plane after passing their paths through the solenoid.

Plot emittances file plots the emittance at the start and at the end of Matlab simulation and takes into account only those particles that are propagating through the solenoid (not the particles which are reflected back), Fig. 4.2. Simulated data can be found in Tab. 4.3. Emittance `eps` is bigger at the end which is unfortunately not expected result and it is caused by non-physical situation connected to solenoid dimensions and its position. The solenoid is very closed to the particle source, becase those from PIC simulation have a large divergence from the start, thus, only way how to catch them into the magnetic device is to put the solenoid closer to the source. Even if this situation differs from reality (solenoid is usually located in distance of few *cm*), the possibility of connecting Matlab code with PIC simulations is demonstrated. In addition, we are planning to focus on this problem more in the future together with optimization of solenoid field in order to achieve the improvement in ion beam divergence by the solenoid.

	eps_y [mm]	*eps_theta* [mrad]	*eps* [mm x mrad]
at the start	0.002	309.019	0.618
at the end	0.153	883.009	135.100

Table 4.3: Comparison of emittances at the start and at the end of Matlab simulation.

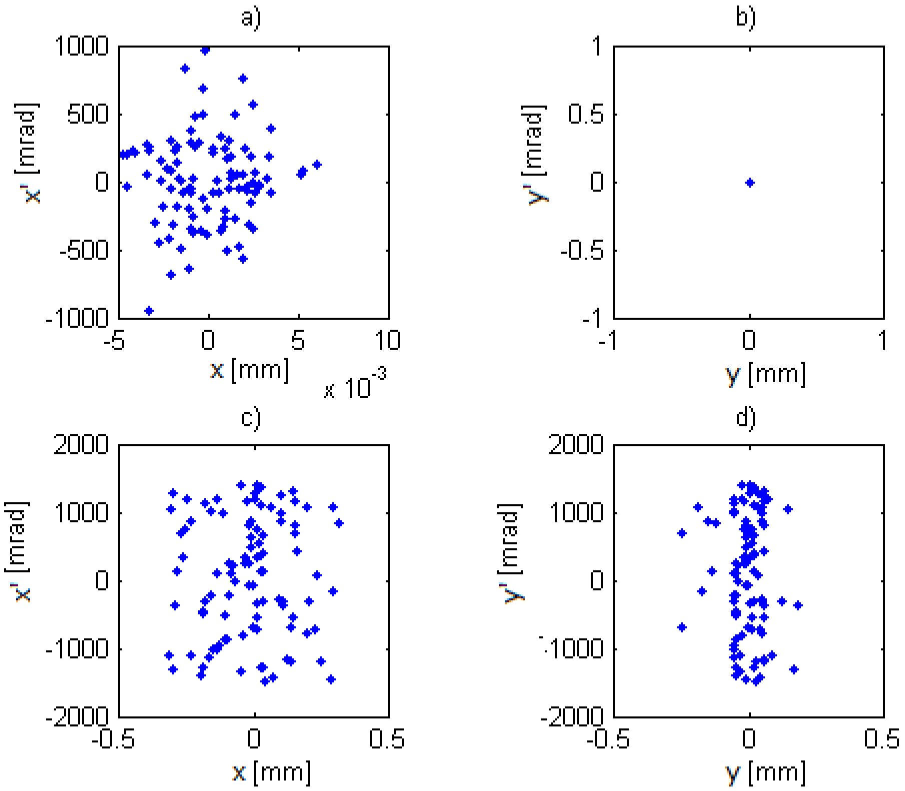


Figure 4.2: Plot emittances output; two planes in phase space x-x' and y-y' are plotted, notation is in agreement with [42]; a) emittance in plane x-x' at the beginning of the Matlab simulation, $xrms = 0.002$ mm, $xprms = 309.019$ mrad; b) emittance in plane y-y' at the beginning of the Matlab simulation is set to zero because of the reason discussed in text; c) emittance in plane x-x' at the end of the Matlab simulation in the time $t\_max0 = 5.1160 \cdot 10^{-8}$ s, $eps\_y = 0.153$ mm, $eps\_theta = 883.009$ mrad; d) emittance in plane y-y' at the end of simulation – particles do not remain in one plane.

Plotted particles trajectories can be found in Fig. 4.3 with detailed view of focusing process in solenoid in Fig. 4.4. In Fig. 4.3 a), b), d) also negative part of x-axis is plotted because some of the particles are reflected back from the solenoid. For our simulated case it is namely 3 particles from 100, i.e. 3% of the beam. Other particles are passing through the solenoid (which is located from 0.001 m to 0.25 m) and they are periodically focused by mechanisms roughly explained in 1.4.2 (see Fig. 1.5) and more detailed in Ref. [42]. The periodical focusing mechanism in the solenoid is demonstrated on more detailed views in Fig. 4.4.

