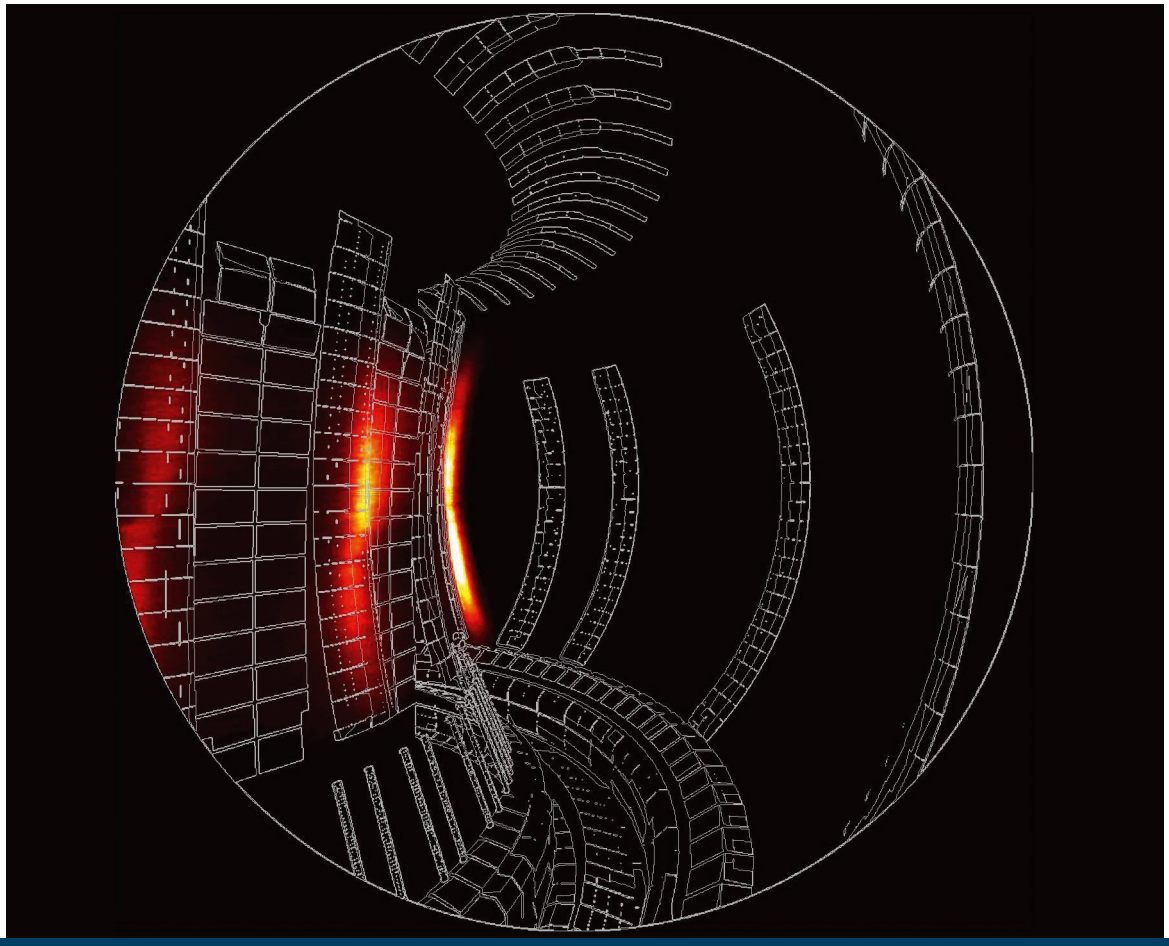


3D simulation of impurity transport in a fusion edge plasma using a massively parallel Monte-Carlo code

Juri Romazanov

Energie & Umwelt / Energy & Environment
Band / Volume 448
ISBN 978-3-95806-377-8

Mitglied der Helmholtz-Gemeinschaft

JÜLICH
Forschungszentrum

Schriften des Forschungszentrums Jülich
Reihe Energie & Umwelt / Energy & Environment Band / Volume 448

Forschungszentrum Jülich GmbH
Institut für Energie- und Klimaforschung
Plasmaphysik (IEK-4)

3D simulation of impurity transport in a fusion edge plasma using a massively parallel Monte-Carlo code

Juri Romazanov

Schriften des Forschungszentrums Jülich
Reihe Energie & Umwelt / Energy & Environment Band / Volume 448

ISSN 1866-1793 ISBN 978-3-95806-377-8

Bibliografische Information der Deutschen Nationalbibliothek.
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der
Deutschen Nationalbibliografie; detaillierte Bibliografische Daten
sind im Internet über http://dnb.d-nb.de abrufbar.

Herausgeber und Vertrieb:	Forschungszentrum Jülich GmbH Zentralbibliothek, Verlag 52425 Jülich Tel.: +49 2461 61-5368 Fax: +49 2461 61-6103 zb-publikation@fz-juelich.de www.fz-juelich.de/zb
Umschlaggestaltung:	Grafische Medien, Forschungszentrum Jülich GmbH
Druck:	Grafische Medien, Forschungszentrum Jülich GmbH
Copyright:	Forschungszentrum Jülich 2018

Schriften des Forschungszentrums Jülich
Reihe Energie & Umwelt / Energy & Environment, Band / Volume 448

D 294 (Diss., Bochum, Univ., 2017)

ISSN 1866-1793
ISBN 978-3-95806-377-8

Vollständig frei verfügbar über das Publikationsportal des Forschungszentrums Jülich (JuSER)
unter www.fz-juelich.de/zb/openaccess.

This is an Open Access publication distributed under the terms of the Creative Commons Attribution License 4.0,
which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Kurzfassung

Thermonukleare Fusion von Deuterium (D) und Tritium (T) hat das Potenzial zu einer effizienten, nachhaltigen und sicheren Energiequelle. Das internationale Tokamak-Fusionsexperiment ITER (lateinisch: "Der Weg"), welches ab 2027 in Betrieb genommen werden soll, ist ein Schlüsselprojekt zur Realisierung dieser Technologie. Verlässliche Vorhersagen zur Plasma-Wand-Wechselwirkung (PWW) sind für den Erfolg von ITER und weiteren zukünftigen Fusionsreaktoren unabdingbar, da diese maßgeblich deren Effizienz und Lebensdauer bestimmen werden. Ein wichtiger PWW-Prozess ist die Erosion der Reaktorwand. Diese bestimmt die Lebensdauer von Wandkomponenten und ist außerdem eine Quelle von Verunreinigungen im Plasma, nämlich Beryllium (Be) und Wolfram (W) im Fall der metallischen ITER-Wand. Co-Deposition und Rückhaltung von radioaktivem T durch erodiertes Be ist ein bedeutendes Problem für ITER, da dessen Wandinventar an T auf eine administrative Grenze von 700 g beschränkt ist, aus Gründen der Sicherheit und des Brennstoffkreislaufs. Das Eindringen von Verunreinigungen in das Kernplasma (insbesondere von W, welches einen hohen Ionisationsgrad erreichen kann) führt aufgrund von Strahlungswärmeverlusten zu dessen Abkühlung, was sich negativ auf den Einschluss und Stabilität auswirkt.

Der dreidimensionale Monte-Carlo (MC) Code ERO ist ein etabliertes Werkzeug zur Untersuchung und zum Vorhersagen der PWW und des Plasma-Verunreinigungstransports in Fusionsexperimenten. Jedoch wurde ERO aufgrund technischer Aspekte, wie vor allem der beschränkten Code-Leistungsfähigkeit, bislang für die Untersuchung kleiner Simulationsvolumina (unter $\sim 1\,\mathrm{m}^3$) benutzt, was einen geringen Bruchteil des ITER-Plasmavolumens ($\sim 800\,\mathrm{m}^3$) darstellt. Daher hat das Angehen des ineinandergreifenden Problems von Verunreinigungstransport und PWW im Tokamak zusätzliche Annahmen erfordert (beispielsweise bezüglich des Verunreinigungsgehalts im Plasma). Zudem waren die Möglichkeiten der Code-Validierung auf der Grundlage von Messungen in Fusionsexperimenten oft eingeschränkt, da nur lokale Diagnostiken verwendet werden konnten.

Im Rahmen dieser Dissertation wurde der ERO-Code von Grund auf neu entwickelt, um diese Beschränkungen aufzuheben. Im neuen Code ERO2.0 sind

Algorithmen implementiert, welche es erlauben, ausgedehnte Wandkomponenten mit komplizierter Geometrie in einem Simulationsvolumen von der Größe des ITER-Plasmagefäßes zu untersuchen. Die dadurch erhöhte Komplexität der Simulation erfordert eine um Größenordnungen verbesserte Rechenleistung. Dies wird mit ERO2.0 durch die Optimierung der zeitaufwendigsten Algorithmen realisiert, was zu einer Code-Beschleunigung um etwa 10^3–10^4 führt. Darüber hinaus ermöglicht es die massive Parallelisierung von ERO2.0, Simulationen parallel auf einer großen Anzahl von beispielsweise 10^3 CPU-Kernen eines Supercomputers auszuführen. Dies führt zu einer weiteren Beschleunigung, welche in der Größenordnung der Anzahl an CPU-Kernen liegt.

Das Hauptaugenmerk dieser Arbeit liegt auf der Beschreibung und Nutzung dieser Verbesserungen, welche es erlauben, die physikalischen Fragestellungen allgemeiner und selbstkonsistenter zu behandeln. Zu diesem Zweck wird ERO2.0 für die Modellierung von Experimenten am JET-Tokamak eingesetzt, welcher seit 2011 mit einer ITER-ähnlichen Wall (*ITER-like Wall*, ILW) ausgestattet ist. Hierdurch wird nicht nur die Leistungsfähigkeit des Codes demonstriert, sondern es werden auch neue, ITER-relevante wissenschaftliche Ergebnisse erzielt. Zu diesen gehören die simulierten Verteilungen von Erosions- und Depositionsflüssen an der Be-Wand. Weitere Ergebnisse sind die räumliche Verteilung der Be-Verunreinigungsdichte im Plasma, sowie die (für die weitere PWW ausschlaggebenden) Ladungs-, Winkel- und Energieverteilungen der auf die Wand treffenden Be-Teilchen. Es wird eine gute Übereinstimmung mit experimentellen Messungen aus Infrarotkameras und der linienintegrierten Spektroskopie erzielt, welche den Wärmestrom zur Wand bzw. die Be-Erosionsquelle charakterisieren. Schließlich ermöglicht die Betrachtung des gesamten JET-ILW-Plasmavolumens den Vergleich mit experimentellen 2D-Bildern aus spektroskopischen Weitwinkelkameras, welche den poloidalen Querschnitt des Tokamaks abbilden. Zu diesem Zweck wurde in ERO2.0 eine neuartige synthetische Diagnostik implementiert. Eine gute Übereinstimmung wird im ersten Vergleich zwischen synthetischen und experimentellen Bilden mit Be II Linienemissionsfiltern gefunden. Es wird eine Bewertung und ein Ausblick auf die Bedeutung von ERO2.0 für die zukünftige Fusionsforschung gegeben.

Abstract

Thermonuclear fusion of deuterium (D) and tritium (T) has the potential to be an efficient, sustainable and safe source of energy. The international tokamak fusion experiment ITER (Latin: "the way"), which is scheduled to start operation in 2027, is a key project for the realization of this technology. Reliable predictions on the plasma-wall interaction (PWI) are critical to the success of ITER and further upcoming fusion reactors, since these will significantly impact their efficiency and lifespan. An important PWI process is the erosion of the reactor wall. It determines the lifetime of wall components and is also a source of impurities in the plasma, namely beryllium (Be) and tungsten (W) in the case of the metallic ITER first wall. Codeposition and retention of radioactive T with eroded Be is a significant issue for ITER, since its T wall inventory has an administrative limit of 700 g for safety and fuel cycle reasons. The penetration of impurities (in particular of W, which can reach a high degree of ionization) into the plasma core leads to its cooling due to radiative energy losses, which has a deteriorating impact on confinement and stability.

The three-dimensional Monte Carlo (MC) code ERO is an established tool for the investigation and prediction of PWI and plasma impurity transport in fusion experiments. However, due to technical aspects, such as most importantly the limited code performance, ERO was used so far for examining small simulation volumes (under $\sim 1\,\mathrm{m}^3$), which is a small fraction of the ITER plasma volume ($\sim 800\,\mathrm{m}^3$). Thus, addressing the interdependent problem of impurity transport and PWI in the tokamak has demanded additional assumptions (for example, on the impurity content in the plasma). Moreover, the possibilities of code validation based on measurements in fusion experiments were limited, since only local diagnostics could be used.

In the framework of this thesis, the ERO code has been redeveloped from scratch to remove these restrictions. The new code ERO2.0 implements algorithms that allow to study large and complexly shaped wall components in a simulation volume of the ITER plasma vessel size. The resulting increased complexity of the simulation requires to enhance the code performance by orders of magnitude.

This is realized with ERO2.0 by the optimization of the most time-consuming algorithms, which leads to a code speedup by roughly 10^3–10^4. Furthermore, the massive parallelization of ERO2.0 makes it possible to execute simulations in parallel on a large number of e.g. 10^3 CPU cores of a supercomputer. This leads to an additional speedup, which is in the order of magnitude of the CPU core number.

The main focus of this thesis is the description and exploitation of these improvements, allowing to address the physical questions in a more general and self-consistent way. For this purpose, ERO2.0 is applied to model experiments at the JET tokamak, which is equipped since 2011 with an ITER-like wall (ILW). The application not only demonstrates the code performance, but also provides new, ITER-relevant scientific results. Among these are the simulated distributions of erosion and deposition fluxes at the wall. Further results are the spatial distribution of the Be impurity density in the plasma, as well as the (significant for the further PWI) distributions of charge, angle and energy of the Be particles striking the wall. A good agreement is found with experimental measurements by infrared cameras and line-integrated spectroscopy, which characterize the heat flux to the wall and the Be erosion source, respectively. Finally, the consideration of the entire JET-ILW plasma volume allows the benchmarking with experimental 2D images from spectroscopic wide-view cameras, which observe the poloidal cross-section of the tokamak. For this purpose, a novel synthetic diagnostic was implemented in ERO2.0. A good agreement is found for the first benchmarking of synthetic and experimental images with Be II line emission filters. An evaluation and outlook with respect to the significance of ERO2.0 for future fusion research is given.

Contents

List of Figures

List of Tables

List of Abbreviations

3D-GAPS	(MC neutrals transport code)
ADAS	Atomic Data and Analysis Structure (set of GCR codes and data)
API	Application programming interface
ASDEX	Axially Symmetric Divertor Experiment (tokamak)
AUG	ASDEX Upgrade (tokamak)
BCA	Binary collision approximation
Be	Beryllium
BM	Blanket module
BoT	Bag-of-tasks
C	Carbon
CAD	Computer aided design
CAPS	Chemically assisted physical sputtering
CCD	Charge-coupled device
CD	Collisional-dielectronic
CFC	Carbon fibre composite
COREDIV	(plasma core code)
CPU	Central processing unit
CSG	Constructive solid geometry
CX	Charge exchange
D	Deuterium
DEMO	Conceptual nuclear fusion power plant
DIVIMP	DIVertor IMPurity (MC impurity transport code)
DOK	Dictionary-of-keys
DORIS	(MC impurity transport code)
DS	Debye sheath
EAST	Experimental Advanced Superconducting Tokamak (tokamak)
ECRH	Electron cyclotron resonance heating

EDGE2D	(plasma edge code)
EFIT	Equilibrium FITting (magnetic equilibrium code)
EIRENE	(MC neutrals transport code)
ELM	Edge-localized mode
EMC3	(plasma edge code)
ERO	Erosion and RedepOsition (MC PWI and impurity transport code)
ETSY	Effective total sputtering yield
EUROFER	Reduced activation ferritic-martensitic steel
FW	First wall
GCA	Guiding center approximation
GCR	Generalized collisional-radiative (theory)
GPU	Graphics processing unit
H	Hydrogen
HDF	Hierarchical Data Format (data model, library and file format)
He	Helium
HELIAS	HELIcally Advanced Stellarator (stellarator configuration)
HFS	High-field side
HMM	Homogeneous mixing model
HPC	High-performance computing
HRTS	High-resolution Thomson scattering
ICRH	Ion cyclotron resonance heating
ILW	ITER-like wall
IPP	Max Planck Institute for Plasma Physics
IR	Infrared
ITER	Latin: 'the way' (tokamak)
IWGL	Inner wall guard limiter
JET	Joint European Torus (tokamak)
JPN	JET pulse number
JURECA	Jülich Research on Exascale Cluster Architectures (supercomputer)
KL1	(JET spectroscopic wide-view camera diagnostic)
Kr	Krypton
KS3	(JET line-integrated spectroscopy diagnostic)
LCFS	Last closed flux surface
LFS	Low-field side

LHD	Large Helical Device (stellarator)
LIM	Limiter IMpurity (MC impurity transport code)
LOS	Line of sight
LTE	Local thermodynamic equilibrium
MC	Monte-Carlo
MD	Molecular dynamics
MHD	Magneto-hydrodynamics
MPI	Message Passing Interface (parallel communication standard)
MPS	Magnetic pre-sheath
NBI	Neutral beam injection
NURBS	Non-uniform rational B-spline
OKMC	Object kinetic Monte Carlo
OpenMP	Open Multi-Processing (parallel communication standard)
OS	Observation spot
PARCAS	(MD code)
PEC	Photon emissivity coefficient
PFC	Plasma-facing component
PFCFlux	(magnetic field line tracing code)
PFM	Plasma-facing material
PISCES-B	(linear plasma device)
PSI-2	(linear plasma device)
PtP	Point-to-point
PWI	Plasma-wall interaction
QMB	Quartz micro-balance
RAM	Random access memory
RCP	Reciprocating probe
RF	Resonant frequency
RK	Runge-Kutta
SDTrimSP	(BCA code)
SOL	Scrape-off layer
SOLPS-ITER	(plasma edge code)
SRIM	Stopping and Range of Ions in Matter (set of BCA codes and data)
T	Tritium
TEXTOR	Tokamak EXperiment for Technology Oriented Research (tokamak)

THEODOR	(heat conduction equation code)
TRIDYN	(BCA code)
TRIM	TRansport of Ions in Matter (BCA code)
TRIM.SP	(BCA code)
W	Tungsten
W_f/W	W-fibre enhanced W-composite material
W7-X	Wendelstein 7-X (stellarator)
WallDYN	(PWI and impurity transport code)

List of Symbols

$A(E_{\text{in}}, \theta_{\text{in}})$	Eckstein sputtering yield angular factor
B	Magnetic field
C	Collision term
D	Diffusion coefficient
E_{SB}	Surface binding energy
E_{in}	Ion impact energy
E_{sheath}	Sheath potential
E_{th}	Sputtering threshold energy
E	Electric field; energy
K	Drift coefficient
L	Connection length
P_F	Fusion power
P_H	External heating power
P_α	Alpha particle heating power
Q	Energy gain factor
R	Torus major radius; reflection coefficient
S	Shadowing factor
T_{e}	Electron temperature
T_{surf}	Surface temperature
T	Temperature
Y	Sputtering yield
Z_{eff}	Effective charge
Z	Ion charge state; atomic number
Γ^{ero}	Flux of eroded particles
Γ^{in}	Flux of incident particles
α	Particle species index
δ	Sheath heat transmission coefficient

γ	Energy transfer factor for sputtering
λ	Coulomb logarithm; emission line index; Eckstein fit parameter; decay length
μ	Eckstein fit parameter
ω_c	Cyclotron frequency
ϕ	Toroidal angle; azimuthal angle; electric potential
ρ	Effective plasma radius
τ_E	Energy confinement time
θ_in	Ion impact polar angle
θ	Poloidal angle; polar angle
ε_0	Vacuum permittivity
ε_λ	Emissivity
ε_L	Lindhard reduced energy
ξ	(Pseudo-)random number
a_B	Bohr radius
a_L	Lindhard screening length
a	Torus minor radius
c_s	Ion sound velocity
e	Elementary charge
f	Distribution function
k_B	Boltzmann constant
m, M	Mass
n_e	Electron density
n	Particle density
q	Electric charge; tokamak safety factor; Eckstein fit parameter
r_d	Debye length
r_g	Larmor radius
r	Particle position
s_n	Nuclear stopping power
t	Time
v_D	Drift velocity
v	Velocity

Chapter 1

Introduction

1.1 Fusion reactions

Nuclear fusion is a reaction between two light atomic nuclei forming a heavier nucleus under release of energy. An example is the reaction in which nuclei of deuterium (D) and tritium (T) produce an alpha particle (^{4}He) under release of a neutron (n) and 17.6 MeV of energy (see e.g. [208, p. 4]):

$$\mathrm{D} + \mathrm{T} \rightarrow {}^4\mathrm{He} + \mathrm{n} + 17.6\,\mathrm{MeV}\,. \tag{1.1}$$

The released energy comes from the mass defect, and is carried to 80 % by the neutron and to 20 % by the alpha particle.

A fusion power reactor would provide an "economically affordable, environmentally sustainable and politically acceptable" [175] energy source (see e.g. [71][30][175][75, pp. 3-19][180, pp. 10-12] and [1,2,3,4]).

However, the task is a challenging one, since the cross-sections of fusion reactions are very small unless the nuclei have large kinetic energies. The reason for this lies in the positive charge of the nuclei, which leads to a strong repulsion between them due to the Coulomb force (Coulomb barrier). The nuclei need energies large enough (or somewhat below due to quantum-mechanical tunneling) to overcome this barrier, so that the short-range nuclear forces can take effect.

For energies below several hundred keV, the D-T reaction has a much higher cross-section than other fusion reactions (e.g. D-D or D-^{3}He), with a maximum around 100 keV [208, p. 5]. First-generation fusion reactors are thus foreseen to operate with a D-T fuel [163, p. 9] of the optimum mixture 50:50 [75, p. 50].

[1] `https://www.iter.org/sci/Fusion`
[2] `http://www.ccfe.ac.uk/Why_fusion.aspx`
[3] `http://fusionforenergy.europa.eu/understandingfusion/merits.aspx`
[4] `https://www.euro-fusion.org/faqs/`

1.2 Thermonuclear fusion

A possible way of supplying the required energy to the fusion reactants is to heat them to sufficient temperatures (around 10 keV[5]) [208, p. 2], so that their thermal energy is high enough for the reaction to occur (thermonuclear fusion).

At such high temperatures, the fuel (i.e. the hydrogenic isotopes) is fully ionized and exists in a state of matter called the plasma: the electrically charged electrons and ions composing the substance are not bound to each other, making it highly conductive and its motion dominated by the long-range electro-magnetic forces.

Since the cross-section for the D-T reaction at the envisaged temperatures is still much smaller than the cross-section for elastic Coulomb scattering, the fuel plasma needs to keep the heat produced by the fusion for a sufficient time to achieve a chain reaction leading to net energy gain. This makes a confinement of the plasma necessary, which can be obtained in different ways. The three most important confinement concepts are:

- Gravitational confinement, which is found only in stars (stellar fusion) due to their great mass. During most of their lifetime, stars generate their energy by fusing hydrogen to helium in their core [17, p. 98]. This happens predominantly through a series of fusion chain reactions, i.e. the proton-proton (p-p) and carbon-nitrogen-oxygen (CNO) cycles [17, pp. 98-99].

- Inertial confinement, in which rapid heating by an external 'driver' (e.g. a short laser pulse) is applied to a frozen D-T pellet, causing it to implode, thereby compressing and heating to conditions required for fusion (see e.g. [150, pp. 23-24,101-105]). The time required to burn a sufficient part of the pellet to obtain a net energy gain (which was not yet achieved) lasts no longer than a few nanoseconds [82]. Therefore, the inertia of the fuel is used to provide the confinement.

- Magnetic confinement, which makes use of the Lorentz force that causes charged plasma particles to follow magnetic field lines (see section 2.3.1). Thus, a strong magnetic field ($B \sim 1$–$10\,\mathrm{T}$ [180, p. 5]) can be used to trap the plasma. Various magnetic configurations are developed for this purpose, including tokamaks and stellarators, which are discussed in detail below.

A fusion reactor should confine and heat the plasma until the generated fusion power P_F exceeds the applied external heating power P_H, so that net power is

[5]Here and in the following, the temperature is given in units of eV, which is a common notation in plasma physics. To obtain the value in SI units [K], one has to multiply T [eV] with $1/k_\mathrm{B} \approx 1.16 \times 10^4\,\mathrm{K/eV}$, where k_B is the Boltzmann constant.

generated (break-even condition):

$$Q = \frac{P_F}{P_H} \geq 1\,, \tag{1.2}$$

where Q is called the energy gain factor. Up to now, the highest reported value achieved for a D-T mixture in a fusion experiment is $Q = 0.65$ from the Joint European Torus (JET) [195], which comes close to break-even.[6] The international fusion experiment ITER ('the way' in Latin), which is now under construction and is planned to start operating in 2027[7], is expected to demonstrate the feasibility of a significant net energy gain from fusion, with about $Q \sim 10$ [42]. An even higher Q is envisaged for the conceptual reactor commonly referred to as (EU) DEMO [215], which is foreseen as the first demonstrational fusion power plant [70] by the EUROfusion consortium[8].

In the long term, $Q = \infty$ should be envisaged. This corresponds to a plasma which heats itself sufficiently by the energetic alpha particles produced in the fusion reactions, so that no external heating power is needed (ignition, $P_H = 0$). The condition for ignition to occur can be formulated as [75, p. 66]

$$P_\alpha \geq P_L\,. \tag{1.3}$$

P_α is the internal heating of the plasma by collisions with the alpha particles, which are produced in the D-T reaction and carry about 20 % of the fusion power, i.e. $P_\alpha = P_F/5$ (the neutrons are assumed to leave the plasma without interacting with it). P_L is the power leaving the system (losses). The main loss mechanisms are conduction (energy lost with particles that leave the plasma due to imperfect confinement) and bremsstrahlung radiation. These two processes are described by the energy confinement time τ_E, a characteristic time at which the system loses energy [208, p. 9].

An important quantity that characterizes the efficiency of a fusion reactor is the triple product $nT\tau_E$, where n and T are the plasma density and temperature. The so-called Lawson criterion [117] for achieving ignition can be formulated as a minimum threshold that must be exceeded by the triple product, which is about [208, pp. 11-12]

$$nT\tau_E \approx 3\text{–}5 \times 10^{21}\,\mathrm{keVs/m^3} \tag{1.4}$$

for the D-T reaction.[9] In the last decades of fusion research, considerable progress

[6] An even higher $Q \sim 0.6$–1 was measured at JET transiently [146].

[7] See `https://www.iter.org/construction/timeline`.

[8] `https://www.euro-fusion.org/`

[9] As stated in [208, p. 12], the exact value on the right-hand side of equation (1.4) depends

was made in coming closer to this goal, as shown in Figure 1.1. In JET D-T plasmas for instance, triple products of about 1×10^{21} keVs/m^3 were reported [83].

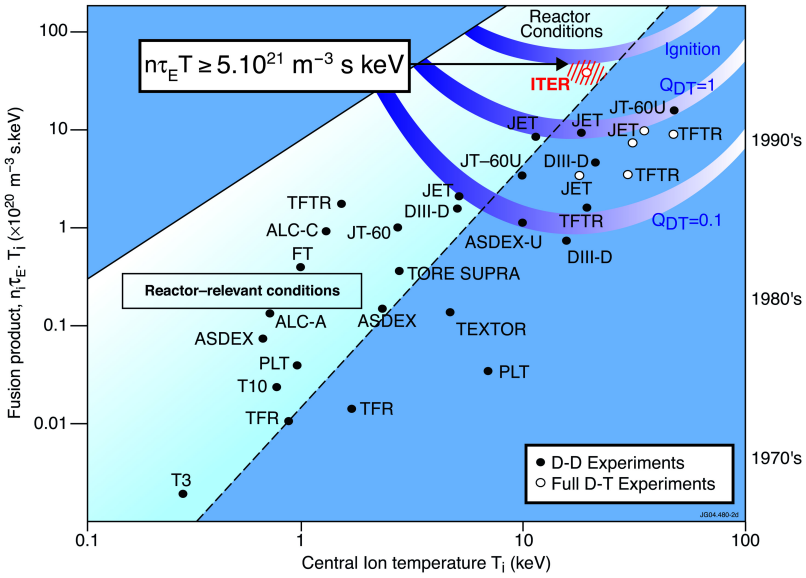


Figure 1.1: Progress of fusion research in the last decades, measured by the triple product $nT\tau_E$ achieved in different fusion devices. Image source: `https://www.euro-fusion.org/2011/09/progress-in-fusion/`.

1.3 Magnetic confinement

For magnetic confinement, strong magnetic fields can be achieved using an arrangement of magnetic coils.[10] The transport of the charged plasma particles then occurs predominantly along the magnetic field lines. If a linear arrangement of coils is used (linear plasma device), particle losses occur at the ends of the field lines. A part of the losses can be mitigated by the magnetic mirror effect: by making the field stronger at the ends of the confinement area, particles will experience a force that eventually forces them to reverse their velocity. However, the mirror effect occurs only for a certain range of particle velocities, while particles outside this range will escape the mirror.

on whether peak or average values are considered for the density and temperature, and which spatial profiles (e.g. flat or parabolic) are assumed for them.

[10] In ITER (but also in other fusion experiments such as the EAST tokamak [214]), superconducting coils are used to avoid coil heating during long plasma pulses and energy losses.

A more efficient approach is therefore to close the field lines by a toroidal ('donut-like') arrangement of coils. Figure 1.2(a) shows an illustration of a torus and its principal directions (toroidal and poloidal angle, minor and major radius). The magnetic field in toroidal direction is created by coils that poloidally enclose the torus (see Figure 1.3).

