

DEVELOPMENT OF
A PLASMA SIMULATION TOOL FOR
RADIO FREQUENCY ION THRUSTERS



DISSERTATION

ZUR ERLANGUNG DES DOKTORGRADES
DER NATURWISSENSCHAFTEN
(DR. RER. NAT.)

VORGELEGT VON

HERRN ROBERT HENRICH

I. PHYSIKALISCHES INSTITUT
JUSTUS-LIEBIG-UNIVERSITÄT GIESSEN

21. November 2013

Contents

1	Introduction	1
2	Theory	3
2.1	Definition and classification of plasmas	3
2.2	Characteristics of a plasma	4
2.3	Plasma sheath	7
2.4	Particle-particle collisions in a low pressure discharge	9
2.5	Low pressure plasma discharges	11
2.5.1	Heating mechanisms	12
2.5.2	DC discharge	13
2.5.3	Inductive discharge	16
2.6	Secondary electron emission - induced electron emission	18
2.7	Neutral gas flow	19
2.8	Micro-Newton radio frequency ion thruster	21
3	“PlasmaPIC” - a 3D plasma simulation program	25
3.1	Motivation for a PIC code	26
3.2	PIC-MCC scheme	27
3.2.1	Integration of the equations of motion	28
3.2.2	Monte Carlo Collision module	29
3.2.3	Particle weighting and force interpolation	34
3.2.4	Constraints for numerical parameters	34
3.3	Derivation of the electromagnetic fields	35
3.3.1	Electrostatic part	36
3.3.2	Electrodynamic part	36
3.4	Solver for elliptic equations	38

3.5	Geometry representation	39
3.6	Supported boundary types	40
4	Massive Parallelization	43
4.1	Comparison of the Parallelization Libraries	45
4.1.1	OpenMP	45
4.1.2	MPI with Domain Decomposition	49
4.1.3	Cuda	53
4.2	Choice of parallelization	57
4.3	Parallelization of the entire plasma simulation	58
5	Principles for the simulation of a micro-Newton RIT	65
5.1	Necessity for 3D plasma simulation	65
5.2	Neutral gas analysis in a micro-Newton RIT	66
5.2.1	Description of the modeling algorithm	67
5.2.2	Simulation results for the neutral gas distribution	68
5.3	Interaction of plasma and neutral gas	70
6	Demonstration of the capabilities of “PlasmaPIC”	73
6.1	Simulation of a micro-Newton RIT 1.0	73
6.1.1	Convergence and long-term stability test	75
6.1.2	Analysis of the simulation results	75
6.1.3	Investigation of a single rf-cycle	82
6.1.4	Variation of the power deposition	83
6.1.5	Influence of neutral gas pressure on the plasma density	85
6.2	Modeling of a DC discharge	90
7	Conclusions	93
	Bibliography	95
	Danksagung	99
	Eidesstattliche Erklärung	101

Chapter 1

Introduction

Low pressure plasmas or technical plasmas are widely used in a number of applications in industry as well as in academia. A main area of application is material processing, where low pressure plasmas are used for coating and etching of surfaces. For example, anti-reflective coating is commonly used for glasses and displays. Etching with low pressure plasmas allows smaller structures because the plasma etching is anisotropic, whereby under-etching is avoided. One possibility of generating the plasma uses an the inductively coupled ion source and a grid extraction system. Due to the grid system the energy as well as the direction of the ions can be controlled with high precision leading to a homogeneous and well controllable deposition.

At the University of Giessen inductively coupled ion sources were adapted for the usage as a spacecraft propulsion system. The so called radio frequency ion thruster (RIT) generates the thrust by ejecting ions with a high velocity of up to 30 km/s. The RIT features a high mass efficiency and a long life time, both of which are essential for the use in space. In the past years the RIT was miniaturized from 35 cm diameter to 2.5 cm leading to thrusts in the order of micro-Newtons ($\mu\text{N-RIT}$), which are needed for ultra fine positioning of the spacecraft. However, the investigation of the plasma discharge required for understanding and optimizing these small thrusters is hardly accessible by experiments.

For this task simulations are a powerful tool. For the simulation of the inductively coupled plasma in a $\mu\text{N-RIT}$, “PlasmaPIC” was developed from scratch in this thesis. “PlasmaPIC” distinguishes itself with a fully three-

dimensional simulation of a plasma discharge. By this means the plasma discharge in arbitrary geometries can be investigated. The geometry can be imported from common computer aided design (CAD) tools. The very long simulation time of a three-dimensional simulation was reduced from month to several hours by incorporating a massive parallelization.

“PlasmaPIC” enables access to high spatial and temporal resolution of plasma parameters that are hardly accessible by experiments. Knowledge of these parameters is not only required for the optimization of the plasma discharge in the thruster, but is also decisive for the design of the ion optics of the grid system. An improvement of the ion optics is also needed for the ion sources used for the material processing.

In this thesis the features of “PlasmaPIC” are presented. Furthermore, the capabilities of “PlasmaPIC” are outlined by applying “PlasmaPIC” to the simulation of a $\mu\text{N-RIT}$. Moreover, it will be shown that “PlasmaPIC” is not limited to the simulation of $\mu\text{N-RIT}$. Indeed, “PlasmaPIC” can be used for many different low pressure plasma discharges.

At the beginning in chapter 2 the fundamentals of a low pressure plasma are described. In this framework several discharge types are presented and the working principle of a RIT is outlined. Next, in chapter 3 the self-developed simulation tool “PlasmaPIC” is presented. This includes an overview of the functions of “PlasmaPIC” as well as a detailed description of the applied algorithms. In chapter 4 a thorough analysis of parallelization possibilities is done and the best one for “PlasmaPIC” is selected. By using up to 1000 CPU cores it will be demonstrated that “PlasmaPIC” has a great scalability. In chapter 5 the principles of simulating a $\mu\text{N-RIT}$ are explained with the focus on the neutral gas and its interaction with the plasma. Afterward, the capabilities of “PlasmaPIC” are demonstrated by showing and analyzing the results of the plasma discharge simulation in a $\mu\text{N-RIT}$. The impact of the power deposition and the neutral gas pressure is investigated in detail, and the latter will be compared with experiment.

Chapter 2

Theory

In this chapter a definition of a plasma is given and low pressure plasma discharges are introduced. Next, fundamental physical aspects of plasmas will be described. The focus lies on low pressure, low temperature plasmas. Therefore, the plasma wall interaction, heating mechanisms, and neutral gas dynamics will be explained. At the end the working principle of a radio frequency ion thruster (RIT) is described.

2.1 Definition and classification of plasmas

“A plasma is a quasi-neutral gas of charged and neutral particles, which exhibits collective behavior” [1]. Quasi-neutrality means that the total charge of the whole plasma is zero but local imbalance of the neutrality is allowed. Collective behavior is defined as the response of the charged particles to external and internal electromagnetic fields. The particles arrange in a manner that the fields will be screened and the quasi-neutrality is established. In the following the charged particles are assumed to be either positive ions, or electrons. In a quasi-neutral plasma the plasma density n_p corresponds to the ion density n_i , which is equal to the electron density n_e . The ionization degree α gives the fractional ionization of the neutral gas n_g

$$\alpha = \frac{n_i}{n_i + n_g} . \tag{2.1}$$

A further characteristic of a plasma is its temperature T . It is being distinguished between hot and cold plasmas. In hot plasmas all particles are in a thermal equilibrium, meaning that $T_e = T_i$. In cold plasmas the electrons and the ions have individual temperatures $T_e \neq T_i$. Commonly, the ions are in a thermal equilibrium with the neutral gas, whose temperature is in the order of several hundred Kelvin. The temperature T of an ensemble of particles can be expressed as a mean energy \bar{E} consisting of $1/2k_B T$ per degree of freedom. For a free moving particle in three dimensions the mean energy is

$$\bar{E} = \frac{3}{2}k_B T. \quad (2.2)$$

In general the temperature in a plasma is given in units of eV [1]. Hence, a temperature of 1 eV corresponds to

$$\frac{1 \text{ eV}}{k_B} \approx 11,600 \text{ K}. \quad (2.3)$$

This thesis treats low pressure plasma discharges. Following Lieberman and Lichtenberg [2] the plasma properties in a low pressure discharge are given by

$$T_e \approx 1 - 10 \text{ eV}, \quad (2.4)$$

$$T_i \ll T_e, \quad (2.5)$$

$$n_p \approx 10^{14} - 10^{19} \text{ m}^{-3}, \text{ and} \quad (2.6)$$

$$n_g \approx 10^{19} - 10^{22} \text{ m}^{-3}. \quad (2.7)$$

The plasma in a low pressure discharge corresponds to the class of low temperature plasmas.

2.2 Characteristics of a plasma

As mentioned before, in a low pressure discharge electrons and ions are typically not in a thermal equilibrium. However, one species can be in a thermal equilibrium with itself. In general, electrons are assumed to be in a thermal equilibrium but there are exceptions. Ions are almost never in thermal equilibrium [2]. For simplification and to demonstrate the basic properties

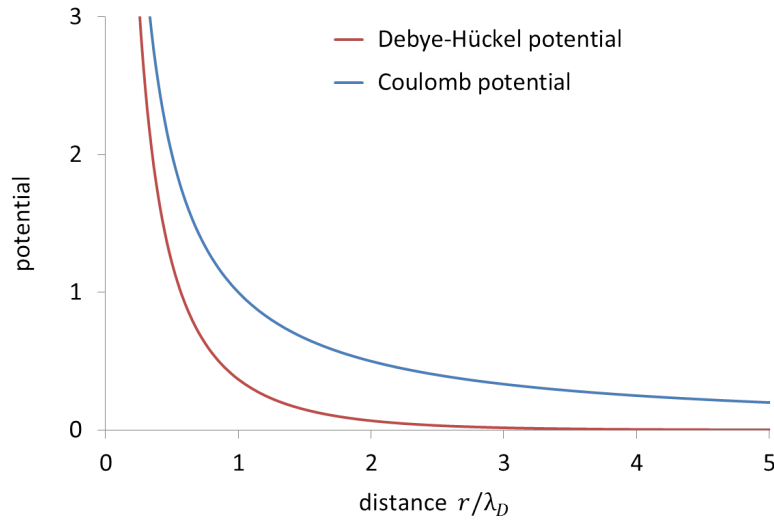


Figure 2.1: Comparison of the potential drop without a screening of the plasma (Coulomb potential) and the Debye-Hückel potential due to the screening effect of the plasma

of a plasma, a thermal equilibrium of the electrons is assumed. Hence, the electrons are described by the Maxwellian distribution

$$f(\vec{v}) = n \left(\frac{m}{2\pi k_B T} \right)^{3/2} \exp \left(-\frac{m\vec{v}^2}{2k_B T} \right), \quad (2.8)$$

where \vec{v} is the velocity, n the density, m the mass, and T the temperature of the species. Furthermore, the ions are assumed as a non-moving background charge due to their larger mass and lower energy. Under these conditions and without an electron drift the density distribution of the electrons is represented by Boltzmann's relation

$$n_e(\vec{r}) = n_0 \exp \frac{e\Phi(\vec{r})}{k_B T_e}, \quad (2.9)$$

where $\Phi(\vec{r})$ is the potential at the position \vec{r} and n_0 is the maximum electron density.

A remarkable characteristic of a plasma is its screening of internal and external electric fields. Free moving charged particles respond to potential differences and compensate them. Hence, the potential drop in a plasma cannot

be described with $1/r$ (Coulomb potential) - instead the potential is given by the Debye-Hückel potential [2]

$$\Phi(x) = \Phi_0 e^{-x/\lambda_{\text{De}}} , \quad (2.10)$$

where x is the distance to the location of the disturbing potential Φ_0 . λ_{De} is called the Debye length. It is the screening length where the potential has dropped to $1/e$ and is given by [2]

$$\lambda_{\text{De}} = \left(\frac{\epsilon_0 k_{\text{B}} T_e}{en_0} \right)^{1/2} . \quad (2.11)$$

The Debye length is a criterion for the distance at which the quasi-neutrality is fulfilled. Figure 2.1 compares the Coulomb potential with the Debye-Hückel potential.

Another fundamental parameter of a plasma is the electron plasma frequency

$$\omega_{\text{pe}} = \left(\frac{e^2 n_0}{\epsilon_0 m_e} \right)^{1/2} . \quad (2.12)$$

Considering the electrons and the ions as two clouds where the ion cloud is stationary, the electron plasma frequency specifies the oscillation frequency that arises due to a displacement of the clouds relative to each other. The electron plasma frequency is related to the Debye length by [3]

$$\lambda_{\text{De}} = \frac{v_e}{\omega_{\text{pe}}} , \quad (2.13)$$

where v_e is the thermal speed of the electrons in one dimension

$$v_e = \sqrt{\frac{k_{\text{B}} T_e}{m_e}} . \quad (2.14)$$

By generalizing that both clouds are moving, the plasma frequency is given by [2]

$$\omega_{\text{p}} = (\omega_{\text{pe}}^2 + \omega_{\text{pi}}^2)^{1/2} , \quad (2.15)$$

where ω_{pi} is the ion plasma frequency. This frequency is given by [2]

$$\omega_{\text{pi}} = \left(\frac{e^2 n_0}{\epsilon_0 M} \right)^{1/2}, \quad (2.16)$$

where M is the mass of the ions. As can be seen the ion plasma frequency is much smaller due to the high mass of the ions. Therefore, the plasma frequency is determined by the electron plasma frequency.

For a plasma a complex dielectric constant as well as a complex conductivity can be defined. Still considering the ions as fixed, the complex dielectric constant is given by [2]

$$\epsilon_{\text{P}} = \epsilon_0 \left[1 - \frac{\omega_{\text{pe}}^2}{\omega(\omega - i\nu_{\text{m}})} \right], \quad (2.17)$$

where ω is the frequency of the applied electric field and ν_{m} is the collision frequency of the electrons with the neutral gas. The complex conductivity is given by [2]

$$\sigma_{\text{P}} = \frac{\epsilon_0 \omega_{\text{pe}}^2}{i\omega + \nu_{\text{m}}}. \quad (2.18)$$

Consequently, the plasma can be considered a conductor or a dielectric. For $\omega \ll \nu_{\text{m}}, \omega_{\text{pe}}$ the plasma is best described by a conductivity [2]

$$\sigma_{\text{dc}} = \frac{\epsilon_0 \omega_{\text{pe}}^2}{\nu_{\text{m}}} = \frac{e^2 n_0}{m_e \nu_{\text{m}}}, \quad (2.19)$$

which is also called the dc plasma conductivity. For $\omega \gg \nu_{\text{m}}$ it is more accurate to consider the plasma a dielectric with [2]

$$\epsilon_{\text{P}} = \epsilon_0 \left(1 - \frac{\omega_{\text{pe}}^2}{\omega^2} \right). \quad (2.20)$$

2.3 Plasma sheath

The last section dealt with fundamental properties of an unbounded plasma. In this chapter the properties of a bounded plasma will be analyzed [2]. The plasma is considered as quasi-neutral, which means that the numbers of electrons and single charged positive ions are equal. Furthermore, particle-particle

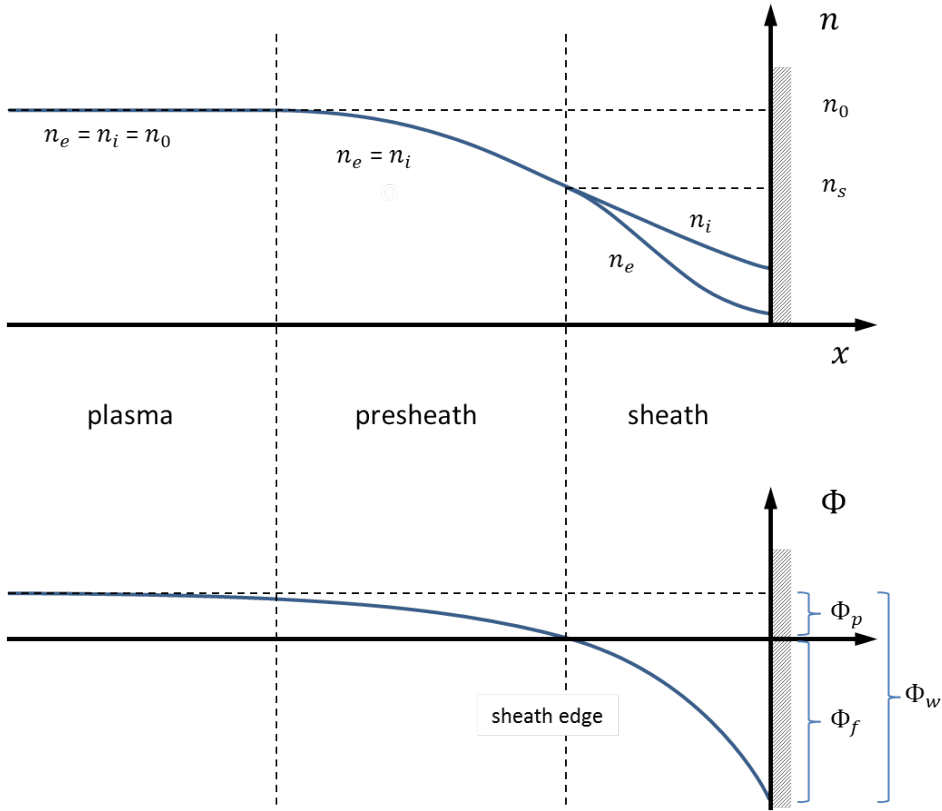


Figure 2.2: Qualitative characteristic of the density (top) and potential (bottom) for a bounded plasma [2].

collisions are neglected. Bringing a plasma in contact with walls, more electrons than ions will strike the wall due to a considerable larger mobility. Consequently, a potential difference between the plasma and the wall is formed. The potential difference retards the electrons and accelerates the ions to the wall. Thereby, the current onto the wall increases for the ions and decreases for the electrons. Equilibrium is reached if both currents onto the wall are equal. This effect is also known as “ambipolar diffusion”. In the region between plasma and wall, a point must exist where the quasi-neutrality of the plasma will break down. This point is called sheath edge and is used to divide the region into two parts, the presheath and the sheath. Figure 2.2 illustrates that the presheath is next to the plasma and the quasi-neutrality of the plasma holds in this region. The sheath itself is next to the wall and positively charged. The size of the presheath is in the order of many Debye lengths, while the sheath extends over only a couple of Debye lengths. It is common to define the potential at the

sheath edge as zero. For the continuity of the ion current the ions must have a certain velocity at the sheath edge. This velocity is called Bohm velocity and is given by [3]

$$u_B = \left(\frac{k_B T_e}{M} \right)^{1/2}, \quad (2.21)$$

where T_e is the electron temperature and M the mass of the ion. To accelerate an ion to the Bohm velocity a potential drop in the presheath is required

$$\Phi_p = \frac{k_B T_e}{2}, \quad (2.22)$$

which is called the plasma potential. Using the Boltzmann relation (2.9) the density at the sheath edge is defined by

$$n_s = n_p e^{-\Phi_p/T_e}, \quad (2.23)$$

where n_p is the plasma density in the bulk. The potential drop in the sheath

$$\Phi_f = \frac{1}{2} k_B T_e \ln \left(\frac{2\pi m_e}{M} \right) \quad (2.24)$$

is called floating potential [3] and is negative. The total potential difference between the plasma and the wall is the sum of the magnitude of both potentials and is called plasma wall potential Φ_w .

2.4 Particle-particle collisions in a low pressure discharge

In the last sections the interaction of electrons and ions in a plasma was discussed. In general, a third interaction partner, the neutral gas, has to be considered as well. This interaction is carried out by collisions of electrons and ions with the neutral gas particles. At densities higher than those found in low pressure discharges, collisions between charged particles have to be considered as well. The collisions are classified into elastic and inelastic collisions. In an elastic collision the total kinetic energy of both collision partners is conserved. However, the total kinetic energy can be redistributed between the collision

partners. If the total kinetic energy is not conserved the collision is called inelastic. Momentum conservation is satisfied in both cases.

To describe a collision several parameters are essential. The most important one is the cross section σ . As the name implies the cross section gives information about an effective area of the interaction between the collision partners.

Using the cross section, a mean free path for a particle moving through an assembly of resting particles with a density n_g can be defined [2]

$$\lambda = \frac{1}{n_g \sigma} . \quad (2.25)$$

For a particle with the velocity v the mean time between two collisions is given by [2]

$$\tau = \frac{\lambda}{v} . \quad (2.26)$$

The number of collisions in a certain time is given by the collision frequency

$$\nu \equiv \frac{1}{\tau} = n_g \sigma v . \quad (2.27)$$

The probability for a particle to have a collision traveling a distance x is given by

$$P(x) = 1 - e^{-n_g \sigma x} . \quad (2.28)$$

In the following, collision types are described that are important for low pressure discharges. The elastic scattering of an electron on a neutral gas particle is of main relevance for the heating of a plasma as will be shown later. For the collision of an electron and a neutral gas particle, the cross section depending on the scattering angle χ is given by [4]

$$\sigma(E, \chi) = \frac{E}{4\pi \left[1 + E \sin^2\left(\frac{\chi}{2}\right)\right] \ln(1 + E)} \cdot \sigma(E) , \quad (2.29)$$

where E is the energy of the incident electron. The neutral gas particle is assumed as stationary.

The elastic collision of an ion and neutral gas particle as well as the collision between two neutral gas particles are described by the so called hard sphere model. In this model the atoms are assumed as solid spheres. The occupied

volume of an atom is described by the van der Waals radius r [5]. The cross section is determined by the smallest distance d between the center of two atoms

$$\sigma = \pi d^2 = 4\pi r^2. \quad (2.30)$$

A collision is inelastic if an excitation or ionization of a collision partner takes place. In that case the total kinetic energy of the collision partners is reduced by the excitation or ionization energy. The excitation as well as the ionization requires a threshold energy. Both threshold energies are in the order of several eV. In a low temperature plasma the ions are at the same temperature as the neutral gas particles, e.g. at room temperature of about $300 \text{ K} \equiv 1/40 \text{ eV}$. Therefore, only the electron-neutral excitation or ionization is important. Moreover, a further excitation or ionization of ions due to electrons is negligible in low temperature plasmas because of the low ionization degree.

For the ion-neutral collision a special case has to be considered. For this collision type it can occur that an electron hops from the neutral gas particle to the ion. This process is called charge exchange collision. If the ion and the neutral gas particle are of the same species this collision is elastic. For particles of different species this is an inelastic collision due to different energy levels. For an ion-neutral collision of the same species it is difficult to distinguish between a charge-exchange and an elastic scattering event.

2.5 Low pressure plasma discharges

According to Lieberman and Lichtenberg [2], a low pressure plasma discharge consists of a plasma with the properties (2.4) - (2.7) that is driven by electrical fields, meaning the plasma is heated by electrical fields. To sustain a plasma discharge, a heating of the plasma is necessary to compensate the energy losses in a plasma. The energy is lost due to excitation and ionization of neutral gas particles as well as by electrons and ions striking a wall. The latter loss is mainly carried by the ions that are accelerated towards the wall by the plasma wall potential.

In the following the heating mechanisms are described. Furthermore, two types of discharges are discussed in detail.

2.5.1 Heating mechanisms

In a low pressure plasma discharge the main energy transfer into the plasma is carried by the coupling of the electric field with the electrons. One reason for this is that the ions are too immobile to react to varying electric fields. Another reason is that even if they gather energy their collision probability is extremely low due to their slow velocity. As a consequence, most ions impact on the walls, where their energy is lost.

Lieberman and Lichtenberg classify the electron heating mechanism into the following categories [2]:

- ohmic heating
- stochastic heating
- resonant wave-particle interaction heating
- secondary electron emission heating

In ohmic heating, collisions of the electrons with the neutral gas scatter the accelerated electrons in arbitrary directions. Thus the energy acquired from the electric field is transformed into thermal energy of the electrons. This heating mechanism is present in all discharges.

Stochastic heating is also called collisionless heating and exists in discharges with alternating electric fields. These alternating electric fields cause the plasma sheath to oscillate. By scattering of the electrons with this alternating sheath the electrons can obtain energy.

Resonant wave-particle interaction heating is present for example in an electron cyclotron resonance plasma. In this case the electron is trapped by a static magnetic field and heated up by an electric field with the cyclotron resonant frequency.

An impinging ion on the wall can release a secondary electron. This secondary electron will be accelerated into the plasma by the sheath potential and acquire energy due to this. Therefore, this type of heating is called secondary electron emission heating.

2.5.2 DC discharge

In a common DC discharge a plasma exists between two plates located opposite to each other that are on different potentials. Due to the electric field a current is established by the free moving charged particles. The current is mainly carried by the electrons moving toward the anode where they will be absorbed. To sustain a discharge the charged particle loss has to be compensated by the generation of new electrons and ions. One generation process is the ionization collision of the accelerated electrons with the neutral gas. This generation process is described by the first Townsend coefficient α that gives the information of ionization processes per distance. The first Townsend coefficient depends on the gas species, the pressure p , and the electric field E . Another generation process is the production of seed electrons by cosmic radiation, photo-emission at the cathode, or the secondary electron emission at the cathode induced by the impinging ions. The secondary electron emission is described by the second Townsend coefficient γ_{se} . It depends on the gas species, the cathode material and the temperature of both of them (see section 2.6).

If the discharge is sustained by the two Townsend coefficient only, the discharge is denoted as self-sustained. The condition of a self-sustained discharge expressed by both Townsend coefficients α and γ_{se} is given by [2]

$$\alpha d = \ln \left(1 + \frac{1}{\gamma_{se}} \right), \quad (2.31)$$

where d is the distance between the two plates. This is also called the breakdown condition. Having in mind that α depends on the gas species, the neutral gas pressure, and the electric field, a breakdown condition can be used to estimate the breakdown voltage for a self-sustained discharge at a certain neutral gas pressure p and a plate distance d by [2]

$$V_b = \frac{Bpd}{\ln(Apd) - \ln \left[\ln \left(1 + \frac{1}{\gamma_{se}} \right) \right]}, \quad (2.32)$$

where A and B are constants for the gas species including the ionization frequency. The resulting voltage curve as a function of the product pressure p and distance d is also called ‘‘Paschen curve’’ [2].