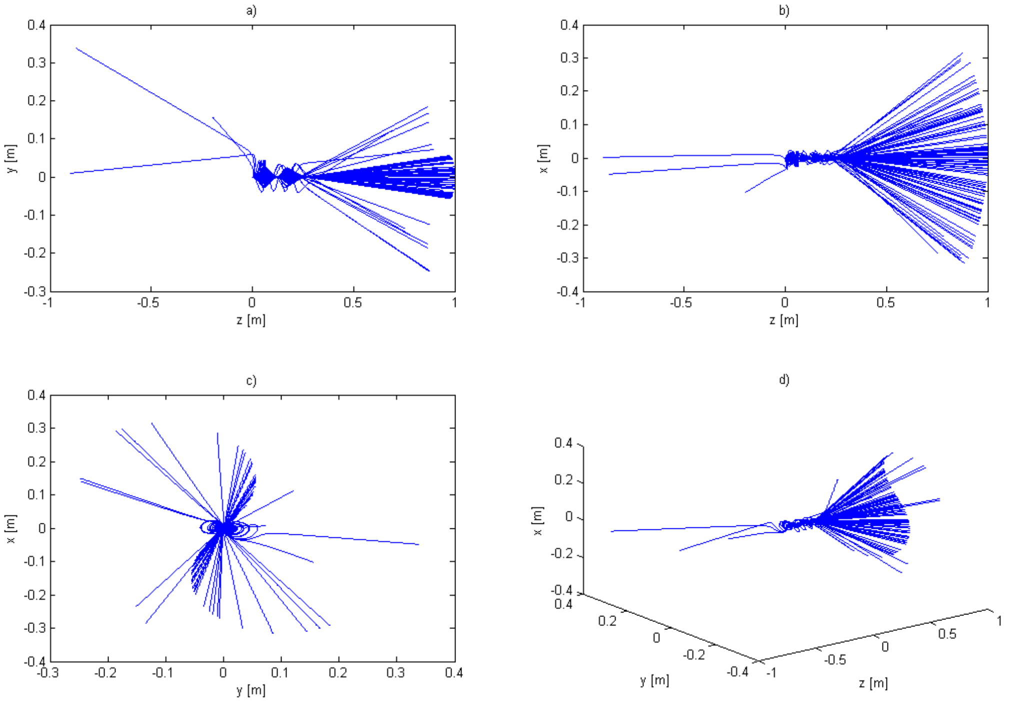


Figure 4.3: a) z-y plane, b) z-x plane, c) cross section and projection of beam trajectories to the y-x plane, d) simple 3D graph; solenoid is located from 0.001 m to 0.25 m, $\beta = 1.5 \cdot 10^{-6}$ m/rad, $\gamma = 3 \cdot 10^{4}$ rad/m, $B = 10.72$ T (magnetic field in the solenoid centre; see Main code in the section 5.4), $eps\_y = 0.153$ mm, $eps\_theta = 883.009$ mrad

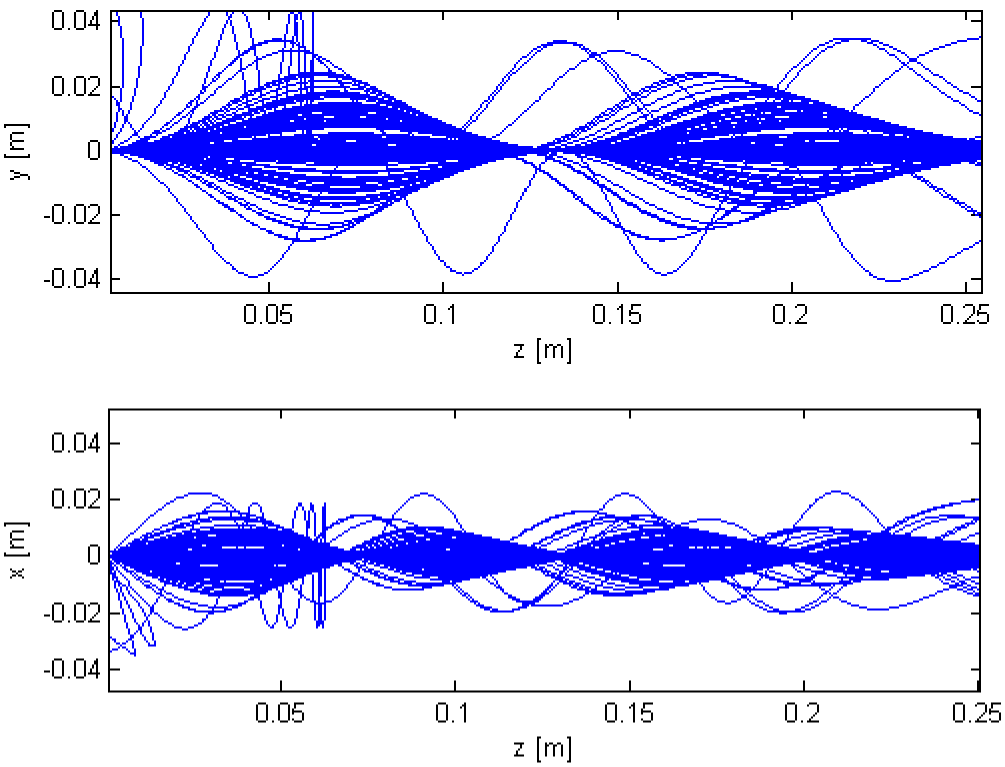


Figure 4.4: Detailed view on focusing mechanism inside the solenoid as explained in chapter 1.4.2 and demonstrated in Fig. 1.5; solenoid is located from 0.001 m to 0.25 m, $\beta = 1.5 \cdot 10^{-6}$ m/rad, $\gamma = 3 \cdot 10^{4}$ rad/m, $B = 10.72$ T (magnetic field in the solenoid centre; see Main code in the section 5.4), $eps\_y = 0.153$ mm, $eps\_theta = 884.7715$ mrad.

Summary and Conclusion

Apart from conventional acceleration techniques there is also a possibility to produce particle beams by ion acceleration based on high intensity laser interaction with matter. Due to their parameters (e.g. energy, divergence, ...) they can be used for various applications including fast ignition, proton radiography, development of compact facilities for laser-driven ion beam radiotherapy, hadrontherapy (a healing of radiation-resistant tumors), nuclear research and so on [29].

The main problem the laser accelerators have to face is producing of divergent proton beams with high energy spread, which caused a big decrease in the number of particles delivered for therapy or other applications. Thus, the main goal is to avoid this divergence which is exactly the topic on which this work is focused on. Actually, the theoretical background of the interaction of ultra intense laser pulses with solid ionized targets are discussed in this text. In addition, a brief description of basic ion-accelerating mechanisms and obtained experimental results focusing on decreasing divergence of accelerated particle beams are discussed. There are many ways how to avoid divergence, for example by various magnetic beam devices [13], [28], [29], using ultra-intense short pulse regimes [14] or by applying different targets (thin foils [3], curved foils [8], foils with microstructures on their rear side [25] etc.), see chapter 1.4.