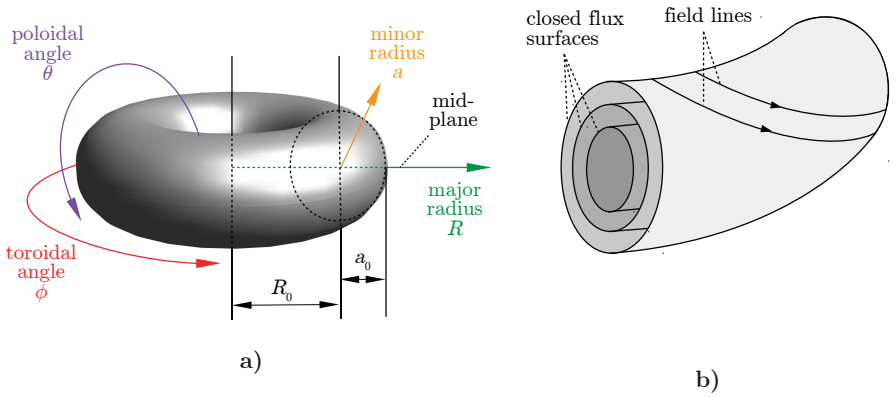


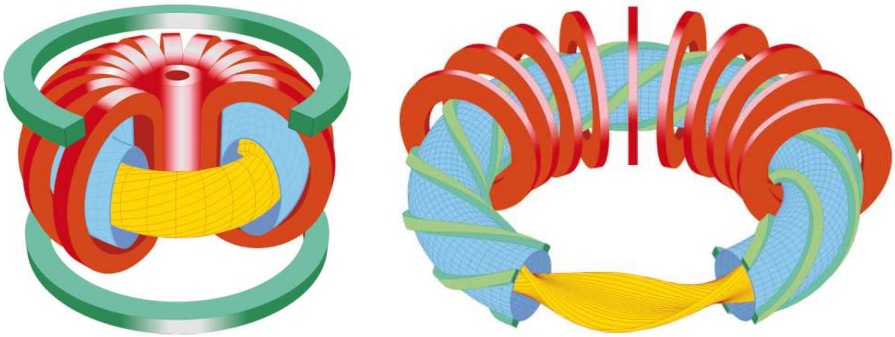
Figure 1.2: (a) Illustration of a torus and its principal directions: toroidal angle, poloidal angle, major radius and minor radius (also referred to as the 'plasma radius' in the context of magnetic confinement fusion). (b) A torus segment illustrating the helically twisted magnetic field lines, which are lying on magnetic flux surfaces.

The bending of field lines to a torus results in a curvature of the magnetic field as well as in a gradient: the magnetic field is stronger at a small major radius (high-field side, HFS) and weaker at a large major radius (low-field side, LFS). These inhomogeneities lead to a drift $\boldsymbol{v}_D \propto (\boldsymbol{R} \times \boldsymbol{B})/q$ of the particles, where $\boldsymbol{R}$ is the radial direction, $\boldsymbol{B}$ the magnetic field and q the particle charge [75, p. 160]. Consequently, positively charged ions drift in one vertical direction, while negatively charged electrons drift in the opposite direction. This charge separation creates an electric field $\boldsymbol{E}$, which in turn creates an additional drift $\boldsymbol{v}_D \propto (\boldsymbol{E} \times \boldsymbol{B})$ that causes both ions and electrons to escape in the outward radial direction [180, p. 45].

To avoid this loss mechanism, an additional magnetic field component is required in the poloidal direction. This adds a twist to the magnetic field lines, which become helical ('rotational transform') as shown in Figure 1.2(b). The field lines lie on a set of nested surfaces that are called the magnetic flux surfaces. These are discussed in more detail below in section 1.5. The result of the rotational transform is that during the poloidal rotation of plasma particles, the vertical drift is compensated [75, pp. 284-286]. Therefore, the build-up of charge separation and outward $\boldsymbol{E} \times \boldsymbol{B}$ drift is prevented, which greatly improves the magnetic confinement.

The twist of the helical field lines is often characterized by the safety factor $q = \Delta\phi/(2\pi)$ [208, p. 111].[11]. Here, $\Delta\phi$ is the change of toroidal angle that is required for a magnetic field line to complete one rotation in the poloidal plane. A large q means a low contribution of the poloidal field component to the total magnetic field, corresponding to a 'slow' poloidal rotation. From magneto-hydrodynamics (MHD) it is known that plasma instabilities, which are harmful for the confinement, occur if q is too low (Kruskal-Shafranov limit, see e.g. [75, p. 391]), explaining the term 'safety factor'. One often uses the quantity q_{95} (q at 95 % of the minor radius at the LFS midplane) to characterize the safety at the plasma edge. E.g. for the experiments at the JET tokamak described in chapter 4, values of $q_{95} \approx 4$ were measured.[12]

Different designs of toroidal magnetic confinement exist, such as the tokamak, the stellarator, the reversed field pinch, the spheromak and others [75, p. 116]. Among these, the tokamak concept has up to now been best studied and achieved the best overall performance, followed by the stellarator concept [75, p. 116]. These two concepts, which are illustrated in Figure 1.3, differ in the way how the poloidal magnetic field is created.



a) **b)**

Figure 1.3: Schematic representation of the (a) tokamak and (b) stellarator designs. The yellow surfaces represent magnetic flux surfaces, with the red lines indicating the twisted magnetic field lines. The blue surfaces represent the plasma vessel. The red and green structures represent magnetic field coils. Image source: `http://earth-chronicles.com/science/fusion-miracle-of-world-energy.html`.

In a tokamak, a time-varying electric current is driven through a central coil. This induces a toroidal current in the plasma, which acts as a secondary transformer winding. The plasma current in turn creates the poloidal magnetic field. The

[11] This quantity is mostly used in the context of tokamaks. For stellarators, it is more common to refer to the inverse quantity $\iota = 2\pi/q$ ('rotational transform') [208, p. 113].

[12] Data were obtained from the JET online data services on `https://data.jet.uk/`.

stellarator in contrast produces the poloidal magnetic field by helical magnetic coils. Examples for tokamaks are the Tokamak EXperiment for Technology Oriented Research (TEXTOR, which was decomissioned in 2013) [169] and the Axially Symmetric Divertor Experiment (ASDEX) Upgrade (AUG) [129] in Germany, the Experimental Advanced Superconducting Tokamak (EAST) in China [214], DIII-D in the USA [124], JET in the United Kingdom [125] and ITER in France [153]. JET is currently the largest operating tokamak, with a major radius $R_0 \approx 3\,\mathrm{m}$, a minor radius $a_0 \approx 1.25\,\mathrm{m}$, a central magnetic field $B \approx 3.5\,\mathrm{T}$ and a central electric current $I \approx 7\,\mathrm{MA}$ [180, p. 5]. Examples for operating stellarators are the Large Helical Device (LHD) in Japan [94] and Wendelstein 7-X (W7-X) in Germany [110].

Both tokamak and stellarator concepts have their specific advantages and disadvantages, as discussed e.g. in [87] or [75, p. 424]. The stellarator offers an intrinsic steady state operation (as no transformer action is required) as well as the absence of disruptions (a sudden decrease in confinement which often occurs in tokamaks [51]). Both are highly desirable properties. However, the so-called neoclassical confinement is worse in the stellarator, and also the technical complexity introduced by the helical coil system is a disadvantage compared to the tokamak. Historically, the tokamak and stellarator concepts evolved around the same time in the 1950s [18, 179]. Following the early success of the T-3 device in achieving high electron temperatures up to 1 keV in 1969 [147], the focus of the research community shifted largely to the tokamak line of magnetic confinement [203, pp. 2-3][89]. Recently, the stellarator regained attention thanks to the availability of modern supercomputers[13]. These made it possible to calculate the complex shape of the magnetic coils required for an optimized confinement. The recently constructed W7-X device is expected to demonstrate the improved confinement in the helical advanced stellarator (HELIAS) configuration [205]. In the long term, W7-X could allow—together with potential next-step HELIAS devices and technology results from a tokamak DEMO—to build a stellarator fusion power plant [70, p. 29].

1.4 Plasma heating

Three main ways exist for heating a tokamak plasma to the required temperature of around 10 keV [75, pp. 118-119, 534-632]. First, the electric current creates ohmic heating due to transfer of kinetic energy to the charged particles and their subsequent scattering. The resistivity of the plasma reduces with its temperature,

[13] See e.g. `https://horizon-magazine.eu/article/twisting-design-fusion-reactor-thanks-supercomputers_en.html`.

therefore ohmic heating is efficient only up to several 10 keV. Second, neutral beam injection (NBI) is used. This means that ions (typically of the same species as the fuel) are accelerated outside the plasma vessel via electrostatic fields, and subsequently neutralized so that they can penetrate deep into the plasma unaffected by the magnetic field. Subsequently, they transfer their kinetic energy to the plasma via collisions. NBI is also useful for plasma fueling. Third, electro-magnetic waves are injected by antennas or gyrotrons. Their frequencies are chosen at the respective resonant frequencies (RF) of the ion or electron cyclotron motion for accelerating them, which leads to plasma heating by subsequent scattering (ICRH, ECRH). As these frequencies depend on the particle mass and the magnetic field strength, specific regions of the plasma can be heated for controlling the temperature profile.

Experimentally, it was found that the confinement deteriorates as higher heating is applied [208, p. 152]. However, in 1982 it was discovered in the ASDEX tokamak that as heating power exceeds a certain limit, a discontinuous improvement of the confinement time by a factor of two occurs [201, 202]. This could subsequently be reproduced in many other tokamaks (see e.g. [49]). The two regions of low and high confinement are referred to as L- and H-mode. While the higher confinement in the H-mode is certainly desirable, it is accompanied by the existence of localized edge instabilities, the so-called edge-localized modes (ELMs) [75, p. 504-506]. These involve a periodic expulsion of energetic particles from the hot plasma core to the wall, which is seen as a potential threat for ITER operation [120, 152].

1.5 The plasma edge and plasma-facing materials

Plasma transport occurs predominantly along magnetic field lines, with cross-field velocities being typically several orders of magnitude lower than the velocities parallel to the field [182, p. 15]. Closed magnetic flux surfaces thus efficiently confine the plasma. However, beyond a certain minor radius the flux surfaces must start intersecting the plasma-facing components (PFCs) of the device (open flux surfaces). At these 'plasma-wetted' PFC areas, particles efficiently escape from the plasma with their parallel velocity (plasma exhaust).

The boundary between open and closed surfaces is called the last closed flux surface (LCFS) or the separatrix, depending on whether it is created by a limiter or a divertor (see below). The region inside the LCFS or separatrix is called the core or main plasma, the region outside is called the scrape-off layer (SOL). The plasma edge is a region that includes the SOL, but also extends to some region inboard of the LCFS or separatrix [182, p. xviii]).

The existence of an SOL and plasma exhaust is not only inevitable [182, p. 4],

but also has beneficial aspects for fusion power plant operation. First, it is an important mechanism to transfer the heat (generated by the fusion reaction) via the wall to the cooling system of the power plant [180, pp. 181-182]. Second, it removes the helium ash produced by the fusion reaction from the plasma. Otherwise, an accumulation of helium in the core plasma would eventually lower the fusion reaction rate [180, p. 214].

However, plasma exhaust and the resulting plasma-wall interaction (PWI) have also undesirable consequences. PWI includes processes such as erosion (e.g. due to sputtering, see section 2.2.1), melting, surface morphology modification etc., which limit the lifetime of PFCs [102]. Erosion is furthermore a source of plasma impurities. High-Z impurities (such as tungsten with $Z = 74$) also efficiently drain energy from the plasma by ionization and in particular by light emission. This radiative cooling will obstruct the fusion reaction in the plasma core or even lead to a collapse of the plasma (whereas radiative cooling in the plasma edge is often desirable to protect the PFCs and is even induced by impurity seeding [99]). Finally, eroded impurities will eventually redeposit again on PFCs. This can lead to material mixing (which can change PFC properties in an undesirable way [6, 39, 151][45, pp. 287, 294, 309]), and also to codeposition of fuel atoms [33, 148]. The latter can lead to the build up of a fuel inventory (retention) in net deposition zones. Retention of tritium, which is radioactive with a half-life time of 12.32 years [122] and therefore not abundant in nature, is especially problematic for fuel cycle and safety reasons, which is why an administrative limit of 700 g was set for the tritium inventory in the ITER plasma vessel [166].

The control of plasma exhaust, PWI and impurity transport in the plasma edge are therefore indispensable for an efficient fusion reactor operation. Two plasma exhaust concepts exist for this: the limiter and the divertor [182, pp. 15-19][75, pp. 500-502]. These are illustrated in Figure 1.4. A limiter is a wall element which 'touches' the plasma and thereby defines the LCFS. Thus, it creates a defined region where the dominant part of PWI occurs, and protects other parts of the wall. A disadvantage of this concept is that impurities eroded from the limiter can easily enter the core plasma. A more advanced concept is the divertor. It uses additional magnetic coils for changing the magnetic topology, resulting in a separatrix with one or more X-points where magnetic field lines cross. This way, the core region is separated from the plasma contact points, which are located on target plates in the so-called divertor legs. Another advantage of the divertor is that it makes the H-mode more easily accessible [75, p. 502]. Today, many fusion devices use a divertor (often in the lower part of the plasma vessel) in combination with limiters.

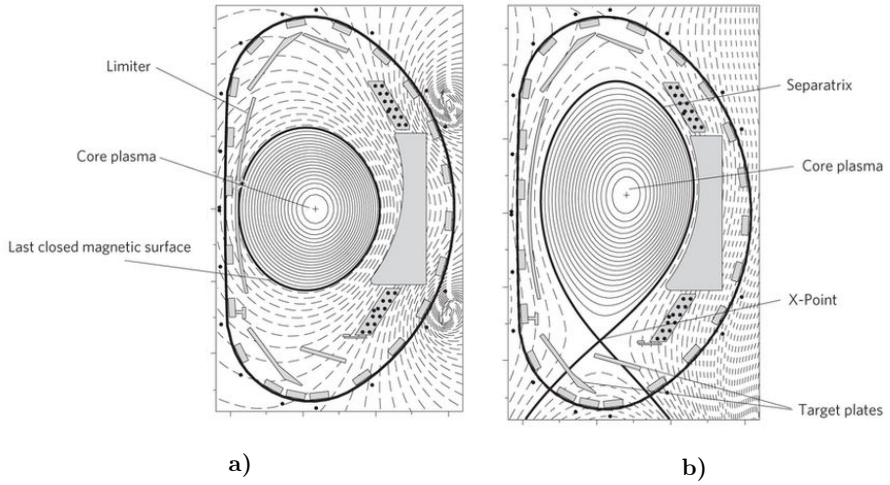


Figure 1.4: Illustration of plasma exhaust concepts: (a) the limiter and (b) the divertor. Image source: [145].

Due to PWI effects, the plasma-facing materials (PFMs) of the first wall (FW) inside the plasma vessel have to be carefully chosen, in particular for the limiters and divertor target plates where particle and heat fluxes are the highest. Currently, the most relevant materials are carbon (C), beryllium (Be) and tungsten (W). Carbon—used in the form of graphite or carbon fibre composite (CFC)—has the advantages of low $Z = 6$, high thermal conductivity, strong bonding and absence of melting [63, 102, 118]. However, it is highly susceptible to chemical erosion and chemical sputtering (see section 2.2.1) and leads to unacceptably high levels of fuel retention by codeposition compared to metallic PFCs [63, 102]. Beryllium has a very low $Z = 4$ and therefore leads to exceptionally low radiative cooling; also it is an excellent oxygen getter [63, 102, 118]. Its disadvantages are its toxicity [86] and its rather low melting point of 1285 °C [118]. Tungsten in contrast has a very high melting point of 3410 °C [118] and has a high resistance to sputtering [63]. Its disadvantages are the high $Z = 74$, which leads to potentially high amounts of radiative cooling [63], and also its brittle nature and higher activation by neutrons [118].

The ITER design foresees a main chamber (the region outside the divertor) with a Be first wall (blanket modules, BMs) [153], and a full-W divertor [151]. JET has been upgraded in 2011 from a carbon wall (JET-C) to an ITER-like wall (JET-ILW), with beryllium limiters in the main chamber and a divertor comprising bulk W as well as W-coated CFC components (see chapter 4). Advanced materials are currently being investigated for DEMO, such as the reduced activation

ferritic-martensitic steel EUROFER [123], W-fibre enhanced W-composite materials (W_f/W) [46, 140], smart self-passivating alloys [121, 207] and others (see [119] and references therein).

1.6 Modeling of plasma-wall interactions and impurity transport

The ILW makes JET an ideal testbed for ITER-relevant studies, including the investigation of PWI and impurity transport. Experimentally, it was found that the primary erosion source (Be) is strongly reduced in JET-ILW compared to the carbon source in JET-C, as well as the long-term fuel retention [33].

Modeling and simulations with computer codes are important for the interpretation and understanding of such experiments, as well as for the direct extrapolation of the results to ITER and quantitative predictions. An example is the 2D Monte-Carlo (MC) PWI and impurity transport code LIM (Limiter IMpurity) [183].[14] Predictive LIM modeling was performed for ITER reference pulses, focusing on selected shaped blanket modules [43]. The simulations predict PFC lifetimes corresponding to 1500–216 000 ITER pulses, accounting for the large uncertainties in the imposed plasma conditions and in the Be sputtering yields. Also, the amount of possible pulses until the tritium inventory limit is reached was evaluated to 1170–13 100, considering the additional uncertainty in the estimated codeposition ratios.

Another MC PWI and impurity transport code is ERO (Erosion and Redeposition) [112, 136, 138, 139]. ERO offers the unique feature of following eroded particles in 3D and with full resolution of their gyromotion orbits (i.e. without the guiding center approximation, see section 2.3.1). This way, effects such as prompt deposition of heavy ions (e.g. W) can be accounted for [53, 54, 106]. The ERO code is well-established and has been successfully applied to the modeling of experiments in a wide range of fusion devices, including TEXTOR [109], PSI-2 [68], JET-ILW [21], DIII-D [54], PISCES-B [20], AUG [137], EAST [55] and PILOT-PSI [199].

In [25], a benchmark between the ERO and LIM codes for the ITER BM modeling was performed, leading to an increased confidence in the initial LIM calculations in [43]. An excellent agreement between the two codes was found for

[14] An extension of LIM for divertors, called DIVIMP (DIVertor IMpurity), was later developed [185], which is capable of simulating the global SOL transport in 2D. DIVIMP is an important part of the WallDYN-DIVIMP code package [172], which allows to investigate the global erosion, deposition and wall composition dynamics in the 2D poloidal cross-section of a tokamak.

the Be erosion and redeposition patterns and the PFC lifetimes, if "all possible input parameters (plasma fluxes, panel geometry, sputtering yields and magnetic shadowing) are made equivalent" [25]. The ERO modeling included additional parameter studies, involving different concentrations of Be impurity content in the plasma, as well as high and low estimates for the Be sputtering yields (which are further discussed in section 2.2.1). The 'worst-case' of high-density pulses led to lifetime estimates of 1100–4200 depending on the Be sputtering assumptions, while the inclusion of Be impurities reduced the net erosion by up to 40 %.

In order to address the uncertainties e.g. in the Be sputtering assumptions, ERO modeling was performed subsequently for Be erosion of JET limiters, in both limiter pulses [21, 23, 24, 33, 35] and L-mode pulses where erosion is enhanced at specific locations by ICRH antennas [115]. After including several improvements, such as a new analytic expression for particle tracking in the sheath region and implementation of BeD release by chemical sputtering, indications were obtained for the validity of the more optimistic lower Be sputtering estimate [21].

However, the ERO code—and consequently, both the validation at JET and predictions for ITER—suffers from a fundamental problem that is related to the limited size of the simulation volume (and PFCs therein) that can be used for the modeling (below $\sim 1\,\mathrm{m}^3$). Since the simulation volume is very small compared to the total size of the plasma volume (e.g. $\sim 840\,\mathrm{m}^3$ for ITER[15]), only a limited part of the eroded impurity trajectories can be considered (local transport). The global transport of impurities, which leads to erosion and deposition in remote areas, cannot be accounted for in a self-consistent way.

This obscurs the evaluation of the JET modeling specifically for pulses with high plasma temperatures, for which high Be concentrations are found. In these pulses, Be self-sputtering dominates the erosion [35]. The description of self-sputtering in ERO requires the simplified assumption of a homogeneously distributed steady-state Be concentration in the plasma, which is treated as an input parameter. This parameter is estimated from spectroscopic measurements, which requires further assumptions (i.e. of a single Be charge state $Z = 3$ or $Z = 4$ being present in the plasma, see appendix C) and leads to further uncertainties due to measurements being integrated across the spectroscopy observation volume, see section 4.3. Furthermore, the usage of such estimations of the Be concentration based on experiments is not possible for predictive ITER modeling, where parameter studies are required instead.

[15] `https://www.iter.org/mach`

1.7 Topic and outline of the thesis

In order to address the uncertainties coming from the limited simulation volume in ERO, it is desirable to increase the volume size significantly. This requires a fundamental redevelopment of the code, which has been done in the frame of this thesis. The new code version ERO2.0 introduces several technical enhancements. Most notably, a new and flexible description of the wall geometry is implemented. Also, ERO2.0 is massively parallelized, and thus can be executed on a supercomputer, benefiting from its large resources in computational power and memory. These improvements allow considering the entire first wall of an ITER-sized fusion device, with large and complexly shaped PFCs, and the global transport of eroded impurities within the device.

Thus, ERO2.0 is a unique code, which offers a self-consistent model of global erosion and deposition in 3D. This not only reduces the uncertainties related to plasma impurity concentrations of the previous local modeling, but also allows a comprehensive understanding of global phenomena in the plasma edge. Furthermore, the global approach allows verification using a larger number and variety of experimental diagnostics.

In *chapter 2*, a description of the PWI and impurity transport model and underlying data used in ERO (which remain unchanged by the code upgrade) is provided. The focus of the description is on Be sputtering and atomic data, since it is the material investigated in the modeling shown in chapter 4.

Chapter 3 provides a comprehensive description of the innovations that are introduced in ERO2.0. This includes the massive parallelization, the new and flexible description of wall geometry, the optimization of algorithms (related to 3D geometry) which are the main code performance bottleneck, the new and more efficient file format and structure, the domain decomposition and sparse matrix techniques, which enable the treatment of large 3D data sets, and the new magnetic field line tracing technique, which is important for calculating connection length and shadowing patterns on the wall as well as the plasma flow velocity.

Chapter 4 presents the first ERO2.0 application for the modeling of JET-ILW experiments. The focus of the modeling is on the physical sputtering of Be PFCs in JET limiter pulses, continuing and extending the above-mentioned efforts with the previous code version [21, 23, 24, 33, 35]. In contrast to those, the simulation volume now covers the entire plasma edge in 3D with all relevant PFC elements, thus enabling a self-consistent description of processes such as Be self-sputtering. General aspects of the simulation results are discussed, including the surface erosion and deposition patterns of Be, its migration in the plasma edge and its energy and

angle distributions upon impact on the wall. Experimental benchmarking with 2D images from infrared (IR) and spectroscopic wide-view cameras is provided. For the latter, a novel synthetic diagnostic is implemented in ERO2.0, which renders 2D images of spectroscopic line emission, as will be described in detail. Finally, a quantitative comparison of the local and global modeling approaches is provided with respect to the Be erosion measured by spectroscopy. The local approach uses estimations of the Be impurity concentration (for the treatment of Be self-sputtering) based on experiments, while the global one obtains it in a self-consistent way.

Chapter 5 provides the summary and conclusions with respect to the performed code development and the first results obtained by applying ERO2.0 to JET-ILW. Furthermore, an outlook is given on possible future applications of ERO2.0 for fusion research.

Chapter 2

Scientific basis of the ERO code and underlying data

In this chapter, the fusion-relevant PWI and impurity transport processes simulated with ERO are reviewed. The focus is on those processes that are incorporated in ERO2.0 and used for the simulations presented in chapter 4. Because those simulations are performed for JET (ITER-like) beryllium-based PFCs, special attention is paid to beryllium data.

2.1 General concept and workflow of the ERO code

Figure 2.1 illustrates the general workflow of the ERO code, which can be divided into the PWI and the impurity transport parts. For a target surface exposed to a plasma, ERO can calculate PWI processes (see section 2.2) such as reflection, physical sputtering, chemical erosion, chemically assisted physical sputtering and (co-)deposition. Subsequently ERO creates an ensemble of computational 'test particles' ($\sim 10^4$–10^5), which represent the eroded target material that enter the plasma as impurities. The initial velocities of these particles are sampled from certain distributions (see section 2.2.1) using pseudo-random numbers (MC method). The transport of the test particles, i.e. their trajectories in the plasma, is calculated using a combination of deterministic and stochastic differential equations (see section 2.3.3). The former are used to account for the influence of electro-magnetic fields on charged particles by the Lorentz force, the latter to account for processes such as ionization, recombination, dissociation of molecules, or the friction with and diffusion in the plasma. An important simplification used for the transport description is the test particle approximation (see section 2.3.3). This means that nonlinear effects due to the interaction of test particles with each other or their

influence on the electro-magnetic fields and plasma parameters (e.g. electron density and temperature) are treated as negligible [106]. The test particle approximation is valid for the typical levels of plasma impurity concentration in most fusion-relevant SOL conditions, which is confirmed by the good agreement found between ERO results and experiments (see section 1.6). Only in specific cases of strong impurity sources, e.g. due to external injection (puffing), the "possible effects of local cooling and increase of plasma density" [106] need to be considered, see [56, 197, 198]. The test particle approximation allows treating the electro-magnetic fields and plasma parameters as a constant 'plasma background' used in ERO as an input (see section 3.6), which is decoupled from the problem of impurity transport. Plasma backgrounds are obtained either from experimental measurements (combined with analytic models [109]) or from dedicated plasma edge codes such as SOLPS-ITER [210], EDGE2D-EIRENE [178], SOLEDGE2D-EIRENE [41] or EMC3-EIRENE [72]. The PWI processes are finally calculated

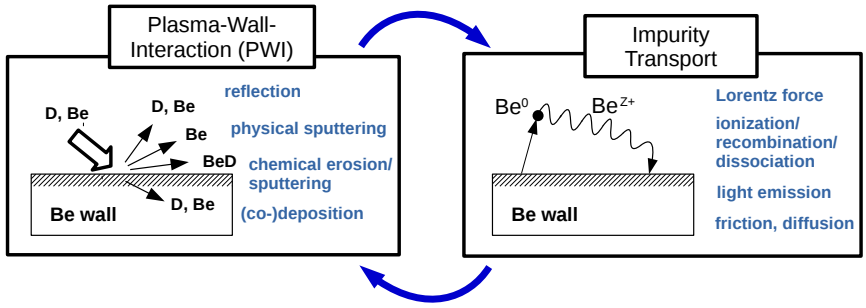


Figure 2.1: Illustration of the general workflow of the ERO code, based on the example of a beryllium surface exposed to a deuterium plasma.

also for the test particles that hit the target. If required for obtaining steady-state conditions (see section 4.3.2 and appendix D), this procedure can be repeated (multi-step runs). In each new step, the additional PWI caused by the test particle impact from the previous step is considered when creating the new test particle ensemble. Also, the change in material composition from the previous time step may be taken into account, which is calculated either using the homogeneous mixing model (HMM) [103, 128] or by coupling ERO with the SDTrimSP code [61, 62, 103].

2.2 Plasma-wall interaction

2.2.1 Erosion

Overview

A reliable and precise prediction of erosion is crucial for the safe and efficient operation of fusion reactors, and is one of the major motivations for ERO simulations. One can distinguish between physical and chemical erosion. Physical erosion (sputtering) is caused by momentum transfer via collisions with energetic projectiles (ions or neutrals) incident on a target, which cause target atoms to overcome the surface binding energy E_{SB} and to be ejected from the surface. Chemical erosion in contrast involves thermal projectiles, which form chemical bonds with target atoms. A combination of physical and chemical erosion, in which both kinetic energy and chemical bonds are involved, is referred to as chemical sputtering or chemically assisted physical sputtering (CAPS).

As shown by experiments at PISCES-B [58, 141, 142] and JET [35], the erosion of Be under D irradiation is determined mainly by physical sputtering and CAPS with the release of BeD molecules. The latter mechanism is also confirmed by molecular dynamics (MD) simulations [14–16]. Both experiments and MD modeling show a significant fraction (up to 80 %) of the total eroded Be being released as BeD [35].

In JET limiter pulses, where deuteron impact energies E_{in} and limiter surface temperatures T_{surf} are typically higher than in divertor pulses, physical sputtering is the dominant erosion mechanism [35]. At energies $E_{\mathrm{in}} \sim 75\,\mathrm{eV}$ and limiter temperatures $T_{\mathrm{surf}} \sim 200\,^{\circ}\mathrm{C}$, two third of the eroded Be originates from physical sputtering and one third from CAPS. At higher limiter temperatures $T_{\mathrm{surf}} \sim 520\,^{\circ}\mathrm{C}$, the latter contribution vanishes. At high impact energies $E_{\mathrm{in}} > 150\,\mathrm{eV}$, Be erosion is entirely dominated by self-sputtering.

In the following, the focus will be on physical sputtering due to its dominant role (more than two thirds of the total sputtering) in the modeling of Be erosion in JET limiter plasmas presented in chapter 4. Modeling of Be CAPS (and also of chemical erosion e.g. of carbon-based PFCs) is possible with ERO1.0 but requires not only knowledge of the corresponding erosion yields, but also needs the consideration of molecular reactions within the plasma during the transport, which is not implemented yet in ERO2.0. Details on the treatment of chemical erosion, CAPS and molecular reactions in ERO1.0, as well as on other erosion processes not yet introduced in ERO1.0/ERO2.0, such as blistering or radiation enhanced sublimation, can be found e.g. in [102].

