UiT The Arctic University of Norway

Faculty of Science and Technology
Department of Physics and Technology

Investigation into the photon interaction of small mesospheric dust and its impact on the ionospheric charge balance

Isabel Maria Sundström Nerland

FYS-3931 Master's Thesis in Space Physics 30 SP
June 2024

The cover illustration is a watercolor painting of noctilucent clouds over a Nordic landscape painted by my mother.

Abstract

This thesis investigates the charging mechanisms of nano-sized meteoric smoke particles (MSP) in the mesosphere, with a particular focus on the photoionization and photodetachment processes on MSP assuming different materials. The study begins with three theoretical chapters that lay the groundwork for understanding relevant topics like the atmospheric region in question, MSP, the solar spectrum, light scattering, Mie theory, and charging processes of MSP. Subsequently, computational simulations are made to explore how photoionization and photodetachment is affected by different parameters like MSP material properties and solar irradiation and how photoionization and photodetachment affect the charge states of MSP by varying MSP material and ionospheric properties.

The model calculations carried out in this thesis indicate that while photoionization and photodetachment do impact MSP charge states, their effect may not be as significant as previously thought. The models suggest that a fraction of sub-nanometer sized MSP between 15 and 40% can remain negatively charged, and that larger MSP with $r > 1$ nm have a chance of being positively charged. Material properties like refractive index and work function play a crucial role in affecting the photoionization of MSP, which subsequently affect the charge state of MSP, underscoring the need for more experimental data on the optical properties of possible MSP materials.

Acknowledgements

I would like to extend my heartfelt gratitude to my supervisor, Ingrid Mann. Thank you for introducing me to this fascinating topic and for your invaluable guidance and support throughout this journey. Your mentorship has been immensely appreciated.

I am also incredibly grateful for the support from my family. I could not have accomplished this without you. A special thank you to my mom for painting the cover illustration on such short notice; I will treasure and display the original artwork in my living room forever.

My sincere thanks go to my fellow students and friends in the master's office. Your support, encouragement, and the joy of our discussions and banter have meant the world to me. Special thanks to Elias for reading through my thesis and providing invaluable feedback.

Lastly, I would like to thank Aigen Li for the invaluable discussions and assistance with finding work functions and refractive indices. Your help has been greatly appreciated.

Contents

List of Figures

List of Tables

1

Introduction

A continuous influx of meteoric material enters the Earth's atmosphere daily. As meteors travel through the atmosphere, they heat up and ablate, with the ablated material coagulating to form meteoric smoke particles (MSP) (Megner, 2008). Over the last couple of decades, there has been increasing interest in studying MSP, especially due to their potential to participate in processes which can be used to monitor the effects of climate change in the middle atmosphere.

MSP can collect charged particles and are therefore important for the charge balance of the middle atmosphere. MSP are also believed to be the primary condensation nuclei for the formation of ice particles in the mesosphere (Dunker, 2018). These ice particles are thought to be the main component of noctilucent clouds (NLC) and polar mesospheric summer echoes (PMSE).

A study on the global transportation of dust in the mesosphere by Megner et al. (2008a) discovered that there are not enough MSP of adequate size in the mesosphere to account for the amount of ice particles needed for NLC. This model assumes that ice nucleation can only happen if the dust is larger than a critical radius, around 1 nm at mesospheric altitudes.

One proposed solution to this is the idea of nucleation of ice on charged mete-

oric smoke particles. It has been shown that if a dust particle is charged, the critical radius is reduced and, under certain conditions, it can even disappear. This means that ice particles can form on dust particles smaller than 1 nm, which are more abundant in the mesosphere. It is estimated that the number of MSP smaller than 1 nm in the mesosphere is on the order of 10^4 cm^{-3} (Knappmiller et al., 2011; Megner et al., 2008b). Thus, if a small fraction of these particles are charged, a sufficient number of condensation nuclei exists to account for the formation of NLC. It is however thought that negatively charged MSP is effectively neutralized by photodetachment by solar irradiation during daytime (Rapp, 2009). However, this study only investigated the photoionization of MSP down to a lower size limit of $r = 1$ nm.

This thesis aims to investigate how photoionization and photodetachment affects the charge state of MSP, with a focus on how photoionization and photodetachment of MSP varies with MSP material properties and solar irradiation. Additionally, I aim to investigate how photoionization and photodetachment affects the charge balance of MSP in the mesosphere. By computing the charge probability distribution of MSP for a size distribution ranging down to $r = 0.5$ nm and various possible MSP materials, I aim to determine whether sub-nanometer-sized MSP could be negatively charged during the daytime and if this population could possibly account for the number of condensation nuclei needed for noctilucent clouds.

To achieve this, I will carry out two types of model calculations. The first will investigate the variability of photoionization and photodetachment on MSP, and the second will investigate the effect of photoionization and photodetachment on the charge state of MSP. The first model will calculate the photoionization on MSP based on the study Rapp (2009) and varying different parameters such as MSP material properties and solar irradiation. The second model solves a set of kinetic rate equations describing the change in number density of MSP, electrons, and positive ions in the mesosphere, based on the approaches by Asmus et al. (2015) and Knappmiller et al. (2011). By running the differential equations until a steady state is reached, I can investigate the different charging mechanisms of MSP and obtain a charge probability distribution for the MSP.

Chapter 2 introduces the middle atmosphere and MSP, focusing on the mesosphere and mesospheric phenomena attributed to MSP. It also details MSP characteristics. Chapter 3 discusses the solar spectrum, explains how small particles scatter light, and how this scattering is quantified using Mie theory. Chapter 4 introduces MSP charging methods, emphasizing plasma particle

collection and photoionization and photodetachment. Chapter 5 presents the model developed to investigate the variability of photoionization of MSP and discusses the results. Chapter 6 describes the model created to investigate the charge probability of MSP and presents the results. The conclusion is given in chapter 7. Appendix A contains additional figures from chapter 6 and appendix B lists the code used to run the model calculations.

2

MSP and the middle atmosphere

This chapter introduces the middle atmosphere, focusing specifically on the mesosphere and mesospheric conditions, as well as the mesospheric phenomena of noctilucent clouds and polar mesospheric summer echoes. Subsequently, the chapter describes MSP formation, characteristics, and role as potential condensation nuclei for mesospheric ice particles.

2.1 The middle atmosphere

The term 'middle atmosphere' commonly refers to the stratosphere and mesosphere, two layers of the Earth's atmosphere. The stratosphere extends from about 10 to 50 kilometers above the Earth's surface. One of its defining features is the presence of the ozone layer, which absorbs a significant amount of the Sun's ultraviolet radiation. This layer is also characterized by a temperature inversion, where temperatures generally increase with altitude due to the absorption of solar radiation by ozone (Butchart, 2022).

Positioned directly above the stratosphere, the mesosphere is the third layer of the atmosphere, typically spanning altitudes from 50 to 85 kilometers. The boundaries of the mesosphere exhibit seasonal variations, with its upper limit, referred to as the mesopause, potentially extending up to 100 km (Venkat Ratnam et al., 2010). Notably, the mesosphere is the coldest region on Earth, with temperatures plummeting as low as −150° C during the summer months (Australian Antarctic Program, 2020). Measured vertical temperature variations for winter and summer in Fort Churchill, Canada (59°N) can be seen in fig. 2.1.

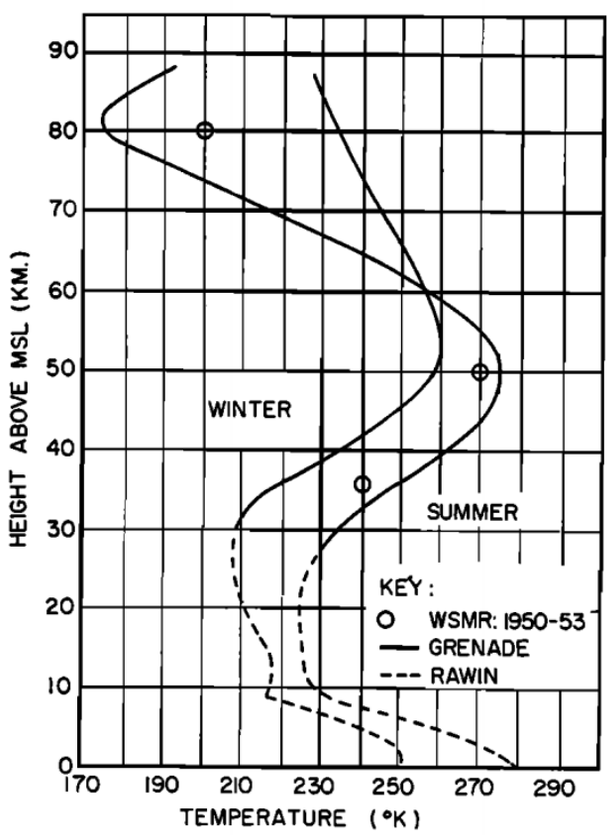

Figure 2.1: Vertical temperature data measured at Fort Churchill (59°N) in summer and winter. The solid and dashed curves are measured by rocket-grenade method and radiosondes, respectively. Open circles represent the average temperatures at White Sands (23°N). From Stroud et al. (1959) with description by Brekke (2013).

The ionosphere is the ionized part of the atmosphere and the lowest part of the ionosphere, called the D-region, extends down into the mesosphere

from about 90 km to 60 km. The dominant plasma particles in this region are electrons, O_2^+, and NO^+ above 85 km, while heavy hydrated ions are abundant below 80 km. Negative ions, such as O_2^-, can also be found at this altitude, but these are effectively neutralized by photodetachment during daytime due to the low electron affinity of O_2 (Brekke, 2013).

The mesosphere's high altitude presents a challenge for exploration. In-situ measurements can only be obtained through sounding rockets, which can only take measurements for a few minutes per launch, making them very costly. Consequently, in-situ observations of the mesosphere are sparse compared to the lower layers of the atmosphere. Although ground-based instruments, such as radar and LIDAR, along with satellite-based Earth observation, serve as alternative means to study the mesosphere, many phenomena and processes are still not fully understood (Megner, 2008).

2.2 Noctilucent clouds

Noctilucent clouds, often referred to as "night-shining clouds," are a rare meteorological phenomenon. They are the highest observed clouds on Earth, appearing at around $\sim$ 83 km in the summer mesosphere at high latitudes, and manifest as thin sheets of clouds. NLC consists of ice particles, which can only form at temperatures below $\sim$ 150 K ($\approx -123°$C) at mesospheric altitudes. These conditions are often met at the mesopause from June to mid-August at northern latitudes (and in winter at southern polar latitudes) (Kiliani et al., 2013).

NLC are thought to be caused by light scattering by ice particles. Ice particles of size $r > 20$ nm are visible to ground observers as well as optical instruments like cameras and LIDAR. Smaller ice particles play a major role in generating polar mesospheric summer echoes, which are often observed together with NLC and can be detected with radar (Kiliani et al., 2013).

Ice particle formation in the mesosphere presents a significant challenge due to the limited water content at this altitude. Supersaturation, a condition where the air contains more water vapor than it can hold at a given temperature and pressure, is required for ice particle formation. The mesosphere's low density necessitates extremely low temperatures for supersaturation. Despite its freezing conditions, homogeneous nucleation, which involves the spontaneous condensation of water vapor without pre-existing nuclei, is not realistically

feasible in the mesosphere. Instead, nucleation requires a surface or particle for the water vapor to condense onto. The abundance of meteoric smoke particles in the mesosphere makes them promising candidates as condensation nuclei (Megner et al., 2008b).

2.3 Polar mesospheric summer echoes

Polar mesospheric summer echoes (PMSE) are radar echoes that occur over a broad wavelength range of ~ 20 cm to 100 m, close to the mesopause (Megner, 2008). They occur at high latitudes in the summer, hence the name.

Ice particle formation starts around the region with the coldest temperatures, at approximately ~ 88 km. These particles subsequently grow and sediment. The ice particles are immersed in the D-region plasma, where electrons can attach to their surface, rendering them charged. Turbulence, created by the breaking of gravity waves in the 80 – 90 km region, transports the charged ice particles and creates small-scale structures in their distribution. To maintain charge neutrality, this induces small-scale structures in the electron number density. Consequently, irregularities arise in the radio refractive index, primarily determined by electron density at this altitude. When these irregularities are spaced at distances of half the radar wavelength, constructive interference of scattered wave amplitudes occurs. These strong echoes are observed on the ground as PMSE (Rapp and Lübken, 2004; Latteck and Bremer, 2017). Figure 2.2 summarizes the main characteristics of both NLC and PMSE.

2.4 Meteoric smoke particles

Meteoric smoke particles are of significant interest due to their potential to participate in processes which might provide insights into climate change impacts in the middle atmosphere. MSP play a role in various geophysical processes, such as the nucleation of mesospheric ice particles, mesosphere metal chemistry, maintaining D-region charge balance, heterogeneous water vapor formation, and stratospheric cloud particle nucleation, which is important for ozone hole formation (Rapp et al., 2012a).

Annually, an estimated 16-40 kilotons of meteoric material enter the Earth's atmosphere (Bardeen et al., 2008). As meteoroids enter the atmosphere, they

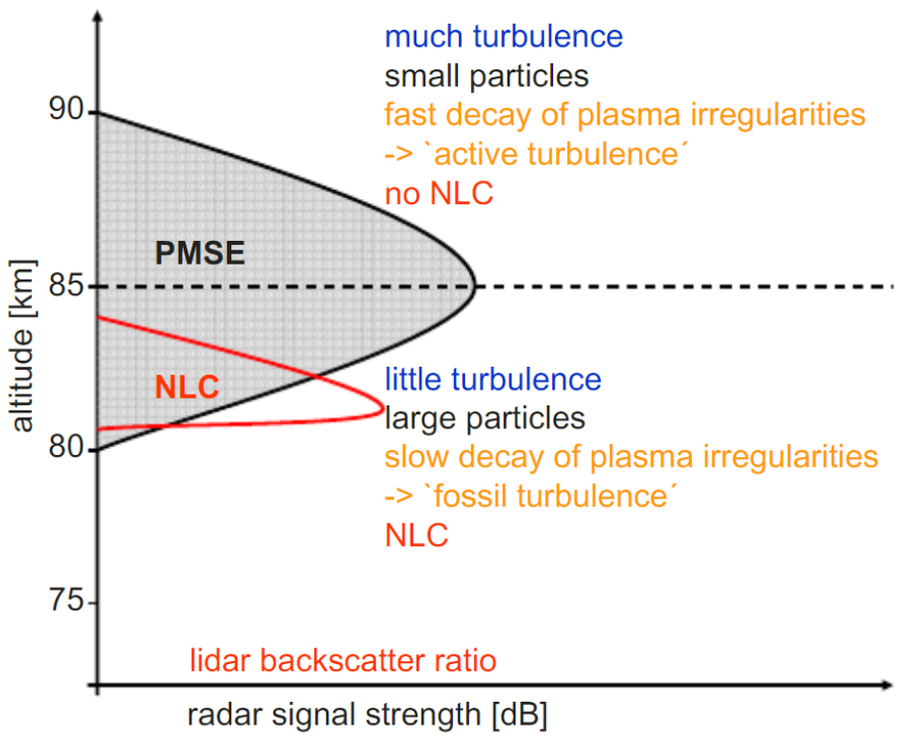

Figure 2.2: Schematic of the altitude distribution of PMSE (black line, gray shading) and NLC (red line) and the relevant physical mechanisms at different altitudes from Rapp and Lübken (2004).

undergo atmospheric drag and heating, leading to ablation, where the outer layers vaporize. This meteoric smoke then coagulates and sediment. This process results in the formation of nanometer-sized MSP in the lower thermosphere and upper mesosphere. These particles are composed primarily of silicates, oxides, and metals from the original meteoroid (Hunten et al., 1980; Megner et al., 2006; Bardeen et al., 2008). Figure 2.3 illustrates MSP formation processes.

The precise composition of MSP remains uncertain. Meteoric ablation introduces elements like iron (Fe), magnesium (Mg), and silicon (Si) into the atmosphere. Hematite has been regarded as the most likely candidate for a while (Bohren and Olivero, 1984). Laboratory experiments suggest these elements oxidize to form compounds such as olivine ($Fe_{2x}Mg_{1-x}SiO_4$, $x = 0 - 1$) and pyroxene ($Fe_xMg_{1-x}SiO_3$, $x = 0 - 1$) (Saunders and Plane, 2011; Plane et al., 2023). Optical measurements indicate that meteoric remnants, though not necessarily smoke, may consist of olivine and hematite (Klekociuk et al.,

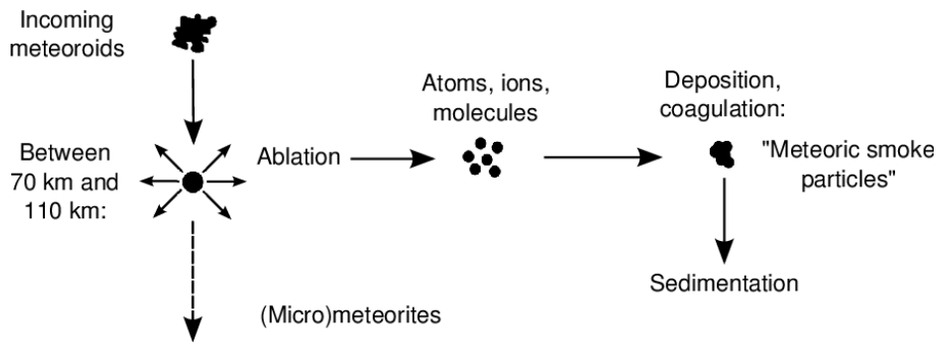

Figure 2.3: Processes involved in the formation of meteoric smoke particles. Illustration borrowed from T. Dunker, adapted from J. Gumbel, Meteorologiska Institutionen, Stockholms Universitet, Sweden (Dunker, 2018).

2005; Bohren and Olivero, 1984). Extinction measurements from the SOFIE satellite suggest MSP consist of magnetite (Fe_3O_4), wüstite (FeO), magnesiowüstite ($Mg_xFe_{1-x}O$, $x = 0 - 0.6$), or iron-rich olivine ($Mg_{2x}Fe_{2-2x}SiO_4$, $x = 0.4 - 0.5$) (Hervig et al., 2017). More recent SOFIE studies indicate that MSP likely consist solely of iron-rich olivine ($Mg_{0.8}Fe_{1.2}O_4$) (Hervig et al., 2021). While the suggestion of MSP consisting of silicates is not new, recent findings have strengthened their candidacy over metal oxides like hematite as the primary MSP material (Rapp, 2009; Knappmiller et al., 2011; Plane, 2003; Bohren and Olivero, 1984).

MSP are believed to be important condensation nuclei for ice particles in the middle atmosphere. The critical radius, which is the minimum particle size necessary for spontaneous condensation, is estimated to be around 1 nm under mesospheric conditions for a neutral MSP (Gumbel and Megner, 2009). Therefore, MSP must exceed this radius to act as effective condensation nuclei.

One-dimensional models, such as the one by Hunten et al. (1980), have traditionally suggested that MSP are available as condensation nuclei in sufficient quantities. However, a two-dimensional model by Megner et al. (2008b) indicate that MSP are transported away from the summer pole before they grow beyond the critical radius. This leaves only about 10 particles per cm^{-3} with a radius exceeding 1 nm in the polar summer mesosphere, which is about two orders of magnitude lower than expected number densities of ice particles needed for NLC (Gumbel and Megner, 2009). Satellite observations (Hervig et al., 2009) support the idea of effective MSP transport, suggesting a smaller pool of MSP able to serve as condensation nuclei.

If neutral MSP are insufficient for nucleation, charged MSP might play a critical role. Notably, when a particle is charged the critical radius for condensation is reduced, or may even vanish entirely. This has the potential of making the entire population of approximately 10^4 MSP available as condensation nuclei (Gumbel and Megner, 2009). Negatively charged particles have been thought to be candidates for condensation nuclei due to the availability of free electrons and sub-nanometer MSP in the mesosphere. However, a study by Rapp (2009) suggested that photodetachment caused by solar illumination could effectively neutralize MSP particles during the daytime.

3

The solar spectrum and light scattering of small particles

The previous chapter highlighted the importance of investigating the charging of very small dust particles, such as MSP, through photoionization. This process depends significantly on the illuminating solar radiation and the scattering process of light by small particles.

The main purpose of this chapter is to describe the mechanisms through which small particles scatter and absorb energy from solar irradiation. Section 3.1 discusses the characteristics of the solar spectrum. Section 3.2 outlines the principles governing the scattering and absorption of light by small particles. Subsequently, Section 3.3 describes Mie scattering theory, which will be employed later to compute the absorption cross-sections of MSP.

3.1 The solar spectrum

The Sun emits electromagnetic radiation across a wide range of wavelengths, approximated by a blackbody at 5776 K. Solar irradiance is typically measured

in watts per square meter and varies with incident photon wavelength. The solar constant is the total radiation energy received from the Sun per unit of time per unit of area at 1 AU and is approximately 1366 W/m^2 (Zirin, 2012). Figure 3.1 shows this wavelength dependence, illustrating the typical solar irradiance at 1 AU. Although it closely resembles a blackbody spectrum, notable deviations occur below 600 nm. One such deviation is the prominent Lyman-α line at approximately 121.6 nm in the far ultraviolet (Gunár et al., 2020).

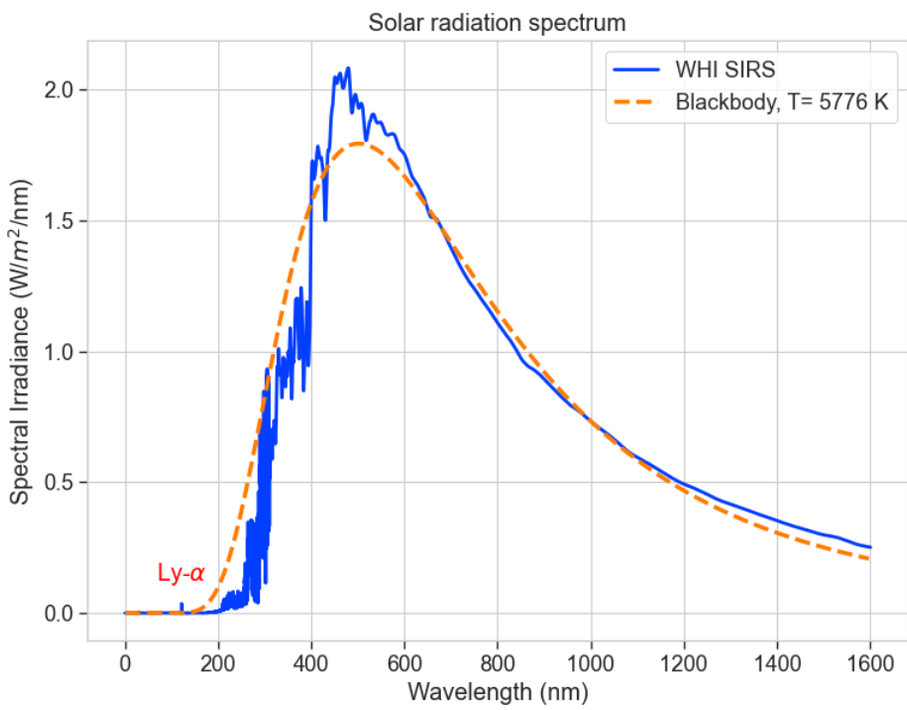

Figure 3.1: Measured solar irradiance at 1 AU averaged over April 10 - April 16, 2008 and blackbody with T = 5776 K plotted for comparison. Plotted with data from the 2008 Whole Heliosphere Interval (WHI) Solar Irradiance Reference Spectra (SIRS) described in Woods et al. (2009). This data was accessed via the LASP Interactive Solar Irradiance Datacenter (LISIRD) on 05.03.2024.

The solar spectrum can be divided into several regions, which are listed in table 3.1 (Woods et al., 2009). Approximately 50% of the Sun's radiant energy lies in the infrared region (700 nm), 40% in the visible region (400 – 700 nm), and 10% in the UV region (< 400 nm) (Fu, 2003).

Solar irradiance fluctuates with solar activity, which follows an 11-year solar

Table 3.1: Regions of the solar spectrum from Woods et al. (2009).

Region	**Wavelength range** (nm)
X-ray ultraviolet (XUV)	0 – 30
Extreme ultraviolet (EUV)	30 – 120
Far ultraviolet (FUV)	120 – 200
Middle ultraviolet (MUV)	200 – 300
Near ultraviolet (NUV)	300 – 400
Visible (VIS)	400 – 800
Near infrared (NIR)	800 – 2400

cycle characterized by variations in sunspots, solar flares, and irradiance levels. During periods of high solar activity, irradiance increases slightly, while it decreases during low activity. Observations suggest that the solar constant varies on the order of 0.1% with the solar cycle. However, the solar variability is larger in the UV region, which is notable as most of this radiation is absorbed in the atmosphere above 80 km (Fu, 2003; Rees, 1989).

3.2 The absorption of light by small particles

This section and section 3.3 were written as part of my efforts to understand the scattering process and Mie theory. Rather than providing a comprehensive review, these sections offer a summary of the key steps involved in deriving the refractive index and the scattering and absorption properties using Mie theory.

When a parallel monochromatic beam of light travels through a vacuum, its intensity and polarization state remain unchanged. However, when encountering a small particle, the particle may cause several effects to the beam. The particle may absorb some of the beam's energy, converting it into other forms, such as heat, in a process known as absorption. Additionally, the particle may scatter some of the incident energy in all directions at the same frequency as the incident beam, known as elastic scattering. The combined effects of absorption and scattering lead to a reduction in the intensity of the original light beam, which is referred to as extinction (Bohren and Huffman, 1983).

From an electromagnetic standpoint, the incident beam is an oscillating electromagnetic plane wave, while the particle consists of numerous discrete

elementary charges. The oscillating electromagnetic field of the incident wave induces oscillations in these charges, causing them to emit secondary electromagnetic waves. The total elastically scattered field results from the superposition of these secondary waves. This is illustrated in fig. 3.2. However, computing the total scattered field directly is impractical for micrometer-sized particles due to the complexity of interactions between the many charges. Instead, macroscopic electromagnetics treats the particle as a macroscopic body with a specific refractive index distribution, allowing the scattered field to be computed by solving Maxwell's equations for macroscopic electromagnetic fields subject to appropriate boundary conditions (Mishchenko et al., 2002).

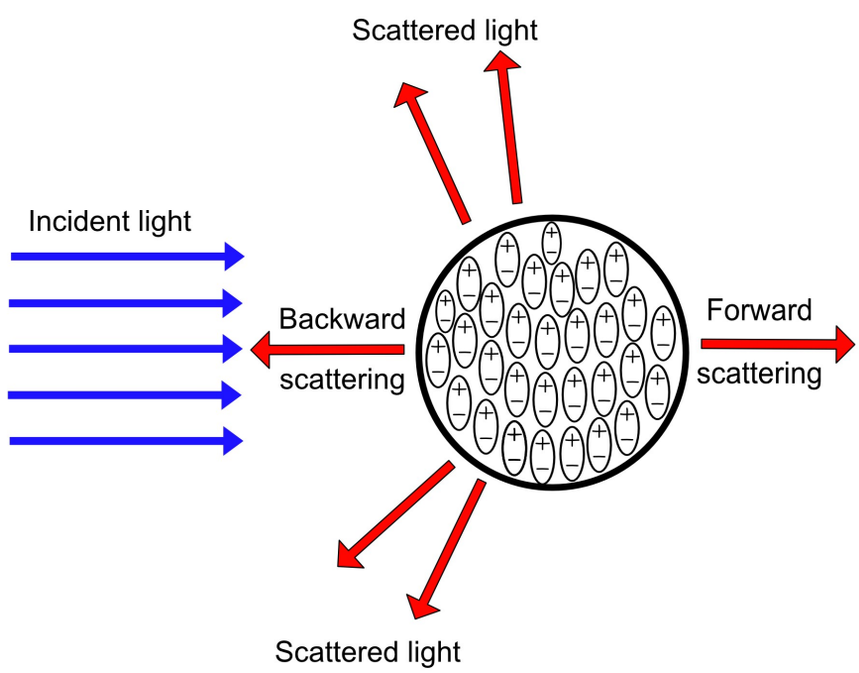

Figure 3.2: A conceptual depiction of light scattering by a small particle composed of numerous discrete elementary charges. This illustration is adapted from Li (2008).

The refractive index is a fundamental optical constant derived from Maxwell's equations, which in SI units are:

$$\nabla \cdot \boldsymbol{D} = \rho \tag{3.1}$$

$$\nabla \cdot \boldsymbol{H} = \boldsymbol{J} + \frac{\partial \boldsymbol{D}}{\partial t} \tag{3.2}$$

$$\nabla \cdot \boldsymbol{B} = 0 \tag{3.3}$$

$$\nabla \cdot \boldsymbol{E} = -\frac{\partial \boldsymbol{B}}{\partial t} \tag{3.4}$$

where t is time, $\boldsymbol{E}$ is the electric field, $\boldsymbol{B}$ is the magnetic induction, and ρ and $\boldsymbol{J}$ are the macroscopic free charge density and current density, respectively. The electric displacement $\boldsymbol{D}$ and the magnetic field $\boldsymbol{H}$ are defined by (Mishchenko et al., 2002):

$$\boldsymbol{D} = \varepsilon_0 \boldsymbol{E} + \boldsymbol{P} \tag{3.5}$$

$$\boldsymbol{H} = \frac{\boldsymbol{B}}{\mu_0} - \boldsymbol{M} \tag{3.6}$$

where $\boldsymbol{P}$ is the electric polarization, $\boldsymbol{M}$ is the magnetization, ε_0 is the permittivity of free space, and μ_0 is the permeability of free space. The continuity equation, derived from Maxwell's equations, is:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot \boldsymbol{J} = 0 \tag{3.7}$$

The constitutive relations are given by (Li, 2008):

$$\boldsymbol{J} = \sigma \boldsymbol{E} \tag{3.8}$$

$$\boldsymbol{B} = \mu \boldsymbol{H} \tag{3.9}$$

$$\boldsymbol{P} = \varepsilon_0 \chi \boldsymbol{E} \tag{3.10}$$

where σ is the electric conductivity, χ is the electric susceptibility, and μ is the magnetic permeability. Assuming time-dependent fields and substituting the constitutive relations into Maxwell's equations, we obtain the following set of equations:

$$\nabla \cdot (\varepsilon \boldsymbol{E_c}) = 0 \tag{3.11}$$

$$\nabla \times \boldsymbol{E_c} = i\omega\mu \boldsymbol{H}_c \tag{3.12}$$

$$\nabla \cdot \boldsymbol{H}_c = 0 \tag{3.13}$$

$$\nabla \times \boldsymbol{H}_c = -i\omega\varepsilon \boldsymbol{E}_c \tag{3.14}$$

where the complex permittivity ε is defined as:

$$\varepsilon = \varepsilon_0(1 + \chi) + i\frac{\sigma}{\omega} \tag{3.15}$$

We now seek a plane-wave solution to eqs. (3.11) to (3.14). Only electromagnetic fields that satisfy Maxwell's equations are physically realizable. Therefore, we look for conditions under which the following electromagnetic plane waves are compatible with Maxwell's equations:

$$\boldsymbol{E}_c = \boldsymbol{E}_0 \exp(i\boldsymbol{k} \cdot \boldsymbol{x} - i\omega t),\ \boldsymbol{H}_c = \boldsymbol{H}_0 \exp(i\boldsymbol{k} \cdot \boldsymbol{x} - i\omega t) \tag{3.16}$$

where $\boldsymbol{E}_0$ and $\boldsymbol{H}_0$ are constant vectors. Here, the angular frequency ω is given by the wavenumber $\boldsymbol{k}$ and the speed of light c as $\frac{\omega}{k} = c$. In order to obtain conditions for possible wave vectors, we substitute eq. (3.16) into eqs. (3.11) to (3.14), which yields:

$$\boldsymbol{k} \cdot \boldsymbol{E}_0 = 0 \tag{3.17}$$

$$\boldsymbol{k} \cdot \boldsymbol{H}_0 = 0 \tag{3.18}$$

$$\boldsymbol{k} \times \boldsymbol{E}_0 = \omega\mu \boldsymbol{H}_0 \tag{3.19}$$

$$\boldsymbol{k} \times \boldsymbol{H}_0 = -\omega\varepsilon \boldsymbol{E}_0 \tag{3.20}$$

From eq. (3.17) and eq. (3.18), we see that $\boldsymbol{k}$ is perpendicular to both $\boldsymbol{E}_0$ and $\boldsymbol{H}_0$. We can also see from eq. (3.19) and eq. (3.20) that $\boldsymbol{E}_0$ and $\boldsymbol{H}_0$ are perpendicular to each other. Taking the vector product of both sides of eq. (3.19) with $\boldsymbol{k}$,

$$\boldsymbol{k} \times (\boldsymbol{k} \times \boldsymbol{E}_0) = \omega\mu \boldsymbol{k} \times \boldsymbol{H}_0 = -\omega^2 \varepsilon\mu \boldsymbol{E}_0 \tag{3.21}$$

and using the triple product expansion:

$$\boldsymbol{A} \times (\boldsymbol{B} \times \boldsymbol{C}) = \boldsymbol{B}(\boldsymbol{A} \cdot \boldsymbol{C}) - \boldsymbol{C}(\boldsymbol{A} \cdot \boldsymbol{B}) \tag{3.22}$$

together with eq. (3.17), we obtain:

$$\boldsymbol{k} \cdot \boldsymbol{k} = \omega^2 \varepsilon \mu \tag{3.23}$$

So far, we have shown that the plane waves in eq. (3.16) are compatible with the Maxwell equations as long as $\boldsymbol{k}$, $\boldsymbol{E}_0$ and $\boldsymbol{H}_0$ are perpendicular and $\boldsymbol{k}$ satisfy eq. (3.23). From eq. (3.23), we can define k:

$$k = \frac{\omega N}{c} \tag{3.24}$$

where c is the speed of light in vacuum and N is the complex refractive index $N = c\sqrt{\mu\varepsilon}$. By inserting eq. (3.15) for ε_0, we can rewrite N in terms of a real part n and an imaginary part κ,

$$N = n + i\kappa \tag{3.25}$$

where n and κ are non-negative. This is the common for of the refractive index. The real part, n, describes how much the material slows down the speed of light compared to its speed in a vacuum, indicating the degree of refraction. The imaginary part, κ, quantifies how much of the light is absorbed by the material as it passes through it. A higher κ value indicates greater absorption of light by the material. The refractive index of a material is typically measured experimentally.