In the second chapter, possibilities of numerical modeling in laser plasma physics are reported, especially the particle-in-cell method (PIC), whose basic algorithms are explained in very detailed way. For our calculations open souce Warwick's EPOCH PIC code was chosen mainly because of its user-friendly input parameters platform and great users manual.

Particle-in-cell simulations for two types of targets (flat foil and curved foil sections) were performed. Outputs are series of commented figures (energy spectrum, angular distribution, energy balance...) and tables summarising the results of done simulations. The paper contains computing and visualising scripts as same as the code of modified Matlab program. This code can be used for calculating the movements of accelerated particles in the magnetic field of the solenoid as well as for evaluating its emittances. Even if the connection between Matlab program created under the bachelor degree project with PIC simulation were problematic, mainly because of different used dimension (3D for Matlab and 2D for EPOCH), the possibility to link together these two approaches were demonstrated. In summary, the output parameters from PIC simulations were used for calculations of initial values of Twiss parameters and emittance, which were employed as an input to Matlab program. The resulting particle trajectories and particle emittances are demonstrated on the attached figures. According to our simulations particle beams have decreasing divergence with increasing energy; this result is with agreement with previous theoretical and experimental research [15]. Less divergent protons were obtained from curved foil. In short, the protons are accelerated perpendicularly to the target from its rear side which is exactly the reason why more collimated protons are observed in the case of curved foil and not in the case of flat foil section. Higher maximum proton energy (as well as maximum energy of carbon ions) was obtained from curved foil section simulation because of bigger absorption coefficient caused for example by Brunel vacuum heating. In other words, more energy from incident laser pulse is transformed into kinetic energy of particles. This work is important for the implementation and optimization of future experiments at the ELI-Beamlines and also experiments carried out in the Institute of GIST in Gwangju, Korea. Future plans are focused on the use of a solenoid to optimize accelerated ion-beam and of more realistic parameters in the simulations as same as on studying further possibilities of decreasing beam divergence such as structures on the target rear side or using of the second target focusing a proton beam.

Bibliography

[1] Metacentrum wiki, 2013. online, accessed 05-04-2014.

[2] Keith Bennett. *Users Manual for the EPOCH PIC codes.* University of Warwick, 2014.

[3] JH Bin, WJ Ma, K Allinger, HY Wang, D Kiefer, S Reinhardt, P Hilz, K Khrennikov, S Karsch, XQ Yan, et al. On the small divergence of laser-driven ion beams from nanometer thick foils. *Physics of Plasmas (1994-present)*, 20(7):073113, 2013.

[4] Charles K Birdsall and Dieter Fuss. Clouds-in-clouds, clouds-in-cells physics for many-body plasma simulation. *Journal of Computational Physics*, 3(4):494–511, 1969.

[5] Charles Kennedy Birdsall and Allan Bruce Langdon. *Plasma Physics Via Computer Simulation.* CRC Press, 2005.

[6] Christopher Brady. The university of warwick: Centre for scientific computing - portfolio, 2012. online, accessed 03-04-2014.

[7] CESNET. Metacentrum, 2012. online, accessed 05-04-2014.

[8] SN Chen, E d'Humières, E Lefebvre, L Romagnani, T Toncian, P Antici, P Audebert, E Brambrink, CA Cecchetti, T Kudyakov, et al. Focusing dynamics of high-energy density, laser-driven ion beams. *Physical review letters*, 108(5):055001, 2012.

[9] Quanli Dong and Jie Zhang. Vacuum heating of solid target irradiated by femtosecond laser pulses. *Science in China Series G: Physics, Mechanics and Astronomy*, 46(1):71–77, 2003.

[10] T Zh Esirkepov. Exact charge conservation scheme for particle-in-cell simulation with an arbitrary form-factor. *Computer Physics Communications*, 135(2):144–153, 2001.

[11] T Zh Esirkepov, SV Bulanov, K Nishihara, T Tajima, F Pegoraro, VS Khoroshkov, K Mima, H Daido, Y Kato, Y Kitagawa, et al. Proposed double-layer target for the generation of high-quality laser-accelerated ion beams. *Physical review letters*, 89(17):175003, 2002.

[12] K Flippo, BM Hegelich, BJ Albright, L Yin, DC Gautier, S Letzring, M Schollmeier, J Schreiber, R Schulze, and JC Fernandez. Laser-driven ion accelerators: Spectral control, monoenergetic ions and new acceleration mechanisms. *Laser and Particle Beams*, 25(01):3–8, 2007.

[13] Asian Regional Team for ILC Accelerator Development. A batch of quadrupole magnets built for atf2, 2006. online, accessed 12-04-2014.

[14] JS Green, NP Dover, M Borghesi, CM Brenner, FH Cameron, DC Carroll, PS Foster, P Gallegos, G Gregori, P McKenna, et al. Enhanced proton beam collimation in the ultra-intense short pulse regime. *Plasma Physics and Controlled Fusion*, 56(8):084001, 2014.

[15] YJ Gu, Z Zhu, XF Li, Q Yu, S Huang, F Zhang, Q Kong, and S Kawata. Stable long range proton acceleration driven by intense laser pulse with underdense plasmas. *Physics of Plasmas (1994-present)*, 21(6):063104, 2014.

[16] Björn Manuel Hegelich. *Acceleration of heavy ions to MeV/nucleon energies by ultrahigh-intensity lasers*. PhD thesis, lmu, 2002.

[17] BM Hegelich, D Jung, BJ Albright, M Cheung, B Dromey, DC Gautier, C Hamilton, S Letzring, R Munchhausen, S Palaniyappan, et al. 160 mev laser-accelerated protons from ch2 nano-targets for proton cancer therapy. *arXiv preprint arXiv:1310.8650*, 2013.

[18] O. Klimo J. Pšikal. Základy fyziky laserového plazmatu [online]. `http://pi.fjfi.cvut.cz/html/joomla/index.php/en/vyuka/zaklady-fyziky-laseroveho-plazmatu`, 2013. Accessed: 2014-05-01.

