

Multi-dimensional numerical investigation of laser-induced particle acceleration

Bachelor Thesis

Sven Ahrens

Technical University Darmstadt
2006

Abstract

The interaction of an ultra-intense laser pulse ($I \sim 3 \cdot 10^{19} \frac{W}{m^2}$) with a thin overdense plasma is investigated numerically. A multi-dimensional fully relativistic Particle-in-Cell-Code (PIC) has been used for the simulation of this laser plasma interaction. Careful data postprocessing gives insight in the evolution of the complex acceleration process. As expected, a strong electrostatic field builds up at the rear surface of the target with a transverse extension much larger than the laser focal spot. A nicely laminar proton acceleration follows inside this field. The observed proton energies as well as their momentum distribution are in agreement with experimental findings.

Contents

1	Introduction	3
2	About the simulation code PSC	3
3	How the PSC package works	4
3.1	General function	4
3.1.1	The Vlasov-Boltzmann Equation	4
3.1.2	Discretization	4
3.1.3	Time evolution of the particles or particle motion	5
3.1.4	Maxwell's Equations	6
3.2	Operating the PSC	9
3.3	Output data processing	9
4	The examined scenario	11
4.1	Experimental parameters	11
5	Evaluation of the simulation data	12
5.1	Ion acceleration	13
6	Improvements	22
7	Acknowledgements	22

1 Introduction

Collimated proton beams with an energy of about $5 - 6 MeV$ per particle were observed by irradiating a thin foil with an ultra-intense laser beam [1]. The protons originate from water vapor and hydrocarbons, which are usually adsorbed on the surface of the foil. During the laser plasma interaction, part of the free electrons is accelerated to high energies, which propagate through the target foil and build an electron plasma at the nonirradiated rear side of the target. The electron plasma induces a strong electric field ($> 10^{12} \frac{V}{m}$), which ionizes the water molecules and hydrocarbons and accelerates the protons normally to the target rear surface over a distance of several μm up to energies of some MeV . This mechanism is called target normal sheath acceleration (TNSA). Preferably the protons are accelerated in comparison to other ion species because of their larger charge-to-mass ratio $\frac{e}{m}$. The described scheme of proton (ion) acceleration is a new possibility to produce short and intense bunches of high-energy particles with a high directionality [2]. It allows also focussing of the ion beam, when targets with a shaped (curved) rear surface are used [3].

To get high-quality ion beams for applications as isochoric heating [4], radiography [5] or fast ignition in inertial confinement fusion (ICF) [6], the physics of the very complex acceleration process must be understood in detail. Especially, the transport of hot electrons through the foil and the problem of the return current need further investigations. Numerically observed instabilities in this transport would show up in a filamentation of the ion beam and destroy its shape [7].

This work is devoted to sophisticated multi-dimensional kinetic simulations for a typical proton acceleration scenario with a short-pulse high-power laser driver.

2 About the simulation code PSC

For the investigation of quasi-particle motion in a strong laser field, I've used the PSC (Plasma Simulation Code), which was written by Hartmut Ruhl and Andreas Kemp [8]. The PSC is a relativistic particle-in-cell code, which assumes atoms, ions and electrons as classical particles. The particles are treated as "quasi-particles", which have masses and charges artificially increased in a manner that this would not affect the simulated physics but limit the amount of calculation.

3 How the PSC package works

3.1 General function

3.1.1 The Vlasov-Boltzmann Equation

In the Plasma Simulation Code, the ansatz for the equations of motion is based on the Vlasov equation [9]-[10]:

$$\frac{df}{dt} = \frac{\partial f}{\partial t} + \dot{\vec{x}} \frac{\partial f}{\partial \vec{x}} + \dot{\vec{p}} \frac{\partial f}{\partial \vec{p}} = 0, \quad (1)$$

where $f = f(\vec{x}, \vec{p}, t)$ is the time-dependent particle density in phase space. The Vlasov equation has its origin from Liouville's theorem, which states that the volume occupied by a given number of particles in phase space remains constant in time (which is implied by $\frac{df}{dt} = 0$) [11]. If there are binary collisions between the particles, the constant volume of these particles in phase space is changed with the collision integral $St(f)$:

$$\frac{df}{dt} = \frac{\partial f}{\partial t} + \dot{\vec{x}} \frac{\partial f}{\partial \vec{x}} + \dot{\vec{p}} \frac{\partial f}{\partial \vec{p}} = St(f). \quad (2)$$

But the binary collisions can be handled in a separate process in one time step such that $\frac{df}{dt} = 0$ remains.

Additionally, Eqn. (2) gets normalized, where the substitutions $t \rightarrow \frac{\tilde{t}}{\omega}$, $\vec{x} \rightarrow \lambda_D \tilde{\vec{x}}$, $\vec{p} \rightarrow mc \tilde{\vec{p}}$, $\vec{E} \rightarrow E_0 \tilde{\vec{E}}$, $\vec{B} \rightarrow B_0 \tilde{\vec{B}}$, $\vec{j} \rightarrow j_0 \tilde{\vec{j}}$ and $\vec{\rho} \rightarrow \rho_0 \tilde{\vec{\rho}}$, with the plasma wavelength $\lambda_D = \frac{c}{\omega}$, the speed of light c and the laser frequency ω , are made. Together with the electromagnetic force $\dot{\vec{p}} = q[\tilde{\vec{E}} + \vec{v} \times \tilde{\vec{B}}]$, this leads to

$$\frac{\partial f}{\partial t} + \vec{v} \frac{\partial f}{\partial \vec{x}} + \frac{qM\eta}{mQ} [\tilde{\vec{E}} + \vec{v} \times \tilde{\vec{B}}] \frac{\partial f}{\partial \vec{p}} = 0, \quad (3)$$

where Q is the electric charge and M is the physical mass. The other global factors are explained in chapter 4.1 of [8] and in subsection 4.1. Here and further on, the tilde from notations with dimensionless units will be omitted. Notice that there are different types of particles (electrons and ions), which have different charges q and masses m (commonly multiples of Q and M). We will not distinguish between them in this work, but one should have in mind that there isn't only one sort of particles.