Depending on the current related to the voltage, DC discharges are divided into different types as shown in figure 2.3. Beneath a current I_A the voltage increases up to the breakdown voltage with the current. This discharge is not self-sustained and is called “Townsend discharge” [2]. Above a current I_C an arc discharge is established. It is characterized by a fast decreasing voltage for higher currents. Between the currents I_A and I_C the glow discharge exists. It is self-sustained and is distinguished by an almost constant voltage over a large range of the current. The plasma densities are in the order of 10^{14} m^{-3} . In the following the glow discharge is analyzed.

In a glow discharge the area between the two plates is separated into different regions as shown in figure 2.4. The figure shows which region gives light and shows the spatial distribution of the potential and the space charge. In the following, these regions will be explained by tracking the traversing electrons from the cathode to the anode [6]. The emitted electrons of the cathode are accelerated in a strong electric field. Reaching the cathode sheath the electrons have gathered enough energy for excitation events and therefore this region gives light. By a further acceleration the electrons have sufficient energy for ionization events and an avalanche effect is initiated leaving the ions behind. Hence, a positive charge is formed that screens the electric field. This region is called Hittorf dark space. Next to this region the negative glow is located. In this region the electrons are weakly accelerated due to the screened field. Consequently, the electron density is high and the excitation events are dominating. By the excitation events the electrons lose energy that cannot be compensated due to the weak electric field. The Faraday dark space is formed where the electron energy is too low for further excitation events. In the positive column the electrons are further weakly accelerated by a relative constant electric field. The electrons in this region have an average energy of 1-2 eV and the excitation and ionization takes place due to the fast electrons of the Maxwell distribution. The positive column can expand arbitrary depending on the plate spacing, in contrast to the other fixed regions. In the positive column striations can occur if the electric field per distance is sufficient [7, 8]. This is similar to the Frank-Hertz effect except that the striations can move.

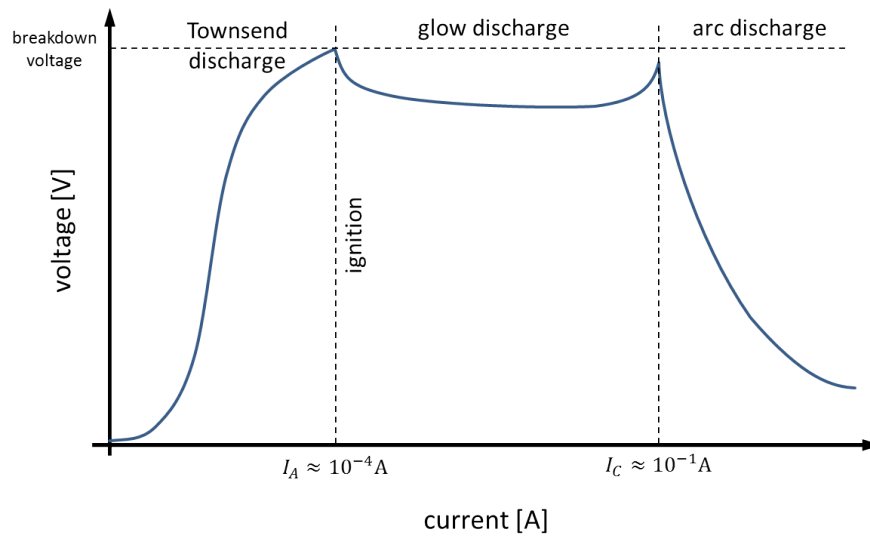


Figure 2.3: Different types of DC discharges depending on the current voltage characteristic (note the logarithmic scale at the x axis).

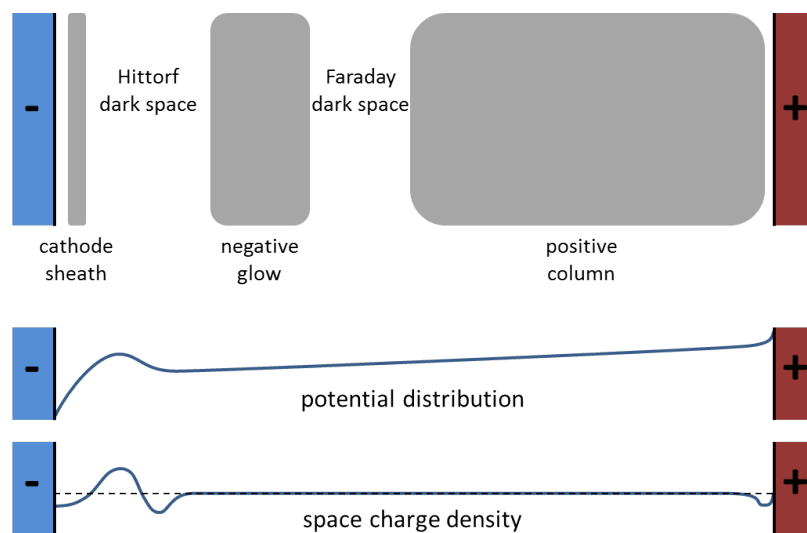


Figure 2.4: Behavior of the potential and charge density in the different areas of a glow discharge.

As it can be expected the power absorption of the DC glow discharge is comparable to an ohmic conductor and is given for a cylindrical discharge chamber with a radius R by [2]

$$P_{\text{abs}} = 2\pi \int_0^R j \cdot E r \, dr , \quad (2.33)$$

where j is the current density and E the electric field. Both are perpendicular to the plates.

2.5.3 Inductive discharge

Inductive discharge sources are categorized in cylindrical and planar types [2]. In the planar type a wire is wrapped in circles with decreasing radius. This coil is located on top of a cylindrical discharge chamber. Therefore, it is also known as a top coil. In the cylindrical type the coil is wrapped around a cylindrical discharge chamber.

In both cases an alternating current in the coil produces an alternating electric field that accelerates the electrons inside the discharge chamber. The power absorption during one period T in a volume V is

$$P = \frac{1}{T} \int_T \int_V \vec{j}(\vec{r}, t) \cdot \vec{E}(\vec{r}, t) \, dV dt , \quad (2.34)$$

where $\vec{j}(\vec{r}, t)$ is the current in the plasma and $\vec{E}(\vec{r}, t)$ the total electric field of the coil and the plasma. Due to the high mass ratio of electrons and ions only the electron motion contributes to the plasma current. The induced electron current produces an electrical field with opposite direction, due to Lenz's law. For that reason, the electrical field of the coil will be screened and can penetrate the plasma up to a certain depth. Only in this region power can be coupled to the plasma. For the inner regions of the plasma the electrical field will vanish and the power deposition is zero due to formula (2.34). The skin depth δ defines the depth of which the electrical field has reduced to $1/e$. For the skin depth three cases are distinguished.

The first case is the collisionless case. The collision frequency of the electrons is considerably smaller than the driving frequency of the coil ($\nu_m \ll \omega$). In this case the electric field is reflected. Assuming a sinusoidal electric field of

the coil the current of the electrons is also sinusoidal with a phase shift of $\pi/2$, because without collision they can perfectly follow the electric field. Hence the power deposition is zero in the plasma. The collisionless skin depth is given by [2]

$$\delta_r = \left(\frac{m_e}{e^2 \mu_0 n_s} \right), \quad (2.35)$$

where n_s is the electron density in the skin layer assumed as constant in that region. In this case the skin depth depends on the electron density only.

The second case is the so called collisional case. The collision frequency of the electrons is higher than the driving frequency ($\nu_m \gg \omega$). Due to the collisions of the electrons the phase shift is changed and a power absorption takes place. The collisional skin depth is given by [2]

$$\delta_c = \left(\frac{2m_e \nu_m}{\omega \mu_0 e^2 n_s} \right)^{1/2}. \quad (2.36)$$

The electron density as well as the collision frequency of the electrons are assumed as constant in the skin layer. Expressing the collisional skin depth in terms of the collisionless skin depth and having in mind that ($\nu_m \gg \omega$)

$$\delta_c = \delta_r \left(\frac{2\nu_m}{\omega} \right)^{1/2}. \quad (2.37)$$

It is obvious that the collisional skin depth is larger than in the collisionless scenario. This is the case because the current of the electrons is reduced by the collisions.

The third type is the anomalous skin effect [2, 3, 9]. In this effect the distance an electron travels during one frequency period without a collision is comparable to or larger than the skin depth for that case. The condition for the anomalous skin depth is given by [3]

$$\omega \delta_a \leq \left(\frac{k_B T_e}{m_e} \right)^{1/2}. \quad (2.38)$$

Even without particle-particle collisions a power absorption is possible in that situation. The power is absorbed by stochastic heating. As a consequence of the fast motion of the electrons through the skin layer the conductivity is a

function of the electric field at each position. The skin layer is larger than both earlier ones. For a description of the skin layer a kinetic treatment is required. A detailed analysis can be found in [2].

By reducing the plasma density the skin layer will increase. If the skin layer exceeds the radius of the discharge chamber the inductive coupling becomes less efficient. To sustain a plasma at this condition the eddy electric field has to be increased by applying a higher current of the coil. With an increasing current in the coil the ohmic losses of the coil lead to an increasing voltage drop along the wire to the coil. As a consequence, a strong AC electric field arises between the single loops of the coil, which act like a capacitor. The plasma is now heated by capacitive coupling that is carried by stochastic heating. For inductive discharge sources it is distinguished between the e-mode (capacitive coupling) and h-mode (inductive coupling). Both cases differ in the total power absorption that is larger in the inductive mode. The plasma density for the h-mode is in the range of $10^{17} - 10^{19}$ whereas the density for the e-mode is a factor of 10 smaller [3].

2.6 Secondary electron emission - induced electron emission

Secondary electron emission describes the effect of the emission of an electron from a surface due to an ion that strikes the surface. The electron emission can be caused by a potential emission or a kinetic emission. Regarding a potential emission, the striking ion is deionized by an electron of the surface. If the freed ionization energy E_I of the striking species is at least twice as large as the work function of the surface W_A an additional electron can be emitted from the surface

$$E_I \geq 2 \cdot W_A . \quad (2.39)$$

This effect is related to the auger neutralization [10, 11]. The emission probability for the potential emission is independent of the ion velocity. On the contrary, the velocity of the incident ion is the cause for the electron emission in the case of kinetic emission.

In this case, the kinetic energy of the ion is transferred to the electron due to multiple collisions in the solid. Multiple collisions are required because the energy transfer in one ion-electron collision is tiny due to the high mass ratio. As a result, the incident ion must have a high threshold energy for a kinetic electron emission. This threshold energy depends on the ion species, the surface material, and surface temperature. However, this threshold energy is in general in the order of several keV [12].

Moreover, it is important to mention that the potential emission takes place just before the ion strikes the wall. Hence, it is possible that the new neutral particle can induce a kinetic emission.

In a low pressure discharge the most common occurrence is that the incident ion has a kinetic energy of a few eV, which it has gained due to the plasma wall potential. For that reason the kinetic emission can be neglected in the following.

2.7 Neutral gas flow

As we have seen, the neutral gas is an essential part of low pressure plasma discharges. It is the source for the creation of new charged particles. Furthermore, it is the cause for ohmic heating. In low pressure plasma discharges the pressure is in the range of 0.1 Pa - 100 Pa. In contrast to charged particles the neutral particles can only change their momentum and energy in collisions. At such low pressures the particle-particle collisions are quite unlikely and particle-wall interactions have an increasing influence on the neutral gas flow. To distinguish the different flows that can occur at low pressures the Knudsen number was defined [13]

$$Kn := \frac{\lambda}{d}, \quad (2.40)$$

where d is the distance between the walls and λ is the mean free path of a particle in a gas that is given by [13]

$$\lambda = \frac{1}{\sqrt{2}\pi d^2 n}. \quad (2.41)$$

The Knudsen number gives the probability that a particle undergoes a particle-particle collision between two wall collisions. Using the Knudsen number the gas flows for low pressures are divided into the following kinds [13]

$$Kn > 0.5 \qquad \qquad \qquad \text{molecular flow} \qquad \qquad (2.42)$$

$$0.5 > Kn > 0.01 \qquad \qquad \text{transition flow} \qquad \qquad (2.43)$$

$$Kn < 0.01 \qquad \qquad \qquad \text{viscous flow} \qquad \qquad (2.44)$$

In the viscous flow, the particle-particle collisions dominate. The gas can be described by fluid dynamics. On the contrary, the particle-particle interactions are negligible for molecular flow. In that case the particles travel between the walls on straight paths. The transition flow is, as the name implies, a zone where the fluid dynamics are not valid but particle-particle interactions are still relevant. That flow is the most difficult one to describe. Because the neutral gas flow in a RIT is mainly in the molecular flow, that type will be described in more detail.

As it was shown, the particle-wall interactions are important for the molecular and transition flow. There are two different types of particle-wall interaction shown in figure 2.5. The specular reflection occurs at special treated surfaces that are clean and smooth. The velocity of the particle is mirrored. The diffuse reflection is present at common rough surfaces. For this kind it is assumed that the incident particle resides at the wall for a short moment. During this time the particle can gain or lose energy, depending on the particle and wall temperature. This process is called accommodation. Afterward, the particle is re-emitted with a new velocity that is independent of the incident energy and direction. To preserve the isotropic velocities in the gas, the angle distribution has to be a cosine function. To sustain a Maxwell distribution in the gas the energy distribution of the emitted particles has to be a Maxwellian distribution multiplied with the velocity to take care of the fact that fast particles strike a wall more often [14].

The diffuse reflection at the surface leads to an interesting effect for the conductance of a tube with molecular gas flow. A particle that enters a tube on one side can exit the tube on either side. This occurs because a particle can change its direction to either side with the same probability at each wall

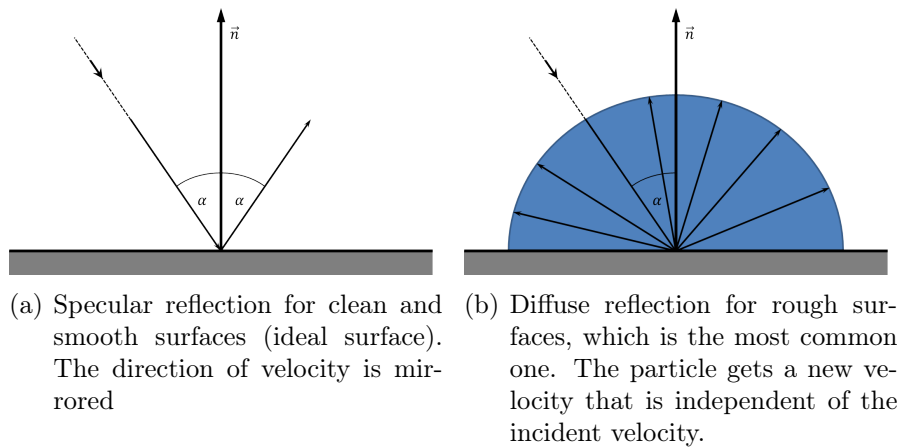


Figure 2.5: Different wall interaction types depending on the surface.

collision. On the contrary, in viscous flow the particles will be pushed through a tube due to particle-particle collisions assuming a pressure gradient in the tube. For molecular flow, the transmission coefficient of a tube depends only on the geometry. Depending on the geometry the transmission coefficient and hence the conductance can get very small. This aspect is important for the radio frequency ion thruster, which will be described in the next section.

2.8 Micro-Newton radio frequency ion thruster

A micro-Newton radio frequency ion thruster ($\mu\text{N-RIT}$) is designed for space operation. As the name implies, the thrust is in the order of μN . Due to the momentum conservation the thrust is achieved by ejecting ions out of the $\mu\text{N-RIT}$. Figure 2.6 shows the basic set-up of a $\mu\text{N-RIT}$. Through the gas inlet neutral gas enters the discharge chamber which is surrounded by a coil. By applying an alternating current to the coil, a plasma is generated and sustained due to inductive coupling. With an electrostatic extraction system only the ions will be accelerated out of the thruster. In the simplest case the extraction system consists of two grids on different potentials. After traversing the extraction system, exiting ions have a kinetic energy up to a few

keV depending on the grid potentials. For the current I of exiting ions with the mass M the resulting thrust is given by [15]

$$T = I \sqrt{2 \frac{M}{q} U}, \quad (2.45)$$

where q is the charge of the ions, which are most often single charged, and is U the total potential drop. The potential drop consists of the potential difference of both grids and the plasma potential. As can be seen from (2.45) heavier ions lead to a higher thrust for the same potential difference. For that reason xenon is used as a propellant. The extraction current is determined by the current of the electrons on the inner grid. Due to the quasi-neutrality of the plasma the magnitudes of both currents have to be the same. By knowing the extraction current and the applied potential difference the thrust can be determined.

The main characteristics of a RIT are the following: The inductive coupling does not require electrodes, which would be destroyed due to ion sputtering. Therefore, RITs have a long life time that is essential for space operations. The extraction grid has two advantages. Firstly, the ions can be specifically accelerated. By increasing the potential difference the propellant is more efficiently used. Secondly, the extraction grid causes a high difference in the flow conductance for ions and neutral gas. Ions are pulled out by the electric field, whereas the neutral gas particles have a very low probability for traversing the grid system. As a consequence the ratio of leaving ions to leaving neutral gas particles can be 70 to 30, although the ionization degree in the plasma is in the order of 1%.

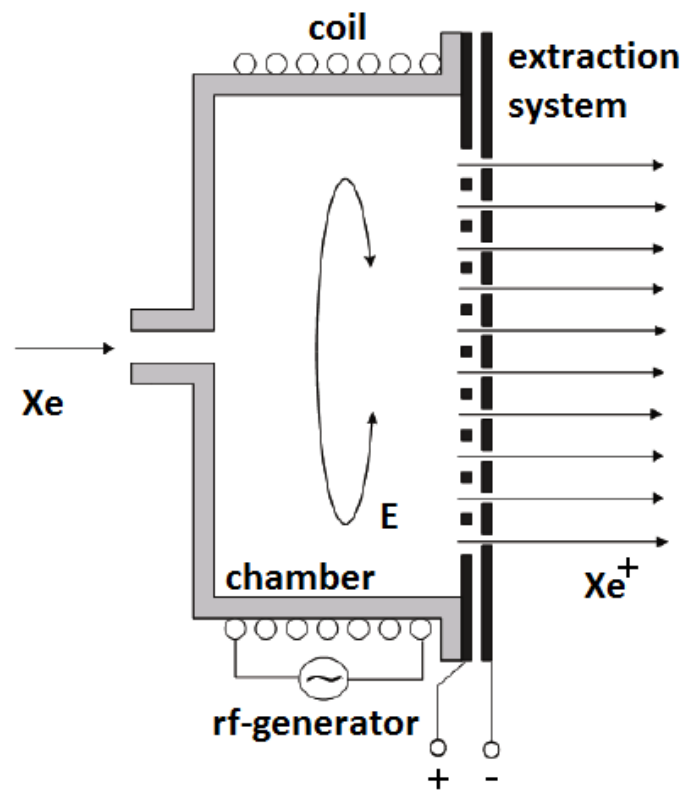


Figure 2.6: Operating principle of a radio frequency ion thruster [16].

Chapter 3

“PlasmaPIC” - a 3D plasma simulation program

“PlasmaPIC” was developed from the scratch within the scope of this PhD thesis. That was necessary because for the modeling of a $\mu\text{N-RIT}$ no applicable simulation software was available at this time. Although “PlasmaPIC” was intended for modeling a $\mu\text{N-RIT}$, it can be used for a wide range of low pressure low temperature plasmas. It is designed in the style of “Xoopic” [17] and is equally an object oriented plasma simulation code written in C++. In contrast to “Xoopic”, which is a two-dimensional rotationally symmetric code, “PlasmaPIC” is a fully three-dimensional code. It supports electrostatic as well as electrodynamic treatment. The most unique characteristic of “PlasmaPIC” is the incorporated massive parallelization of all program routines, which is highly efficient as will be demonstrated in chapter 4. This sophisticated massive parallelization is essential for the three-dimensional simulation of a $\mu\text{N-RIT}$ within a reasonable time. By this means the computation time is reduced from months to several hours. Moreover, “PlasmaPIC” distinguishes itself by the handling of arbitrary geometries. That way, “PlasmaPIC” is well suited for simulation of DC-discharge and RF-discharge devices with nonsymmetric geometries. For a high flexibility of “PlasmaPIC” a vast set of parameters can be committed by the user due to the usage of input cards. In the following a motivation for the used numerical technique is provided. Next, an overview of the used algorithm is given as well as a description of the individual parts of “PlasmaPIC”.

3.1 Motivation for a PIC code

To simulate a plasma there are several different numerical techniques. Birdsall and Langdon [18] divide numerical plasma simulations into two main classes – kinetic and fluid codes.

In plasma fluid codes, magnetohydrodynamic equations are solved. The plasma is treated as a continuum described by distribution functions. In kinetic codes the focus lies on single particle behavior instead of a continuum. In this context, kinetic equations (Vlasov or Fokker-Planck) are solved or particles are simulated. In the latter case single particles are simulated by moving and interacting with each other as well as with existing fields. If the distribution function is unknown the fluid description is not applicable unlike the kinetic codes, where distribution functions are not required as they are calculated by them. For the example of a low particle-particle interaction the Maxwell distribution cannot be guaranteed. Thus, the distribution function is unknown.

Kinetic codes have much higher computational requirements than the fluid codes. To take advantage of both codes there are also hybrid codes in which the distribution function is calculated by the kinetic part and used in the fluid part with less computational effort.

Hockney and Eastwood [19] classify the particle simulation into three parts. First, in the particle-particle simulation all particles interact with each other by calculating the forces between them. With an increasing number of particles the computational effort rises rapidly. To model a larger system the particle mesh simulation is applied in which particles do not interact directly with each other but via a mesh. Although this approach is faster, it does not reflect an accurate description of the near field interaction of particles. In addition, the mesh size has to resolve the smallest wave length of importance. Hybrid codes exist here as well called pppm or p3m (particle-particle particle-mesh). In low temperature plasma, as present in a radio frequency ion thruster, there are less particle-particle interactions and hence the distribution function is not well-known. Therefore, a kinetic code is required.

Due to the high number of particles and for reasons of computational effort only a particle-mesh code is applicable. Derived from that, a particle-mesh

code – commonly called particle in cell (PIC) – is utilized in “PlasmaPIC” and will be introduced in the following.

3.2 PIC-MCC scheme

In this section the functionality of a particle in cell code including the extension of a Monte Carlo Collision (PIC-MCC) [18, 20] is described, on which “PlasmaPIC” is based.

For the understanding of the PIC method one needs to know that the simulation space is represented by a mesh consisting of cells and grid points. To reduce the computational effort, not every real existing particle is considered in the simulation, but only a representative number of particles (super-particles). These super-particles behave in the same manner as normal particles. The only difference is that they stand for a certain amount of normal particles. The simulation time is discretized and advanced in time steps. One cycle of a time step is shown in figure 3.1. For each grid point j a force F_j exists. This force is weighted to the position \vec{x}_i of each particle i . All particles are accelerated and thus a new velocity \vec{v}_i and position are calculated for each particle. After moving these particles they have to be tested for intersections with boundaries. In the case of an intersection, several effects need to be considered. For example, depending on the boundary properties, the particle could be absorbed or a new particle emitted. Next, each particle has to be checked for a Monte-Carlo-Collision. This can be a collision with another particle or with the neutral background gas. Due to the collision with the neutral back-

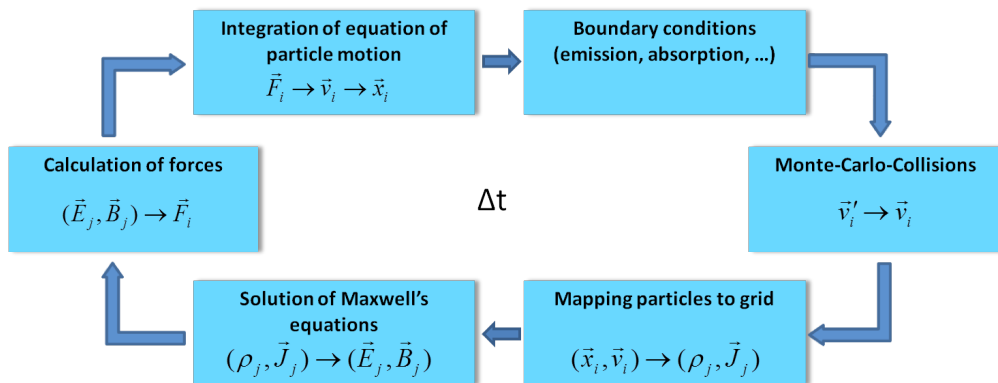


Figure 3.1: One cycle of a time step in a PIC-MCC simulation.

ground gas new particles can be created and thus sustain a plasma discharge. Thereafter, the charge and the velocity of each particle are weighted to the grid depending on its position. This leads to a new charge ρ_j and current \vec{J}_j at the grid points of the mesh. By applying Maxwell's equations to each grid point the new electric \vec{E}_j and magnetic \vec{B}_j fields are derived. Within the last step of the cycle a new force at each grid point is determined.