[19] S Kar, K Markey, PT Simpson, C Bellei, JS Green, SR Nagel, S Kneip, DC Carroll, B Dromey, L Willingale, et al. Dynamic control of laser-produced proton beams. *Physical review letters*, 100(10):105004, 2008.

[20] O Klimo, J Psikal, J Limpouch, and VT Tikhonchuk. Monoenergetic ion beams from ultrathin foils irradiated by ultrahigh-contrast circularly polarized laser pulses. *Physical Review Special Topics-Accelerators and Beams*, 11(3):031301, 2008.

[21] Petr Kulhánek. *Úvod do teorie plazmatu*. AGA, 2011.

[22] Vinit Kumar. Understanding the focusing of charged particle beams in a solenoid magnetic field. *American Journal of Physics*, 77(8):737–741, 2009.

[23] Roland Lichters, Robert EW Pfund, and Jürgen Meyer-ter Vehn. Lpic++. a parallel one-dimensional relativistic electromagnetic particle-in-cell code for simulating laser-plasma-interaction. Technical report, Max-Planck-Institut fuer Quantenoptik, Garching (Germany), 1997.

[24] Jiri Limpouch. Transparencies of lectures principles of plasma physics, 2012. online, accessed 01-09-2014.

[25] Jirí Limpouch, Jan Pšikal, Ondřej Klimo, Jiří Vyskoččil, Jan Proška, Filip Novotnỳ, Lucie Štolcová, and Milan Květoň. Laser ion acceleration: from present to intensities achievable at eli-beamlines. In *SPIE Optics+ Optoelectronics*, pages 878027–878027. International Society for Optics and Photonics, 2013.

[26] Andrea Macchi, Marco Borghesi, and Matteo Passoni. Ion acceleration by superintense laser-plasma interaction. *Reviews of Modern Physics*, 85(2):751, 2013.

[27] Peter Mulser and Dieter Bauer. *High Power Laser-Matter Interaction*, volume 238. Springer, 2010.

[28] M Nishiuchi, I Daito, M Ikegami, H Daido, M Mori, S Orimo, K Ogura, A Sagisaka, A Yogo, AS Pirozhkov, et al. Focusing and spectral enhancement of a repetition-rated, laser-driven, divergent multi-mev proton beam using permanent quadrupole magnets. *Applied Physics Letters*, 94(6):061107, 2009.

[29] M Nishiuchi, AS Pirozhkov, H Sakaki, K Ogura, T Zh Esirkepov, T Tanimoto, M Kanasaki, A Yogo, T Hori, A Sagisaka, et al. Quasi-monochromatic pencil beam of laser-driven protons generated using a conical cavity target holder. *Physics of Plasmas (1994-present)*, 19(3):030706, 2012.

[30] DT Offermann, KA Flippo, J Cobble, MJ Schmitt, SA Gaillard, T Bartal, DV Rose, DR Welch, M Geissel, and M Schollmeier. Characterization and focusing of light ion beams generated by ultra-intensely irradiated thin foils at the kilojoule scale a). *Physics of Plasmas (1994-present)*, 18(5):056713, 2011.

[31] J Psikal, VT Tikhonchuk, J Limpouch, AA Andreev, and AV Brantov. Ion acceleration by femtosecond laser pulses in small multispecies targets. *Physics of Plasmas (1994-present)*, 15(5):053102, 2008.

[32] Jan Pšikal. *Ion Acceleration in Small-size Targets by Ultra-intense Short Laser Pulses (Simulation and Theory)*. PhD thesis, Czech Technical University, 2009.

[33] M Roth, E Brambrink, P Audebert, M Basko, A Blazevic, R Clarke, J Cobble, TE Cowan, J Fernandez, J Fuchs, et al. Laser accelerated ions in icf research prospects and experiments. *Plasma physics and controlled fusion*, 47(12B):B841, 2005.

[34] H Schwoerer, S Pfotenhauer, O Jäckel, K-U Amthor, B Liesfeld, W Ziegler, R Sauerbrey, KWD Ledingham, and T Esirkepov. Laser-plasma acceleration of quasi-monoenergetic protons from microstructured targets. *Nature*, 439(7075):445–448, 2006.

[35] Jongmin Lee Tae Moon Jeong. Femtosecond petawatt laser. *Ann. Phys. (Berlin)*, 16, 2014.

[36] Toma Toncian, Marco Borghesi, Julien Fuchs, Emmanuel d'Humières, Patrizio Antici, Patrick Audebert, Erik Brambrink, Carlo Alberto Cecchetti, Ariane Pipahl, Lorenzo Romagnani, et al. Ultrafast laser-driven microlens to focus and energy-select mega-electron volt protons. *Science*, 312(5772):410–413, 2006.

[37] David Tskhakaya. The particle-in-cell method. In *Computational Many-Particle Physics*, pages 161–189. Springer, 2008.

[38] John Villasenor and Oscar Buneman. Rigorous charge conservation for local electromagnetic field solvers. *Computer Physics Communications*, 69(2):306–316, 1992.

[39] Jiří Vyskočil. 3d pic kód pro simulace interakce krátkých laserových pulsou s terčitřírozměrné jevy. 2012.

[40] Kane S Yee. Numerical solution of initial boundary value problems involving maxwell's equations. *IEEE Trans. Antennas Propag*, 14(3):302–307, 1966.

[41] Jinqing Yu, Xiaolin Jin, Weimin Zhou, Bin Li, and Yuqiu Gu. High-order interpolation algorithms for charge conservation in particle-in-cell simulations. 2013.

[42] Martina Žáková. Manipulace a transport iontových paprskou řízených laserem. 2013.

[43] Xiaomei Zhang, Baifei Shen, Xuemei Li, Zhangying Jin, Fengchao Wang, and Meng Wen. Efficient gev ion generation by ultraintense circularly polarized laser pulse. *Physics of Plasmas (1994-present)*, 14(12):123108, 2007.