Physical sputtering threshold

A necessary condition for physical sputtering to occur is that the projectile can transfer sufficient energy to at least one target atom (through a collision cascade) so that it overcomes the surface binding energy E_{SB}. The binding energy is often assumed to be equal to the heat of sublimation, e.g. $E_{\mathrm{SB}}(\mathrm{Be}) = 3.38\,\mathrm{eV}$ [182, p. 117]. In the most extreme case of a head-on elastic collision between the target and projectile atoms, the maximum amount of transferred kinetic energy is

$$\Delta E_{\mathrm{kin}} = \frac{4M_1M_2}{(M_1+M_2)^2} E_{\mathrm{in}} = \gamma E_{\mathrm{in}}\,, \tag{2.1}$$

with E_{in} being the projectile energy, M_1 and M_2 the projectile and target atom masses, and γ the energy transfer factor. Using equation (2.1), an estimate is derived in [182, p. 118] for the projectile energy threshold required for sputtering to occur,

$$E_{\mathrm{th}} = \frac{E_{\mathrm{SB}}}{\gamma(1-\gamma)}\,, \tag{2.2}$$

with the remark that the usage of fitted values for E_{th} as described below should be preferred. E.g. for self-sputtering, the energy transfer factor is $\gamma = 1$, which leads to $E_{\mathrm{th}} = \infty$. This is clearly unphysical, because self-sputtering is known to occur.

Determination of sputtering yields from experiments and simulations

Physical sputtering (but also chemical erosion and CAPS) can be characterized by the yield Y, which is the flux ratio of eroded particles and incoming projectiles:

$$\Gamma^{\mathrm{in}} = \frac{\text{number of incoming particles}}{\text{area} \times \text{time}}\,, \tag{2.3}$$

$$\Gamma^{\mathrm{ero}} = \frac{\text{number of eroded particles}}{\text{area} \times \text{time}}\,, \tag{2.4}$$

$$Y = \frac{\Gamma^{\mathrm{ero}}}{\Gamma^{\mathrm{in}}}\,. \tag{2.5}$$

Sputtering yields depend on various parameters, such as the target temperature, material properties (such as the chemical composition in the interaction layer, crystal structure and surface morphology) and the projectile irradiation parameters (chemical species, impact energy and angle, flux and fluence).

The main focus here and in the following sections will be on Be physical sputtering by impact of D or Be (self-sputtering) with different energies and angles. Additionally, the influence of the D content in the surface interaction layer of the

Be target is taken into account, since Be is capable of efficiently trapping D at lattice vacancies [78–80], which leads to lower sputtering yields.

Experimentally, sputtering yields can be determined by measuring the amount of material removed from a target, which is exposed to a known incident fluence, by measuring the mass loss or thickness change (profilometry), collection of sputtered material, or spectroscopy on sputtered species [66]. For the irradiation to be well defined and diagnosed, such experiments are often carried out with ion beams or in linear plasma devices. For Be, the task is additionally challenging due to its biological hazard. Measurements of Be sputtering yields, e.g. for D and Be irradiation with various impact energies and angles, are available in literature from the ion accelerator at IPP Garching [165] and the linear divertor plasma simulator PISCES-B in San Diego [141]. Some of the data is shown in Figure 2.3.

Other measurements for Be are available from JET-ILW e.g. via spectroscopy [33, 35, 181]. A well-defined irradiation is more difficult to achieve in a tokamak than in ion beam or linear plasma experiments e.g. due to its 3D shaped wall geometry and plasma configuration. Therefore, the measured sputtering yields are typically averaged over a range of irradiation conditions. An additional difficulty in tokamaks is that in most cases, samples can only be removed during shutdowns between experimental campaigns, so that post-mortem analysis results are mostly not shot-resolved (although there are exceptions such as rotating collector probes). The great advantage of measurements in tokamaks, compared to ion beams and linear plasma devices, is that the experimental conditions are much closer to those in future reactors. For instance, samples exposed in linear devices often show a distinct surface morphology development (e.g. the growth of cones for Be [59]) which affect erosion, while no such effects were observed in tokamaks yet.

For computer simulations of sputtering yields, two most commonly used approaches are the binary-collision approximation (BCA) and molecular dynamics (MD). One of the advantages of BCA is the possibility to perform simulations on a macroscopic spatial and temporal scale due to its comparatively short calculation times (about four to five orders of magnitude below those for MD). However, the BCA assumptions are considered valid only at higher impact energies (typically above $\sim$30 eV [67]). MD is usually expected to give better results at low impact energies and is suitable for treating chemical erosion and CAPS. However, due to being computationally extensive, MD is restricted to simulation volumes and times in the order of about $\sim$1 nm^3 and $\sim$1 ps. This simulation scale is not always sufficient to follow the collision cascade, in particular if the projectile energy is high.

BCA assumes the collision cascade to be a sequence of independent elastic

binary collisions with target atoms, with energy loss occurring due to electronic stopping between the binary collisions [67]. An example of the interaction potentials assumed for the collisions is the Krypton-Carbon (Kr-C) potential[16], which is also used in the Eckstein fit formula presented below. Examples of BCA codes are (in chronological order of development) SRIM/TRIM [11, 216], TRIM.SP [10], TRIDYN [133] and SDTrimSP [135]. The last two are dynamic codes, which means they take into account the cumulative change (with in-depth resolution of the interaction layer) in material composition due to implantation and preferential sputtering during the bombardment [67].

TRIM.SP has been used for determining yields for a large variety of target-projectile combinations (including Be targets), impact energies and angles. The yields were used to obtain fit parameters for the Eckstein fit formula [66]. The most recent code version SDTrimSP was used together with MD for obtaining high and low estimates of Be sputtering yields using the assumptions of (1) elementary pure Be targets ('ERO-max') and (2) Be targets containing 50 % D in the interaction layer ('ERO-min'). In the latter case, the Be sputtering yields (for normal incidence) are about a factor of four lower than for pure Be targets [23] due to the dilution from the presence of D [13]. These yields were subsequently used for ERO modeling of JET Be limiter components erosion [21, 23, 24]. The modeling indicates that the low erosion estimate for 50 % D content gives a good agreement with experimental erosion measurements for plasma-wetted areas [35, 181].

In contrast to BCA codes, MD codes such as PARCAS [143] follow the time evolution of an entire system of particles under taking into account a full multi-body potential, such as the Brenner potential [32]. Using PARCAS, the previously mentioned experimental observation of BeD molecule release by CAPS was explained [15]. The MD modeling was subsequently refined with investigation of parameters which influence the sputtering yields, such as different Be interatomic potentials, target temperature and the D flux, fluence and surface content [13, 60, 116, 167]. Furthermore, the MD sputtering yields were verified with the ERO code via light emission [14, 16], with the latter publication showing a good match between simulated and experimental Be and BeD light emission intensities. Sputtering yields were also calculated by PARCAS for mixed Be-W surfaces under D and Be irradiation [114, 116].

However, due to being limited to short time scales, MD may give results that are unphysical for fusion reactor-like conditions. In the above-mentioned simulations,

[16]This potential is introduced in [211] and has the form of a screened Coulomb potential $V(r) = (Z_1 Z_2 e^2/r)\phi(r)$ with the screening function $\phi(r) = \sum_{i=1}^{3} C_i e^{-b_i r/a}$. The coefficients C_i and b_i are fitted using results from free-electron method calculations for the Kr-C interaction, which is shown to be well representative for a broad range of other element combinations as well.

at high surface temperatures the D surface content increases. This leads to release of larger BeD_x molecules, which is caused by the requirement of a high irradiation flux due to the limited time scale in MD simulations. More recently, the MD simulations for D impact on Be targets were combined with the object kinetic Monte Carlo (OKMC) method in a multi-scale modeling approach [168]. Due to being less restrained to short time scales, OKMC is able to calculate equilibrium D distribution profiles in the substrate under consideration of diffusion processes for D interstitials, vacancies etc. The results of coupled OKMC-MD simulations confirm the above-mentioned assumption of 50 % D content in the interaction layer, and also give better agreement with experimental results from JET and PISCES-B than plain MD simulations.

Eckstein fit formula

ERO calculates the physical sputtering yield $Y(E_{\mathrm{in}}, \theta_{\mathrm{in}})$ as a function of impact energy E_{in} and angle θ_{in} (see Figure 2.2(a)) of the projectile using the fit formula by Eckstein [65]. The fit parameters in the formula are obtained by fitting the

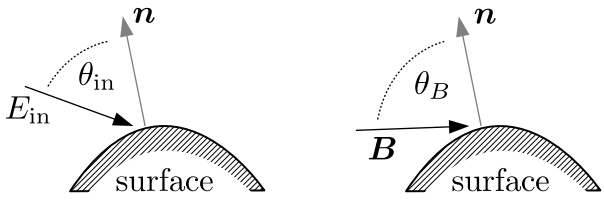


Figure 2.2: Definition of the (a) projectile incidence angle θ_{in} and (b) magnetic field inclination angle θ_B. The angles are defined relative to the surface normal $\boldsymbol{n}$.

experimental or simulated data for sputtering yields. The formula is comprised of a normal incidence factor and an angular factor,

$$Y(E_{\mathrm{in}}, \theta_{\mathrm{in}}) = Y(E_{\mathrm{in}}, 0) A(E_{\mathrm{in}}, \theta_{\mathrm{in}}) \,. \tag{2.6}$$

The normal incidence factor depends on four fit parameters $(\lambda, q, \mu, E_{\mathrm{th}})$ and is given by

$$Y(E_{\mathrm{in}}, 0) = q s_{\mathrm{n}}^{\text{Kr-C}}(\varepsilon_{\mathrm{L}}) \frac{(E_{\mathrm{in}}/E_{\mathrm{th}} - 1)^{\mu}}{\lambda/\omega - (E_{\mathrm{in}}/E_{\mathrm{th}} - 1)^{\mu}} \,, \tag{2.7}$$

with the nuclear stopping power for the Kr-C potential (see footnote 16)

$$s_{\mathrm{n}}^{\text{Kr-C}} = \frac{0.5 \ln(1 + 1.2288 \varepsilon_{\mathrm{L}})}{\omega} \tag{2.8}$$

using the denotation

$$\omega = \varepsilon_{\mathrm{L}} + 0.1728\sqrt{\varepsilon_{\mathrm{L}}} + 0.008\varepsilon_{\mathrm{L}}^{0.1504}\,, \tag{2.9}$$

the reduced energy

$$\varepsilon_{\mathrm{L}} = E_{\mathrm{in}}\frac{M_2}{M_1+M_2}\frac{a_{\mathrm{L}}}{Z_1 Z_2 e^2} \tag{2.10}$$

(with M_1, M_2 being the masses and Z_1 and Z_2 the atomic numbers of projectile and target atoms, respectively), and the Lindhard screening length

$$a_{\mathrm{L}} = \left(\frac{9\pi^2}{128}\right)^{1/3} a_{\mathrm{B}}\left(Z_1^{2/3}+Z_2^{2/3}\right)^{-1/2}\,,\ a_{\mathrm{B}} = 0.052\,917\,7\,\mathrm{nm} \tag{2.11}$$

where a_{B} is the Bohr radius. Note that the sputtering yield in equation (2.7) becomes zero for $E = E_{\mathrm{th}}$, therefore the fit parameter E_{th} has the physical meaning of the sputtering threshold energy. In ERO simulations, the previously mentioned more crude estimation (2.2) is replaced by the value of E_{th} obtained by fitting multiple MD/BCA results.

Figure 2.3 shows the sputtering yield $Y_{\mathrm{Be}\leftarrow\mathrm{D}}(E_{\mathrm{in}}, 0)$ (normal incidence) used in ERO. It is obtained using the ERO-min and ERO-max assumptions as mentioned in the previous subsection. Figure 2.3(a) shows the MD (PARCAS) and BCA (SDTrimSP) simulated data points underlying the ERO-min and ERO-max fits. In the relevant energy range, ERO-max is higher than ERO-min by at least a factor of four. Figure 2.3(b) shows experimental data from [8, 141] for comparison. It illustrates how sputtering yields vary over at least three orders of magnitude with the incidence energy, with a maximum at about 200 eV. One can also see that experimental measurements of sputtering yields for the same impact energy are scattered within two orders of magnitude.

Both plots in Figure 2.3 show for comparison the Eckstein fit based on TRIM.SP data from [66], which mostly lies between ERO-min and ERO-max. The upper half of Table 2.1 shows the corresponding Eckstein fit parameters in $Y_{\mathrm{Be}\leftarrow\mathrm{D}}(E_{\mathrm{in}}, 0)$ for ERO-min, ERO-max and [66].

The angular factor in the Eckstein fit formula depends on three energy-dependent fit parameters (b, c, f) and is given by

$$A(E_{\mathrm{in}}, \theta_{\mathrm{in}}) = \zeta^{-f}\exp\left(b\left\{1-1/\zeta\right\}\right)\,, \tag{2.12}$$

$$\zeta = \cos\left[\left(\frac{\theta_{\mathrm{in}}}{\theta^*_{\mathrm{in}}}\frac{\pi}{2}\right)^c\right]\,, \tag{2.13}$$

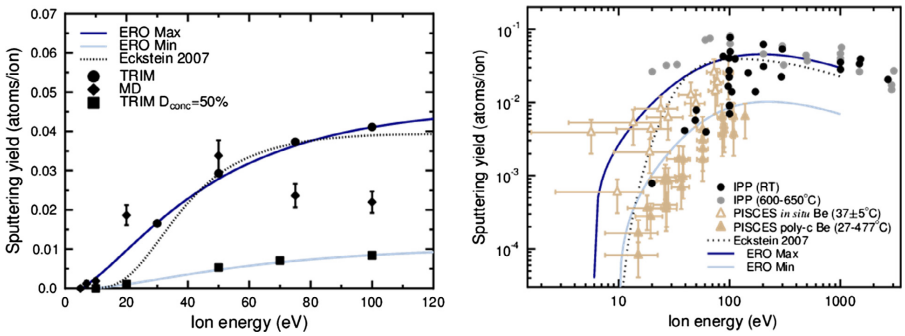


Figure 2.3: Normal incidence physical sputtering yields $Y_{\mathrm{Be}\leftarrow\mathrm{D}}(E_{\mathrm{in}}, 0)$, with comparison between data from simulations, experiments and Eckstein formula (2.7). The dotted line shows the Eckstein formula with fit parameters deduced from TRIM.SP simulations as given in [67], the solid lines for fit parameters determined from various simulations with low and high estimates ERO-min (50 % D surface content) and ERO-max (no D surface content). (a) Simulated data from SDTrimSP (BCA) and PARCAS (MD) used for deducing ERO-min and ERO-max fit parameters. (b) Comparison of ERO-min and ERO-max with experimental data. Image taken from [23].

$Y(E_{\mathrm{in}}, 0)$		Eckstein 2007 [66]	ERO-min [23]	ERO-max [23]
Be ← D	λ	1.7575	1.1874	0.8059
	q	0.1044	0.0323	0.1386
	μ	1.9906	0.7558	0.6206
	E_{th} [eV]	9.5059	9.9686	5.9907
Be ← Be	λ	2.0334	1.4486	2.0334
	q	0.8241	0.1865	0.8241
	μ	1.3437	0.9113	0.8241
	E_{th} [eV]	16.9689	19.5634	16.9689

Table 2.1: Eckstein normal incidence fit parameters for $Y_{\mathrm{Be}\leftarrow\mathrm{D}}(E_{\mathrm{in}}, 0)$ and $Y_{\mathrm{Be}\leftarrow\mathrm{Be}}(E_{\mathrm{in}}, 0)$ deduced from TRIM.SP calculations (see Table 1 in [66]) and SDTrimSP and PARCAS calculations (see Table 1 in [23]).

$$\theta^*_{\mathrm{in}} = \pi - \arccos\sqrt{\frac{1}{1 + E_{\mathrm{in}}/E_{\mathrm{SB}}}}\,. \tag{2.14}$$

The angular part parameters in $Y_{\mathrm{Be}\leftarrow\mathrm{D}}(E_{\mathrm{in}}, \theta_{\mathrm{in}})$ for ERO-max and ERO-min are fitted based on SDTrimSP simulations only [23]. Obtaining impact angle fit parameters requires a scan in both impact angle and energy, which demands a lot of CPU time and effort with the MD approach, therefore such a scan is not available yet. The top half of Figure 2.4 shows the energy and angle dependence of $Y(E_{\mathrm{in}}, \theta_{\mathrm{in}})$ (in the energy range relevant for the modeling in chapter 4) for D impact under the ERO-max and ERO-min assumptions. One can see that the yields are increasing with the angle relative to the surface normal, with a maximum

at about $\theta_{\text{in}} \approx 70°$ for ERO-max and $\theta_{\text{in}} \approx 80°$ for ERO-min. In general, the sputtering yield for a target-projectile combination is about an order of magnitude larger at its maximum compared to normal incidence. However, the yield drops sharply for larger impact angles beyond the maximum [8].

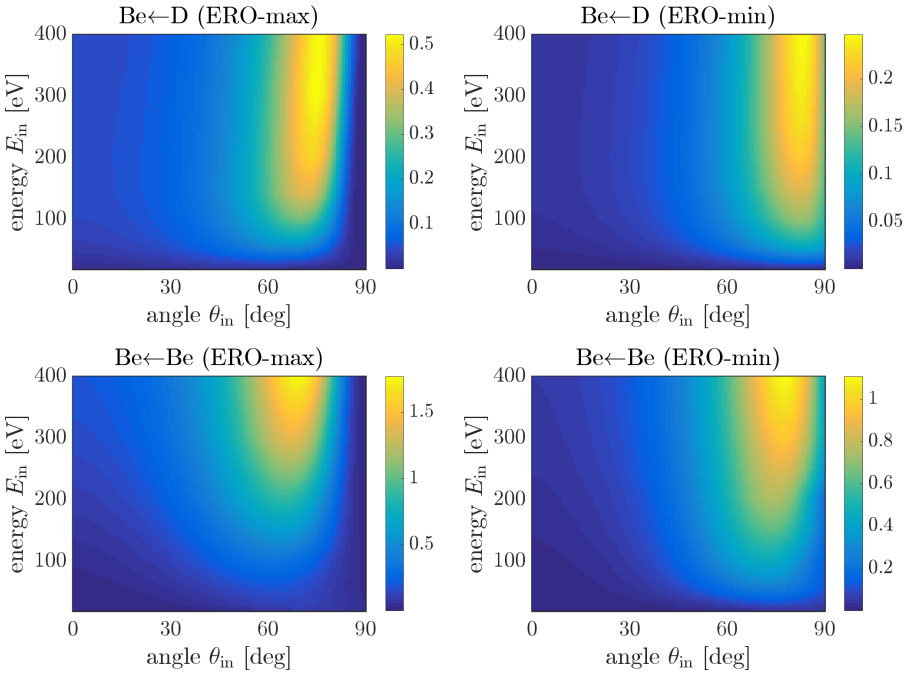


Figure 2.4: Sputtering yields $Y(E_{\text{in}}, \theta_{\text{in}})$ as a function of impact energy E_{in} and impact angle θ_{in}: Be target, D and Be projectiles under ERO-min and ERO-max assumptions.

Be self-sputtering is also considered in the modeling presented in chapter 4. The corresponding fit parameters have been obtained in a similar way as those for D impact [23]. The bottom half of Table 2.1 shows the corresponding Eckstein fit parameters in $Y_{\text{Be}\leftarrow\text{Be}}(E_{\text{in}}, 0)$ for ERO-min, ERO-max and [66]. The bottom half of Figure 2.4 shows the sputtering yields for Be impact obtained with the fit parameters. One can see that the yields are generally higher for Be impact than for D impact. Therefore, self-sputtering is an important erosion mechanism if a certain amount of Be impurities is present in the plasma.

Pre-calculated effective yields

Technically, ERO calculates erosion from two contributions, namely due to the followed test particles (or more precisely, the fraction of test particles that hits a wall element) and due to the background plasma impact. The first contribution

$\Gamma^{\mathrm{ero,tp}}$ is calculated, just after the transport simulation, for each test particle trajectory i ending up on a PFC surface cell with the area $\mathrm{d}A$ (determined by the user-defined surface cell resolution, see section 3.3). The impact energy and angle (E_i, θ_i) are known, therefore the Eckstein fit formula (2.6) can be used directly:

$$\Gamma^{\mathrm{ero,tp}} = \frac{1}{\mathrm{d}A} \sum_i Y(E_i, \theta_i) \dot{N}_i \,, \tag{2.15}$$

where $\dot{N}_i$ is the number of real atoms per second 'carried' by a test particle.

The second contribution $\Gamma^{\mathrm{ero,bg}}$ has to be calculated before the transport simulation in order to obtain an ensemble of eroded test particles in first instance. The energy and angle distribution of impinging particles is not known for the background plasma, so the Eckstein fit formula cannot be applied offhandedly. Therefore, ERO either assumes a reasonable (e.g. mean) constant impact energy and angle, or utilizes analytically or numerically pre-calculated distributions in order to obtain an effective yield Y_{eff}. For the results presented in this thesis the analytical approach is used, with the effective yield $Y_{\mathrm{eff}}(T_\mathrm{e}, \theta_B)$ being parametrized by the electron temperature T_e and magnetic inclination angle θ_B relative to the surface normal (see Figure 2.2(b)). The erosion flux then becomes

$$\Gamma^{\mathrm{ero,bg}} = \sum_\alpha Y_{\mathrm{eff}}(T_\mathrm{e}, \theta_B) \Gamma_\alpha^{\mathrm{in}} \,, \tag{2.16}$$

where $\Gamma_\alpha^{\mathrm{in}}$ is the impinging ion flux for the background species α (e.g. D^+ or Be^{3+}). It is calculated as

$$\Gamma_\alpha^{\mathrm{in}} = n_\mathrm{e} c_{\mathrm{s},\alpha} \cos(\theta_B) f_\alpha S \,, \tag{2.17}$$

where

$$c_{\mathrm{s},\alpha} = \sqrt{\frac{k_\mathrm{B}(T_\mathrm{e} + T_\mathrm{i})}{m_\alpha}} \tag{2.18}$$

is the ion sound velocity of background ions of mass m_α, f_α is the relative concentration of this background ion species in the flux, and S is a shadowing factor between zero and one, which accounts for the fact that less background ions are reaching magnetically shadowed surface areas. The procedure for calculating the shadowing factor is discussed in section 4.2.3.

Figure 2.5 shows the effective yields $Y_{\mathrm{eff}}(T_\mathrm{e}, \theta_B)$ for the electron temperature range relevant for the modeling in chapter 4. The effective yields are obtained by averaging the yields from Figure 2.4, using impact energy and angle distributions that are obtained by the procedure described in [28].

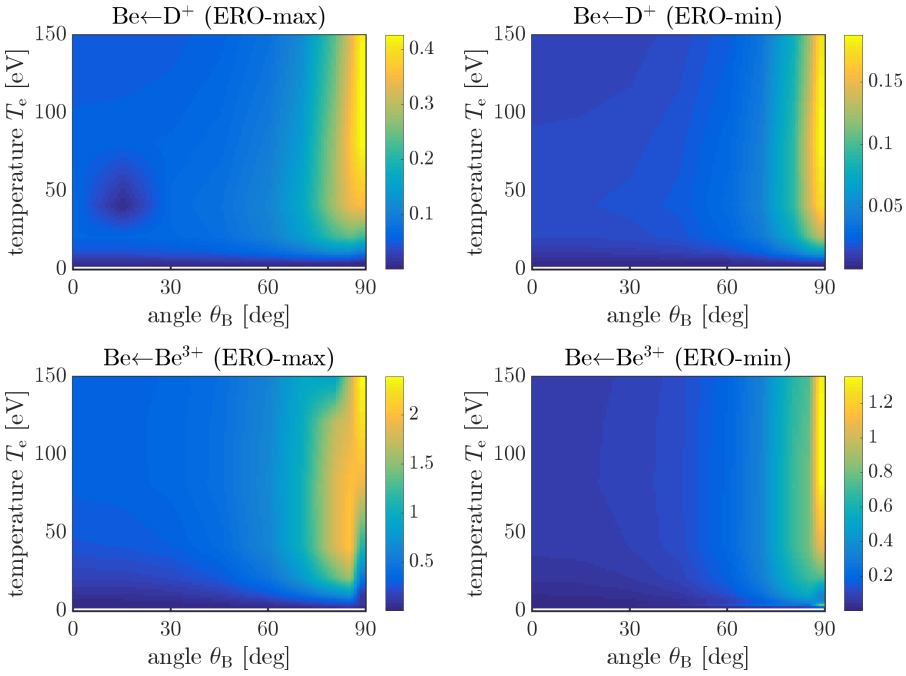


Figure 2.5: Effective sputtering yields $Y_{\mathrm{eff}}(T_{\mathrm{e}}, \theta_B)$ as a function of electron temperature T_{e} and magnetic field angle θ_B: Be target, D^+ and Be^{3+} projectiles under ERO-min and ERO-max assumptions.

Energy and angle distributions of sputtered neutrals

Sputtered particles are known to leave the surface as neutrals [102]. Their emission energy E can be well approximated by the Thompson-Sigmund distribution [68, 196], which may be written as

$$\frac{\mathrm{d}Y}{\mathrm{d}E} \propto \frac{E}{(E + E_{\mathrm{SB}})^{1+\alpha}} \tag{2.19}$$

with the fit parameter α. For large ion impact energies E_{in} one finds $\alpha \approx 2$ [8]. The maximum of the distribution is then at $E = E_{\mathrm{SB}}/2$ and the distribution drops as $1/E^2$. For lower ion impact energies $E_{\mathrm{in}} \lesssim 600\,\mathrm{eV}$ [37] one finds $\alpha > 2$, the maximum shifts closer to zero and the distribution falls off more steeply. From equation (2.1) it also follows that the distribution has a cut-off at the maximum transferable energy

$$E^{\mathrm{max}} = E_{\mathrm{in}}\gamma(1-\gamma) - E_{\mathrm{SB}} \tag{2.20}$$

that depends on the impact energy. For simplicity, ERO usually neglects this dependence when sampling the distribution[17] and instead assumes a certain constant value for either the impact energy (e.g. $E_{\text{in}} \approx k_{\text{B}}T_{\text{e}} + E_{\text{sheath}}$, with the sheath potential E_{sheath}) or directly for the cut-off energy $E^{\max}$. A parameter study on the effects of $E^{\max}$ on the example of W erosion and prompt deposition is found in [107].

The azimuthal emission angle ϕ of sputtered particles is in most cases symmetrically distributed. The polar emission angle θ (relative to the surface normal) is often found to be distributed by an over-cosine law [8]

$$\frac{\mathrm{d}Y}{\mathrm{d}\Omega} \propto \cos^{y}(\theta)\,. \tag{2.21}$$

where $y \geq 1$ is a fit parameter. However, a variety of other distributions can be found in literature. E.g. for ERO modeling for the linear plasma devices PISCES-B [20] and PSI-2 [68], an over-cosine distribution with a cut-off at low angles and a 'butterfly-like' distribution, respectively, were found to give better agreement with experiment than equation (2.21).

Figure 2.6 shows MD simulation results for the distributions of Be and BeD particles eroded under D impact. One can see that the cosine and Thompson distributions are sensible approximations for the angle and energy distributions of the eroded particles.

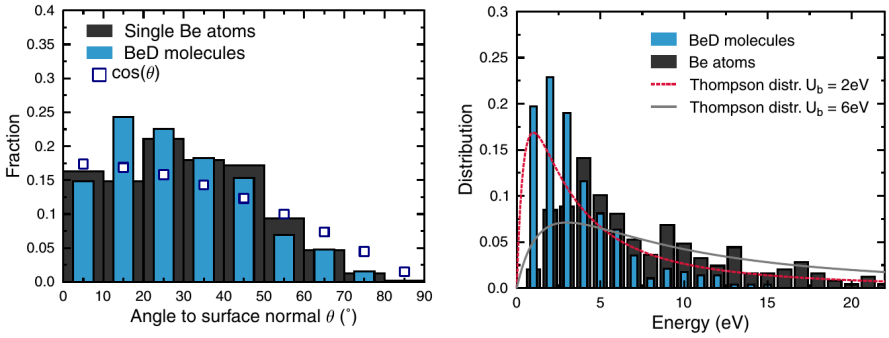


Figure 2.6: MD simulation results for the distributions of Be and BeD particles eroded under D impact. Left: the distribution of the angle θ relative to the surface normal can be approximated by a cosine function. Right: the distribution of the energy E can be approximated by the Thompson distribution (2.19) with an appropriate surface binding energy E_{SB} (labeled U_{b} in the figure). Image from [16].

[17] For random sampling of distributions, ERO generally uses the inversion method, see Theorem 2.1 in [52].

2.2.2 Deposition

A detailed treatment of deposition mechanisms is important for PWI studies but is not the focus of the first ERO2.0 application in chapter 4, therefore this section will provide only a very brief overview. A more thorough overview can be found e.g. in [102].

ERO1.0 acknowledges the fact that a particle impinging on the surface can be reflected with a certain probability. This probability is expressed by the reflection coefficient R ($0 \leq R \leq 1$), which is highly dependent on the impact energy and angle. ERO1.0 obtains R mostly from BCA simulations (e.g. SDTrimSP). All impinging particles that are not reflected are considered as deposited by implantation.

Apart from the implantation of energetic particles, thermal particles can also be adsorbed at a surface either by physisorption (by van der Waals forces with a strength of typically below 0.5 eV) or chemisorption (by electronic bonds). Adsorbed species can subsequently be released (desorbed) either thermally or by particle or photon impact. These adsorption and desorption processes are not implemented in any ERO version yet.

Deposition is important not only in the general context of material migration and mixing, but especially in the context of fuel retention. In addition to implantation and adsorption, fuel atoms can be efficiently codeposited with energetic impurity atoms such as C or Be. If the impurity redeposition occurs in different regions than the erosion regions, codeposited layers are formed. At higher surface temperatures, deposited fuel atoms can subsequently diffuse further into the bulk.

2.3 Impurity transport

This section is concerned with equations and algorithms used in ERO for simulating the impurity transport. The Boris method for numerically calculating the trajectory of a single particle with a charge q is described. Equations for the sheath potential and electric field are presented. The MC method for solving the kinetic equation considering inter-particle collisions is described: for this, the report [161] about the kinetic test particle transport code DORIS for TEXTOR applications (see also [160]) is abridged. Finally, the treatment of elementary atomic processes for particles (e.g. ionization and recombination) is discussed.