3.1.2 Discretization

For a numerical solution of the equations of motion and the Maxwell equations, the electric and magnetic fields and the charge and current distributions are implemented as three-dimensional arrays ($E \rightarrow E_{jkl} \in \mathbb{R}^n \times \mathbb{R}^n \times \mathbb{R}^o$), where each array index $j, k, l \in \mathbb{N}$ represents a spacial grid node at the position $\vec{x} =$

$(j\Delta x, k\Delta y, l\Delta z)$. The density of a particle at the position (x', y', z') is given by

$$S(x', y', z') = \begin{cases} (1 - |\frac{x-x'}{\Delta x}|)(1 - |\frac{y-y'}{\Delta y}|)(1 - |\frac{z-z'}{\Delta z}|), \\ \text{if } |x - x'| \leq \Delta x, |y - y'| \leq \Delta y, |z - z'| \leq \Delta z \\ 0, \quad \text{otherwise} \end{cases} \quad (4)$$

This shape leads to a quadratic spline (QS), which has second order accuracy and is also called "Particle in Cell" (PIC). The particle weight at the grid points i in one dimension e.g. can be obtained by integrating the formfactor S over the corresponding grid:

$$w_i = \int_{(i-\frac{1}{2})\Delta x}^{(i+\frac{1}{2})\Delta x} S(x)dx = \begin{cases} 0, & |x_j - x_i| > \frac{3\Delta x}{2} \\ \frac{1}{2} \left(\frac{3}{2} - \frac{|x_j - x_i|}{\Delta x} \right)^2, & \frac{\Delta x}{2} < |x_j - x_i| \leq \frac{3\Delta x}{2} \\ \frac{3}{4} - \frac{|x_j - x_i|^2}{\Delta x^2}, & |x_j - x_i| \leq \frac{\Delta x}{2} \end{cases} \quad (5)$$

Multiplying the one-dimensional weights w_i, w_l, w_r , we get the three-dimensional weight of the particle $w_{ilr} = w_i w_l w_r$, which is proportional to the charge density.

3.1.3 Time evolution of the particles or particle motion

After multiplying Eqn. (3) and integrating it, one obtains the relativistic equations of motion (for details see [8]), which are modified by some constants after normalization:

$$\dot{\vec{p}} = \vec{F}(\vec{x}, t) = \frac{qM\eta}{mQ} [\vec{E} + \vec{v} \times \vec{B}], \quad (6)$$

$$\dot{\vec{x}} = \vec{v}, \quad \vec{p} = \gamma \vec{v}, \quad \gamma = \sqrt{1 + \vec{p}^2} = \frac{1}{\sqrt{1 - \vec{v}^2}}. \quad (7)$$

For the action of the Lorentz- and electrostatic forces on a particle, a field interpolation is made, as it was mentioned in Section 3.1.2 (see also Figure 2). The discretized version of Equation (6) reads

$$\vec{p}^{n+1} = \vec{p}^n + \frac{qM\eta}{mQ} \Delta t [\vec{E}^{n+\frac{1}{2}} + \vec{v}^{n+\frac{1}{2}} \times \vec{B}^{n+\frac{1}{2}}]. \quad (8)$$

Here, the upper index represents time $f^n \hat{=} f(t_n)$, $t_{n+1} - t_n = \Delta t$. Substituting with

$$\vec{p}^{n+\frac{1}{2} \pm \frac{1}{2}} = \vec{p}^\pm \pm \frac{qM\eta}{2mQ} \Delta t \vec{E}^{n+\frac{1}{2}}, \quad (9)$$

we get the relation

$$\vec{p}^- - \vec{p}^+ = -\frac{qM\eta}{mQ} \Delta t \vec{v}^{n+\frac{1}{2}} \times \vec{B}^{n+\frac{1}{2}}, \quad (10)$$

which must be solved for \vec{p}^+ . The solution can be interpreted as a rotation [12]. For details see also [8].

Now, the motion of the particles is realized in a way that the particles are moved first during half a time step as

$$\vec{x}^{n+\frac{1}{2}} = \vec{x}^n + \vec{v}^n \Delta t/2, \quad (11)$$

after what the new momenta can be calculated in general according to Eqn. (8). We shall use the triplet form of Equation (10). Finally, the second half of the movement of the particles is made with the new velocities \vec{v}^{n+1} :

$$\vec{x}^{n+1} = \vec{x}^n + \vec{v}^{n+1} \Delta t/2 \quad (12)$$

The described numerical procedure is involved in subroutine `PIC_move_part.f` of the code.

3.1.4 Maxwell's Equations

In addition to the particle motion, there is the electromagnetic field, which is also propagating in time. The time evolution of the E-M Field is described by the Maxwell equations (here in normalized units):

$$\nabla \vec{E} = \rho, \quad \nabla \vec{B} = 0, \quad (13)$$

$$\frac{\partial \vec{B}}{\partial t} = -\nabla \times \vec{E}, \quad \frac{\partial \vec{E}}{\partial t} = \nabla \times \vec{B} - \vec{j}, \quad (14)$$

together with the law of charge conservation:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \vec{j} = 0. \quad (15)$$

The discretized form of Eqs. (13) - (15), which are relevant for the plasma simulation, is:

$$\left[\vec{E}_{jkl}^{n+\frac{1}{2}} - \vec{E}_{jkl}^{n-\frac{1}{2}} \right] / \Delta t = \nabla^- \times \vec{B}_{jkl}^n - \vec{j}_{jkl}^n, \quad (16)$$

$$\left[\vec{B}_{jkl}^{n+1} - \vec{B}_{jkl}^n \right] / \Delta t = -\nabla^+ \times \vec{E}_{jkl}^{n+\frac{1}{2}}, \quad (17)$$

where

$$\nabla^- F_{jkl}^n = \begin{pmatrix} F_{jkl}^n - F_{j-1kl}^n / \Delta x, t \\ F_{jkl}^n - F_{jk-1l}^n / \Delta y, t \\ F_{jkl}^n - F_{jkl-1}^n / \Delta z, t \end{pmatrix}, \quad \nabla^+ F_{jkl}^n = \begin{pmatrix} F_{j+1kl}^n - F_{jkl}^n / \Delta x, t \\ F_{jk+1l}^n - F_{jkl}^n / \Delta y, t \\ F_{jkl+1}^n - F_{jkl}^n / \Delta z, t \end{pmatrix}. \quad (18)$$

The lower indices jkl represent the three spacial dimensions.

Both formulas 16 and 17 are applied two times for a half time step respectively (see Figure 2). Between the first half time step evolution of the field (which is made in `PIC_msa.f`) and the second half time step (which is made in `PIC_msb.f`), the subroutine `PIC_move_part.f` of chapter 3.1.3 will be executed.