Chapter 5

Appendix

5.1 Matlab script for drawing Energy spectrum

Listing 5.1: Energy spectrum

```
clear all;
l=GetDataSDF('0020.sdf');
px=l.Particles.Px.proton.data;
py=l.Particles.Py.proton.data;
mass_p=1836;
mass_c=22032;
mass_e=1;
px1=px./(9.11e-31*mass_p*3.0e+8);        // p/(mc)
py1=py./(9.11e-31*mass_p*3.0e+8);
i=find(abs(px1)>0.0);
px2=px1(i);
py2=py1(i);
gamma=sqrt(1.0+px2.^2+py2.^2);
energy=0.511*mass_p*(gamma-1.0);
estep=0.02;
ei=0.0:estep:5.0;
h=hist(energy,ei);
// hist(data,nbins) sorts data into the number of bins specified by the scalar nbins.
%figure(1);
hold on;
semilogy(ei,h,'g-','LineWidth',2);
hold on;
%plot(ei,h,'r-','LineWidth',2);
gg=gca;
grid on;
set(gg,'GridLineStyle','--');
set(gg,'MinorGridLineStyle','--');
set(gg,'Fontsize',14);
xlabel('energy (MeV)','Fontsize',14,'Fontweight','bold');
ylabel('N (a.u.)','Fontsize',14,'Fontweight','bold');
maxenergy=max(energy)
%fileID=fopen('momentum.txt','w');
%A=[px2;py2];
%fprintf(fileID,'%6.1e %E \r\n',A);
%fclose(fileID);
```

5.2 Matlab script for drawing Angular distributions and for calculation of Emittance and Twiss parameters

Listing 5.2: Angular distribution + Emittance + Twiss parameters

```
clear all;
l=GetDataSDF('0020.sdf');          // load the last file of the simulation
px0=l.Particles.Px.proton.data;
py0=l.Particles.Py.proton.data;
x0=l.Particles.Px.proton.grid.x;
y0=l.Particles.Px.proton.grid.y;
px=px0./(9.11e-31*1836*3.0e+8);              // p/(mc)
py=py0./(9.11e-31*1836*3.0e+8);
x=x0;
y=y0;
x1=-100.0;                                // setting the simulation box
x2=100.0;
y1=-25.0;
y2=25.0;

%%%%%%%% flat foil protons %%%%%%%%
energymin=0.0;
energymax=0.6899;

%energymin=0.6899;
%energymax=1.3798;

%energymin=1.37980;
%energymax=2.0697;

%energymin=2.0697;
%energymax=2.7596;

%energymin=2.7596;
%energymax=3.4495;

energymin=0.5;
energymax=3.4495;

%%%%%%%% curved foil protons %%%%%%%%

%energymin=0.0;
%energymax=0.80942;

%energymin=0.80942;
%energymax=1.61884;

%energymin=1.61884;
%energymax=2.42826;

%energymin=2.42826;
%energymax=3.23768;

%energymin=3.23768;
%energymax=4.0471;

```

```
%energymin=0.5;
%energymax=4.0471;

j=find((y>y1)&(y<y2)&(x>x1)&(x<x2));          // index for the simulation area
xx1=x(j);
yy1=y(j);
px1=px(j);
py1=py(j);
i=find(px1>0.0);                               // take only x+ axis
px2=px1(i);
py2=py1(i);
yy2=yy1(i);
gamma=sqrt(1.0+px2.^2+py2.^2);                      // relativistic factor gamma
energy=0.511*1836*(gamma-1.0);
index=find((energy>energymin)&(energy<energymax));                   // index for energy interval between (Emin, Emax)
px3=px2(index);
py3=py2(index);
yy3=yy2(index);
gamma3=gamma(index);
vx=px3./gamma3;
vy=py3./gamma3;
alfa=atan(vy./vx);
index2=find(abs(alfa)<50.0/180.0*pi);               // angular cut of particles comimng from the sides of the target
yy4=yy3(index2);
alfa2=alfa(index2);
alfai=-pi:pi/100:pi;
halfa=hist(alfa2,alfai);                        //histogram
hold on;
plot(alfai,halfa,'g-','LineWidth',2);
hold on;
gg=gca;
grid on;
set(gg,'GridLineStyle','--');
set(gg,'MinorGridLineStyle','--');
set(gg,'Fontsize',14);
xlabel('','Fontsize',14,'Fontweight','bold');
ylabel('','Fontsize',14,'Fontweight','bold');
fileID=fopen('theta.txt','w');                      // save all thetas into a file
A=alfa2;
fprintf(fileID,'%6.1e \n',A);
fclose(fileID);
Theta_mean=mean(A);                           // mean value of theta
e=zeros(size(A,1),1);
e=(A(:,1)-Theta_mean).^2;
ee=sum(e);                                     // sigma_x'
eps_theta=sqrt(ee/N)
fileID=fopen('y.txt','w');
B=yy4;
fprintf(fileID,'%6.1e \n',B);
fclose(fileID);
N=size(B,1);
y_mean=mean(B);
i=zeros(size(B,1),1);
i=(B(:,1)-y_mean).^2;                          // sigma_x
ii=sum(i);
```

```
eps_y=sqrt(ii/N)
eps=4*eps_y*eps_theta                  //RMS emittance
twiss_beta=eps_y^2/eps                         // Twiss parameter Beta
twiss_gamma=eps_theta^2/eps                       // Twiss parameter Gamma
```