2.3.1 Motion of a single charged particle

In the presence of electro-magnetic fields $\boldsymbol{E}$ and $\boldsymbol{B}$, a single particle with charge q and mass m is accelerated by the Lorentz force:

$$\dot{\boldsymbol{r}} = \boldsymbol{v}\,, \tag{2.22}$$

$$\dot{\boldsymbol{v}} = \frac{q}{m}\left(\boldsymbol{E} + \boldsymbol{v} \times \boldsymbol{B}\right)\,. \tag{2.23}$$

The magnetic contribution of the Lorentz force, which is directed perpendicular to the particle's velocity, leads to a spiraling motion of the particle along magnetic field lines (gyromotion). The gyromotion can be described as a superposition $\boldsymbol{r}(t) = \boldsymbol{R(t)} + \boldsymbol{\rho}(t)$ of a very fast gyration $\boldsymbol{\rho}(t)$, with the cyclotron frequency [208, p. 40]

$$\omega_\mathrm{c} = \frac{|q|B}{m} \tag{2.24}$$

and Larmor radius

$$r_\mathrm{g} = \frac{v_\perp}{\omega_\mathrm{c}} = \frac{m v_\perp}{|q|B}\,, \tag{2.25}$$

around a point $\boldsymbol{R}(t)$ called the guiding center, and a slower drift of the same point $\boldsymbol{R}(t)$. The drift is often of predominant interest in plasma theory, wherefore the gyration is averaged out (guiding center approximation, GCA).

Since the gyromotion, which occurs on time scales in the order of $T_\mathrm{c} = 2\pi/\omega_\mathrm{c}$, is averaged out, GCA allows in principle to use time steps $\Delta t \gg T_\mathrm{c}$. However, there are some restrictions to this. First, the particle step length still need to be small compared to characteristic lengths for e.g. plasma parameter variation, drifts, or atomic processes such as ionization. Second, the GCA approximation is well applicable only if the Larmor radius is much smaller than the electro-magnetic field gradient length scale, which is not the case e.g. in the sheath, where very strong electric field gradients occur. GCA is therefore not well suited for ERO, which requires full orbit resolution (especially for particles close to the surface) in order to obtain accurate particle impact energies and angles. Instead, the Boris algorithm is used to resolve the gyromotion, which will now be described.

ERO calculates a particle's phase-space trajectory $(\boldsymbol{r}_{k+1}, \boldsymbol{v}_{k+1})$ in discrete time steps $t_k = k\Delta t$:

$$\frac{\boldsymbol{r}_{k+1} - \boldsymbol{r}_k}{\Delta t} = \boldsymbol{v}_{k+1}\,, \tag{2.26}$$

$$\frac{\boldsymbol{v}_{k+1} - \boldsymbol{v}_k}{\Delta t} = \frac{q}{m}\left(\boldsymbol{E}_k + \frac{(\boldsymbol{v}_{k+1} + \boldsymbol{v}_k) \times \boldsymbol{B}_k}{2}\right)\,, \tag{2.27}$$

with $\boldsymbol{r}_k = \boldsymbol{r}(t_k)$, $\boldsymbol{v}_k = \boldsymbol{v}(t_k - \Delta t/2)$, $\boldsymbol{E}_k = \boldsymbol{E}(\boldsymbol{r}_k)$ and $\boldsymbol{B}_k = \boldsymbol{B}(\boldsymbol{r}_k)$. Equation (2.27) is implicit, since $\boldsymbol{v}_{k+1}$ appears on both left and right hand side of the equation. An efficient way of solving the above equations is provided by the Boris method [19, 155]:

$$\boldsymbol{v}^- = \boldsymbol{v}_k + \frac{q}{m}\boldsymbol{E}_k\frac{\Delta t}{2}\,, \tag{2.28}$$

$$\frac{\boldsymbol{v}^+ - \boldsymbol{v}^-}{\Delta t} = \frac{q}{2m}\left(\boldsymbol{v}^+ + \boldsymbol{v}^-\right) \times \boldsymbol{B}_k\,, \tag{2.29}$$

$$\boldsymbol{v}_{k+1} = \boldsymbol{v}^+ + \frac{q}{m}\boldsymbol{E}_k\frac{\Delta t}{2}\,. \tag{2.30}$$

$\boldsymbol{v}^+$ is constructed geometrically through a rotation of $\boldsymbol{v}^-$. The equations (2.28)-(2.30) replace (2.27), so that the algorithm becomes explicit. As discussed in [155], the Boris method is the de-facto standard for full-orbit simulations of magnetized plasmas due to its excellent accuracy even for an arbitrarily large number of time steps. In particular, the method conserves energy exactly when there is no electric field. In the presence of an electric field, the numerical error is bounded. The Boris method is able to correctly reproduce effects present in tokamaks such as banana orbits [208, p. 130] and the Ware pinch effect [204], in contrast to e.g. the 4th order Runge-Kutta method, which fails at larger numbers of time steps due to accumulation of numerical error.

The long-term accuracy of the Boris method is important for the transition from ERO1.0 to ERO2.0, where particle trajectories can become orders of magnitude longer. Figure 2.7 demonstrates this accuracy on the example of a single W^+ ion trajectory calculated with ERO2.0 (without taking into account collisions or atomic processes, which are further discussed in the following sections). The magnetic field configuration was taken from the JET EFIT code for a limiter pulse with the JET pulse number (JPN) #80319 at $t = 50\,\mathrm{s}$. The particle trajectory starts at the outer midplane at a minor radius $r = 77\,\mathrm{cm}$, a velocity $v_0 = 10^4\,\mathrm{m/s}$ and a varying pitch angle $\alpha = v_{\parallel 0}/v_{\perp 0}$. From theory [208, p. 128], a transition is expected between passing and banana orbits (due to the magnetic mirror effect at the high-field side) for pitch angles below a threshold

$$\alpha^* = \sqrt{\frac{2r}{R_0 - r}}\,, \tag{2.31}$$

where $R_0 = 3\,\mathrm{m}$ is the major radius of the JET tokamak. The banana orbit has a half width [208, p. 130]

$$\delta r = \frac{v_{\parallel 0} m_i}{q_i B_\theta}\,, \tag{2.32}$$

where B_θ is the poloidal magnetic field. From inserting the starting condition r into equation (2.31), the threshold $\alpha^* \approx 0.85$ is obtained, and after inserting $v_{\parallel 0} = v_0/\sqrt{1+(\alpha^*)^2}$ into equation (2.32), the width $\delta r \approx 3\,\mathrm{cm}$ is obtained. The ERO2.0 results ($\alpha^* \approx 0.87$, $\delta r \approx 3\,\mathrm{cm}$) are in good agreement with those values. Furthermore, even after calculating 100 poloidal rotations of the particle, the orbits remain closed and no energy error accumulation is observed.

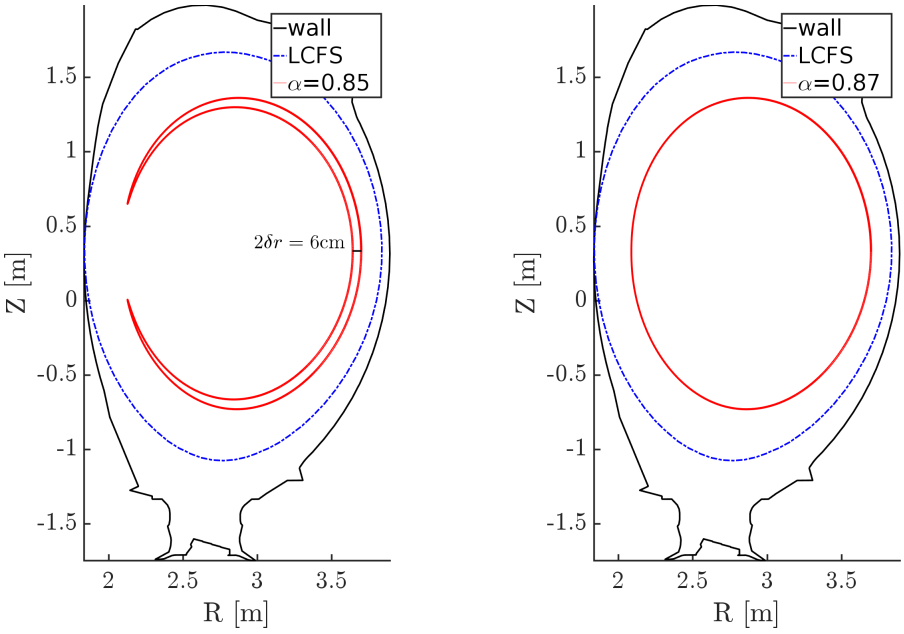


Figure 2.7: (a) Banana orbit and (b) passing orbit of a single W^+ ion starting with a pitch angle $\alpha = v_{\parallel 0}/v_{\perp 0}$ at the outer midplane in the JET tokamak, calculated with ERO2.0.

2.3.2 The sheath potential and electric field

The Debye sheath (DS) is a thin plasma region at the interface to the surface of a material in contact with the plasma, and has a huge influence on the motion of charged particles crossing that region due to its strong electric fields. The DS has a thickness of several Debye lengths

$$r_\mathrm{d} = \sqrt{\frac{\varepsilon_0 k_\mathrm{B} T_\mathrm{e}}{n_\mathrm{e} e^2}}\,. \tag{2.33}$$

For typical fusion reactor plasma edge conditions of $T_\mathrm{e} \sim 20\,\mathrm{eV}$ and $n_\mathrm{e} \sim 10^{19}\,\mathrm{m}^{-3}$, the Debye length is in the order of $r_\mathrm{d} \sim 10\,\mu\mathrm{m}$. The DS arises because the electrons

of the plasma are more mobile than the ions and therefore hit the wall earlier. This leads to the build-up of a negative charge on the surface, which accelerates the ions in its direction [182, p. 28].

In the presence of a magnetic field, an additional sheath called the magnetic pre-sheath (MPS) or Chodura sheath (CS) is formed. Its thickness, which is of the order of a few Larmor radii r_g, and its magnitude depend on the magnetic field angle θ_B relative to the surface normal. The floating potential obtained by adding up the total DS and MPS potential drops is however independent of θ_B [184, p. 79]:

$$\phi_\mathrm{w} = 0.5\frac{k_\mathrm{B}T_\mathrm{e}}{e}\ln\left[2\pi\frac{m_\mathrm{e}}{m_\mathrm{i}}\left(1+\frac{T_\mathrm{i}}{T_\mathrm{e}}\right)\right], \tag{2.34}$$

where the potential $\phi = 0$ is assumed at the sheath entrance.[18] For hydrogenic plasmas the floating potential is about $\phi_\mathrm{w} \approx -3k_\mathrm{B}T_\mathrm{e}$ [182, p. 79]. E.g. for $T_\mathrm{e} = 30\,\mathrm{eV}$ a singly charged particle would obtain an energy of $\sim$100 eV after traversing the sheath towards the surface, which shows the importance of the sheath for the impact energy and thus for the sputtering yield.

Calculating the spatial dependence of the potential drop can be achieved by PIC simulations [44] or by numerically solving the MHD equations [184]. ERO1.0 uses an analytic formula approximation provided by Brooks [38] (or, alternatively, pre-calculated PIC results [106]). Recently, a more accurate analytic formula was derived which shows a good agreement with numerical MHD and PIC calculation results [26]. This formula was used in more recent ERO1.0 simulations [21] and the ERO2.0 simulations in chapter 4:

$$\phi(x) = \begin{cases} \phi_\mathrm{w} + Q\left(1-\exp(-ax/r_\mathrm{d})\right) & \text{if } x \le x_\mathrm{mps} \\ \phi_\mathrm{mps}\exp\left[-2(x-x_\mathrm{mps})/L_\mathrm{mps}\right] & \text{if } x > x_\mathrm{mps} \end{cases} \tag{2.35}$$

where x is the distance to the surface along the surface normal,

$$L_\mathrm{mps} = kr_\mathrm{g}\sin\theta_B \tag{2.36}$$

(k=2–3 [44, 184]) is the thickness of the MPS,

$$x_\mathrm{mps} = -\frac{1}{a}\ln\left(\frac{\phi_\mathrm{w}-\phi_\mathrm{mps}+Q}{Q}\right) \tag{2.37}$$

[18] The effect of electron-induced secondary electron emission (SEE) may be included in equation (2.34) by multiplying the logarithm argument with a factor $(1-\delta_\mathrm{e})^{-2}$, where δ_e is the SEE coefficient. Unless $\delta_\mathrm{e} \ll 1$, SEE will lead to a lower potential drop ϕ_w. More details are found e.g. in [182, pp. 79-80, 114-116, 646-647].

is the coordinate of the DS/MPS interface, and

$$\phi_{\mathrm{mps}} = \frac{k_{\mathrm{B}} T_{\mathrm{e}}}{e} \ln \cos \theta_B \tag{2.38}$$

is the corresponding potential at the DS/MPS interface. The parameters a and Q are derived in [26]. They depend on the plasma parameters n_{e}, T_{e}, and T_{i} at the sheath entrance, the ion mass m_{i}, and the magnetic field strength B and angle θ_B. The MPS vanishes ($\phi_{\mathrm{mps}} = 0$, $L_{\mathrm{mps}} = 0$, $x_{\mathrm{mps}} = \infty$) for normal magnetic field $\theta_B = 0$. In that case, $Q = \phi_{\mathrm{w}}$ and the entire potential drop occurs in the DS:

$$\phi(x) = \phi_{\mathrm{w}} \exp(-ax/r_{\mathrm{d}}) \,. \tag{2.39}$$

On the other hand, the DS vanishes ($\phi_{\mathrm{mps}} = \phi_{\mathrm{w}}$, $L_{\mathrm{mps}} \approx k r_{\mathrm{g}}$, $x_{\mathrm{mps}} = 0$) for shallow magnetic field angles where θ_B exceeds a critical value

$$\theta_B^* = \arccos \left(\sqrt{\frac{2\pi m_{\mathrm{e}}}{m_{\mathrm{i}}} \left(1 + \frac{T_{\mathrm{i}}}{T_{\mathrm{e}}} \right)} \right) \tag{2.40}$$

(e.g. $\theta_B^* = 86.65°$ for a deuterium plasma with $T_{\mathrm{i}} = T_{\mathrm{e}}$ [184]). Then the entire potential drop occurs in the MPS:

$$\phi(x) = \phi_{\mathrm{w}} \exp\left(-2x/L_{\mathrm{mps}}\right) \,. \tag{2.41}$$

Note that the case of a surface without an externally applied bias voltage is considered here. The case for biased surfaces is discussed e.g. in [27].

Figure 2.8 shows the potential distribution according to equation (2.35) for different magnetic field angles. The sheath electric field is obtained simply by taking the derivative of the potential $E(x) = -\mathrm{d}\phi(x)/\mathrm{d}x$. Note that the maximum angle considered in the figure is $\theta_B = 88°$. This is because at extremely shallow angles ($\theta_B \gtrsim 89°$ for a deuterium plasma), "the ions reach the wall faster than electrons because of their larger Larmor radius and as a result a more complex sheath arises" [184]. This fact is neither accounted for in the model at hand, nor is it relevant for the JET simulations in chapter 4, where such extremely shallow angles are not encountered (see Figure 4.11(c)).

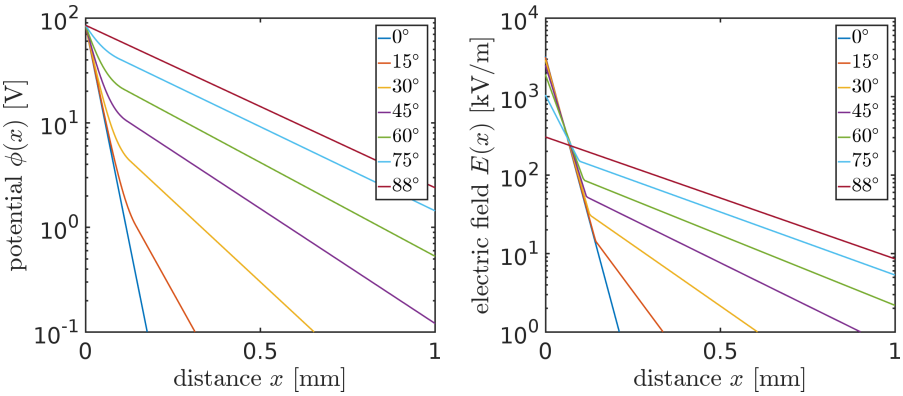


Figure 2.8: Distribution of electric potential and field in the Debye sheath (DS) and magnetic pre-sheath (MPS) according to equations (2.35), (2.39) and (2.41). The curves were calculated for a deuterium plasma with $n_\mathrm{e} = 10^{19}\,\mathrm{m}^{-3}$, $T_\mathrm{e} = T_\mathrm{i} = 30\,\mathrm{eV}$ at the sheath entrance, and a magnetic field $B = 4\,\mathrm{T}$ with different angles θ_B relative to the surface normal. The kink occurs at the DS/MPS interface coordinate x_mps.

2.3.3 Monte-Carlo simulation of collective transport

Kinetic equation and Fokker-Planck equation

In section 2.3.1, the motion of a single charged particle in the presence of electromagnetic fields was described. However, the description of transport in a plasma requires to take into account the interaction between its charged particles (mainly due to the Coulomb force). Due to the large number ($\sim 10^{23}$) of particles involved even if only a small part of a fusion plasma is considered, it is neither feasible nor desirable to solve the coupled system of their equations of motion in a microscopic approach. To reduce the complexity of the problem, a common approach in plasma theory is kinetic theory (see e.g. [12, pp. 27-28] for an overview also on other approaches, such as fluid theory). This approach involves a statistical description of the particles via distribution functions.

If the index α denotes a certain particle species (e.g. Be^+), the distribution function $f_\alpha(\boldsymbol{r}, \boldsymbol{v}, t)$ describes the probability to find a particle of this species in a phase space element $\mathrm{d}^3r\mathrm{d}^3v$ at time t. The interaction between particles of species α and β is described by a collision term $C_{\alpha\beta}(f_\alpha, f_\beta)$. The temporal and spatial evolution of the distribution function is then given by the kinetic equation (Boltzmann equation) [161, p. 7]

$$\frac{\partial f_\alpha}{\partial t} + \frac{\partial}{\partial \boldsymbol{r}}(\boldsymbol{v} f_\alpha) + \frac{\partial}{\partial \boldsymbol{v}}(\dot{\boldsymbol{v}} f_\alpha) = \sum_{\alpha\beta} C_{\alpha\beta}(f_\alpha, f_\beta)\,. \tag{2.42}$$

ERO calculates $C_{\alpha\beta}$ for impurities using the test particle approximation. This approximation assumes the concentration of impurities (test particles) to be low enough that they do not influence the distribution of the 'background' main plasma species (e.g. deuterium ions). Furthermore, the test particles do not interact with each other and do not affect the electro-magnetic fields. This allows to decouple the system of kinetic equations (2.42). In the following, it is assumed (without loss of generality) that the background plasma is composed of a single species denoted by the index b, which allows to drop the Greek indices α and β. The kinetic equation for a test particle species then becomes

$$\frac{\partial f}{\partial t} + \frac{\partial}{\partial \boldsymbol{r}}(\boldsymbol{v} f) + \frac{\partial}{\partial \boldsymbol{v}}(\dot{\boldsymbol{v}} f) = C(f, f_b)\,. \tag{2.43}$$

In the case of small angle scattering, which dominates in the plasma, the collision term may be written as a Fokker-Planck term [161, p. 8]

$$C(f, f_b) = -\sum_i \frac{\partial}{\partial v_i}(K_i f) + \frac{1}{2}\sum_{i,j} \frac{\partial^2}{\partial v_i \partial v_j}\left(D_{ij} f\right)\,, \tag{2.44}$$

where K_i and D_{ij} are components of the so-called drift vector and diffusion tensor. They are functionals of the background plasma distribution f_b:

$$K_i = \left(1 + \frac{m}{m_b}\right) \Lambda \frac{\partial \phi(\boldsymbol{v})}{\partial v_i}\,, \tag{2.45}$$

$$D_{ij} = \Lambda \frac{\partial^2 \psi(\boldsymbol{v})}{\partial v_i \partial v_j}\,. \tag{2.46}$$

Here, the Trubnikov potentials ϕ and ψ are defined as

$$\phi(\boldsymbol{v}) = \int \mathrm{d}\boldsymbol{v}' \frac{f_b(\boldsymbol{v}')}{|\boldsymbol{v} - \boldsymbol{v}'|}\,, \tag{2.47}$$

$$\psi(\boldsymbol{v}) = \int \mathrm{d}\boldsymbol{v}' |\boldsymbol{v} - \boldsymbol{v}'| f_b(\boldsymbol{v}')\,. \tag{2.48}$$

The numerical constant Λ is given by

$$\Lambda = \lambda \frac{Z^2 Z_b^2 e^4}{4\pi \varepsilon_0^2 m^2} n_b\,, \tag{2.49}$$

where λ is the Coulomb logarithm (see e.g. [92]), ε_0 is the dielectric constant, Z, Z_b, m and m_b are the charge states and masses of test and background ions, and n_b is the density of background ions. By inserting the collision term (2.44) into the

kinetic equation (2.42) and rearranging the terms, the kinetic equation is brought into the form of a Fokker-Planck equation:

$$\frac{\partial f}{\partial t} = -\sum_i \frac{\partial}{\partial r_i}(v_i f) - \sum_i \frac{\partial}{\partial v_i}\left[(\dot{v}_i + K_i) f\right] + \frac{1}{2}\sum_{i,j} \frac{\partial^2}{\partial v_i \partial v_j}\left(D_{ij} f\right) . \tag{2.50}$$

The Fokker-Planck form allows a particularly simple MC procedure for solving the kinetic equation, as will be shown below. But beforehand, an explicit form of the drift and diffusion coefficients will be given in the following subsection.

Drift and diffusion coefficients for Coulomb interaction with a Maxwellian background plasma

For calculating the coefficients K_i and D_{ij} in equation (2.44), the background plasma distribution f_b is assumed to be a Maxwellian shifted to the background flow velocity $\boldsymbol{v}_b$ and with ion temperature T_b [161, p. 61],

$$f_b(\boldsymbol{v}) = \left(\frac{m_b}{2\pi k_\mathrm{B} T_b}\right)^{3/2} \exp\left(-\frac{m_b(\boldsymbol{v}-\boldsymbol{v}_b)^2}{2k_\mathrm{B}T_b}\right) . \tag{2.51}$$

For convenience, the notations [161, p. 18]

$$u = \sqrt{\frac{2k_\mathrm{B}T_b}{m_b}} , \tag{2.52}$$

$$\boldsymbol{\chi} = \frac{(\boldsymbol{v}-\boldsymbol{v}_b)}{u} . \tag{2.53}$$

are introduced, where u is a measure of the background species thermal velocity and χ is a measure of the velocity of the impurity relative to the background species flow velocity. After inserting the Maxwellian (2.51) in the Trubnikov potentials (2.47) and (2.48), the drift and diffusion coefficients can be written as [161, p. 60]

$$\boldsymbol{K} = \left(1 + \frac{m}{m_b}\right)\frac{\Lambda}{u}\frac{\boldsymbol{\chi}}{\chi}\phi_0' , \tag{2.54}$$

$$\underline{\underline{\mathbf{D}}} = \frac{\Lambda}{u^2}\left[\left(\underline{\underline{\mathbf{I}}} - \frac{\boldsymbol{\chi}\boldsymbol{\chi}^\mathrm{T}}{\chi^2}\right)\frac{\psi_0'}{\chi} + \frac{\boldsymbol{\chi}\boldsymbol{\chi}^\mathrm{T}}{\chi^2}\psi_0''\right] . \tag{2.55}$$

Here, $\underline{\underline{\mathbf{I}}}$ is the identity matrix, the superscript T denotes matrix transposition, and $\phi_0(\chi)$ and $\psi_0(\chi)$ are potential functions of χ (see [160] or [161, p. 130])

$$\phi_0(\chi) = \frac{\Phi}{u\chi}\,, \tag{2.56}$$

$$\psi_0(\chi) = \frac{u\Phi}{2} + \left(\frac{u}{2\chi} + \frac{1}{u\chi}\right)\Phi\,, \tag{2.57}$$

with the error function $\Phi(\chi) = \mathrm{erf}(\chi)$. After inserting the derivatives of these potentials in equations (2.54) and (2.55) and introducing the notations

$$D_\parallel = \frac{\Lambda\psi_0''}{u^2} = \frac{\Lambda}{u}\left(\frac{\Phi}{\chi^3} - \frac{\Phi'}{\chi^2}\right)\,, \tag{2.58}$$

$$D_\perp = \frac{\Lambda\psi_0'}{u^2\chi} = \frac{\Lambda}{u}\left(\frac{\Phi'}{2\chi^2} + \frac{\Phi}{\chi} - \frac{\Phi}{2\chi^3}\right)\,, \tag{2.59}$$

the final form is obtained,

$$\boldsymbol{K} = -\frac{D_\parallel}{u}\left(1 + \frac{m}{m_b}\right)\boldsymbol{\chi}\,, \tag{2.60}$$

$$\underline{\underline{\mathbf{D}}} = D_\perp\left(\underline{\underline{\mathbf{I}}} - \frac{\boldsymbol{\chi}\boldsymbol{\chi}^{\mathrm{T}}}{\chi^2}\right) + D_\parallel\frac{\boldsymbol{\chi}\boldsymbol{\chi}^{\mathrm{T}}}{\chi^2}\,. \tag{2.61}$$

The drift vector $\boldsymbol{K}$ acts in the negative direction of the relative velocity $\boldsymbol{\chi}$ and tends to minimize it ('friction'), while the diffusion tensor $\underline{\underline{\mathbf{D}}}$ has components both parallel and perpendicular to the relative velocity. Note that for small χ,

$$D_\parallel = D_\perp = \frac{\Lambda}{u}\frac{4}{3\sqrt{\pi}}\,, \tag{2.62}$$

$$\boldsymbol{K} = -\left(1 + \frac{m}{m_b}\right)\frac{\Lambda}{u^2}\frac{4}{3\sqrt{\pi}}\boldsymbol{\chi}\,, \tag{2.63}$$

$$\underline{\underline{\mathbf{D}}} = \frac{\Lambda}{u}\frac{4}{3\sqrt{\pi}}\underline{\underline{\mathbf{I}}}\,, \tag{2.64}$$

which means that for $\chi = 0$ the drift vanishes altogether while the diffusion becomes isotropic.

The calculations above were performed for a shifted Maxwellian background plasma. The presence of strong temperature gradients, such as in the tokamak divertor, can be modeled by adding a perturbation to the shifted Maxwellian distribution (2.51) that is proportional to the temperature gradient [161, pp. 28-35]. The result of the perturbation is an additional term in the drift vector, the thermal

force, which is proportional to the temperature gradient.

Numerical solution of the Fokker-Planck equation by the Monte-Carlo method

The Fokker-Planck equation (2.50) can be written in the more general form [161, p. 44]

$$\frac{\partial p}{\partial t} = -\sum_i \frac{\partial}{\partial z_i}(K_i p) + \frac{1}{2}\sum_{i,j}\frac{\partial^2}{\partial z_i \partial z_j}\left(D_{ij} p\right) , \tag{2.65}$$

where $p = p(\boldsymbol{z}, t)$ is the probability density function for a multi-dimensional stochastic variable $\boldsymbol{Z}(t)$, and K_i and D_{ij} are again the components of a drift vector and diffusion tensor. It can be shown that equation (2.65) can be solved numerically by [161, p. 46]

$$\boldsymbol{z}(t + \Delta t) = \boldsymbol{z}(t) + \boldsymbol{K}\Delta t + \underline{\underline{\mathbf{B}}}\Delta \boldsymbol{W} , \tag{2.66}$$

which describes a diffusive stochastic process (Brownian motion) in the phase space. $\Delta \boldsymbol{W}$ is a stochastic increment given by

$$\Delta \boldsymbol{W} = (\Delta t)^{1/2}\boldsymbol{\xi} , \tag{2.67}$$

where $\boldsymbol{\xi}$ is a set of Gaussian distributed random numbers with mean zero and variance one, which is obtained numerically with a pseudo-random number generator. $\underline{\underline{\mathbf{B}}}$ is a matrix defined by

$$\underline{\underline{\mathbf{B}}}\,\underline{\underline{\mathbf{B}}}^{\mathrm{T}} = \underline{\underline{\mathbf{D}}} . \tag{2.68}$$

Applying equations (2.66) and (2.65) to the test particle Fokker-Planck equation (2.50), with the six-tuple $\boldsymbol{z} = (\boldsymbol{r}, \boldsymbol{v})$, the final set of equations used for numerical modeling in ERO can be written as

$$\boldsymbol{r}(t + \Delta t) = \boldsymbol{r}(t) + \boldsymbol{v}(t)\Delta t , \tag{2.69}$$

$$\boldsymbol{v}(t + \Delta t) = \boldsymbol{v}(t) + \left(\boldsymbol{K} + \dot{\boldsymbol{v}}\right)\Delta t + \underline{\underline{\mathbf{B}}}\Delta \boldsymbol{W} . \tag{2.70}$$

With the results from the previous subsection for Maxwellian backgrounds, $\boldsymbol{K}$ is given by equation (2.60) and $\underline{\underline{\mathbf{B}}}$ is calculated from equations (2.61) and (2.68) as

$$\underline{\underline{\mathbf{B}}} = \sqrt{D_\perp}\left(\underline{\underline{\mathbf{I}}} - \frac{\boldsymbol{\chi}\boldsymbol{\chi}^{\mathrm{T}}}{\chi^2}\right) + \sqrt{D_\parallel}\frac{\boldsymbol{\chi}\boldsymbol{\chi}^{\mathrm{T}}}{\chi^2} . \tag{2.71}$$

The numerical procedure for simulating the test particle transport in ERO works as follows: at the beginning, an ensemble of test particles in phase space $(\boldsymbol{r}, \boldsymbol{v})$ is defined, and then the phase-space trajectory of each individual test particle is advanced by repeatedly applying the MC iteration step (2.69)-(2.70).

Note that $\dot{\boldsymbol{v}}$ on the right-hand side of equation (2.70) is given by the Lorentz force (2.23), which makes the equation implicit as discussed at the beginning of the section. For each time step, the equation is solved by applying the Boris method (2.28)-(2.30) and subsequently adding the other terms on the right-hand side of equation (2.70) to the velocity.