3.3 Mie scattering theory

The formal solution to Maxwell's equations for a sphere of arbitrary size and refractive index has been available for many years. This solution was formulated by Gustav Mie in 1908 and independently by Peter Debye in 1909.

Although determining who was first is challenging, we will refer to this theory by its most common name, Mie theory (Bohren and Huffman, 1983).

Mie theory is used to determine the scattering, absorption, and extinction properties of a spherical particle exposed to an incident electromagnetic wave. The solution is characterized by the Mie coefficients a_n and b_n, which are derived from Maxwell's equations and the boundary conditions at the surface of the sphere. While the derivation is lengthy and complex, we will recap the main steps based on the approach by Bohren and Huffman (1983).

Firstly, the incident plane wave, scattered field, and internal field of the particle are expressed using vector spherical harmonics. The incident field $(\boldsymbol{E}_i,\ \boldsymbol{H}_i)$ is expanded using vector spherical harmonics, the internal field of the particle $(\boldsymbol{E}_1,\ \boldsymbol{H}_1)$ is expanded using spherical Bessel functions j_n, and the scattered field $(\boldsymbol{E}_s,\ \boldsymbol{H}_s)$ is expanded using Hankel functions of the first kind $h_n^{(1)}$.

Next, we apply the boundary conditions, which state that the tangential components of the electric and magnetic fields must be continuous across the boundary separating media of different properties. These conditions can be written as:

$$\boldsymbol{E}_{i,\,\text{tangential}} + \boldsymbol{E}_{s,\,\text{tangential}} = \boldsymbol{E}_{1,\,\text{tangential}}$$

$$\boldsymbol{H}_{i,\,\text{tangential}} + \boldsymbol{H}_{s,\,\text{tangential}} = \boldsymbol{H}_{1,\,\text{tangential}}$$

We expand these boundary conditions in terms of spherical harmonics and equate the coefficients of corresponding harmonics on both sides of the equations. This results in a system of linear equations for each mode n. The resulting equations involve spherical Bessel functions j_n, spherical Hankel functions $h_n^{(1)}$, and their derivatives. The Mie coefficients a_n and b_n are then obtained from these equations and are given by:

$$a_n = \frac{\mu m^2 j_n(mx)[x j_n(x)]' - \mu_1 j_n(x)[mx j_n(mx)]'}{\mu m^2 j_n(mx)[x h_n^{(1)}(x)]' - \mu_1 h_n^{(1)}(x)[mx j_n(mx)]'} \tag{3.26}$$

$$b_n = \frac{\mu_1 j_n(mx)[x j_n(x)]' - \mu j_n(x)[mx j_n(mx)]'}{\mu_1 j_n(mx)[x h_n^{(1)}(x)]' - \mu h_n^{(1)}(x)[mx j_n(mx)]'} \tag{3.27}$$

where the relative refractive index m and the size parameter x are given by:

$$m = \frac{k_1}{k} = \frac{N_1}{N} \tag{3.28}$$

$$x = kr = \frac{2\pi a}{\lambda} \tag{3.29}$$

Here, N_1 and N are the refractive indices of the particle and the medium, respectively; k_1 is the wave number inside the particle, and λ is the wavelength of the incoming wave.

The Mie coefficients might look intimidating, but we can break the components down further. The spherical Bessel functions j_n represent the radial part of the wave inside the sphere, and the spherical Hankel functions $h_n^{(1)}$ represent the radial part of the outgoing wave outside the sphere. The relative refractive index adjusts for the different refractive indices of the particle and the surrounding medium, while the size parameter relates the size of the particle to the wavelength of the incident light.

The scattering, absorption, and extinction cross sections C_{sca}, C_{abs}, and C_{ext} are measures of the likelihood of scattering, absorption, and total extinction of the incident light by the particle. The scattering cross section is derived by integrating the scattered power over all directions. The extinction cross section represents the total loss of incident wave energy due to both scattering and absorption and is derived from the forward-scattered field, which relates to the interference between the incident and scattered fields. The resulting scattering and extinction cross sections are:

$$C_{sca} = \frac{2\pi}{k^2} \sum_{n=1}^{\infty} (2n+1)(|a_n|^2 + |b_n|^2) \tag{3.30}$$

$$C_{ext} = \frac{2\pi}{k^2} \sum_{n=1}^{\infty} (2n+1)\Re(a_n + b_n) \tag{3.31}$$

where k is the wave number outside the particle. C_{abs} can be obtained by subtracting C_{sca} from C_{ext}:

$$C_{abs} = C_{ext} - C_{sca} \tag{3.32}$$

The cross sections can be normalized by the geometric cross section $G = \pi a^2$ to obtain the efficiency coefficients:

$$Q_{ext} = \frac{C_{ext}}{G} \tag{3.33}$$

$$Q_{sca} = \frac{C_{sca}}{G} \tag{3.34}$$

$$Q_{abs} = \frac{C_{abs}}{G} \tag{3.35}$$

These are dimensionless measures of the particle's effectiveness in scattering, absorbing, and extinguishing the incident light.

The Mie coefficients, a_n and b_n, and the resulting scattering, absorption, and extinction cross sections are often demanding to calculate and require significant computational effort to evaluate. Consequently, these coefficients are typically calculated using numerical methods implemented in computer algorithms. There are several established computational tools and libraries available that efficiently perform these calculations.

4

Charging of dust

This chapter describes and discusses relevant methods for charging MSP in the mesosphere. While there are other charging methods, this chapter focuses on the ones used in the model later. Section 4.1 presents the charging of MSP by the collection of plasma particles. Section 4.2 describes how photoionization and photodetachment can charge and remove charges from MSP.

4.1 Charging by collection of plasma particles

Plasma particles can attach to MSP. The rates at which plasma particles attach to MSP depend on the charge of the MSP relative to the surrounding plasma.

The plasma capture rates we will consider were first derived by Natanson (1960), who rigorously calculated the ion capture rates for neutral and charged sub-micron aerosol particles. These equations were later adjusted by Rapp (2000) to fit the collection of charged plasma particles by MSP.

Natanson's analysis considered factors such as the Coulomb potential of the net MSP charge and the induced image potential (Jensen and Thomas, 1991;

Natanson, 1960). The interaction between an ion (or more generally, a point charge including electrons) and an MSP with Z elementary charges is described by the following potential (Rapp, 2000):

$$\phi = \frac{Ze^2}{r} + \frac{e^2 a^3}{2r^2(r^2 - a^2)} \tag{4.1}$$

Here, e is the elementary charge and r is the distance between the MSP center and the ion, and a is the MSP radius. The first term describes the Coulomb force between an ion and an MSP with a net charge of Z elementary charges. The second term describes the induced image force of the MSP with the ion (Rapp, 2000). Using basic gas kinetics, Natanson found the following expression for the attachment rate α:

$$\alpha = \frac{\gamma\mu\pi\rho^2 c_s e^{\phi(\rho+\lambda)/k_B T}}{1 + (\gamma\mu\rho^2 c_s/4D)\int_{\rho+\lambda}^{\infty}(1/r^2)e^{(\phi(\rho+\lambda)-\phi(r))/k_B T}dr} \tag{4.2}$$

where c_s is the mean thermal ion velocity, k_B is Boltzmann's constant, T is the ion temperature, D is the ion diffusion coefficient, ρ is the radius of a sphere around the MSP center in which the ions can be captured, γ is a coefficient expressing the change in the number of collisions of the ions with the sphere ρ due to the presence of interaction forces, μ is the probability that an ion approaching closer than ρ is captured, and λ is the mean free path of the ion (Rapp, 2000; Natanson, 1960).

Equation (4.2) must be solved numerically as the integral in the denominator cannot be solved analytically when both the Coulomb and image potentials are considered. If $a \ll \lambda$, which is typical for mesospheric particles where $\lambda \approx 1$ cm and $a \leq 100$ nm (Rapp, 2000), the following analytical approximations can be obtained:

$$\alpha^s_{Z>0} = \pi a^2 c_s \left[1 + \frac{16}{81}\frac{\lambda}{a}\left(\frac{|Z|e^2}{k_B T\lambda}\right)\right] \text{ (attractive interaction)} \tag{4.3}$$

$$\alpha^s_{Z=0} = \pi a^2 c_s \left(1 + \sqrt{\frac{\pi e^2}{2k_B T a}}\right) \text{ (neutral interaction)} \tag{4.4}$$

$$\alpha^s_{Z<0} = \pi a^2 g^2 c_s \exp\left[-\frac{|Z|e^2}{g k_B T a}\left(1 - \frac{1}{2g(g^2-1)|Z|}\right)\right] \text{ (repulsive interaction)} \tag{4.5}$$

where g is a dimensionless factor defined by the distance where the force changes sign between the repulsive Coulomb force and attracting image force (Knappmiller et al., 2011), and can be calculated from:

$$|Z| = \frac{2g^2 - 1}{g(g^2-1)^2} \tag{4.6}$$

To validate the accuracy of the approximations, Rapp (2000) calculated electron capture rates numerically from eq. (4.2) and analytically from eqs. (4.3) to (4.5). Rapp found that for neutral and negatively charged MSP, the numerical and analytical results agreed well. For positively charged particles with $a \geq 50$ nm, the electron capture rate was lower compared to a neutral particle of the same size, which is unphysical. However, this is of little concern to us as we are primarily interested in smaller particles.

While the consistency of numerical and analytical results does not guarantee that Natanson's rates are correct, Rapp compared the measured charge distribution of sub-micron silver particles with the distribution calculated using eq. (4.2) and found agreement within 5% (Rapp, 2000).

4.2 Photoionization and photodetachment

Interactions between solar photons, particularly in the ultraviolet and visible regions, and MSP can result in the ejection of an electron. When a photon impacts an MSP, some of its energy is absorbed. If this energy surpasses a certain threshold, it can overcome the forces binding electrons within the particle. In photoionization, an electron is ejected from within a neutral or positively charged MSP, whereas in photodetachment, an electron is released from the surface of a negatively charged MSP.

Before describing the rate equations for photoionization and photodetachment, it is essential to introduce some key concepts. Firstly, the work function of a

material signifies the amount of energy required to remove an electron from the solid. Similarly, but notably different, the electron affinity of a material denotes the energy needed to remove an electron from its surface. Both are typically expressed in electron volts. Importantly, the work function is used for determining the photoionization of a material and the electron affinity is used in photodetachment. The work function of a material is usually higher than its electron affinity, as it requires more energy to remove an electron from inside the solid compared to removing an electron from its surface.

The photoionization and photodetachment rates can be calculated using the following integral equation (Rapp, 2009):

$$\beta = \int_0^{\lambda_0} F(\lambda) \cdot \sigma(a, N, \lambda) \cdot Y(\lambda) \cdot d\lambda \tag{4.7}$$

where $F(\lambda)$ is the solar photon flux at wavelength λ, and $\sigma(a, N, \lambda)$ is the absorption cross section of the MSP, dependent on the MSP radius a, the material's refractive index $N = n + i\kappa$, and the solar photon wavelength λ. $Y(\lambda)$ is the quantum yield, which is the probability of an electron being ejected after a photon impact. The integral is bounded by the cut-off wavelength λ_0, determined by the material's work function or electron affinity. The cut-off wavelength λ_0 can be calculated from the following equation:

$$\lambda_0 = \frac{hc}{E^*} \tag{4.8}$$

where h is Planck's constant, c is the speed of light, and E^* is the work function or electron affinity of the material.

According to Rapp (2009), eq. (4.7) describes photoionization and photodetachment as a single process, assuming that the photon is initially absorbed by the particle, and if its energy is higher than the work function or electron affinity, an electron is emitted with the probability Y.

5

Modeling the variability of photoionization of MSP

There are many challenges in calculating the photoionization of MSP, primarily due to the unknown composition and material properties of MSP. It is therefore important to investigate how these different properties can affect the photoionization rate.

This chapter focuses on examining the variability of photoionization of MSP with different materials, work functions, quantum yields, and solar photon flux. Section 5.1 outlines the input data collection, programming procedures, and relevant functions, packages, and libraries. Section Section 5.2 presents and discusses the results of the program. The program calculating the photoionization and photodetachment will be used for charge state calculations presented later.

5.1 Calculation and computation

5.1.1 Program description

This section describes how the various variability calculations were implemented in the program. The program was made in Python. All functions are organized into a separate file named *functions.py*, which is then imported into the primary program file *photoionization.py*. This structure allows for a clean separation of concerns, where the main program files handle the execution flow and plotting, while the *functions.py* file contains the core computational functions. It is worth noting that there can be some confusion regarding the units of wavelength used in different functions. To maintain consistency, all functions take input wavelengths in nanometers, except for the *photoionization()* function, which uses microns. This study examines the effect of:

- Different refractive indices
- Far ultraviolet
- Work function
- Quantum yield
- Solar photon flux

Absorption cross section

The *mie(m,x)* function from the *miepython* library, explained in section 5.1.4, calculates the extinction and scattering efficiencies defined by eqs. (3.33) and (3.34). This function takes the refractive index m as defined by eq. (3.25) and the size parameter x defined by eq. (3.29). Using these equations as well as eq. (3.32), we get the following equation for the absorption cross section C_{abs}:

$$C_{abs} = (Q_{ext} - Q_{sca}) \cdot G \tag{5.1}$$

where Q_{ext} is the extinction efficiency, Q_{sca} is the scattering efficiency, and G

is the geometric cross section of the MSP.

This equation has been implemented in the function *abs_cross(r, material)*, which takes the MSP radius r and refractive index in a nested list as *material*, in appendix .2. Figure 5.1 shows the absorption cross sections for a particle with a radius of 1 nm for the materials listed in table 5.1.

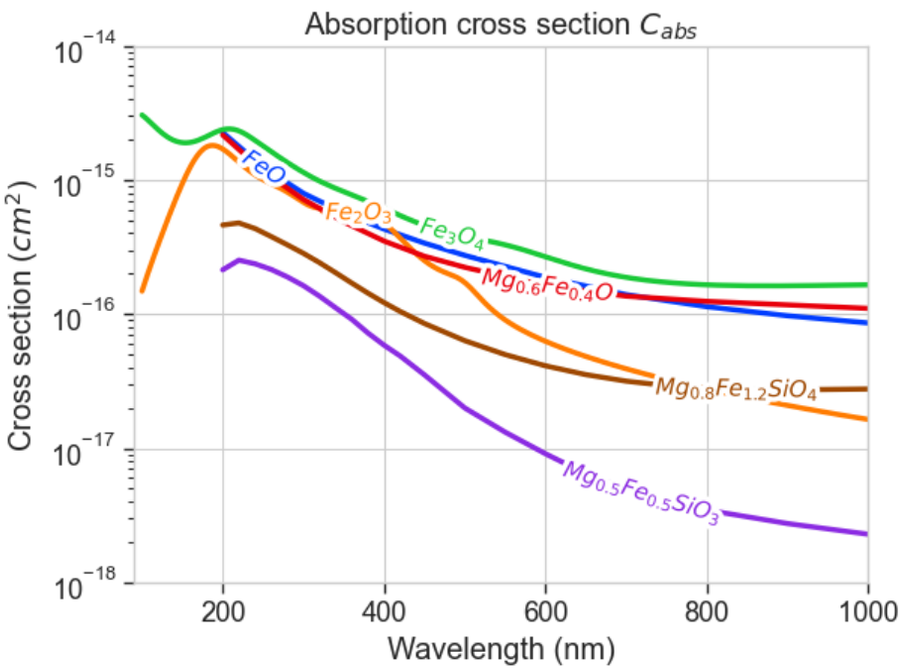

Figure 5.1: Absorption cross section for a particle of radius $r = 1$ nm of the materials in table 5.1.

Photoionization

In order to calculate the photoionization, eq. (4.7) is implemented into the function *photoionization(material, r, Y, E)*. This function takes the following inputs: *material*, a list of the refractive indices and their respective wavelengths; r, the radius of the MSP in nanometers; Y, the quantum yield; and E, the work function or electron affinity in electron volts.

In order to integrate the absorption cross section and the solar photon flux, the lists containing these variables must be matched to the same wavelengths. The solar photon flux has a much higher spectral resolution, which makes it is easier to adjust it to fit the absorption cross section. The photoionization function

takes the wavelength range of a refractive index and finds the closest solar spectrum wavelengths. A new list of solar irradiance is then made from the corresponding values. The irradiance is converted into solar photon flux, and the cut-off wavelength is calculated from the given work function or electron affinity. Integration is then carried out from the earliest possible wavelength to the cut-off wavelength using the *SciPy* function *integrate.simpson()*.

To validate the matching wavelength approach, fig. 5.2 shows the difference between the refractive index wavelength used and the closest solar spectrum wavelength. For wavelengths below 700 nm the error is less than 1 nm.

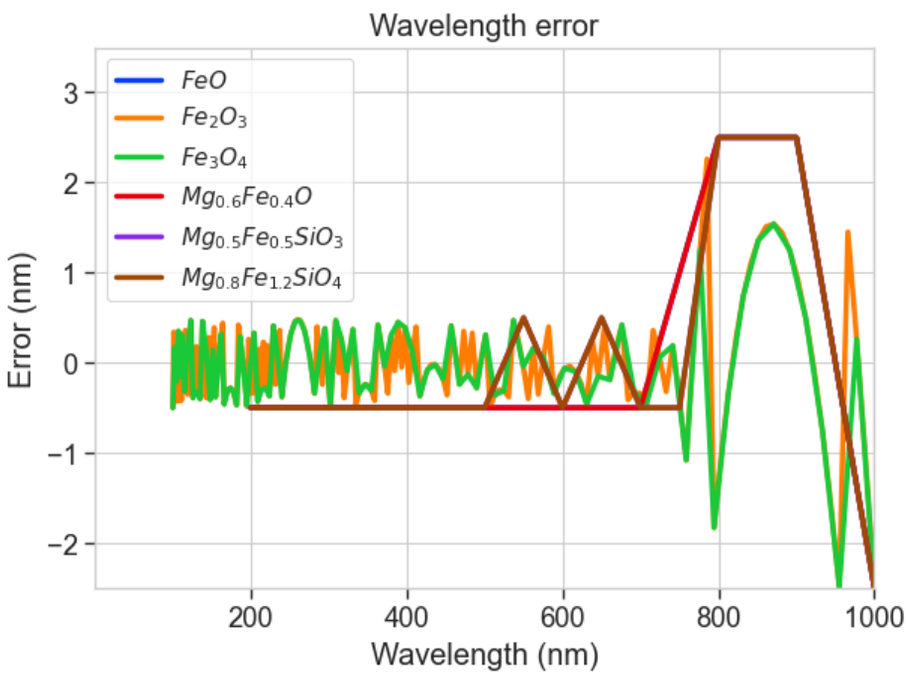

Figure 5.2: Difference between the refractive index wavelength used and the solar spectrum wavelength.

Effect of the far ultraviolet

The far ultraviolet (FUV) solar spectrum ranges from 120 to 200 nm and a prominent feature in this range is the Lyman-alpha ($Ly - \alpha$) line at 121.6 nm. As we will see in the next section, only some refractive indices were available from 100 nm onwards, with most starting at 200 nm. It is therefore important to explore the influence of the FUV on the photoionization rate in order to see the importance of having refractive indices in the 100 – 200 nm region.

To investigate this effect, I compared the photoionization rates of hematite and magnetite using both the original refractive indices and once obtained after excluding the 100 – 200 nm region. Two scenarios were considered: one with a constant work function and varying radii, and the other with a constant radius and varying work function. The percentage difference between the photoionization rates was then calculated.

Effect of the work function

The impact of the work function on photoionization was investigated by calculating the photoionization rate for a range of work functions from 1.5 to 5.5 eV across different radii.

Effect of the quantum yield

To study the effect of varying the quantum yield, the photoionization was calculated for different values of Y ranging from 1 to 0.01.

Effect of the solar photon flux

To investigate the impact of varying solar photon fluxes, the solar spectrum irradiance was multiplied by a list of factors ranging form 0.1 to 2 before calculating the photon flux. The resulting photoionization rates were then plotted.

5.1.2 Optical constants

A major limitation in exploring the variability of the photoionization of MSP is the limited experimental refractive index measurements available for possible MSP materials and their limited spectral range. Some possible MSP materials were presented in section 2.4, and due to the availability of data, this study examines the photoionization rate of wüstite, hematite, magnesiowüstite, pyroxene, and olivine. A summary of the refractive indices of these materials can be found in table 5.1. All the refractive indices were sourced from the Database of Optical Constants for Cosmic Dust by the Laboratory Astrophysics Group of AIU Jena (`https://www.astro.uni-jena.de/Laboratory/OCDB/`).

Table 5.1: Summary of optical constants of possible MSP materials.

Material (source)	**Chemical composition**	**Wavelength range** (μm)	**Source**
Wüstite	FeO	0.2 – 500	Henning et al. (1995)
Hematite	Fe_2O_3	0.1 – 1000	Amaury H.M.J. Triaud, unpublished.
Magnetite	Fe_3O_4	0.1 – 1000	Amaury H.M.J. Triaud, unpublished.
Magnesiowüstite	$Mg_{0.6}Fe_{0.4}O$	0.2 – 500	Henning et al. (1995)
Pyroxene	$Mg_{0.5}Fe_{0.5}SiO_3$	0.2 – 10^4	Jaeger et al. (1994); Dorschner et al. (1995)
Olivine	$Mg_{0.8}Fe_{1.4}SiO_4$	0.2 – 10^4	Jaeger et al. (1994); Dorschner et al. (1995)

From table 5.1, we observe that only two of the six materials have refractive indices starting from 0.1 μm (= 100 nm), while the rest begin at 0.2 μm. This is significant because we want to integrate from as close to 0 as possible up to the cut-off wavelength, which is determined by the electron affinity or work function. The Sun emits a significant amount of energy in the FUV/UV range, which could be of mayor contribution to the photoionization of MSP.

Not having refractive indices for the 100 – 200 nm range also means that the photoionization of these materials is limited to a lower range of work functions. A high work function results in a low cut-off wavelength, which can easily go below 200 nm. Some refractive indices, like those of wüstite and magnesiowüstite, have relatively low spectral resolutions of 50 nm. The program calculating the photoionization can only integrate between data points, so without interpolating the data, the integration is often cut short. To avoid this, I have interpolated the refractive index data down to a spectral resolution of 0.001 μm = 1 nm. From this list, we can also classify the materials into two main groups: metal oxides (wüstite, hematite, magnetite, and magnesiowüstite) and silicates (pyroxene and olivine).

The refractive indices of the six materials are plotted in fig. 5.3. As we are

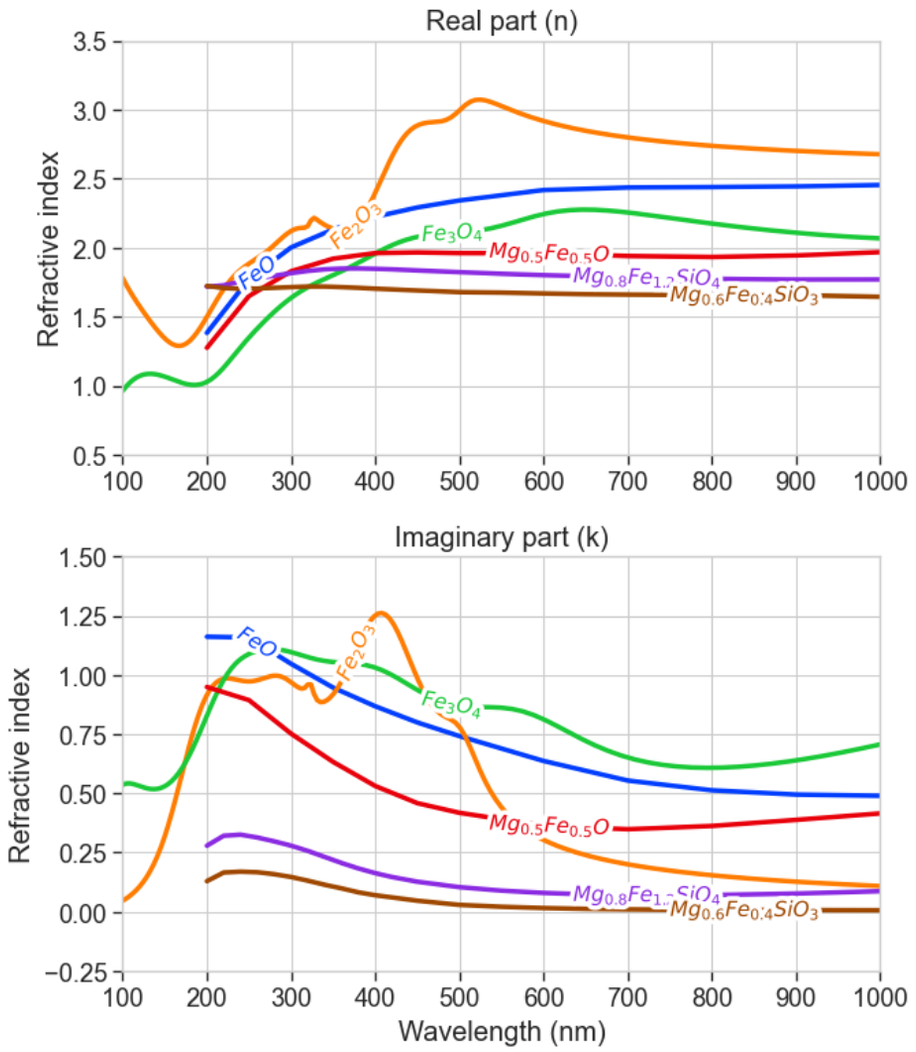

Figure 5.3: The real and imaginary parts of the refractive index of the materials listed in table 5.1. The top figure shows the real part while the bottom figure shows the imaginary part.

interested in the absorption cross section of MSP made of these materials, we will focus on the imaginary part of the refractive index. Firstly, we can see that the metal oxides have generally a higher complex refractive index in the UV and visual part of the spectrum than the silicates. This indicates that they are are generally more absorbing than the silicates, especially at shorter wavelengths. This implies that silicates are less absorbing than non-silicates. We can also observe in the 100 – 200 nm range that the imaginary part for both hematite and magnetite rapidly decreases. This could be important for

the absorption of Lyman-α and the FUV part of the solar spectrum.

I reviewed the available literature to identify experimental data on the work function and electron affinity of the materials listed in table 5.1. The work functions are presented in table 5.2, while the electron affinities are presented in table 5.3. Unfortunately, I was unable to obtain work functions for wüstite and magnesiowüstite and electron affinities for magnesiowüstite, pyroxene, and olivine.

The work function for MSP material, listed in the first row of table 5.2, was estimated from rocket flight measurements, where attempts were made to measure the work function of MSP in situ. Although this study could not conclusively identify the MSP composition, the results suggested the presence of $Mg - Fe$ hydroxide clusters (Rapp et al., 2012b). The work function for hematite given by Rapp (2009) was originally presented and used in calculations and has been commonly referenced in similar studies (see Knappmiller et al. (2011); Baumann et al. (2013)), although no specific source was cited in the original work for this value. Fortunately, I found two more recent studies indicating a similar work function for hematite. Notably, among all the work functions listed, olivine stands out as having the largest value, exceeding that of the others by more than 2 eV.

Table 5.2: Work function of possible MSP materials.

Material	**Work function** (eV)	**Source**
MSP material	4.0 – 4.6	Rapp et al. (2012b)
Hematite	5.5	Rapp (2009)
Hematite $\alpha - Fe_2O_3$	5.4 ± 0.2	Kraushofer et al. (2018)
Hematite $\alpha - Fe_2O_3$	5.49	He (2017)
Magnetite	5.20 ± 0.15	Fonin et al. (2005)
Pyroxene	5.14 ± 0.36	Li et al. (2016)
Olivine	7.90 ± 0.35	Li et al. (2016)

The electron affinities presented in table 5.3 are generally lower compared to the corresponding work functions.I was unfortunately only able to find data for three out of the six materials. While we can safely assume that the electron affinity for the remaining three materials will be lower than their work functions, estimating the photodetachment rate can still be challenging. The values reported in the study by Rienstra-Kiracofe et al. (2002) are derived from experimental results using photoelectric techniques for atoms and molecules. Equation (4.7), on the other hand, assumes bulk properties of the material. As

the bulk and molecular properties of a material might differ, this introduces a possible source of uncertainty.

Table 5.3: Electron affinity of possible MSP materials.

Material	**Electron affinity** (eV)	**Source**
Wüstite	1.5	Wang et al. (1996). Interpreted from fig. 2.
Wüstite	1.4945 ± 0.0001	Rienstra-Kiracofe et al. (2002, tab. 10)
Hematite	2.5 ± 0.2	Wang et al. (1996). Interpreted from fig. 2.
Hematite	3.06 ± 0.04	Rienstra-Kiracofe et al. (2002, tab. 10)
Magnetite	2.4	Wang et al. (1996). Interpreted from fig. 2.
Magnetite	2.56 ± 0.06	Rienstra-Kiracofe et al. (2002, tab. 10)

In the photoionization rate calculations, the work function and electron affinity are treated as the same variable, as the only distinction lies in their typical values. Consequently, in the discussion later on, the work function and electron affinity may be referred to as only the work function. The distinction between a value being a work function or an electron affinity is not clearly defined. Thus, when plotting the photoionization for a work function range of $1.5 - 5.5$ eV, the lower range of work functions is more likely to cause photodetachment rather then photoionization.

To assess how different work function values impact the cut-off wavelength for integration in eq. (4.7), eq. (4.8) is plotted in fig. 5.4 with marks denoting the cut-off wavelength for specific work functions. For instance, a work function of 4 eV corresponds to a cut-off wavelength of 310 nm, indicating that eq. (4.7) would integrate over the wavelength interval $0 - 310$ nm (or from the closest available wavelength to 0).

Notably, the cut-off wavelength for olivine's work function (7.9 eV) is 157 nm, which is smaller than the 200 nm starting point of the available refractive index for olivine. Consequently, calculating the photoionization of olivine using this work function is not possible with our current data. Additionally, for materials with refractive indices starting at 200 nm, the upper limit for the work function is restricted to 6 eV or $\approx$ 200 nm. Using a work function of 6 eV

creates limited data points for integration. To ensure a sufficient amount of data for integration, the highest work function utilized will be 5.5 eV, resulting in a cut-off wavelength of 225 nm.