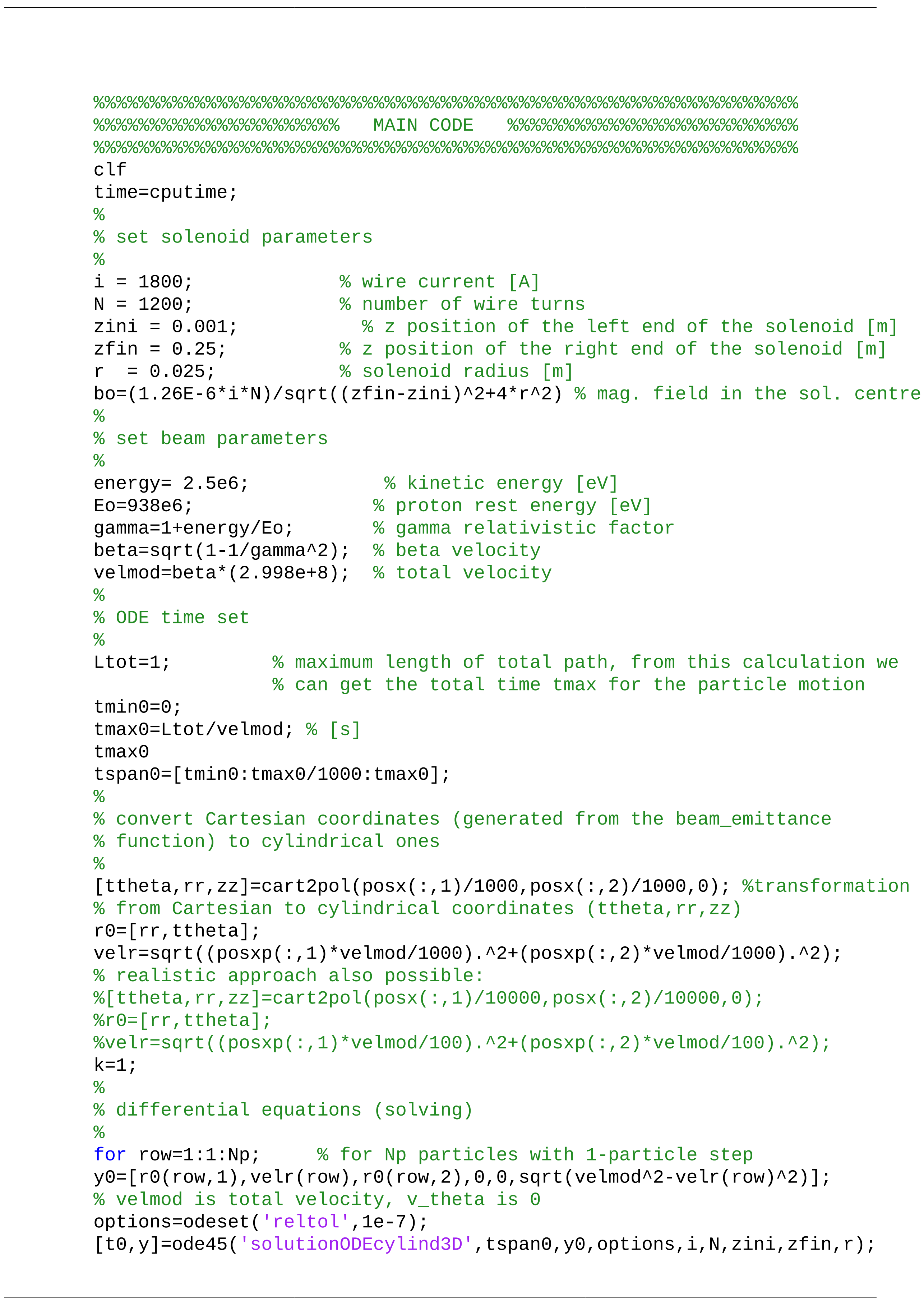
5.3 Matlab script for drawing Energy balance

Listing 5.3: Energy balance

```
=-1.6021766e-19;
m_e=9.10938e-31;
c=299792458;
lambda=800;
omega=2*pi*c/(lambda*1e-9);
mass_particle_p=1836;
mass_particle_e=1;
mass_particle_c=22032;

numberSDF = 21; // how many .sdf files in folder + 1

time=zeros(numberSDF,1);
W_kin_p=zeros(numberSDF,1);
W_kin_e=zeros(numberSDF,1);
W_kin_c=zeros(numberSDF,1);
W_kin=zeros(numberSDF,1);
W_field=zeros(numberSDF,1);
W_total=zeros(numberSDF,1);

for j=0:(numberSDF-1) // starting from zero
  currentSDF=sprintf('000%d.sdf',j);
   if j>9
     currentSDF=sprintf('00%d.sdf',j);
   end
l=GetDataSDF(currentSDF);

Ex=l.Electric_Field.Ex.data;
Ey=l.Electric_Field.Ey.data;
Ez=l.Electric_Field.Ez.data;
Bx=l.Magnetic_Field.Bx.data;
By=l.Magnetic_Field.By.data;
Bz=l.Magnetic_Field.Bz.data;
%dimensionless
Ex2=Ex.*e./(m_e*omega*c);
Ey2=Ey.*e./(m_e*omega*c);
Ez2=Ez.*e./(m_e*omega*c);
Bx2=Bx.*e./(m_e*omega);
By2=By.*e./(m_e*omega);
Bz2=Bz.*e./(m_e*omega);

    n=150/20;                    // number of particles on critical density
    px_p=l.Particles.Px.proton.data;
    px_e=l.Particles.Px.electron.data;
    px_c=l.Particles.Px.carbon.data;
    py_p=l.Particles.Py.proton.data;
    py_e=l.Particles.Py.electron.data;
```

```
        py_c=l.Particles.Py.carbon.data;
        pz_p=l.Particles.Pz.proton.data;
        pz_e=l.Particles.Pz.electron.data;
        pz_c=l.Particles.Pz.carbon.data;
        px1_p=px_p./(m_e*mass_particle_p*c);
        px1_e=px_e./(m_e*mass_particle_e*c);
        px1_c=px_c./(m_e*mass_particle_c*c);
        py1_p=py_p./(m_e*mass_particle_p*c);
        py1_e=py_e./(m_e*mass_particle_e*c);
        py1_c=py_c./(m_e*mass_particle_c*c);
        pz1_p=pz_p./(m_e*mass_particle_p*c);
        pz1_e=pz_e./(m_e*mass_particle_e*c);
        pz1_c=pz_c./(m_e*mass_particle_c*c);
        gamma_p=sqrt(1.0+px1_p.^2+py1_p.^2+pz1_p.^2);
        gamma_e=sqrt(1.0+px1_e.^2+py1_e.^2+pz1_e.^2);
        gamma_c=sqrt(1.0+px1_c.^2+py1_c.^2+pz1_c.^2);

      W_kin_p(j+1)=sum(mass_particle_p/n*(gamma_p-1))
        W_kin_e(j+1)=sum(mass_particle_e/n*(gamma_e-1))
        W_kin_c(j+1)=sum(mass_particle_c/n*(gamma_c-1))
        W_kin(j+1)=W_kin_p(j+1)+W_kin_e(j+1)+W_kin_c(j+1)

W_field=0.5*sum(sum((Ex2.^2+Ey2.^2+Ez2.^2)))+0.5*sum(sum((Bx2.^2+By2.^2+Bz2.^2)))   //2D matrix (sum twice)
W_total=W_kin+W_field

time(1,j+1)=l.time*1e15;       //in [fs]
end

kappa=max(W_kin)/max(W_field)
plot(time(1,:),W_field,'r-','LineWidth',2);
hold on
plot(time(1,:),W_total,'b-','LineWidth',2);
hold on
plot(time(1,:),W_kin,'g-','LineWidth',2);
hold on
plot(time(1,:),W_kin_p,'k-','LineWidth',2);
hold on
plot(time(1,:),W_kin_e,'c-','LineWidth',2);
hold on
plot(time(1,:),W_kin_c,'m-','LineWidth',2);
hold on
```