3.2.1 Integration of the equations of motion

For advancing the charged particles, electrons and ions, the leap-frog method is applied [18] in "PlasmaPIC". In the pure electrostatic treatment the two required first-order differential equations for the particle integration are as follows

$$m \frac{d\vec{v}}{dt} = q\vec{E} \quad \text{and} \quad (3.1)$$

$$\frac{d\vec{x}}{dt} = \vec{v}, \quad (3.2)$$

where m is the mass and q is the charge of the particle. These formulas can be rewritten as finite-difference equations

$$m \frac{\vec{v}_{\text{new}} - \vec{v}_{\text{old}}}{\Delta t} = q\vec{E}_{\text{old}} \quad \text{and} \quad (3.3)$$

$$\frac{\vec{x}_{\text{new}} - \vec{x}_{\text{old}}}{\Delta t} = \vec{v}_{\text{new}}. \quad (3.4)$$

These formulas are time centered. Hence the leap-frog method is second-order accurate [19], meaning the order of the error is $\sim \Delta t^2$. This accuracy is required for a stable simulation. In the presence of a magnetic field the formula (3.1) transforms to

$$m \frac{d\vec{v}}{dt} = q \left(\vec{E} + \vec{v} \times \vec{B} \right). \quad (3.5)$$

To achieve a second-order accurate approximation the velocity at the right hand side has to be time centered as well. This is carried out by averaging the old, as well as the new velocity

$$m \frac{\vec{v}_{\text{new}} - \vec{v}_{\text{old}}}{\Delta t} = q \left(\vec{E} + \frac{\vec{v}_{\text{new}} + \vec{v}_{\text{old}}}{2} \times \vec{B} \right). \quad (3.6)$$

This formula can be simplified for an efficient implementation by decoupling the electric and magnetic forces. This is accomplished by using

$$\vec{v}_{\text{old}} = \vec{v}^- - \frac{q\vec{E}}{m} \frac{\Delta t}{2} \quad \text{and} \quad (3.7)$$

$$\vec{v}_{\text{new}} = \vec{v}^+ + \frac{q\vec{E}}{m} \frac{\Delta t}{2} \quad (3.8)$$

in (3.6) leading to

$$\frac{\vec{v}^+ - \vec{v}^-}{\Delta t} = \frac{q}{2m} (\vec{v}^+ + \vec{v}^-) \times \vec{B}. \quad (3.9)$$

In this formalism a particle with the velocity \vec{v}_{old} of the last time step is accelerated by the electric field for a half time step leading to an interim velocity \vec{v}^- of the particle. This interim velocity \vec{v}^- is rotated by the magnetic field and a new interim velocity \vec{v}^+ is obtained. Finally, the particle is accelerated again by the electric field leading to the new velocity \vec{v}_{new} of the particle for this time step. This scheme is called ‘‘Boris pusher’’ as described in [18].

3.2.2 Monte Carlo Collision module

The Monte Carlo Collision (MCC) module performing the ‘‘null collision’’ method [20] is required for modeling a plasma discharge with ‘‘PlasmaPIC’’. In the ‘‘null collision’’ method there is also a probability that no collision can occur. For the case that few collisions take place this method can save a huge amount of computational time, which will be described later on in more detail.

Via the MCC module the plasma-neutral gas interactions are described. This is an important aspect regarding the production of new ions and electrons, energy losses, and heating mechanisms. In the MCC module the neutral gas is assumed as a background gas and with a known density distribution.

Furthermore, it is assumed that the neutral gas distribution is not affected by the plasma and therefore is constant in time. In “PlasmaPIC” a neutral gas density distribution has to be provided in the input card, which can also be a single value for a homogeneous neutral gas distribution. The MCC module is also applied to model Coulomb collisions. These collisions are important at higher plasma densities but can be neglected for simulating a $\mu\text{N-RIT}$.

The probability for a particle with a velocity v to undergo a collision in a time step Δt is given by [20]

$$P = 1 - \exp(-\Delta t v \sigma_T(E) n(\vec{x})) , \quad (3.10)$$

where $\sigma_T(E)$ is the sum over all collision cross sections of interest at the kinetic energy E of the particle. $n(\vec{x})$ is the neutral gas density at the particle position \vec{x} . To test for a collision, a uniform random number between 0 and 1 is generated. If the random number is smaller than P , the collision takes place. For each particle the probability as well as a random number have to be determined. This leads to a high computational effort that is proportional to the number of particles. The computational effort can be reduced by introducing a maximum collision frequency [20]

$$\nu_{max} = \max(n(\vec{x})) \max(\sigma_T(E)) \max(v) \quad (3.11)$$

that a particle of this species can have. Knowing this frequency the maximum probability P_{max} and thus the maximum number of particles [20]

$$N_{max} = N (1 - \exp(-\nu_{max} \Delta t)) \quad (3.12)$$

that may undergo a collision in this time step are determined. Only for these particles a collision calculation has to be executed, rather than for each particle. Because this maximum number is in general smaller than the total number of particles by a factor of 100, a remarkable computational saving is achieved. The maximum number of colliding particles has to be chosen randomly from the total number of particles. For each particle the collision frequency of all collision types has to be determined. The difference of the maximum collision frequency and the sum of all collision type frequencies is the null collision

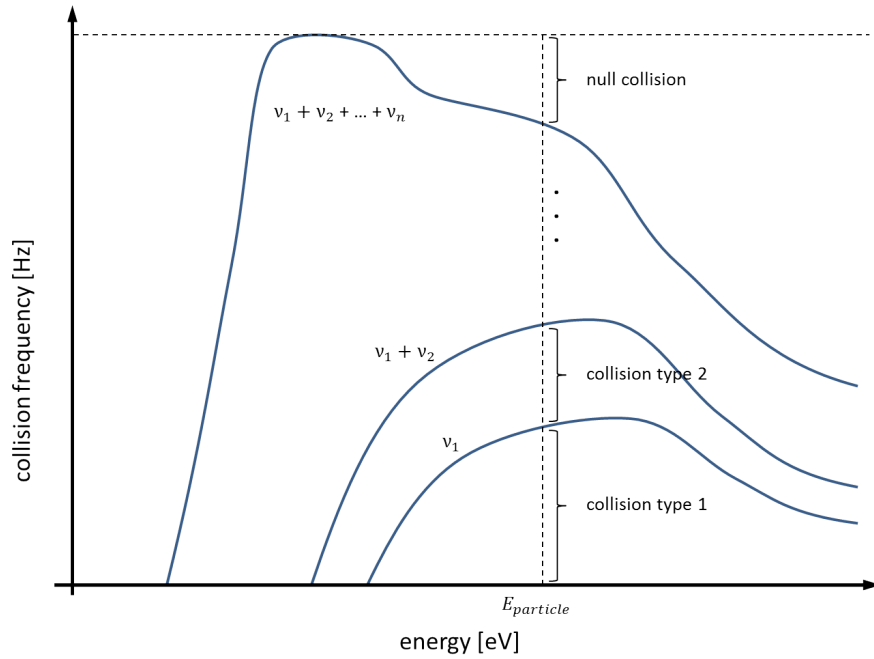


Figure 3.2: Summation over all collision frequencies for different collision types including the null collision leads to the constant maximum collision frequency.

frequency, which is demonstrated in figure 3.2. The probability of a certain collision type including the null collision type is proportionally weighted by their collision frequencies and selected by a uniform random number between 0 and 1. The computational effort for deriving the maximum collision frequency can be neglected for the following reasons. Since the cross sections of all reactions are fixed, the maximum of the sum over all cross sections has only to be calculated once before the first time step. The same is valid for the neutral gas density if it is assumed as constant in time, which is the most common case. The maximum velocity can be determined for the initialized distribution and has to be updated if a particle velocity is higher. Since the new velocity is calculated in each time step the additional computational cost can be neglected.

Electron-neutral collision:

For the electron-neutral collision the elastic scattering, the excitation, and the ionization are considered in “PlasmaPIC”. Due to the high mass and velocity difference of the collision partners, the neutral particle is assumed as fixed. The new velocity vector after one of these collision types is given by deriving

the scatter angle χ and the azimuthal scattering angle Φ . To calculate χ the formula (2.29) is rearranged to [20]

$$\cos(\chi) = \frac{2 + E - 2(1 + E)^{R_1}}{E}, \quad (3.13)$$

where R_1 is a uniform random number between 0 and 1. With a second uniform random number R_2 the azimuthal angle is determined by [20]

$$\Phi = 2\pi R_2. \quad (3.14)$$

With the information of the scattering angle χ the energy loss of the scattered electron is given by [21]

$$\Delta E = \frac{2m_e}{M} (1 - \cos(\chi)) E, \quad (3.15)$$

where m_e is the electron mass, M the mass of a neutral gas particle, and E the kinetic energy of the electron before the collision. The energy loss is subtracted from the kinetic electron energy and removed from the simulation. Hence, a heating of the neutral gas is not considered.

In an excitation, the threshold energy of the excitation type is subtracted from the electron first. Afterward, the scattering is treated as in the elastic case. The excited neutral is not tracked assuming a short decay time. Depending on the neutral species this treatment may be not accurate due to a relatively long decay time of meta-stable excitation states. Furthermore, the excitation energy is removed from the simulation without any further effect. Unfortunately, there is a great number of excitation states. Therefore, only the most probable will be considered. Incorporating all of these excitation types individually would result in a high computational effort. Therefore, the sum of these excitation types is used as one excitation type. Since a further distinction of the excitation state is not possible, one threshold energy has to be applied for the sum of excitation types. This has to be the smallest threshold energy of all excitation types.

In an ionization, a new electron and a new ion are created by the incident electron. The velocity of the new ion is randomly chosen assuming a Maxwellian

distribution with a given temperature of the neutral gas. The energy of the incident electron E_{inc} is reduced by the ionization energy E_{ion} and has to be partitioned between the incident and the newly created electron. The energy of the new created electron is given by [20]

$$E_{created} = B(E_{inc}) \tan \left[R \arctan \left(\frac{E_{inc} - E_{ion}}{2B(E_{inc})} \right) \right], \quad (3.16)$$

where R is again a uniform random number. $B(E_{inc})$ is determined for Xenon as a constant value of 8.7 eV [22] over an E_{inc} range of 1-70 eV. After determination of the energy of both electrons, the scattering of each electron is executed individually using equations (3.13) and (3.14).

Ion-neutral collision:

For the ion-neutral collision, only the elastic collisions are considered in “PlasmaPIC”. In contrast to electron-neutral scattering the neutral gas cannot be assumed as non-moving. For the collision test the relative velocity of the ion and the neutral particle has to be calculated. Because the neutrals are considered as a background density, the velocity of a pseudo gas particle has to be created using a Maxwellian distribution. Assuming the hard sphere model (2.4) the collision can be described by a uniform and isotropic scattering in the center of mass frame. The required angles for the rotation in the center of mass frame can be determined with [20]

$$\cos \theta = 1 - 2R_1 \quad (3.17)$$

and equation (3.14).

The special type of elastic scattering, the charge-exchange, is treated separately. In this context, the new created velocity is assigned to the ion, reflecting the fact that the neutral becomes an ion.

In “PlasmaPIC” the used cross sections were taken from the plasma simulation tool “Xoopic” [17].

3.2.3 Particle weighting and force interpolation

In a PIC code like “PlasmaPIC” particles have continuous positions, but physical quantities like forces or charge densities are defined at the discrete grid points. The charge of the particles has to be assigned to these grid points. This process is called weighting [18]. A simple weighting is to assign the charge to the nearest grid point (NGP), which corresponds to a zero-order weighting. This is computationally beneficial since only one grid point access and one addition have to be performed. The disadvantage is that the charge of the particle jumps from one grid point to another leading to a noisy charge density. The noise can be compensated by increasing the number of particles leading to more computational effort. Another possibility for noise reduction is the application of a first-order weighting. Here, the particles are linearly weighted to their nearest neighbors. Since there are eight nearest neighbors in a 3D cartesian grid, the computational effort increases by eight grid point look-ups and several additions and multiplications. A higher order of the weighting would further improve the accuracy. However, the computational effort blows up. Hence, it has to be carefully considered if a higher-order is necessary. Like the charge has to be weighted to the grid point, the force at the grid point has to be interpolated to the exact particle position. In this context, it is important to use the same weighting scheme to avoid a non-physical particle self-acceleration [18]. In “PlasmaPIC” a first-order weighting is applied.

3.2.4 Constraints for numerical parameters

In consequence of the numerical treatment the physics is described with a certain precision. The numerical parameters have to be chosen in a way that the numerical error does not have an impact on the physics. A PIC simulation consists of at least three numerical parameters. These are the time step Δt , the grid size Δx , and the ratio of super-particles to real particles. For a stable solution the following constraints for the numerical parameters have to be satisfied.

The size of the time step depends on the largest frequency ω_0 in the simulation. Commonly, this is the electron plasma frequency ω_{pe} . To resolve the oscillation sufficiently the size of each time step is limited by [18]

$$\omega_0 \Delta t < 2. \quad (3.18)$$

In PIC simulations this constraint is in general tightened by [23]

$$\omega_0 \Delta t \leq 0.2 \quad (3.19)$$

to minimize the error especially for simulations with a large number of time steps. For advancing the ions, a larger time step can be chosen, because the ion plasma frequency is considerably smaller.

For a minimal numerical error the grid size should resolve the debye length to describe the particle-particle interaction by the grid quantities accurately [23]

$$\Delta x \leq \lambda_{De}. \quad (3.20)$$

Hockney and Eastwood have shown that $\Delta x < 3.4 \lambda_{De}$ is sufficient for a stable simulation [19].

The ratio of super-particles to real particles determines the number of super-particles in a cell, or the number of particles in the distance of one debye length. To preserve the shielding character the number of particles per one debye distance has to be much larger than one. Hockney and Eastwood proclaimed that a number in the range of 10 to 100 is appropriate [19] whereas the numerical heating reduces with a higher number of particles. In general, the number of particles can be much larger depending on the mechanism that forms the shape of the energy distribution function of the particles [24].

3.3 Derivation of the electromagnetic fields

Solving the whole set of Maxwell equations is very time consuming. First, this has to be done in every time step. Furthermore, this requires to resolve the speed of light, which leads to smaller and hence more time steps. For the modeling of an inductive coupled plasma, as in a RIT, a common approach

is to separate Maxwell equations into an electrostatic [17] and an electrodynamic part [25, 26, 27, 28]. With this technique both parts can be solved independently. The electrical fields from both parts are then added due to the superposition of the fields. This enables to solve the electrodynamic part in “PlasmaPIC” only once per rf-cycle. Another advantage is that both parts can be described by an elliptic equation. Consequently, the same solver can be applied for both parts. Due to these reasons this algorithm is applied in “PlasmaPIC”. In the following the electrostatic as well as the electrodynamic part are described in more detail.

3.3.1 Electrostatic part

In the electrostatic part the Poisson equation

$$\Delta\phi = -\frac{\rho}{\epsilon_0} \quad (3.21)$$

has to be solved in every time step, where ϕ is the potential and ρ the charge. For solution, Dirichlet boundary conditions have to be incorporated. The electrostatic part takes care of the interaction of charged particles and the plasma-wall effects (plasma sheath). Once the potential is determined by the finite-difference technique “SOR”, which will be introduced in section 3.4, the electric field is calculated by the potential difference over two cells [29].

3.3.2 Electrodynamic part

The electrodynamic part is described by the inhomogeneous wave equation also known as Telegraphers equation

$$\left(\Delta - \mu_0\epsilon_0\frac{\partial^2}{\partial t^2}\right)\vec{E}(\vec{r}, t) = \mu_0\frac{\partial}{\partial t}\vec{j}(\vec{r}, t), \quad (3.22)$$

where $\vec{E}(\vec{r}, t)$ is the electric field and $\vec{j}(\vec{r}, t)$ is the current. In the Telegraphers equation charge neutrality is assumed. This approach is applicable because the plasma is quasi-neutral. Although the plasma sheath is not quasi-neutral, the approach still holds, because the sheath is negligibly small compared to the

plasma region. By assuming that the fields and currents are varying harmonically in time, they can be expressed by a complex amplitude

$$\vec{E}(\vec{r}, t) = \vec{\tilde{E}}(\vec{r}) e^{i\omega t}, \quad (3.23)$$

$$\vec{B}(\vec{r}, t) = \vec{\tilde{B}}(\vec{r}) e^{i\omega t}, \text{ and} \quad (3.24)$$

$$\vec{j}(\vec{r}, t) = \vec{\tilde{j}}(\vec{r}) e^{i\omega t}, \quad (3.25)$$

where ω is the applied frequency. Using this in (3.22) the time dependent part can be separated, leading to

$$(\Delta + \mu_0 \epsilon_0 \omega^2) \vec{\tilde{E}}(\vec{r}) = i\omega \vec{\tilde{j}}(\vec{r}), \quad (3.26)$$

which has a spatial dependence only. For the determination of the electric field the current at each position is required. The current $\vec{\tilde{j}}(\vec{r})$ has contributions of the current in the coil $\vec{\tilde{j}}(\vec{r})_{\text{coil}}$ and of the current of the electrons $\vec{\tilde{j}}(\vec{r})_{\text{electrons}}$ in the plasma. The current of the ions is neglected due to their low mobility. In general, there is a phase shift between the coil and the electron current. As a consequence, the electric field is also phase shifted. These phase shifts are considered by the complex amplitude. Furthermore, it is important to mention that equation (3.26) can be separated into six dependent equations, three each for the real and imaginary part.

To solve (3.26), boundary conditions are necessary. These boundary conditions are determined by writing the electric field in terms of the vector potential \vec{A}

$$\vec{\tilde{E}}(\vec{r}, t) = -\frac{\partial}{\partial t} \vec{A}(\vec{r}, t). \quad (3.27)$$

By using the Biot-Savarts law for the vector potential the electric field can be calculated by

$$\vec{\tilde{E}}(\vec{r}) = -i\omega \vec{\tilde{A}}(\vec{r}) = -i\omega \frac{\mu_0}{4\pi} \int \frac{\left(\vec{\tilde{j}}(\vec{r}')_{\text{coil}} + \vec{\tilde{j}}(\vec{r}')_{\text{electrons}} \right)}{|\vec{r} - \vec{r}'|} d^3 r' \quad (3.28)$$

at each point. Of course, this equation could be used for the derivation of the electric field at each point. However, the computational effort is extremely high due to the summation over all grid points of the mesh. Therefore, equa-

tion (3.28) is used for deriving the electric field for boundary grid points and equation (3.26) is applied for the inner grid points.

Once the complex electric field amplitude is calculated, the complex magnetic field amplitude is determined by

$$\vec{B}(\vec{r}) = \frac{i}{\omega} \nabla \times \vec{E}(\vec{r}) . \quad (3.29)$$

Although the complex electric and magnetic amplitude is solved once in each rf-cycle only, the electric and magnetic field has to be calculated in each time step using equation (3.23) and (3.24) and added to the electrostatic part. Furthermore, the complex amplitude of the electron current is derived by a Fourier transformation at the applied frequency for the electron current in each time step.

3.4 Solver for elliptic equations

In ‘‘PlasmaPIC’’ all physical quantities are discretized due to the mesh. This implies the use of finite differences. For solving the Poisson equation (3.21) and the Telegraphers equation (3.26) with the finite difference method the matrix inversion and iterative techniques are applicable. Matrix inversions are fast but difficult to parallelize efficiently. Furthermore, the required memory rises faster than linear for increasing system sizes. For these reasons, the decision was made for an iterative solver. The memory requirements increase proportional to the grid points and the iterative solver can be massively parallelized as will be shown in chapter 4.

Applying the finite-difference scheme to the Poisson equation and solving the potential Φ at the position i, j, k results in

$$\Phi_{i,j,k} = \frac{\Delta x^2 \rho_{i,j,k}}{6\epsilon_0} + \frac{1}{6} (\Phi_{i-1,j,k} + \Phi_{i+1,j,k} + \Phi_{i,j-1,k} + \Phi_{i,j+1,k} + \Phi_{i,j,k-1} + \Phi_{i,j,k+1}) , \quad (3.30)$$

where a uniform rectangular mesh with the same inter-node spacing Δx in each dimension i, j, k is applied. This equation clarifies that the potential $\Phi_{i,j,k}$ depends on the charge density at this grid point and the potential at

the six neighboring grid points. The used finite-difference scheme is of second order.

An iterative solver operates in iteration steps. In each iteration step the potential at each grid point is calculated once. The iteration step is repeated until a given accuracy of the solution is achieved. For performance reasons a Successive over Relaxation (SOR) algorithm was implemented from scratch. In the SOR algorithm, the newly calculated $\Phi_{i,j,k}$ is mixed with the $\Phi_{i,j,k}^{m-1}$ of the last iteration step to determine the

$$\Phi_{i,j,k}^m = (1 - \omega) \Phi_{i,j,k}^{m-1} + \omega \Phi_{i,j,k} \quad (3.31)$$

of this iteration step where ω is the mixing parameter. For ω between 0 and 2 a convergence is guaranteed. For $\omega < 1$ an under-relaxation is achieved whereas for $\omega > 1$ an over-relaxation is accomplished, which leads to a faster convergence. Furthermore, it can be shown that an optimal ω_{opt} exists depending on the number N of grid points in one direction [30]

$$\omega_{\text{opt}} = \frac{2}{1 + \sin(\pi/(N - 1))}. \quad (3.32)$$

3.5 Geometry representation

As already mentioned, “PlasmaPIC” can handle arbitrarily shaped 3D objects. These objects can be designed by a standard CAD (computer aided design) tool. Using the export function, the surface of every object is represented by triangles. After importing the triangles in “PlasmaPIC”, the triangles are used to identify which cells of the simulation grid are occupied by the object. By marking these cells the object is reproduced in “PlasmaPIC”. That way the shape of the object is cornered. Unfortunately, the surface is represented stepwise and hence the smoothness of the surface is lost. Nevertheless, this is important to reduce the computational effort. Furthermore, this procedure of marking the cells needs to be done only once at the beginning.

The marking of the cells occupied by an object is done in the following way. For exporting the objects the stl file format is used. Hereby, the surfaces of the object are represented by triangles. Only objects with a volume are allowed (and no single 2D planes). The normal vector of each triangle is oriented in

the way that they point out of the volume of the object. As a consequence every point is determined as being inside or outside the object. For this task an arbitrary ray, which starts from the point of interest, has to be tested for an intersection with every triangle. The intersection test is applied by running the ray tracing algorithm [31] used in “FlowSim” [32].

In case of no intersection the point lies outside the object. If there is at least one intersection, only the first intersection needs to be considered in the following. Treating the intersected triangle as a plane, it has to be checked on which side of the plane the point of interest is located. This information is obtained by the scalar product of the ray and the normal vector of the intersected triangle. The earlier mentioned definition of the normal vector of the triangle allows determining whether the point lies inside or outside the object. This algorithm needs to be conducted for every point of the simulation grid. The eight neighboring simulation cells of each point located in the object have to be marked. If the smallest object diameter is smaller than the grid spacing, the test has to be performed on a finer grid. Afterward, the finer grid has to be extrapolated to the simulation grid. With this method the maximum error of the object representation is smaller than twice the cell diagonal.

“PlasmaPIC” supports the importing of several objects holding different properties. In the following section pre-defined types of objects are described. In “PlasmaPIC” the coil is represented by a current string whose route is given by a function. Figure 3.3 shows a CAD model of a μ N-RIT 1.5 and its representation in “PlasmaPIC”.

3.6 Supported boundary types

In this section the different types of boundaries supported by “PlasmaPIC” will be introduced. A short explanation of their physical behavior and their treatment in “PlasmaPIC” is provided.

Dielectric:

A dielectric object is an insulator. Once an electron or ion strikes the surface the particle sticks at that position of the surface. Therefore this position of the surface can be charged by these particles. For every dielectric object a specific permittivity is considered and can be defined by the user.

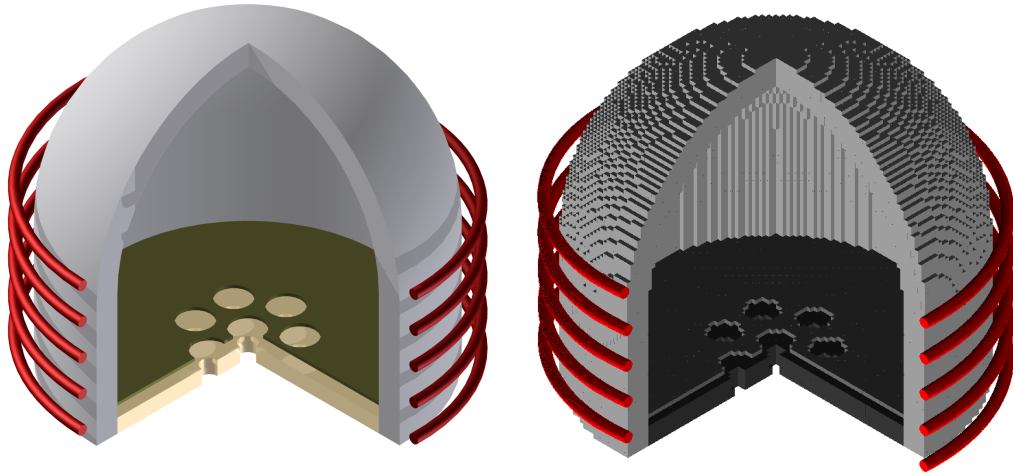


Figure 3.3: CAD model of $\mu\text{N-RIT 1.5}$ (left) and its representation in “PlasmaPIC” (right).

Exit:

In contrast to a dielectric object, the permittivity of an exit is always one. Striking particles are removed from the simulation. The surface won’t be charged by the particles.

The following three types are considered as perfectly conducting.

Equipotential:

This class of objects has a fixed potential that cannot be altered. So the particles will be removed once they struck these objects without charging the object. It is assumed that any quantity of current can drain off instantaneously. This treatment corresponds to an ideal voltage source.

Conductor:

In contrast to an equipotential object, no current can drain off. Striking particles are able to charge the object. Due to the perfect conduction each object has one charge in total. Thus, all cells within the object obtain the same potential.

Current driven:

A current driven object is considered as an ideal current source. The quantity of the current can be specified. Depending on the current of striking particles compared to the specified current, charging the object is possible. The potential of the object will be adjusted accordingly.

Plasma region:

By importing a geometry consisting of different objects the area containing plasma is not defined. For this reason an additional object is needed. All cells occupied by the plasma region object are allowed to hold plasma.

For all described types except the plasma region, a constant value for the secondary electron emission probability can be specified in the input card.

Chapter 4

Massive Parallelization

The last chapter has shown that plasma simulations using the PIC method require an enormous computational effort and even more effort is needed for 3D PIC for system sizes of our interest. To perform simulations in appropriate time an increased computational power is necessary. Due to the computational power limitation of only one arithmetic unit, several arithmetic units need to be interconnected and the computation needs to be split among the arithmetic units. This process is called parallelization. Massive parallelization means the number of interconnected arithmetic units is much larger than 100. By splitting one large process into smaller sub-processes and sharing them among the arithmetic units, a linear time reduction for the computation is expected. Indeed this theoretical upper boundary is hard to reach, because parallelization requires communication between the arithmetic units at all times. The required total communication cost scales with the number of processes. In most cases the communication cost exceeds the additional computational power at a certain amount of interconnected arithmetic units. In this case the speed-up of the simulation will even decrease. Depending on the task and the kind of parallelization the communication cost between two processes can become very large. In this case the parallelization with only a few arithmetic units will be inefficient and a massive parallelization is impossible. Therefore, a focus in parallelization is on the communication between the arithmetic units.