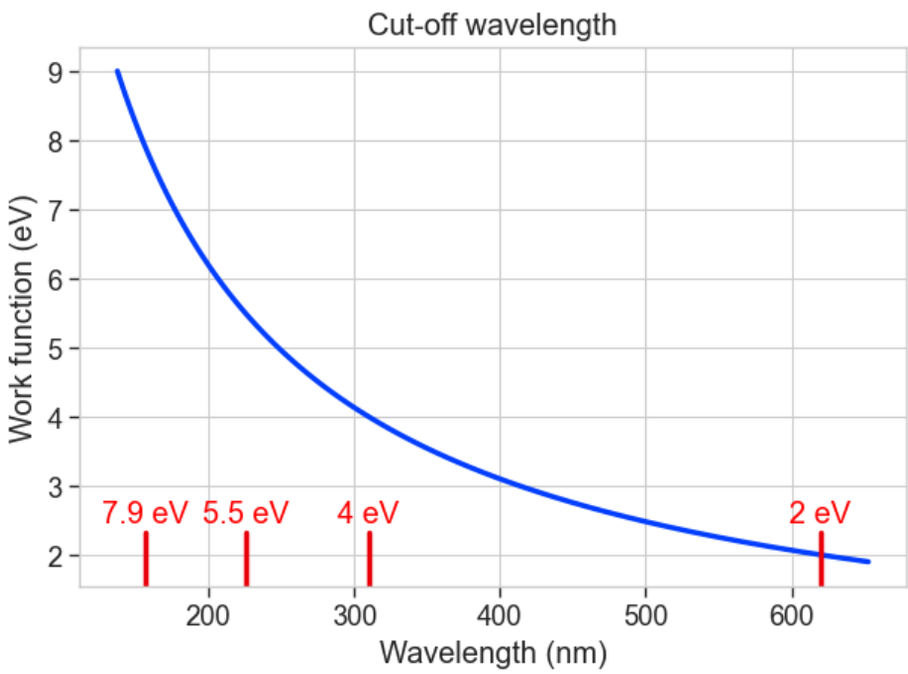

Figure 5.4: The cut-off wavelength for integration in eq. (4.7) as a function of work function in electron volts. The cut-off wavelengths corresponding to work functions of 7.9, 5.5, 4, and 2 eV are marked in red.

5.1.3 Solar spectrum

In order to calculate the solar photon flux, I needed to use a solar irradiance spectrum. I had originally planned to use the WHI SIRS spectrum shown in fig. 3.1; however, its high spectral resolution resulted in long compiling times for the program. Therefore, I decided to use the LASP GSFC Composite Solar Spectral Irradiance (SSI3) spectrum. This data set is based on several satellite measurements of the solar spectrum tracing back to the 1970s. It has a spectral range from 0.5 nm to 1597.5 nm, with a spectral resolution of 1 nm in the range 0.5 – 752.5 nm (Woods and DeLand, 2021). I used the data from 30.06.2019, and the spectrum is plotted in fig. 5.5. This data was accessed via the LASP Interactive Solar Irradiance Datacenter (LISIRD) (`https://lasp.colorado.edu/lisird/`) on 10.04.2024.

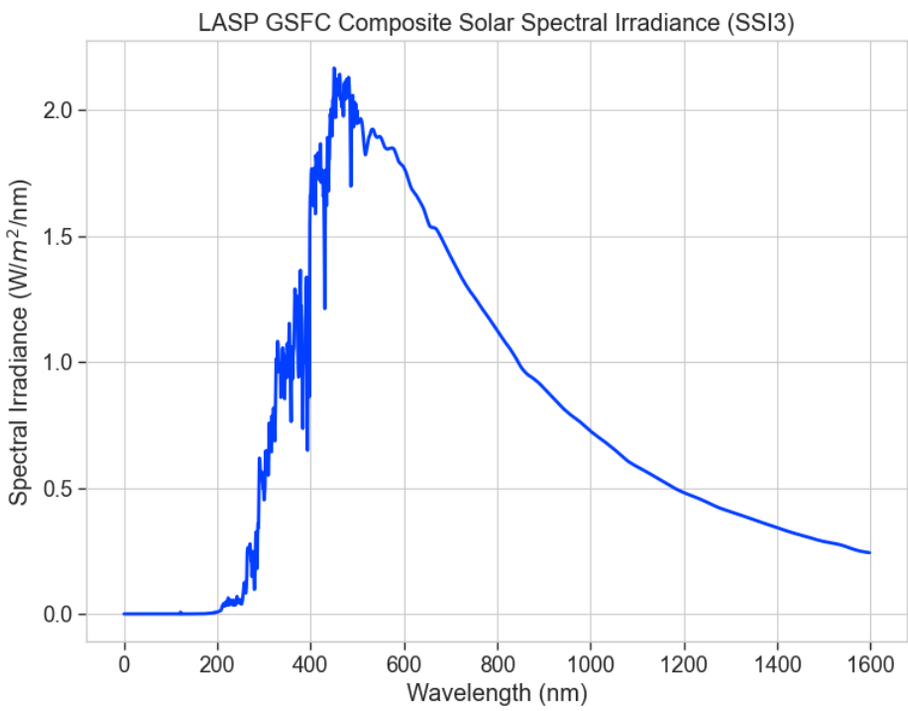

Figure 5.5: The SSI3 reference irradiance spectrum.

The solar photon flux was calculated from the solar irradiance using the following equation:

$$F(\lambda) = \frac{I(\lambda)\lambda}{hc} \tag{5.2}$$

where $I(\lambda)$ is the solar irradiance at wavelength λ, h is Planck's constant, and c is the speed of light.

5.1.4 Miepython

To calculate the photoionization from eq. (4.7), the absorption cross section must first be determined. As seen in section 3.3, calculating the absorption cross section of a sphere using Mie theory is not straightforward. It has become common practice to use computer programs for Mie solutions. Various programs are available for different programming languages such as Fortran, MATLAB, and C++, among others. As Python is my primary programming

language, I sought a Mie scattering Python library that would fit my needs. For my calculations, I used the Python package *miepython* made by Scott Prahl (2023). This package follows the computation procedure described by Wiscombe (1979) in his book "Mie scattering calculations: Advances in technique and fast, vector-speed computer codes" and can calculate light scattering by non-absorbing, partially-absorbing, and perfectly conducting spheres.

There are a few reasons I chose this package over others. Firstly, *miepython* employs a logarithmic derivative approach rather than a special built-in function in the commonly used Python library *SciPy* for calculating the derivative of the Riccati-Bessel function, as seen in eqs. (3.26) and (3.27), ensuring greater accuracy. A detailed description of this procedure can be found at `https://miepython.readthedocs.io/en/latest/07_algorithm.html`. Secondly, the program has implemented a special case for small spheres which is faster and more accurate than other programs. This is significant for this work as the MSP we are looking at are often two to three orders of magnitude smaller than the incoming wavelength.

5.2 Results of variability of photoionization

5.2.1 Refractive index

Figure 5.6 shows the photoionization rate for MSP with radii ranging from 0.5 to 10 nm, using a work function of 4.5 eV. By using the same work function for all the different materials, we can observe how the photoionization varies solely with the refractive indices. Notably, the photoionization rate exhibits a logarithmic increase with radii across all materials. This trend aligns with expectations, as the absorption cross section expands with the geometric cross section, which scales with the MSP radius squared.

Furthermore, metal oxides generally exhibit higher photoionization rates compared to silicates across all radii. This difference is slightly less than one order of magnitude and is consistent with the higher absorptive nature of metal oxides compared to silicates, as indicated by their refractive indices. Additionally, there is minimal disparity between the metal oxides, whereas a more pronounced distinction exists among the silicates, with olivine demonstrating a higher photoionization rate than pyroxene.

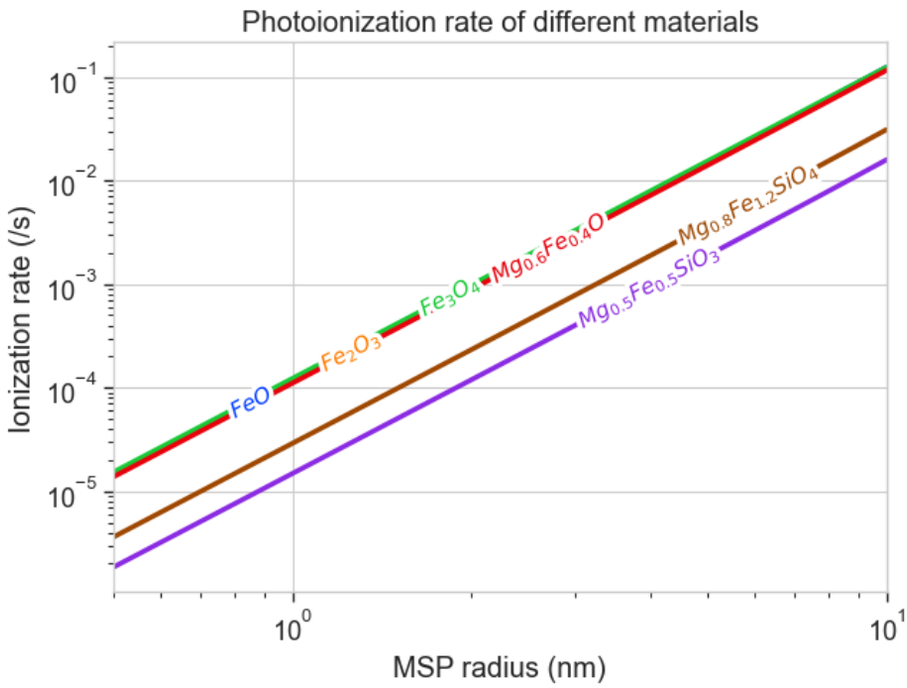

Figure 5.6: Photoionization rate for particles made of different materials with radii ranging from 0.5 to 10 nm and a work function of 4.5 eV. FeO, Fe_2O_3, Fe_3O_4, and $Mg_{0.6}Fe_{0.4}O$ are partially or fully overlapping. Both of the axes are plotted with a logarithmic scale.

5.2.2 Effect of the extreme/far UV

Figure 5.7 (a) displays both the original photoionization rate and the rate obtained after excluding the $100 - 200$ nm interval from the refractive index. These rates are calculated for particles with a work function of 4.5 eV and radii ranging from 0.5 to 10 nm, for both hematite (Fe_2O_3) and magnetite (Fe_3O_4). The corresponding percentage difference is illustrated in fig. 5.7 (b). The discrepancy between the original and removed photoionization rates is relatively constant, around 5% for hematite and 9.5% for magnetite. This percentage difference tends to decrease marginally at larger radii. Similar trends are observed when examining the difference using other work functions, with the percentage remaining relatively stable across all radii.

Figure 5.8 (a) illustrates the photoionization rate as it varies with the work function, ranging from 1.5 to 5.5 eV, for a particle with a radius of $r = 1\ nm$. The corresponding percentage difference is plotted in fig. 5.8 (b). Across

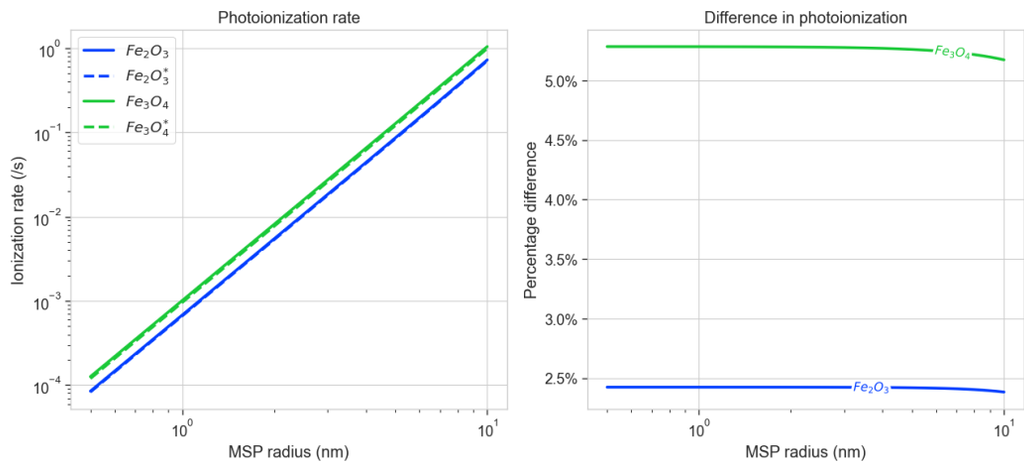

Figure 5.7: a) Calculated photoionization for a particle with a work function of 5.5 eV. The dashed lines denote photoionization calculated where the 100 – 200 nm range was excluded. **b)** Percentage difference of photoionization.

both materials, we observe an increase in the difference as the work function increases, with the removed photoionization rate being lower than the original rate. For hematite, the difference remains close to 0% up to approximately 3 eV, after which it gradually increases to reach 5% at 4.6 eV, peaking at 19% at 5.5 eV. In contrast, magnetite has an earlier onset of increase, with the difference reaching 5% at 3.5 eV and peaking at almost 30% at 5.5 eV. The observed increase in difference with higher work functions is unsurprising, given that the integration interval is significantly reduced for the removed refractive index compared to the original.

In summary, excluding the 100 – 200 nm interval from the refractive index resulted in a slightly difference in photoionization rate across different particle radii. Additionally, as the work function increased, we observed a growing difference in photoionization rates. This suggests that for photoionization rates calculated from materials with refractive indices starting at 200 nm and using high work functions, the actual rates are likely to be higher. This seems to mostly be a concern for photoionization and not for photodetachment, as the extent of this increase depends on the material's work function and range of wavelengths the photoionization is being integrated over. Despite this, it remains reasonable to utilize the current refractive indices and regard the results as lower bounds until refractive indices within these ranges are available.

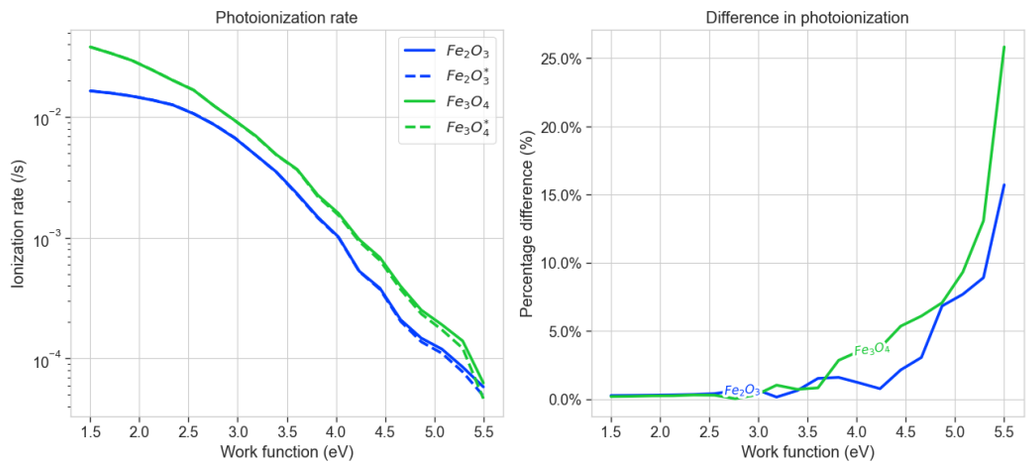

Figure 5.8: a) Calculated photoionization for a particle with a radius of $r = 1$ nm and varying work function. The dashed lines denote photoionization calculated where the $100 - 200$ nm range was excluded. **b)** Percentage difference of photoionization.

5.2.3 Work function

The photoionization rates for MSP of the six materials were calculated using work functions ranging from 1.5 to 5.5 eV and are depicted in fig. 5.9. Across all materials, the photoionization rate logarithmically increases with increasing particle radii. Comparing the magnitude of the photoionization rate for different work functions, we can observe that for larger work functions, the difference appears to be approximately one order of magnitude for all materials. For instance, 4.5 eV is roughly one order of magnitude higher than 5.5 eV, and 3.5 eV surpasses 4.5 eV by a similar magnitude. This increase diminishes as the work function decreases, suggesting that variations in electron affinity may not have as significant of an impact on photodetachment compared to the effect of variation in work function on photoionization.

5.2.4 Quantum yield

The photoionization rates calculated with various quantum yields Y are plotted in fig. 5.10. We observe that the photoionization rate decreases with decreasing quantum yield. As Y acts as a scaling factor, we see about one order of magnitude difference between 1 and 0.1, and the same for 0.1 and 0.01. All the materials respond similarly to the change in quantum yield.

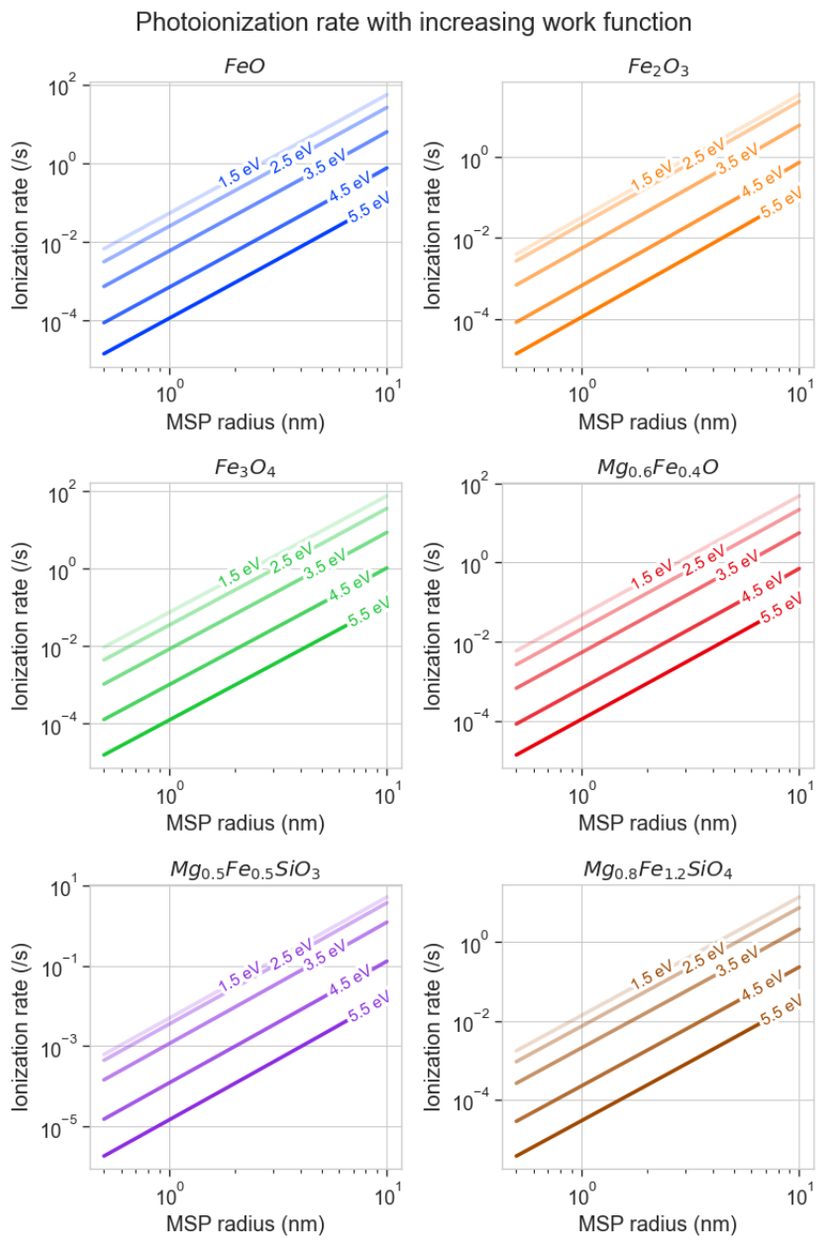

Figure 5.9: The photoionization rate of different materials plotted for different work functions as a function of particle radius. The line label denotes the value of the work function in eV.

The quantum yield is expected to be rather high. As Rapp and Strelnikova (2009) notes, experimental findings indicate that nanoparticles might have very high yields, even up to three orders of magnitude larger than the yield for

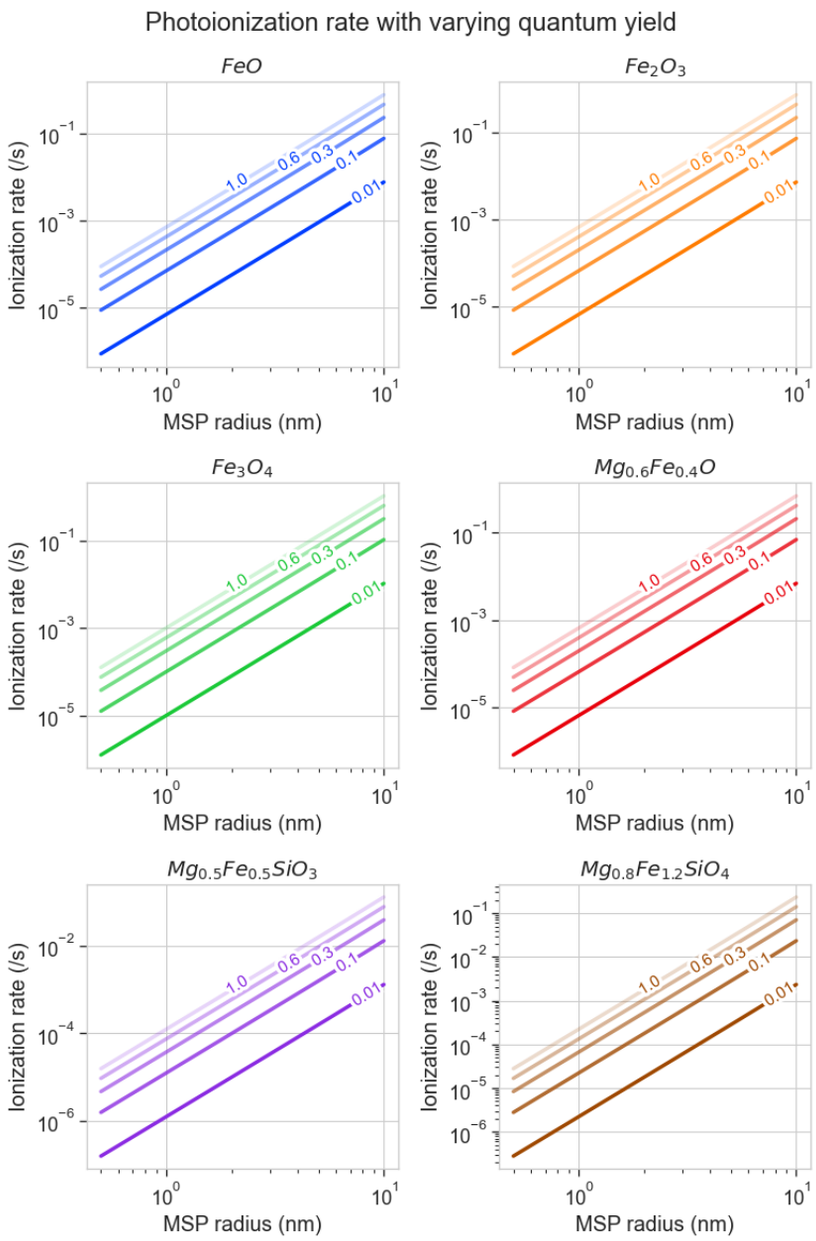

Figure 5.10: The photoionization rate of different materials plotted for different quantum yields as a function of particle radius. The line label denotes the value of the quantum yield.

corresponding bulk materials. When impacted by a photon, a nano-particle doesn't have as many degrees of freedom as bulk material to dissipate the energy, so it is likely to rid itself of the energy by ejecting an electron.

5.2.5 Solar photon flux

The photoionization rates calculated using solar irradiation multiplied by a range of factors are plotted in fig. 5.11, with the line labels indicating the multiplied factor. Like with both the work function and quantum yield, the different materials respond similarly to the different solar photon fluxes. Notably, to affect the photoionization by one order of magnitude, a factor of 0.1 is needed. The photoionization with a 10% increase in solar irradiation (factor 1.1) almost perfectly overlaps with the unscaled photoionization (factor 0) for all materials. The solar irradiation needs to increase or decrease by about 50% (factor 0.5 and 1.5) in order to have a notable effect on the photoionization. The solar constant varies by about 0.1% over a solar cycle (Willson and Hudson, 1991), however solar variability can be higher in the UV region. It is still unlikely for the solar irradiation to vary by 50% or more. It is therefor unlikely that a realistic change in solar irradiance will have a major effect on the photoionization of MSP.

5.2.6 Summary

In summary, the variability of photoionization rates for MSP is influenced by several key parameters. From the results of the computations, we see that variations in the refractive index and work functions are the most important factors. We saw a big difference in the behavior of metal oxides and silicates, with the silicates being less absorbing and having lower photoionization rates in general.

Removing the FUV range from the refractive index resulted in only slight differences in photoionization rates across different particle radii. However, when plotting against a range of work functions, as the work function increased, the photoionization rates increased significantly. This suggest that for photoionization calculations, measurements of the refractive index in the $100 - 200$ nm range are needed. to get accurate estimates of the photoionization

The quantum yield and solar photon flux also affect the photoionization rates, but their impact is more predictable and less variable across different materials.

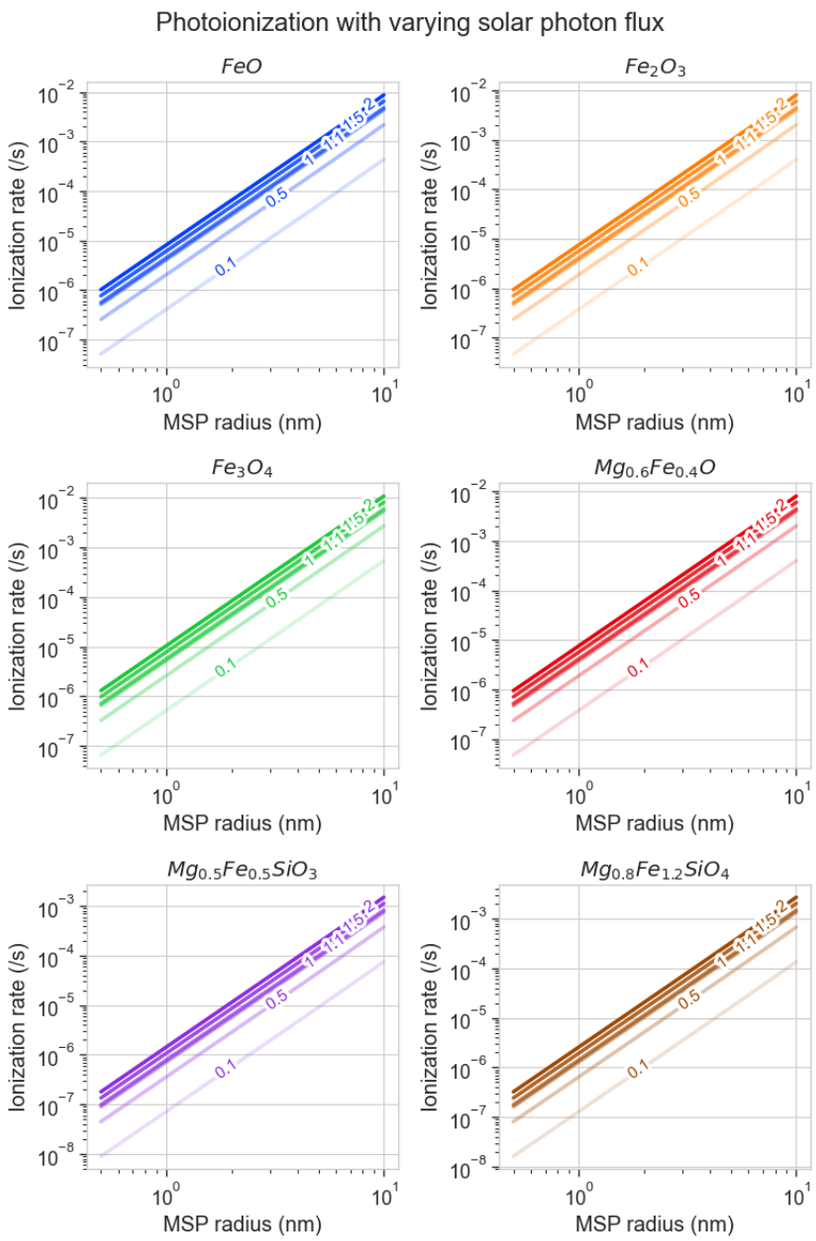

Figure 5.11: The photoionization rate of different materials plotted for different solar photon flux strengths as a function of particle radius. The line label denotes the factor by which the solar photon flux is changed.

6

Modeling the charge state of MSP

In this chapter, I describe and present the results of a model designed to calculate the charging rates, number densities and charge probability of MSP within the mesosphere. Section 6.1 outlines the construction of the model and details the dependent variables of the functions. Following this, in section 6.2, I present and discuss the outcomes of the model.

6.1 Model Description

The main purpose of this program is to investigate how the number density of electrons, ions, and MSP changes with the inclusion of photoionization and photodetachment, assuming different MSP materials. Previous studies have primarily focused on hematite and considered MSP radii down to a minimum size of $r = 1$ nm. By varying the material properties of the MSP and extending the size range down to 0.5 nm, this study aims to examine whether the charge probability of MSP changes and to assess the effectiveness of photodetachment in neutralizing negatively charged MSP.

To calculate the number density of MSP with a charge Z, denoted as N_Z, we solve the kinetic rate equation (Draine and Sutin, 1987; Jensen and Thomas, 1991; Rapp and Lübken, 2001; Knappmiller et al., 2011):

$$\frac{dN_Z}{dt} = n_e \alpha^e_{Z+1} N_{Z+1} - (\nu^+_Z + n_e \alpha^e_Z) N_Z + \nu^+_{Z-1} N_{Z-1} \tag{6.1}$$

Here, n_e is the electron density, and α^e_Z is the electron attachment rate. ν^+_Z is the positive charging rate, which includes positive ion attachment, photodetachment, and photoemission, expressed as:

$$\nu^+_Z = n_i \alpha^+_Z + \nu^{det}_Z + \nu^{emis}_Z \tag{6.2}$$

Where n_i is the ion density, α^+_Z is the ion attachment rate, and ν^{det}_Z and ν^{emis}_Z are the rates of charging from photodetachment and photoionization, respectively.

Maintaining quasi-neutrality within the system necessitates the inclusion of two additional equations accounting for changes in electron and ion densities:

$$\frac{dn_e}{dt} = Q - \alpha_{ie} n_i n_e - \sum_Z (n_e \alpha^e_Z - \nu^{det}_Z - \nu^{emis}_Z) N_Z \tag{6.3}$$

$$\frac{dn_i}{dt} = Q - \alpha_{ie} n_i n_e - \sum_Z n_i \alpha^i_Z N_Z \tag{6.4}$$

Here, Q is the rate of ion-electron pair production from ionization, and α_{ie} is the ion-electron recombination rate. The coefficients α^e and α^i, representing electron and ion attachment respectively, are calculated from eqs. (4.3) to (4.5) in section 4.1. As we are focusing on nanoparticles, we assume that each MSP can only carry one additional charge, leading to an examination of three distinct MSP populations: neutral MSP ($Z = 0$), positively charged MSP ($Z = 1$), and negatively charged MSP ($Z = -1$). Using eq. (6.1), we obtain the following rate equations describing the change in number density of the three MSP populations:

$$\frac{dN_{Z=1}}{dt} = n_i \alpha_0^i N_0 - [\alpha_1^e n_e + \alpha_1^i n_i] N_Z + \beta_{ion} N_0 \tag{6.5}$$

$$\frac{dN_{Z=0}}{dt} = \alpha_{-1}^i N_{-1} n_i - [\alpha_0^e n_e + \alpha_0^i n_i] N_0 + \alpha_1^e N_1 n_e + \beta_{det} N_{-1} - \beta_{ion} N_0 \tag{6.6}$$

$$\frac{dN_{Z=-1}}{dt} = \alpha_0^e N_0 n_e - [\alpha_{-1}^e n_e + \alpha_{-1}^i n_i] N_{-1} - \beta_{det} N_{-1} \tag{6.7}$$

Equations (6.3) to (6.7) form the set of differential equations to be solved to estimate the number density of the electrons, positive ions, positive MSP, neutral MSP, and negative MSP. A comprehensive summary of all coefficients in the set of differential equations can be found in table 6.1.