Figure 2.9 and Figure 2.10 show the results of an ERO2.0 simulation (motivated by similar simulations with the DORIS code presented in [161, pp. 62-63]), which demonstrates the relaxation of the test particle distribution function to a shifted Maxwellian distribution (2.51). An ensemble of 10^4 Be^+ particles was created with the same initial velocity $v_{x0} = 10^4\,\mathrm{m/s}$, $v_{y0} = 0$, $v_{z0} = 10^4\,\mathrm{m/s}$. The background plasma has the constant conditions

$$v_{b,z} = 5 \times 10^4\,\mathrm{m/s}\,,\; n_\mathrm{e} = 10^{20}\,\mathrm{m^{-3}}\,,\; T_\mathrm{e} = T_\mathrm{i} = 100\,\mathrm{eV}\,,\; B_z = 4\,\mathrm{T}\,.$$

After about $10^{-3}\,\mathrm{s}$, the test particles are fully thermalized with the background plasma, i.e. they have attained its temperature and flow velocity.

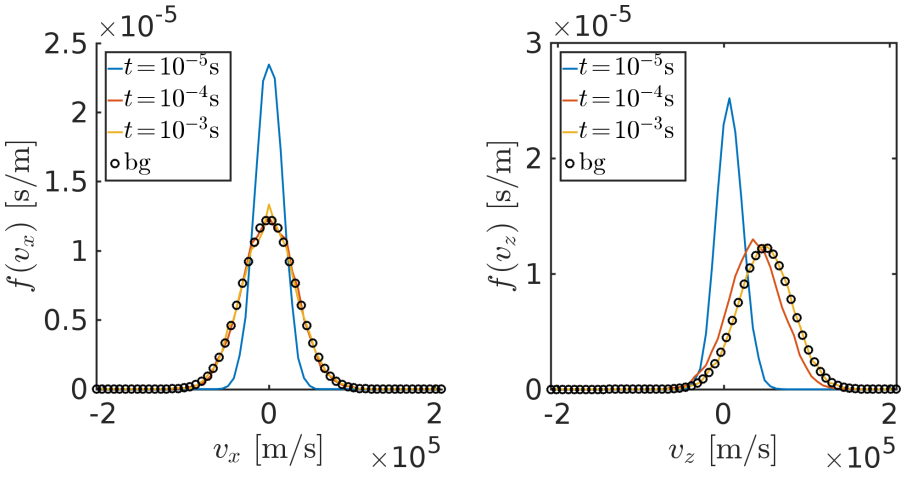


Figure 2.9: ERO2.0 simulation of the temporal evolution of the test particle distribution function, which relaxes to a shifted Maxwellian distribution ('bg') due to collisions with the background plasma.

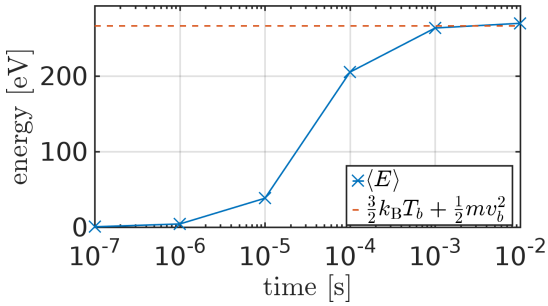


Figure 2.10: ERO2.0 simulation of the mean energy of the test particle ensemble, which converges against the energy prescribed by the background plasma.

Anomalous diffusion

As explained e.g. in [208, pp. 150f], the term anomalous transport denotes the experimentally observed enhanced transport of charged particles perpendicular to the magnetic field in tokamaks, which exceeds the values expected from classical or neoclassical theory. The origin of anomalous transport is often attributed to micro-fluctuations of the plasma which result in $\boldsymbol{E} \times \boldsymbol{B}$ drifts [208, p. 198]. A common way to empirically describe this effect is by assuming the anomalous transport to be diffusive with a corresponding diffusion coefficient $D_\perp^{\mathrm{a}}$, which is typically in the range of 0.1–10 m^2/s [182, p. 157]. Empirically, one often finds

$$D_\perp^{\mathrm{Bohm}} \approx 0.06\, T_\mathrm{e}/B \tag{2.72}$$

(Bohm diffusion) [182, p. 156]. In ERO, anomalous transport can be formally introduced as an additional diffusive term in phase-space

$$\frac{1}{2} \sum_{i,j} \frac{\partial^2}{\partial r_i \partial r_j} \left[(\delta_{ij} - b_i b_j) D_\perp^{\mathrm{a}} f \right] \tag{2.73}$$

in the Fokker-Planck equation (2.50), with $\boldsymbol{b}$ being the unit vector of the magnetic field. This correspondingly adds a term $\underline{\underline{\mathbf{B}}} \Delta \boldsymbol{W}$ in the MC step (2.69), with

$$\underline{\underline{\mathbf{B}}} = \left(\underline{\underline{\mathbf{I}}} - \boldsymbol{b}\boldsymbol{b}^{\mathrm{T}} \right) \sqrt{D_\perp^{\mathrm{a}}}\,, \tag{2.74}$$

which acts as a diffusion perpendicular to the magnetic field direction $\boldsymbol{b}$.

2.3.4 Atomic and molecular data

The test particles followed by ERO in a plasma can undergo a number of atomic and molecular processes, such as

- ionization (e.g. $Be^0+e^- \rightarrow Be^+ + 2e^-$ by electronic impact) and recombination (e.g. $Be^+ + 2e^- \leftrightarrow Be^0 + e^-$ by dielectronic excitation), in which the test particle charge state Z is increased or decreased
- light emission (e.g. the Be II 467.4 nm line for the $4f1\,^2F_{6.5} \rightarrow 3d1\,^2D_{4.5}$ transition)
- molecular dissociation by electron impact (e.g. $BeD + e^- \rightarrow Be + D + e^-$)

Due to molecular release and transport not being considered in the first ERO2.0 application presented in chapter 4, the focus in this section is on ionization, recombination and light emission. For describing the corresponding transitions between different energy levels of plasma ions, numerous collisional and radiative processes need to be accounted for that provide these transitions, such as collisional ionization, radiative recombination, photoionization, spontaneous and stimulated emission etc. [50, pp. 41-42]. Because of the dominant role of electron collisions in most processes, the rate coefficients used to characterize those are typically functions of the electron density and temperature.

Two models frequently used for the description are the corona model and the local thermodynamic equilibrium (LTE) model [50, pp. 42-50]. The corona model is valid for an optically thin plasma of low density, in which the population of a state is determined by a balance between collisional excitation and spontaneous emission. The LTE model is valid for a high density plasma, in which collisional effects dominate the radiative ones. In this case a detailed balance exists between the ionization states, which is described by the Saha equation.

A more universal description is provided by the generalized collisional-radiative (GCR) theory (see e.g. [50, pp. 50-57] or [191]). In contrast to the corona and LTE models, all relevant transitions between energetic levels are taken into account and described by elements C_{ij} of a collisional-radiative matrix and additionally by the coefficients q_i and r_i for direct ionization and recombination. For instance, taking the simple case of a system where the dominant populations are a recombined ion ground state N_1 and a recombining ion ground state N_+, the time-dependent equation for N_1 is [191]

$$\frac{\mathrm{d}N_1(t)}{\mathrm{d}t} = -N_\mathrm{e} S_\mathrm{CD} N_1(t) + N_\mathrm{e} \alpha_\mathrm{CD} N_+(t),, \tag{2.75}$$

where S_{CD} and α_{CD} are the collisional-dielectronic[19] (CD) ionization and recombination coefficients, which depend on C_{ij}, q_i and r_i.

The Atomic Data and Analysis Structure (ADAS) provides an interconnected set of computer codes and data collections for coefficients such as S_{CD} and α_{CD} within the framework of GCR theory [45, 191]. An important assumption used in ADAS is the quasi-static approximation: populations of short-living excited states are assumed to be in quasi-equilibrium with respect to the instantaneous dominant populations of long-living states (ground and metastable states). This way, the atomic problem can be partially decoupled from the transport problem, so that universal coefficients can be calculated that depend just on the electron density and temperature.

Ionization and recombination can be integrated into the ERO MC approach of calculating the test particle transport, i.e. equations (2.69)-(2.70). For instance, focusing on the first right-hand term (ionization) in equation (2.75) and neglecting the second right-hand term (recombination), the solution of the rate equation is an exponential decay

$$N_1(t) = N_1(0) \exp\left(-\frac{t}{N_{\mathrm{e}} S_{\mathrm{CD}}}\right) = N_1(0) \exp(-t/t_{\mathrm{ion}}) \tag{2.76}$$

with the ionization time $t_{\mathrm{ion}} = (N_{\mathrm{e}} S_{\mathrm{CD}})^{-1}$. The number of particles being ionized during a time interval Δt is

$$\Delta N_1(\Delta t) = N_1(0) - N_1(\Delta t) = N_1(0)\left(1 - \exp(-t/t_{\mathrm{ion}})\right) , \tag{2.77}$$

thus the probability of one particle being ionized is given by[20]

$$P_{\mathrm{ion}}(\Delta t) = \Delta N_1(\Delta t)/N_1(0) = 1 - \exp\left(-\Delta t/t_{\mathrm{ion}}\right) . \tag{2.78}$$

This is used at each time step of a test particle in ERO to decide whether it was ionized. A random number ξ between zero and one is drawn and compared to P_{ion}. If $\xi > P_{\mathrm{ion}}$, the particle gets ionized.

In a similar way, this method can be applied to recombination events, with a recombination probability P_{rec} as in equation (2.78), but with $t_{\mathrm{ion}} = (N_{\mathrm{e}} S_{\mathrm{CD}})^{-1}$ replaced by the recombination time $t_{\mathrm{rec}} = (N_{\mathrm{e}} \alpha_{\mathrm{CD}})^{-1}$. Figure 2.11 shows an

[19] The term 'collisional-dielectronic' is used as a synonym for 'collisional-radiative', see `http://open.adas.ac.uk/terminology`.

[20] An alternative approach for obtaining equation (2.78) is to consider the ionization a rare event described by a Poisson process. The average time Δt between two occurences (ionizations) is exponentially distributed, with the cumulative distribution function of the exponential distribution being exactly given by equation (2.78).

example of Be^0 ionization and Be^+ recombination rate coefficients from ADAS.

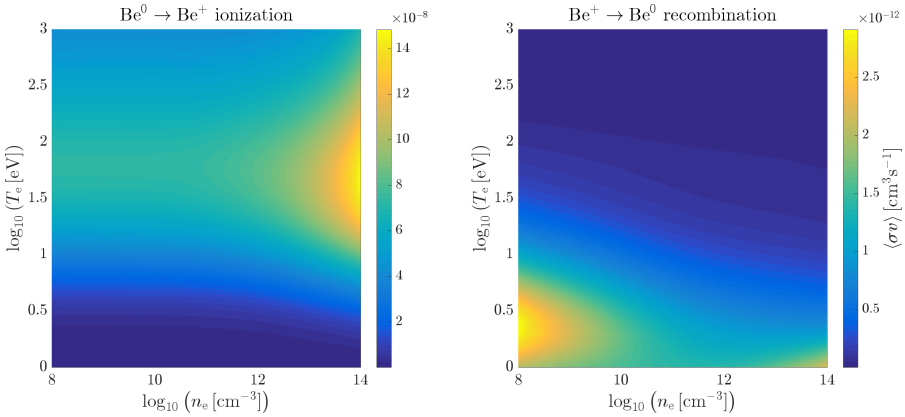


Figure 2.11: ADAS rate coefficients S_{CD} for $Be^0 \rightarrow Be^+$ ionization and α_{CD} for $Be^+ \rightarrow Be^0$ recombination as functions of the electron density and temperature.

In contrast to ionization and recombination, impurity radiation line emission is calculated in ERO only at the end of the simulation, using the calculated 3D distributions of ion densities n_α for each individual particle species α. The emissivity ε_λ [ph cm^{-3} s^{-1}] for a certain spectrum line λ is then given by [190, 191]

$$\varepsilon_\lambda = \left(\mathrm{PEC}_\lambda^{\mathrm{exc}}(n_\mathrm{e}, T_\mathrm{e}) + \mathrm{PEC}_\lambda^{\mathrm{rec}}(n_\mathrm{e}, T_\mathrm{e}) \right) n_\mathrm{e} n_\alpha \,, \tag{2.79}$$

where $\mathrm{PEC}_\lambda^{\mathrm{exc}}$ and $\mathrm{PEC}_\lambda^{\mathrm{rec}}$ [ph cm^3 s^{-1}] are the excitation and recombination photon emissivity coefficients for this spectrum line. The emissivity ε_λ gives the total number of photons emitted per plasma volume element and time unit.[21] Similar to the rate coefficients for ionization and recombination, ERO obtains the photon emissivity coefficients from ADAS. Figure 2.12 shows examples of $\mathrm{PEC}_\lambda^{\mathrm{exc}}$ for the Be I 457 nm and Be II 467 nm emission lines ($\mathrm{PEC}_\lambda^{\mathrm{rec}}$ is negligible for those).

[21] For the purpose of validation with experimental spectroscopic measurements, one should take into account the properties of the measurement system, which receives only a fraction of these photons, determined by the aperture and the corresponding solid angle (see section 4.2.6).

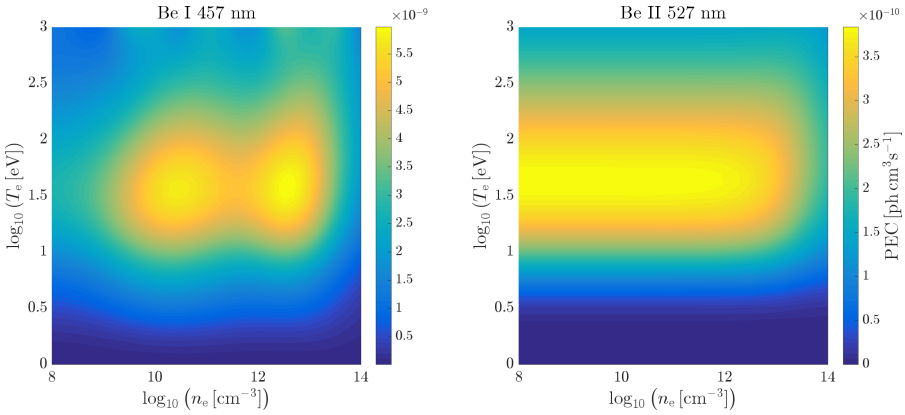


Figure 2.12: ADAS excitation photon emissivity coefficients $\mathrm{PEC}_{\lambda}^{\mathrm{exc}}$ for Be I 457 nm and Be II 467 nm emission as functions of the electron density and temperature.

Chapter 3

ERO2.0 code development

3.1 Motivation

The new code version ERO2.0 relies on the same scientific concept (described in chapter 2) as ERO1.0. However, several profound changes required developing the new version from scratch. The improvements that were achieved by the ERO2.0 development comprise three main points:

1. *Increasing the simulation volume size*: Due to the various code improvements and new algorithms presented in this chapter, ERO2.0 makes it possible to increase the simulation volume size such that it covers all relevant wall components of a fusion device in 3D ('global' modeling approach). This allows new possibilities in modeling compared to the previously used 'local' approach. One example is the modeling of global PWI and migration of impurities in a tokamak as presented in chapter 4. In such a simulation, the presence of impurities in the plasma is an outcome of the code rather than a model parameter, which allows a self-consistent treatment of erosion by impurities (e.g. Be self-sputtering, see section 4.3.2). A second example is the implementation of novel synthetic diagnostics for validation of the code, such as the 2D images from wide-angle cameras presented in section 4.2.6. A third example is the magnetic field line tracing, e.g. for calculating connection lengths on the wall in section 3.8 and resulting shadowing patterns in section 4.2.3 and 4.2.5. An outlook on potential future applications of ERO2.0 is given in chapter 5.

2. *Enhancing code performance*: A larger simulation volume size increases the code demand for computational performance and memory: larger or more complexly shaped wall geometries need to be included, more and longer test particle trajectories must be calculated to obtain satisfactory statistics,

etc. The necessary computational performance and memory are in first instance provided by massive parallelization for supercomputers (section 3.2), so that computing tasks and memory are distributed among a high number of computer processing units (CPUs), but also by optimization of time- and memory-intensive algorithms (sections 3.4–3.7).

3. *Improving code design*: ERO2.0 introduces a more universal and flexible approach of performing modeling for different applications (e.g. different fusion devices): a huge part of the simulation parameters, such as the wall geometry (see section 3.3) and the plasma backgrounds (see section 3.6), is now provided by exchangeable input files and thus separated from the code itself. This approach makes it obsolete to maintain multiple code versions for the individual applications. With the transition from C to C++ as the main programming language, ERO2.0 also provides a more flexible and modular code structure: the process of adding, testing, modifying or exchanging individual code functionalities (e.g. a certain PWI or test particle transport mechanism) is greatly simplified due to the C++ paradigm of object-oriented programming [131]. Even though these improvements are neither directly related nor required for the scientific goals mentioned in the first bullet above (and are therefore not further elaborated in this text), they are expected to greatly simplify the current and future (collaborative) code development.

3.2 Massive parallelization

3.2.1 Introduction

The upscaling of the simulation volume with ERO generally increases the code execution time and memory demand. Both are, for most applications, determined almost entirely by the test particle transport simulation, see appendix A. The execution time of the transport simulation grows with the volume size because (1) trajectories (which are otherwise 'cropped' at the volume boundaries) become longer and (2) more trajectories are required to obtain sufficient statistics for the entire volume. The memory demand scales with the volume size due to increasing sizes of the 3D grids on which plasma parameters and particle densities are stored (if the grid resolution is kept constant). The increased demand for performance and memory can be met by a supercomputer, which however requires an efficient parallelization of the transport simulation.

3.2.2 Parallelization with MPI and OpenMP

For the following discussion, it is necessary to introduce a few terms. A CPU core is an independent processing unit, to which a process or thread can be assigned for execution, within a multi-core CPU.[22] A node is a computing unit of a supercomputer which is equipped with its own multi-core CPU(s) and memory. A process is an instance of a program being executed by a CPU (or CPU core) with its own memory. A process may execute multiple threads concurrently (each by a separate CPU core) which typically share the memory of the process. More information can be found e.g. in [130, 158].

On computers that support the multiple instruction, multiple data (MIMD) parallelism (appendix B), the Message Passing Interface (MPI) and Open Multi-Processing (OpenMP) standards are two widely used approaches for parallelizing a code. These two approaches are illustrated in Figure 3.1.

MPI [193] is primarily designed for parallelism in distributed memory systems using processes (see appendix B). An MPI-parallelized program starts a number of processes, each of them executing the same code independently using its own memory. The parallelism occurs because each process has a unique ID ('rank'), which can be used to tell the process which part of the global problem it should work on. Additionally, the IDs can be used, like a postal address, to send messages between processes. For instance, process 0 might send a message containing a single integer to process 1 using the `MPI_Send()` command, which receives the message using the `MPI_Recv()` command.

The MPI Standard provides a large variety of communication modes. For instance, communication may be blocking (a process needs to wait until the communication is completed) and non-blocking (which allows to perform other computation while waiting for the communication to be completed). Collective communication modes are provided as well, e.g. a process sending data to multiple other processes with a single command, as opposed to point-to-point (PtP) communication between two processes (such as the `MPI_Send()` and `MPI_Recv()` commands).

OpenMP [194] in contrast is designed for parallelism in shared memory systems using threads (see appendix B). It uses the fork-join model of parallel execution (see Figure 3.1): a master thread is executed until a parallel region is entered. Here it forks into multiple threads which are executed in parallel, each using a separate CPU core. After the parallel region, the threads join again with the

[22] Here and in the following, 'CPU core' refers not only to the physical but also to the additional logical cores enabled by the hyper-threading technology, see section 3.2.3.

master thread.[23]

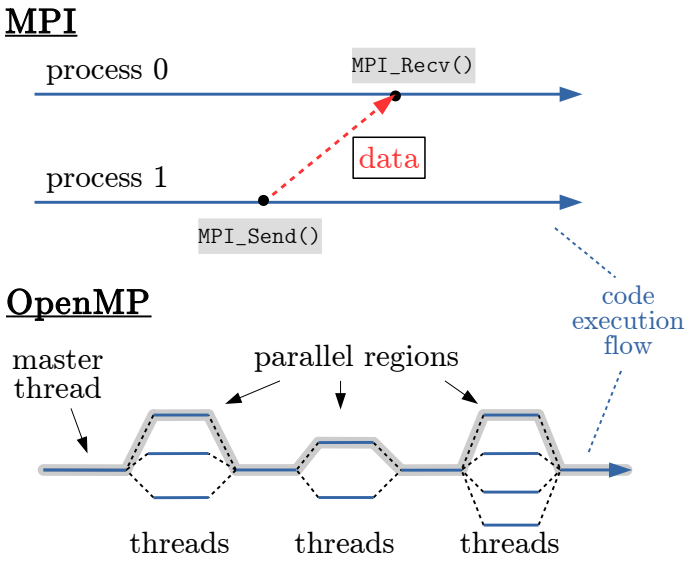


Figure 3.1: Illustration of parallel code execution flow in MPI and OpenMP.

ERO2.0 makes use of both MPI and OpenMP (hybrid MPI/OpenMP parallelization [156]). Thereby it combines the specific advantages of both approaches (good scaling by using many computing nodes in MPI, reduced memory consumption due to usage of shared memory of a node in OpenMP). This is illustrated in Figure 3.2. Using n nodes, the MPI system starts p processes per node, each of which can fork into a maximum of t threads (each executed by a CPU core). Following [156], we distinguish three cases, which are illustrated in Figure 3.2(a)-(c):

- Pure MPI: each CPU core of a node is used to execute an MPI process, no OpenMP threads are used.
- Mixed hybrid: fewer MPI process are started per node, but each of them has several CPU cores available for executing OpenMP threads.
- Fully hybrid: one MPI process is started per node, and all CPU cores of

[23] Parallel regions can be explicitly specified in the code by compiler directives. By default all variables in memory are shared among the threads of a parallel region, which in certain situations might lead to program errors due to concurrent write access of a variable. This can be prevented using various OpenMP compiler directives, e.g. by declaring this variable as private (each thread has an own copy of it), or by declaring the instruction with the write access as critical (it can be performed only by one thread at a time).

the node are used for OpenMP threads. MPI communication occurs only between the nodes.

As discussed in [156], a mixed or fully hybrid MPI/OpenMP approach certainly reduces memory consumption compared to the pure MPI approach. However, the optimal approach regarding code execution time strongly depends on the problem type and supercomputer architecture.

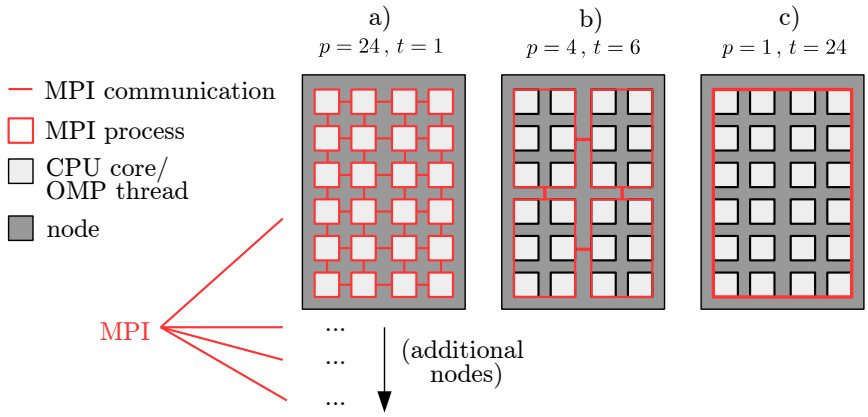


Figure 3.2: Illustration of different hybrid MPI and OpenMP parallelization approaches on a cluster with 24 CPU cores per node. p is the number of MPI processes per node, t is the number of OpenMP threads per MPI process. In each of the cases (a)-(c), the product $p \cdot t = 24$ is constant. a) Pure MPI (all cores used for MPI processes, no OpenMP threads), b) mixed hybrid (more than one MPI process per node, fewer cores used for OpenMP threads), c) fully hybrid (one MPI process per node, all cores used for OpenMP threads).

3.2.3 The JURECA supercomputer

The ERO2.0 results shown in this work were obtained using the Jülich Research on Exascale Cluster Architectures (JURECA) supercomputer [98]. JURECA comprises 1872 compute nodes. 1604 of the nodes have 128 GB of memory, the remaining nodes have 256 GB or 512 GB of memory. Each node has two sockets with a total of 24 physical CPU cores (Figure 3.3). In addition, due to hyper-threading technology each physical core can execute two processes or threads simultaneously, thereby doubling the total number of so-called logical cores or hardware threads (HWTs) to 48.

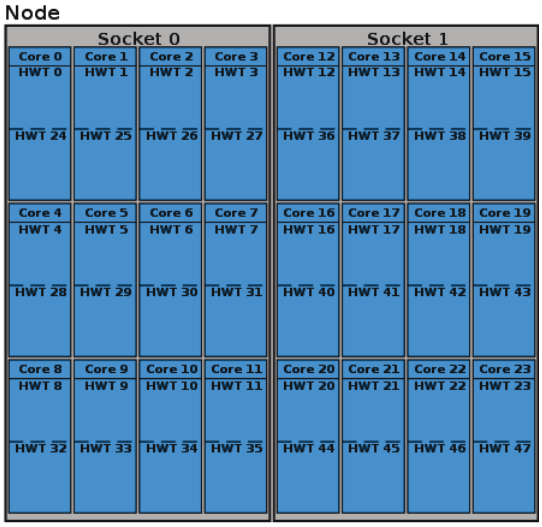


Figure 3.3: Illustration of the JURECA compute node, including hardware threads (HWTs). Image source: `http://www.fz-juelich.de/ias/jsc/EN/Expertise/Supercomputers/JURECA/UserInfo/SMT.html`.

3.2.4 Parallelization of ERO2.0

Outline of the transport loop parallelization scheme

The transport simulation in ERO requires calculating the trajectory for each test particle of the ensemble (transport loop). In the following, the trajectory calculation for a single test particle is referred to as a 'task'. A task is determined by the initial state of the particle at time $t = 0$, i.e. its chemical element, charge state, position and velocity. In the frame of the test particle approximation, all tasks are independent. In particular, the order in which the tasks are performed does not matter. In computer science, this type of problem is often called a bag-of-tasks (BoT) [9].

At the start of the program execution, one of the MPI process is defined as the 'master'. All other MPI processes are 'workers'. The master is responsible for creating the tasks, distributing them to the workers, and collecting the results, while the workers perform the actual computation of the tasks. In the hybrid approach, each worker can additionally use available CPU cores for creating OpenMP threads, each of which executes a part of the tasks in parallel.

For an efficient parallelization of the transport loop, particular attention must be paid to task scheduling, i.e. the way in which the individual tasks are distributed among the workers. Two different approaches are implemented in ERO2.0:

- Static scheduling: the tasks are distributed in even chunks among the processes at the beginning of the transport loop (Figure 3.4(a)).
- Dynamic scheduling: the tasks are dynamically assigned during the transport loop to processes which are idle (Figure 3.4(b)).

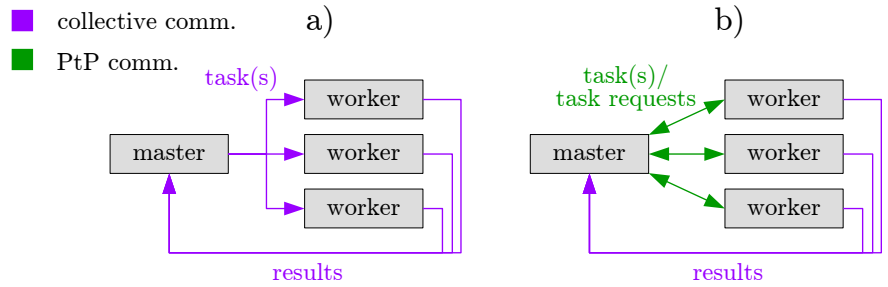


Figure 3.4: Illustration of the parallel communication pattern in the ERO2.0 transport loop. (a) Static schedule: equal chunks of tasks are distributed between the workers at the beginning of the loop. (b) Dynamic schedule: tasks are assigned dynamically to idle workers during the loop.

Static scheduling has several advantages. First, it has a particularly simple and robust implementation. Second, no further communication is required during the transport loop. Third, the master process can be treated as a worker as well, which means that it gives a fraction of the tasks to itself.

Dynamic scheduling in contrast has a more complex implementation and requires point-to-point (PtP) communication between the master and the workers during the transport loop. The master process does not compute tasks itself and is used entirely for keeping track of the queue of tasks and distributing them dynamically to the workers. This is done in the following way: any worker which is idle (i.e. has no tasks to perform) sends a corresponding message ('task request') to the master. The master answers the message by sending a new chunk of tasks to this worker. This is repeated until the master has no more tasks to distribute.

During the initial development of the ERO2.0 parallelization, dynamic scheduling was found to be far more efficient than static scheduling. The reason is that the tasks in ERO2.0 may be very different by the work load. For instance, a particle that redeposits just after a few time steps (prompt deposition) leads to a small work load. Due to the stochasticity introduced in the transport equations (2.70) and (2.73), a particle with only a slightly different initial state might, instead of being promptly deposited, performs a long trajectory with multiple turns around the torus, which leads to a much larger work load.

Consequently, even if equal numbers of tasks are distributed to the workers in static scheduling, the work load distribution will be unbalanced, as illustrated in Figure 3.5(a). The wall-clock time of the code is determined by the longest execution time of all processes (worker 2 in the illustration). This means that other processes run idly (illustrated by the white bars) and the parallelization is inefficient. In the dynamic scheduling case illustrated in Figure 3.5(b), smaller chunks of tasks are dynamically assigned to workers so that idle run of processes is minimized.