Another important point is an equal splitting of the computations among the sub-processes. This is necessary because all sub-processes have to wait until the sub-process with the most computations has finished. These idle times have

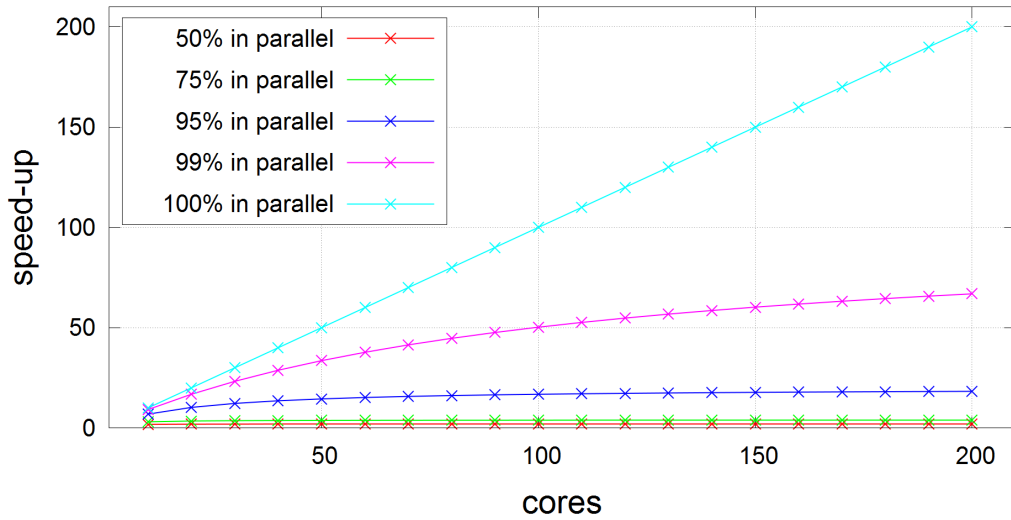


Figure 4.1: Speed-up values assuming an optimal parallelization for different levels of parallelization of the process.

to be avoided or at least minimized for effective parallelization. In some cases the computational load can change over time. For example, the number of particles in a simulation will vary. Therefore, the splitting should be steadily re-adjusted for best performance. Especially for a massive parallelization the interconnected arithmetic units can have different performance values. In such a heterogeneous scenario the splitting must be adapted as well.

So far it was assumed that every task can be done in parallel. In general, this is not the case. Calculations that directly depend on results of another calculation cannot be parallelized. A simulation consists of a lot of different tasks. As soon as only one task does not fulfill the requirements of parallelization or isn't parallelized the total speed-up of such a simulation will be influenced dramatically as can be seen in figure 4.1. Even if 99% of all calculations are parallelized the speed-up has an upper boundary of 100. This number is intolerable, in particular for a massive parallelization. Taking this into account every task in a simulation must be parallelized, as long as it is possible, independently of the programming effort.

Unfortunately, there is no general automatism for the parallelization of a calculation. However, there are a lot of parallelization libraries, which provide basic functions for parallelization. To find the best choice for the plasma simu-

lation using PIC, three parallelization libraries are compared in the following chapter.

4.1 Comparison of the Parallelization Libraries

Each parallelization library is adapted to a specific hardware architecture of the arithmetic unit and their interconnections. They provide functions for the parallelization. These functions handle the communication at least on the hardware level but for some cases also on the software level. On the following pages the functionality of three parallelization libraries is described including their advantages and disadvantages. Moreover, the solver for the Poisson equation is parallelized for all three kinds and compared to a serial version. This analysis is constrained to the field solver because that is the task with the highest communication effort and therefore the hardest part within the parallelization of a plasma simulation. Hence, the field solver forces the choice of the parallelization library, and, in consequence, the choice of hardware architecture.

4.1.1 OpenMP

“Open Multi-Processing” (OpenMP) has been in development since 1997 by various companies. It is based on shared memory and hence suitable for a multiple CPUs system with each CPU consisting of several cores. Due to the shared memory these CPUs have to be located on the same mainboard. For that reason the number of cores is limited to a certain amount, which is about 32 at the moment. As a result there are low communication times because every process can directly read all the data from the memory without any communication among the processes. Communication is only needed for writing data at the same memory position and synchronizing, e.g. waiting of the processes. The parallelization is done by preprocessor commands splitting loop runs among a certain number of arithmetic units. Therefore, the programming effort is quite small and the achievable speed-up is near the theoretical one, because of low communication. The disadvantage of splitting loop runs arises from the fact that each OpenMP process must execute the same section of code at a given time. This reduces the flexibility to do different tasks simulta-

```

for (int z=0; z < z_max; z++)
for (int y=0; y < y_max; y++)
for (int x=((z+y+redblack) mod 2); x < x_max; x=x+2)
{
    \\ calculation of phi[x,y,z]
}

```

Listing 4.1: Field solver using SOR.

neously. Concerning the field solver this kind should match quite well because the used algorithm consists of three nested loops, one for each dimension that is demonstrated in listing 4.1.

In total, there are $x_max \times y_max \times z_max$ loop runs. Using OpenMP these runs will be distributed among a predefined number of processes. For nested loops there are two possibilities for parallelization in OpenMP. The first is to parallelize the outermost loop, which is called “simple version” in the following. Figure 4.2 shows the speed-up for different numbers of used cores and different system sizes compared to one core. With increasing system size the speed-up deviates more and more from the optimum speed-up. This happens because the computational effort is quite small and the calculation speed is determined by the access speed of the memory and its bandwidth. With increasing system sizes, a smaller percentage of the total data can be stored in the CPU cache, which has a large bandwidth and fast access speed. Consequently, more access is required to the random access memory which has a smaller bandwidth and a slower access time.

For larger systems it is necessary to optimize the order of traversing the three nested loops in such a way that the probability to catch up data in the CPU cache increases. For this purpose OpenMP has the collapse statement, which is the second possibility to parallelize nested loops. This collapse statement takes care of an optimal traversing of the three nested loops. Nevertheless, in that case all three loops have to be independent of each other and statements are only allowed in the innermost loop. Therefore, the calculation of the red-black variable has to be moved inside the innermost loop. Figure 4.3 shows the speed-up of that version related to the result for one core in the simple version. All speed-up values are much smaller than the values of the simple version. This is the result of moving the modulo operation inside the

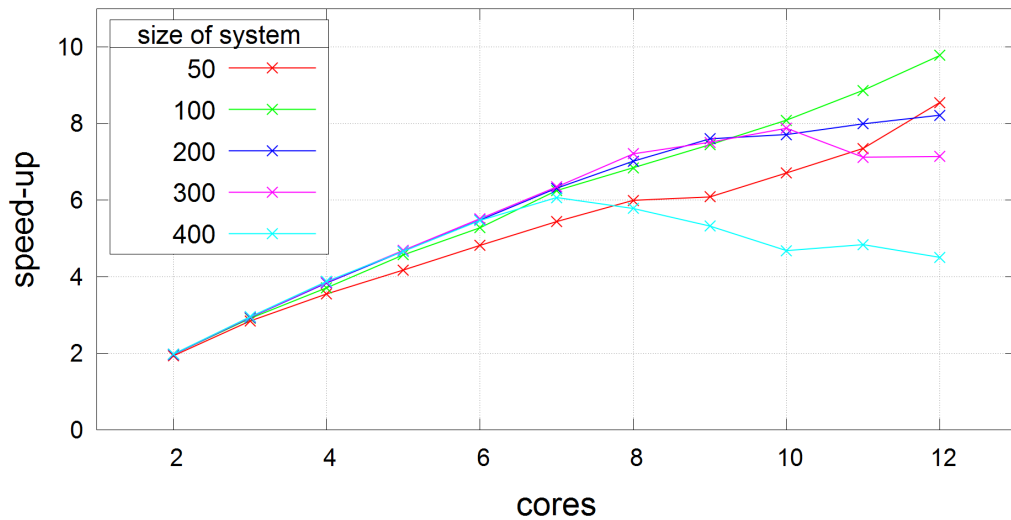


Figure 4.2: Speed-up for the parallelization over the outermost loop, where the size of system denotes the number of grid points in each dimension.

innermost loop, which increases the number of needed modulo operations by a factor of three. Nevertheless, the speed-up values for all system sizes are not saturated and can still increase. As a consequence, with more available cores the collapse version could beat the simple version. This example shows that beside communication cost also additional calculation cost has to be considered for an efficient parallelization.

Another possibility to optimize the memory access is by manually adjusting the memory access. This can be done as shown in listing 4.2. In this variant the parallelization is done by the collapse statement over the three outer loops. The speed-up values for this version are shown in figure 4.4. The enhancement of the speed-up values for large systems amounts to a factor of almost two. Unfortunately, the speed-up values for smaller system sizes are worse due to the additional overhead of the block calculations, which dominates the calculations for smaller systems. All speed-up comparisons were performed on a two CPU system. Both CPUs are Intel(R) Xeon(R) CPU E5-2620 consisting of six cores each.

An important result of this chapter is that it is always necessary to weight the parallelization cost against the achievable performance gain. There is no general solution for the best way of the parallelization of a specific algorithm.

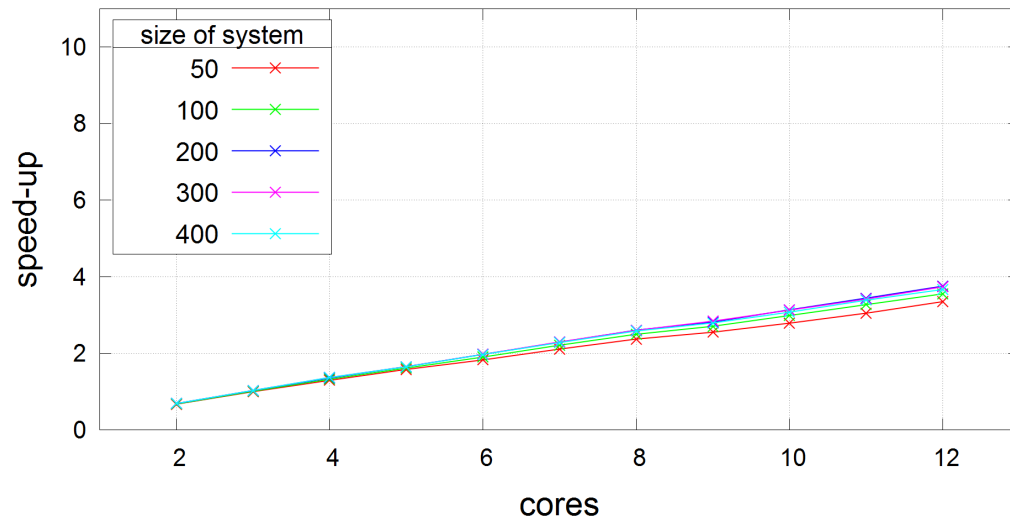


Figure 4.3: Speed-up for the collapse statement related to one core of simple version, whereas the size of system denotes the number of grid points in each dimension

```

for (int z_block=0; z_blockz < blocks_in_z; z_block++)
for (int y_block=0; y_blocky < blocks_in_y; y_block++)
for (int x_block=0; x_blockx < blocks_in_x; x_block++)
{
    for (int z=z_block*size; z < (z_block+1)*size; z++)
    for (int y=y_block*size; y < (y_block+1)*size; y++)
    for (int x=x_block*size+(z+y+redblack) mod 2;
        x < (x_block+1)*size; x++)
    {
        \\ calculation of phi[x,y,z]
    }
}

```

Listing 4.2: Manually optimized field solver using SOR.

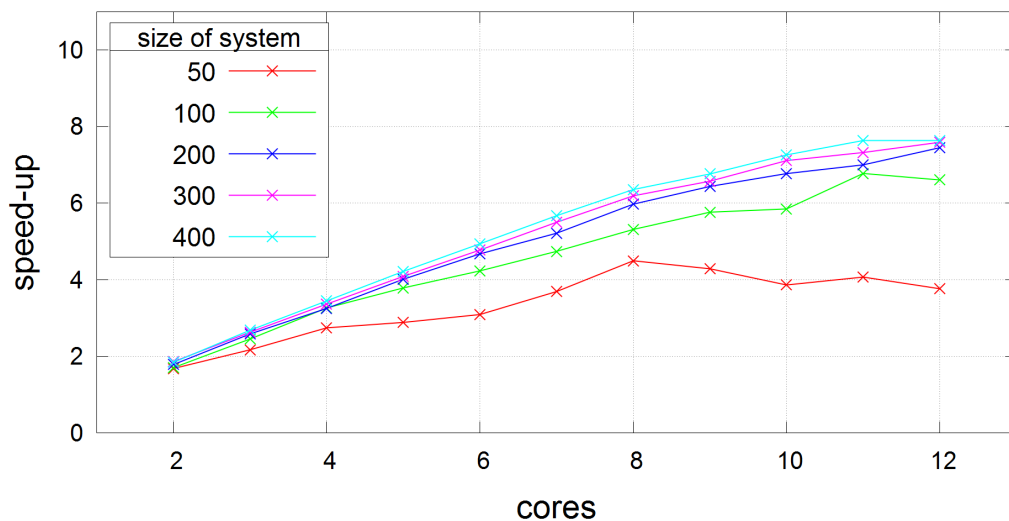


Figure 4.4: Speed-up for the manually optimized solver related to one core of simple version, where the size of system denotes the number of grid points in each dimension.

As shown, it depends on the number of used cores as well as the size of the system.

4.1.2 MPI with Domain Decomposition

Another parallelization option is offered by the “Message Passing Interface” (MPI) library which has been in development since 1992. Applying this method multiple processes are created. They are operating independently and can exchange information by communicating over the network protocol. The advantages of MPI are as follows. By interconnecting the processes over the network the number of usable cores is not limited and the total available memory is increased. Furthermore, the execution of individual tasks is possible and leads to higher flexibility. The great disadvantage lies in the arising overhead for the network communication between the processes. This overhead depends on the amount of communication data, and on the latency and bandwidth of the present kind of network. The latency as well as the bandwidth are decisively better with the infiniband. A comparison of both networks is shown in table 4.1.

Table 4.1: Comparison of gigabit and infiniband network. Values for latency were measured with ping command.

	gigabit ethernet	infiniband
data transfer rate	1 Gb/s	10 Gb/s
latency	0.4 μ s	0.1 μ s

To reduce the communication overhead MPI permits the functionality of non-blocking communication. This means that each process can do the communication simultaneously to other calculations if these calculations are independent of the communication data. Performing non-blocking communication, the needed communication time can be partially or completely hidden.

A common process hierarchy is the master-slave concept. One master distributes the tasks and data to the slaves and collects the results afterward. That allows an even balance of the tasks among the slaves by reducing the programming effort. Unfortunately, this leads to a high communication and memory load of the master. Especially, if the communication demand is quite high (such as for the “SOR” field solver) the domain decomposition is a more appropriate parallelization strategy.

In this strategy the simulation space is divided into sub-spaces. Each sub-space forms one process. All processes are equal and each holds only the information belonging to its sub-space. Thereby, the needed memory per process reduces with the number of sub-spaces. Most calculations in one sub-space can be done independently of the other processes. Communication is only needed between neighboring sub-spaces and only for their area of contact. The sub-space division can be done along one, two, and three axes in 3D. With an increasing number of parallelization axes the programming effort increases extensively. This is related to the quantity of maximum neighbors of a sub-space. Performing parallelization along only one axis, there are up to two neighbors, along two axes there are eight neighbors, and along three axes there are 26 neighbors. Each communication direction has to be considered individually. The increased programming effort is absolutely necessary for a massive parallelization. Without splitting along three axes, the surface-to-volume ratio of a sub-space gets worse for an increased number of sub-spaces, which leads to more communication cost. These communication costs rapidly excel the

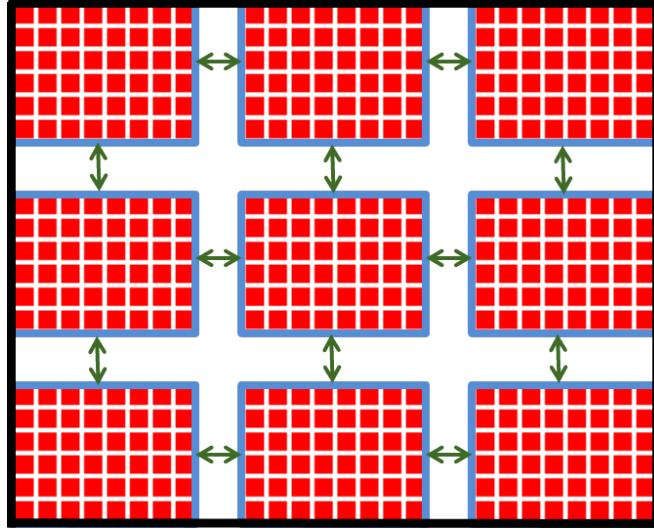


Figure 4.5: Domain decomposition in 2D.

advantage of more computational power. Therefore, an optimal surface-to-volume ratio is inevitable. Figure 4.5 shows the domain decomposition for a 2D case because of the better comprehensibility.

To achieve optimal parallelization performance for this strategy the non-blocking communication is applied. For this task the order of the calculation of the grid points is changed. To calculate the surface grid points of a sub-space the surface grid points of the neighboring sub-spaces are required. Therefore, the surface grid points are sent to the neighboring sub-spaces by the non-blocking communication first. During the send and receive process the inner grid points, are calculated. After the calculation of the inner grid points the surface grid points have to be calculated. If at this moment the communication is finished, no further waiting is necessary. In that case the communication is perfectly hidden. Otherwise the thread has to wait until the communication is finished, which considerably slows down the computation. An optimal surface-to-volume ratio is achieved if the communication time of the surface grid points is equal to the calculation time of the inner grid points. Of course this is not a constant value. It depends on the computational power, CPU cache size, bandwidth to the random access memory, latency and bandwidth of the network connection.

This MPI variant was tested for two cases. The first one was the loopback interface. In that case no real network communication was required, because

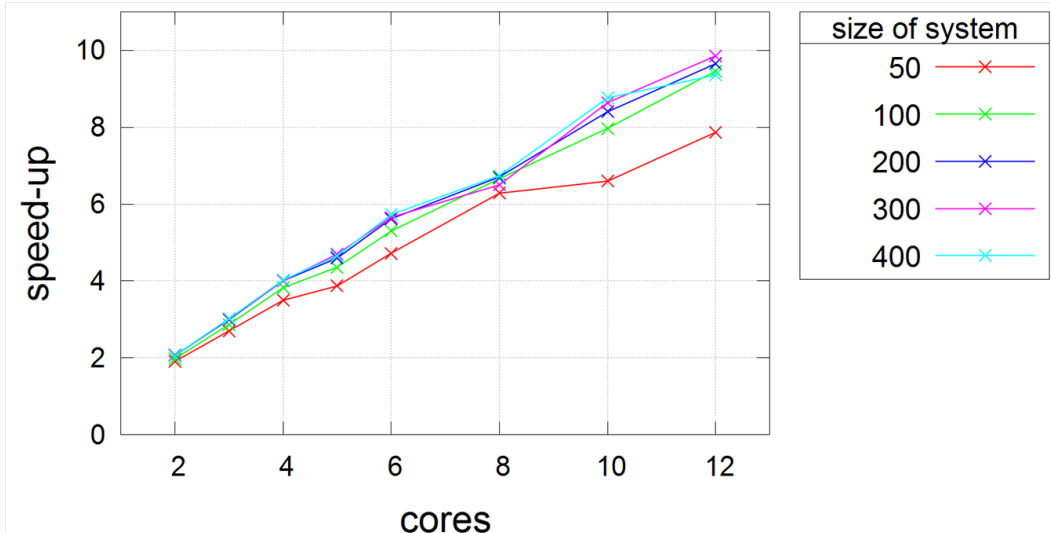


Figure 4.6: Speed-up of the MPI variant utilizing the loopback device, where the size of system denotes the number of grid points in each dimension.

all threads were located on the same mainboard. The speed-up is shown in figure 4.6. For all system sizes except the system size of 50 a very good parallelization can be observed. The underperformance of the system size of 50 results from the fact that this small system completely fit into the cache of one CPU and thus the computation time is much slower than the communication time between the CPUs.

The second and quite more interesting case is the utilization of the infiniband. Hereby, communication has to be done over the network resulting in increased communication cost. Figure 4.7 shows the speed-up values related to one CPU core. For small systems the speed-up values reach a saturation even for a few cores. With increased system size the surface-to-volume ratio increases and good speed-up values can be achieved up to hundreds of cores. Indeed, the speed-up values for large systems of at least 300 underperform for a number of cores less than 200. Above 200 cores the speed-up escalates for the system sizes of 300 and 350. This behavior cannot be explained with certainty, but is most likely linked to the CPU cache. For large systems an additional abrupt rise of the calculation time can be achieved if all data of grid points fit into the CPU cache. In this case access to the random access memory is

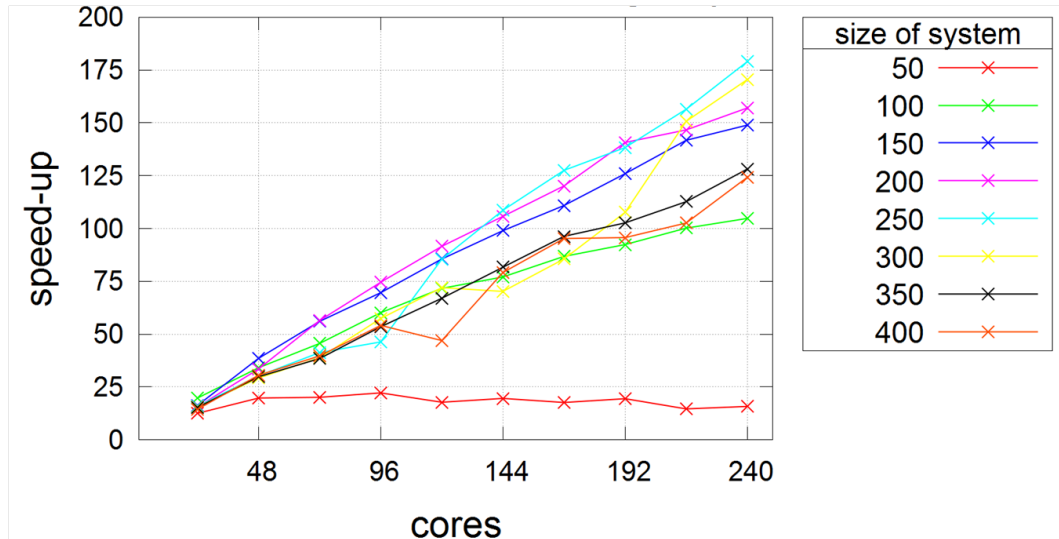


Figure 4.7: Speed-up of the MPI variant utilizing the infiniband, where the size of system denotes the number of grid points in each dimension.

necessary only for the surface grid points, which are communicated over the network.

4.1.3 Cuda

“Compute Unified Device Architecture” (CUDA[®]) developed by NVIDIA Corporation extremely differs from the already mentioned parallelization strategies. This is because it is based on a completely different hardware architecture. Instead of using a CPU as processing unit, a “graphic processing unit” (GPU), also known as graphic card, performs the calculations. Figure 4.8 and table 4.2 compare the GPUs and CPU used in this work and demonstrate the differences. The main statement of figure 4.8 is that a CPU and GPU consist more or less of the same quantity of transistors, as can be seen in table 4.2, but with different assignments. In the GPU each row forms a multiprocessor (MP) with a small cache and control unit compared to the single core CPU on the left-hand side. In return the multiprocessor of the GPU holds much more arithmetic logic units (ALU) also called CUDA core in Nvidia terms. Due to this the theoretical computational power of a GPU (table 4.2) beats any CPU by far. Furthermore, the bandwidth to the RAM is enlarged to supply the fast calculating MPs with sufficient data. However, all CUDA cores of a

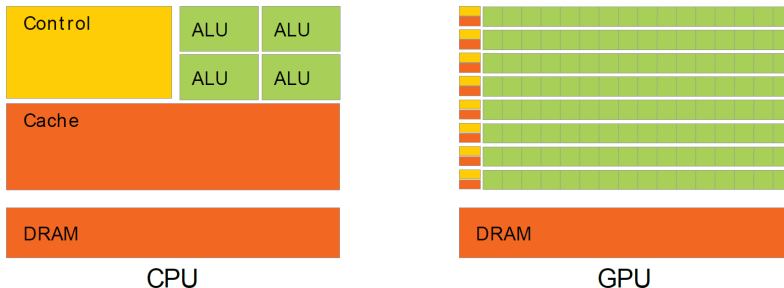


Figure 4.8: CPU and GPU hardware architecture (from [33]).

Table 4.2: Comparison of used NVIDIA GPUs and Intel CPU (values from www.nvidia.com, www.intel.com, and www.pcgameshardware.de)

	GPU		CPU
	GTX 460	Kepler K20	2×E5-2620
transistors [10^6]	1950	7100	2×2270
cores	384	2496	2×6
single precision [Gflops]	907	3520	2×96
double precision [Gflops]	76	1170	2×96
memory bandwidth [GB/s]	115.2	208	2×42.6
peak power [watt]	160	225	2×95
price [euro]	~ 150	~ 2500	~ 400
achieved usage for system size of 400 [Gflops]	11.25	11.83	7.49
achieved Gflops per euro	0.075	0.005	0.009
achieved Gflops per watt	0.070	0.053	0.039

multiprocessor or a CPU core can only execute the same instructions but utilizing various data, which is called “single instruction multiple data” (SIMD). For that reason, the GPU will always outperform the CPU for large vector or matrix operations. For a lot of dissimilar and small tasks the CPU has the edge over the GPU. Additionally, the CPU consists of a larger hardware instruction set than the GPU.