Table 6.1: Summary of all coefficients used in differential equations.

Coefficient	Unit	Comment
α_{ie}	$cm^3 s^{-1}$	ion-electron dissociative recomb.
Q	$cm^{-3} s^{-1}$	ionization (ion-electron pairs)
α_{-1}^e	$cm^3 s^{-1}$	electron attachment to MSP-
α_{-1}^i	$cm^3 s^{-1}$	pos. ion attachment to MSP-
α_0^e	$cm^3 s^{-1}$	electron attachment to MSP
α_0^i	$cm^3 s^{-1}$	pos. ion attachment to MSP
α_1^e	$cm^3 s^{-1}$	electron attachment to MSP+
α_1^i	$cm^3 s^{-1}$	pos- ion attachment to MSP+
β_{det}	s^{-1}	MSP- electron photodetachment
β_{ion}	s^{-1}	MSP electron photoionization

The Python program used for all calculations is accessible in appendix .3. The differential equations were solved using the *odeint()* function from the SciPy package. The model's input variables include the ionization rate Q, integration time t, MSP radius r, total MSP number density N_{tot}, and, if applicable, parameters for photoionization like the refractive index of the material, its work function, and electron affinity.

The model assumes a constant ion-electron dissociative recombination rate of $\alpha_{ie} = 10^6$ m^{3}s^{-1} to align with previous studies (Knappmiller et al., 2011; Jensen and Thomas, 1991). Additionally, it assumes equal initial number densities for electrons and ions, with all MSP starting as neutral, i.e., $N_e = N_i$ and $N_0 = N_{tot}$, $N_1 = N_{-1} = 0$. The initial number densities of electrons and ions are determined by $N_e = N_i = \sqrt{Q/\alpha_{ie}}$ (Jensen and Thomas, 1991).

Unless stated otherwise, the program was executed with initial values of $Q = 10\ \mathrm{cm^{-3}s^{-1}}$, yielding an initial electron and ion number density of 3162 $\mathrm{cm^{-3}}$. This was done to be consistent with the previous works of Jensen and Thomas (1991) and Knappmiller et al. (2011), where these values are consistent for an altitude of 87 km during the daytime with a mixture of water cluster ions and molecular ions. For the ion mass, the program uses a mass of 50 AMU which corresponds to an average ion species dominated by water cluster ions (Knappmiller et al., 2011). An electron and ion temperature of 200 K is also assumed. Other initial values were the MSP radius $r = 0.8$ nm and total MSP population $N_{tot} = 10^4\ \mathrm{cm^{-3}}$.

As we will see later in the chapter, I decided to run calculations for three different cases. Two of these cases include the photoionization and photodetachment of hematite and olivine, respectively, because previous chapters showed that metal oxides and silicates behaved similarly. Hematite and olivine were chosen as representative materials due to their likelihood as MSP candidates. However, there is a significant challenge: the data for olivine is limited. Specifically, we have no refractive index in the $100 - 200$ nm range, no estimate for electron affinity, and the work function is too high for calculating photoionization. Therefore, I used the same work function and electron affinity for both materials, adopting the values for hematite: $WF = 5.5$ eV and $EA = 2.5$ eV.

It is also worth considering the minimum particle size to be included in this analysis. Previous studies have set a lower limit of $r = 1$ nm because eqs. (4.3) to (4.5) were designed for microscopic-sized particles and may become invalid below a certain size. I have tried to take this into account by introducing an efficiency factor, discussed in section 6.1.2. Another consideration is that eq. (4.7) assumes the absorption properties of particles can be described using a refractive index measured for bulk material (Rapp, 2009). Therefore, it is uncertain how particles too small to be characterized by bulk material properties would behave. However, a small particle impacted by a photon would have fewer ways of dissipating the photon energy, making the ejection of an electron likely. I have therefore found it worthwhile to expand the MSP size range down to 0.5 nm in order to investigate if an MSP so small could be charged.

6.1.1 Quasi-neutrality

As outlined in Knappmiller et al. (2011), quasi-neutrality can be approached through two scenarios: a low-density case and a high-density case. In the low-density case, we assume constant electron and ion densities, with the MSP population deemed small enough to negligibly affect these densities. Consequently, only eqs. (6.5) to (6.7) form the set of differential equations. In the high-density scenario, the MSP population is comparable to the electron and ion number densities, resulting in a notable depletion of these densities. Here, eqs. (6.3) and (6.4) are included in the set of differential equations.

Most of the calculations have been done following the high-density case. However, as Knappmiller et al. (2011) explored both scenarios, I wanted to investigate whether these different approaches would yield different probabilities for the MSP charge state. The high and low-density approaches are therefore compared in section 6.2.3.

6.1.2 Charging efficiency

Megner and Gumbel (2009) postulates that there must be a lower size limit to where MSP can collect charged particles. Theoretically, this is to be expected as the charging process can be viewed to consist of two steps: Initially, an electron must be captured by the induced electrostatic field of the MSP. Secondly, the energy supplied by the collision must be accommodated so that a stable charged particle is produced. While electromagnetic theory describes the first step through the charge coefficients proposed by Natanson (1960), the latter phase remains challenging for experiments and theory (Megner and Gumbel, 2009).

To address this size-dependent charging probability, a dimensionless parameter γ_{charging} has been incorporated into the program. This parameter is multiplied to the neutral MSP attachment coefficients and is given in eq. (6.8). The parameter was adopted from Baumann et al. (2013), who first used it in their model following the arguments from Megner and Gumbel (2009).

$$\gamma_{\text{charging}}\left(r_{\text{p}}\right) = \begin{cases} 0, & \text{for } r_{\text{p}} < 0.25 \text{ nm} \\ 0.8 \cdot r_{\text{p}} - 0.2, & \text{for } 0.25 \leq r_{\text{p}} \leq 1.5 \text{ nm} \\ 1, & \text{for } r_{\text{p}} > 1.5 \text{ nm} \end{cases} \tag{6.8}$$

6.2 Results

This section presents and discusses the results from the model. To effectively convey the data, three cases are presented; case 1: photoionization of hematite, case 2: photoionization of olivine, and case 3: the absence of photoionization and photodetachment. This approach is based on the observed similarity in behavior among metal oxides and silicates regarding photoionization, as discussed in the previous section. Consequently, hematite and olivine are chosen to represent these two material categories. Plots of the data for all six materials are provided in appendix A.

6.2.1 Charging rates

Figure 6.1 shows the different charging rates listed in table 6.1 for the three cases with MSP radii ranging from 0.5 to 3 nm. The charging rates shown are included in the set of differential equations and are plotted to illustrate the dominant charging mechanisms in each case.

In case 1 and 2, we can see that photodetachment is almost two orders of magnitude stronger than photoionization. Additionally, photodetachment and photoionization are stronger for hematite than for olivine, consistent with observations from section 5.2. We also see that electron attachment to neutral MSP ($\alpha_0^e N_0$) is slightly stronger than photodetachment for olivine for radii between $0.5 - 1$ nm. This results in a stronger ion attachment to negative MSP ($\alpha_{-1}^i N_{-1}$) for olivine than for hematite to maintain balance.

For all three cases, the charging processes involving negatively charged MSP are more rapid than other charging mechanisms. In case 3, electron attachment to neutral MSP and subsequent ion attachment to negative MSP are approximately one and a half orders of magnitude faster for smaller radii, and slightly less than one order of magnitude faster for larger radii. This is expected, as lighter electrons have higher mobility compared to heavier and slower ions, leading to faster electron impacts on MSP.

6.2.2 Number density

This section examines the expected number density of electrons, positive ions, positive MSP, neutral MSP, and negative MSP while varying the ionization rate Q, MSP radius r, and total MSP number density N_{tot}.

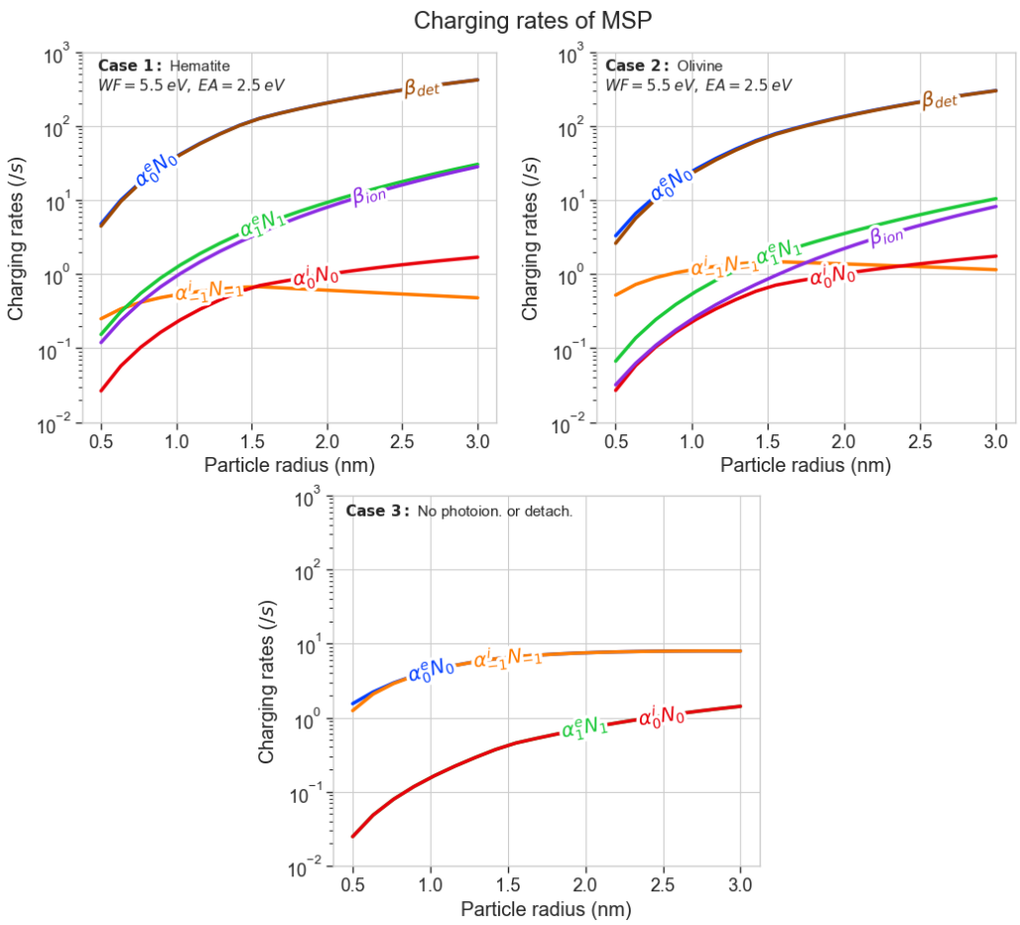

Figure 6.1: Charging rates of MSP with **a)** photoionization of hematite, **b)** photoionization of olivine, and **c)** no photoionization and detachment for a range of MSP radii.

Variable ionization rate

Figure 6.2 shows the number density of different plasma species varying with the ionization rate Q from 10^{-2} to 10^{4} $\mathrm{cm}^{-3}\mathrm{s}^{-1}$. The initial values for electron and ion number densities are also adjusted according to $N_e = N_i = \sqrt{Q/\alpha_{ie}}$.

As the initial values of electron and ion number densities directly depend on the ionization rate Q, there is a strong correlation between increasing Q and increasing electron and ion number densities in all three cases. At lower ionization rates, fewer electrons and ions are available, resulting in the neutral

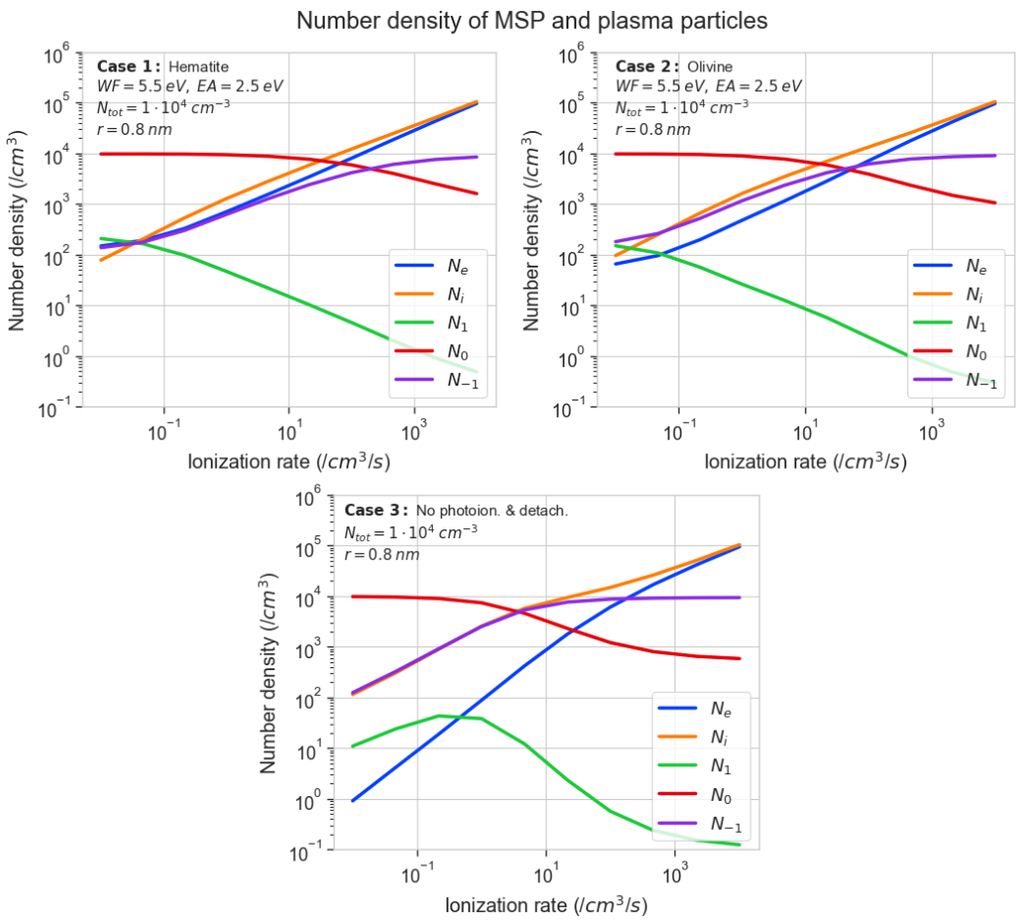

Figure 6.2: Number density of electrons, ions, positive MSP, neutral MSP, and negative MSP with varying ionization rate.

MSP population remaining largely unaffected. For $Q \leq 1\ \mathrm{cm}^{-3}\mathrm{s}^{-1}$, case 1 and 2 sees much higher electron and positive MSP population than case 3. In cases involving photoionization, lower ionization rates lead to similar number densities for positive and negative MSP, each around 100 cm^{-3}. As ionization rates increase, the number of positive MSP decreases while the number of negative MSP increases. This decrease in positive MSP with increasing Q is consistent across all three cases, likely due to the high mobility and availability of electrons effectively neutralizing positive MSP.

Variable radius

To examine how the number density varies with MSP radii, the number densities were calculated for MSP radii ranging from 0.5 to 10 nm. Figure 6.3 presents the three cases.

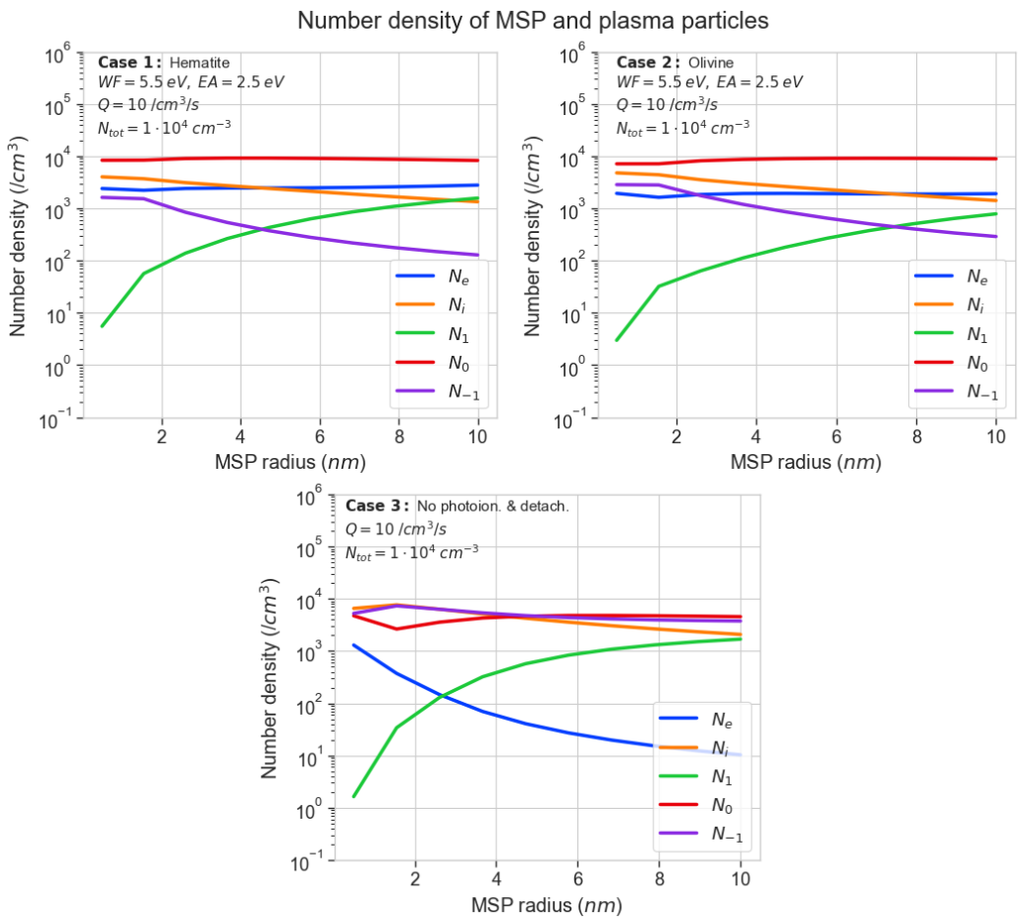

Figure 6.3: Number density of electrons, ions, positive MSP, neutral MSP, and negative MSP with varying MSP radii.

For all three cases, there is a general increase in the positive MSP population with size. In cases 1 and 2, the negative MSP population decreases by about one order of magnitude from 2 nm to 10 nm. This could be due to the dependence of photodetachment and photoionization on the geometric cross section of the MSP, which would make them more effective at larger radii.

We can observe in Figure 6.3 that for cases 1 and 2, the negatively charged

and positively charged MSP populations intersect at 4.5 nm and 7.5 nm, respectively. Specifically, smaller radii are associated with a higher population of negatively charged MSP, while larger radii are associated with a higher population of positively charged MSP. In case 3, the negatively charged MSP population consistently exceeds the positively charged population. However, the positively charged population rises to a comparable order of magnitude as the negatively charged population at approximately 6 nm. This indicates that when photoionization and photodetachment are included, smaller MSP are more likely to be negatively charged, whereas larger MSP are more likely to be positively charged. Case 3 also suggests that under conditions without solar irradiation, MSP is likely to be negatively charged.

Additionally, in cases 1 and 2, the neutral MSP population remains relatively stable but reaches its lowest point before 2 nm. This suggests that smaller MSP are more likely to be charged compared to larger MSP. This observation is consistent with the expectation that the photodetachment rate increases with increasing radii, effectively neutralizing negatively charged MSP.

Variable total MSP number density

By varying the total number density of MSP, we can investigate how a fluctuating MSP population would affect the charge state of MSP and whether there is an upper limit to the number of MSP that can be charged. Figure 6.4 shows the number densities from the three cases varying with the total number density, which ranges from 10^2 to 10^5 cm^{-3}.

For all three cases, the neutral MSP population increases approximately exponentially with the total MSP population, which is expected since the initial value of neutral MSP is directly dependent on the total MSP number density.

The negative MSP population shows a nearly exponential increase across all three cases until reaching a total MSP number density of approximately 10^4 cm^{-3}, after which it begins to plateau. In contrast, the positive population exhibits a linear increase until the same point, after which the rate increases. Additionally, there is a faster depletion of electrons beyond this point, while the ion population remains relatively stable.

The higher charging rate of MSP from electrons compared to ions results in the negative MSP population tapering off as the electrons are depleted. When

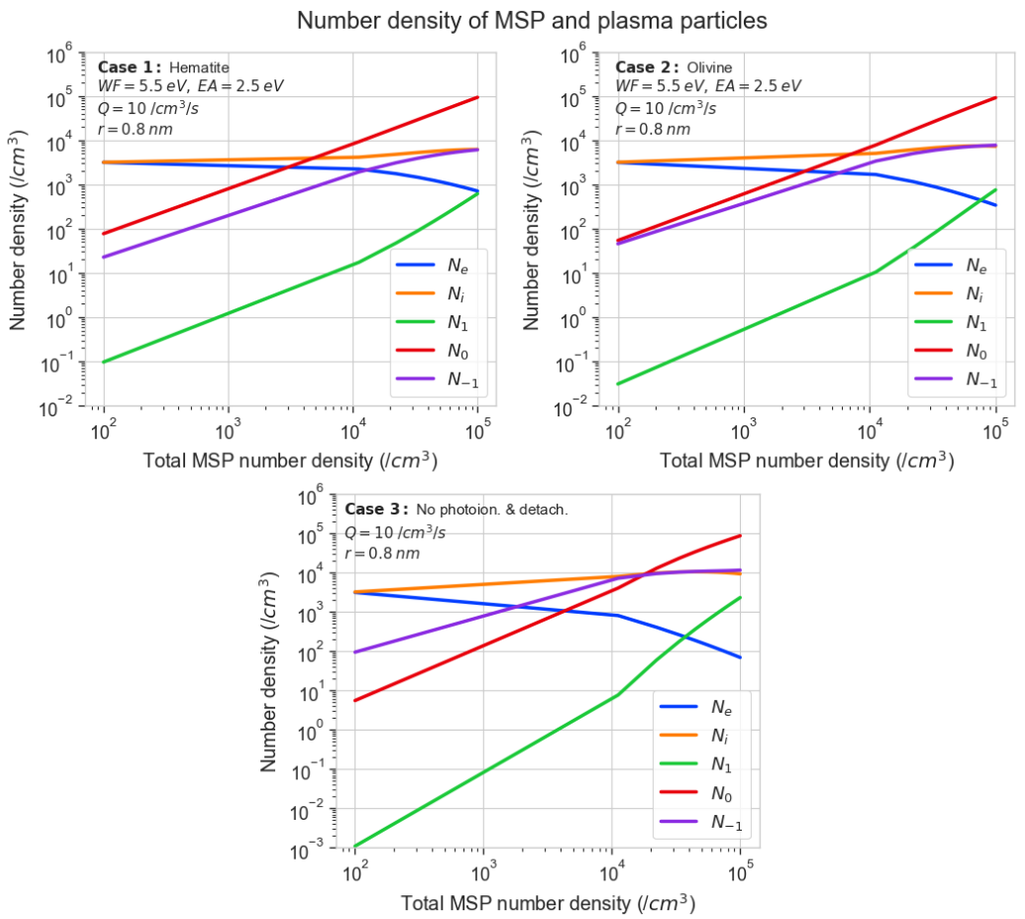

Figure 6.4: Number density of electrons, ions, positive MSP, neutral MSP, and negative MSP with varying total MSP number density.

the electron population is significantly smaller than the ion population, ion charging becomes more dominant, leading to an increase in the positively charged MSP population.

Comparing cases 1 and 2 with photoionization to case 3, we see that the negative MSP population is generally lower at the lower range of total MSP population when photodetachment is present. However, all three cases seem to converge towards the same point when $N_{tot} = 10^5$ cm^{-3}. Therefore, it seems like there is a limit to how many MSP particles can be charged, which depends on the electron and ion populations.

6.2.3 Charge probability

To more clearly illustrate the probability of a MSP being charged, I calculated the charge probability for MSP of different radii using both the low density and high density approaches for quasi-neutrality, inspired by Knappmiller et al. (2011) and described in section 6.1.1.

Low density case

When using the low density model, the electron and ion number densities are kept constant and the total MSP number density is $N_{tot} = 1$ cm^{-3}. The charge probability of MSP using the low density model is shown in fig. 6.5 for the three cases.

Firstly, we can observe a notable difference between the probabilities of the three cases, with case 1 and 2 having a significantly higher probability of being neutral for all radii compared to case 3.

For case 1, there is a slightly above 15% probability of a MSP having a $Z = -1$ charge for an MSP radius of 1 nm or smaller. All sizes have a probability of at least 80% of being neutral, and 10 nm has the highest probability of being positively charged, with approximately 15%.

For case 2, smaller MSP have a significantly higher likelihood of being negatively charged, with MSP smaller than 1 nm exhibiting a $35 - 40\%$ probability of being negatively charged. This is consistent with the fact that olivine, being a silicate, is less absorbing. Additionally, the probability of an MSP being positively charged is lower compared to case 1, down to less than 5% for a 10 nm MSP. This may be attributed to hematite's higher absorptivity, which leads to a stronger photoionization rate, resulting in a higher population of positively charged MSP.

For case 3, the model shows a high probability for a negative MSP for all radii, with 0.5 nm being the lowest at $\sim 80\%$. The probability of a MSP being positively charged is approximately zero.

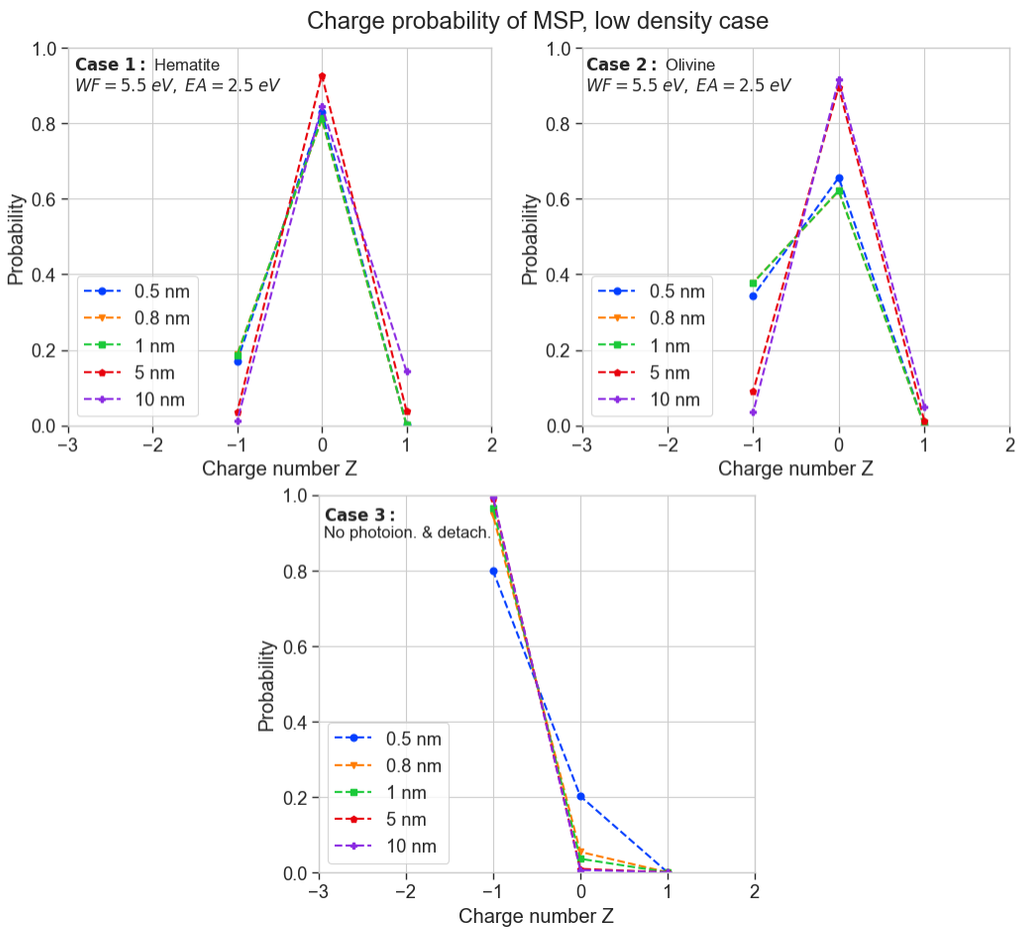

Figure 6.5: Probability of MSP of different radii to have a charge Z. Calculated using low density method. For the hematite and olivine case, 1 and 0.8 nm overlaps.

High density case

The charge probability of MSP using the high-density model is shown in fig. 6.6.

Both case 1 and 2 look quite similar to their low density model counterparts, however the most significant difference is in the charge probability for case 3. There is a major decrease in the probability of a negatively charged particle, with the lowest now being a 10 nm MSP with a slightly lower than 40% probability of being negatively charged. 0.8 nm and 1 nm have the highest probabilities of approximately 70%. The chance of a positive particle has

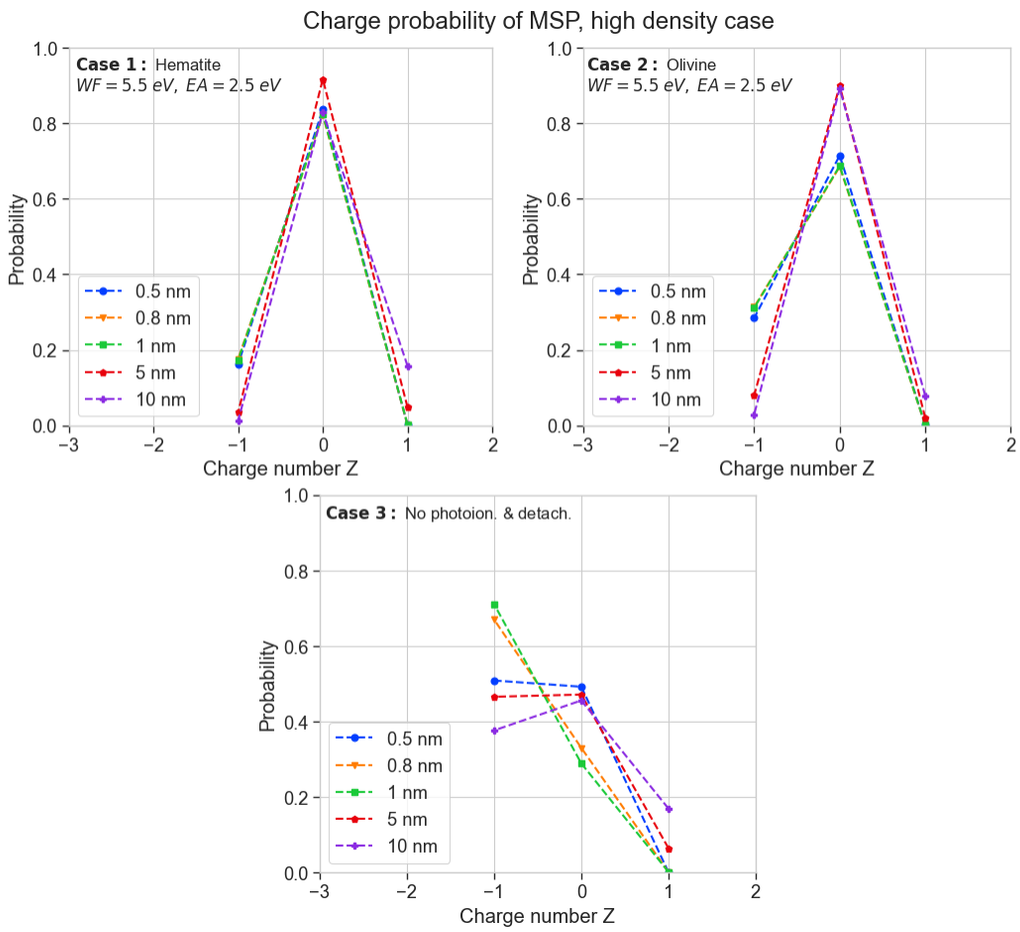

Figure 6.6: Probability of MSP of different radii to have a charge Z. Calculated using high density method. For case 1, 0.5, 0.8, and 1 nm overlap. For case 2, 0.8 and 1 nm overlap.

also increased, but only notably for the larger particles of 10 and 5 nm, with the highest being for the 10 nm MSP with an almost 20% chance of being positively charged.