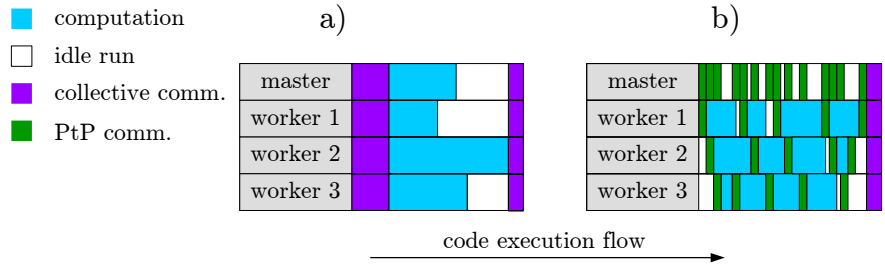


Figure 3.5: Illustration of potential idle run of CPU cores in the transport loop. (a) Static schedule: idle run may occur at the end of the loop due to different work load between processes. (b) Dynamic schedule: idle run may occur if communication of different workers with the master overlaps.

However, a potential pitfall in dynamic scheduling should be mentioned: the master process can work only on one task request at a time by PtP communication (green bars in Figure 3.5(b)). If several requests are in a queue, these are treated by the 'first-come-first-serve' principle. This might lead to a bottleneck in the task scheduling, namely if (1) the average time for receiving and processing a request by the master is not negligible compared to the average time for computing a chunk of tasks by a worker, and (2) the number of workers is very large. However, this potential bottleneck was so far not observed for the investigated range of MPI process numbers, as shown below by the parallel performance scaling tests.

Parallel performance scaling

Parallel performance is often characterized by 'strong' and 'weak' scaling [130]. For strong scaling, the problem size (number of test particles, simulation volume size etc.) remains fixed while the total number of CPU cores c used for the calculation is increased, with the aim of reducing the execution time $T(c)$. For weak scaling, the problem size is increased proportionally to c (the problem size per CPU core remains fixed) in order to keep $T(c)$ constant for increasingly large problems. The

aim of ERO2.0 is to increase the problem size, therefore the focus is mainly on weak scaling.

Figure 3.6 shows a weak scaling measurement of ERO2.0 execution time on JURECA. The simulation conditions are those of the Be global erosion and transport simulations for JET-ILW, which are presented in chapter 4. For the weak scaling measurement, the particle number N is taken as a measure of the problem size. The total number of CPU cores c is increased from 1 to 1536 (corresponding to 64 JURECA nodes with 24 CPU cores each). The particle number N is scaled accordingly, with $N(1) \approx 400$ test particles per CPU core.

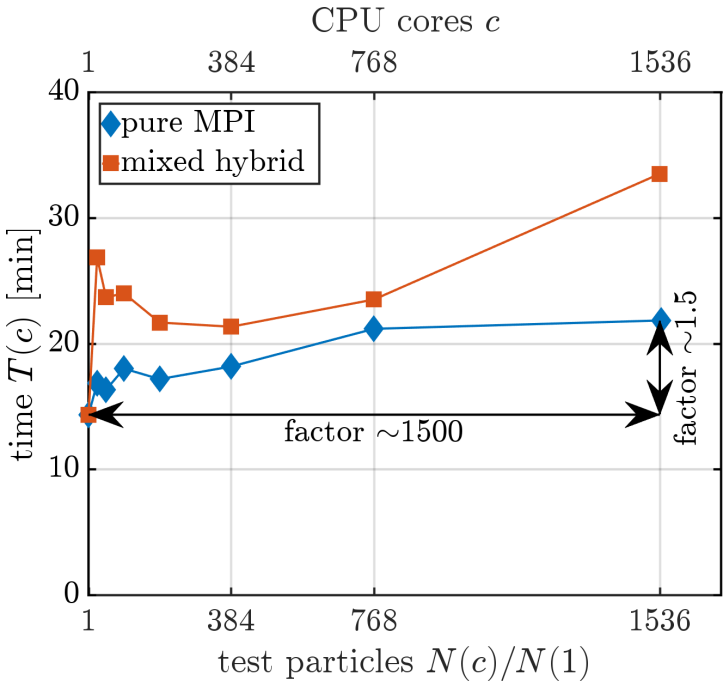


Figure 3.6: ERO2.0 parallel performance measurements using the Jülich supercomputer JURECA. The number of test particles $N(c)$ is scaled proportionally to the number of CPU cores c (weak scaling) with $N(1) \approx 400$. The code execution time $T(c)$ is measured. Two different parallel configurations are tested, namely pure MPI (all CPU cores are used for MPI processes) and mixed hybrid MPI/OpenMP (each MPI process launches six OpenMP threads, each of them using a separate CPU core).

Two different parallel configurations are used in Figure 3.6, namely pure MPI (with all CPU cores of each node used for an MPI process, i.e. $p = 24$ and $t = 1$, see Figure 3.2(a)) and mixed hybrid MPI/OpenMP (with $p = 4$ MPI process launched per node, each of those launching $t = 6$ OpenMP threads, see Figure 3.2(b)). It is observed that pure MPI leads generally to a better parallel performance compared

to mixed hybrid MPI/OpenMP. A fully hybrid MPI/OpenMP ($p = 1$ and $t = 24$, see Figure 3.2(b)) has also been tested (not shown) and leads to a significantly worse parallel performance (in the order of $T(c) \sim 60\,\text{min}$). However, it should be stressed that mixed or fully hybrid MPI/OpenMP can be useful in cases where memory allocated on the heap (e.g. for large 3D polygon meshes) exceeds the available memory on a node. In such cases, mixed or fully hybrid MPI/OpenMP will lead to a reduction of the memory usage by the factor t, since OpenMP threads make use of shared memory. However, for the simulations in chapter 4, memory is not a concern, wherefore the pure MPI configuration was used.

With respect to the parallel scaling for pure MPI in Figure 3.6, one observes a mild increase in execution time by a factor of $T(1536)/T(1) \approx 1.5$. This reflects the increasing fraction of time spent on communication between the cores. The increase of $T(c)$ by 1.5x is an acceptable price for calculating 1500x more test particles, which corresponds to a parallel speedup of 1000x. Also, the increase in $T(c)$ seems to converge around $c = 1536$. This means that the dynamic schedule is efficient in the investigated range of c, and in particular the idle run due to overlapping communication between the master and worker processes (as illustrated in Figure 3.5(b)) does not occur. It is expected that a higher speedup can be obtained using a higher number of CPU cores c.

For mixed hybrid MPI/OpenMP, the parallel scaling is more complex. At low numbers of c, a peak in the runtime occurs. The peak occurs because the 'sacrificing' of one MPI process (the master process, which does not perform any calculations) leads to the loss of t CPU cores. This can be improved (but is outside the scope of this work) by more involved hybrid algorithms, in which only one OpenMP thread of the master MPI process is responsible for the communication, while other threads perform calculations. Interestingly, an almost linear increase in the runtime is observed after a minimum around $c = 384$. This increase contradicts the expectation, since hybrid MPI/OpenMP actually requires fewer MPI processes and should hence lead to a lower communication overhead than pure MPI. A detailed investigation of the effect requires parallel code profiling at high c, which is outside the scope of this work.

3.3 Coordinate system and wall geometry

In ERO simulations, the wall can act both as a particle source (erosion) and sink (deposition). Furthermore, the wall influences the motion of nearby particles via the sheath electric field. The wall geometry is treated as a 2D surface in 3D space, which means that it formally has no thickness, although a certain PWI interaction

layer thickness can be introduced for modeling material mixing in the frame of the homogeneous mixing model. The 2D manifold is then further subdivided into a regular mesh of surface cells. For instance for the JET-ILW simulations in chapter 4, the cell size is about $1\,\mathrm{mm}^2$-$1\,\mathrm{cm}^2$. PWI properties such as erosion and deposition are calculated for each surface cell based on the local plasma parameters as well as the geometric and physical properties of the cell. Geometric properties are the surface normal (which is required to calculate the magnetic field inclination) and the area. Physical properties are the surface temperature and the material composition.

In ERO1.0, typical wall geometries are

- a (round or rectangular) flat sample exposed in a linear plasma device [22, 68],
- a simple test limiter geometry (e.g. TEXTOR roof [104, 105] or spherical [109, 111] test limiter),
- a shaped limiter in JET [1, 23] or ITER [25], a selection of JET divertor tiles [108], or a proxy tile assembly in EAST [55].

Since ERO1.0 has been designed for the modeling of small PFCs and surrounding simulation volumes of size $\lesssim 1\,\mathrm{m}^3$, a local cartesian coordinate system (x, y, z) is used, where x and y are chosen in such way that they span the surface plane and z is the perpendicular direction pointing away from the surface. E.g. in the case of a tokamak limiter, x and y are the toroidal and poloidal directions tangential to the limiter tip and z is the radial direction (Figure 3.7). This allows a convenient description of the wall geometries mentioned above: the surface cell centers x_i, y_j are regularly spaced in the (x, y)-plane, and their coordinates z_{ij} are initialized using a function $z_{\mathrm{wall}}(x, y)$ (which is typically an analytic approximation).

With the transition to much larger simulation volumes in ERO2.0, the usage of local coordinates becomes unfavorable when the entire (or at least a large part) of the curved torus geometry is considered, so that there is no fixed radial direction to associate the z-coordinate with. The definition of surface cells and wall geometry using local coordinates is especially problematic, since the wall coordinate function $z_{\mathrm{wall}}(x, y)$ becomes ambiguous as soon as the simulation volume spans toroidal or poloidal angles higher than 180° (or even at much lower angles if strongly shaped PFCs are involved).

Therefore, ERO2.0 utilizes a more general representation of wall geometry. In the context of computational geometry, computer graphics science or computer aided design (CAD), a surface (2D manifold in 3D space) is usually approximated

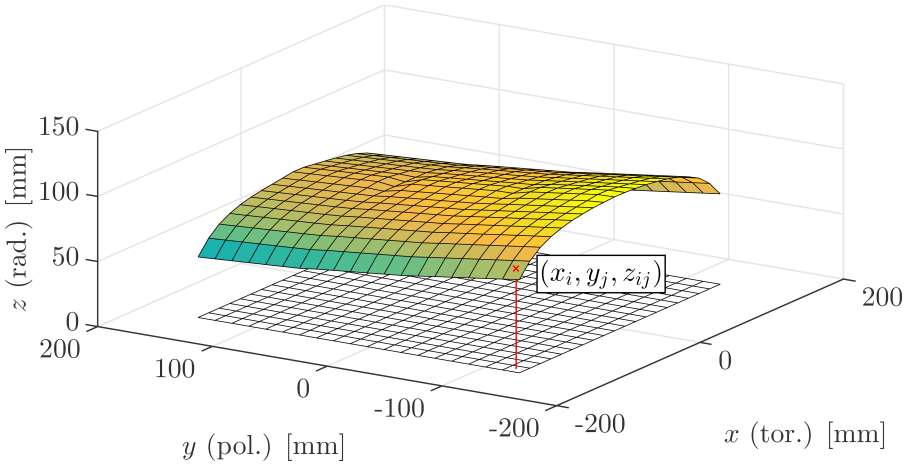


Figure 3.7: Illustration of the coordinate convention and the definition of surface cells and wall geometry in ERO1.0. The example shows the geometry of a JET Be limiter tile as used in [23]. A coarse surface cell resolution of $\sim$1 cm^2 cell size is used for clarity. The quadrilateral edges outline the surface cells with coordinates x_i, y_j and $z_{ij} = z_{\text{wall}}(x_i, y_j)$.

using either parametric or implicit representations [29]. Examples of parametric representations are non-uniform rational B-splines (NURBS) and polygon meshes. An example of an implicit representation is constructive solid geometry (CSG). The polygon mesh representation was chosen for ERO2.0, because it offers the most direct way of associating surface cells with the wall geometry by defining each polygon of the mesh to be a surface cell. Figure 3.8 shows an example of polygon meshes used in ERO2.0. The most common type of polygon meshes is the triangle mesh. ERO2.0 also supports quadrilateral meshes as well as mixed triangle and quadrilateral meshes. Each quadrilateral is internally represented as a set of two triangles sharing an edge. The polygon mesh is represented in ERO2.0 as an *indexed face set* [29], which consists of an array of V vertices (polygon corners),

$$\mathcal{V} = \{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_V\}\,, \tag{3.1}$$

and an array of F faces (polygons),

$$\mathcal{F} = \{f_1, \ldots, f_F\}\,. \tag{3.2}$$

Each vertex $\boldsymbol{v}$ is a 3D vector and each face f is a set of indices that point into the vertex array. A triangle is thus represented as $\mathcal{F} = \{i_1, i_2, i_3\}$ and a quadrilateral as $\mathcal{F} = \{i_1, i_2, i_3, i_4\}$. The indexed face set representation has a compact memory usage, because vertices can be shared between adjacent polygons, thus avoiding

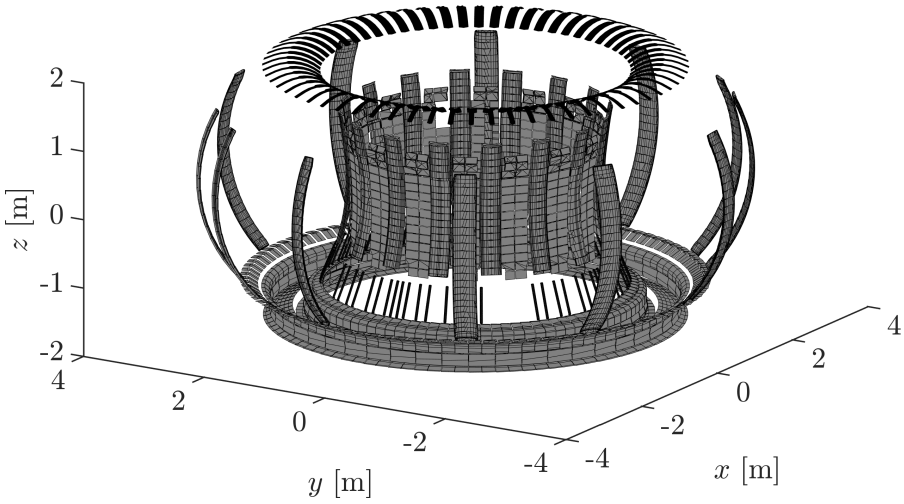


Figure 3.8: Polygon mesh used in ERO2.0 with selected JET ITER-like wall components (from the inner, outer and upper poloidal limiters, divertor, inner liner and protection bars, see e.g. [125]). The mesh consists of about 200,000 faces with a mix of triangles and quadrilaterals.

redundant storage of vertices (duplicates).[24]

After importing a polygon mesh, ERO2.0 needs to perform various operations on it, such as distance or intersection queries, which will be discussed in the following section. Because no suitable C++ program library was available, the polygon mesh algorithms were implemented from scratch and are tailored for ERO2.0 requirements.[25]

The polygon mesh representation allows to drop the local coordinate convention used in ERO1.0 and use any arbitrary cartesian coordinate system instead. In the case of tokamak simulations, the usage of so-called machine coordinates is preferred in ERO2.0. These are cartesian coordinate systems defined individually for each tokamak device, thereby allowing a consistent specification of e.g. diagnostics locations and lines-of-sight. The z axis is chosen as the axis of revolution, with $z = 0$ at the torus midplane. The location of the toroidal angle $\phi = 0°$ is fixed, thereby defining the x and y axes.

[24]ERO2.0 supports importing of meshes from files in the Polygon File Format (PLY). This is a common polygon mesh file format that also uses the indexed face set representation and is supported by various software tools. E.g. the polygon mesh shown in Figure 3.8 is stored in a PLY file which requires ∼16 MB of disk storage in text format, with a floating-point precision of 8 digits.

[25]At this point, I would like to thank Dr. Dmitriy Matveev for kindly providing his extensions to the LinAlg and MeshLib C++ libraries, see `https://bitbucket.org/sboz/meshlib/wiki/Home`. Several of his algorithms and ideas were used in the development of the ERO2.0 3D wall geometry module.

Using machine coordinates allows convenient switching between cartesian and cylindrical coordinates (R, ϕ, z). The latter are supported by ERO2.0 for a more convenient definition of simulation volume boundaries as well as for storage of plasma parameters, particle densities and spectroscopic emission on a 3D grid. As shown in the next chapter, it is useful to represent these in the poloidal plane (R, z) if toroidal symmetry is assumed. Also, cylindrical coordinates offer a convenient way of exchanging data with 2D edge fluid codes (SOLPS-ITER, EDGE2D-EIRENE, SOLEDGE2D-EIRENE), which use computational grids that lie in the (R, z)-plane.

In summary, the concept of polygon meshes allows ERO2.0 to efficiently describe even large and complex wall geometries, which in turn allows increasing the simulation volume to cover an entire fusion device such as ITER.

3.4 Optimized query operations on polygon meshes

Many algorithms in ERO2.0 require operations to be performed on the polygon meshes, e.g. for deciding whether a test particle trajectory or a magnetic field line hits a surface or not, or for calculating the direction and magnitude of the sheath electric field. The two most basic operations are the intersection and distance queries. These are very time-consuming operations (as discussed in appendix A) and were identified from early on as the main performance bottlenecks for the JET-ILW simulations in chapter 4. This is why the intersection and distance queries have been significantly optimized to ensure an efficient execution of the code, as will be now discussed in detail.

A distance query (see Figure 3.9(a)) means finding the 'foot point' $\boldsymbol{q}_0$ on the polygon mesh that has the closest distance $d_0 = |\boldsymbol{q}_0 - \boldsymbol{p}_0|$ to a certain 'query point' $\boldsymbol{p}_0$. As an example, such a query is required in order to calculate the distance of a test particle to the closest surface cell. Based on this distance, it is decided whether the test particle is inside the sheath. If so, the magnitude and direction of the sheath electric field are calculated as described in section 2.3.2, using the distance and direction vector between the test particle location and the query point.

For the further explanations in this section, it is instructive to imagine the distance query as the step-wise shrinking of a 'query sphere', which is centered at the query point $\boldsymbol{p}_0$ and has a radius R. At the beginning of the query, R is set to infinity. Each time a polygon with a footpoint $\boldsymbol{q}_i$ and distance $d_i = |\boldsymbol{q}_i - \boldsymbol{p}_0| < R$ is found within the query sphere, R is updated to d_i. This way, the sphere 'touches' the polygon mesh without intersecting it ($R = d_0$) at the end of the query.

An intersection query (see Figure 3.9(b)) means finding an intersection $\boldsymbol{q}_0$

between a line segment S, which is defined by its end points $\boldsymbol{a}$ and $\boldsymbol{b}$, and the polygons of the mesh. As an example, the intersection should in principle be queried for each discrete step (i.e. a line segment) of a test particle trajectory in order to decide whether it has crossed the wall. Note however that for most steps, the segment length $|\Delta \boldsymbol{r}| = |\boldsymbol{b} - \boldsymbol{a}|$ is smaller than the surface distance d_0, in which case an intersection is not possible and the query can be spared. Note also that due to test particle step lengths being much smaller than the dimensions of a polygon[26], it is safe to assume that there is no more than one intersection with a single polygon. The query may therefore be stopped as soon as an intersection is found.

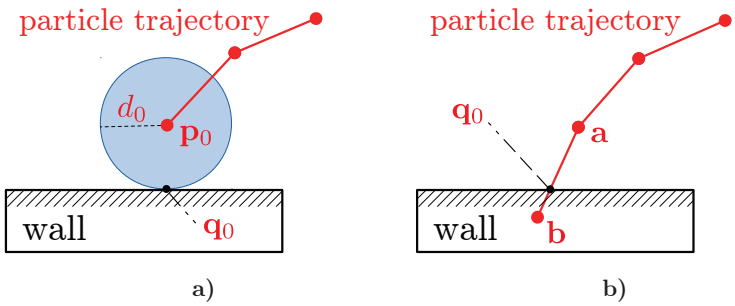


Figure 3.9: Illustration of the distance and intersection queries on polygon meshes (shown in 2D for simplicity). (a) Query of the surface distance d_0 (and footpoint $\boldsymbol{q}_0$) for a test particle location. The blue circle illustrates the concept of the query sphere. (b) Query of the intersection point $\boldsymbol{q}_0$ between a test particle step (i.e. a segment of its trajectory) and the surface.

For calculating the distance between a point and a single triangle in 3D, algorithms can be found e.g. in [174, pp. 376-382]. The algorithm can be trivially extended to the case of a quadrilateral by subdividing it into two triangles. The same applies to the intersection between a line segment and a triangle in 3D [174, pp. 485-488].

Extending this to a polygon mesh consisting of N polygons requires to repeat the algorithm for each of the polygons. The computational time for both intersection and distance queries on a polygon mesh therefore increases linearly with N ('linear method').[27] For typical ERO2.0 applications, polygon numbers of $N = 10^5$ or more

[26] E.g. for the JET simulations in chapter 4, typically step lengths of $\sim 0.1 r_g$ are chosen, with the Larmor radius r_g being in the order of ~ 1 mm. However, significantly lower step sizes of $\sim 1\,\mu$m are chosen for test particles inside the sheath. This is much smaller than the typical polygon edge lengths of ~ 0.7 cm in the JET polygon mesh model.

[27] Even though sometimes the intersection query for test particles may encounter the objective polygon relatively fast, in most cases no intersections exist and therefore all polygons need to be tested.

are not uncommon. The linear method becomes inefficient at large N, because the algorithm spends a high amount of time calculating intersections or distances for polygons that are obviously located too far away and could have been excluded a-priori from the query.

The performance of the query can be improved by introducing spatial partitioning structures. The space containing the polygons is recursively subdivided into a tree of smaller structures. The subdivision is usually made in such way that the tree is finer (i.e. has smaller structures) in regions with high numbers of polygons. Examples for spatial partitioning structures are the octree, the k-d tree and the binary space partitioning tree [5]. ERO2.0 uses the octree, which is particularly simple to implement. Figure 3.10 shows an example of an octree.

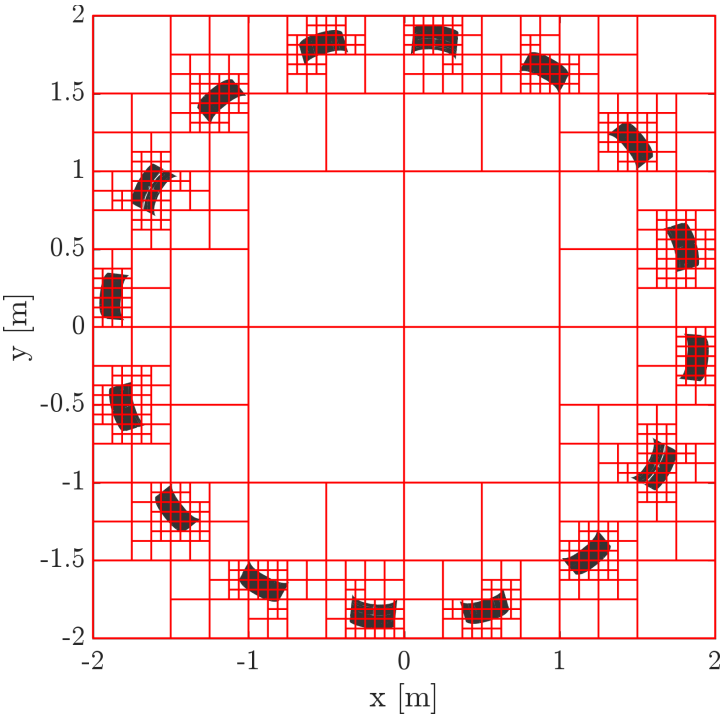


Figure 3.10: Example of an octree (red) used in ERO2.0, which adapts to a polygon mesh (black) that describes the 16 inner poloidal limiters of the JET ITER-like wall (viewed from the machine top). The tree has maximum node level $l_{\max} = 6$ and maximum $n_{\max} = 5$ polygons per node.

The octree construction algorithm starts with a top-level cuboid and recursively subdivides it into eight equally sized octants (children). This is repeated for the children in a recursive way. The tree construction is schematically illustrated in Figure 3.11(a) (in 2D for better view). In the tree terminology, each octant is

called a node, and nodes that have no children are called leaves. ERO2.0 stops the recursive subdivision (base case) if either the recursion level l exceeds $l_{\max}$, or if the number of polygons n that are inside the node are below $n_{\max}$, where $l_{\max}$ and $n_{\max}$ are user-specified thresholds. The so-created octree leaf node stores references to the polygons that are completely or partly inside. The algorithm for deciding whether a polygon is inside a node is based on the separating axis theorem as described in [2]. Since the wall geometry is static in ERO2.0, the octree needs to be constructed just once at the beginning of the simulation.

During the later course of the simulation, the octree is used for accelerating the distance and intersection queries as illustrated in Figure 3.11. The query starts at the top-level node and continues (if a certain condition is met as described below) by recursively traversing the children of each node. When a leaf node is reached (base case), the distance or intersection tests with the polygons within the leaf node are performed, and the query continues with the next branch of the octree.

The performance gain of the octree comes from the possibility to spare a huge part of polygons from the query. During the octree traversal, a test is performed for each node. In the case of an intersection query, the test is whether the segment intersects the node. In the case of a distance query, the test is whether the distance between the query point and the node boundaries is lower than the current query sphere radius R. The algorithms for calculating intersections and distances for nodes (i.e. axis-aligned boxes) are described in [174, pp. 626-630] and [69, pp. 131-132], respectively. They are significantly faster than the corresponding algorithms for polygons.

If a node fails the test, the recursion for the corresponding octree branch is stopped, which means that all polygons inside this node are spared from the query. Thus at high polygon numbers N, a performance gain can be achieved (compared to the linear method) that far outweighs the computational overhead due to the construction of the octree and the tests performed for each node during the query.

A measurement was performed for the intersection query in order to demonstrate the increase in computational performance using the octree method compared to the linear method. A number of 1000 line segments was created, with size 100 mm (i.e. a typical test particle step length, see footnote 26) and of random orientation and location in the volume shown in Figure 3.10. The number N of faces in the polygon mesh shown in Figure 3.10 was increased by 1x, 4x and 16x. Figure 3.12(a) shows the results. One can see that the octree method decreases the query time by more than two orders of magnitude for low numbers N, and by more than three orders of magnitude for high numbers N.

To achieve a similar performance increase for the distance query, it is additionally

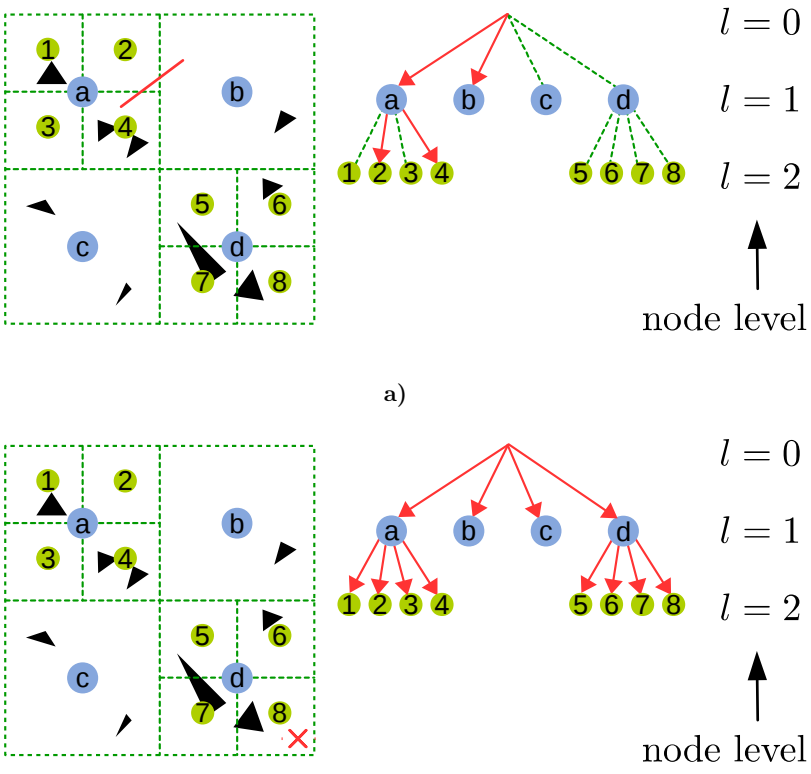


Figure 3.11: Schematic view in 2D space (left) and graph representation (right) of the octree. The circled characters and numbers indicate the nodes at different octree levels. The black triangles represent elements of the polygon mesh (disjoint for better visibility). The green dashed lines respectively indicate the node boundaries and the connections of parent to child nodes. The red arrows represent the node traversal during intersection or distance query. The octree is recursively constructed with maximum node level $l_{\max} = 2$ and maximum $n_{\max} = 2$ polygons per node.
(a) Example of an intersection query with a line segment (red) using the octree. During the recursive octree traversal from top to bottom, the nodes '1', '3', 'c' and 'd' are dismissed, since they are not intersected by the segment. Finally, intersection tests will be performed with the polygons within nodes 'b' and '4'. (b) Example of a distance query with a point (red cross) where the octree is inefficient because all nodes are traversed. The query can be made efficient if an upper estimate for the minimum distance is known at the beginning.

required to initialize the query sphere radius R_{start} at the beginning with an appropriate value, e.g. with an estimate for the real distance $R_{\text{start}} = d_{\text{upper}} \geq d_0$. The closer the estimate d_{upper} is to the real minimum distance d_0, the more nodes

can be dismissed from the query and the more efficient the algorithm becomes. Otherwise if the sphere is initialized with $R_{\text{start}} = \infty$, it may happen that (depending on the order in which the children of each node are traversed) the leaf containing a certain query point is the last one to be traversed. In the worst case, this would mean that all polygons are tested despite the octree. An example of such a situation is illustrated in Figure 3.11(b). Two methods can be used in ERO2.0 to provide an upper estimate $d_{\text{upper}} \geq d_0$ for a certain query point:

- For distance queries where the query point is the test particle position after a certain time step, $d_{\text{upper}} = d_0^\star + |\Delta \boldsymbol{r}|$ is used, where $d_0^\star$ is the surface distance previously found before the time step, and $|\Delta \boldsymbol{r}|$ is length travelled during that step. This is usually a very precise estimate due to the test particle step lengths $|\Delta \boldsymbol{r}|$ being much smaller than the polygon dimensions (see footnote 26).