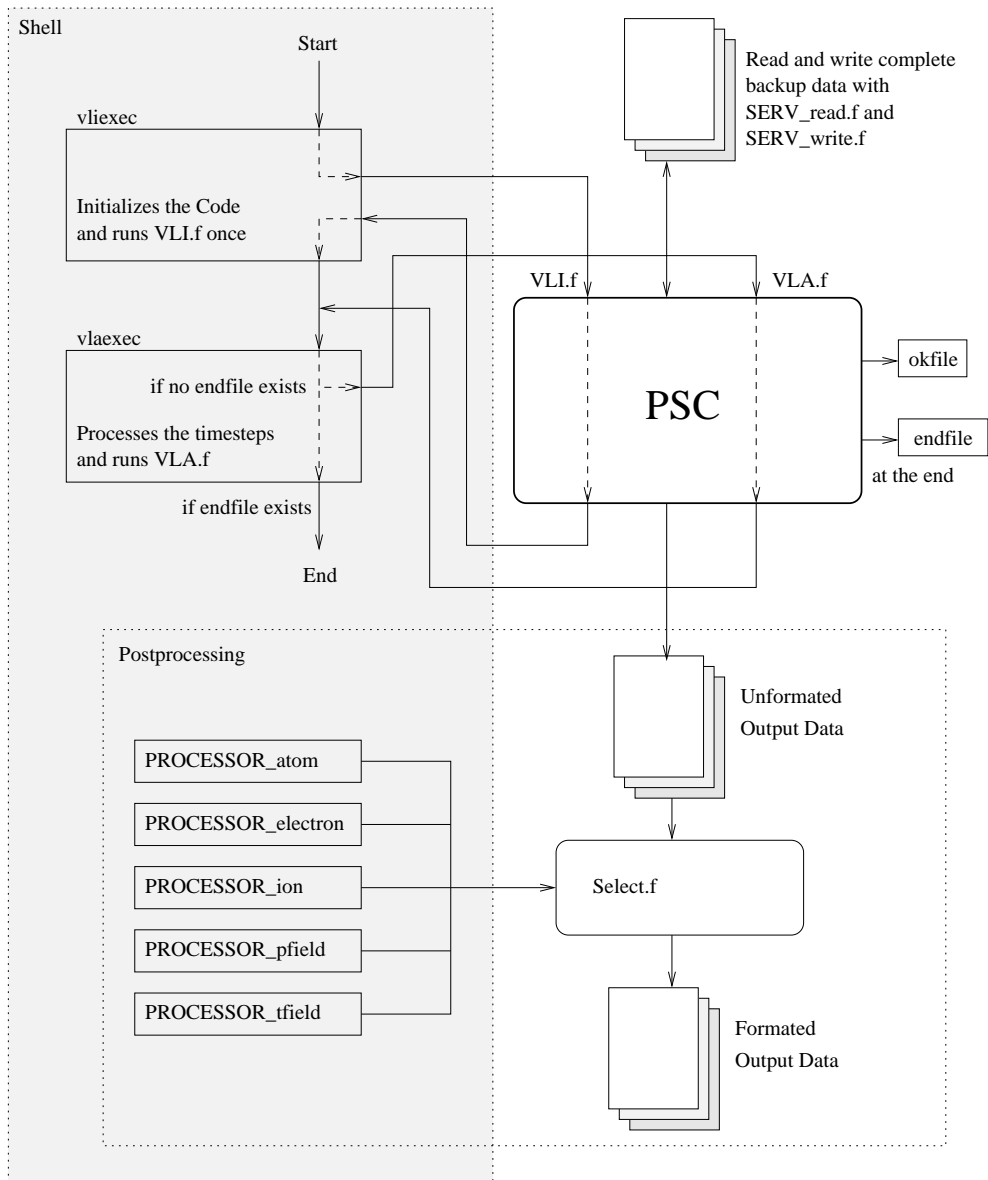


Figure 1: Program call

The user starts the raw calculating process with vliexec. After the program has been finished, the user may operate the data to create images after post-processing with PROCESSOR_* .

PSC

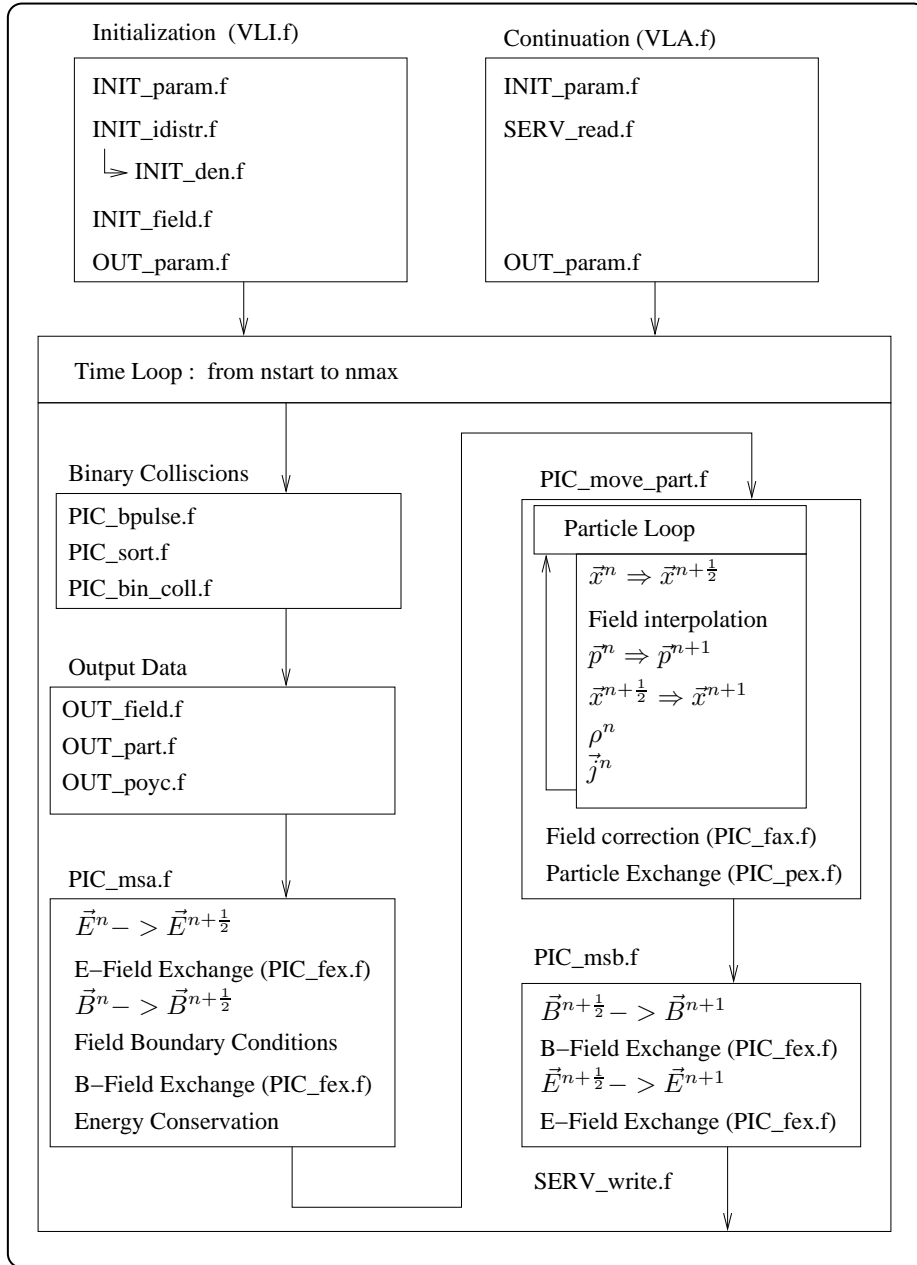


Figure 2: Program flow

3.2 Operating the PSC

For running the PSC, one should have a well-performed computer platform, which may have multiple calculating nodes (multi core processors in multiple workstations in a network), because the PSC is designed for distributed calculations. The following software should be installed:

- an operating system, which is most often Linux
- a batch system, which is able to operate on multiple hosts, for example OpenPBS.
- a fortran compiler for compiling the code, which is written in Fortran90
- The MPI (Message Passing Interface), which serves the distributed computing of multiple processor applications with the Message Passing technique.

Before the calculation can be started, the Fortran90 source code has to be compiled. This can comfortably be done with the shellscript `Makefile`, which automatically recognizes a change of a file in the source package since the last compilation.