Both the low-density and high-density approaches suggest that MSP with $r \leq 1$ nm have a chance of being positively charged. This likelihood increases when assuming a silicate material composition over a metal oxide composition. If we take the probability to be somewhere between $15 - 40\%$, a total MSP population of 10^4 cm^{-3} would yield a negatively charged MSP population on the order of 10^3 cm^{-3}. If all these particles serve as condensation nuclei, it would be enough to account for the formation of NLC.

6.2.4 Summary

The model computations indicate that photoionization and photodetachment significantly influence the charge state of nano-sized MSP, as evidenced by the charging rates. When examining the number densities of mesospheric species, we observed that MSP with smaller radii are more likely to be negatively charged, while those with larger radii are more likely to be positively charged. Additionally, a larger fraction of the total MSP population is charged at smaller radii. Both the low-density and high-density methods consistently show that MSP with a radius of 1 nm or smaller have a minimum probability of at least 15% of being negatively charged. Notably, the probability for a negatively charged 1 nm olivine MSP using the low-density method is almost 40%. This suggests that MSP with $r \leq 1$ nm can carry a negative charge. This is significant considering the estimated number of MSP smaller than 1 nm in the mesosphere is on the order of 10^4 cm^{-3} (Megner et al., 2008b).

One potential improvement to this model could involve accounting for MSP of different radii having different total number densities. As MSP grow by coagulation while sedimenting, the number density decreases (Dunker, 2018). Currently, the model assumes uniform radii r and total number density N_{tot} for all MSP. By combining the approach to calculate charge probabilities and number densities with a model estimating MSP number densities for different radii, a more accurate depiction of the charged particle distribution could be obtained. Another improvement would be to expand the set of differential equations to account for additional chemical reactions occurring in the mesosphere. This is particularly important for considering the production of negatively charged ions and their impact on the overall charge balance.

It is also worth noting that the calculations for olivine are more limited compared to hematite. Firstly, olivine lacks refractive indexes in the $100 - 200$ nm range. The work function from table 5.2 was also much larger than what we could calculate for. Additionally, we did not have an estimate for the electron affinity. Consequently, I used the work function and electron affinity for hematite instead. It is possible that the effect of photoionization and photodetachment on olivine could be even lower than calculated here, primarily due to the high work function. Further investigation into this area is important, and obtaining experimental values for these missing parameters would be greatly beneficial.

7

Conclusion

In this thesis, I have examined the variability of the photoionization and photodetachment of MSP considering material factors and solar irradiation. I have also investigated how this variability affects the charge state of nanosized MSP.

The results of the model calculations indicate that photoionization and photodetachment do influence the charge states of nano-sized MSP, though perhaps not to the extent previously assumed. As shown in chapter 6, MSP are more likely to be charged without the presence of photodetachment, illustrating how photodetachment increases the number of neutral MSP. The impact of photoionization and photodetachment processes is highly dependent on the material properties of the MSP, particularly the refractive index and work function. We observed that silicates, being less absorbing, were less affected by photoionization and photodetachment compared to metal oxides.

Despite the partial neutralization of very small MSP, our observations indicate that a portion of the MSP population can still retain a charge. The models suggest that between 15% to 40% of MSP with $r \leq 1$ nm could remain negatively charged in the presence of photoionization and photodetachment, depending on their material composition. If the total number density of MSP smaller than 1 nm is 10^4 cm^{-3} as some models suggest (Megner et al., 2008b),

then a negatively charged MSP population on the order of 10^3 cm^{-3}. This would be sufficient to account for the number of ice particles needed for NLC. The model also indicated that larger MSP have a chance of being positively charged, with the strongest likelihood being for 10 nm sized MSP of hematite with a probability slightly larger than 15%.

It should be noted that the charge probabilities presented here are only estimates. The models have several limitations, including the assumption that all MSP have the same size and number density, which does not account for the actual size distribution of MSP. Future work could involve applying the calculation approach presented here to a modeled population size distribution of MSP to obtain a more accurate representation of the charge distribution across the entire population.

Furthermore, we noted that the photoionization of materials with high work functions was significantly affected by the lack of spectral range in the refractive indices. Specifically, a lack of refractive indices in the FUV range led to up to a 25% difference in the calculated photoionization rates. Acquiring refractive indices for potential MSP materials in this spectral range would be highly beneficial. Additionally, obtaining work functions and electron affinities for the possible materials, perhaps for both bulk material and molecular clusters, would greatly improve the accuracy of calculations and models, allowing for the utilization of higher work functions.

This thesis has also attempted to address the application of equations designed for microscopic particles and the use of parameters for bulk materials to represent MSP of sizes perhaps too small to be properly represented by these. However, there is limited research available on the lower size limit of MSP for holding a charge. Investigating this lower size limit further could be an interesting avenue for future exploration.

Bibliography

Asmus, H., Robertson, S., Dickson, S., Friedrich, M., and Megner, L. (2015). Charge balance for the mesosphere with meteoric dust particles. *Journal of Atmospheric and Solar-Terrestrial Physics*, 127:137–149.

Australian Antarctic Program (2020). Atmospheric regions. Accessed: 08.04.2024 at https://www.antarctica.gov.au/about-antarctica/ice-and-atmosphere/atmosphere/from-the-ground-to-space/atmospheric-regions/.

Bardeen, C., Toon, O., Jensen, E., Marsh, D., and Harvey, V. (2008). Numerical simulations of the three-dimensional distribution of meteoric dust in the mesosphere and upper stratosphere. *Journal of Geophysical Research: Atmospheres*, 113(D17).

Baumann, C., Rapp, M., Kero, A., and Enell, C.-F. (2013). Meteor smoke influences on the D-region charge balance – review of recent in situ measurements and one-dimensional model results. *Annales Geophysicae*, 31(11):2049–2062.

Bohren, C. F. and Huffman, D. R. (1983). *Absorption and Scattering of Light by Small Particles*.

Bohren, C. F. and Olivero, J. J. (1984). Evidence for haematite particles at 60 km altitude. *Nature*, 310(5974):216–218.

Brekke, A. (2013). *Physics of the Upper Polar Atmosphere*. Springer-Praxis Books in Environmental Sciences. Springer Published in association with Praxis Pub, Berlin Chichester, U.K, 2nd ed edition.

Butchart, N. (2022). The stratosphere: A review of the dynamics and variability. *Weather and Climate Dynamics*, 3(4):1237–1272.

Dorschner, J., Begemann, B., Henning, T., Jaeger, C., and Mutschke, H. (1995). Steps toward interstellar silicate mineralogy. II. Study of Mg-Fe-silicate glasses of variable composition. *Astronomy and Astrophysics*, 300:503.

Draine, BT. and Sutin, B. (1987). Collisional charging of interstellar grains. *The Astrophysical Journal*, 320:803–817.

Dunker, T. (2018). Detection and effects of meteoric smoke particles in the atmosphere.

Fonin, M., Pentcheva, R., Dedkov, Yu. S., Sperlich, M., Vyalikh, D. V., Scheffler, M., Rüdiger, U., and Güntherodt, G. (2005). Surface electronic structure of the Fe 3 O 4 (100) : Evidence of a half-metal to metal transition. *Physical Review B*, 72(10):104436.

Fu, Q. (2003). Radiation (solar). *Encyclopedia of atmospheric sciences*, pages 1859–1863.

Gumbel, J. and Megner, L. (2009). Charged meteoric smoke as ice nuclei in the mesosphere: Part 1—a review of basic concepts. *Journal of Atmospheric and Solar-Terrestrial Physics*, 71(12):1225–1235.

Gunár, S., Schwartz, P., Koza, J., and Heinzel, P. (2020). Quiet-Sun hydrogen Lyman- α line profile derived from SOHO/SUMER solar-disk observations. *Astronomy & Astrophysics*, 644:A109.

He, C. (2017). Work function of α-Fe_{2}O_{3} : A DFT calculation.

Henning, T., Begemann, B., Mutschke, H., and Dorschner, J. (1995). Optical properties of oxide dust grains. *Astronomy and Astrophysics Supplement Series*, 112:143.

Hervig, M. E., Brooke, J. S. A., Feng, W., Bardeen, C. G., and Plane, J. M. C. (2017). Constraints on Meteoric Smoke Composition and Meteoric Influx Using SOFIE Observations With Models. *Journal of Geophysical Research: Atmospheres*, 122(24).

Hervig, M. E., Gordley, L. L., Deaver, L. E., Siskind, D. E., Stevens, M. H., Russell III, J. M., Bailey, S. M., Megner, L., and Bardeen, C. G. (2009). First satellite observations of meteoric smoke in the middle atmosphere. *Geophysical Research Letters*, 36(18).

Hervig, M. E., Plane, J. M., Siskind, D. E., Feng, W., Bardeen, C. G., and Bailey, S. M. (2021). New global meteoric smoke observations from SOFIE: Insight regarding chemical composition, meteoric influx, and hemispheric asymmetry. *Journal of Geophysical Research: Atmospheres*, 126(13):e2021JD035007.

Hunten, D. M., Turco, R. P., and Toon, O. B. (1980). Smoke and dust particles of meteoric origin in the mesosphere and stratosphere. *Journal of Atmospheric Sciences*, 37(6):1342–1357.

Jaeger, C., Mutschke, H., Begemann, B., Dorschner, J., and Henning, Th. (1994). Steps toward interstellar silicate mineralogy. I. Laboratory results of a silicate glass of mean cosmic composition. *Astronomy and Astrophysics*, 292:641–655.

Jensen, E. J. and Thomas, G. E. (1991). Charging of mesospheric particles: Implications for electron density and particle coagulation. *Journal of Geophysical Research: Atmospheres*, 96(D10):18603–18615.

Kiliani, J., Baumgarten, G., Lübken, F.-J., Berger, U., and Hoffmann, P. (2013). Temporal and spatial characteristics of the formation of strong noctilucent clouds. *Journal of Atmospheric and Solar-Terrestrial Physics*, 104:151–166.

Klekociuk, A. R., Brown, P. G., Pack, D. W., ReVelle, D. O., Edwards, W. N., Spalding, R. E., Tagliaferri, E., Yoo, B. B., and Zagari, J. (2005). Meteoritic dust from the atmospheric disintegration of a large meteoroid. *Nature*, 436(7054):1132–1135.

Knappmiller, S., Rapp, M., Robertson, S., and Gumbel, J. (2011). Charging of meteoric smoke and ice particles in the mesosphere including photoemission and photodetachment rates. *Journal of atmospheric and solar-terrestrial physics*, 73(14-15):2212–2220.

Kraushofer, F., Jakub, Z., Bichler, M., Hulva, J., Drmota, P., Weinold, M., Schmid, M., Setvin, M., Diebold, U., Blaha, P., and Parkinson, G. S. (2018). Atomic-Scale Structure of the Hematite α-Fe $_2$ O $_3$ ($1\bar{1}02$) "R-Cut" Surface. *The Journal of Physical Chemistry C*, 122(3):1657–1669.

Latteck, R. and Bremer, J. (2017). Long-term variations of polar mesospheric summer echoes observed at andøya (69 n). *Journal of atmospheric and solar-terrestrial physics*, 163:31–37.

Li, A. (2008). Optical Properties of Dust. In Mann, I., Nakamura, A., and

Mukai, T., editors, *Small Bodies in Planetary Systems*, volume 758, pages 1–22. Springer Berlin Heidelberg, Berlin, Heidelberg.

Li, X. Y., Gan, H., Mo, B., Wang, S. J., Wei, G. F., Tang, H., and Zhao, Y. Y. S. (2016). Indication of Mineral Work Function in Lunar Dust Electrostatic Migration. page 1993.

Megner, L. (2008). *Meteoric aerosols in the middle atmosphere*. PhD thesis, Meteorologiska institutionen (MISU).

Megner, L. and Gumbel, J. (2009). Charged meteoric particles as ice nuclei in the mesosphere: Part 2: A feasibility study. *Journal of atmospheric and solar-terrestrial physics*, 71(12):1236–1244.

Megner, L., Gumbel, J., Rapp, M., and Siskind, D. (2008a). Reduced meteoric smoke particle density at the summer pole – Implications for mesospheric ice particle nucleation. *Advances in Space Research*, 41(1):41–49.

Megner, L., Rapp, M., and Gumbel, J. (2006). Distribution of meteoric smoke–sensitivity to microphysical properties and atmospheric conditions. *Atmospheric Chemistry and Physics*, 6(12):4415–4426.

Megner, L., Siskind, D., Rapp, M., and Gumbel, J. (2008b). Global and temporal distribution of meteoric smoke: A two-dimensional simulation study. *Journal of Geophysical Research: Atmospheres*, 113(D3).

Mishchenko, M. I., Travis, L. D., and Lacis, A. A. (2002). *Scattering, absorption, and emission of light by small particles*. Cambridge university press.

Natanson, GL. (1960). On the theory of the charging of amicroscopic aerosol particles as a result of capture of gas ions. *Sov. Phys. Tech. Phys.*, 30:573–588.

Plane, J. M. C. (2003). Atmospheric Chemistry of Meteoric Metals. *Chemical Reviews*, 103(12):4963–4984.

Plane, J. M. C., Gumbel, J., Kalogerakis, K. S., Marsh, D. R., and Von Savigny, C. (2023). Opinion: Recent developments and future directions in studying the mesosphere and lower thermosphere. *Atmospheric Chemistry and Physics*, 23(20):13255–13282.

Prahl, S. (2023). Miepython: Pure python implementation of Mie scatter-

ing. Documentation available at `https://miepython.readthedocs.io`. Last accessed 16.04.2024.

Rapp, M. (2000). Capture rates of electrons and positive ions by mesospheric aerosol particles. *Journal of Aerosol Science*, 31(11):1367–1369.

Rapp, M. (2009). Charging of mesospheric aerosol particles: the role of photodetachment and photoionization from meteoric smoke and ice particles. In *Annales Geophysicae*, volume 27, pages 2417–2422. Copernicus Publications Göttingen, Germany.

Rapp, M. and Lübken, F.-J. (2001). Modelling of particle charging in the polar summer mesosphere: Part 1—General results. *Journal of Atmospheric and Solar-Terrestrial Physics*, 63(8):759–770.

Rapp, M. and Lübken, F.-J. (2004). Polar mesosphere summer echoes (pmse): Review of observations and current understanding. *Atmospheric chemistry and physics*, 4(11/12):2601–2633.

Rapp, M., Plane, J., Strelnikov, B., Stober, G., Ernst, S., Hedin, J., Friedrich, M., and Hoppe, U.-P. (2012a). In situ observations of meteor smoke particles (msp) during the geminids 2010: constraints on msp size, work function and composition. In *Annales geophysicae*, volume 30, pages 1661–1673. Copernicus Publications Göttingen, Germany.

Rapp, M., Plane, JMC., Strelnikov, B., Stober, G., Ernst, S., Hedin, J., Friedrich, M., and Hoppe, U.-P. (2012b). In situ observations of meteor smoke particles (MSP) during the Geminids 2010: Constraints on MSP size, work function and composition. In *Annales Geophysicae*, volume 30, pages 1661–1673. Copernicus Publications Göttingen, Germany.

Rapp, M. and Strelnikova, I. (2009). Measurements of meteor smoke particles during the ECOMA-2006 campaign: 1. Particle detection by active photoionization. *Journal of Atmospheric and Solar-Terrestrial Physics*, 71(3-4):477–485.

Rees, M. H. (1989). *Physics and chemistry of the upper atmosphere*. Cambridge University Press.

Rienstra-Kiracofe, J. C., Tschumper, G. S., Schaefer, H. F., Nandi, S., and Ellison, G. B. (2002). Atomic and Molecular Electron Affinities: Photoelectron Experiments and Theoretical Computations. *Chemical Reviews*, 102(1):231–

282.

Saunders, R. W. and Plane, J. M. (2011). A photo-chemical method for the production of olivine nanoparticles as cosmic dust analogues. *Icarus*, 212(1):373–382.

Stroud, W. G., Nordberg, W., Bandeen, W. R., Bartman, F. L., and Titus, P. (1959). Rocket-grenade observation of atmospheric heating in the Arctic. *Journal of Geophysical Research*, 64(9):1342–1343.

Venkat Ratnam, M., Patra, A., and Krishna Murthy, B. (2010). Tropical mesopause: Is it always close to 100 km? *Journal of Geophysical Research: Atmospheres*, 115(D6).

Wang, L.-S., Wu, H., and Desai, S. R. (1996). Sequential Oxygen Atom Chemisorption on Surfaces of Small Iron Clusters. *Physical Review Letters*, 76(25):4853–4856.

Willson, R. C. and Hudson, H. S. (1991). The Sun's luminosity over a complete solar cycle. *Nature*, 351(6321):42–44.

Wiscombe, W. J. (1979). *Mie scattering calculations: Advances in technique and fast, vector-speed computer codes*, volume 10. National Technical Information Service, US Department of Commerce.

Woods, T. N., Chamberlin, P. C., Harder, J. W., Hock, R. A., Snow, M., Eparvier, F. G., Fontenla, J., McClintock, W. E., and Richard, E. C. (2009). Solar Irradiance Reference Spectra (SIRS) for the 2008 Whole Heliosphere Interval (WHI). *Geophysical Research Letters*, 36(1):2008GL036373.

Woods, T. N. and DeLand, M. T. (2021). An Improved Solar Spectral Irradiance Composite Record. *Earth and Space Science*, 8(8):e2021EA001740.

Zirin, H. (2012). Solar constant. *Encyclopedia Britannica*.

Appendix A: Additional figures

This appendix shows plots of the calculations done in chapter 6 using all six materials. The same input variables as discussed in chapter 6 is used.

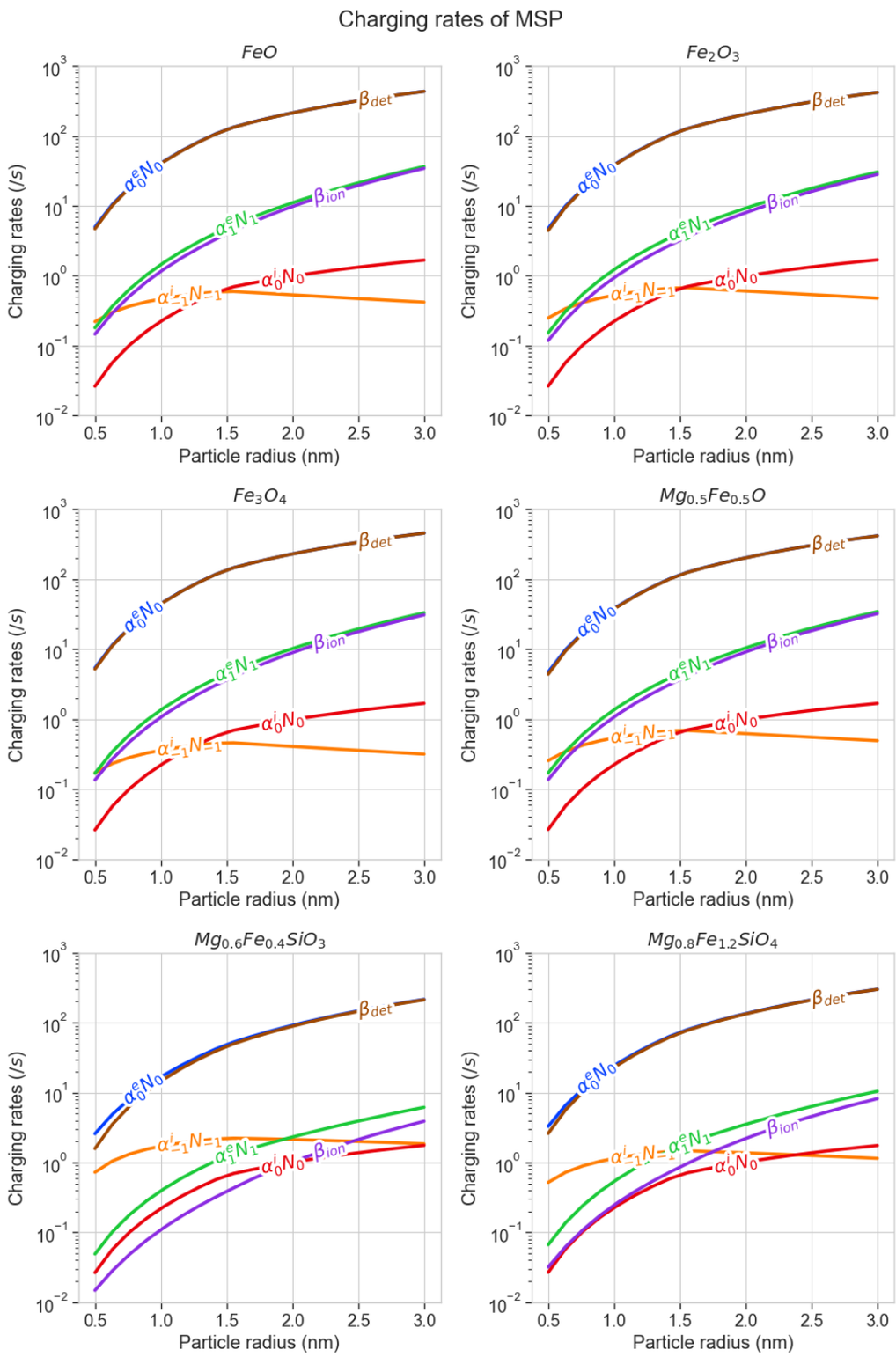

Figure 1: Charging rates of MSP consisting of six different materials for a range of MSP radii.

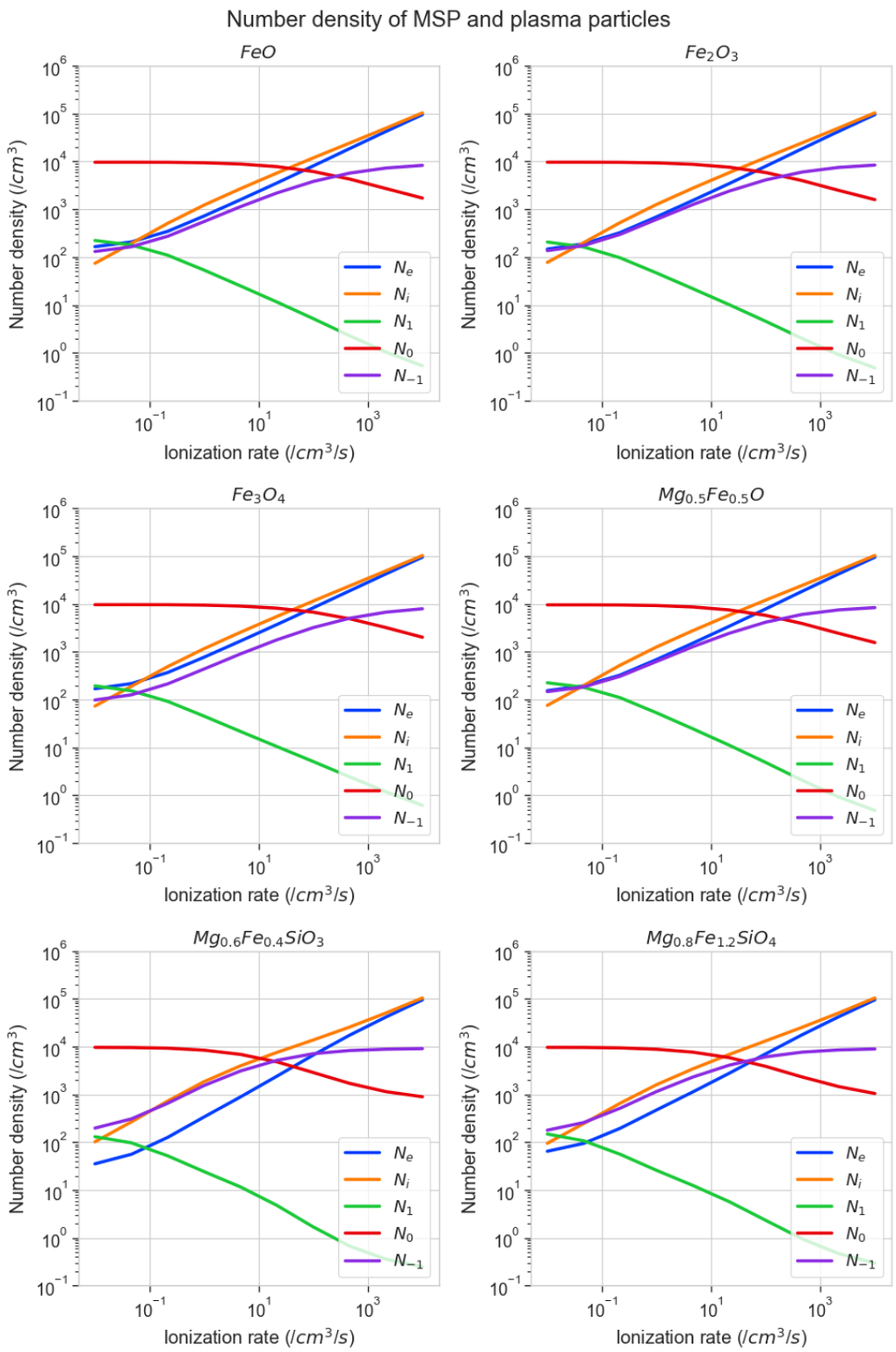

Figure 2: Number density of electrons, ions and positively, neutral and negatively charged MSP consisting of six different materials with varying ionization rate.

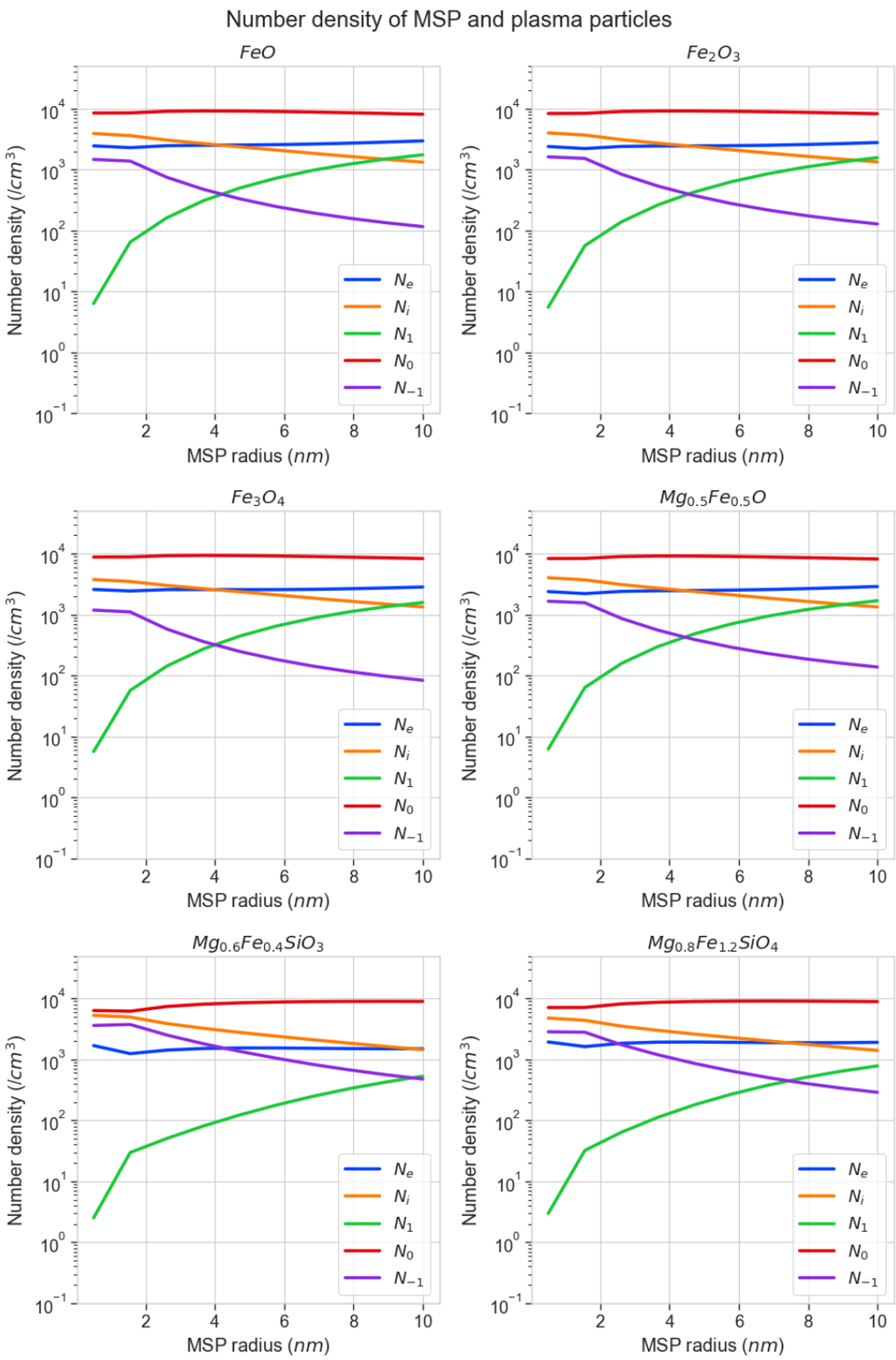

Figure 3: Number density of electrons, ions and positively, neutral and negatively charged MSP consisting of six different materials with varying MSP radii.

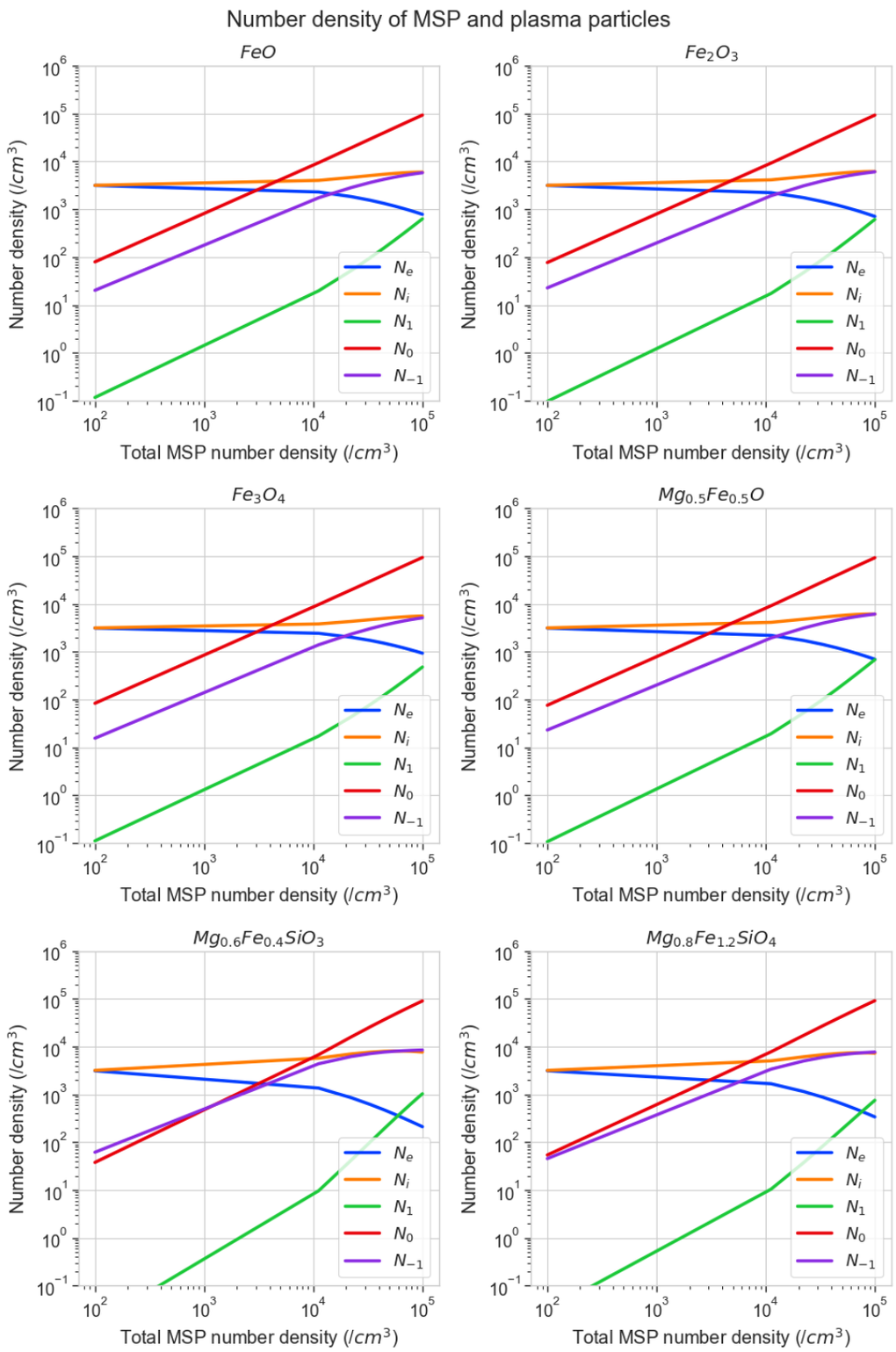

Figure 4: Number density of electrons, ions and positively, neutral and negatively charged MSP consisting of six different materials with varying total MSP populations.