5.4 Modified Matlab program from bachelor thesis



```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%   MAIN CODE   %%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clf
time=cputime;
%
% set solenoid parameters
%
i = 1800;             % wire current [A]
N = 1200;             % number of wire turns
zini = 0.001;           % z position of the left end of the solenoid [m]
zfin = 0.25;          % z position of the right end of the solenoid [m]
r  = 0.025;           % solenoid radius [m]
bo=(1.26E-6*i*N)/sqrt((zfin-zini)^2+4*r^2) % mag. field in the sol. centre
%
% set beam parameters
%
energy= 2.5e6;            % kinetic energy [eV]
Eo=938e6;                % proton rest energy [eV]
gamma=1+energy/Eo;       % gamma relativistic factor
beta=sqrt(1-1/gamma^2);  % beta velocity
velmod=beta*(2.998e+8);  % total velocity
%
% ODE time set
%
Ltot=1;         % maximum length of total path, from this calculation we
                % can get the total time tmax for the particle motion
tmin0=0;
tmax0=Ltot/velmod; % [s]
tmax0
tspan0=[tmin0:tmax0/1000:tmax0];
%
% convert Cartesian coordinates (generated from the beam_emittance
% function) to cylindrical ones
%
[ttheta,rr,zz]=cart2pol(posx(:,1)/1000,posx(:,2)/1000,0); %transformation
% from Cartesian to cylindrical coordinates (ttheta,rr,zz)
r0=[rr,ttheta];
velr=sqrt((posxp(:,1)*velmod/1000).^2+(posxp(:,2)*velmod/1000).^2);
% realistic approach also possible:
%[ttheta,rr,zz]=cart2pol(posx(:,1)/10000,posx(:,2)/10000,0);
%r0=[rr,ttheta];
%velr=sqrt((posxp(:,1)*velmod/100).^2+(posxp(:,2)*velmod/100).^2);
k=1;
%
% differential equations (solving)
%
for row=1:1:Np;     % for Np particles with 1-particle step
y0=[r0(row,1),velr(row),r0(row,2),0,0,sqrt(velmod^2-velr(row)^2)];
% velmod is total velocity, v_theta is 0
options=odeset('reltol',1e-7);
[t0,y]=ode45('solutionODEcylind3D',tspan0,y0,options,i,N,zini,zfin,r);
```

```
cylcoord(:,:,k)=y;  % saving information form ODEs for each k-th particle
k=k+1;

end
%
%  convert cylindrical coordinates to Cartesian ones because of plotting
%
for kk=1:1:(k-1)
[xx,yy,zz]=pol2cart(cylcoord(:,3,kk),cylcoord(:,1,kk),cylcoord(:,5,kk));
end
for kk=1:1:(k-1)
[xx(:,kk),yy(:,kk),zz(:,kk)]=pol2cart(cylcoord(:,3,kk),cylcoord(:,1,kk),
                                                        cylcoord(:,5,kk));
end
for kk=1:1:(k-1)
coord(:,:,kk)=[xx(:,kk),yy(:,kk),zz(:,kk)]; % saving in coord
end

for kk=1:1:(k-1)
[vxx,vyy,vzz]=pol2cart(cylcoord(:,4,kk),cylcoord(:,2,kk),cylcoord(:,6,kk));
end
for kk=1:1:(k-1)
[vxx(:,kk),vyy(:,kk),vzz(:,kk)]=pol2cart(cylcoord(:,4,kk),cylcoord(:,2,kk),
                                                         cylcoord(:,6,kk));
end
for kk=1:1:(k-1)
vcoord(:,:,kk)=[vxx(:,kk),vyy(:,kk),vzz(:,kk)]; % saving in vcoord
end
%
%  plot trajectories
%
%
for k=1:1:Np % for each of Np particles
    hold on
    subplot(2,2,1);
    plot(coord(:,3,k),coord(:,2,k)); %zy plane
    hold off
    hold on
    subplot(2,2,2);
    plot(coord(:,3,k),coord(:,1,k)); %zx plane
    hold off
    hold on
    subplot(2,2,3);
    plot(coord(:,2,k),coord(:,1,k)); %yx plane
    hold off
    hold on
    subplot(2,2,4);
    plot3(coord(:,3,k),coord(:,2,k),coord(:,1,k)); %simple 3D
    hold off

end
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%   SOLUTION OF CYLINDRICAL ODEs   %%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [dydt]=f(t,y,nothing,i,N,zini,zfin,r)
amu=1.66053886e-27;            % atomic mass unit [kg]
charge = 1.60217653e-19;       % elementary charge [C]
Ap=1.00727647;                 % mass number proton (= mass of proton
                               % divided by 1u (atomic mass unit)
Q = 1;                         % charge state
mu_0=1.26E-6;                  % permeability constant [H/m]
L = (zfin - zini);             % length of solenoid [m]
z2 = y(5) - zini;
z1 = y(5) - zfin;
bz=(mu_0*i*N)/(2*L)*(z2/sqrt(z2*z2+r*r)-z1/sqrt(z1*z1+r*r)); %[T]
btheta=0;
br=-0.5*y(1)*(mu_0*i*N)/(2*L)*(1/sqrt(z2*z2+r*r)-1/sqrt(z1*z1+r*r)-
        -z2*z2/(z2*z2+r*r)^(1.5)+z1*z1/(z1*z1+r*r)^(1.5));
ratio_QA=Q/Ap*charge/amu;
   dydt=[y(2)
        ratio_QA*y(1)*y(4)*bz+y(1)*y(4)^2                  % focusing
      y(4)
        (ratio_QA*(y(6)*br-y(2)*bz)-2*y(2)*y(4))/y(1)      % rotation
      y(6)
        -ratio_QA*y(1)*y(4)*br];                           % acceleration
```