The NVIDIA Kepler K20 is composed of 13 multiprocessors each holding 192 CUDA cores that sum up to 2496 CUDA cores. The difficulty to utilize the capacity of a GPU lies in the division of a process into thousands of sub-processes called threads. Therefore, it is essential to vectorize an algorithm to get a full load of a GPU. Although the theoretical computational power of a GPU is up to 20 times larger than a CPU, there are additional limitations to consider.

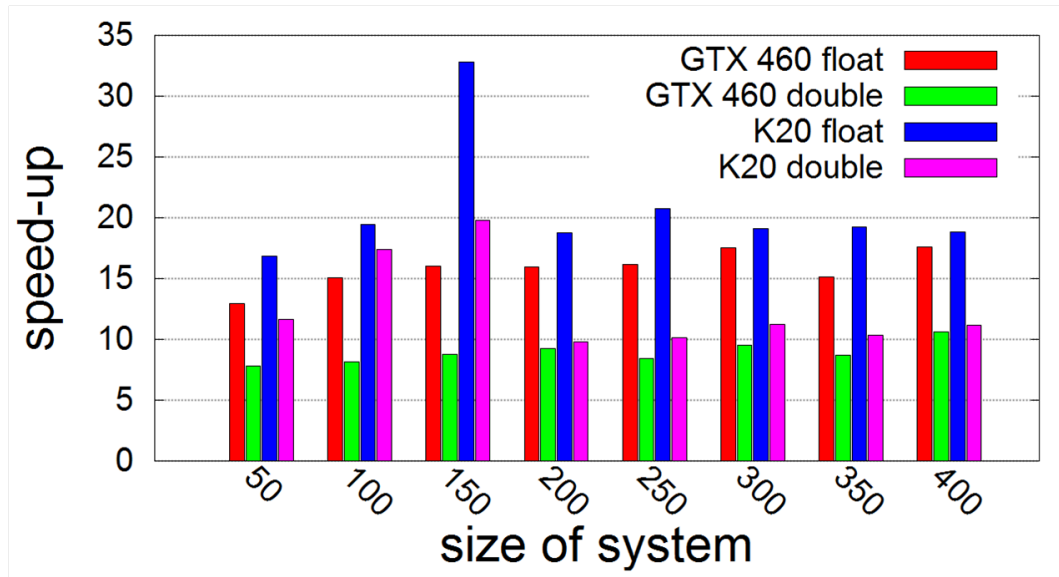


Figure 4.9: Speed-up values for single and double precision at two different GPUs related to the performance of one CPU core. The system size denotes the number of grid points in each dimension.

First of all, the main program has to be launched on a CPU. Tasks can now be outsourced onto the GPU. Because the threads on the GPU can only access the memory of the GPU, additional time is required to transfer data between CPU and GPU. On that account it is important to trade off the transfer time against the computational speed-up. This balance can be considerably increased by operating all tasks of the program on the GPU. Unfortunately, the GPU memory is significantly smaller than the CPU memory. Furthermore, not every task can be efficiently done on a GPU as mentioned earlier.

The iterative field solver SOR fits quite well to the aspect of the utilization of thousands of threads. This is because every point of the potential can be assigned to a single thread and calculated by it. Figure 4.9 shows the speed-up of two different GPUs compared with a non-parallelized single CPU core version for single and double precision. For single precision, speed-up values of at least 15 are achieved for all system sizes larger than 50, and for double precision, speed-up values above 7. The speed-up values for the system size of 50 underperform, because the number of grid points and therefore the number of created threads are too small to get a full utilization of the GPU.

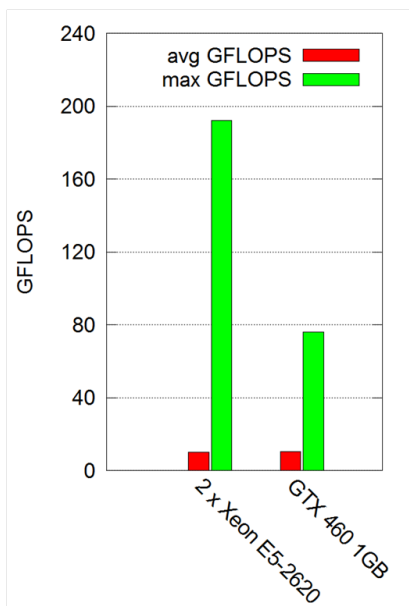


Figure 4.10: Comparison of the maximum computational power and the achieved utilization.

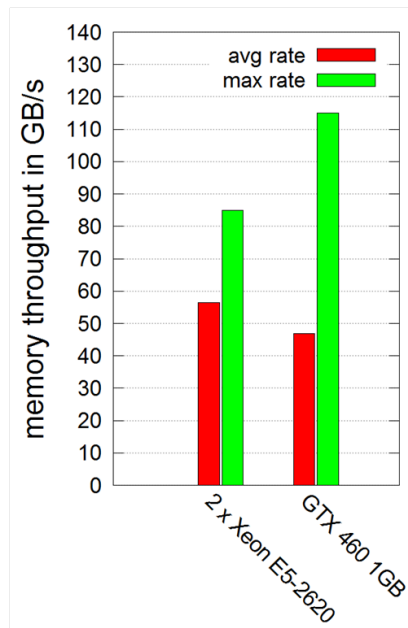


Figure 4.11: Comparison of the maximum memory bandwidth and the achieved utilization.

It is suspicious that the speed-up difference between single and double precision is about a factor of two for all systems, whereas the theoretical computational power for single and double precision at the GTX 460 differs in a factor of 12. This indicates that the computational power is not the limiting factor. Indeed, figure 4.10 reveals that less than 10% of the theoretical computational power is used. The reason is that a lot of GPU memory accesses are required that are very time consuming. Figure 4.11 shows that the average utilization of the bandwidth is at around 50%. Therefore, the GPU CUDA cores are idle and waiting for data most of the time. By changing from double precision to single precision the same amount of grid points requires half of the memory. Consequently, the double amount of grid points can be transferred from the GPU memory to the CUDA cores in the same time. On that account the computational speed-up between single and double precision is a factor of two for all systems and any GPU.

The most important observation of this speed-up comparison is that the speed-up difference between the two GPUs is extremely slight while the theoretical computational power differs by a factor of about four for single precision and a factor of 15 for double precision. This has two reasons. First the band-

width is only increased by a factor of roughly two and as already mentioned this is the limiting factor for this kind of used field solver. But this still implies a factor of two in the speed-up increase. However, this can only be observed for a system size of 150 (see fig. 4.9). A much more decisive reason for the negligible speed-up difference between both GPUs results from the change in the hardware architecture of different GPU generations. For example the number of total threads or the method of memory access can vary. Furthermore, new compiler features are available to achieve a better GPU utilization. All these things have to be considered in the source code to get a good utilization. In particular, the source code has to be readjusted for every new generation of GPU. Depending on the changes in the hardware architecture and their influence on the special task to perform the source code modification can be done by a simple adjustment of constants or by a complete revision of the source code. Normally, this kind of readjustment should be hidden from the programmer level by the compiler. For the CPU this works quite well, whereas it is quite complicated to do this with the GPU due to the massive parallelization that is needed in vector calculators.

The used field solver algorithm was specially adapted to fit the GTX 460, because this was at that time the only available GPU for testing. There were no changes performed to adapt it to the new K20. The increase of the speed-up at the system size of 150, shown in figure 4.9, was some kind of random fit for this special system size to the architecture. Moreover, the lack of speed-up using the K20 in comparison to the GTX 460 emphasizes the need to reprogram the algorithm for the K20.

In conclusion, the speed-up of interest for double precision is on average slightly better than the 12 core CPU system presented in the OpenMP and MPI section.

4.2 Choice of parallelization

For the choice of the parallelization type one can simply compare the ratio of theoretical computational power to price but this is not sufficient. Even taking into account the ratio of theoretical computational power to electrical power is too general. It is important to test the suitability of the algorithm

for the parallelization type and compare the achieved computational power to the price and electrical power. Table 4.2 shows that the GTX 460 has the best values. In this comparison the system price of the required workstation was neglected as well as the power consumption of the workstation. At a workstation price of 400 euro and approximately 200 W the values for the GTX 460 are most negative influenced. Furthermore, the programming effort has to be considered, which is hard to measure. In particular, this also includes the programming effort which is necessary for an adaption of the source code to a new generation of hardware. Especially for GPUs this is a complicated task as shown in the last chapter. However, the most important criterion in the case of a PIC plasma simulation is the scalability. This means the highest speed-up value that is efficient. Only with this the calculation time can be reduced from months to a few hours.

It was shown that the MPI variant is the best one for the field solver. Due to the increasing communication effort in the MPI variant that result was not predictable. This was possible by the efficient hiding of the communication described in section 4.1.2. An additional advantage of the MPI variant is the low needed random access memory due to the domain decomposition. In this way no programming time was necessary to minimize the used random access memory and a huge amount of physical values can be tracked in a plasma simulation. In the following chapter the MPI parallelization variant is applied to the entire plasma simulation.

4.3 Parallelization of the entire plasma simulation

As mentioned at the beginning of this chapter a parallelization of every task in a simulation is essential for a high scalability. So far only the parallelization of the field solver was analyzed, because the field solver is the part with the most communication in a PIC plasma simulation and hence determines the parallelization strategy. Because of the impressive speed-up values of the MPI variant the MPI parallelization was applied to the entire plasma simulation. Besides the calculations of the fields there are several particle operations to perform, namely particle weighting, moving, and collisions. Since the domain

decomposition is used for the field solver, each thread has only the information of the fields inside its domain. Hence, a parallelization by a random distribution of the particles to each thread would require to transfer all field information and hence a larger memory demand for all threads. Consequently, the domain decomposition strategy is applied to all other tasks. Therefore, each thread holds and operates only the particles that are located in its sub-space. If a particle is leaving its sub-space it has to be sent to the neighboring thread, which operates the new sub-space of the particle.

Several grid quantities at the sub-space borders have to be exchanged between neighboring threads like the charge density. For this purpose the functions of the field solver parallelization can be used. Moreover, the field solver can be applied for the Poisson equation as well as the Telegraphers equation.

The following analysis is based on a proceeding published at the “33rd International Electric Propulsion Conference” [28]. The speed-up of the entire 3D PIC plasma simulation is shown in figure 4.12. The speed-up is related to one core in this comparison and is split into particle operations and field solver operations. Additionally, the expected maximum speed-up is shown that results from the fact that 10 times more cores have 10 times more computational power and thus the speed-up should increase by a factor of 10 only in the ideal case of a perfect parallelization. Therefore, it is suspicious that the speed-up of the particle mover is obviously larger than the expected maximum speed-up. This contradiction can be explained by the hardware architecture of the CPU and the algorithm design. Most particle operations have a higher demand of memory transactions than computational effort. For that reason, the limiting factor for the speed-up is the bandwidth of the memory and not the computational power. By increasing the number of sub-spaces the size of a sub-space will shrink. As a consequence more sub-space information will reside on the CPU cache, which has a larger memory bandwidth. Consequently, a better performance of the particle operations can be facilitated compared to the non-parallelized version. Of course this can also be achieved by optimizing the serial version. For example, weighting the particles to the grid has a chaotic disordered memory access. By some kind of presorting of the particles, the memory access can be considerably improved due to a better usage of the CPU cache. Yet the presorting requires additional calculation time. However,

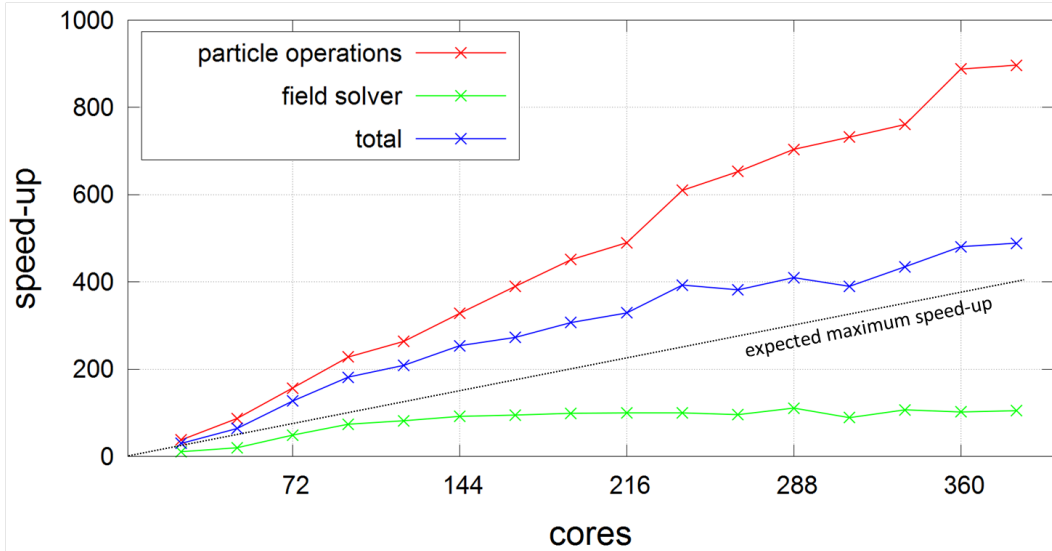


Figure 4.12: Speed-up of a plasma simulation compared to a single CPU core with a system size of $100 \times 100 \times 100$ grid points and $3 \cdot 10^7$ electrons and ions.

presorting is not required in a massively parallelized PIC plasma simulation, because a better CPU cache usage is achieved by reducing the system size of a process. The CPU cache usage optimization is done automatically by the parallelization without any additional programming or computational effort.

The speed-up of the field solver is not as good as that of the particle mover. It is limited to a maximum speed-up of 100. This is related to the increasing number of cells located on sub-space boundaries compared to the cells inside a sub-space. As a result the communication between the threads increases while the calculation time for inner cells decreases. Nonetheless, it is remarkable that the speed-up of the field solver does not decrease.

Since the computational effort of the field solver and the particle mover are comparable, an excellent speed-up of the total simulation is obtained.

Simply checking whether the total speed-up compared to one core is above the expected maximum speed-up is not sufficient. Indeed, a speed-up value above the maximum expected speed-up, which still increases, will reduce the computation time but the most efficient speed-up is the one with the highest slope of its connecting line to the origin of coordinates, which is shown in figure 4.13. In this specific case the most efficient value is reached at 96 cores. Figure 4.14 shows the renormalization of the speed-up of figure 4.12 to 96 cores.

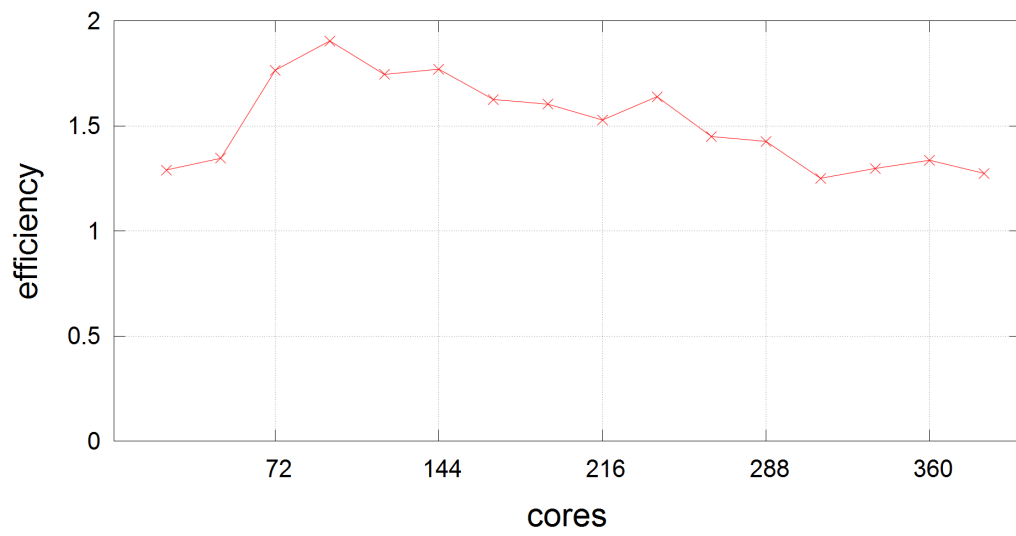


Figure 4.13: Efficiency of parallelization for speed-up values in figure 4.12.

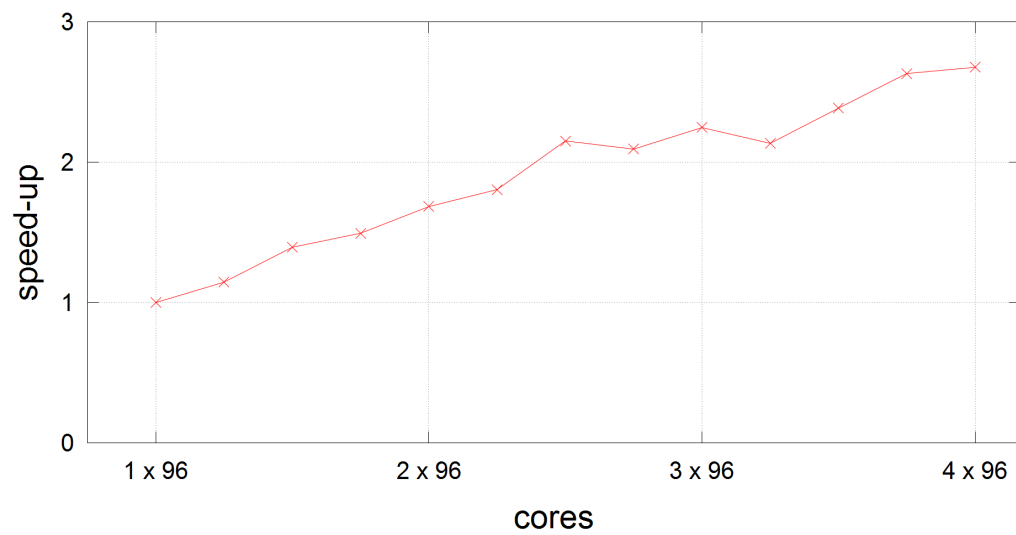


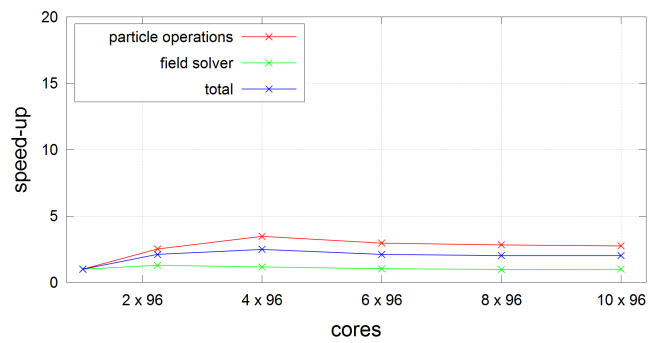
Figure 4.14: Speed-up values of figure 4.12 related to 96 cores.

Table 4.3: Comparison of the four RITs.

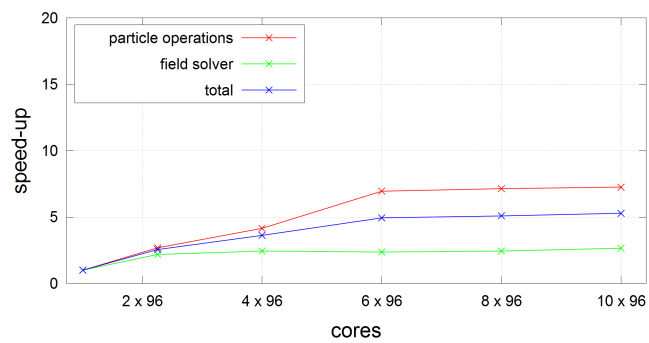
	RIT 1.0	RIT 1.5	RIT 2.0	RIT 2.5
volume increase related to RIT 1.0	1.00	2.96	6.56	12.00
calculation time 96 cores [s]	642	2401	8432	20939
shortest calculation time [s]	258	454	818	1238
used cores	384	960	960	960

As mentioned in 4.1.2 the parallelization efficiency of the field solver depends on the ratio of grid points located on the surface to the number of grid point inside the sub-space. Similar assumptions can be done for the particle operations, where in addition the communication is less compared to the field solver. This means that the surface-to-volume ration has to be optimized for the field solver only. To examine the influence of the system size on the parallelization speed-up four different system sizes were compared, namely a RIT 1.0, RIT 1.5, RIT 2.0, and RIT 2.5. The geometry of the RIT was scaled up without changing any other physical or numerical values except the value for ω of the field solver to have an optimal convergence (see section 3.4).

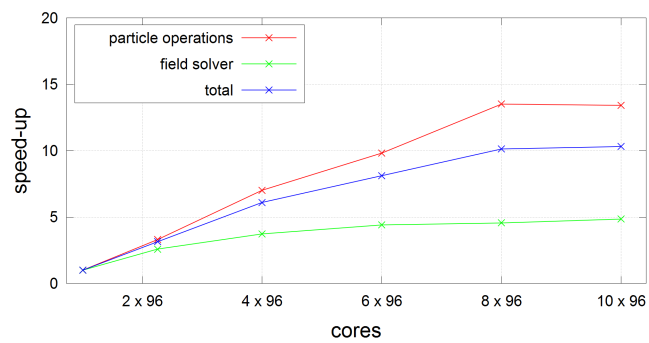
Figure 4.15 shows the speed-up values of the four systems compared to 96 cores of each system. It can be seen that with increasing system size the speed-up values are improving. The smaller the system size, the earlier the speed-up values reach saturation. In table 4.3 it is shown that the calculation time of a system increases faster than the system size. This is, because on the one hand the computational effort of the field solver is stronger than linear. On the other hand the cache optimization is worse for larger systems at the same number of cores.



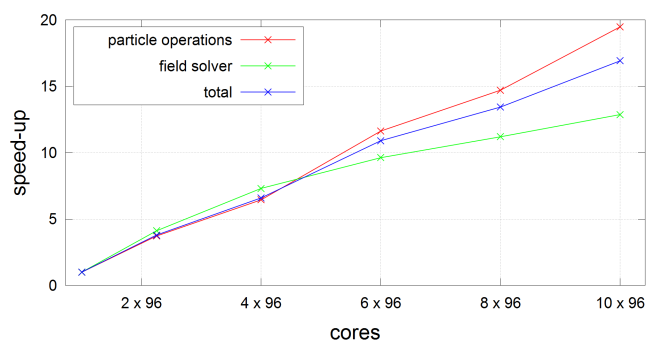
(a) RIT 1.0



(b) RIT 1.5



(c) RIT 2.0



(d) RIT 2.5

Figure 4.15: Speed-up values for four differently sized RITs.

Chapter 5

Principles for the simulation of a micro-Newton RIT

Before applying “PlasmaPIC” to plasma simulations inside a $\mu\text{N-RIT}$, some issues have to be solved and preliminary work has to be done. First, the necessity for a 3D simulation is stated. Afterward, the plasma and neutral gas interaction is analyzed. In this scope the technique for determining the neutral gas density is described. Furthermore, the neutral gas distribution will be investigated as a function of the gas flow and the geometry of the $\mu\text{N-RIT}$.

5.1 Necessity for 3D plasma simulation

A three-dimensional modeling of a $\mu\text{N-RIT}$ is necessary for several reasons. Firstly, a $\mu\text{N-RIT}$ is not completely rotationally symmetric. Secondly, the symmetry is broken due to the coil, and a two-dimensional symmetric simulation disregards the slope of the windings. Depending on the slope the plasma will be influenced. Furthermore, the symmetry is broken due to the apertures in the extraction grids. The shape as well as the location of these apertures play a crucial role for the number of extracted ions, which determines the extraction current. This current is a characteristic parameter for the operation mode of the thruster. Furthermore, this current can be measured very precisely. Hence, it is an ideal candidate for comparison of the simulation with the experiment. In a two-dimensional rotationally symmetric simulation the shape and the location of the apertures are lost, except for the aperture in the

Table 5.1: Neutral gas properties for μ N-RITs with 1 cm, 2.5 cm, and 4 cm diameter.

	RIT 1	RIT 2.5	RIT 4
pressure [Pa]	0.53	0.21	0.13
density [10^{19} m^{-3}]	12.88	5.15	3.22
max distance in vessel [m]	0.01	0.025	0.04
mean free path [m]	0.009	0.023	0.037
Knudsen number	0.94	0.94	0.94
collision probability [%]	52.99	52.99	52.99

middle. Unfortunately, the numerical noise in a two dimensional symmetric simulation is at its highest at the rotation axis, because the volume of the cells decrease to zero there. Apart from these reasons, a three-dimensional simulation allows the investigation of arbitrary geometries. In such a manner completely new thruster designs can be analyzed.

5.2 Neutral gas analysis in a micro-Newton RIT

As seen in section 3.2 the neutral gas density distribution is a required input parameter in the MCC module of “PlasmaPIC”. Due to the small size of the μ N-RIT a measurement of the neutral gas distribution is difficult to accomplish. In order to determine the neutral gas distribution in an arbitrarily shaped vessel with gas sources and gas drains a simulation is the most effective approach. The choice of the simulation method depends on the Knudsen number. By applying the scaling laws [34], which are derived from larger RITs, average densities can be determined. The values for three μ N-RITs with 1 cm, 2.5 cm, and 4 cm diameter are shown in table 5.1. As can be seen, the values for the Knudsen number are barely inside the molecular flow regime. Furthermore, the particle-particle collision probability between two particle-wall collision is about 50%. Since these values are based on estimates, the particle-particle collision will be considered in the following.