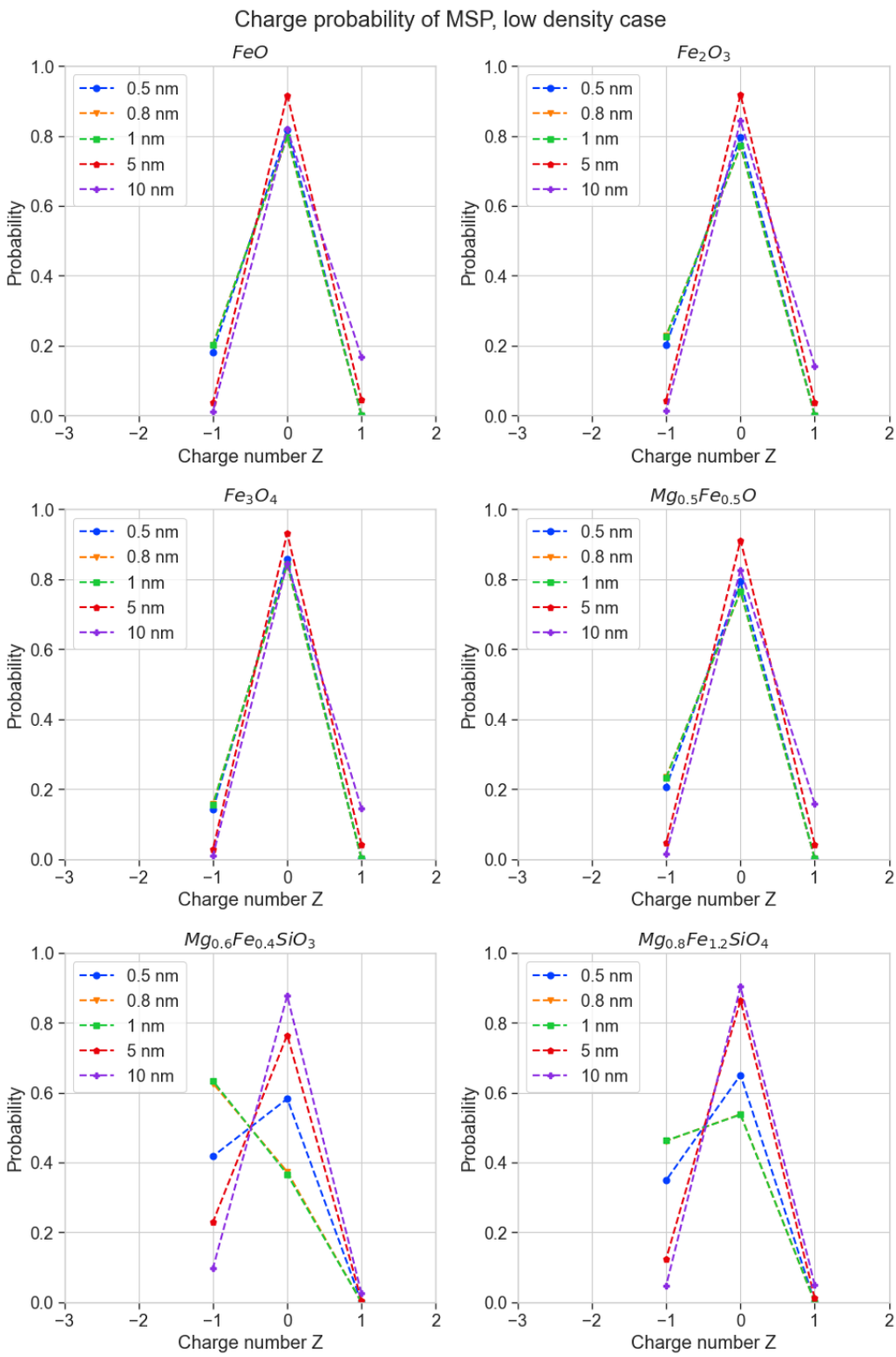

Figure 5: Charge probability of MSP of different materials. Calculated using the low density method.

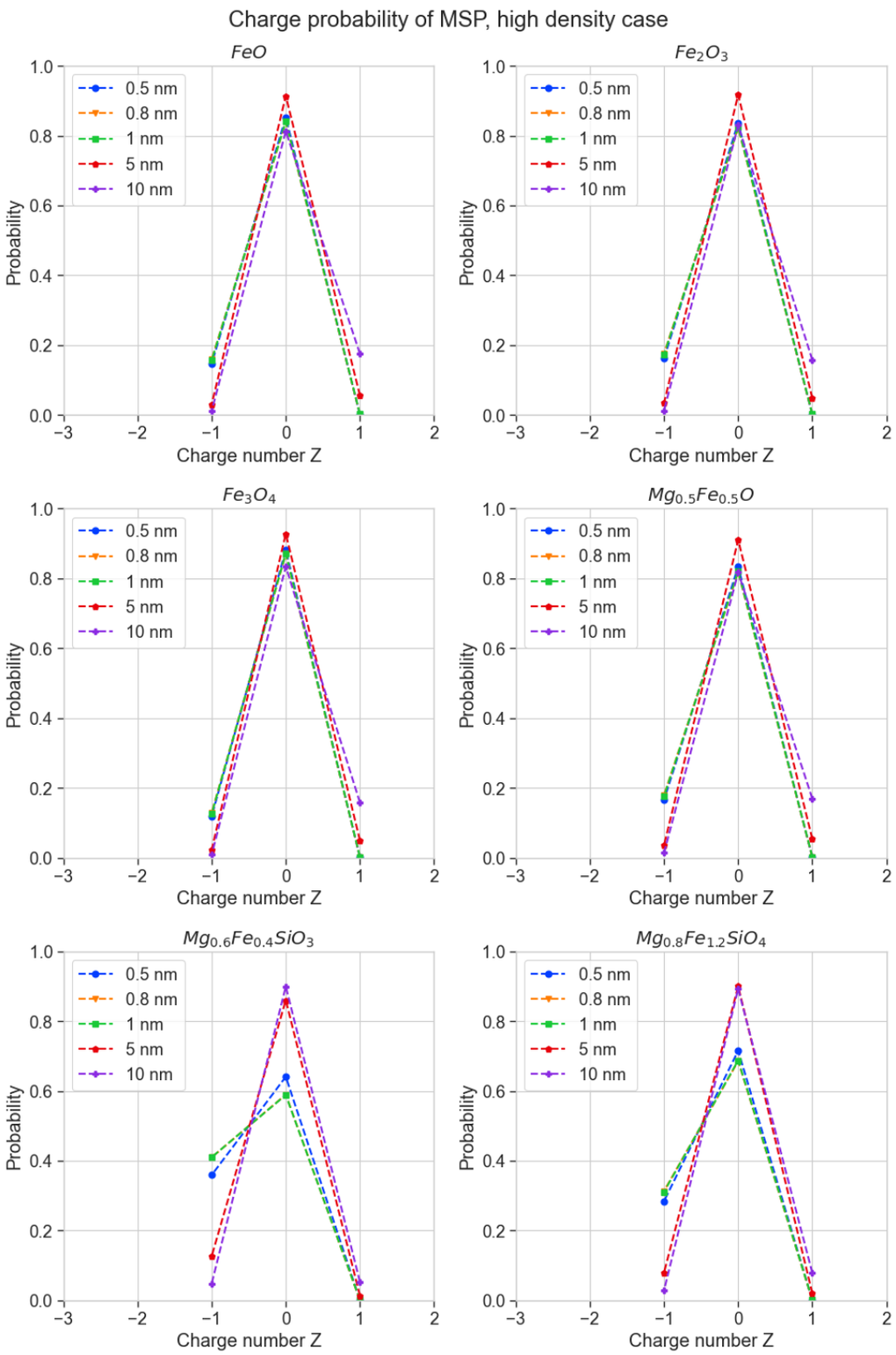

Figure 6: Charge probability of MSP of different materials. Calculated using the high density method.

Appendix B: Programming

.1 functions.py

The python program below contains all the functions used in my calculations and is imported in both the *photoionization.py* and *charge_state.py* files.

```
import miepython
import numpy as np
import matplotlib.pyplot as plt
import scipy as sp
from scipy.integrate import odeint
import seaborn as sns
import matplotlib as mpl
from labellines import *
from scipy.interpolate import interp1d
import pandas as pd
import matplotlib.ticker as mtick
from scipy import integrate

def import_oc(file_name, skip_lines):
    '''
    Importing optical constants.
    File in format wl, n, k.
    Returns three lists of wl, n, k

    file_name: path+name of file as a sting.
    skip_lines: lines to skip when importing the data
    '''
    file = np.genfromtxt(file_name, delimiter='\t', skip_header=skip_lines)

    file_lam = file[:, 0] #wavelenght in microns
    file_mre = file[:, 1] #real part of refractive index
    file_mim = file[:, 2] #imaginary part of refractive index

    return file

def abs_cross(r, material):
    '''
    Calculates absolute cross section in microns^2

    r : radius of particle in microns
    material : nested list of optical constants of material wl (microns), n, k
    '''
    geometric_cross_section = np.pi * r**2
    x = 2*np.pi*r/material[:, 0]  #size parameter
    m = material[:, 1] - 1.0j * material[:, 2] #complex index of refraction

    qext, qsca, qback, g = miepython.mie(m,x)
    absorb  = (qext - qsca) * geometric_cross_section

    return absorb

def cut_off(E):
    '''
    Calculates the cut-off wavelenght given a work function or electron affinity.

    E: work function or elelctron affinity in eV
    '''
    return 1.23984193/E
```

```
54
55 def closest_index(lst, N):
56     '''
57     Returns the index of the object in lst closest to N.
58
59     lst: list
60     N: number you want to find in the list
61     '''
62     lst = np.asarray(lst)
63     return (np.abs(lst - N)).argmin()
64
65 def photoionization(material, r, Y, E):
66     '''
67     Calculates the photoionization rate in photons/second.
68
69     material: nested list of optical constants of material wl (microns), n, k
70     r: radius of particle in microns
71     Y: quantum yield of photoelectrons
72     E: energy of photon in eV
73     '''
74
75     solar = np.genfromtxt('lasp_gsfc_solarspectrum.txt', delimiter='\t')
76     solar_irrad = solar[:, 2]*1e-3 #mW/m^2/nm to W/m^2/nm
77     solar_wl = solar[:, 1] #nm
78
79
80     #Converting to microns
81     h = 6.626070e-34 #Js
82     c = 2.998e8 #m/s
83
84     lam_cut = cut_off(E)
85     solar_wl = solar_wl*1e-3 #nm to microns
86     solar_irrad = solar_irrad*1e-3 #W/m^2/nm to W/m^2/microns
87
88     #Calculating absorption cross section
89     sigma = abs_cross(r, material)*1e-12 #microns^2 to m^2
90
91
92     mat_cut = closest_index(material[:, 0], lam_cut)
93     sigma = np.asarray(sigma[0:mat_cut])
94     wl = material[:, 0][0:mat_cut] #microns
95
96     new_solar_wl= []
97     new_solar_irrad = []
98
99     for i in range(len(wl)):
100         new_solar_wl.append(solar_wl[closest_index(solar_wl, wl[i])])
101         new_solar_irrad.append(solar_irrad[closest_index(solar_wl, wl[i])])
102
103
104     new_solar_wl = np.asarray(new_solar_wl)
105     new_solar_irrad = np.asarray(new_solar_irrad)
106     new_solar_irrad = new_solar_irrad/((h*c)/(wl*1e-6))
107
108     abs_photon = new_solar_irrad*sigma*Y #abs cross section * solar photon flux: photon
109
110     photoionization = integrate.simpson(abs_photon, x=wl*1e3) #1/s
111
112     if photoionization == 0:
113         return None
114
115     else:
116         return photoionization #s
117
118 def res(material, E):
119     '''
120     Calculates the error in matching the wavelenght lists of the solar irradiance and the refractive
      index
121
122     material: nested list of optical constants of material wl (microns), n, k
123     E: work function or electron affinity in eV
124     '''
125
126     solar = np.genfromtxt('lasp_gsfc_solarspectrum.txt', delimiter='\t')
127     solar_wl = solar[:, 1] #nm
128
129     #Converting to microns
130     solar_wl = solar_wl*1e-3 #nm to microns
131
132     wl = material[:, 0]
133
134     new_solar_wl= []
135
```

```
    for i in range(len(wl)):
        new_solar_wl.append(solar_wl[closest_index(solar_wl, wl[i])])

    new_solar_wl = np.asarray(new_solar_wl)

    residuals = new_solar_wl - wl

    return residuals*1e3 #micron to nm



def odes(x, t, q, r):
    '''
    Solves the kintetic rate equations without photoionization.
    Output: electron density, ion density, positive MSP, neutral MSP, negative MSP in #/cm^3

    x: start_variables of MSP in the order n_po, n_0, n_ne
    t: time of integration
    q: ionization rate of electrons and ions
    r: radius of MSP in nm
    '''
    #constants
    r_cm = r*1e-7 #radius of the particle (cm)
    T = 200 #electron and ion temperature (K)
    k = 1.3807e-16  #Boltzmann constant (cm^2 g s^-2 K^-1)
    e = 4.8032e-10 #elementary charge (esu/cgs units/ cm^3/2 g^1/2 s^-1))
    m_e = 9.11e-28 #electron mass (g)
    amu = 1.66e-24 #atomic mass unit (g)
    m_i = 50*amu #ion mass (g)
    g_e = 1.61 #0.8 dimensionless constant
    g_i = g_e
    i_fmp = 1 #free mean path of ions (cm)

    a_ie = 1e-6 #electron-ion recombination rate (/cm^3/s)

    c_e = np.sqrt((8*k*T)/(np.pi*m_e)) #electron thermal velocity (cm/s)
    c_i = np.sqrt((8*k*T)/(np.pi*m_i)) #ion thermal velocity (cm/s)

    if r<0.25:
        G=0
    elif r>=0.25 and r<=1.5:
        G=0.8*r-0.2
    else:
        G=1

    #Attachment coefficients
    a_epo = np.pi*(r_cm**2)*c_e*(e**2/(k*T*r_cm)*(1+(16/81)*(i_fmp/r_cm)*(e**2/(k*T*i_fmp))**2)) #
     electron-positive msp recombination rate (/cm^3/s)
    a_e0 = G*np.pi*(r_cm**2)*c_e*(1 + np.sqrt((np.pi*e**2)/(2*k*T*r_cm))) #electron-neutral msp
     recombination rate (/cm^3/s)
    a_ene = np.pi*(r_cm**2)*c_e*(g_e**2)*np.exp(-((e**2)/(g_e*k*T*r_cm))*(1 - 1/(2*g_e*(g_e**2-1)))) #
     electron-negative msp recombination rate (/cm^3/s)

    a_ipo = np.pi*(r_cm**2)*c_i*(g_i**2)*np.exp(-((e**2)/(g_i*k*T*r_cm))*(1 - 1/(2*g_e*(g_i**2-1)))) #
     ion-positive msp recombination rate (/cm^3/s)
    a_i0 = G*np.pi*(r_cm**2)*c_i*(1 + np.sqrt((np.pi*e**2)/(2*k*T*r_cm))) #ion-neutral msp
     recombination rate (/cm^3/s)
    a_ine = np.pi*(r_cm**2)*c_i*(e**2/(k*T*r_cm)*(1+(16/81)*(i_fmp/r_cm)*(e**2/(k*T*i_fmp))**2)) #ion-
     negative msp recombination rate (/cm^3/s)

    #initial values
    n_e = x[0]
    n_i = x[1]
    n_po = x[2]
    n_0 = x[3]
    n_ne = x[4]

    #differential equations
    dn_e = q - a_ie*n_e*n_i - n_e*(a_epo*n_po + a_e0*n_0)
    dn_i = q - a_ie*n_e*n_i - n_i*(a_i0*n_0 + a_ine*n_ne)

    dn_po = a_i0*n_0*n_i - (a_epo*n_e)*n_po
    dn_0 = a_ine*n_ne*n_i + a_epo*n_po*n_e - (a_e0*n_e + a_i0*n_i)*n_0
    dn_ne = -(a_ine*n_i)*n_ne + a_e0*n_0*n_e

    return [dn_e, dn_i, dn_po, dn_0, dn_ne]



def odes_photoion(x, t, q, r, material, Y, WF, EA):
    '''
    Solves the kintetic rate equations with photoionization and photodetachment.
    Output: electron density, ion density, positive MSP, neutral MSP, negative MSP in #/cm^3
```

```
213
214     x: start_variables in the order n_e, n_i, n_po, n_0, n_ne
215     t: time of integration
216     q: ionization rate of electrons and ions
217     r: radius of MSP in  nm
218     material: nested list of optical constants of material wl (microns), n, k
219     Y: quantum yield of photoelectrons
220     WF: work function in eV
221     EA: electron affinity in eV
222     '''
223     #constants
224     r_cm = r*1e-7 #radius of the particle (cm)
225     r_um = r*1e-3 #radius of the particle (um)
226     T = 200 #electron and ion temperature (K)
227     k = 1.3807e-16  #Boltzmann constant (cm^2 g s^-2 K^-1)
228     e = 4.8032e-10 #elementary charge (esu/cgs units/ cm^3/2 g^1/2 s^-1))
229     m_e = 9.11e-28 #electron mass (g)
230     amu = 1.66e-24 #atomic mass unit (g)
231     m_i = 30*amu #ion mass (g)
232     g_e = 1.61 #0.8 dimensionless constant
233     g_i = g_e
234     i_fmp = 1 #free mean path of ions (cm)
235
236     #Photoionization
237     b_pho = photoionization(material, r_um, Y, WF)
238     b_det = photoionization(material, r_um, 1, EA)
239
240     a_ie = 1e-6 #electron-ion recombination rate (/cm^3/s)
241
242     c_e = np.sqrt((8*k*T)/(np.pi*m_e)) #electron thermal velocity (cm/s)
243     c_i = np.sqrt((8*k*T)/(np.pi*m_i)) #ion thermal velocity (cm/s)
244
245     if r<0.25:
246         G=0
247     elif r>=0.25 and r<=1.5:
248         G=0.8*r-0.2
249     else:
250         G=1
251
252     #Attachment coefficients
253     a_epo = np.pi*(r_cm**2)*c_e*(e**2/(k*T*r_cm)*(1+(16/81)*(i_fmp/r_cm)*(e**2/(k*T*i_fmp))**2)) #
          electron-positive msp recombination rate (/cm^3/s)
254     a_e0 = G*np.pi*(r_cm**2)*c_e*(1 + np.sqrt((np.pi*e**2)/(2*k*T*r_cm))) #electron-neutral msp
          recombination rate (/cm^3/s)
255     a_ene = np.pi*(r_cm**2)*c_e*(g_e**2)*np.exp(-((e**2)/(g_e*k*T*r_cm))*(1 - 1/(2*g_e*(g_e**2-1)))) #
          electron-negative msp recombination rate (/cm^3/s)
256
257     a_ipo = np.pi*(r_cm**2)*c_i*(g_i**2)*np.exp(-((e**2)/(g_i*k*T*r_cm))*(1 - 1/(2*g_e*(g_i**2-1)))) #
          ion-positive msp recombination rate (/cm^3/s)
258     a_i0 = G*np.pi*(r_cm**2)*c_i*(1 + np.sqrt((np.pi*e**2)/(2*k*T*r_cm))) #ion-neutral msp
          recombination rate (/cm^3/s)
259     a_ine = np.pi*(r_cm**2)*c_i*(e**2/(k*T*r_cm)*(1+(16/81)*(i_fmp/r_cm)*(e**2/(k*T*i_fmp))**2)) #ion-
          negative msp recombination rate (/cm^3/s)
260
261     #initial values
262     n_e = x[0]
263     n_i = x[1]
264     n_po = x[2]
265     n_0 = x[3]
266     n_ne = x[4]
267
268     #differential equations
269     dn_e = q - a_ie*n_e*n_i - n_e*(a_epo*n_po + a_e0*n_0) + b_det*n_ne + b_pho*n_0 #electrons
270     dn_i = q - a_ie*n_e*n_i - n_i*(a_i0*n_0 + a_ine*n_ne) #pos. ions
271
272     dn_po = a_i0*n_0*n_i - (a_epo*n_e)*n_po + b_pho*n_0 #pos. msp
273     dn_0 = a_ine*n_ne*n_i + a_epo*n_po*n_e - (a_e0*n_e + a_i0*n_i)*n_0 +  b_det*n_ne -b_pho*n_0 #
          neutral msp
274     dn_ne = -(a_ine*n_i)*n_ne + a_e0*n_0*n_e - b_det*n_ne #neg. msp
275
276     return [dn_e, dn_i, dn_po, dn_0, dn_ne]
277
278 def ion_den(points, t_0, r, N_tot, photoion=True, material=None, WF=None, EA=None):
279     '''
280     Function to calculate the densities of the different species as a function of the ionization rate.
281     Returns dataframe with columns: ionization, electron density, ion density, positive MSP density,
          neutral MSP density, negative MSP density
282     in #/cm^3
283
284     points: number of points between start and stop ionization rate
285     t_0: time of integration
286     r: radius of MSP in nm
287     N_tot: initial number of neutral particles
```

```
    photoion: True or False, include photoionization and detachment or not
    material: nested list of optical constants of material wl (microns), n, k
    WF: work function in eV
    EA: electron affinity in eV
    '''
    t = np.arange(0, t_0, 1)

    ion = np.logspace(-2, 4, points)

    N_e = []
    N_i = []
    N_po = []
    N_0 = []
    N_ne = []


    if photoion == True:
        for i in ion:
            initial_values = [np.sqrt(i/1e-6), np.sqrt(i/1e-6), 0, N_tot, 0]
            ode = odeint(odes_photoion, initial_values, t, args=(i, r, material, 1, WF, EA))
            N_e.append(ode[-1,0])
            N_i.append(ode[-1,1])
            N_po.append(ode[-1,2])
            N_0.append(ode[-1,3])
            N_ne.append(ode[-1,4])

    else:
        for i in ion:
            initial_values = [np.sqrt(i/1e-6), np.sqrt(i/1e-6), 0, N_tot, 0]
            ode = odeint(odes, initial_values, t, args=(i, r))
            N_e.append(ode[-1,0])
            N_i.append(ode[-1,1])
            N_po.append(ode[-1,2])
            N_0.append(ode[-1,3])
            N_ne.append(ode[-1,4])

    data = pd.DataFrame(list(zip(ion, N_e, N_i, N_po, N_0, N_ne)), columns=['ion', 'N_e', 'N_i', 'N_po
      ', 'N_0', 'N_ne'])
    return data

def rad_den(points, t_0, q, N_tot, photoion=True, material=None, WF=None, EA=None):
    '''
    Function to calculate the densities of the different species as a function of MSP radius.
    Returns dataframe with columns: MSP radius, electron density, ion density, positive MSP density,
      neutral MSP density, negative MSP density
    in #/cm^3

    points: number of points between start and stop ionization rate
    t_0: time of integration
    q: ionization rate of electrons and ions
    N_tot: initial number of neutral particles
    photoion: True or False, include photoionization and detachment or not
    material: nested list of optical constants of material wl (microns)| n | k
    WF: work function in eV
    EA: electron affinity in eV
    '''
    t = np.arange(0, t_0, 1)

    r = np.linspace(0.5, 10, points)

    N_e = []
    N_i = []
    N_po = []
    N_0 = []
    N_ne = []


    if photoion == True:
        for i in r:
            initial_values = [np.sqrt(q/1e-6), np.sqrt(q/1e-6), 0, N_tot, 0]
            ode = odeint(odes_photoion, initial_values, t, args=(q, i, material, 1, WF, EA))
            N_e.append(ode[-1,0])
            N_i.append(ode[-1,1])
            N_po.append(ode[-1,2])
            N_0.append(ode[-1,3])
            N_ne.append(ode[-1,4])

    else:
        for i in r:
            initial_values = [np.sqrt(i/1e-6), np.sqrt(i/1e-6), 0, N_tot, 0]
            ode = odeint(odes, initial_values, t, args=(q, i))
            N_e.append(ode[-1,0])
            N_i.append(ode[-1,1])
```

```
369             N_po.append(ode[-1,2])
370             N_0.append(ode[-1,3])
371             N_ne.append(ode[-1,4])
372
373     data = pd.DataFrame(list(zip(r, N_e, N_i, N_po, N_0, N_ne)), columns=['r', 'N_e', 'N_i', 'N_po', '
          N_0', 'N_ne'])
374     return data
375
376 def tot_den(points, t_0, q, r, photoion=True, material=None, WF=None, EA=None):
377     '''
378     Function to calculate the densities of the different species as a function of the ttoal MSP number
          density.
379     Returns dataframe with columns: Total MSP density, electron density, ion density, positive MSP
          density, neutral MSP density, negative MSP density
380     in #/cm^3
381
382     points: number of points between start and stop ionization rate
383     t_0: time of integration
384     q: ionization rate of electrons and ions
385     r: radius of MSP in nm
386     photoion: True or False, include photoionization and detachment or not
387     material: nested list of optical constants of material wl (microns)| n | k
388     Y: quantum yield of photoelectrons
389     WF: work function in eV
390     EA: electron affinity in eV
391     '''
392     t = np.arange(0, t_0, 1)
393
394     N_tot = np.linspace(1e2, 1e5, points)
395
396     N_e = []
397     N_i = []
398     N_po = []
399     N_0 = []
400     N_ne = []
401
402
403     if photoion == True:
404         for i in N_tot:
405             initial_values = [np.sqrt(q/1e-6), np.sqrt(q/1e-6), 0, i, 0]
406             ode = odeint(odes_photoion, initial_values, t, args=(q, r, material, 1, WF, EA))
407             N_e.append(ode[-1,0])
408             N_i.append(ode[-1,1])
409             N_po.append(ode[-1,2])
410             N_0.append(ode[-1,3])
411             N_ne.append(ode[-1,4])
412
413     else:
414         for i in N_tot:
415             initial_values = [np.sqrt(i/1e-6), np.sqrt(i/1e-6), 0, i, 0]
416             ode = odeint(odes, initial_values, t, args=(q, r))
417             N_e.append(ode[-1,0])
418             N_i.append(ode[-1,1])
419             N_po.append(ode[-1,2])
420             N_0.append(ode[-1,3])
421             N_ne.append(ode[-1,4])
422
423     data = pd.DataFrame(list(zip(N_tot, N_e, N_i, N_po, N_0, N_ne)), columns=['tot', 'N_e', 'N_i', '
          N_po', 'N_0', 'N_ne'])
424     return data
425
426
427 def charging_rates(r, ne, n_tot, Q, photo_ion=False, material=None, Y=None, WF=None, EA=None):
428     '''
429     Function to calculate the different charging rates for a given particle radius and electron/ion
          number density in /s.
430
431     r: list of radii of the particle (nm)
432     ne: electron density (cm^-3)
433     n_tot: total MSP number density
434     Q: electron-ion pair production/ionization
435     photoion: True/False, include photoionization in calculation
436     material: material of the particle
437     Y: work function of the material (eV)
438     WF: work function of the material (eV)
439     EA: electron affinity of the material (eV)
440     '''
441     #constants
442     T = 200 #electron and ion temperature (K)
443     k = 1.3807e-16  #Boltzmann constant (cm^2 g s^-2 K^-1)
444     e = 4.8032e-10 #elementary charge (esu/cgs units/ cm^3/2 g^1/2 s^-1))
445     m_e = 9.11e-28 #electron mass (g)
446     amu = 1.66e-24 #atomic mass unit (g)
```

```
447     m_i = 50*amu #ion mass (g)
448     g_e = 1.61 #dimensionless constant
449     g_i = g_e
450     i_fmp = 1 #free mean path of ions (cm)
451 
452     c_e = np.sqrt((8*k*T)/(np.pi*m_e)) #electron thermal velocity (cm/s)
453     c_i = np.sqrt((8*k*T)/(np.pi*m_i)) #ion thermal velocity (cm/s)
454 
455     r_cm = r*1e-7 #particle radius (cm)
456     r_um = r*1e-3 #particle radius (microns)
457 
458     a_epo = []
459     a_e0 = []
460     a_ene = []
461     a_ipo = []
462     a_i0 = []
463     a_ine = []
464     b_pho = []
465     b_det = []
466 
467     if photo_ion==True:
468         a_epo = []
469         a_e0 = []
470         a_ene = []
471         a_ipo = []
472         a_i0 = []
473         a_ine = []
474         b_pho = []
475         b_det = []
476 
477         for i in range(len(r)):
478 
479             if r[i]<0.25:
480                 G=0
481             elif r[i]>=0.25 and r[i]<=1.5:
482                 G=0.8*r[i]-0.2
483             else:
484                 G=1
485 
486             t = np.arange(0, 3000, 1)
487             densities = odeint(odes_photoion, [ne, ne, 0, n_tot, 0], t, args=(Q, r[i], material, Y, WF
    , EA))
488 
489             #Photoionization
490             b_pho.append(photoionization(material, r_um[i], Y, WF)*densities[-1,3])
491             b_det.append(photoionization(material, r_um[i], 1, EA)*densities[-1,4])
492 
493             #Plasma attachment
494             a_epo.append((np.pi*(r_cm[i]**2)*c_e*(e**2/(k*T*r_cm[i])*(1+(16/81)*(i_fmp/r_cm[i])*(e
    **2/(k*T*i_fmp))**2))*densities[-1,0]*densities[-1,2])) #electron-positive msp recombination
    rate (/cm^3/s)
495             a_e0.append((G*np.pi*(r_cm[i]**2)*c_e*(1 + np.sqrt((np.pi*e**2)/(2*k*T*r_cm[i])))*
    densities[-1,0]*densities[-1,3])) #electron-neutral msp recombination rate (/cm^3/s)
496             a_ene.append((np.pi*(r_cm[i]**2)*c_e*(g_e**2)*np.exp(-((e**2)/(g_e*k*T*r_cm[i]))*(1 -
    1/(2*g_e*(g_e**2-1))))*densities[-1,0]*densities[-1,4])) #electron-negative msp recombination
    rate (/cm^3/s)
497 
498             a_ipo.append((np.pi*(r_cm[i]**2)*c_i*(g_i**2)*np.exp(-((e**2)/(g_i*k*T*r_cm[i]))*(1 -
    1/(2*g_e*(g_i**2-1))))*densities[-1,1]*densities[-1,2])) #ion-positive msp recombination rate (/
    cm^3/s
499             a_i0.append((G*np.pi*(r_cm[i]**2)*c_i*(1 + np.sqrt((np.pi*e**2)/(2*k*T*r_cm[i])))*
    densities[-1,1]*densities[-1,3]) #ion-neutral msp recombination rate (/cm^3/s)
500             a_ine.append((np.pi*(r_cm[i]**2)*c_i*(e**2/(k*T*r_cm[i])*(1+(16/81)*(i_fmp/r_cm[i])*(e
    **2/(k*T*i_fmp))**2))*densities[-1,1]*densities[-1,4])) #ion-negative msp recombination rate (/
    cm^3/s)
501 
502 
503 
504         a_epo = np.array(a_epo)
505         a_e0 = np.array(a_e0)
506         a_ene = np.array(a_ene)
507         a_ipo = np.array(a_ipo)
508         a_i0 = np.array(a_i0)
509         a_ine = np.array(a_ine)
510         b_pho = np.array(b_pho)
511         b_det = np.array(b_det)
512 
513         charging_rates = np.stack((r, a_epo, a_e0, a_ene, a_ipo, a_i0, a_ine, b_pho, b_det), axis=0)
514 
515     else:
516         a_epo = []
517         a_e0 = []
518         a_ene = []
```

```
519         a_ipo = []
520         a_i0 = []
521         a_ine = []
522 
523         for i in range(len(r)):
524 
525             if r[i]<0.25:
526                 G=0
527             elif r[i]>=0.25 and r[i]<=1.5:
528                 G=0.8*r[i]-0.2
529             else:
530                 G=1
531 
532             t = np.arange(0, 3000, 1)
533             densities = odeint(odes, [ne, ne, 0, n_tot, 0], t, args=(Q, r[i]))
534 
535             #Plasma attachment
536             a_epo.append((np.pi*(r_cm[i]**2)*c_e*(e**2/(k*T*r_cm[i])*(1+(16/81)*(i_fmp/r_cm[i])*(e
        **2/(k*T*i_fmp))**2))*densities[-1,0]*densities[-1,2])) #electron-positive msp recombination
        rate (/cm^3/s)
537             a_e0.append((G*np.pi*(r_cm[i]**2)*c_e*(1 + np.sqrt((np.pi*e**2)/(2*k*T*r_cm[i])))*
        densities[-1,0]*densities[-1,3])) #electron-neutral msp recombination rate (/cm^3/s)
538             a_ene.append((np.pi*(r_cm[i]**2)*c_e*(g_e**2)*np.exp(-((e**2)/(g_e*k*T*r_cm[i]))*(1 -
        1/(2*g_e*(g_e**2-1))))*densities[-1,0]*densities[-1,4])) #electron-negative msp recombination
        rate (/cm^3/s)
539 
540             a_ipo.append((np.pi*(r_cm[i]**2)*c_i*(g_i**2)*np.exp(-((e**2)/(g_i*k*T*r_cm[i]))*(1 -
        1/(2*g_e*(g_i**2-1))))*densities[-1,1]*densities[-1,2])) #ion-positive msp recombination rate (/
        cm^3/s
541             a_i0.append((G*np.pi*(r_cm[i]**2)*c_i*(1 + np.sqrt((np.pi*e**2)/(2*k*T*r_cm[i]))))*
        densities[-1,1]*densities[-1,3]) #ion-neutral msp recombination rate (/cm^3/s)
542             a_ine.append((np.pi*(r_cm[i]**2)*c_i*(e**2/(k*T*r_cm[i])*(1+(16/81)*(i_fmp/r_cm[i])*(e
        **2/(k*T*i_fmp))**2))*densities[-1,1]*densities[-1,4])) #ion-negative msp recombination rate (/
        cm^3/s)
543 
544 
545 
546         a_epo = np.array(a_epo)
547         a_e0 = np.array(a_e0)
548         a_ene = np.array(a_ene)
549         a_ipo = np.array(a_ipo)
550         a_i0 = np.array(a_i0)
551         a_ine = np.array(a_ine)
552 
553         charging_rates = np.stack((r, a_epo, a_e0, a_ene, a_ipo, a_i0, a_ine), axis=0)
554 
555 
556     return charging_rates
557 
558 
559 def photoionization_solar(material, r, Y, E, solar):
560     '''
561     Function that calculates photoionization which takes solar irradiance as an input variable.
562     Output: photoionization in photons/second
563 
564     material: nested list of optical constants of material: wl (microns), n, k
565     r: radius of particle in microns
566     Y: quantum yield of photoelectrons
567     E: energy of photon in eV
568     solar: nested list of solar spectrum:  wl (nm) | irradiance (W/m^2/nm)
569 
570     returns the photoionization time constant in seconds/photon
571     '''
572     solar_irrad = solar[:, 1] #W/m^2/nm
573     solar_wl = solar[:, 0] #nm
574 
575 
576     #Converting to microns
577     hc = 1.23984193 #eV*microns
578     h = 1.0545718e-34 #J*s
579     c = 299792458 #m/s
580     lam_cut = cut_off(E)
581     solar_wl = solar_wl*1e-3 #nm to microns
582     solar_irrad = solar_irrad*1e-3 #W/m^2/nm to W/m^2/microns
583 
584     #Calculating absorption cross section
585     sigma = abs_cross(r, material)*1e-12 #microns^2 to m^2
586 
587     mat_cut = closest_index(material[:, 0], lam_cut)
588 
589     wl = material[:, 0][0:mat_cut]
590 
591     new_solar_wl= []
```

```
592     new_solar_irrad = []
593
594     for i in range(len(wl)):
595         new_solar_wl.append(solar_wl[closest_index(solar_wl, wl[i])])
596         new_solar_irrad.append(solar_irrad[closest_index(solar_wl, wl[i])])
597
598
599     new_solar_wl = np.asarray(new_solar_wl)
600     new_solar_irrad = np.asarray(new_solar_irrad)
601     new_solar_irrad = new_solar_irrad/((h*c)/(wl*1e-6))
602
603     photoionization = np.trapz(sigma[0:mat_cut]*(new_solar_irrad), wl)*Y #1/s
604
605     if photoionization == 0:
606         return None
607
608     else:
609         return photoionization #s
610
611
612 def charge_prob(n_e, n_tot, t_0, Q, r, photo_ion=False, material=None, WF=None, EA=None):
613     '''
614     Function to calculate the probability of a particle having a certain charge.
615     Returns list of probabilities for positive MSP, neutral MSP, negative MSP.
616
617     n_e: initial electron density
618     n_tot: total number of particles
619     t_0: time
620     Q: ionization rate
621     r: radius of MSP (nm)
622     photo_ion: boolean to determine if photoionization is included
623     material: optical constants of the material
624     Y: secondary electron yield
625     WF: work function of the material
626     EA: electron affinity of the material
627     '''
628
629     initial = [n_e, n_e, 0, n_tot, 0]
630     t = np.arange(0, t_0, 1)
631
632     N_e = []
633     N_i = []
634     N_po = []
635     N_0 = []
636     N_ne = []
637
638     if photo_ion == True:
639         #Calculate the densities of all species with photoionization
640         densities = odeint(odes_photoion, initial, t, args=(Q, r, material, 1, WF, EA))
641
642         N_e.append(densities[-1,0])
643         N_i.append(densities[-1,1])
644         N_po.append(densities[-1,2])
645         N_0.append(densities[-1,3])
646         N_ne.append(densities[-1,4])
647
648     else:
649         #Calculate the densities of all species without photoionization
650         densities = odeint(odes, initial, t, args=(Q, r))
651
652         N_e.append(densities[-1,0])
653         N_i.append(densities[-1,1])
654         N_po.append(densities[-1,2])
655         N_0.append(densities[-1,3])
656         N_ne.append(densities[-1,4])
657
658     prob = [N_po[0]/n_tot, N_0[0]/n_tot, N_ne[0]/n_tot]
659
660     return prob
661
662 def ode_lowden(x, t, r, ne, material, WF, EA):
663     '''
664     ODE for calculating number density using the low density approach and including photoionization.
665     Returns list of probabilities for positive MS, neutral MSP and negative MSP.
666
667     x: start_variables in the order n_1, n_0, n_-1
668     t: time of integration
669     r: radius of MSP in  nm
670     ne: electron/ion density /cm^3
671     material: nested list of optical constants of material wl (microns)| n | k
672     WF: work function in eV
673     EA: electron affinity in eV
674     '''
```

```
675     #constants
676     r_cm = r*1e-7 #radius of the particle (cm)
677     r_um = r*1e-3 #radius of the particle (um)
678     T = 200 #electron and ion temperature (K)
679     k = 1.3807e-16  #Boltzmann constant (cm^2 g s^-2 K^-1)
680     e = 4.8032e-10 #elementary charge (esu/cgs units/ cm^3/2 g^1/2 s^-1))
681     m_e = 9.11e-28 #electron mass (g)
682     amu = 1.66e-24 #atomic mass unit (g)
683     m_i = 50*amu #ion mass (g)
684     g_e = 1.62 #dimensionless constant
685     g_i = g_e
686     i_fmp = 1 #free mean path of ions (cm)
687
688     #Photoionization
689     b_pho = photoionization(material, r_um, 1, WF)
690     b_det = photoionization(material, r_um, 1, EA)
691
692     c_e = np.sqrt((8*k*T)/(np.pi*m_e)) #electron thermal velocity (cm/s)
693     c_i = np.sqrt((8*k*T)/(np.pi*m_i)) #ion thermal velocity (cm/s)
694
695
696     if r<0.25:
697         G=0
698     elif r>=0.25 and r<=1.5:
699         G=0.8*r-0.2
700     else:
701         G=1
702
703
704     #Rapp-rates
705     a_epo = np.pi*(r_cm**2)*c_e*(e**2/(k*T*r_cm)*(1+(16/81)*(i_fmp/r_cm)*(e**2/(k*T*i_fmp))**2)) #
          electron-positive msp recombination rate (/cm^3/s)
706     a_e0 = G*np.pi*(r_cm**2)*c_e*(1 + np.sqrt((np.pi*e**2)/(2*k*T*r_cm))) #electron-neutral msp
          recombination rate (/cm^3/s)
707
708     a_ipo = np.pi*(r_cm**2)*c_i*(g_i**2)*np.exp(-((e**2)/(g_i*k*T*r_cm))*(1 - 1/(2*g_e*(g_i**2-1)))) #
          ion-positive msp recombination rate (/cm^3/s)
709     a_i0 = G*np.pi*(r_cm**2)*c_i*(1 + np.sqrt((np.pi*e**2)/(2*k*T*r_cm))) #ion-neutral msp
          recombination rate (/cm^3/s)
710     a_ine = np.pi*(r_cm**2)*c_i*(e**2/(k*T*r_cm)*(1+(16/81)*(i_fmp/r_cm)*(e**2/(k*T*i_fmp))**2)) #ion-
          negative msp recombination rate (/cm^3/s)
711
712
713     #initial values
714     n_po = x[0] # Z=1
715     n_0 = x[1] # Z=0
716     n_ne = x[2] # Z=-1
717
718     #differential equations
719     n_e = ne #electrons
720     n_i = ne #pos. ions
721
722     dn_po = a_i0*n_0*n_i - a_epo*n_e*n_po + b_pho*n_0 #pos. msp
723     dn_0 = a_ine*n_ne*n_i + a_epo*n_po*n_e - (a_e0*n_e + a_i0*n_i)*n_0 + b_det*n_ne -b_pho*n_0 #
          neutral msp
724     dn_ne = -a_ine*n_i*n_ne + a_e0*n_0*n_e - b_det*n_ne #neg. msp
725
726     return [dn_po, dn_0, dn_ne]
727
728 def ode_lowden_0(x, t, r, ne):
729     '''
730     ODE for calculating number density using the low density approach and excluding photoionization.
731     Returns list of probabilities for positive MS, neutral MSP and negative MSP.
732
733     x: start_variables in the order n_po, n_0, n_ne
734     t: time of integration
735     r: radius of MSP in  nm
736     ne: electron/ion density /cm^3
737     '''
738     #constants
739     r_cm = r*1e-7 #radius of the particle (cm)
740     r_um = r*1e-3 #radius of the particle (um)
741     T = 200 #electron and ion temperature (K)
742     k = 1.3807e-16  #Boltzmann constant (cm^2 g s^-2 K^-1)
743     e = 4.8032e-10 #elementary charge (esu/cgs units/ cm^3/2 g^1/2 s^-1))
744     m_e = 9.11e-28 #electron mass (g)
745     amu = 1.66e-24 #atomic mass unit (g)
746     m_i = 50*amu #ion mass (g)
747     g_e = 1.62 #0.8 dimensionless constant
748     g_i = g_e
749     i_fmp = 1 #free mean path of ions (cm)
750
751     c_e = np.sqrt((8*k*T)/(np.pi*m_e)) #electron thermal velocity (cm/s)
```

```
752     c_i = np.sqrt((8*k*T)/(np.pi*m_i)) #ion thermal velocity (cm/s)
753
754     if r<0.25:
755         G=0
756     elif r>=0.25 and r<=1.5:
757         G=0.8*r-0.2
758     else:
759         G=1
760
761     #Rapp-rates
762     a_epo = np.pi*(r_cm**2)*c_e*(e**2/(k*T*r_cm)*(1+(16/81)*(i_fmp/r_cm)*(e**2/(k*T*i_fmp))**2)) #
          electron-positive msp recombination rate (/cm^3/s)
763     a_e0 = G*np.pi*(r_cm**2)*c_e*(1 + np.sqrt((np.pi*e**2)/(2*k*T*r_cm))) #electron-neutral msp
          recombination rate (/cm^3/s)
764     a_ene = np.pi*(r_cm**2)*c_e*(g_e**2)*np.exp(-((e**2)/(g_e*k*T*r_cm))*(1 - 1/(2*g_e*(g_e**2-1)))) #
          electron-negative msp recombination rate (/cm^3/s)
765
766     a_ipo = np.pi*(r_cm**2)*c_i*(g_i**2)*np.exp(-((e**2)/(g_i*k*T*r_cm))*(1 - 1/(2*g_e*(g_i**2-1)))) #
          ion-positive msp recombination rate (/cm^3/s)
767     a_i0 = G*np.pi*(r_cm**2)*c_i*(1 + np.sqrt((np.pi*e**2)/(2*k*T*r_cm))) #ion-neutral msp
          recombination rate (/cm^3/s)
768     a_ine = np.pi*(r_cm**2)*c_i*(e**2/(k*T*r_cm)*(1+(16/81)*(i_fmp/r_cm)*(e**2/(k*T*i_fmp))**2)) #ion-
          negative msp recombination rate (/cm^3/s)
769
770
771     #initial values
772     n_po = x[0]
773     n_0 = x[1]
774     n_ne = x[2]
775
776     #differential equations
777     n_e = ne #electrons
778     n_i = ne #pos. ions
779
780     dn_po = a_i0*n_0*n_i - a_epo*n_e*n_po  #pos. msp
781     dn_0 = a_ine*n_ne*n_i + a_epo*n_po*n_e - (a_e0*n_e + a_i0*n_i)*n_0  #neutral msp
782     dn_ne = - a_ine*n_i*n_ne + a_e0*n_0*n_e #neg. msp
783
784
785     return [dn_po, dn_0, dn_ne]
786
787
788 def charge_prob_lowden(n_e, n_tot, t_0, Q, r, photo_ion=False, material=None, WF=None, EA=None):
789     '''
790     Function to calculate the probability of a particle having a certain charge
791
792     n_e: initial electron density
793     n_tot: total number of particles
794     t: time
795     points: number of points in the integration
796     Q: ionization rate
797     r: radius of the particle (nm)
798     photo_ion: boolean to determine if photoionization is included
799     material: optical constants of the material
800     WF: work function of the material
801     EA: electron affinity of the material
802     '''
803
804     initial = [ 0, n_tot, 0]
805     t = np.arange(0, t_0, 1)
806
807     N_po = 0
808     N_0 = 0
809     N_ne = 0
810
811
812     if photo_ion == True:
813         #Calculate the densities of all species with photoionization
814         densities = odeint(ode_lowden, initial, t, args=(r, n_e, material, WF, EA))
815
816         N_po = densities[-1,0]
817         N_0 = densities[-1,1]
818         N_ne = densities[-1,2]
819
820     else:
821         #Calculate the densities of all species without photoionization
822         densities = odeint(ode_lowden_0, initial, t, args=(r, n_e))
823
824         N_po = densities[-1,0]
825         N_0 = densities[-1,1]
826         N_ne = densities[-1,2]
827
828
```

```
829     prob = [N_po/n_tot, N_O/n_tot, N_ne/n_tot]
830
831     return prob
```