Now, the PSC can be started with the call of `vliexec` from the shell. As it is shown in Figure 1, `vliexec` executes the routine `VLI.f`, which can, together with `VLA.f`, be regarded as the main program of PSC. The difference between `VLI.f` and `VLA.f` is, that the former performs the initialization of the simulation, e.g. sets up the particle distributions and determines the right dispatching for the multiple nodes, and the latter just reads the full configuration of the variables from the backup file which has always been exported at the end of the preceding simulation. Another difference is that `VLA.f` is called by `vlaexec`.

Because of their analogous meaning, we will talk about "the program" when thinking about `VLI.f` or `VLA.f`. The calculation will be finished after a running time, which can be set with the `cpum` variable in the code. If this time is reached, the program calls the `SERV_write.f`-subroutine, which performs a writedump of all data. After that, the program creates an `okfile`, which signalsizes that this run of the program has been executed properly. If the time step variable `n` has grown up to `nmax`, the program additionally starts the `endfile`, which is signalizing, that the complete calculation process should be terminated. Notice, that the counter `n` starts working from the `nstart` value.

So, the whole simulation runs from `nstart` to `nmax` timesteps and, after the `cpum` running time was reached, the program stops for backup purposes and can be restarted by `vlaexec`.

During the runtime, the program writes out the field-, particle- and Poynting vector data with the subroutines `OUT_field.f`, `OUT_part.f`, `OUT_poyc.f`. The timing, at which time step the data are stored, is shown in Table 1.

3.3 Output data processing

In addition to the backup, which is read and written by `SERV_read.f` and `SERV_write.f`, the program also produces some general output of the electro-

Output information	starting time step	ending time step	time step increment
field	N/A	nprf	dnprf
particles	N/A	nprparti	dnprparti
time averaged poynting law	tmnvp	tmxvp	np
time averaged field	tmnvf	tmxvf	np

Table 1: Intervals of data output

The program outputs data, if the time step counter `n` reaches the final time step number. The next ending timestep is redetermined as ending timestep + timestep increment.

magnetic field, the particles and the Poynting law, which is intended for later evaluation of the simulation results, while the raw data are saved in binary format. A transformation to formatted numbers for postprocessing will be provided by the routines `PROCESSOR_atoms`, `PROCESSOR_electrons`, `PROCESSOR_ions`, `PROCESSOR_pfield` and `PROCESSOR_tfield`.

After that, one can perform graphics with the ASCII data. The recommended graphical software in PSC is IDL (Interactive Data Language), because the authors of the code have already prepared some graphical evaluation software in IDL. One may use any other programming languages for the graphical evaluation, for example Matlab, Mathematica, Gnuplot or Xmgrace.

All output files, that are postprocessed by `SELECT`, are zipped in the `*.data.gz` format, where the data are saved as numbers (one number per line) in a `*.data` file. First, some initial constants are written, which are assigned in `INIT_param.f` and may also be seen in `VLA.data` or with the call of `SELECT.x -> c1`. Then, the field and particle data follow.

In the field data, the value of each grid point is written, where the index of the x-axis is altering fastest. The index for the y-axis runs slower, and along the z-axis we have the the slowest running index.

Every particle is represented by 11 values, characterizing it in normalized units:

`xi`, `yi`, `zi`: The particle position

`pxi`, `pyi`, `pzi`: The particle momenta

`qni`: The particle charge

`mni`: The particle mass

`cni`: The cell, the particle is in

`lni`: The particle number

`wni`: The particle weight

These 11 values are stored chronologically, particle by particle, in the output data file.

4 The examined scenario

I have examined in my thesis high-intensity laser-induced particle acceleration. In this scenario, an ultra-intense laser beam irradiates a thin (several tens of micrometers thick) gold foil. After an inevitable laser prepulse has created a plasma on the front side of the solid foil, the main laser pulse accelerates electrons from there into the plasma. These electrons fly with nearly the speed of light through the foil and exit its rear side, where a strong electric field is created. This quasistatic field accelerates ions, e.g. protons from field-ionized hydrogen atoms in the outer layer of the rear target surface.

A typical PSC run for such a scenario may last weeks or months in dependence on resolution demands and produce many gigabytes of data. I have analyzed a calculation made recently by Marius Schollmeier. I am very grateful to him for data support.

4.1 Experimental parameters

A complete description of how to set up the parameters of a two-dimensional simulation in `INIT_param.f`, is given in the Chapters 6.1.1 and 6.2.2 of the PSC-Documentation [8].

The 2-dimensional simulation lies in the y-z-plane. The size of the simulation box is $60\mu m \times 60\mu m$, where the mesh has a resolution of 1000×1000 grid points. The target is a foil with a thickness of $10\mu m$ consisting of hydrogen atoms with a density of $9 \cdot 10^{28} \frac{\text{Atoms}}{m^3}$. These atoms are implemented as 15 single charged ($q = 1e$) proton quasi-particles per grid cell¹. Additionally, there are 150 electrons ($q = -1e$) per cell with the same distribution. Since the code obeys the conservation law of charges and this will be used to obtain the current distributions, with which the electric and magnetic fields can be obtained via Maxwell's equations, the higher charge density of the electrons does not lead to a strong static electric field and hence does not affect the simulation in a dramatic sense. The particle distribution is not completely homogeneous at 160 particles per cell, but 160 means the maximum density per cell. One can see the real distribution in Figures 3 and 4. Because the laser comes from the left, the part in front of the $z = 20\mu m$ plane will be called the frontside and the part behind the $z = 35\mu m$ plane the rear of the foil. On the frontside, the foil plasma has a smaller spatial gradient than on the backside because of the laser prepulse, which generates a preplasma. Altogether, the simulation box contains about 27.5 Million quasi-particles. The incoming laser beam has a Gaussian envelope with a peak intensity of $I_0 = 3 \cdot 10^{23} \frac{W}{m^2}$. The FWHM of the initial spot is $3.3\mu m$ and the pulse length (the FWHM in propagation direction) of the laser beam is $49.96\mu m$, which corresponds to a pulse duration of $166,6fs$. The laser wavelength is $\lambda_L = 1.053\mu m$, what corresponds to the laser frequency $\omega = \frac{2\pi c}{\lambda_L} = 1,79 \cdot 10^{15} Hz$ of a neodymium-glass laser. The electric field is oscillating in the y-plane and the maximum field strength, which

¹grid cell means the cube, which is bordered by the neighbouring grid points. So the cube has a volume of $60nm \times 60nm \times 60nm$.