- For other distance queries (i.e. when no information about d_0 is available from previous steps), a rougher estimate is used:

$$d_{\text{upper}} = \begin{cases} L/2 + d_{\text{leaf}} & \text{if empty leaf,} \\ L & \text{else.} \end{cases} \tag{3.3}$$

  Here, L is the diagonal length of the leaf node which contains the query point and d_{leaf} is distance from the center of the leaf to the closest polygon outside its boundaries (if it doesn't contain any polygons itself). It can be shown by purely geometrical considerations that equation (3.3) is a valid upper estimate of d_0 for any query point inside the leaf. The values of d_{upper} are calculated once for each leaf node during the octree construction.

An even better performance of the octree distance query for test particles was achieved by setting $R_{\text{start}} = \min(d_{\text{upper}}, R_{\text{cut-off}})$, where $R_{\text{cut-off}}$ is a user-defined cut-off radius. Octree nodes and polygons beyond this radius are excluded. $R_{\text{cut-off}}$ is chosen as the maximum sheath width expected in the simulation volume (typically $\sim$1 cm), because for particles outside the sheath one is not interested in the outcome of the distance query. This way, performance is improved in particular for test particles that are well outside the sheath ($d_0 \gg R_{\text{cut-off}}$), for which otherwise $R_{\text{start}} = d_{\text{upper}}$ would result in a large initial sphere radius and thus in the testing of many polygons.

Finally, the distance query in general (with and without octree) was further optimized by first checking the distance of the query point to the pre-calculated axis-aligned bounding box (AABB) of each polygon, and dismissing the polygon if

this distance is larger than the query sphere radius. The AABB corners $\boldsymbol{a}, \boldsymbol{b}$ are obtained as

$$a_i = \min_j v_{ji}\,, \qquad b_i = \max_j v_{ji} \tag{3.4}$$

where v_{ji} is the ith coordinate of the jth polygon vertex. This way the number of polygon distance queries can be further reduced.

Figure 3.12(b) shows the performance results for the distance query in a similar test as for the intersection query. The octree method with $R_{\text{start}} = \infty$ decreases the query time slightly for high N (compared to the linear method). The octree method with $R_{\text{start}} = d_{\text{upper}}$ from equation (3.3) decreases the query time by ∼3x for low N and by ∼6x for high N. The relevant method for test particle tracing however is the one with $R_{\text{start}} = R_{\text{cut-off}}$ (here $R_{\text{cut-off}} = 1\,\text{cm}$ was chosen). In this case, the octree decreases the query time by ∼200x for low N and by ∼1200x for high N.

In summary, the octree method enables ERO2.0 to work efficiently even if wall geometries with high numbers of polygons are used. In fact, the total computational time for the simulations in chapter 4 (with a number of 5×10^5 polygons) decreases roughly by a factor of 10^4 due to these improvements, as discussed in appendix A.

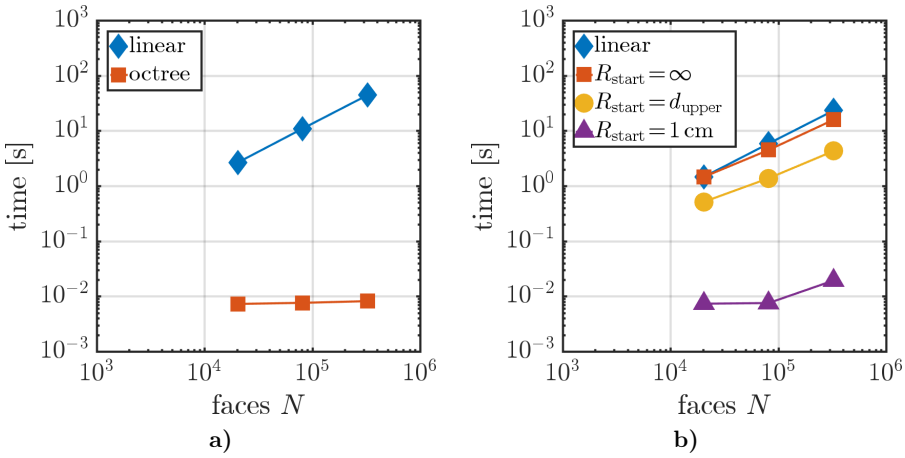


Figure 3.12: Dependence of the computational time for (a) intersection and (b) distance queries, with different numbers N of faces in the polygon mesh, and using the linear and octree methods. The octree construction parameters are $l_{\text{max}} = 7$ and $n_{\text{max}} = 5$. An ensemble of 1000 line segments and points are queried respectively, which are randomly distributed in the volume shown in Figure 3.10. In the case of octree distance queries, different values for the initial query sphere radius are used: $R_{\text{start}} = \infty$, $R_{\text{start}} = d_{\text{upper}}$ as in equation (3.3), and $R_{\text{start}} = R_{\text{cut-off}} = 1\,\text{cm}$.

3.5 File format and structure

For reading and writing (input and output, I/O) of data, ERO1.0 uses text files. These are generally advantageous due to being human-readable and -editable as well as for their cross-platform portability. However, text files are less suitable for high-performance computing (HPC) applications foreseen for ERO2.0, where data sets do not only have a complex structure but are also very large ('heavy data'), which may lead to a performance bottleneck if I/O is inefficient. Therefore, ERO2.0 relies almost entirely on the Hierarchical Data Format, which is specifically designed for HPC applications, in its most recent fifth version (HDF5) [192] for the input and output files.[28]

The main advantage of using HDF5 for ERO2.0 is the particularly efficient I/O of large data sets compared to text files. In order to demonstrate this, Figure 3.13 shows the run-time for such operations measured with a C++ test code. A 1D N-sized array of double-precision floating-point numbers (doubles) was created, then written to an HDF5 and text file (with 10-digit precision in the latter case), and read from those into memory again. The code run time for each of the I/O operations was measured. Compared to text files, HDF5 is less efficient at small array sizes up to about $N = 500$, but is faster by more than an order of magnitude (for reading, about two orders of magnitude) at large array sizes. Similar results were obtained for 2D arrays, and also for the I/O of HDF5 and text files with MATLAB (which is the primary software used for visualization of ERO2.0 results). Therefore, HDF5 is a more appropriate file format choice for ERO2.0 applications with large in- and output files.

A second advantage of HDF5 is its flexible file structure. An HDF5 file stores objects (e.g. data sets, groups containing data sets or other groups, and attributes describing those) in a hierarchical way that resembles a filesystem. This is illustrated in Figure 3.14, which shows the structure of the ERO2.0 output file for surface data. A group is created for each plasma-facing component (e.g. limiter, divertor). Each group is further subdivided into groups (e.g. plasma-wall interaction data, geometric data). A group can contain multiple data sets (e.g. the erosion fluxes of particles sputtered by different mechanisms). Each group and data set can have attributes (e.g. the physical units of a data set given as a character string, e.g.

[28] There are some exceptions to this. One important example is the input parameter file for ERO2.0, which determines the paths to the other input and output files but also various simulation parameters, such as the number of test particles to be started, is written as a text file in the Extensible Markup Layout (XML). This is because XML files can be more easily viewed and edited (e.g. with a text editor) than HDF5 files. Another example is the input wall geometry that is given in the PLY format, which is supported by various polygon mesh processing software, see footnote 24.

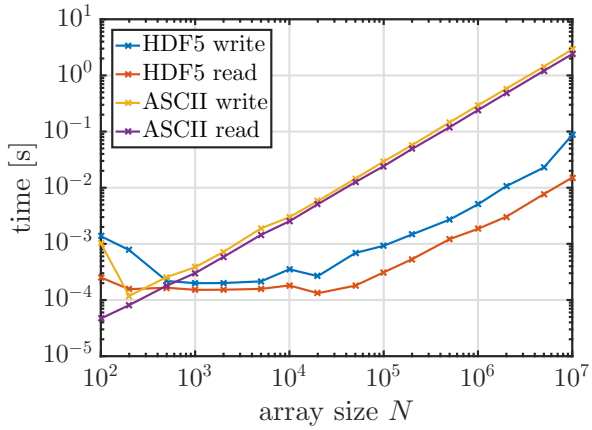


Figure 3.13: Measured time of a C++ code for writing and reading an array of N doubles, using HDF5 files and text files. For large arrays, HDF5 is far more efficient.

$cm^{-2}s^{-1}$). This structure offers a very convenient, flexible and compact way of storing large and complex data.

A third advantage of HDF5 is its support for advanced operations such as partial I/O. This is utilized by ERO2.0 in the context of domain decomposition (see section 3.6) in order to read the subsets of a plasma parameters data matrix corresponding to a certain region of the simulation volume.

A fourth advantage is that HDF5 provides application programming interfaces (APIs) for various programming languages or software tools, including C/C++, Fortran, Python and MATLAB. Each object can be individually accessed for I/O by its unique path in the file. For instance, the MATLAB command

```
data=h5read('/path/to/HDF5file','/limiter/geometry/vertices')
```

would read the limiter vertices from the file example in Figure 3.14 into memory. Thus, the programming effort for developing routines for I/O in ERO2.0 and post-, pre-processing or visualization of the corresponding files is greatly reduced.

Finally, HDF5 is a standardized file format widely used for HPC applications. An example is the NASA Earth Observing System (EOS), where HDF5 is in use since many years.[29,30] More importantly, HDF5 is widely used in the modeling for fusion research, for instance in the European Integrated Tokamak Modelling (EU-ITM) task force [95], or in plasma edge codes (EMC3-EIRENE, SOLEDGE2D-EIRENE). This potentially simplifies the data exchange between ERO2.0 and these codes.

[29] https://www.hdfgroup.org/about/our-history/
[30] https://earthdata.nasa.gov/user-resources/standards-and-references/hdf5

In summary, the usage of the HDF5 file format enables ERO2.0 to perform computationally efficient and user-friendly I/O with large and complex data sets, and to more easily exchange data with other scientific codes.

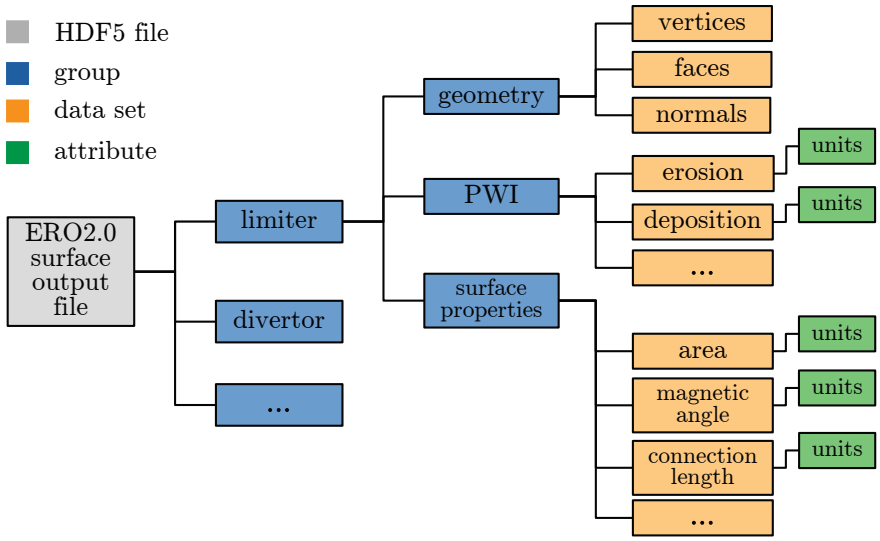


Figure 3.14: Illustration of the hierarchical data layout in HDF5 files, with the ERO2.0 surface output file as an example.

3.6 3D plasma background and domain decomposition

ERO2.0 loads the set of 3D plasma parameters ('plasma background') from an HDF5 input file. These are stored on a regular grid in 3D space. The grid is specified by its extents $(\boldsymbol{r}_{\min}, \boldsymbol{r}_{\max})$, and the numbers (n_1, n_2, n_3) of grid points (nodes) in each direction. The boundaries of this grid are also the boundaries of the simulation volume. The simulation volume is thus a cuboid, defined by two corners $\boldsymbol{r}_{\min} \leq \boldsymbol{r} \leq \boldsymbol{r}_{\max}$, in either cartesian or cylindrical coordinates (Figure 3.15).

The plasma background consists of a set of $(n_1 \times n_2 \times n_3)$-sized matrices for the plasma parameters,

- electron density n_e
- electron temperature T_e
- ion temperature T_i

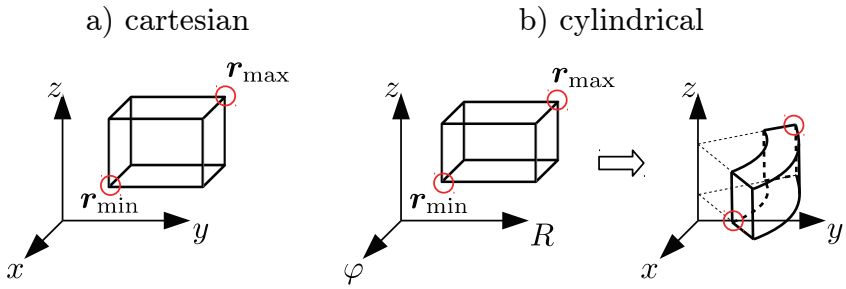


Figure 3.15: Illustration of the ERO2.0 simulation volume and grid boundaries. It is defined as a cuboid $\boldsymbol{r}_{\min} \leq \boldsymbol{r} \leq \boldsymbol{r}_{\max}$ in either (a) cartesian or (b) cylindrical coordinates.

- magnetic field $\boldsymbol{B}$
- ion flow velocity $v_{\parallel}$ (parallel to $\boldsymbol{B}$)

which are mapped to the 3D grid. The plasma parameters at a certain query point (e.g. a test particle position) are obtained using trilinear interpolation [5], for which the plasma parameters of the eight neighbor nodes surrounding the query point are required. Other grid structures are possible besides the regular grid (also called uniform grid), such as the rectilinear [85], tetrahedral [3] and hexahedral [76] grid (Figure 3.16). Such more complex grids may be more adapted in their resolution, e.g. have higher resolution in regions of large gradients in the plasma parameters, in order to achieve a more efficient storage. The regular grid was however chosen due to a number of other advantages, namely a particular simple and robust implementation as well as a very fast grid generation, interpolation and visualization.

For many applications, toroidal symmetry of the plasma may be assumed, so that the 3D plasma background can be reduced to a 2D plasma background in the (R, z)-plane (see chapter 4). This reduces not only the memory demand for the plasma background, but also the computational effort by using bilinear instead of trilinear interpolation. Note also that many plasma edge codes (e.g. SOLPS-ITER, EDGE2D-EIRENE, SOLEDGE2D-EIRENE) assume toroidal symmetry in simulations for tokamaks such as JET or ITER, and thus provide 2D plasma backgrounds.

However, in many other situations one is either dependent on 3D plasma backgrounds (e.g. for stellarators such as W7-X), or needs to account for 3D effects in tokamak plasma backgrounds. The latter might be either introduced in a post-processing step to 2D plasma backgrounds (see chapter 4), or directly provided with a 3D edge fluid code such as EMC3-EIRENE. One example is the

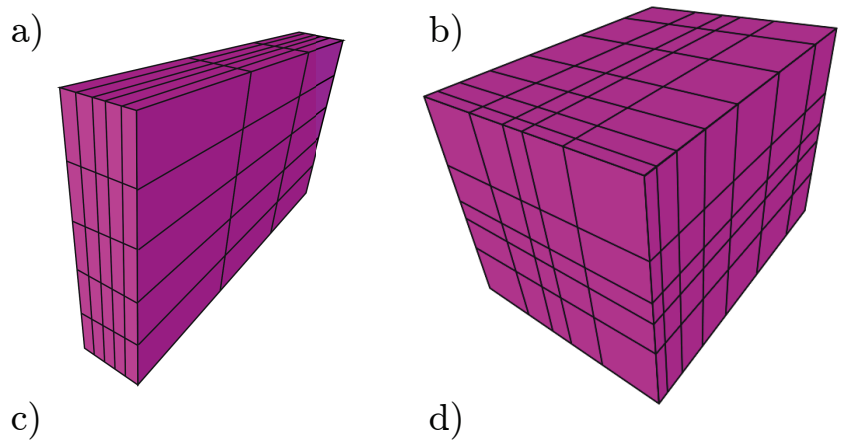


Figure 3.16: Illustration of different 3D grid structure types: (a) regular, (b) rectilinear, (c) tetrahedral and (d) hexahedral grid. Image sources: (a)-(b) `https://en.wikipedia.org/wiki/Regular_grid`, (c) [3], (d) [76].

3D distribution of plasma flow in the shadowed zone (i.e. behind the poloidal limiter ridge) of JET, as discussed in chapter 4. Other examples might be ELM filaments [77, 152, 177] or the magnetic ripple [154, 159]. In such situations where an ERO2.0 simulation is to be performed with a 3D plasma background, the grid may be very large and pose a bottleneck for memory.

A hypothetical ERO2.0 simulation with a 3D plasma background for ITER is considered to estimate the memory demand (Figure 3.17). ITER has a major radius of 6.2 m and a minor radius of 2 m. To cover the entire device, the simulation volume sizes should be approximately 2 m in radial direction, 8 m in z-direction and $6.2\,\mathrm{m} \times 2\pi \approx 40\,\mathrm{m}$ in toroidal direction. The simulation volume is then $640\,\mathrm{m}^3$. A grid resolution of 1 cm (cell size of $10^{-6}\,\mathrm{m}^3$) would then require 640×10^6 nodes.

Each node stores seven double-precision floating-point numbers (see the bullets above, with $\boldsymbol{B}$ having three components), each requiring eight bytes. The entire plasma background with such resolution requires 36 GB of memory. This is certainly possible in terms of file storage.[31] However, for most CPUs such large data exceed the limits of their main memory. For instance on JURECA, most nodes have 128 GB of memory, see section 3.2.3. If pure MPI is used and the memory is split among the 24 processes of a node, and each process requires its own copy of the data, the program will crash.

In order to avoid this limit, ERO2.0 introduces a spatial partitioning of the simulation volume into a regular grid of domains (domain decomposition, Figure 3.17). Each domain contains multiple smaller grid cells of the plasma background. Thus, instead of loading the entire data into memory at the beginning of the simulation, only smaller chunks of data (each corresponding to one domain) are loaded into memory when required. For instance, when a test particle crosses a domain boundary, the data from the last domain are replaced in memory by the data in the new domain. This is done by special HDF5 functions (partial I/O) designed for reading chunks from a data set of a file.

This domain decomposition must not be confused with the spatial partitioning with the octree described above in section 3.4. The latter is related to the wall geometry and is meant to increase the performance of wall distance and intersection queries, while the former is related to the plasma background and is meant to reduce the memory consumption.

It should also be noted that domain decomposition is a technique widely used in parallel programming [97, 130], often in the context of computational fluid dynamics [57] e.g. for climate modeling [7]. In many such applications, systems of partial differential equations are numerically solved on grid nodes. The grid size is thus proportional to both computational effort and memory consumption. Therefore, each domain is assigned to a single MPI process, which computes the solution for this domain independently (however, grid data need to be exchanged between processes at domain boundaries). In contrast, in ERO2.0 the grid is required for data storage only, therefore its size is not directly related to computational effort. The latter is instead related to the number of test particles, which is why these are distributed among processes (instead of domains). Consequently, domains in ERO2.0 are not bound to processes, and the domain decomposition works

[31] HDF5 files are in principle not limited in size. However, many computer filesystems have a maximum allowed file size of 2 GB. This may be circumvented using the so-called 'file family' driver of HDF5, which can be used to split a large file into multiple smaller files which are still treated as a single file in the HDF5 API. See `https://support.hdfgroup.org/HDF5/doc1.6/UG/08_TheFile.html`.

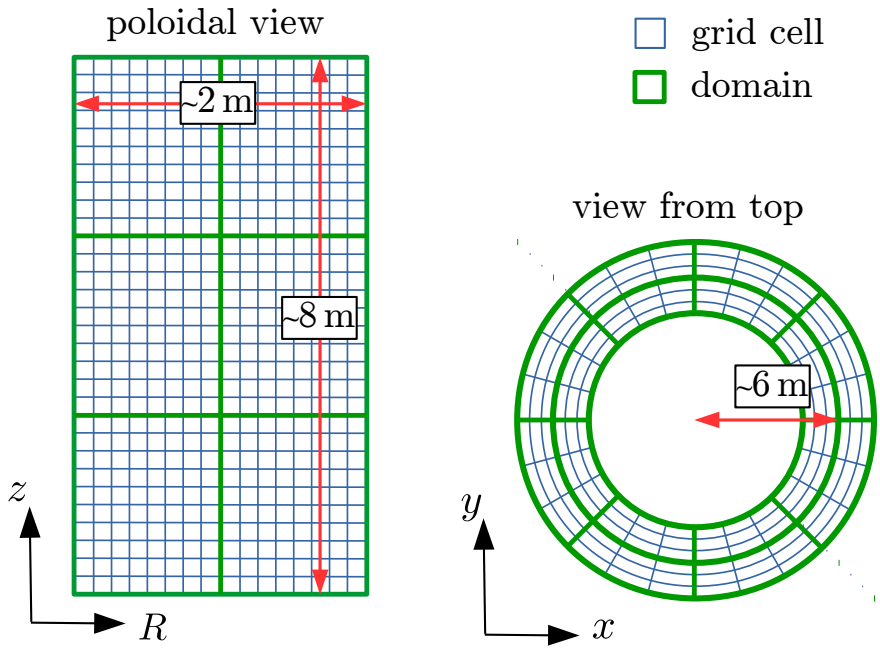


Figure 3.17: Schematic illustration of the plasma background grid and domain decomposition for a hypothetical ERO2.0 simulation for ITER. The grid resolution is made coarse for better visibility. The ITER poloidal cross-section image, with a SOLPS grid (black), is taken from [113].

independently of the parallelization.

While domain decomposition does reduce memory consumption in ERO2.0, it also introduces a computational overhead due to the need to read plasma background data chunks from the file multiple times (e.g. each time a test particle crosses a domain boundary). This overhead depends on the number of domains in a non-trivial way. If many small domains are used, smaller chunks of data need to be read from the file, however this needs to be done more often, since test particles cross the domain boundaries more frequently. Also, one needs to consider that domains themselves carry information, a too large number of domains may therefore increase the total memory consumption. In order to investigate the dependence of ERO2.0 memory consumption and code run time on the number of domains used in the decomposition, a profiling was performed for the JET simulations from chapter 4. The number of domains in each direction was varied as shown in Table 3.1. The total number of grid points in the plasma background is about 3×10^6, the total volume of the cylindrical volume is $160\,\mathrm{m}^3$.

Figure 3.18 shows the results of the profiling. The memory profiling in Fig-

case	1	2	3	4	5	6	7	8	9	10
n_x	1	2	3	2	4	4	8	16	32	64
n_y	1	2	3	1	1	4	8	16	32	64
n_z	1	2	3	2	4	4	8	16	32	64
n_{tot}	1	8	27	4	16	64	512	4.1×10^3	3.2×10^4	2.6×10^5

Table 3.1: Number of domains in each direction (n_x, n_y, n_z) and in total ($n_{\mathrm{tot}} = n_x n_y n_z$) used for the profiling.

ure 3.18(a) was performed using the *Valgrind* tool (see appendix A). ERO2.0 allocates the arrays in which plasma background data are stored on a memory area called the heap. For the profiling shown here, the peak heap memory usage is taken as a measure for memory consumption. As expected, the memory consumption at first decreases with the number of domains. The region of lowest memory consumption lies between 10^2 and 10^4 domains (which corresponds to 3×10^4 and 3×10^2 grid points per domain and domain sizes of $1.6\,\mathrm{m}^3$ and $0.016\,\mathrm{m}^3$, respectively). At this point, the peak heap memory usage is reduced to about 6–8 % of the usage without domain decomposition. When further increasing the number of domains ($\gtrsim 10^4$), the storage required for the domains themselves becomes dominant and the total memory consumption increases again.

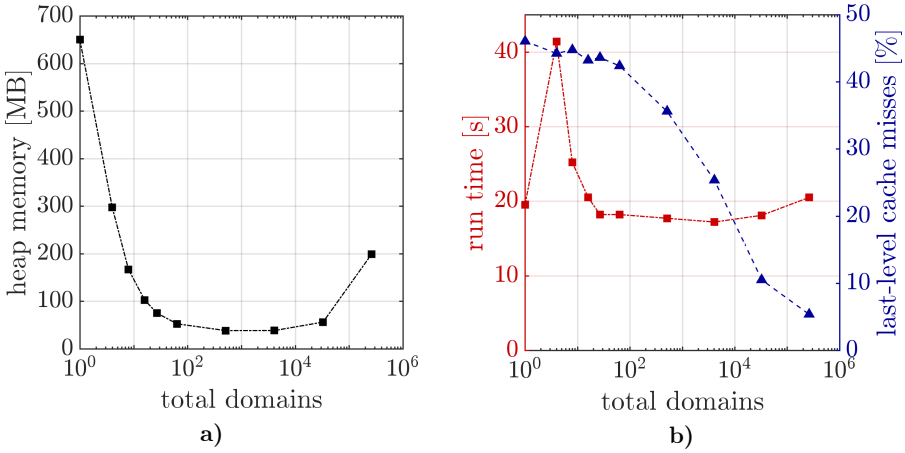


Figure 3.18: Profiling of ERO2.0 for different number of domains. (a) Peak heap memory consumption measured with Valgrind. (b) Code run time (squares) and last-level cache misses measured with Perf (triangles).

The dependence of code run time is shown in Figure 3.18(b). The impact of the domain subdivision on run time is generally less pronounced than the effect on memory consumption, except that there is a sharp increase of about 200 % when introducing domains in small numbers ($\lesssim 10$). This indicates that if domain

decomposition is used, the domain sizes must not be chosen too large, otherwise the reading of large data chunks from the plasma background file introduces a computational overhead in the transport loop. Interestingly, a minimum in run-time is observed around 10^2–10^4, which corresponds to the region of lowest memory consumption in Figure 3.18(a). In this region, the run-time is even about 12 % lower than for the case without domain decomposition. The cache profiling with the *Perf* tool (see appendix A) shows that for simulations with large numbers of domains, the number of last-level cache misses is reduced by a factor of 10 compared to the simulation without domain decomposition. This means that because memory blocks are smaller if domain decomposition is used, the cache is used more efficiently, which results in an overall decrease in code run-time.

In summary, it is observed that domain decomposition with a medium number of domains leads to a significant reduction in memory consumption and also to a mild decrease of code run-time. This way, ERO2.0 is suited to efficiently perform full-device simulations with fine plasma backgrounds not only for small and medium devices such as PSI-2 or W7-X, but also for large devices such as ITER.

3.7 Sparse matrices for storing 3D impurity and emission densities

During test particle transport, ERO creates 3D arrays on a grid in which particle densities are stored. Individual arrays are created for different charge states Z of each impurity. From the particle densities, ERO additionally calculates light emission densities, for a user-specified number of emission lines, using photon emissivity coefficients (see section 2.3.4). These data can be used to verify the code with experimental data from spectroscopy, e.g. using line-of-sight integrated intensities [23], 1D profiles [206] or 2D patterns [68]. Typically a resolution in the order of 1 mm in each direction is required for a meaningful comparison with experiment.

The concept of particle density recording is illustrated in Figure 3.19(a) for an ERO2.0 simulation of Be transport in JET limiter plasmas, similar to those described in chapter 4. The trajectory of a single Be test particle is shown in the (R, Z)-plane, together with the corresponding particle density after averaging in toroidal direction and adding the contributions of all particle charge states Z. The trajectory covers only a minor fraction of the grid cells, i.e. the density D of the matrix (defined as the fraction of non-zero matrix entries) is much smaller than unity. In 3D (not shown), the density is even much lower. The 3D matrices are

therefore sparse. For many ERO applications, this remains true even for very large numbers of test particle trajectories, because certain regions of the simulation volume are never reached by test particles. In general, the sparsity also increases with the grid resolution. In Figure 3.19(a), a very coarse grid resolution of 10 cm is selected for better view.

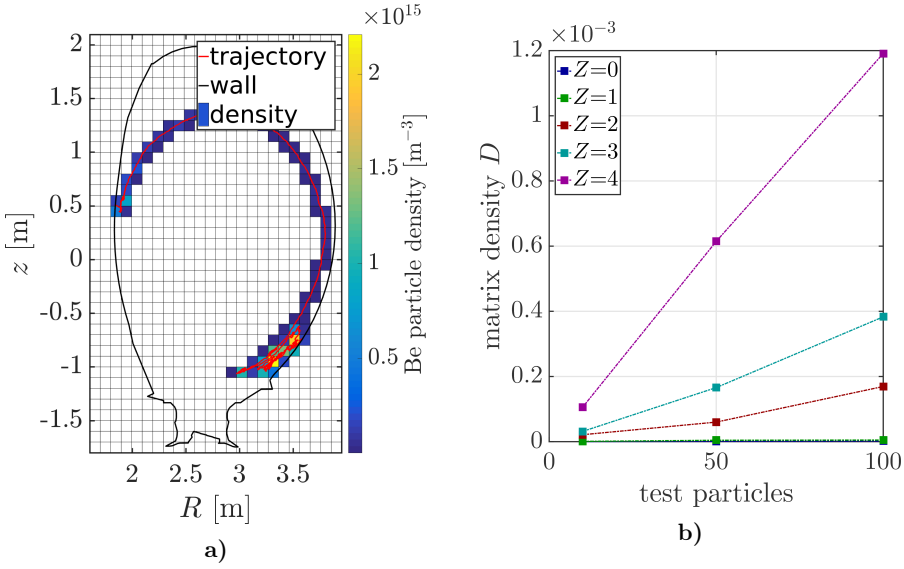


Figure 3.19: (a) Illustration of the sparsity of particle density data by ERO2.0 (viewed in (R, Z)-plane after averaging in toroidal direction): a single Be test particle trajectory creates markers in only a small fraction of grid cells. For simplicity, the matrices for all Be charge states Z are added up. A very coarse grid resolution of 10 cm is chosen for better view. (b) Density of the matrices for Be at different charge states Z, depending on the number of test particles used for the ERO2.0 simulation. Here the grid resolution is 2.5 mm.