.2 photoionization.py

The code below was used to calculate photoionization and create the plots displayed in section 5.2.

```
1  from functions import *
2  sns.set_theme(context='paper', style='whitegrid', font_scale=1.5, palette='bright', rc={'lines.
       linewidth': 2.5, 'xtick.bottom': True, 'ytick.left' : True})
3
4
5
6
7
8  '''
9  Importing optical constants of materials and interpolate data with insufficient resolution
10 '''
11 #Importing data
12
13 wustite = import_oc('optical constants\wustite_Henning(1995).txt', 1)#wurstite
14 Fe2O3 = import_oc('optical constants\hematite_Triaud_2005.txt', 10) #Hematite
15 Fe3O4 = import_oc('optical constants\magnetite_Triaud_2005.txt', 3) #Magnetite
16 MgFeO = import_oc('optical constants\magwurst_5_5_Henning(1995).txt', 1) #Magnesiowstite
17 pyroxene = import_oc('optical constants\pyroxene6_4_Dorschner(1995).txt', 1) #Pyroxene
18 olivine = import_oc('optical constants\olivine_8_12_Dorschner(1995).txt', 1)#olivine
19
20 #List of refractive indices and name of materials
21 species = ['$FeO$', '$Fe_2O_3$', '$Fe_3O_4$', '$Mg_{0.6}Fe_{0.4}O$', '$Mg_{0.5}Fe_{0.5}SiO_3$', '$Mg_
       {0.8}Fe_{1.2}SiO_4$']
22 data = [wustite, Fe2O3, Fe3O4, MgFeO, pyroxene, olivine]
23
24 #Interpolating materials with lower resolution to allow for integration later
25 wustite_n_interp = interp1d(wustite[:, 0], wustite[:, 1], kind='cubic')
26 MgFeO_n_interp = interp1d(MgFeO[:, 0], MgFeO[:, 1], kind='cubic')
27 olivine_n_interp = interp1d(olivine[:, 0], olivine[:, 1], kind='cubic')
28 pyroxene_n_interp = interp1d(pyroxene[:, 0], pyroxene[:, 1], kind='cubic')
29
30 wustite_k_interp = interp1d(wustite[:, 0], wustite[:, 2], kind='cubic')
31 MgFeO_k_interp = interp1d(MgFeO[:, 0], MgFeO[:, 2], kind='cubic')
32 olivine_k_interp = interp1d(olivine[:, 0], olivine[:, 2], kind='cubic')
33 pyroxene_k_interp = interp1d(pyroxene[:, 0], pyroxene[:, 2], kind='cubic')
34
35 wl_new = np.arange(0.2, 1, 0.01)
36
37 wustite_interp = np.array([wl_new, wustite_n_interp(wl_new), wustite_k_interp(wl_new)]).T
38 MgFeO_interp = np.array([wl_new, MgFeO_n_interp(wl_new), MgFeO_k_interp(wl_new)]).T
39 olivine_interp = np.array([wl_new, olivine_n_interp(wl_new), olivine_k_interp(wl_new)]).T
40 pyroxene_interp = np.array([wl_new, pyroxene_n_interp(wl_new), pyroxene_k_interp(wl_new)]).T
41
42 #list of interpolated refrctive indices
43 data_interp = [wustite_interp, Fe2O3, Fe3O4, MgFeO_interp, pyroxene_interp, olivine_interp]
44
45
46
47
48
49 '''
50 Plotting optical constants
51 '''
52
53 fig, axs = plt.subplots(2, figsize=(7, 8))
54
55 axs[0].plot(wustite[:, 0]*1e3, wustite[:, 1], label='$FeO$')
56 axs[0].plot(Fe2O3[:, 0]*1e3, Fe2O3[:, 1], label='$Fe_2O_3$')
57 axs[0].plot(Fe3O4[:, 0]*1e3, Fe3O4[:, 1], label='$Fe_3O_4$')
58 axs[0].plot(MgFeO[:, 0]*1e3, MgFeO[:, 1], label='$Mg_{0.5}Fe_{0.5}O$')
59 axs[0].plot(olivine[:, 0]*1e3, olivine[:, 1], label='$Mg_{0.8}Fe_{1.2}SiO_4$')
60 axs[0].plot(pyroxene[:, 0]*1e3, pyroxene[:, 1], label='$Mg_{0.6}Fe_{0.4}SiO_3$')
61
```

```
axs[0].set_title('Real part (n)')
axs[0].set_xlim(100, 1000)
axs[0].set_ylim(0.5, 3.5)

axs[1].plot(wustite[:, 0]*1e3, wustite[:, 2], label='$FeO$')
axs[1].plot(Fe2O3[:, 0]*1e3, Fe2O3[:, 2], label='$Fe_2O_3$')
axs[1].plot(Fe3O4[:, 0]*1e3, Fe3O4[:, 2], label='$Fe_3O_4$')
axs[1].plot(MgFeO[:, 0]*1e3, MgFeO[:, 2], label='$Mg_{0.5}Fe_{0.5}O$')
axs[1].plot(olivine[:, 0]*1e3, olivine[:, 2], label='$Mg_{0.8}Fe_{1.2}SiO_4$')
axs[1].plot(pyroxene[:, 0]*1e3, pyroxene[:, 2], label='$Mg_{0.6}Fe_{0.4}SiO_3$')

axs[1].set_title('Imaginary part (k)')
axs[1].set_xlim(100, 1000)
axs[1].set_ylim(-0.25, 1.5)
axs[0].set_ylabel('Refractive index')
axs[1].set_ylabel('Refractive index')
axs[1].set_xlabel('Wavelength (nm)')

labelLines(axs[0].get_lines(), zorder=2.5, fontsize=11)
labelLines(axs[1].get_lines(), zorder=2.5, fontsize=11)

fig.tight_layout()
plt.show()




'''
Calculating and plotting absorption cross section
'''

#Calculating and plotting absorption cross section for 1 nm particle
r_abs= 1e-3 #1 nm in microns
acs = [] #list of cross sections

for i in range(len(data)):
    acs.append(abs_cross(r_abs, data[i]))

#Plotting absorption cross section
for i in range(len(acs)):
    plt.plot(data[i][:, 0]*1e3, acs[i]*1e-8, label=species[i])

plt.title('Absorption cross section $C_{abs}$ for $r=1$ nm')
plt.xlabel('Wavelength (nm)')
plt.ylabel('Cross section ($cm^2$)')
plt.xlim(90, 1000)
plt.ylim(1e-18, 1e-14)
plt.yscale('log')
labelLines(plt.gca().get_lines(), zorder=2.5, fontsize=11)
plt.tight_layout()
plt.show()

#PLotting absorption cross section for range of radii for all materials
fig, axs = plt.subplots(3, 2, figsize=(8, 12))

radii = [0.5e-3, 1e-3, 3e-3, 5e-3, 10e-3]
colors = ["C0", "C1", "C2", "C3", "C4", "C5"]

for i in range(len(species)):
    for j in range(len(radii)):
        abs_cross_section = abs_cross(radii[j], data[i])

        #Generate a color with increasing transparency based on the radius
        color = (0, 0, 1, (j+1)/len(radii))
        alpha= (j+1)/len(radii)

        #Determine the subplot position
        row = i // 2
        col = i % 2

        #Plot the absorption cross section in the corresponding subplot
        axs[row, col].plot(data[i][:, 0]*1e3, abs_cross_section*1e-8, alpha=alpha, color=colors[i],
     label=f'{radii[j]*1e3:.2f} nm')
        axs[row, col].set_title(f'{species[i]}')

for ax in axs.flat:
    ax.set_xlabel('Wavelength (nm)')
    ax.set_ylabel('Cross section ($cm^2$)')
    ax.set_xlim(90, 1000)
    ax.set_ylim(1e-18, 1e-11)
    labelLines(ax.get_lines(), zorder=2.5, fontsize=11)
    ax.set_yscale('log')
```

```
144
145 plt.suptitle('Absorption cross section for various radii')
146 plt.tight_layout()
147 plt.show()
148
149
150
151
152
153 '''
154 Calculating and plotting cut-off wavelength as a function of work function
155 '''
156
157 wf = np.linspace(1.9, 9, 100)
158 cf = []
159 cf_fe2o3 = cut_off(5.5)
160 cf_olivine = cut_off(7.9)
161 cf_msp = cut_off(4)
162 cf_2 = cut_off(2)
163
164 for i in range(len(wf)):
165     cf.append(cut_off(wf[i]))
166
167 plt.plot(np.array(cf)*1e3,wf)
168 plt.title('Cut-off wavelength')
169 plt.axvline(cf_fe2o3*1e3, ymin=0, ymax=0.1, color='r')
170 plt.text(cf_fe2o3*1e3, 2.8, '5.5 eV', color='red', ha='center', va='top')
171 plt.axvline(cf_olivine*1e3, ymin=0, ymax=0.1, color='r')
172 plt.text(cf_olivine*1e3, 2.8, '7.9 eV', color='red', ha='center', va='top')
173 plt.axvline(cf_msp*1e3, ymin=0, ymax=0.1, color='r')
174 plt.text(cf_msp*1e3, 2.8, '4 eV', color='red', ha='center', va='top')
175 plt.axvline(cf_2*1e3, ymin=0, ymax=0.1, color='r')
176 plt.text(cf_2*1e3, 2.8, '2 eV', color='red', ha='center', va='top')
177 plt.ylabel('Work function (eV)')
178 plt.xlabel('Wavelength (nm)')
179 labelLines(plt.gca().get_lines(), zorder=2.5)
180 plt.tight_layout()
181 plt.show()
182
183
184
185
186
187 '''
188 Calculating and plotting photoionization rate
189 '''
190
191 radii = np.linspace(0.5, 10, 100)*1e-3 #0.5-10 nm to microns
192 tc_interp = [] #Nested list of photoionizations
193 work_function = 4.5 #eV
194
195 for i in range(len(data)):
196     rad_tc = []
197     for j in range(len(radii)):
198         rad_tc.append(photoionization(data_interp[i], radii[j], 1, work_function))
199     tc_interp.append(rad_tc)
200
201 plt.figure(figsize=(7, 5))
202 for i in range(len(tc_interp)):
203     plt.plot(radii*1e3, tc_interp[i], label=species[i])
204 plt.title(f'Photoionization rate, WF={work_function} eV')
205 plt.xlabel('MSP radius (nm)')
206 plt.ylabel('Ionization rate (/s)')
207 plt.xlim(0.5, 10)
208 plt.yscale('log')
209 plt.xscale('log')
210 labelLines(plt.gca().get_lines(), zorder=2.5, fontsize=11)
211 plt.show()
212
213
214
215
216
217 '''
218 Calculating and plotting error in matching the wavelength of the refractive index to the wavelength of
        the solar spectrum
219 '''
220
221 error = []
222 for i in range(len(data)):
223     error.append(res(data[i], 4))
224
225 for i in range(len(error)):
```

```
    plt.plot(data[i][:, 0]*1e3, error[i], label=species[i])

plt.title('Wavelength error (solar spectrum wavelength - refractive index wavelength)')
plt.xlabel('Wavelength (nm)')
plt.ylabel('Error (nm)')
plt.xlim(0.1, 1000)
plt.ylim(-0.5, 1)
plt.legend(fontsize="11", loc ="upper right")
plt.tight_layout()
plt.show()




'''
Removing 100-2000 nm from hematite and magnetite and calculating photoionization with respect to radii
'''

#Removing the 100-200 nm range
Fe2O3_cut = import_oc('optical constants\hematite_Triaud_2005.txt', 71)
Fe3O4_cut = import_oc('optical constants\magnetite_Triaud_2005.txt', 34)

wf = 5.5

#Calculate difference for range of radii
Fe2O3_tc_cut = []
Fe3O4_tc_cut = []
Fe2O3_wf1 = []
Fe3O4_wf1 = []

for i in range(len(radii)):
    Fe2O3_tc_cut.append(photoionization(Fe2O3_cut, radii[i], 1, wf))
    Fe3O4_tc_cut.append(photoionization(Fe3O4_cut, radii[i], 1, wf))
    Fe2O3_wf1.append(photoionization(Fe2O3, radii[i], 1, wf))
    Fe3O4_wf1.append(photoionization(Fe3O4, radii[i], 1, wf))

Fe2O3_diff = np.abs(np.array(Fe2O3_wf1) - np.array(Fe2O3_tc_cut))
Fe3O4_diff = np.abs(np.array(Fe3O4_wf1) - np.array(Fe3O4_tc_cut))

#Plotting the original and cut photoionization
plt.plot(radii*1e3, Fe2O3_wf1, label=species[1], color='b')
plt.plot(radii*1e3, Fe2O3_tc_cut, label='$Fe_2O_3$ removed', linestyle='--', color='b')
plt.plot(radii*1e3, Fe3O4_wf1, label=species[2], color='g')
plt.plot(radii*1e3, Fe3O4_tc_cut, label='$Fe_3O_4$ removed', linestyle='--', color='g')
plt.title('Photoionization')
plt.xlabel('Radius (nm)')
plt.ylabel('Ionization rate (/s)')
plt.yscale('log')
plt.xscale('log')
plt.legend(fontsize="11")
plt.show()

#Plotting the percentage difference
plt.plot(radii*1e3, Fe2O3_diff/(Fe2O3_wf1), label='$Fe_2O_3$', color='b')
plt.plot(radii*1e3, Fe3O4_diff/(Fe3O4_wf1), label='$Fe_3O_4$', color='g')
plt.title('Difference in photoionization')
plt.xlabel('Radius (nm)')
plt.ylabel('Percentage difference (%)')
plt.gca().set_yticklabels([f'{x:.0%}' for x in plt.gca().get_yticks()])
plt.xscale('log')
labelLines(plt.gca().get_lines(), zorder=2.5, fontsize=11)
plt.show()




'''
Removing 100-200 nm and calculating photoionization with respect to work function
'''

wf_range = np.linspace(1.5, 5.5, 10)

#Removing the 100-200 nm range
Fe2O3_cut1 = import_oc('optical constants\hematite_Triaud_2005.txt', 71)
Fe3O4_cut1 = import_oc('optical constants\magnetite_Triaud_2005.txt', 34)

Fe2O3_cut = []
Fe3O4_cut = []
Fe2O3_wf = []
Fe3O4_wf = []

```

```
for i in range(len(wf_range)):
    Fe2O3_cut.append(photoionization(Fe2O3_cut1, 1e-3, 1, wf_range[i]))
    Fe3O4_cut.append(photoionization(Fe3O4_cut1, 1e-3, 1, wf_range[i]))
    Fe2O3_wf.append(photoionization(Fe2O3, 1e-3, 1, wf_range[i]))
    Fe3O4_wf.append(photoionization(Fe3O4, 1e-3, 1, wf_range[i]))

Fe2O3_diff = np.abs(np.array(Fe2O3_wf) - np.array(Fe2O3_cut))
Fe3O4_diff = np.abs(np.array(Fe3O4_wf) - np.array(Fe3O4_cut))

#Plotting the original and cut photoionization
plt.plot(wf_range, Fe2O3_wf, label=species[1], color='b')
plt.plot(wf_range, Fe2O3_cut, label='$Fe_2O_3$ removed', linestyle='--', color='b')
plt.plot(wf_range, Fe3O4_wf, label=species[2], color='g')
plt.plot(wf_range, Fe3O4_cut, label='$Fe_3O_4$ removed', linestyle='--', color='g')
plt.title('Photoionization')
plt.xlabel('Work function (eV)')
plt.ylabel('Ionization rate (/s)')
plt.yscale('log')
plt.legend(fontsize='11')
plt.show()

#Plotting the percentage difference
plt.plot(wf_range, Fe2O3_diff/(Fe2O3_wf), label='$Fe_2O_3$', color='b')
plt.plot(wf_range, Fe3O4_diff/(Fe3O4_wf), label='$Fe_3O_4$', color='g')
plt.title('Difference in photoionization')
plt.xlabel('Work function (eV)')
plt.ylabel('Percentage difference (%)')
plt.gca().set_yticklabels([f'{x:.0%}' for x in plt.gca().get_yticks()])
labelLines(plt.gca().get_lines(), zorder=2.5, fontsize=11)
plt.show()




'''
Calculating and plotting photoionization for different work function for all the materials.
'''


fig, axs = plt.subplots(3, 2, figsize=(8, 12))

workf = [1.5, 2.5, 3.5, 4.5, 5.5]

for i in range(len(species)):
    for j in range(len(workf)):
        photoion = []
        for k in range(len(radii)):
            photoion.append(photoionization(data_interp[i], radii[k], 1, workf[j]))

        alpha= (j+1)/len(workf)
        row = i // 2
        col = i % 2

        axs[row, col].plot(radii*1e3, photoion, alpha=alpha, color=colors[i], label=f'{workf[j]} eV')
        axs[row, col].set_title(f'{species[i]}')

for ax in axs.flat:
    ax.set_xlabel('Wavelength (nm)')
    ax.set_ylabel('Ionization rate (/s)')
    labelLines(ax.get_lines(), zorder=2.5, fontsize=11)
    ax.set_yscale('log')
    ax.set_xscale('log')

plt.suptitle('Photoionization rate with varying work function and interpolation')
plt.tight_layout()
plt.show()




'''
Calculating and plotting the photoionization for different quantum yields
'''

qy = np.linspace(0.2, 1, 5) #range of Y
work_function = 4.5 #eV

radii = np.array(radii)

fig, axs = plt.subplots(3, 2, figsize=(8, 12))