Published with MATLAB® R2013b

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%  BEAM EMITTANCE  %%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear all;
time=cputime;
% twiss parameters
gamma=6e4               %typical value from PIC simulations
beta=8e-7               %tzpical value from PIC simulations
alpha=sqrt(beta*gamma-1)

%
eps0=10/pi;                % RMS emittance [mm*mrad]
Np=100;                    % particles number
xrms=sqrt(eps0*beta)       % [mm]
xprms=sqrt(eps0*gamma)     % [mrad]
count=0;                   % a counter
posx=zeros(Np,2);          % allocate memory array for x and y plane;
                           % Np-by-2 array of zeros
                           % a storage for coordinates
posxp=zeros(Np,2);         % allocate memory array for x' and y' plane
                           % a storage for divergences
emittance=zeros(Np,2);     % allocate memory array
while count<Np
    x=normrnd(0,xrms,1,2);   % Gaussian distribution with sigma=xrms(x,y),
                             % mu=0; generates an 1-by-2-by random array
    xp=normrnd(0,xprms,1,2); % Gaussian distribution with sigma=xprms(x',y')
    % eps= gamma*x.^2+2*alpha*x.*xp+beta*xp.^2; % calculate the emittance
                                              % considering twiss par.

    %if eps<9*eps0 & sqrt(x(1,1)^2+x(1,2)^2)<3*xrms;
        count=count+1;
        posx(count,:)=x; % here coordinates are stored,
                         % posx(count,1) and posx(count,2)
        posx(count,2)=0.0;
        posxp(count,:)=xp; % here divergences are stored
        posxp(count,2)=0.0;
        %emittance(count,:)=eps;
    %end % from this we have x, x', y, y', x=(x,y), xp=(x',y') and emittance
end
posxp
plot(posx(:,1),posxp(:,1),'.'); axis equal; %x-x' plane
e_time=cputime-time
```

Published with MATLAB® R2013b

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%   PLOTTING EMITTANCES   %%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
subplot(2,2,1);
plot(posx(:,1),posxp(:,1),'.');  %emittance x-x' at the beginning;
axis square;
subplot(2,2,2);
plot(posx(:,2),posxp(:,2),'.');  %emittance y-y' (set as zero
                                 % because of 3D,2D problem)
axis square;
s_coord=size(coord);
s_vcoord=size(vcoord);
t_coord=s_coord(1);
t_vcoord=s_vcoord(1);
index=find(coord(t_coord,3,:)>zfin);  %only particles which went through
                                      % solenoid
x_coord=coord(t_coord,1,:);
y_coord=coord(t_coord,2,:);
z_coord=coord(t_coord,3,:);
vx_vcoord=zeros(Np,1);
vx_vcoord=vcoord(t_vcoord,1,:);
vy_vcoord=vcoord(t_vcoord,2,:);
vz_vcoord=vcoord(t_vcoord,3,:);
x_coord1=x_coord(index);
y_coord1=y_coord(index);
z_coord1=z_coord(index);
vx_vcoord1=vx_vcoord(index);
vy_vcoord1=vy_vcoord(index);
vz_vcoord1=vz_vcoord(index);
angle=atan(vy_vcoord1(1,1,:)./vx_vcoord1(1,1,:))*1000;    %in [mrad]
posx1=zeros(length(index),1);
pospx1=zeros(length(index),1);
for i1=1:length(index)
posx1(i1)=x_coord1(1,1,i1);
posxp1(i1)=angle(1,1,i1);
end
posy1=zeros(length(index),1);
pospy1=zeros(length(index),1);
for i1=1:length(index)
posy1(i1)=y_coord1(1,1,i1);
posyp1(i1)=angle(1,1,i1);
end
mean_y=mean(posx1);
mean_theta=mean(posxp1);
sigma_y=(posx1-mean_y).^2;
sigma_theta=(posxp1-mean_theta).^2;
eps_yy=sum(sigma_y);
eps_theta2=sum(sigma_theta);
eps_y=sqrt(eps_yy/length(sigma_y))       % eps_y same as from PIC data
eps_theta=sqrt(eps_theta2/length(sigma_theta))  %eps_theta -//-
subplot(2,2,3);
plot(posx1,posxp1,'.');  %emittance x-x' at the end;
```

```
axis square;
subplot(2,2,4);
plot(posy1,posyp1,'.');  %emittance y-y' at the end;
axis square;
```

Published with MATLAB® R2013b