$t=0.00000fs$
 $x=29.9700\mu m$
 $max=9.56069$
 $min=0.00000$

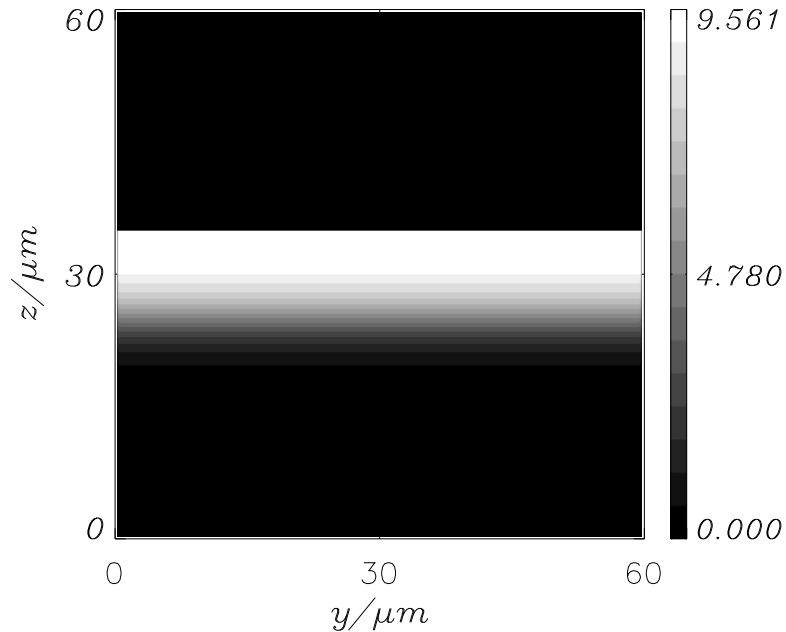


Figure 3: Initial electron particle distribution. The ion particle distribution looks similar.

is also the field normalization factor mentioned in chapter 3.1.1, is equal to $E_0 = \sqrt{\frac{2I_0}{\epsilon_0 c} \frac{\mu m}{\lambda_L}} = 14.27 \frac{TV}{m}$. Equivalently, we find $B_0 = 47,6 \cdot 10^3 T$.

5 Evaluation of the simulation data

While propagating $15\mu m$ through the underdense plasma, the laser beam diverges, but the field strength at the front side is a bit higher because of reflection. This can be seen in Figures 5 and 6.

The strong laser beam accelerates electrons at the front side via the ponderomotive force. These fast electrons propagate through the foil and leave it at the back. The first hot electrons can be observed after $t = 183fs$, as one can see in Figure 7. When the laser pulse reaches half of its peak intensity at the

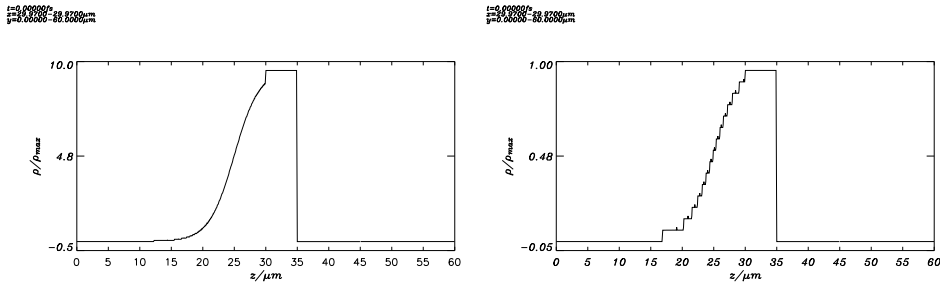


Figure 4: Initial particle density.

On the left, the electron density, on the right the ion particle density are shown. A density of 1 corresponds to 15 quasi-particles per cell. The rugged ion density distribution is caused by the low particle number per cell.

target front at time $t = 150 fs$, the fast electrons have approximately the speed of light. This estimate we conclude from the momentum distribution along the z -axis, which is shown in Figure 8. The investigation of various phase space distributions of the fast electrons (momenta in y or z direction) does not show any regular feature, which means that the fast electrons have a continuous, washy distribution in phase space. One observes about 10000 quasi-electrons, which have left the foil at the rear side after a time of $326 fs$, what corresponds to a charge of $Q = 10^{-10} C$ crossing the target surface at position $z = 35 \mu m$. Because the first occurrence of fast electrons was observed at $183 fs$, one gets an average current of about $I = 700 A$ during a time of approximately $140 fs$. Assuming that this high current initially flew through the foil, this massive charge movement should be compensated by a counter propagating electron flux to prevent the build-up of a strong electric field along the z -axis inside the foil. In fact, one observes an exceeding current density in negative direction at low energies, as it is shown in Figure 9. The current density is determined by counting the number of electrons in equidistant intervals of their momentum. A histogram consisting of 1000 intervals was used.

5.1 Ion acceleration

Outside the foil, we do not observe slow thermal electrons in the beginning of the electric field setup, which could compensate the fast electron current through the rear surface of the target. So, a very strong electric field is established in z -direction there (see Fig. 10). This field has a maximum strength of $E \approx 2 \cdot 10^{12} \frac{V}{m} = 2 \frac{MV}{\mu m}$. This high electric field causes field ionization of the hydrogen atoms. The expelled bound electrons are pushed into the foil and protons are being left. The protons will be accelerated by the strong electric field in positive z -direction, shown in Figure 11. The trace of the momenta is astonishing thin (see Fig. 12).

$t=210.600fs$
 $x=29.9700\mu m$
 $max=1.55454$
 $min=-1.25304$

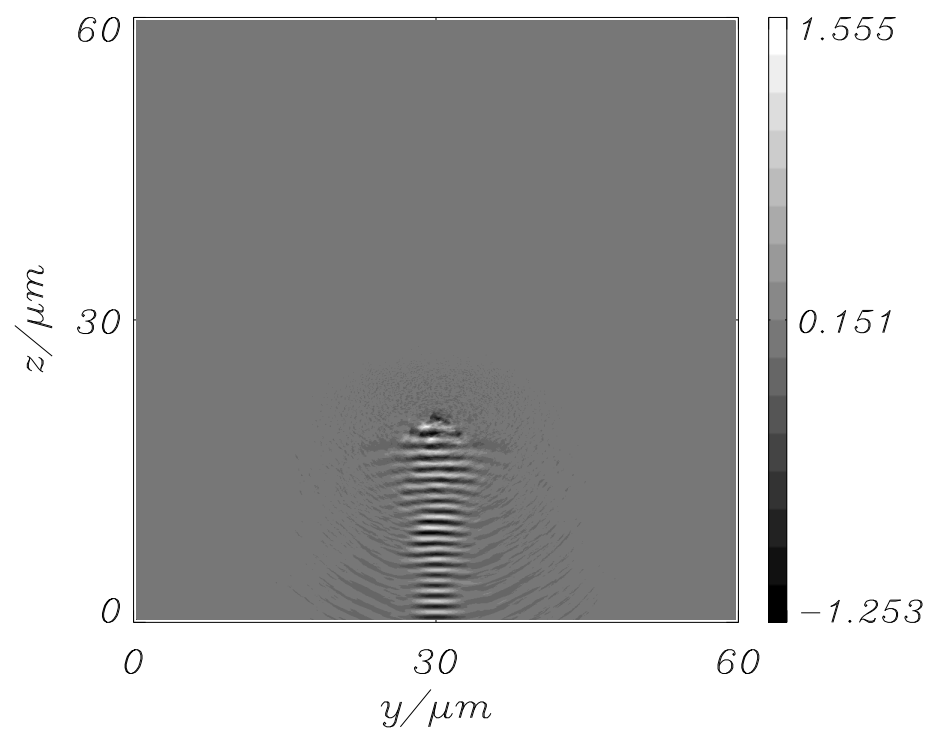


Figure 5: Electric field of the incident laser wave.
Here, the y -component of the electric field is shown at time step 2460, what corresponds to $t = 210.6fs$.