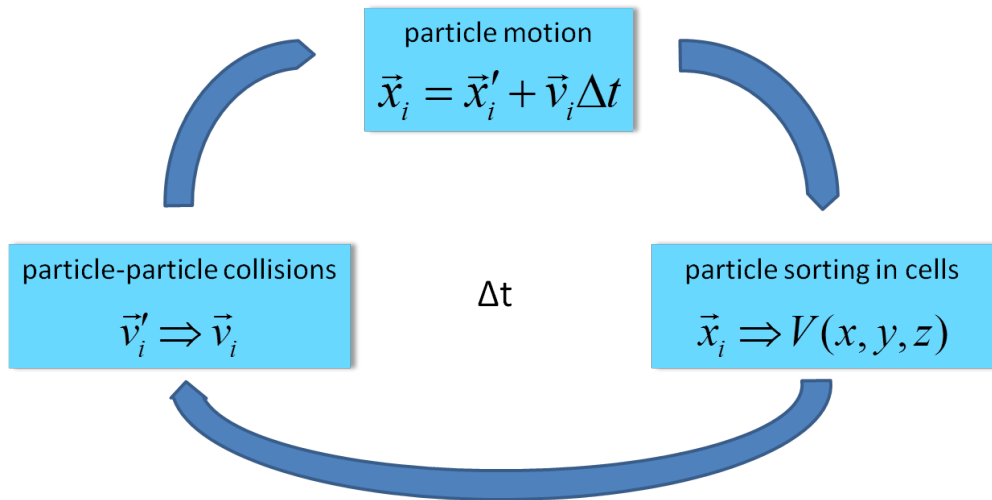


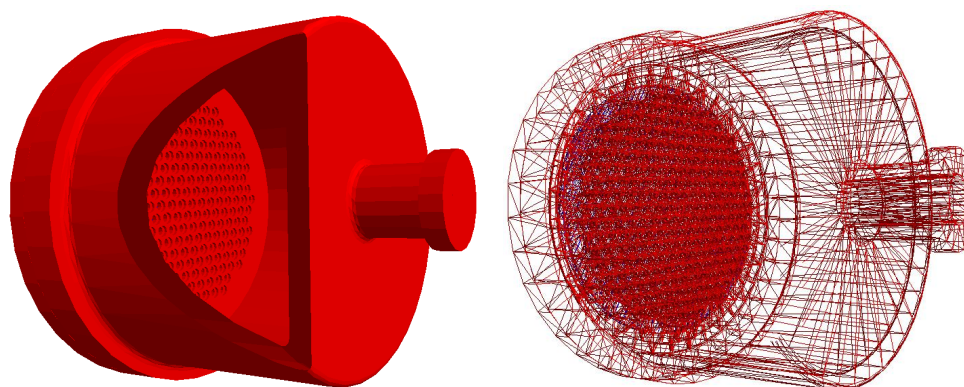
Figure 5.1: One cycle of a time step in a DSMC simulation.

5.2.1 Description of the modeling algorithm

For the modeling of the neutral gas distribution with reduced particle-particle interactions the Direct Simulation Monte Carlo (DSMC) method [35] is well suited. It can also be applied for the molecular flow without losing accuracy, but is computationally wasteful for this task.

The DSMC is similar to the PIC algorithm described earlier. As in a PIC algorithm the particles are represented by super-particles and the space as well as the time are discretized. One time step of the DSMC is shown in figure 5.1. Each particle i is moved from its old position \vec{x}'_i to the new position \vec{x}_i according to its velocity \vec{v}_i . In contrast to the PIC the collisions with walls have to be included in the movement algorithm, because in DSMC a particle will not be absorbed at the wall as in PIC. Afterward the particles are sorted into cells V_j depending on their position. Once the particles are sorted it has to be checked for each particle whether a collision with another particle in its cell occurs. This scheme has to be iterated until the system reaches a steady state. The main differences between PIC and DSMC are the absence of electromagnetic forces and the particle-wall collisions. The velocity of a particle in DSMC is changed only in particle-particle or particle-wall collisions.

The particle-particle collisions are treated in a similar way as the ion-neutral collision in the PIC algorithm. The difference is that a super-particle in DSMC collides with another super-particle in its cell instead of a collision with a local



(a) solid body representation

(b) triangle mesh representation

Figure 5.2: Geometry representation of a RIT 10 in “FlowSim”.

density as in PIC. The hard sphere model is assumed and the collision of two randomly chosen particles is executed by a uniform isotropic rotation of the relative velocity vector in the center of mass frame. The test for a collision is done by the no time counter method [35] that can be seen as a simplified version of the earlier mentioned null collision method, because the maximum collision frequency in DSMC depends only on the relative velocity of both particles only.

The basis of the neutral gas simulation is implemented in the program “FlowSim” [32]. “FlowSim” was originally designed for a molecular gas flow where collisions between particles were neglected. The program features a full 3D treatment of the neutral gas. “FlowSim” supports arbitrarily shaped geometries, which can be designed with a common computer aided design tool (CAD). The imported geometry is represented by triangles. An example is given in figure 5.2.

The figure shows a RIT 10. A gas inlet is located at the right side. The inflowing gas can exit the RIT through the grid system at the left side only. In order to fit the requirements of a DSMC simulation the particle-particle collisions were implemented in “FlowSim” as described before.

5.2.2 Simulation results for the neutral gas distribution

In this section the neutral gas distribution in a μN -RIT 2.5 is examined using the self-developed simulation program “FlowSim”. The following results were

Table 5.2: Average densities for different system parameters [10^{19}m^{-3}] [36].

	mass flows [sccm]					
	0.06	0.07	0.08	0.17	0.19	0.21
13 apertures; top inlet	6.94	8.04	9.18	-	-	-
13 apertures; bottom inlet	6.96	8.11	9.24	-	-	-
37 apertures; top inlet	-	-	-	6.99	7.82	8.65
37 apertures; bottom inlet	-	-	-	6.93	7.75	8.60

presented at the “32nd International Electric Propulsion Conference” [36]. The used discharge chamber geometry of the $\mu\text{N-RIT 2.5}$ is dome-shaped. For the distribution analysis two different gas inlet positions are considered. On the one hand the gas inlet is located above the extraction grid and is pointing towards the grid. On the other hand several gas inlets are distributed at the edge of the extraction grid and directed against it. Furthermore, two cases for the number of grid apertures, 13 and 37, are considered. Each of these four configurations is evaluated for three different mass flows. The mass flow is given in standard cubic centimeter (sccm). For the configuration with 13 grid apertures the mass flow is changed from 0.06 to 0.08 sccm. To meet the same operation pressure at 37 grid apertures the mass flow is varied from 0.17 to 0.21 sccm. For all simulations a constant temperature of 300 K is applied to the inflowing gas and all parts of the thruster.

Table 5.2 shows the average densities for the described configurations. As expected the average densities are raising with increasing mass flows. In detail, the average densities are increasing almost with the same factor as the mass flow. Consequently, the flow conductance of the grid system is constant for different densities in the discharge chamber. This corresponds to a molecular gas flow in the grid system. Considering the Knudsen number for the grid system (which is larger than 10 due to the small aperture diameter of less than one millimeter) this observation is confirmed.

In the following the density profiles of the four configurations are investigated. For clarity the density profiles are presented as 2D profiles that are cut planes of the 3D simulation. The cut plane, illustrated in figure 5.3, is perpendicular and centered at the circular base of the thruster. Although the average densities in table 5.2 are less affected by the gas inlet position, figures 5.2.2

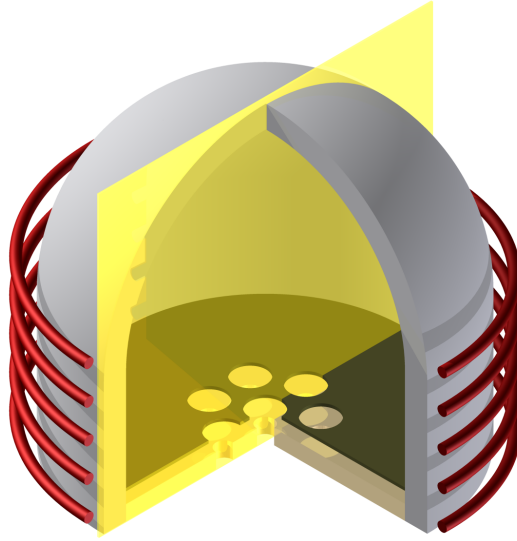


Figure 5.3: Illustration of the cut plane for the 2D density profile in a μ N-RIT.

and 5.2.2 reveal that the gas inlet position has an impact on the density distribution. Moreover, an inhomogeneous density profile is observed. Because the variations of the density are small for the majority of the discharge chamber, a homogeneous distribution can be applied for the plasma simulation. However, the neutral gas density distribution can be influenced by the plasma. This will be analyzed in the next section.

5.3 Interaction of plasma and neutral gas

The concept for simulating a μ N-RIT is as follows. Firstly, the neutral gas density is calculated and afterward the plasma is simulated using this density. However, the plasma simulation considers the neutral gas distribution as constant in time. Thus the influence of the plasma on the neutral gas is neglected. If this is not the case the neutral gas simulation and the plasma simulation have to be iterated self-consistently considering the influence of each other. A simulation that includes neutral gas as well as plasma is not applicable due to the large differences in the time scales. Firstly, the electrons are more mobile than the neutral gas particles due to the high mass difference and energy difference. Secondly, the electrons and ions interact by electric fields due to their charge. In contrast, the neutral gas particles are influenced by particle-wall and particle-particle collisions only, which are less frequent.

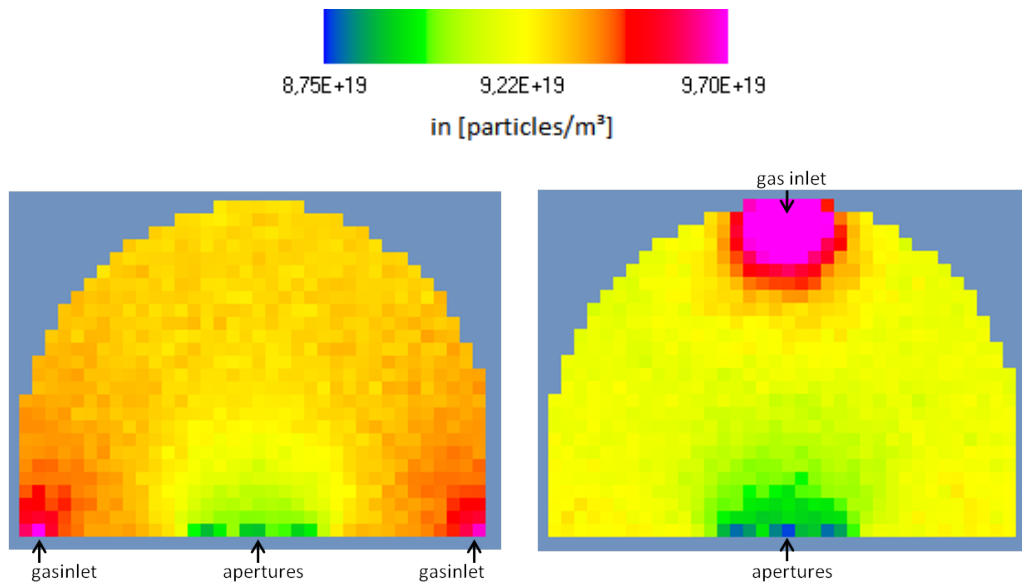


Figure 5.4: 2D density profiles for 13 apertures at a mass flow of 0.08 sccm with the gas inlet located at the bottom (left) and at the top (right) [36].

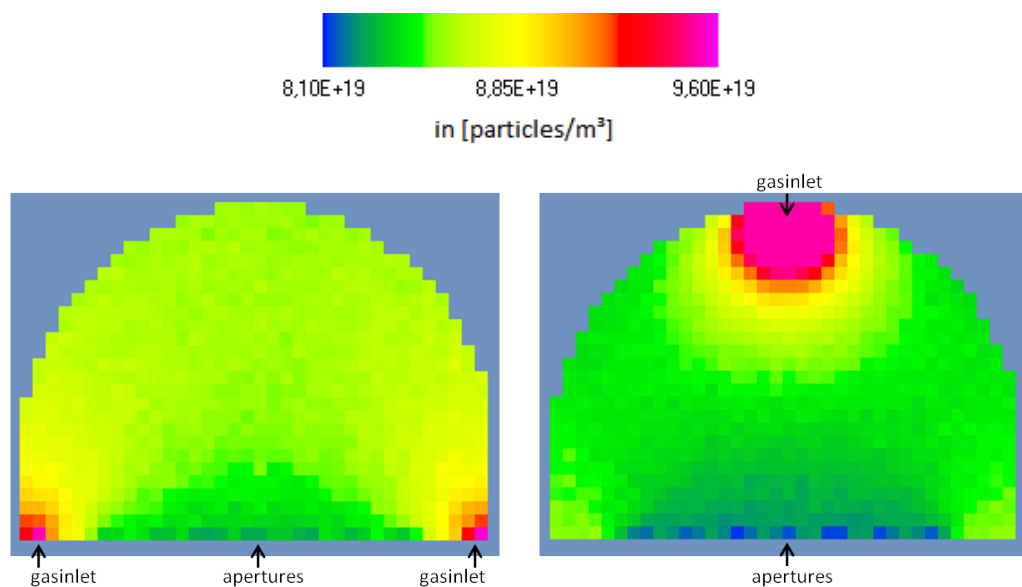


Figure 5.5: 2D density profiles for 37 apertures at a mass flow of 0.21 sccm with the gas inlet located at the bottom (left) and at the top (right) [36].

For a low pressure discharge an influence of the plasma onto the neutral gas is not important in general, because the ionization degree is less than one percent. However, for the simulation of a $\mu\text{N-RIT}$ a special situation arises. By turning on the extraction, ions are leaving the discharge chamber beside the neutral gas particles. Because the total number of exiting particles has to be constant, the quantity of neutral particles has to decrease by the number of leaving ions. However, the number of leaving neutral particles is directly proportional to the density, because the gas conductance for the grid is constant for different densities as shown in the last section. Assuming a ratio of one for ions and neutral gas particles, which is a realistic value for a RIT (see section 2.8), the neutral density in the discharge chamber has to drop by a factor of two.

Chapter 6

Demonstration of the capabilities of “PlasmaPIC”

This chapter demonstrates the great capabilities of “PlasmaPIC”. The focus lies on inductively coupled plasmas in a micro-Newton RIT ($\mu\text{N-RIT}$). At the beginning an overview of the simulation set-up for a $\mu\text{N-RIT}$ is given. Thereafter, a first simulation is shown that demonstrates the convergence and stability of a long term run. In the following the variation during one rf-cycle is analyzed. Moreover, the influence of the neutral gas density and the applied power on the plasma parameters as well as the thrust are investigated. Finally, it will be pointed out that “PlasmaPIC” is not limited to a $\mu\text{N-RIT}$ by applying “PlasmaPIC” to the simulation of a DC discharge.

6.1 Simulation of a micro-Newton RIT 1.0

For the demonstration of the capabilities of “PlasmaPIC” a $\mu\text{N-RIT}$ 1.0 is chosen. The small diameter of 1 cm for the discharge chamber minimizes the computational effort without losing the character of an inductively coupled discharge in a $\mu\text{N-RIT}$. According to the scaling laws [34], the interior height of a $\mu\text{N-RIT}$ is 1 cm also. The $\mu\text{N-RIT}$ 1.0 geometry is shown in figure 6.1. The coil (red) is wrapped in 5 loops around the discharge chamber (grey) starting with a radius of 0.6 cm, which decreases from a certain height to follow the dome-shaped discharge chamber. In this way the distance of the coil to the plasma is kept constant at 0.1 cm for the whole discharge chamber.

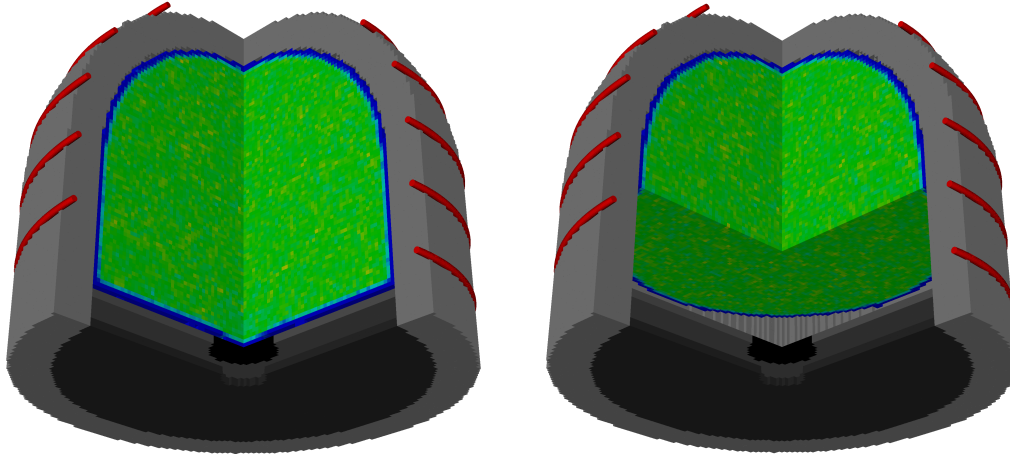


Figure 6.1: μ N-RIT 1.0 with initial homogeneous ion density distribution; variations due to statistical causes; the selected ion density area of interest at the right side; for scale see figure 6.2(a)

The μ N-RIT 1.0 is completed by two grids (dark grey) having one aperture each. Each grid has a thickness of 0.2 cm and the grid gap is 0.6 cm. At the inner grid a voltage of 1500 V is applied whereas the outer grid is set to -150 V. In this manner the extraction is simulated. All particles that strike the exit plane (black) filling the aperture of the inner grid contribute to the extraction current and will be removed from the simulation. At the beginning the electron and ion densities are initialized with a homogeneous distribution of $2.5 \cdot 10^{16} \frac{1}{\text{m}^3}$ (green) in the whole discharge chamber except in a small region at the border. The initial density is set to zero in this region (blue) to account for the plasma sheath that will be formed. This reduces the computational costs because the initial conditions are closer to the real ones. The fluctuations in the density distribution are due to statistical causes.

In “PlasmaPIC” the inductively coupled discharge is controlled by the power deposition, which is a predefined value and is set by the input card. Starting with a given initial coil current the actually achieved power deposition in the plasma of the μ N-RIT is determined in each rf-cycle. The coil current will be readjusted in each following rf-cycle until the achieved power deposition matches the predefined target value [28].

In the following the above system will be used to demonstrate the capabilities of “PlasmaPIC”. Furthermore, the plasma properties and the operating parameters of a μ N-RIT 1.0 are presented.

6.1.1 Convergence and long-term stability test

First of all, it will be shown that the simulation of a $\mu\text{N-RIT}$ converges and a stable state is established over a long-term run as well. For this task a frequency of 5 MHz and a power of 0.1 W are applied to the $\mu\text{N-RIT}$ 1.0. The neutral gas pressure is $8.5 \cdot 10^{19} \text{ m}^{-3}$ and the utilized neutral gas is Xenon. The numerical parameters are as follows: a cell size of 0.11 cm, a time step of $5 \cdot 10^{-12}$ s, and a super-particle ratio of 1000. This leads to around $100 \times 100 \times 100$ cells and ten million electrons and ions that have to be processed. The simulation was executed over nine million time steps that correspond to 45 μs , or 225 rf-cycles. The simulation was performed with 384 CPU cores that required 14 days for the simulation, which corresponds to over 10^5 CPU hours.

In order to demonstrate the convergence different quantities are examined. Firstly, figure 6.2 illustrates the evolution of the ion density distribution for the selected area shown in figure 6.1. As of figure 6.2(k), which is taken after five million time steps, no further change of the ion density distribution is observable.

Secondly, figure 6.3 reveals that the total number of particles in the plasma, the mean energy of electrons and ions, the coil current, and the power deposition have reached a steady-state after roughly one million time steps and are stable over the next eight million time steps. With this, it is demonstrated that “PlasmaPIC” converges and is also long-term stable.

In the following section the already presented figures 6.2 and 6.3 will be analyzed in more detail.

6.1.2 Analysis of the simulation results

At the beginning of this section a brief description of the transient seen in figures 6.2 and 6.3 is given. Figures 6.2(a) and 6.2(b) show the forming of the sheath and presheath. Furthermore, the plasma burns out. This is related to the small initial coil current and the low power deposition resulting from this. In each rf-cycle the coil current is therefore increased until the power deposition is reached. The adjustment has to be made slowly to avoid a dramatic increase in the plasma density, which would lead to numerical instabilities. Figures 6.2(d)-6.2(f) show that starting from a certain coil current value the ion density

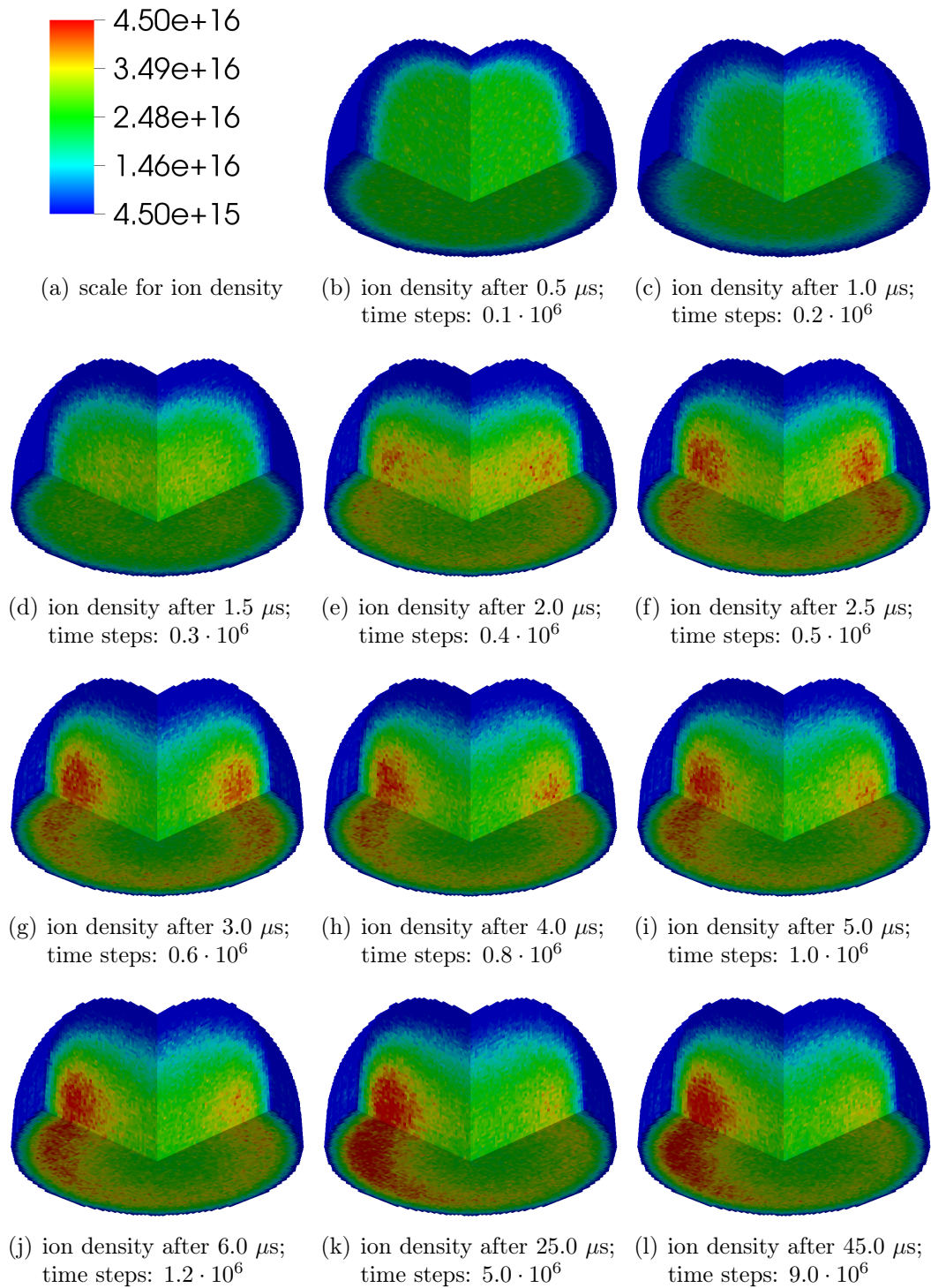
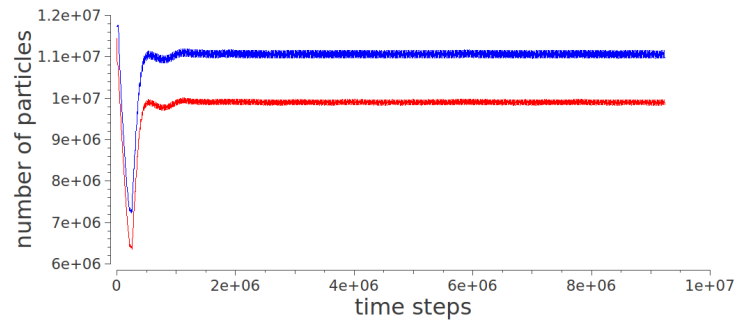
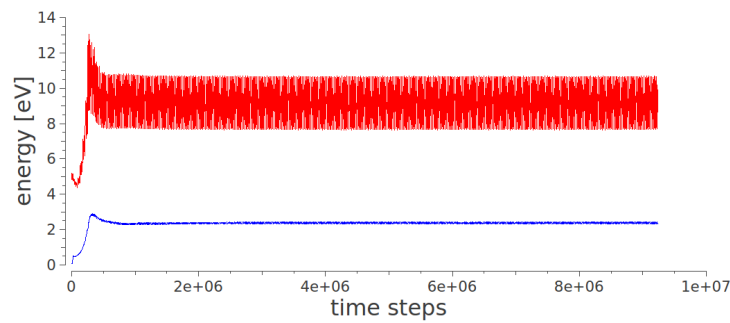


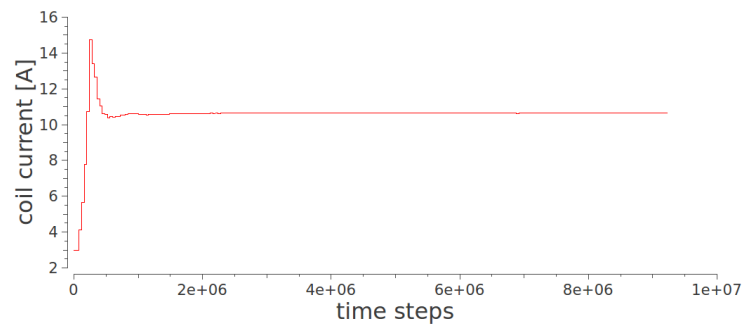
Figure 6.2: Reaching the steady-state of a plasma discharge simulation in a $\mu\text{N-RIT 1.0}$



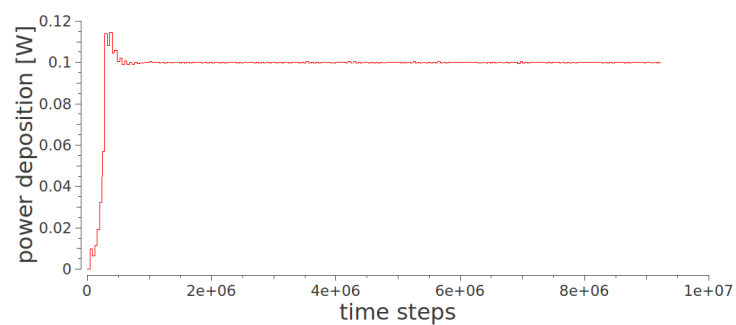
(a) Evolution of the total number of electrons (red) and ions (blue)



(b) Evolution of the electron (red) and the ion (blue) temperature



(c) Evolution of the coil current



(d) Evolution of the actually achieved power deposition

Figure 6.3: Convergence of all shown quantities is reached after approximately one million time steps. Furthermore, all quantities are long-term stable.