```

```
for i in range(len(data)):
    for j in range(len(qy)):
        photoion = []
        for k in range(len(radii)):
            photoion.append(photoionization(data[i], radii[k], qy[j], work_function))

        alpha= (j+1)/len(qy)
        row = i // 2
        col = i % 2

        axs[row, col].plot(radii*1e3, photoion, alpha=alpha, color=colors[i], label=f'{qy[j]:.0%}')
        axs[row, col].set_title(f'{species[i]}')

for ax in axs.flat:
    ax.set_xlabel('Wavelength (nm)')
    ax.set_ylabel('Photoionization (/s)')
    labelLines(ax.get_lines(), zorder=2.5, fontsize=11)
    ax.set_yscale('log')
    ax.set_xscale('log')

plt.suptitle('Photoionization rate with varying quantum yield Y')
plt.tight_layout()
plt.show()




'''
Arteficially creating stronger solar spectrums to check the effect of the solar spectrum on the
    photoionization
'''

#Importing solar spectrum
solar_new = np.genfromtxt('lasp_gsfc_solarspectrum.txt', delimiter='\t')

factors = [0.01, 0.1, 1, 3, 5, 10, 100]
Y = 1
E= 4.5
solar_list = []

#Increasing solar irradiance by the list of factors
for i in range(len(factors)):
    solar_irr = solar_new[:, 1]*factors[i]*1e-3 #mW/m^2/nm to W/m^2/nm
    solar = np.array([solar_new[:, 0],  solar_irr]).T
    solar_list.append(solar)

fig, axs = plt.subplots(3, 2, figsize=(8, 12))

for i in range(len(data)):
    for j in range(len(factors)):
        photoionization = []
        for r in radii:
            photoionization.append(photoionization_solar(data[i], r, Y, E, solar_list[j]))

        alpha= (j+1)/len(factors)
        row = i // 2
        col = i % 2

        axs[row, col].plot(radii*1e3, photoionization, alpha=alpha, color=colors[i], label=f'{factors[
    j]}')
        axs[row, col].set_title(f'{species[i]}')

for ax in axs.flat:
    ax.set_xlabel('Wavelength (nm)')
    ax.set_ylabel('Ionization rate (/s)')
    labelLines(ax.get_lines(), zorder=2.5, fontsize=11)
    ax.set_yscale('log')
    ax.set_xscale('log')

plt.suptitle('Photoionization with varying solar photon flux')
plt.tight_layout()
plt.show()
```

.3 charge_state.py

The code below was used to calculate and generate the figures of the charging rates, number densities and charge probabilities of MSP presented in chapter 6.

```
from functions import *
import matplotlib.gridspec as gridspec

sns.set_theme(context='paper', style='whitegrid', font_scale=1.5, palette='bright', rc={'lines.
    linewidth': 2.5, 'xtick.bottom': True, 'ytick.left' : True})




'''Importing optical constants of materials and interpolate data with insufficient resolution'''

#Importing data
wustite = import_oc('optical constants\wustite_Henning(1995).txt', 1)#wurstite
Fe2O3 = import_oc('optical constants\hematite_Triaud_2005.txt', 10) #Hematite
Fe3O4 = import_oc('optical constants\magnetite_Triaud_2005.txt', 3) #Magnetite
MgFeO = import_oc('optical constants\magwurst_5_5_Henning(1995).txt', 1) #Magnesiowstite
pyroxene = import_oc('optical constants\pyroxene6_4_Dorschner(1995).txt', 1) #Pyroxene
olivine = import_oc('optical constants\olivine_8_12_Dorschner(1995).txt', 1)#olivine

species = ['$FeO$', '$Fe_2O_3$', '$Fe_3O_4$', '$Mg_{0.5}Fe_{0.5}O$', '$Mg_{0.6}Fe_{0.4}SiO_3$', '$Mg_
    {0.8}Fe_{1.2}SiO_4$']
data = [wustite, Fe2O3, Fe3O4, MgFeO, pyroxene, olivine]

# Interpolatng materials with lower resolution to allow for integration later
wustite_n_interp = interp1d(wustite[:, 0], wustite[:, 1], kind='cubic')
MgFeO_n_interp = interp1d(MgFeO[:, 0], MgFeO[:, 1], kind='cubic')
olivine_n_interp = interp1d(olivine[:, 0], olivine[:, 1], kind='cubic')
pyroxene_n_interp = interp1d(pyroxene[:, 0], pyroxene[:, 1], kind='cubic')

wustite_k_interp = interp1d(wustite[:, 0], wustite[:, 2], kind='cubic')
MgFeO_k_interp = interp1d(MgFeO[:, 0], MgFeO[:, 2], kind='cubic')
olivine_k_interp = interp1d(olivine[:, 0], olivine[:, 2], kind='cubic')
pyroxene_k_interp = interp1d(pyroxene[:, 0], pyroxene[:, 2], kind='cubic')

wl_new = np.arange(0.2, 1, 0.001)

wustite_interp = np.array([wl_new, wustite_n_interp(wl_new), wustite_k_interp(wl_new)]).T
MgFeO_interp = np.array([wl_new, MgFeO_n_interp(wl_new), MgFeO_k_interp(wl_new)]).T
olivine_interp = np.array([wl_new, olivine_n_interp(wl_new), olivine_k_interp(wl_new)]).T
pyroxene_interp = np.array([wl_new, pyroxene_n_interp(wl_new), pyroxene_k_interp(wl_new)]).T

data_interp = [wustite_interp, Fe2O3, Fe3O4, MgFeO_interp, pyroxene_interp, olivine_interp]




'''
Calculating integration time for ODE including and excluding photoionization (hematite) for 1 nm sized
    MSP
'''

Q = 10 #ionization rate (/cm^3/s)
N_tot = 10000 #total number of particles
n_e0 =  3162 #np.sqrt(Q/1e-6) # initial electron and ion densities (/cm^3)
r = 0.8 #nm

#Time
t = np.arange(0, 3000, 1)
label_list=['$N_e$', '$N_i$', '$N_{1}$', '$N_0$', '$N_{-1}$']

ode_1 = odeint(odes_photoion, [n_e0, n_e0, 0, N_tot, 0], t, args=(Q, r, Fe2O3, 1, 5.5, 2.5))
for i in range(len(ode_1[0])):
    plt.plot(t, ode_1[:, i], label=label_list[i])
plt.title('ODE integration for MSP with photoionization')
plt.yscale('log')
plt.legend( loc='lower right')
plt.show()

ode_2 = odeint(odes, [n_e0, n_e0, 0, N_tot, 0], t, args=(Q, r))
for i in range(len(ode_2[0])):
```

```
    plt.plot(t, ode_2[:, i], label=label_list[i])
plt.title('ODE integration for MSP without photoionization')
plt.yscale('log')
plt.legend( loc='lower right')
plt.show()




'''
Calculating number desities of all species with no and full photodetachment using the variables above
'''

test= ion_den(10, 3000, 0.8, 1e4, material=Fe2O3, WF=5.5, EA=2.5) #Hematite
test2= ion_den(10, 3000, 0.8, 1e4, photoion=False) #No photoionization
test3= ion_den(10, 3000, 0.8, 1e4, material=olivine_interp, WF=5.5, EA=2.5) #Olivine

labels = ['N_e', 'N_i', 'N_po', 'N_0', 'N_ne']

'''
Plotting number densities of electrons, ions, positive, neutral and negative MSP
'''

fig = plt.figure(figsize=(12,9), constrained_layout=True)
gs = fig.add_gridspec(2, 5)

ax1 = fig.add_subplot(gs[0, 0:2])
for i in range(5):
    ax1.plot(test['ion'], test[labels[i]], label=label_list[i])
ax1.set_xlabel('Ionization rate (/$cm^3$/$s$)')
ax1.set_ylabel('Number density (/$cm^3$)')
ax1.set_xscale('log')
ax1.set_yscale('log')
ax1.set_ylim(1e-1, 1e6)
ax1.legend(loc='lower right')
plt.figtext(0.08, 0.89, 'Hematite \n$WF=5.5\;eV,\; EA=2.5\;eV$  \n$n_{MSP}=1\cdot 10^4 \;cm^{-3}$ \n$r
      =0.8\;nm$', fontsize=11)

ax2 = fig.add_subplot(gs[0, 2:4])
for i in range(5):
    ax2.plot(test3['ion'], test3[labels[i]], label=label_list[i])
ax2.set_xlabel('Ionization rate (/$cm^3$/$s$)')
ax2.set_ylabel('Number density (/$cm^3$)')
ax2.set_xscale('log')
ax2.set_yscale('log')
ax2.set_ylim(1e-1, 1e6)
ax2.legend( loc='lower right')
plt.figtext(0.52, 0.89, 'Olivine \n$WF=5.5\;eV,\; EA=2.5\;eV$ \n$n_{MSP}=1\cdot 10^4 \;cm^{-3}$ \n$r
     =0.8\;nm$', fontsize=11)

ax3 = fig.add_subplot(gs[1, 1:3])
for i in range(5):
    ax3.plot(test2['ion'], test2[labels[i]], label=label_list[i])
ax3.set_xlabel('Ionization rate (/$cm^3$/$s$)')
ax3.set_ylabel('Number density (/$cm^3$)')
ax3.set_xscale('log')
ax3.set_yscale('log')
ax3.legend( loc='lower right')
ax3.set_ylim(1e-1, 1e6)

plt.figtext(0.29, 0.41, 'No photoion. & detach. \n$n_{MSP}=1\cdot 10^4 \;cm^{-3}$ \n$r =0.8\;nm$',
     fontsize=11)
fig.suptitle('Number density of MSP and plasma particles', x=0.45, y=1.03)
plt.show()

'''
Plotting all materials
'''

fig, axs = plt.subplots(3, 2, figsize=(10, 15))

colors = ["C0", "C1", "C2", "C3", "C4", "C5"]

for i in range(len(data_interp)):
    material_den = ion_den(10, 3000, 0.8, 1e4, material=data_interp[i], WF=5.5, EA=2.5)

    #determine the subplot position
    row = i // 2
    col = i % 2

    for j in range(5):
        axs[row, col].plot(material_den['ion'], material_den[labels[j]], label=label_list[j])
```

```
150
151     axs[row, col].set_title(f'{species[i]}')
152
153 for ax in axs.flat:
154     ax.set_xlabel('Ionization rate (/$cm^3$/$s$)')
155     ax.set_ylabel('Number density (/$cm^3$)')
156     ax.set_xscale('log')
157     ax.set_yscale('log')
158     ax.legend( loc='lower right')
159     ax.set_ylim(1e-1, 1e6)
160
161 plt.suptitle('Number density of MSP and plasma particles')
162 plt.tight_layout()
163 plt.show()
164
165
166
167
168
169 '''
170 Calculating desities of all species with no and full photodetachment
171 range of radii 0.5-10 nm, n_tot = 1e4cm^3, Q=10, ne=ni=3162cm^3
172 '''
173
174 #Without photoionization
175 rad_test1= rad_den(10, 4000, 10, 1e4, photoion=False) #without photoionization
176 rad_test2= rad_den(10, 4000, 10, 1e4, material=Fe2O3, WF=5.5, EA=2.5) #hematite
177 rad_test3= rad_den(10, 4000, 10, 1e4, material=olivine_interp, WF=5.5, EA=2.5) #olivine
178
179 '''
180 Plotting number densities of electrons, ions, positive, neutral and negative MSP
181 '''
182
183 fig = plt.figure(figsize=(12,9), constrained_layout=True)
184 gs = fig.add_gridspec(2, 5)
185
186 ax1 = fig.add_subplot(gs[0, 0:2])
187 for i in range(5):
188     ax1.plot(rad_test2['r'], rad_test2[labels[i]], label=label_list[i])
189 ax1.set_xlabel('MSP radius ($nm$)')
190 ax1.set_ylabel('Number density (/$cm^3$)')
191 #ax1.set_xscale('log')
192 ax1.set_yscale('log')
193 ax1.set_ylim(1e-1, 1e6)
194 ax1.legend(loc='lower right')
195 plt.figtext(0.08, 0.89, 'Hematite \n$WF=5.5\;eV,\; EA=2.5\;eV$ \n$Q=10\; /cm^3/s$  \n$n_{MSP}=1\cdot
      10^4 \;cm^{-3}$', fontsize=11)
196
197 ax2 = fig.add_subplot(gs[0, 2:4])
198 for i in range(5):
199     ax2.plot(rad_test3['r'], rad_test3[labels[i]], label=label_list[i])
200 ax2.set_xlabel('MSP radius ($nm$)')
201 ax2.set_ylabel('Number density (/$cm^3$)')
202 #ax2.set_xscale('log')
203 ax2.set_yscale('log')
204 ax2.set_ylim(1e-1, 1e6)
205 ax2.legend( loc='lower right')
206 plt.figtext(0.52, 0.89, 'Olivine \n$WF=5.5\;eV,\; EA=2.5\;eV$ \n$Q=10\; /cm^3/s$ \n$n_{MSP}=1\cdot
      10^4 \;cm^{-3}$', fontsize=11)
207
208 ax3 = fig.add_subplot(gs[1, 1:3])
209 for i in range(5):
210     ax3.plot(rad_test1['r'], rad_test1[labels[i]], label=label_list[i])
211 ax3.set_xlabel('MSP radius ($nm$)')
212 ax3.set_ylabel('Number density (/$cm^3$)')
213 #ax3.set_xscale('log')
214 ax3.set_yscale('log')
215 ax3.legend( loc='lower right')
216 ax3.set_ylim(1e-1, 1e6)
217
218 plt.figtext(0.29, 0.41, 'No photoion. & detach. \n$Q=10\; /cm^3/s$ \n$n_{MSP}=1\cdot 10^4 \;cm^{-3}$',
       fontsize=11)
219 fig.suptitle('Number density of MSP and plasma particles', x=0.45, y=1.03)
220 plt.show()
221
222 '''
223 Plotting all materials
224 '''
225
226 fig, axs = plt.subplots(3, 2, figsize=(10, 15))
227
228 for i in range(len(data_interp)):
229
```

```
    material_den = rad_den(10, 4000, 10, 1e4, material=data_interp[i], WF=5.5, EA=2.5)

    #determine the subplot position
    row = i // 2
    col = i % 2

    for j in range(5):
        axs[row, col].plot(material_den['r'], material_den[labels[j]], label=label_list[j])

    axs[row, col].set_title(f'{species[i]}')

for ax in axs.flat:
    ax.set_xlabel('MSP radius ($nm$)')
    ax.set_ylabel('Number density (/$cm^3$)')
    ax.set_yscale('log')
    ax.legend( loc='lower right')
    ax.set_ylim(1e-1, 5e4)

plt.suptitle('Number density of MSP and plasma particles')
plt.tight_layout()
plt.show()




'''
Calculating number desities of all species with no and full photodetachment
r= 0.8 nm, n_tot = 1e2-1e5cm^3, Q=10, ne=ni=3162cm^3
'''

tot_test1= tot_den(10, 4000, 10, 0.8, photoion=False) #no photoionization
tot_test2= tot_den(10, 4000, 10, 0.8, material=Fe2O3, WF=5.5, EA=2.5) #hematite
tot_test3= tot_den(10, 4000, 10, 0.8, material=olivine_interp, WF=5.5, EA=2.5) #olivine

'''
Plotting number densities of electrons, ions, positive, neutral and negative MSP
'''

fig = plt.figure(figsize=(12,9), constrained_layout=True)
gs = fig.add_gridspec(2, 5)

ax1 = fig.add_subplot(gs[0, 0:2])
for i in range(5):
    ax1.plot(tot_test2['tot'], tot_test2[labels[i]], label=label_list[i])
ax1.set_xlabel('Total MSP number density (/$cm^3$)')
ax1.set_ylabel('Number density (/$cm^3$)')
ax1.set_xscale('log')
ax1.set_yscale('log')
ax1.set_ylim(1e-2, 1e6)
ax1.legend(loc='lower right')
plt.figtext(0.08, 0.89, 'Hematite\n$WF=5.5\;eV,\; EA=2.5\;eV$ \n$Q=10\; /cm^3/s$  \n$r = 0.8 \;nm$',
     fontsize=11)

ax2 = fig.add_subplot(gs[0, 2:4])
for i in range(5):
    ax2.plot(tot_test3['tot'], tot_test3[labels[i]], label=label_list[i])
ax2.set_xlabel('Total MSP number density (/$cm^3$)')
ax2.set_ylabel('Number density (/$cm^3$)')
ax2.set_xscale('log')
ax2.set_yscale('log')
ax2.set_ylim(1e-2, 1e6)
ax2.legend( loc='lower right')
plt.figtext(0.52, 0.89, 'Olivine\n$WF=5.5\;eV,\; EA=2.5\;eV$ \n$Q=10\; /cm^3/s$ \n$r = 0.8 \;nm$',
     fontsize=11)

ax3 = fig.add_subplot(gs[1, 1:3])
for i in range(5):
    ax3.plot(tot_test1['tot'], tot_test1[labels[i]], label=label_list[i])
ax3.set_xlabel('Total MSP number density (/$cm^3$)')
ax3.set_ylabel('Number density (/$cm^3$)')
ax3.set_xscale('log')
ax3.set_yscale('log')
ax3.legend( loc='lower right')
ax3.set_ylim(1e-3, 1e6)

plt.figtext(0.29, 0.41, 'No photoion. & detach. \n$Q=10\; /cm^3/s$ \n$r = 0.8 \;nm$', fontsize=11)
fig.suptitle('Number density of MSP and plasma particles', x=0.45, y=1.03)
plt.show()

'''
Plotting all materials
'''
```

```

fig, axs = plt.subplots(3, 2, figsize=(10, 15))

for i in range(len(data_interp)):

    material_den = tot_den(10, 4000, 10, 0.8, material=data_interp[i], WF=5.5, EA=2.5)

    #determine the subplot position
    row = i // 2
    col = i % 2

    for j in range(5):
        axs[row, col].plot(material_den['tot'], material_den[labels[j]], label=label_list[j])

    axs[row, col].set_title(f'{species[i]}')

for ax in axs.flat:
    ax.set_xlabel('Total MSP number density (/$cm^3$)')
    ax.set_ylabel('Number density (/$cm^3$)')
    ax.set_yscale('log')
    ax.set_xscale('log')
    ax.legend(loc='lower right')
    ax.set_ylim(1e-1, 1e6)

plt.suptitle('Number density of MSP and plasma particles')
plt.tight_layout()
plt.show()




'''
Calculating and plotting charging rates for a range of MSP radii
'''

radii = np.linspace(0.5, 3, 20)

charging_rate = charging_rates(radii, 3162, 1e4, 10, photo_ion=True, material=Fe2O3, Y=1, WF=5.5, EA
    =2.5) #hematite
olivine_rate = charging_rates(radii, 3162, 1e4, 10, photo_ion=True, material=olivine_interp, Y=1, WF
    =5.5, EA=2.5) #olivine
noion_rate = charging_rates(radii, 3162, 1e4, 10) #no photoionization

'''
Plotting charging rates
'''

fig = plt.figure(figsize=(12,9), constrained_layout=True)
gs = fig.add_gridspec(2, 5)

ax1 = fig.add_subplot(gs[0, 0:2])
ax1.plot(charging_rate[0], charging_rate[2], label=r'$\alpha_{0}^{e}N_{0}$') #e+Z_0
ax1.plot(charging_rate[0], charging_rate[6], label=r'$\alpha_{-1}^{i}N_{-1}$') #ion + Z-1
ax1.plot(charging_rate[0], charging_rate[1], label=r'$\alpha_{1}^{e}N_{1}$') #e+Z_1
ax1.plot(charging_rate[0], charging_rate[5], label=r'$\alpha_{0}^{i}N_{0}$') #ion + Z0
ax1.plot(charging_rate[0], charging_rate[7], label=r'$\beta_{ion}$') #photoionization
ax1.plot(charging_rate[0], charging_rate[8], label=r'$\beta_{det}$') #photodetachment
ax1.set_yscale('log')
ax1.set_xlabel('Particle radius (nm)')
ax1.set_ylabel('Charging rates ($/s$)')
plt.figtext(0.08, 0.94, 'Hematite \n$WF=5.5\;eV,\; EA=2.5\;eV$', fontsize=11)
labelLines(ax1.get_lines(), zorder=2.5)
ax1.set_ylim(1e-2, 1e3)

ax2 = fig.add_subplot(gs[0, 2:4])
ax2.plot(olivine_rate[0], olivine_rate[2], label=r'$\alpha_{0}^{e}N_{0}$') #e+Z_0
ax2.plot(olivine_rate[0], olivine_rate[6], label=r'$\alpha_{-1}^{i}N_{-1}$') #ion + Z-1
ax2.plot(olivine_rate[0], olivine_rate[1], label=r'$\alpha_{1}^{e}N_{1}$') #e+Z_1
ax2.plot(olivine_rate[0], olivine_rate[5], label=r'$\alpha_{0}^{i}N_{0}$') #ion + Z0
ax2.plot(olivine_rate[0], olivine_rate[7], label=r'$\beta_{ion}$') #photoionization
ax2.plot(olivine_rate[0], olivine_rate[8], label=r'$\beta_{det}$') #photodetachment
ax2.set_yscale('log')
ax2.set_xlabel('Particle radius (nm)')
ax2.set_ylabel('Charging rates ($/s$)')
plt.figtext(0.51, 0.94, 'Olivine \n$WF=5.5\;eV,\; EA=2.5\;eV$', fontsize=11)
labelLines(ax2.get_lines(), zorder=2.5)
ax2.set_ylim(1e-2, 1e3)

ax3 = fig.add_subplot(gs[1, 1:3])
ax3.plot(noion_rate[0], noion_rate[2], label=r'$\alpha_{0}^{e}N_{0}$') #e+Z_0
ax3.plot(noion_rate[0], noion_rate[6], label=r'$\alpha_{-1}^{i}N_{-1}$') #ion + Z-1
ax3.plot(noion_rate[0], noion_rate[1], label=r'$\alpha_{1}^{e}N_{1}$') #e+Z_1
```

```
ax3.plot(noion_rate[0],noion_rate[5], label=r'$\alpha_{0}^{i}N_{0}$') #ion + Z0
ax3.set_yscale('log')
ax3.set_xlabel('Particle radius (nm)')
ax3.set_ylabel('Charging rates ($/s$)')
plt.figtext(0.29, 0.46, 'No photoion. or detach.', fontsize=11)
labelLines(ax3.get_lines(), zorder=2.5)
ax3.set_ylim(1e-2, 1e3)

fig.suptitle('Charging rates of MSP', x=0.45, y=1.03)
plt.show()

'''
Calculating and plotting charging rates for all materials
'''
fig, axs = plt.subplots(3, 2, figsize=(10, 15))

for i in range(len(data_interp)):

    material_rate = charging_rates(radii, 3162, 1e4, 10, photo_ion=True, material=data_interp[i], Y=1,
      WF=5.5, EA=2.5)

    #determine the subplot position
    row = i // 2
    col = i % 2

    axs[row, col].plot(material_rate[0], material_rate[2], label=r'$\alpha_{0}^{e}N_{0}$') #e+Z_0
    axs[row, col].plot(material_rate[0], material_rate[6], label=r'$\alpha_{-1}^{i}N_{-1}$') #ion + Z
     -1
    axs[row, col].plot(material_rate[0], material_rate[1], label=r'$\alpha_{1}^{e}N_{1}$') #e+Z_1
    axs[row, col].plot(material_rate[0], material_rate[5], label=r'$\alpha_{0}^{i}N_{0}$') #ion + Z0
    axs[row, col].plot(material_rate[0], material_rate[7], label=r'$\beta_{ion}$') #photoionization
    axs[row, col].plot(material_rate[0], material_rate[8], label=r'$\beta_{det}$') #photodetachment

    axs[row, col].set_title(f'{species[i]}')

for ax in axs.flat:
    ax.set_yscale('log')
    ax.set_xlabel('Particle radius (nm)')
    ax.set_ylabel('Charging rates ($/s$)')
    labelLines(ax.get_lines(), zorder=2.5)
    ax.set_ylim(1e-2, 1e3)

plt.suptitle('Charging rates of MSP')
plt.tight_layout()
plt.show()




'''
Calculating charge probability for high density case
'''

Z = [1, 0, -1] #for calcualtions and plotting
radii = [0.5, 0.8, 1, 5, 10] #nm
markers = ['o','v','s','p','P','*','h','H','+','x','X','D','d','|','_'] #for plotting

test_hem = [] #hematite
test_oli = [] #olivine
test_non = [] #no photoionization

for i in range(len(radii)): #calcualtes charge probability for given radii
    test_hem.append(charge_prob(3162, 1e4, 3000, 10, radii[i], photo_ion=True, material=Fe2O3, WF=5.5,
      EA=2.5))
    test_oli.append(charge_prob(3162, 1e4, 3000, 10, radii[i], photo_ion=True, material=olivine_interp
     , WF=5.5, EA=2.5))
    test_non.append(charge_prob(3162, 1e4, 3000, 10, radii[i], photo_ion=False))

'''
Plotting charge probabilities
'''

fig = plt.figure(figsize=(12,9), constrained_layout=True)
gs = fig.add_gridspec(2, 5)

ax1 = fig.add_subplot(gs[0, 0:2])
for i in range(len(radii)):
    ax1.plot(Z, test_hem[i], label=f'{radii[i]} nm', linewidth=1.5, linestyle= '--', marker=markers[i
     ])
ax1.set_xlabel('Charge number Z')
ax1.set_ylabel('Probability')
ax1.legend(loc='lower left')
```

```
plt.figtext(0.06, 0.94, 'Hematite \n$WF=5.5\;eV,\; EA=2.5\;eV$', fontsize=12)
ax1.set_xlim(-3, 2)
ax1.set_ylim(0, 1)

ax2 = fig.add_subplot(gs[0, 2:4])
for i in range(len(radii)):
    ax2.plot(Z, test_oli[i], linestyle= '--', marker=markers[i], label=f'{radii[i]} nm', linewidth=1.5)
ax2.set_xlabel('Charge number Z')
ax2.set_ylabel('Probability')
ax2.legend(loc='lower left')
plt.figtext(0.49, 0.94, 'Olivine\n$WF=5.5\;eV,\; EA=2.5\;eV$', fontsize=12)
ax2.set_xlim(-3, 2)
ax2.set_ylim(0, 1)

ax3 = fig.add_subplot(gs[1, 1:3])
for i in range(len(radii)):
    ax3.plot(Z, test_non[i], linestyle= '--', marker=markers[i], label=f'{radii[i]} nm', linewidth=1.5)
ax3.set_xlabel('Charge number Z')
ax3.set_ylabel('Probability')
ax3.legend(loc='lower left')
plt.figtext(0.27, 0.45, 'No photoion. & detach.\n', fontsize=12)
ax3.set_xlim(-3, 2)
ax3.set_ylim(0, 1)

fig.suptitle('Charge probability of MSP, high density case', x=0.45, y=1.03)
plt.show()

'''
Calculating and plotting charge probability using the high density case for all materials
'''

fig, axs = plt.subplots(3, 2, figsize=(10, 15))

for i in range(len(data)):

    for j in range(len(radii)):
        prob = (charge_prob(3162, 1e4, 3000, 10, radii[j], photo_ion=True, material=data_interp[i], WF=5.5, EA=2.5))

        row = i // 2
        col = i % 2

        axs[row, col].plot(Z, prob, label=f'{radii[j]} nm', linewidth=1.5, linestyle= '--', marker=markers[j])
    axs[row, col].set_title(f'{species[i]}')

for ax in axs.flat:
    ax.set_xlabel('Charge number Z')
    ax.set_ylabel('Probability')
    ax.legend(loc='upper left')
    ax.set_xlim(-3, 2)
    ax.set_ylim(0, 1)

plt.suptitle('Charge probability of MSP, high density case')
plt.tight_layout()
plt.show()




'''
Calculating charge probability for low density case
'''

test_hem = [] #hematite
test_oli = [] #olivine
test_non = [] #no photoionization

for i in range(len(radii)):
    test_hem.append(charge_prob_lowden(3162, 1, 3000, 10, radii[i], photo_ion=True, material=Fe2O3, WF=5.5, EA=2))
    test_oli.append(charge_prob_lowden(3162, 1, 3000, 10, radii[i], photo_ion=True, material=olivine_interp, WF=5.5, EA=2))
    test_non.append(charge_prob_lowden(3162, 1, 3000, 10, radii[i], photo_ion=False))

'''
Plotting charge probabilities
'''

fig = plt.figure(figsize=(12,9), constrained_layout=True)
```

```
gs = fig.add_gridspec(2, 5)

ax1 = fig.add_subplot(gs[0, 0:2])
for i in range(len(radii)):
    ax1.plot(Z, test_hem[i], label=f'{radii[i]} nm', linewidth=1.5, linestyle= '--', marker=markers[i])
ax1.set_xlabel('Charge number Z')
ax1.set_ylabel('Probability')
ax1.legend(loc='lower left')
plt.figtext(0.06, 0.94, 'Hematite \n$WF=5.5\;eV,\; EA=2.5\;eV$', fontsize=12)
ax1.set_xlim(-3, 2)
ax1.set_ylim(0, 1)

ax2 = fig.add_subplot(gs[0, 2:4])
for i in range(len(radii)):
    ax2.plot(Z, test_oli[i], linestyle= '--', marker=markers[i], label=f'{radii[i]} nm', linewidth=1.5)
ax2.set_xlabel('Charge number Z')
ax2.set_ylabel('Probability')
ax2.legend(loc='lower left')
plt.figtext(0.49, 0.94, 'Olivine\n$WF=5.5\;eV,\; EA=2.5\;eV$', fontsize=12)
ax2.set_xlim(-3, 2)
ax2.set_ylim(0, 1)

ax3 = fig.add_subplot(gs[1, 1:3])
for i in range(len(radii)):
    ax3.plot(Z, test_non[i], linestyle= '--', marker=markers[i], label=f'{radii[i]} nm', linewidth=1.5)
ax3.set_xlabel('Charge number Z')
ax3.set_ylabel('Probability')
ax3.legend(loc='lower left')
plt.figtext(0.27, 0.45, 'No photoion. & detach.\n', fontsize=12)
ax3.set_xlim(-3, 2)
ax3.set_ylim(0, 1)

fig.suptitle('Charge probability of MSP, low density case', x=0.45, y=1.03)
plt.show()


'''
Calcualting and plotting charge probabilities using the low density model for all species
'''

fig, axs = plt.subplots(3, 2, figsize=(10, 15))

for i in range(len(data)):

    for j in range(len(radii)):
        prob = (charge_prob_lowden(3162, 1e4, 1000, 10, radii[j], photo_ion=True, material=data_interp[i], WF=5.5, EA=2.5))

        #determine the subplot position
        row = i // 2
        col = i % 2

        axs[row, col].plot(Z, prob, label=f'{radii[j]} nm', linewidth=1.5, linestyle= '--', marker=markers[j])
    axs[row, col].set_title(f'{species[i]}')

for ax in axs.flat:
    ax.set_xlabel('Charge number Z')
    ax.set_ylabel('Probability')
    ax.legend(loc='upper left')
    ax.set_xlim(-3, 2)
    ax.set_ylim(0, 1)

plt.suptitle('Charge probability of MSP, low density case')
plt.tight_layout()
plt.show()
```