$t=210.600\text{fs}$
 $x=29.9700-29.9700\mu\text{m}$
 $y=0.00000-60.0000\mu\text{m}$

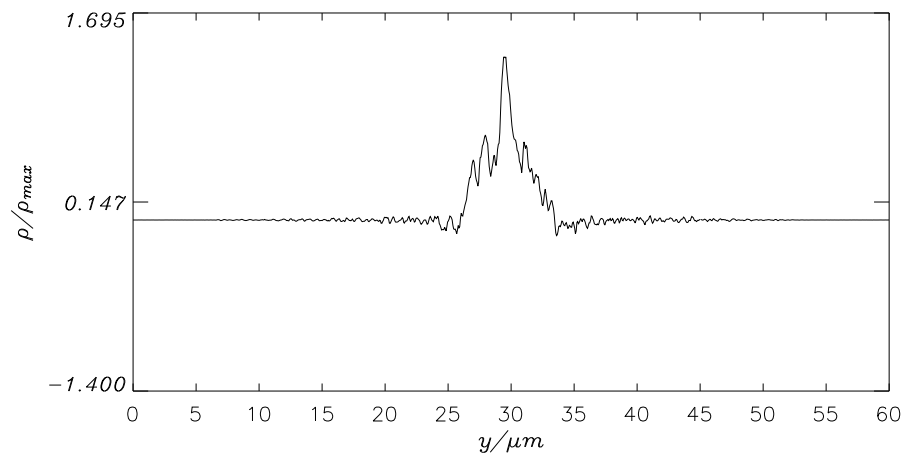


Figure 6: Electric field strength. The y-component of the electric field is plotted over the y-axis in normalized units E_0 at time step 2460. This field distribution is a cut at $z = 18\mu\text{m}$ (300th grid point).

$t=182.520fs$
 $x=29.9700\mu m$
 $max=9.63020$
 $min=0.00000$

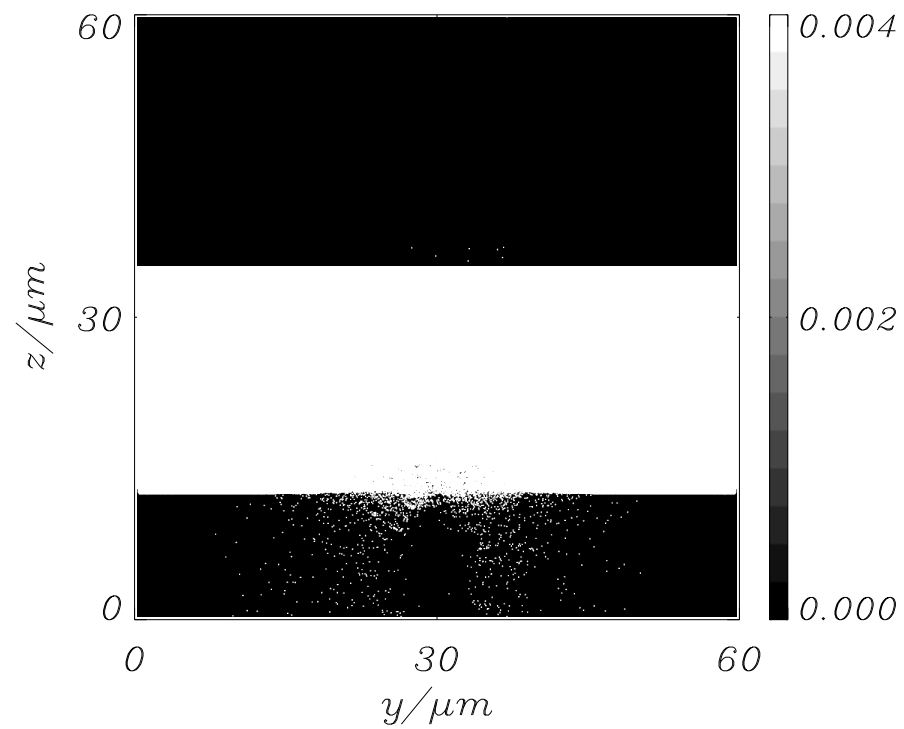


Figure 7: First electrons emerging from the rear side of the target. The electron charge density is plotted in a very small density range. This filtering results in a very high contrast between vacuum and any occurring charge density. The time of $183fs$ corresponds to the time step 2132.

$t=224.640\text{fs}$
 $x=29.9099-30.0300\mu\text{m}$
 $y=-0.0600601-60.0601\mu\text{m}$
 $px=-4.00000-4.00000mc$
 $py=-4.00000-4.00000mc$
 $N=6.58662e+14$

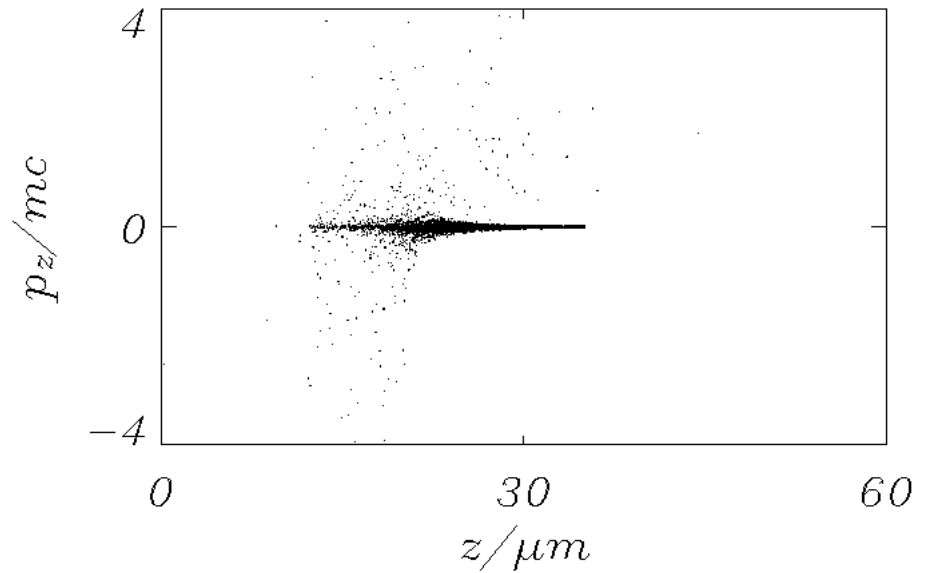


Figure 8: Electron momentum distribution.

In the area of $25\mu\text{m}$ to $35\mu\text{m}$, one observes only electrons with high momenta in positive z -direction, but not in the opposite. Fast electrons with momenta of more than 20 times of the value, corresponding to the rest mass, can be detected. Of course, their number is rather low at these high energies.