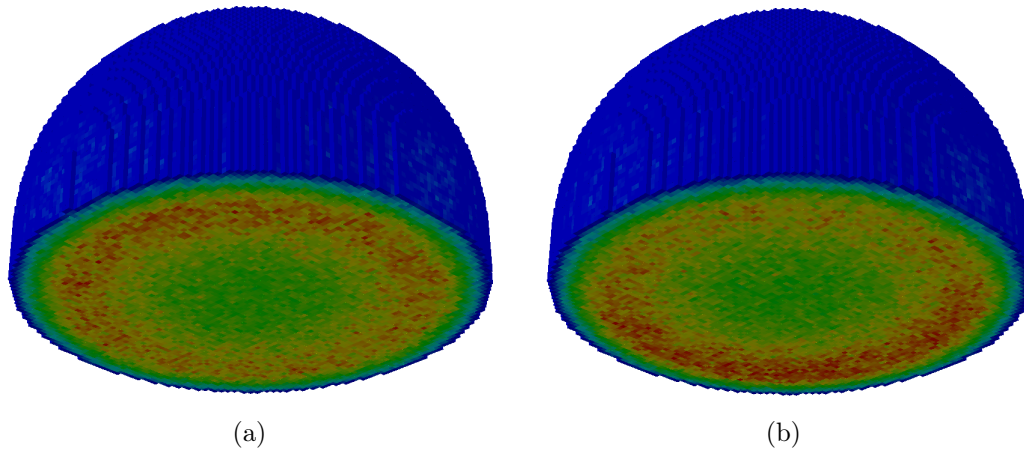


Figure 6.4: Ion density distribution for five single loops that are rotationally symmetric. In the right figure the coil is shifted backwards by 0.3 mm compared to the left figure. For scale see 6.2(a)

rises. More noticeably, the ion density increases in a torus-shaped region. From figures 6.2(g) to 6.2(k) it can be seen that the almost rotationally symmetric ion density distribution changes to a nonsymmetric one. On the left side a high density area of ions is formed that stays constant until the end of the simulation in figure 6.2(l).

This highly irregular density distribution has a two-fold cause: First of all, the coil is not rotationally symmetric because of the slope and the decreasing radius at the curved top of the μ N-RIT. By substituting the coil with five single loops (which have no slope and hence are rotationally symmetric), it is shown in figure 6.4(a) that the ion density is almost rotationally symmetric. The left variation in the torus-shaped high ion density results from a slight shift of the coil against the discharge chamber. In figure 6.4(b) the coil is shifted backwards by 0.3 mm and it can be seen that the ion density distribution is very sensitive even to this small shift of 3%. This irregular ion density distribution is an interesting result and was not investigated for μ N-RITs in experiment.

Although the ion density distribution was considered up to now, figure 6.5 shows that the electron density distribution is similarly shaped. However, the electron density is smaller at the edge of the plasma, which is related to the plasma sheath.

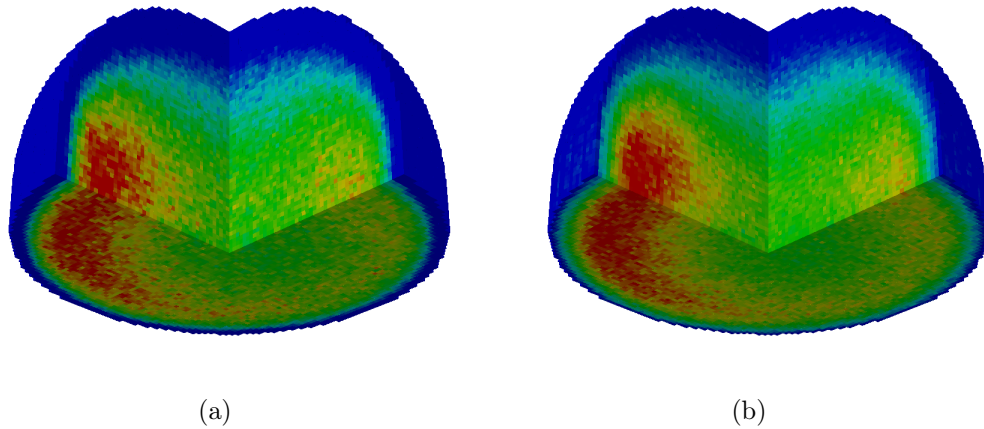


Figure 6.5: Electron (left) and ion (right) density distribution are similar shaped. At the edge the electron density is quite smaller due to the plasma sheath.

Figure 6.6 shows the distribution of the electric field magnitude. The electric field is always greater than zero despite the rotational character. This results from the slope of the coil, which generates a small electric field component pointing upwards. Moreover, the slope and the small shift of the coil give rise to the asymmetrical electric field, which is stronger on the right side.

The resulting magnetic field magnitude is shown in figure 6.7. The high values at the edge indicate the position of the coil. Throughout the system the magnetic field magnitude is relatively homogeneous as expected.

Figure 6.8 shows the distribution of the power deposition in the plasma. As mentioned in section 2.5.3, the power deposition is related to the electron current and the electric field. The electron current in turn is proportional to the electron density. Figure 6.8 demonstrates that the power deposition distribution follows the electron density distribution. In contrast, the electric field in figure 6.6 is stronger on the right side. This indicates that the highest plasma densities do not have to be located at the position with the highest electric field, which is quite surprising.

In the next section a single rf-cycle will be investigated in more detail.

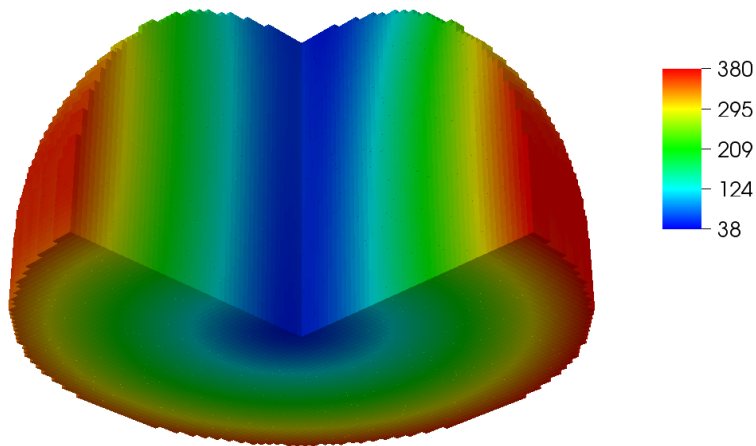


Figure 6.6: Map of the electric field magnitude [V/m]. The electric field is stronger on the right side and greater than zero in the middle. Both effects are related to the slope of the coil. In contrast, the electron density is higher on the left side (see fig 6.5(a)).

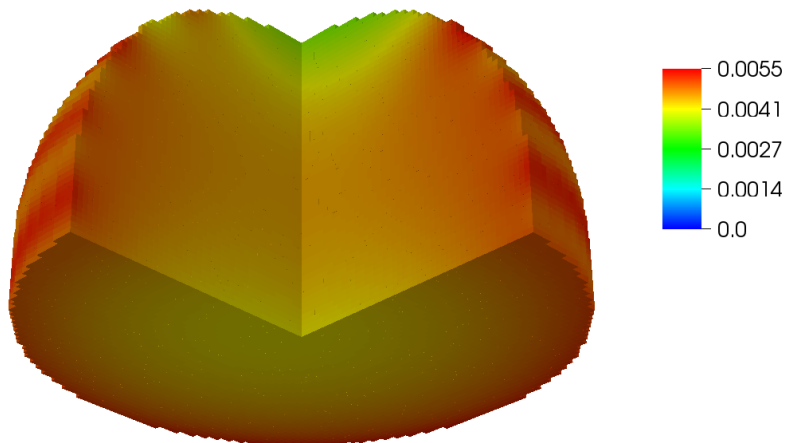


Figure 6.7: Map of the magnetic field magnitude [T]. Throughout the plasma region the magnetic field is relative homogeneous. Merely the regions located close to the coil at the plasma border are slightly raised.

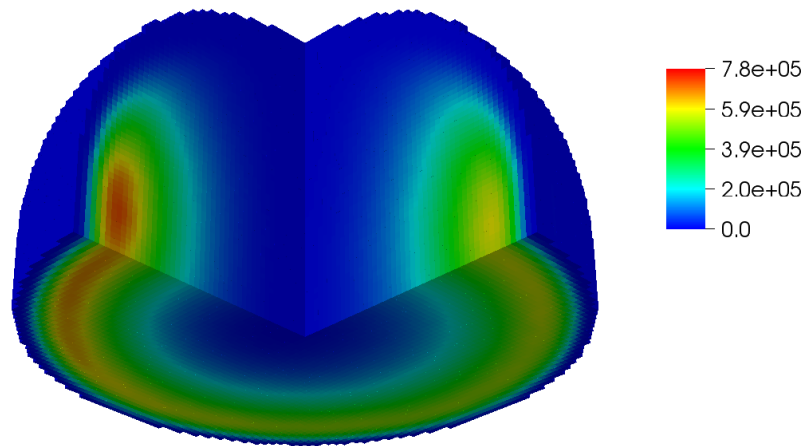


Figure 6.8: Map for the power deposition [W/m^3] averaged over one rf-cycle. The highest values are achieved for the regions with the highest electron densities (see fig. 6.5(a)) and not for the regions with the highest electric field (see fig. 6.6).

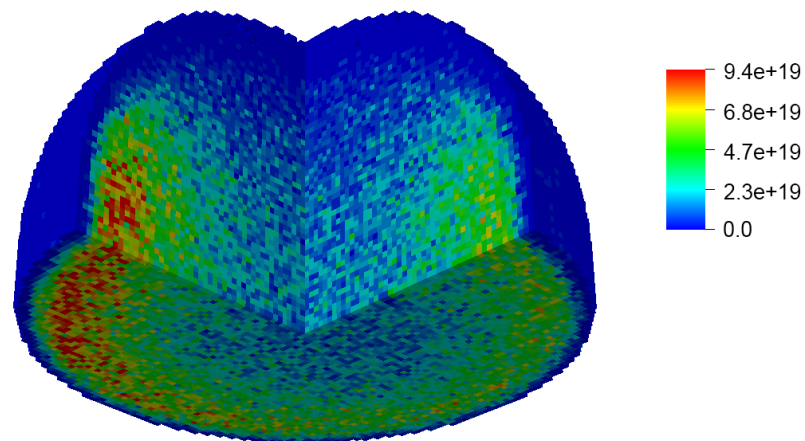
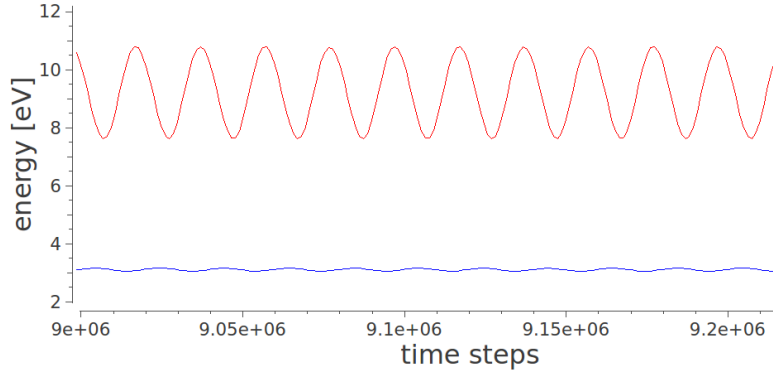
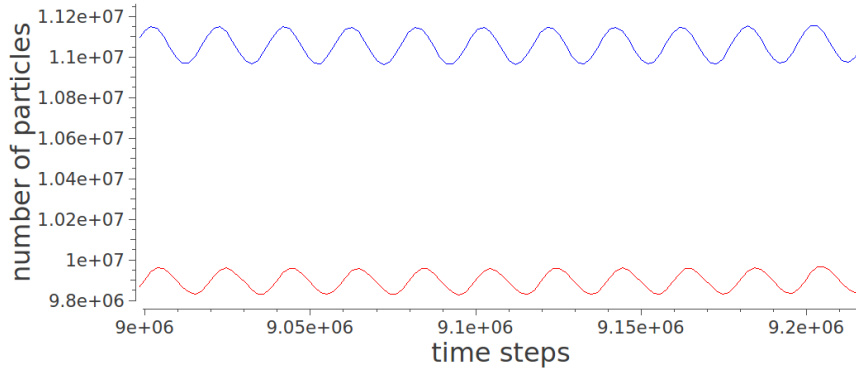


Figure 6.9: Map for the ionization rate [Hz/m^3] averaged over one rf-cycle. The highest values are achieved for the regions with the highest electron densities (see fig. 6.5(a))



(a) electron temperature (red) and ion temperature (blue)



(b) total number of electrons (red) and ions (blue) in the system

Figure 6.10: Fluctuations of the temperature and number of particles during several rf-cycles.

6.1.3 Investigation of a single rf-cycle

Up to now the fluctuations of the total particles in the system, as seen in figure 6.3(a), as well as the fluctuations of the temperature, as seen in figure 6.3(b), have not been discussed. These fluctuations and their impact will be described in this section in detail.

Figure 6.10 is an enlargement of the figures 6.3(a) and 6.3(b). Figure 6.10(a) shows that the electron temperature (red) is fluctuating with a frequency of 10 MHz, which is twice of the applied frequency of the coil current (5 MHz). This is because the electric field accelerates and decelerates the electrons twice in each rf-cycle. It is remarkable that the electron temperature is varying by 3 eV. Because the smallest temperature is 8 eV, the electron temperature increases in one rf-cycle by about 35%. As a consequence the plasma-wall potential, which is proportional to the electron temperature, is fluctuating with

the same frequency. Figure 6.11 shows the electron energy distribution and the plasma-wall potential at the time point of minimal and maximal average electron energy. For the minimal average electron energy the electron energy distribution is quite homogeneous and the plasma wall potential is small with a drop in the middle. For the maximal average electron energy the electrons at the edge are considerably hotter than the electrons in the middle. The plasma wall potential is correspondingly higher due to the temperature increase and has its maximum in the middle.

This oscillation of the plasma-wall potential explains the energy oscillation of the ions in figure 6.10(a), which cannot be explained by the electric field of the coil due to the high mass difference of xenon ions and electrons. Furthermore, the oscillation of the temperature causes slight oscillations of the total number of electrons and ions in the system.

6.1.4 Variation of the power deposition

An important property of a μ N-RIT is its behavior at different power values. This investigation can also be performed with “PlasmaPIC”. In the following the influence of different power deposition values on the coil current, the maximal and minimal electron energy of one rf-cycle, and the extraction current density will be analyzed. This examination was done for seven different power deposition values. Note that this power deposition is the power coupled into the plasma and not the power which has to be applied to the coil. The applied power to the coil is usually quite larger due to electromagnetic losses.

Figure 6.12 shows the values mentioned above as functions of the power deposition. The extraction current is proportional to the power deposition. As a consequence the plasma density has to rise by the same factor, which is demonstrated in figure 6.12 for two power deposition values of 0.08 W and 0.16 W. This was to be expected. In contrast, the coil current shows a surprising behavior: With increasing power deposition the coil current reduces, which is the opposite of what one would expect. The explanation for this behavior is given in figure 6.13, which shows the ion density distribution and power deposition distribution. At a higher power deposition value the ion density increases throughout the whole system. By that the ion density is considerably increased at the edge of the plasma, where the electric field is stronger due to

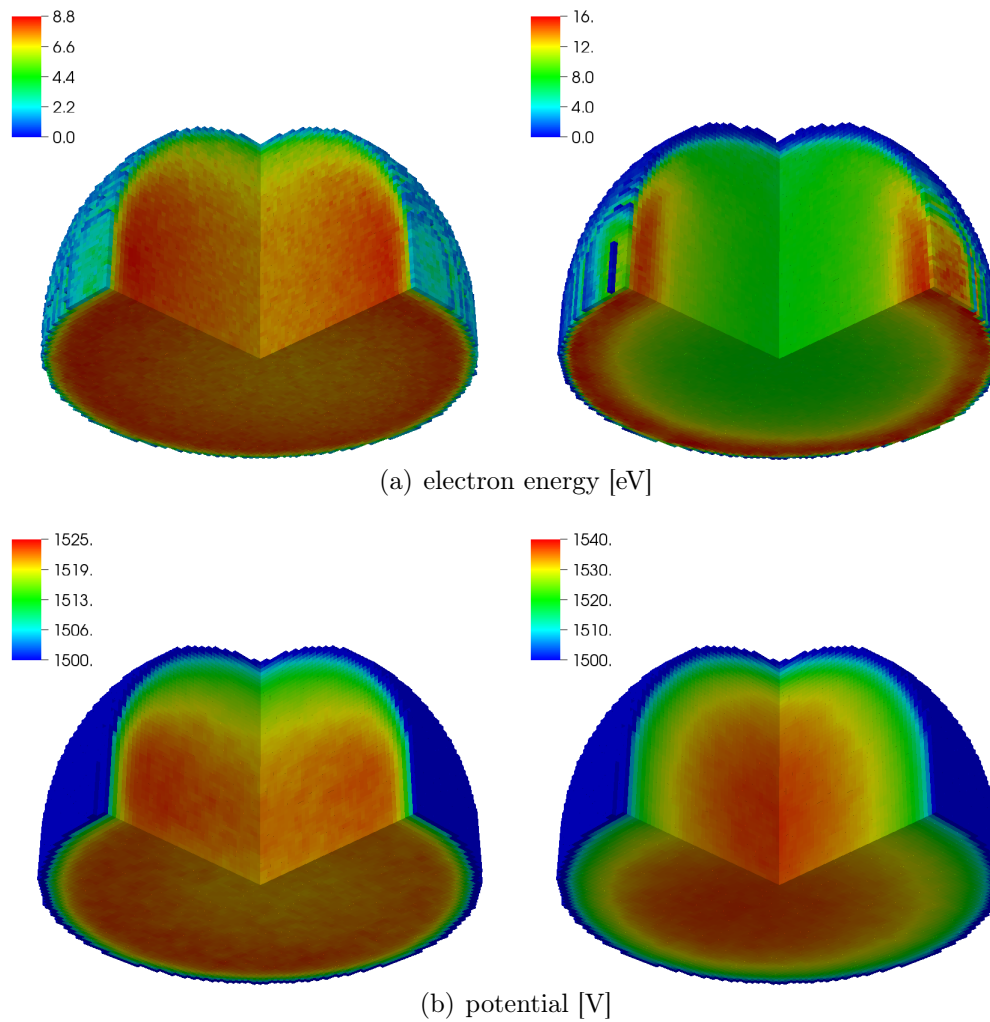


Figure 6.11: On the left side the electron energy distribution and the potential distribution for the time point of the minimal average electron energy is shown. On the right side both distributions are shown for the time point of the maximal average electron energy.

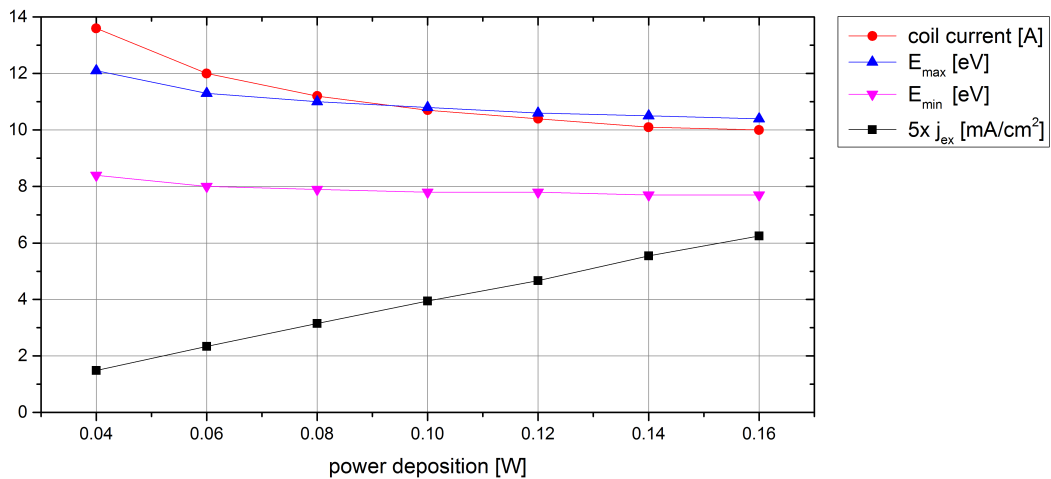


Figure 6.12: Extraction current density j_{ex} , coil current, and average maximal (minimal) electron temperature E_{\max} (E_{\min}) in an rf-cycle as a function of the power deposition. Mind that the extraction current j_{ex} was multiplied by five for a better representation.

the rotational character. For that reason a smaller current is sufficient for the same power deposition. In this way the power deposition is shifted towards the edge of the plasma.

The decrease of the maximal and minimal electron energy in figure 6.12 is related to the current drop. As mentioned before the current drop causes a weaker electric field and thus a smaller acceleration of the electrons. In conclusion, the coupling of the power into the plasma has improved due to an expansion of the plasma towards the discharge chamber wall.

6.1.5 Influence of neutral gas pressure on the plasma density

In general it is assumed that the plasma density in the discharge chamber is at its highest value in the middle. However, another plasma density distribution was observed with “PlasmaPIC”. Indeed, the plasma density distribution depends on the neutral gas pressure in the discharge chamber. In this section the influence of the neutral gas pressure on the plasma discharge will be examined.

For a RIT 10 Schäfer, [37] measured the radial plasma density distribution as a function of the neutral gas pressure. Figure 6.14 shows the results, which

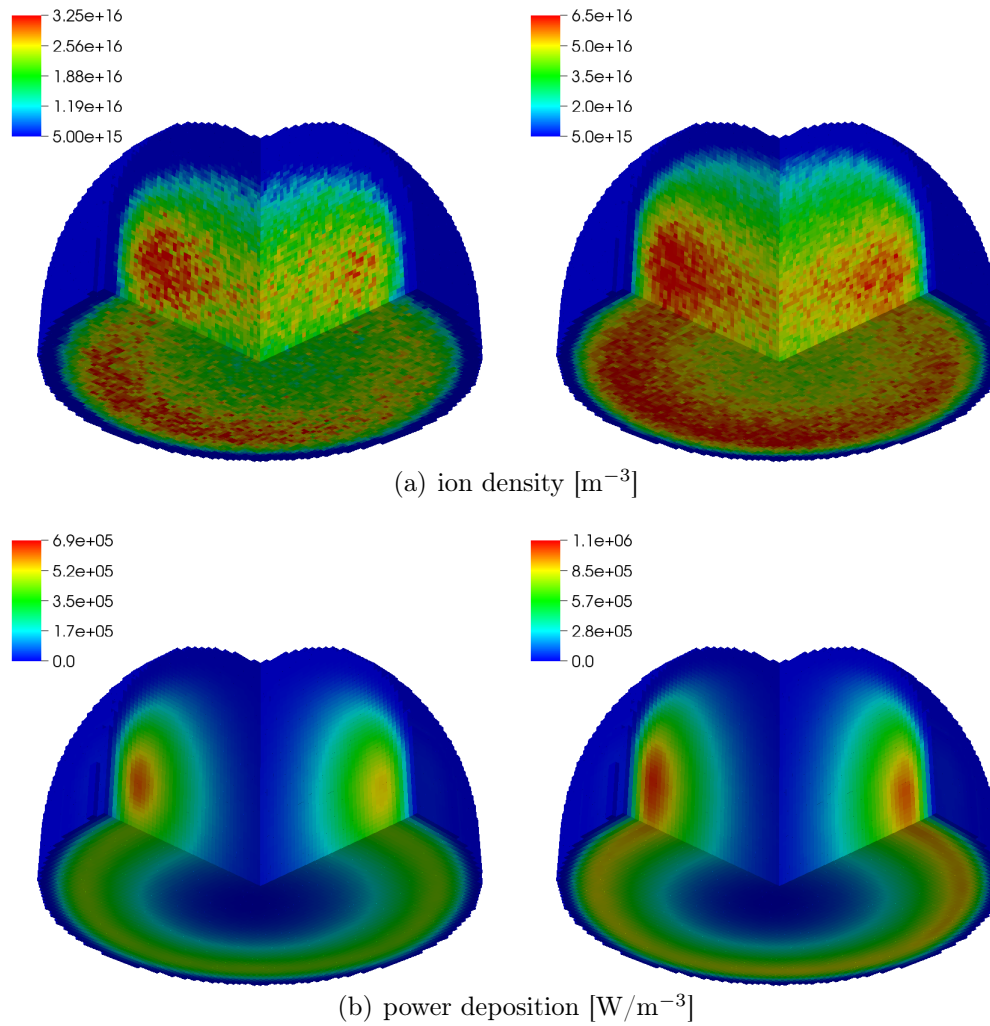


Figure 6.13: On the left side the ion density distribution and the power deposition distribution for a total power deposition value of 0.08 W are shown. On the right side both distributions are shown for a total power deposition value of 0.16 W.