$t=224.640fs$
 $x=29.9099-30.0300\mu m$
 $y=-0.0600601-60.0601\mu m$
 $z=-0.0600601-60.0601\mu m$
 $px=-0.100000-0.100000mc$
 $py=-0.100000-0.100000mc$
 $pz=-0.100000-0.100000mc$
 $m=1.00000m_e$
 $N=6.50176e+14$

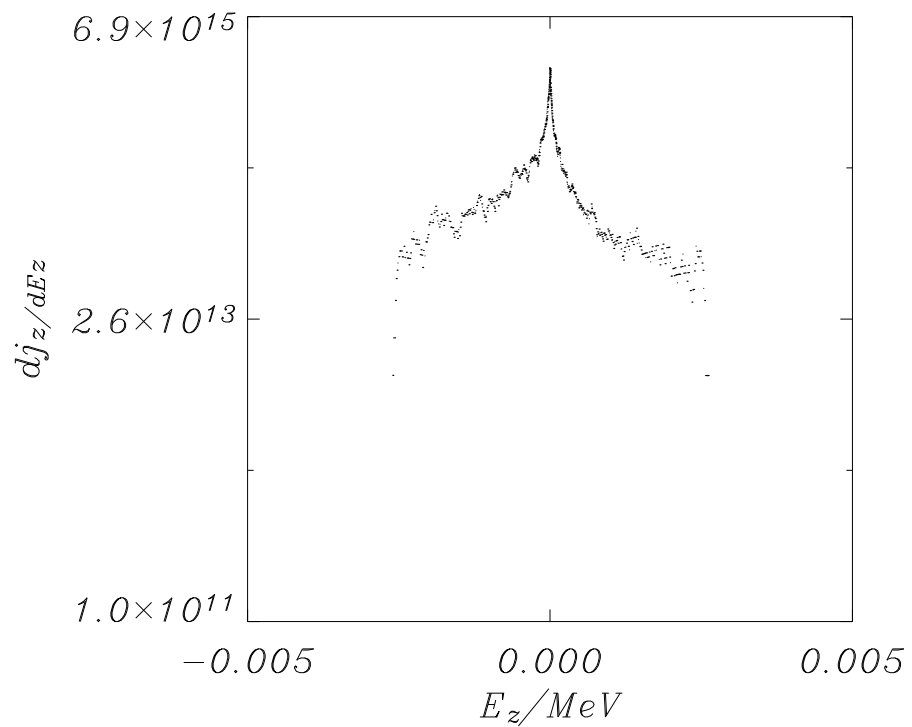


Figure 9: Current density at low electron energies.
 The maximum energy of the particles in z-direction is $E = 5keV$, which is 100 times less than the rest mass of an electron. One can see, that the current produced by electrons flying in the negative z-direction is a bit higher than the electron current in positive z-direction.

$t=336.959\text{fs}$
 $x=29.9700\mu\text{m}$
 $\text{max}=0.437663$
 $\text{min}=-0.573121$

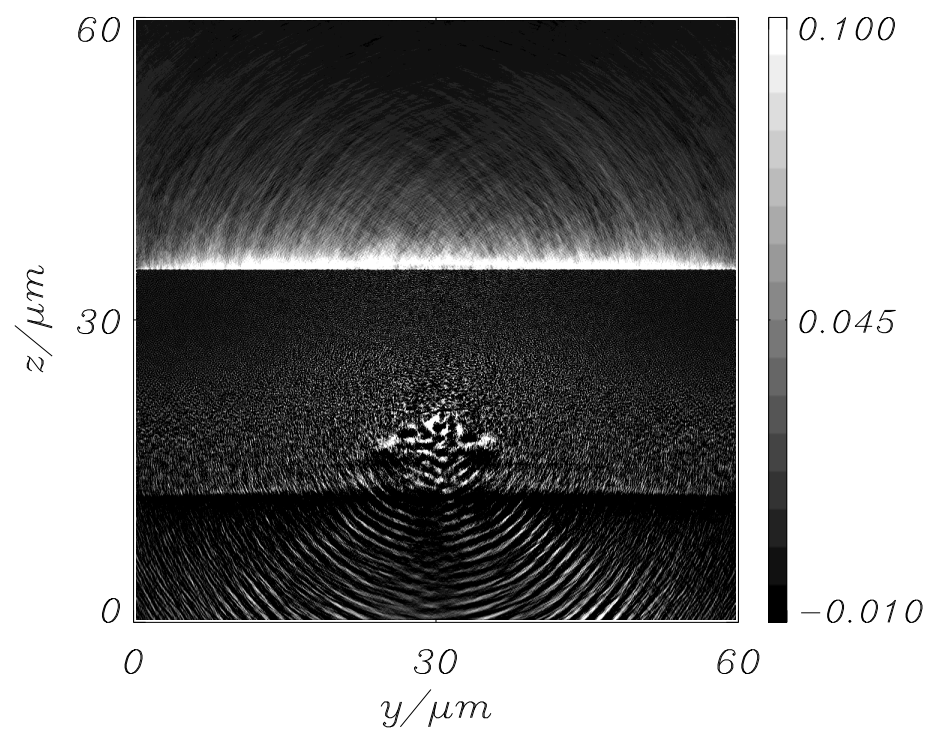


Figure 10: Strong electric field at the rear side of the target. The electric field in z-direction is shown - the white stripe at the position of $z = 35\mu\text{m}$.

$t=505.439\text{fs}$
 $x=29.9700\mu\text{m}$
 $max=2.52931$
 $min=0.00000$

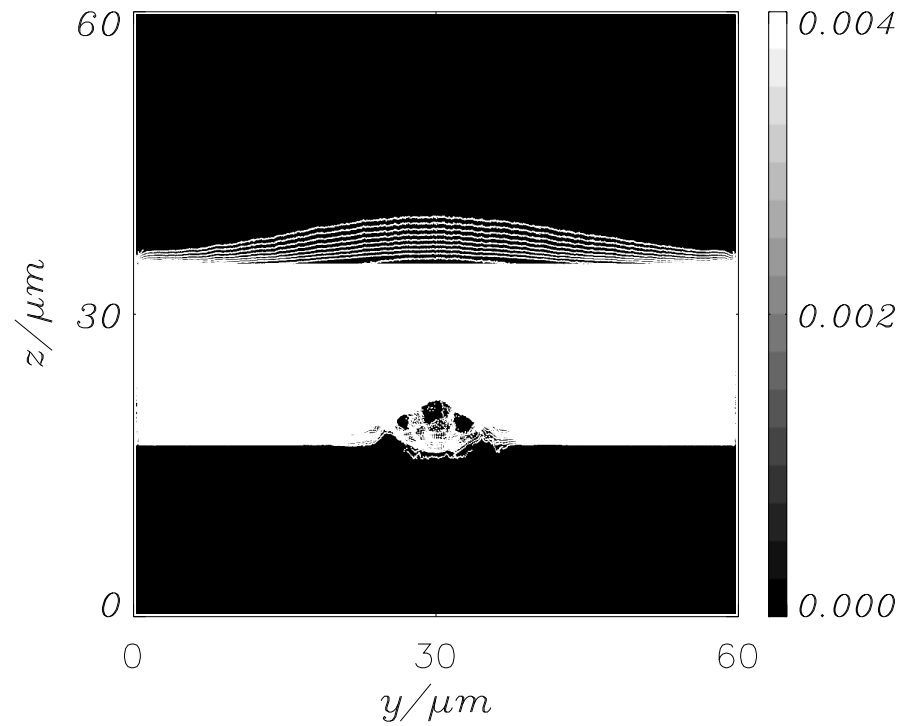


Figure 11: Protons accelerated from the rear side. Like in Figure 7, the ion current density is plotted in a very small range, leading to a high contrast in the image. The stripes are the result of the original ion distribution. The ions are initiated at discrete positions with zero momentum. So, they remain at their location until they are pulled into vacuum by the space charge field.

```

t=505.439fs
x=29.9099-30.0901μm
y=-0.0600601-60.1201μm
px=-0.100000-0.100000mc
py=-0.100000-0.100000mc
N=1.68073e+11

```

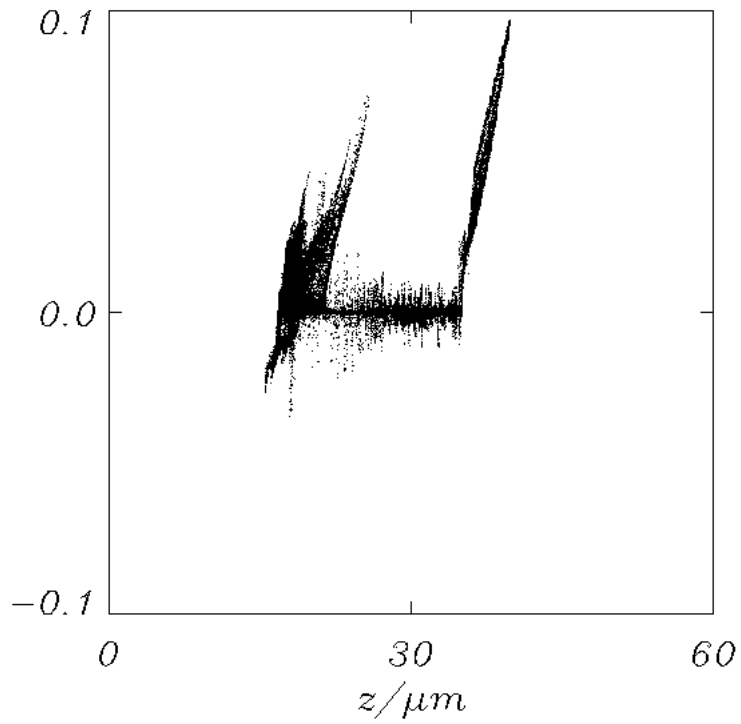


Figure 12: Momentum distribution of the protons. On one hand, ions are accelerated at the front side of the target. Additionally, high-energy ions (protons) come also from the backside of the target. Here, the momentum of the particles is growing linearly with increasing distance from the initial rear target surface in z -direction and reaches 0.1 times of the rest mass of the protons. This value corresponds to a kinetic energy of $5MeV$.

6 Improvements

There are some points, which may be reconsidered in future work to improve the code:

- In the investigated simulation binary collisions were taken into account. An important question is, how much they affect the simulated physical process. The simplest way of understanding the collisions is to run the calculation with the same initial parameters but inhibit collision processes.
- The particle number in the particle data should be initiated in the correct manner. At the moment, it is set to 0, what makes identification of the particles impossible. This would be also a prerequisite to investigate the trajectories of distinguished particles.
- The IDL-Routines should be checked carefully, whether they are working properly. In some of the routines, the implemented energy-momentum-relation is questionable.
- A clever "after postprocessing" of the data should be implemented, because the data files produced by the code are very large, so that the processing takes much time. This makes it difficult to debug or test data evaluation software.

7 Acknowledgements

Special thanks go to Dipl.-Phys.Ing. Marius Schollmeier for providing me with a simulation, which he set up and controlled during the runtime of four weeks. The evaluation of these data is a significant part of my bachelor thesis. Marius gave me a lot of useful tips and answered many of my questions very quickly. Also, I want to acknowledge gratefully Dr. Theodor Schlegel for his far-seeing supervision of my thesis and many substantial explanations concerning laser plasma interaction. His guidance was very tolerant and I have enjoyed the freedom in my research. Many thanks to Prof. Karlheinz Langanke for his official supervision and motivating discussions and Prof. Markus Roth, who introduced me to the field of ultra-intense laser-generated plasmas.

Erklärung zur Bachelor Thesis gemäß §23 Abs. 7 APB:

Hiermit versichere ich, die vorliegende Bachelor Thesis ohne Hilfe Dritter nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die aus den Quellen entnommen wurden, sind als solche kenntlich gemacht worden. Diese Arbeit hat in gleicher Form noch keiner Prüfungsbehörde vorgelegen.

References

- [1] R.A. Snavely et al., *Intense High-Energy Proton Beams from Petawatt-Laser Irradiation of Solids*, Phys. Rev. Lett. **85**, 2945 (2000)
- [2] T.E. Cowan et al., *Ultralow Emittance, Multi-MeV Proton Beams from a Laser Virtual-Cathode Plasma Accelerator*, Phys. Rev. Lett. **92**, 204801 (004)
- [3] R. Sonobe et al. *Suppression of transverse proton beam divergence by controlled electron cloud in laser-plasma interactions*, Phys. Plasmas **12**, 073104 (2005)
- [4] P.K. Patel et al., *Isochoric Heating of Solid-Density Matter with an Ultra-fast Proton Beam*, Phys. Rev. Lett. **91**, 125004 (2003)
- [5] M. Borghesi et al., *Proton Imaging: a diagnostic for inertial confinement fusion/fast ignitor studies*, Plasma Phys. Control. Fusion **43**, A267 (2001)
- [6] M. Roth et al., *Fast Ignition by Laser-Accelerated Proton Beams*, Phys. Rev. Lett. **86**, 436 (2001)
- [7] J. Fuchs et al., *Spatial Uniformity of Laser-Accelerated Ultrahigh-Current MeV Electron Propagation in Metals and Insulators*, Phys. Rev. Lett. **91**, 255002 (2003)
- [8] H. Ruhl, A. Kemp, *Classical Particle Simulations with PSC code (An Introduction into the PSC)*, code documentation (unpublished)
- [9] W.L. Kruer *The physics of laser plasma interactions*, Frontiers in Physics Series Vol. 73, Addison-Wesley, Redwood City (1988)
- [10] O. Boine-Frankenheim, *Introduction to the Physics of High Current Ion Beams in Accelerators and Storage Rings*, Vorlesungsskript (SS01)
- [11] L.D. Lifschitz, L.P. Pitajewski, *Lehrbuch der Theoretischen Physik X: Physikalische Kinetik*, Akademie-Verlag, Berlin (1983)
- [12] C.K. Birdsall, A.B. Langdon, *Plasma Physics via Computer Simulation*, Adam Hilger, Bristol (1991)