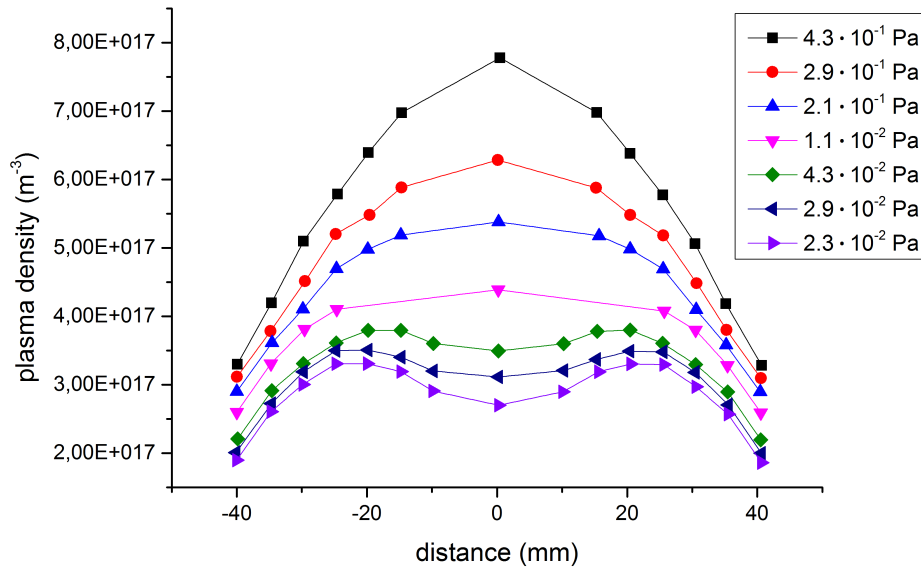


Figure 6.14: Measured radial plasma density distribution in a RIT 10 for different neutral gas pressures [37]; the measurement was done from 0 to 40 mm and was mirrored afterward.

were measured with a Langmuir probe¹. For pressures above $1.1 \cdot 10^{-2}$ Pa the plasma density is at its highest value in the middle of the discharge chamber and decreases to the wall, as expected. In contrast, for pressures below $1.1 \cdot 10^{-2}$ the maximum of the plasma density is located of around half of the discharge chamber radius, and a drop of the density towards the middle of the discharge chamber is observed. The latter one corresponds to the values simulated with “PlasmaPIC” for a μ N-RIT 1.0. However, for the already presented simulation results a neutral gas pressure of 0.35 Pa was used in the simulation of μ N-RIT 1.0. Using the scaling laws [34], the neutral gas pressure of 0.35 Pa in a μ N-RIT 1.0 corresponds to a neutral gas pressure of $3.5 \cdot 10^{-2}$ in a RIT 10. Hence, the shape of the simulated plasma density distribution qualitatively fits the measured values. The quantitative variation arises from the different power values applied.

In the following it will be analyzed whether the same dependence of the neutral gas pressure on the plasma density distribution in a RIT 10 can be

¹A Langmuir probe consists of a conducting thin needle that is put into the plasma. By measuring the current on the needle the electron density and temperature can be determined.

observed with “PlasmaPIC” for a $\mu\text{N-RIT}$ 1.0. Figure 6.15(a) shows the simulated results of the ion density distribution in $\mu\text{N-RIT}$ for 0.35 Pa on the left side and for 4.1 Pa on the right side. This corresponds to a neutral gas pressure of 0.41 Pa in a RIT 10. For both simulations a power deposition of 0.06 W was applied. The characteristics of the ion density distribution correspond to the ones measured in figure 6.14. By that it is demonstrated that the plasma density distribution in a $\mu\text{N-RIT}$ 1.0 shows a similar dependence on the neutral gas pressure as the one in a RIT 10. Note that the measurement in figure 6.14 was done from the middle to one edge and mirrored afterward. Therefore, the asymmetric calculated ion density distribution in figure 6.14 cannot be observed with the measurement setup from Schäfer.

This shows the power of “PlasmaPIC” because this fact cannot be measured with a Langmuir probe in a $\mu\text{N-RIT}$ due to the small system size.

Next, a brief analysis of the two different pressure simulations will be given. Figure 6.1.5 shows a comparison of the ionization rate for the two applied pressures. In both cases the ions are created at the positions with the highest density and do not diffuse to the middle at higher pressures. Furthermore, the power deposition characteristic in figure 6.15(c) shows an interesting behavior. For the low pressure simulation the maximal power deposition is located at the highest plasma densities, which is not at the position of highest electric field. In contrast, the power deposition is at its highest value at the right side, where the electric field is at its strongest.

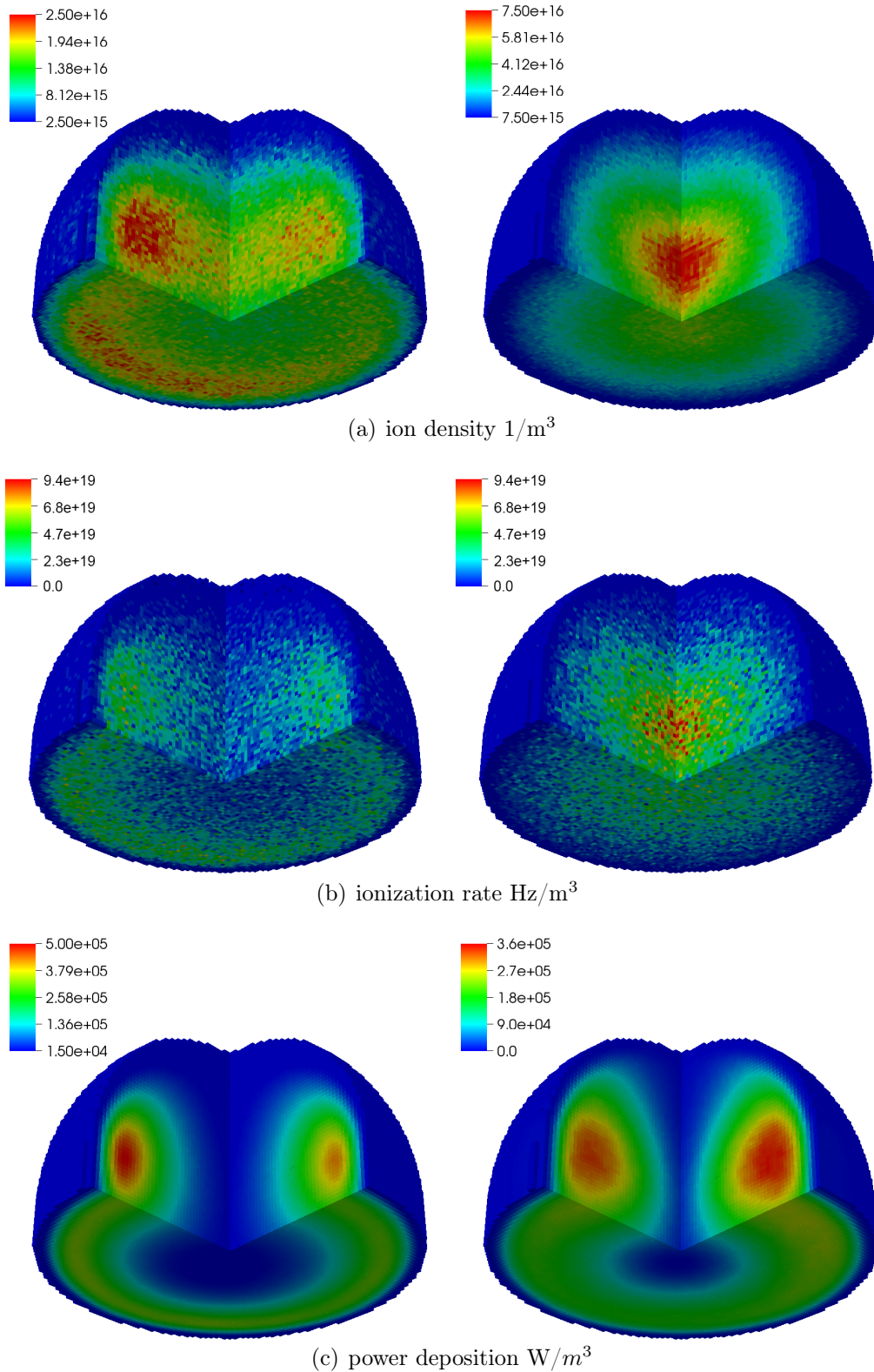


Figure 6.15: Comparison of the influence of the neutral gas pressure on the plasma discharge. The neutral gas pressure is at 0.35 Pa in the left-hand images and at 4.1 Pa in the right-hand ones. The power deposition is set to 0.06 W.

6.2 Modeling of a DC discharge

In this section it will be demonstrated that “PlasmaPIC” is not limited to the modeling of a μ N-RIT or other inductively coupled plasmas. “PlasmaPIC” can be used for many different low pressure discharges. For example, “PlasmaPIC” can be applied to the simulation of a DC discharge. For this task the plasma discharge in a cylindrical glass tube enclosed by two electrodes is simulated. The length of the tube is 0.2 m and the diameter is 0.03 m. The neutral gas used is neon and the pressure is set to 62 Pa. The simulation is controlled by the current, which is in contrast with the power controlled simulation for the inductively coupled plasma. The applied current amounts to 0.5 mA. For the representation of the three-dimensional simulation, 2D cut planes are chosen, which is accurate because of the rotationally symmetric geometry.

Figure 6.16 shows the evolution of the ion density distribution over 6 μ s. Figure 6.16(b) displays the density distribution 1.2 μ s after the start of the simulation. At the cathode on the right side a high density bulk is formed, which shows only slight variations during the simulation. On the contrary, two smaller ion bulks are formed that move towards the cathode. These bulks are the so called striations, which are mentioned in chapter 2.5.2.

As can be seen, “PlasmaPIC” is well-suited for the modeling of other plasma discharges.

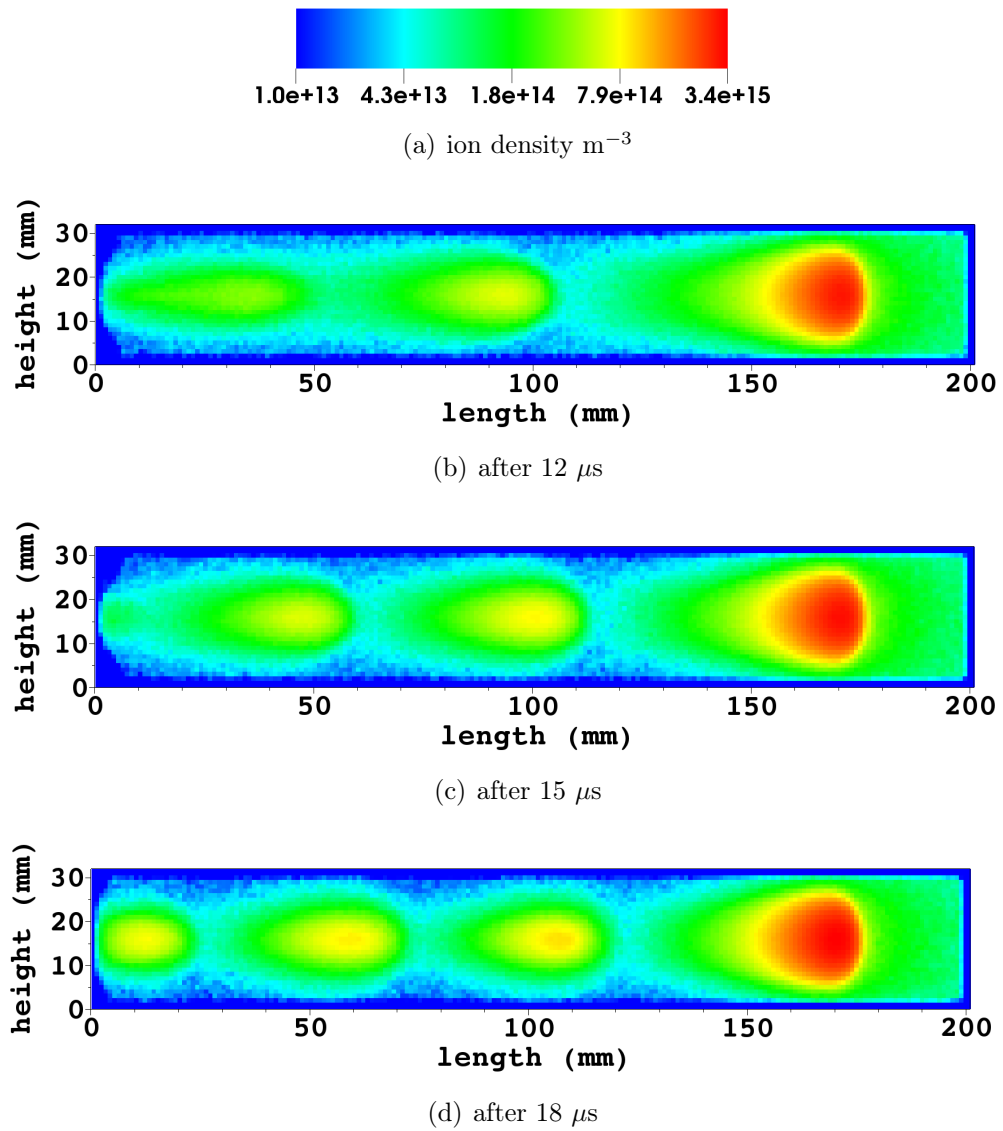


Figure 6.16: Evolution of the ion density distribution in a DC discharge showing the creation and motion of striations.

Chapter 7

Conclusions

In this PhD thesis the plasma simulation tool “PlasmaPIC” was developed from scratch. “PlasmaPIC” distinguishes itself with a fully three-dimensional simulation of a plasma discharge. Even though a full three-dimensional simulation requires a huge amount of computational power, “PlasmaPIC” reduces the required simulation time from month to several hours. This is achieved by a massive parallelization of all functions in “PlasmaPIC”. For this task different software parallelization libraries as well as different hardware architectures, CPU and GPU, were tested and compared. It was shown that the CPU architecture performing the message passing interface (MPI) is the most promising one due to the great scalability. Performance tests have demonstrated that “PlasmaPIC” features an excellent speed-up. Depending on the mesh size, up to thousand cores were efficiently used.

Beside the massive parallelization, “PlasmaPIC” can handle arbitrarily shaped geometries, which can be easily imported from computer aided design tools.

“PlasmaPIC” was designed for the modeling of inductively coupled low pressure plasmas. However, “PlasmaPIC” supports the modeling of different low pressure discharges. Among others it was successfully applied to the modeling of a DC discharge. However, the main focus lies on the simulation of a $\mu\text{N-RIT}$.

“PlasmaPIC” offers new insights into plasma properties within a $\mu\text{N-RIT}$ that are not accessible by experiments. It was shown that a vast set of plasma parameters can be obtained with a high spatial resolution of less than one millimeter in three dimensions. This means that plasma density profiles in a

μ N-RIT are revealed that were previously known only for larger RITs. Furthermore, the variation of plasma parameters during one rf-cycle in a μ N-RIT can be examined with “PlasmaPIC”. This insight is hard to get in experiments, because experiments with temporal resolution are required for this purpose.

Although the investigations were done for a μ N-RIT 1.0, “PlasmaPIC” is able to model a μ N-RIT 2.5 or larger. Even though the computational costs are at least a factor of 12 larger for a μ N-RIT 2.5 due to the system size increase, the required computation time will only slightly increase by applying more CPU cores.

The main advantage of “PlasmaPIC” is its ability to predict plasma and performance parameters for new thruster designs on a microscopic scale. By this means “PlasmaPIC” introduces a new way of understanding and optimizing μ N-RITs.

Bibliography

- [1] Davide Curreli and Francis F. Chen. Equilibrium theory of cylindrical discharges with special application to helicons. *PHYSICS OF PLASMAS*, 18:113501, 2011.
- [2] Michael A. Lieberman and Allan J. Lichtenberg. *Principles of Plasma Discharges and Materials Processing*. Wiley, second edition edition, 2005.
- [3] Pascal Chabert and Nicholas Braithwaite. *Physics of Radio-Frequency Plasmas*. Cambridge, 2011.
- [4] M. Surrendra, D.B.Graves, and G.M. Jellum. Self-consistent model of a direct-current glow discharge: Treatment of fast electron. *Physical Review A*, 41(2):1112–1125, 1990.
- [5] A. Bondi. van der waals volumes and radii. *Journal of Physical Chemistry*, 68(3):441–451, 1964.
- [6] Ulrich Stroth. *Plasmaphysik*. Vieweg+Teubner, 2011.
- [7] Vladimir I Kolobov. Striations in rare gas plasmas. *Journal of Physics D: Applied Physics*, 39:R487, 2006.
- [8] T. Maruyama, S. Nishina, H. Kitamura, K. Itagaki, and H. Mizuochi. Stationary striations due to interaction of two ionization waves in xenon glow discharge. *Contributions to Plasma Physics*, 30:497–510, 2006.
- [9] V I Kolobov and D J Economou. The anomalous skin effect in gas discharge plasmas. *Plasma Sources Science and Technology*, 6(2):R1, 1997.
- [10] Homer D. Hagstrum. Auger ejection of electrons from tungsten by noble gas ions. *Phys. Rev.*, 96:325–335, Oct 1954.
- [11] Homer D. Hagstrum. Theory of auger ejection of electrons from metals by ions. *Phys. Rev.*, 96:336–365, Oct 1954.
- [12] G. Lakits, F. Aumayr, M. Heim, and H. Winter. Threshold of ion-induced kinetic electron emission from a clean metal surface. *Phys. Rev. A*, 42:5780–5783, Nov 1990.

- [13] Karl Jousten, editor. *Wutz Handbuch Vakuumtechnik*. Vieweg+Teubner, 2010.
- [14] Francis J. Alexander and Alejandro L. Garcia. The direct simulation monte carlo method. *Computers in Physics*, 11(6), 1997.
- [15] Peter Köhler. Design und inbetriebnahme eine strahlprofil-detektors und azimutale strahlanalyse an triebwerken des mikro newton rit-typs. Master's thesis, Justus-Liebig-Universität Gießen, 2010.
- [16] Hans Leiter. *Entwicklung, Bau und Test eines RIT15 "Breadboard Engineering Models"*. PhD thesis, Justus-Liebig-Universität Gießen, 2000.
- [17] N.T. Gladd J.P. Verboncoeur, A.B. Langdon. An object-oriented electromagnetic pic code. *Computer Physics Communications*, 87:199–211, 1995.
- [18] A. B. Langdon C. K. Birdsall. *Plasma Physics via Computer Simulation*. Taylor and Francis, 2005.
- [19] Roger W. Hockney and J. W. Eastwood. *Computer Simulation Using Particles*. Inst of Physics Pub, 1988.
- [20] V. Vahedi and M. Surendra. A monte carlo collision model for the particle-in-cell method: applications to argon and oxygen discharges. *Computer Physics Communications*, 87(1-2):179–198, 1995. <ce:title>Particle Simulation Methods</ce:title>.
- [21] H. Neuert. *Atomare Stoßprozesse*. Teubner, 1984.
- [22] C. B. Opal, W. K. Peterson, and E. C. Beaty. Measurements of secondary:electron spectra produced by electron impact ionization of a number of simple gases. *The Journal of Chemical Physics*, 55(8):4100–4106, 1971.
- [23] D. Tskhakaya, K. Matyash, R. Schneider, and F. Taccogna. The particle-in-cell method. *Contributions to Plasma Physics*, 47(8-9):563–594, 2007.
- [24] M. M. Turner. Kinetic properties of particle-in-cell simulations compromised by monte carlo collisions. *Physics of Plasmas (1994-present)*, 13(3):–, 2006.
- [25] Benjamin W. Yu and Steven L. Girshick. Modeling inductively coupled plasmas: The coil current boundary condition. *Journal of Applied Physics*, 69(2):656–661, 1991.
- [26] K.-I. You and N. S. Yoon. Discharge impedance of solenoidal inductively coupled plasma discharge. *Phys. Rev. E*, 59:7074–7084, Jun 1999.

- [27] Yoshinori Takao, Naoki Kusaba, Koji Eriguchi, and Kouichi Ono. Two-dimensional particle-in-cell monte carlo simulation of a miniature inductively coupled plasma source. *Journal of Applied Physics*, 108(9):093309, 2010.
- [28] R. Henrich and C. Heiliger. Three dimensional simulation of micro newton rits. In *33rd International Electric Propulsion Conference*, 2013.
- [29] Vahid Vahedi and G. DiPeso. Simultaneous potential and circuit solution for two-dimensional bounded plasma simulation codes. *J. Comput. Phys.*, 131:JOURNAL OF COMPUTATIONAL PHYSICS, 1997.
- [30] Shiming Yang and Matthias K. Gobbert. The optimal relaxation parameter for the {SOR} method applied to the poisson equation in any space dimensions. *Applied Mathematics Letters*, 22(3):325 – 331, 2009.
- [31] Ingo Wald. *Realtime Ray Tracing and Interactive Global Illumination*. PhD thesis, Computer Graphics Group Saarland University Saarbrücken, 2004.
- [32] R. Henrich. Entwicklung eines Simulationsprogramms zur Bestimmung des Dichteprofiles bei molekularer Strömung. Master's thesis, Justus-Liebig-Universität Gießen, 2009.
- [33] NVIDIA. *CUDA C PROGRAMMING GUIDE*, pg-02829-001_v5.5 edition, July 2013.
- [34] Horst W. Loeb, K.-H. Schartner, Stefan Weis, Davar Feili, and Bruno K. Meyer. Development of rit_microthrusters. In *55th International Astronautical Congress*, 2004.
- [35] G.A.Bird. *Molecular Gas Dynamics and the Direct Simulation of Gas Flows*. Oxford Science Publications, 1994.
- [36] R. Henrich, D. Feili, and C. Heiliger. Self-consistent simulation of the coupling between plasma and neutral gas in μ n-rit. In *32nd International Electric Propulsion Conference*, 2011.
- [37] Manfred Schäfer. *Plasmadiagnostik und Energiebilanzuntersuchung an dem HF-Ionentriebwerk RIT 10*. PhD thesis, Justus-Liebig-Universität Gießen, 1971.

Danksagung

Zunächst möchte ich an dieser Stelle Herrn Prof. Dr. Christian Heiliger meinen größten Dank aussprechen. Er hat mir die selbständige Bearbeitung dieses herausfordernden Themas ermöglicht und mich während meiner Diplomarbeit und später während der Promotion stets angespornt und unterstützt. Neben der sehr guten Betreuung bin ich ihm besonders dafür dankbar, dass er sich auf dieses Wagnis außerhalb seines Forschungsschwerpunktes eingelassen hat. Er hat mir beigebracht, dass jedes „geht nicht“ hinterfragt werden muss und sich dabei in ein „Was muss ich dafür tun?“ verwandelt.

Mein besonderer Dank gilt auch Herrn Prof. Dr. Bruno Meyer für sein Engagement auf dem Gebiet der Erforschung der Ionentriebwerke. Dieses Forschungsgebiet bildet die Hauptmotivation für diese Arbeit.

Herrn Prof. Dr. Peter Klar möchte ich für seinen Einsatz und die Leitung des LOEWE-Schwerpunktes „RITSAT“ bedanken, in dessen Rahmen diese Arbeit angesiedelt ist. Er trug maßgeblich zum Erfolg des Projektantrages bei.

Sehr dankbar bin ich Herrn Prof. Dr. Markus Thoma für das Ermöglichen des viermonatigen Gastaufenthalts am Max-Planck-Institut für extraterrestrische Physik sowie seine fortwährende Unterstützung. Die dort gewonnenen Erfahrungen waren eine große Stütze für meine Arbeit.

Herrn Dr. Michael Bachmann möchte ich an dieser Stelle ganz herzlich für seine große Unterstützung beim Austüfteln physikalischer, simulations- und „Kicker“-technischer Probleme danken.

Für die Einarbeitung in die PIC-Methode möchte ich Herrn Dr. Lujing Hou danken. Seine gute Betreuung hat den Einstieg in das Gebiet der Plasmasimulation erheblich vereinfacht und zur Überwindung der anfänglichen Verzweiflung erheblich beigetragen.

Herrn Dr. Michael Czerner möchte ich für seine Hilfe beim Einstieg in die Parallelisierung bedanken. Geduldig ertrug er einen „Abschuss“ des von ihm betreuten Clusters nach dem anderen.

Herrn Dr. Davar Feili danke ich für die Bereitstellung der genauen Plasma-parameter eines RITs.

Bei Herrn Waldemar Gärtner und Herrn Peter Köhler möchte ich mich für die sehr gute Zusammenarbeit bedanken. In zahlreichen Diskussionen erlangte ich ein besseres Verständnis des Plasmas in einem RIT sowie dessen Funktionsweise. Außerdem möchte ich besonders ihre Leidensbereitschaft hervorheben, wenn ich sie bei der einen oder anderen Kaffeepause mit unverständlichem Programmierkauderwelsch überschüttete.

Für das angenehme Arbeitsklima möchte ich mich herzlich bei der gesamten Arbeitsgruppe Heiliger bedanken. Neben den anregenden Diskussionen bleiben mir auch die eine oder andere heitere Erörterung gewisser Sachverhalte in positiver Erinnerung.

Der Justus-Liebig-Universität Gießen bin ich sehr dankbar für die finanzielle Unterstützung im Rahmen eines Graduiertenstipendiums.

Außerdem danke ich meinen fleißigen Korrekturlesern, die es sicher auch nicht immer einfach mit mir hatten.

Ganz besonders möchte ich mich bei meiner Familie bedanken, die mir in allen Lebenslagen zur Seite steht.

Eidesstattliche Erklärung

Hiermit erkläre ich, dass ich die vorliegende Arbeit mit dem Titel

„Development of a Plasma Simulation Tool for Radio Frequency Ion Thrusters“

selbständig und ohne fremde Hilfe angefertigt habe, keine anderen als die von mir angegebenen Quellen und Hilfsmittel verwendet habe, sowie die den benutzten Werken wörtlich oder inhaltlich entnommenen Stellen als solche kenntlich gemacht habe.

Gießen, den 21.11.2013

Robert Henrich