Similarly to the 3D plasma backgrounds described above, the 3D particle and emission densities are a potential memory bottleneck in ERO2.0 if simulations in a large volume and with a fine grid resolution are desired. The memory required for a dense matrix of doubles is given by

$$M_{\text{dense}} = N M_{\text{double}} \tag{3.5}$$

where N is the matrix size and M_{double} is the memory size of a double. Also, if the density data are to be collected by an MPI master process in a parallel ERO2.0 simulation (e.g. for writing a 3D output file containing the densities), the communication of such large data sets may quickly become a performance bottleneck.

To avoid this bottleneck, a sparse matrix representation was implemented in ERO2.0 which replaces the usual dense matrix representation. ERO2.0 uses the dictionary-of-keys (DOK) format, in which the matrix index triples (i, j, k) are mapped to corresponding non-zero matrix entries [212]. Entries which are missing from the dictionary are taken to be zero. The memory required to store a matrix in this format is thus

$$M_{\mathrm{DOK}} = ND(M_{\mathrm{double}} + 3M_{\mathrm{int}}) \tag{3.6}$$

where $0 \leq D \leq 1$ is the matrix density (i.e. the fraction of non-zero entries), and $3M_{\mathrm{int}}$ is the memory required to store an index triple of integers (i, j, k). Assuming $M_{\mathrm{double}} = 2M_{\mathrm{int}}$, the memory ratio of DOK to dense storage becomes

$$M_{\mathrm{DOK}}/M_{\mathrm{dense}} = 2.5D\,. \tag{3.7}$$

If $D \ll 1$, the storage size can thus be greatly reduced using the DOK layout.

In order to estimate the typical matrix density, a number of the JET simulations as in chapter 4 were performed with increasing number of test particles and a grid resolution of 2.5 mm. Figure 3.19(b) shows the results for the matrices, which store the Be particle densities with different charge states Z. The matrix density increases with Z, because particles are ionized quickly, so that trajectory parts with low Z are more localized at the particle origins (see chapter 4). Also, the proportionality of the matrix density to the test particle number, which is roughly linear, is well visible. Even for the highest density of 1.2×10^3, which is obtained for $Z = 4$ for 100 test particles, the DOK layout is still beneficial to memory. For larger numbers of test particles in an ERO2.0 simulation one would use parallelization, so that the number of test particles (and thus the matrix densities) per MPI process will still remain low. In this case, the DOK layout is especially beneficial, since the MPI processes need to communicate less data.

3.8 Magnetic field line tracing

The magnetic field line tracing is a new algorithm implemented in ERO2.0 that allows calculating the distance s between a point in the SOL and a wall component along the field line going through the point. This can also be used to calculate the magnetic connection length L, which is defined as half of the distance between two plasma-facing components along a closed magnetic field line, and is typically in the order of several meters in present fusion devices [182, p. 17]. Due to the smaller

simulation volume size, calculating L was generally not possible with ERO1.0, and was either taken as a constant parameter or obtained from other codes [21, 115] such as PFCFlux [73].

The parameters s and L can be used e.g. for calculating the plasma flow velocity in the SOL (see section 4.2.2). Also, L is used as a correction factor to the incident particle and heat fluxes on the surface (shadowing model, see section 4.2.3 and section 4.2.5).

The algorithm required for the field tracing is to start from a point in 3D space and to numerically step along the magnetic field direction until an intersection with a surface element is found. The intersection algorithms for line segments with polygon meshes are implemented in ERO2.0 (see section 3.4), as well as the interpolation of the 3D magnetic field (see section 3.6). The stepping along the magnetic field is implemented using Runge-Kutta (RK) algorithms up to the 2nd order.

Figure 3.20 shows an example of a field line traced with ERO2.0 using the Euler (RK-1) and Heun (RK-2) methods. The starting point for the tracing lies inside the separatrix and must therefore lead to a closed field line. The maximum distance for tracing was 300 m, which corresponds to about 4.5 poloidal rotations. For the step size of 1 cm used, the Heun method is significantly more precise, which leads to a almost perfectly closed field line, and is therefore the recommended algorithm.

Figure 3.21 shows the connection lengths calculated on the JET polygon mesh. The results are used in section 4.2.3 and section 4.2.5 in order to calculate the shadowing patterns. For each polygon, the connection length is determined by tracing the field line starting from its center. A maximum tracing distance (threshold) of $L_{\max} = 6\,\mathrm{m}$ is used. As described in section 4.2.3, all areas with higher connection lengths than this threshold are assumed to be plasma-wetted.

Figure 3.22 shows connection length patterns in the JET (R, z)-plane. The results are used in section 4.2.2 in order to calculate the plasma flow velocity using the simple SOL model. Here, a threshold of $L_{\max} = 200\,\mathrm{m}$ is chosen, because it was found that L diverges as one moves closer to the separatrix (without any threshold, the computational effort increases dramatically).

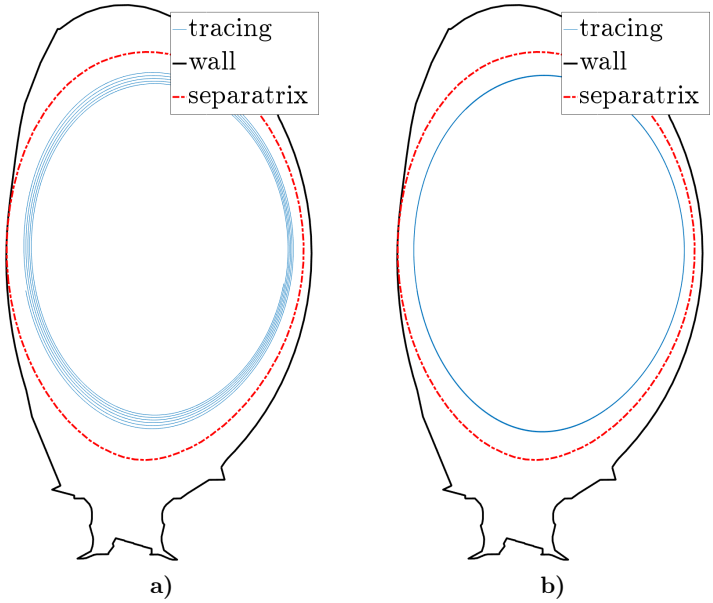


Figure 3.20: Example of ERO2.0 magnetic field line tracing in the JET (R, z)-plane using the magnetic configuration of JET pulse number #80319 at $t = 50\,\mathrm{s}$. The step size is 1 cm. The field line tracing is performed inside the separatrix up to a length of 300 m. (a) Euler method, (b) Heun method.

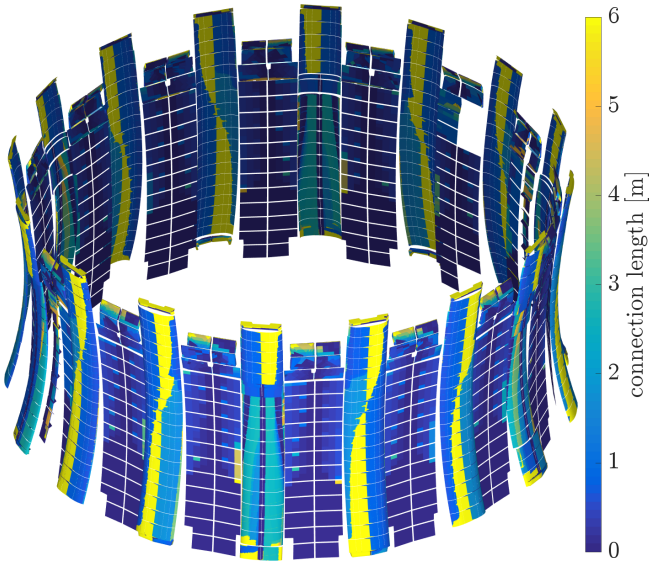


Figure 3.21: Connection lengths in the JET (R, z)-plane calculated with ERO2.0 for the magnetic configuration of JET pulse number #80319 at $t = 50\,\mathrm{s}$. The field line tracing threshold is $L_{\max} = 6\,\mathrm{m}$.

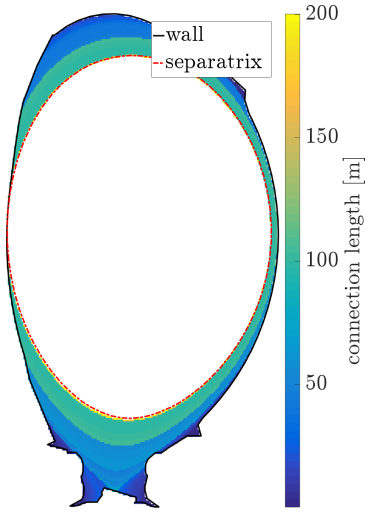


Figure 3.22: Connection lengths in the JET (R, z)-plane calculated with ERO2.0 for the magnetic configuration of JET pulse number #80319 at $t = 50\,\mathrm{s}$. The field line tracing threshold is $L_{\max} = 200\,\mathrm{m}$.

Chapter 4

ERO2.0 modeling of Be erosion at JET-ILW

The JET ITER-like wall (ILW), comprising Be in the main chamber and W in the divertor, was installed at JET by replacing the carbon wall components during a 15-month shutdown from 2009 to 2011. An overview over the ILW and its experimental program can be found e.g. in [91] and [125]. Figure 4.1 shows important PFCs and the materials used for them. The most relevant components for the discussion in this chapter are the inner wall guard limiters (IWGLs).

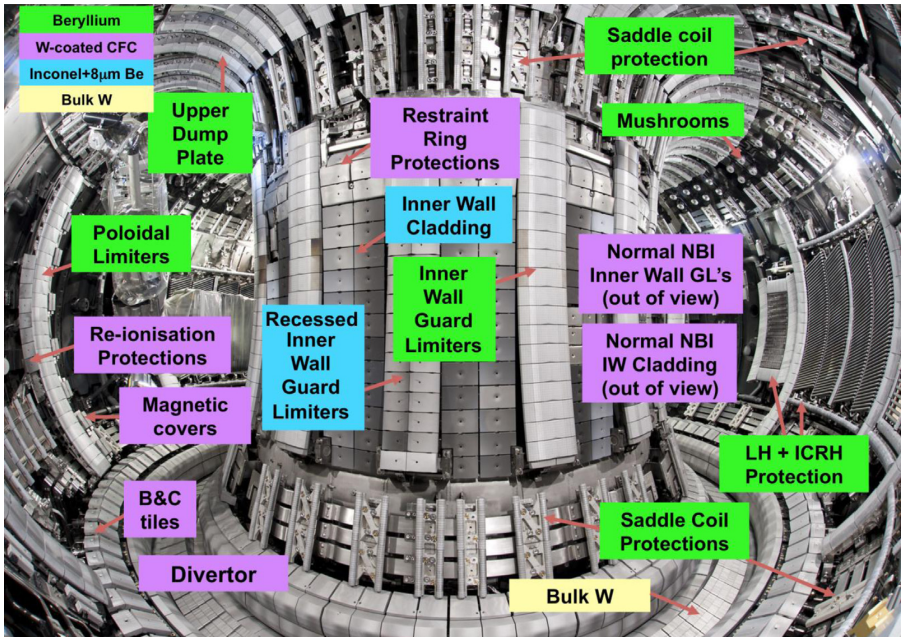


Figure 4.1: Distribution of materials in the JET-ILW. Image taken from [91].

Shortly after installation of the ITER-like wall, dedicated experiments were carried out in order to investigate the erosion of the Be limiters [35]. The experiments aimed at determining the effective total sputtering yield (ETSY) for plasma-wetted Be surface areas by spectroscopic means, and understanding the composition of this yield from various erosion processes such as sputtering by deuterons, Be self-sputtering and CAPS. The ETSY is determined by measuring the radio of eroded Be to incident deuterons in a spectroscopy observation spot on an Be IWGL tile, as discussed shortly in more detail. In this context, 'effective' means that the respective fluxes are integrated over the observation spot area (not to be confused with the effective sputtering yields $Y(E_{\mathrm{in}}, \theta_{\mathrm{in}})$ in section 2.2.1), and 'total' means that the contributions by the different erosion mechanisms are added.

The experiments were accompanied by 'local' ERO1.0 modeling, which is described in [21, 23, 24]. Here, 'local' means that the ERO1.0 simulation volume covers three limiter tiles, namely the one on which the spectroscopic observation spot was located and its two neighboring tiles in poloidal direction (see Figure 4.2, Figure 4.6). This allows to directly calculate the Be sputtering by deuterons on the observation spot using the equations (2.5) and (2.16). However, the limited size of the ERO1.0 simulation volume (due to the technical reasons discussed in chapter 3) poses certain problems to the proper treatment of Be self-sputtering. Namely, most test particles ($\sim 90\,\%$) representing eroded Be in ERO1.0 leave the simulation volume quickly along magnetic field lines. Vice versa, the Be particles which are eroded outside the simulation volume but might reach the observation spot along field lines (and contribute to self-sputtering therein) are not accounted for. In this sense the treatment of Be transport and self-sputtering in ERO1.0 is not self-consistent. Instead, a parameter study was performed in [21, 24], in which different levels of (spatially homogeneously distributed) Be impurity concentration and charge state distribution were assumed in the plasma.

In this chapter, the new ERO2.0 'global' modeling approach is presented, which extends the local ERO1.0 modeling.[32] The new approach embraces the enhanced capabilities of ERO2.0 in order to increase the simulation volume substantially, so that the entire plasma edge is covered (see Figure 4.2(a)). This provides the advantage of treating Be transport and self-sputtering in a self-consistent way. The Be impurity concentration and charge state distributions are thus an outcome of the modeling rather than a free parameter. Also, additional experimental diagnostics (IR cameras, wide-view cameras), which require a large simulation volume for a meaningful comparison with their synthetic counterparts, are now used to validate the modeling. This cross-checking of different diagnostics is used

[32] Part of the results have recently been published in [164].

to validate different modeling aspects (plasma backgrounds, sputtering yields and atomic data used for input) in order to reduce the model uncertainties.

The chapter is structured as follows: In section 4.1, a description of the above-mentioned Be erosion experiment is given. In section 4.2, the new global modeling approach is described in detail and its results are presented and discussed. Finally in section 4.3, the ETSY is compared between experiment and the local and global ERO modeling approaches.

4.1 Beryllium erosion experiments at JET-ILW

The above-mentioned Be erosion experiment from [35] was performed in limiter configuration with purely ohmic heating and with the plasma contact point at the HFS close to the midplane. Seven pulses were performed (JPN #80319-80323, 80272 and 80274). The level of deuterium fueling was varied between the pulses, while keeping all other parameters (such as the magnetic configuration) constant. Thereby, the plasma density and temperature were varied, allowing to determine the ETSY as a function of the deuteron impact energy.

By passive spectroscopy measuring the line emission of D and eroded Be, the Be ETSY could be determined *in situ*. The intensity of spectroscopic lines was measured using the KS3 diagnostic [100] with a horizontal line-of-sight (LOS). Figure 4.2(a)-(c) show the location of the spectroscopy observation spot (OS), which is located at the inner midplane on tile 7 of the IWGL in JET octant 7X. The plasma contact point in the density scan pulses was located at $z = 50\,\mathrm{cm}$ above the midplane on the HFS.

Figure 4.2(c) shows the spectroscopy LOS and OS as used for the ERO2.0 modeling. The coordinates of the LOS are obtained from [96]. The observation cone is approximated to a cylindrical shape, with the radius of 6 cm [23] on the OS used for the entire LOS. The part of the LOS crossing the zone of Be I and Be II light emission (in the vicinity of the inner wall) is small in comparison to its total length, therefore divergence can be neglected.

Figure 4.3 shows time traces of important experimental signals for JPN #80319 and 80323, which are the pulses with the lowest and highest fueling in the experiment, respectively. The density is inversely related to the temperature (i.e. their product, which is proportional to the plasma pressure, is roughly constant). This means that #80319 and 80323 are also the pulses with the highest and lowest temperature, respectively. Table 4.1 shows the list of JET signals used in the time traces.

The gray zone at $t = 47$–$52\,\mathrm{s}$ indicates the time window of the HFS-limited

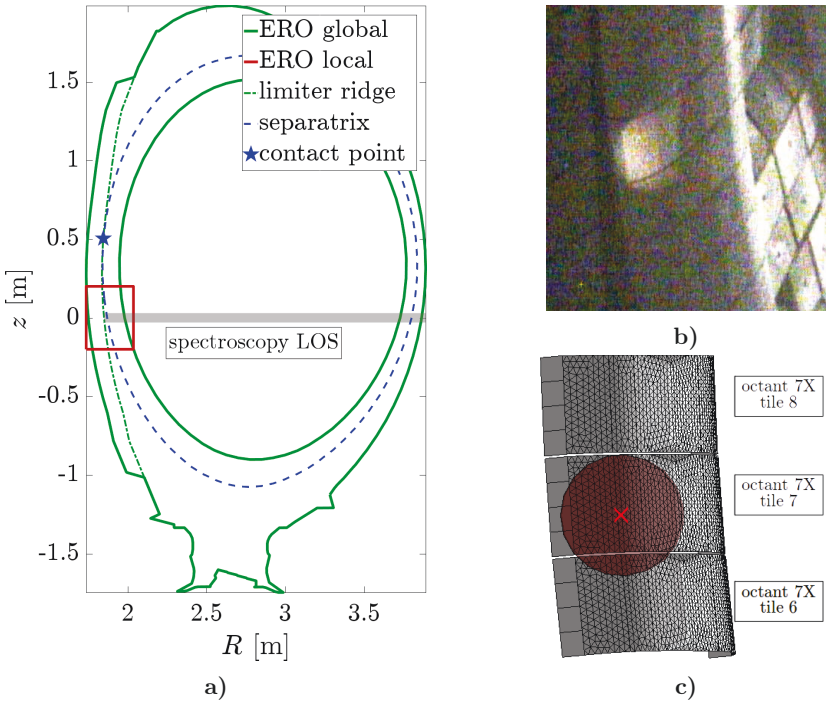


Figure 4.2: (a) Illustration of the horizontal KS3 spectroscopy LOS (gray bar) in the JET (R, z)-plane. The separatrix and plasma contact point for JPN #80319, $t = 50\,\mathrm{s}$, from the density scan experiment [35] are also shown, as well as the simulation volume boundaries for the global and local ERO2.0 modeling. (b) Location of the spectroscopy observation on the IWGL in JET octant 7X, visualized with the backscattered light spot (image source: [96]). (c) ERO2.0 approximation of the LOS by a cylinder with radius 6 cm through the points $\boldsymbol{R}_1 = (9.99, -17.50, 0)$ m and $\boldsymbol{R}_2 = (1.78, -0.31, 0)$ m (both aligned with the red cross). These coordinates were obtained from [96].

region in which the analysis was conducted.[33] In this time window, the central magnetic field strength was $B_t = 2.8\,\mathrm{T}$, and the plasma current was $I_p = 2.0\,\mathrm{MA}$. The inner midplane gap between the wall (and thus the spectroscopy OS, which is also located at the inner midplane) and the separatrix was $R_{ig} = 19.5\,\mathrm{cm}$. The total radiated power was varying within about $P_{\mathrm{rad}} = 0.1\text{–}0.3\,\mathrm{MW}$. The line-averaged electron density and central electron temperature were varying within about $\langle n_{\mathrm{e}}^c \rangle = 1.0\text{–}2.6 \times 10^{19}\,\mathrm{m}^{-3}$ and $T_{\mathrm{e}}^c = 1.7\text{–}4.0\,\mathrm{keV}$, respectively (see also Figure 4.5(a)).

Figure 4.4 shows further time traces, measured with the KS3 spectroscopy diagnostic (in the horizontal LOS, KS3H), for JPN #80319 and 80323. The

[33] Note that the time counting for a pulse starts before the actual plasma appears, which is then at about $t = 40\,\mathrm{s}$.

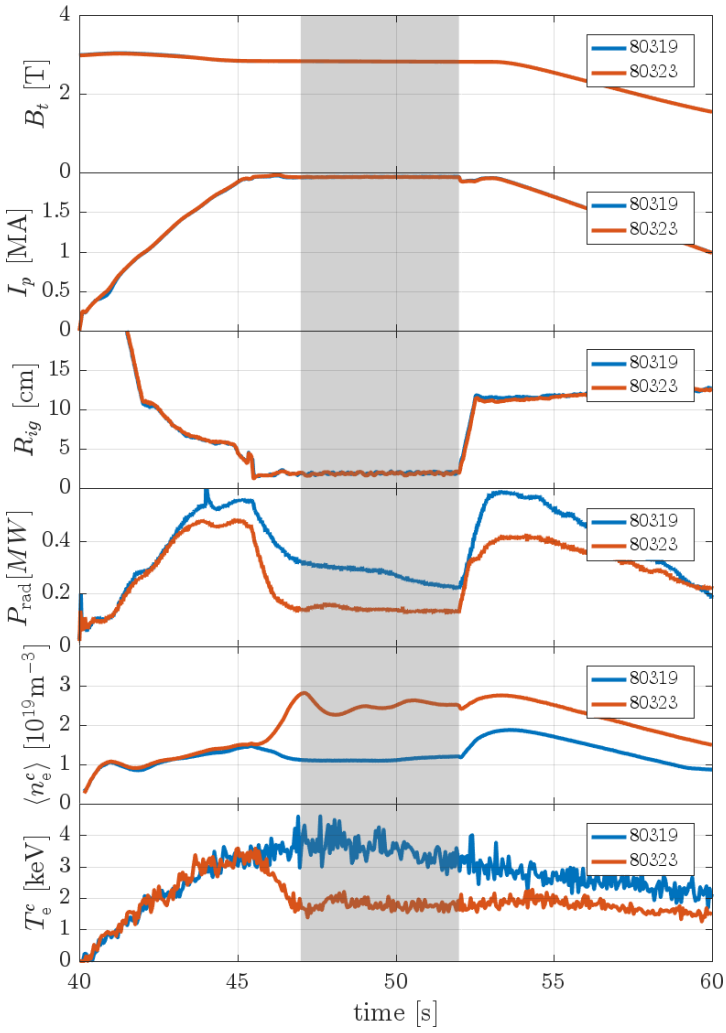


Figure 4.3: Time traces of various signals from the density scan experiment [35]. The list of corresponding JET signals is given in Table 4.1. JPN #80319 has the lowest fueling and highest temperature in the experiment. For JPN #80323 it is vice versa. The gray area indicates the time window used for the analysis.

effective charge Z_{eff} (see appendix C for the definition) is calculated from the LOS-integrated bremsstrahlung. As pointed out in [35], for the hottest plasma edge case (JPN #80319) a high effective charge up to $Z_{\mathrm{eff}} \approx 4$ indicates that a high level of Be impurity concentration is present, so that strong Be self-sputtering must take place. In contrast, for the coldest plasma edge case (JPN #80323) a low $Z_{\mathrm{eff}} \approx 1.2$ is observed, which corresponds to typical values reported for diverted plasma conditions in JET-ILW [47].

quantity	diagnostic	JET PPF
central magnetic field B_t	coils	MAGN/BVAC
central plasma current I_p	coils	MAGN/IPLA
inner midplane wall-separatrix gap R_{ig}	EFIT	EFIT/RIG
radiated power P_{rad}	bolometry	BOLO/TOPI
line-averaged central electron density $\langle n_{\mathrm{e}}^c \rangle$	interferometry	KG1L/LAD3
central electron temperature T_{e}^c	HRTS	HRTX/TE0
intensity $I(\mathrm{D}_\gamma)$	spectroscopy	KS3H/DGAM
intensity $I(\mathrm{Be\,II\ 527\,nm})$	spectroscopy	KS3H/BE2A
intensity $I(\mathrm{Be\,II\ 436\,nm})$	spectroscopy	KS3H/BE2B
intensity $I(\mathrm{Be\,II\ 467\,nm})$	spectroscopy	KS3H/BE2C
effective charge Z_{eff}	bremsstrahlung	ZEFF/ZEFH

Table 4.1: Diagnostics and JET Processed Pulse Files (PPFs) used in the time traces in Figure 4.3 and Figure 4.4.

The authors of [35] emphasize that such high Z_{eff} are outside the usual JET-ILW baseline pulses with $Z_{\mathrm{eff}} < 1.8$: only in dedicated experiments such as the density scan experiment, the full operational window is explored in order to study also the erosion for extremely high electron temperatures.

Figure 4.5(a) shows the central electron temperature T_{e}^c plotted over the line-averaged density $\langle n_{\mathrm{e}}^c \rangle$. Figure 4.5(b) shows the effective charge Z_{eff}. The values are taken from all seven density scan pulses and for various time snapshots within the time window $t = 47$–$52\,\mathrm{s}$. Both T_{e}^c and Z_{eff} are inversely proportional to $\langle n_{\mathrm{e}} \rangle$.

The local electron temperature $T_{\mathrm{e}}^{\mathrm{loc}}$ on the OS (Figure 4.5(c)) was calculated from the Be II 467 nm and 436 nm transition line intensity ratio (see Figure 4.4). The procedure for this is described in [35] (see figure 3 therein): since the ratio of the ADAS $\mathrm{PEC}(n_{\mathrm{e}}, T_{\mathrm{e}})$ is known for these lines and depends strongly on the electron temperature, the latter can be calculated for a given line intensity ratio. Assuming a typical local electron density $n_{\mathrm{e}}^{\mathrm{loc}} = 10^{18}$–$10^{19}\,\mathrm{m}^{-3}$ on the OS, one can estimate $T_{\mathrm{e}}^{\mathrm{loc}}$ to be 10–35 eV.

Figure 4.5(d) shows the Be ETSY $Y_{\mathrm{eff}}^{\mathrm{tot}}$ on the spectroscopy OS. It is calculated as

$$Y_{\mathrm{eff}}^{\mathrm{tot}} = \frac{\int_{\mathrm{OS}} \mathrm{d}A\, \Gamma_{\mathrm{Be}}^{\mathrm{ero}}}{\int_{\mathrm{OS}} \mathrm{d}A\, \Gamma_{\mathrm{D}}^{\mathrm{in}}}\,, \tag{4.1}$$

where $\int_{\mathrm{OS}} \mathrm{d}A$ indicates integration over the OS. The impinging deuteron flux $\Gamma_{\mathrm{D}}^{\mathrm{in}}$ and the eroded Be flux $\Gamma_{\mathrm{Be}}^{\mathrm{ero}}$ in equation (4.1) are calculated from the D_γ and Be II 527 nm emission line intensities, respectively. For converting intensities to fluxes, ionization per photon (so-called S/XB [45, pp. 152-153]) coefficients obtained from ADAS are used. Note that the ETSY (4.1) contains contributions from different Be sputtering processes, i.e. physical sputtering and CAPS.

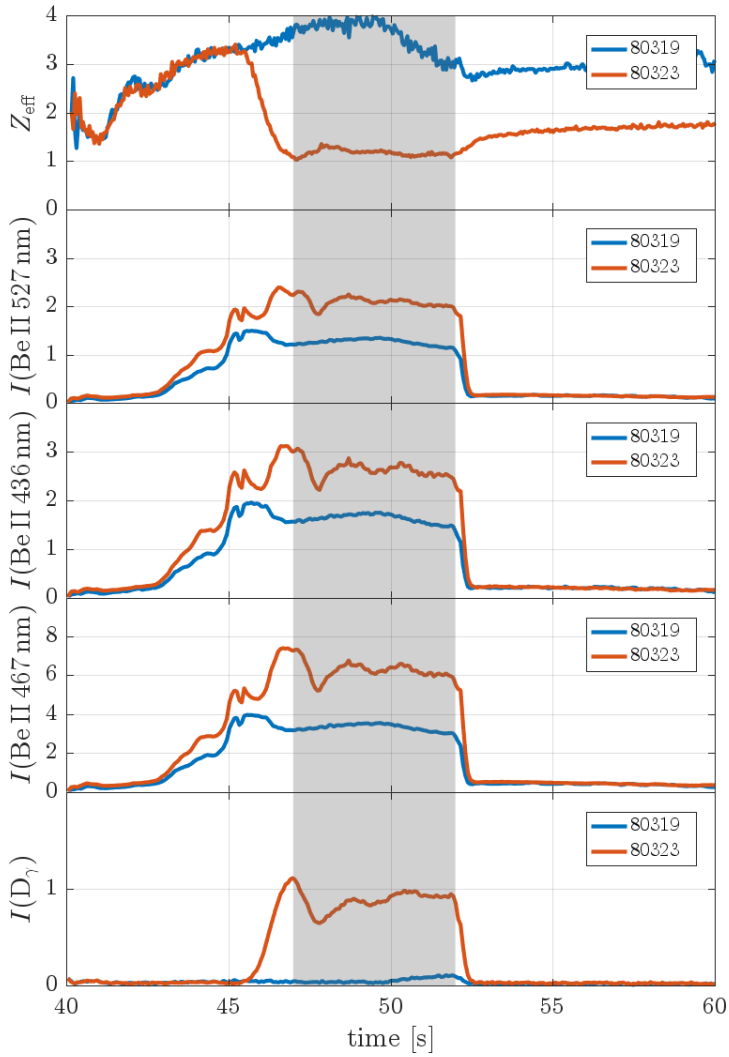


Figure 4.4: Time traces of spectroscopy signals (KS3) from the density scan experiment [35], again for JPN #80319 (lowest fueling, highest temperature) and JPN #80323 (vice versa). The list of corresponding JET signals is given in Table 4.1. Z_{eff} is calculated from the LOS-integrated bremsstrahlung. The Be II and D$_\gamma$ intensities are given as LOS-integrated photon fluxes in $[10^{14}\frac{\mathrm{ph}}{\mathrm{s\,cm^2\,sr}}]$.

Figure 4.5(a)-(d) illustrates that there is a clear correlation between electron temperature, effective charge and ETSY. As the local electron temperature $T_{\mathrm{e}}^{\mathrm{loc}}$ increases from about 10–35 eV, the sputtering yields for D and Be impact increase as well. This can be seen e.g. in Figure 2.5 after considering a magnetic field angle on the limiter of about 75° on the OS. Due to stronger Be erosion by D

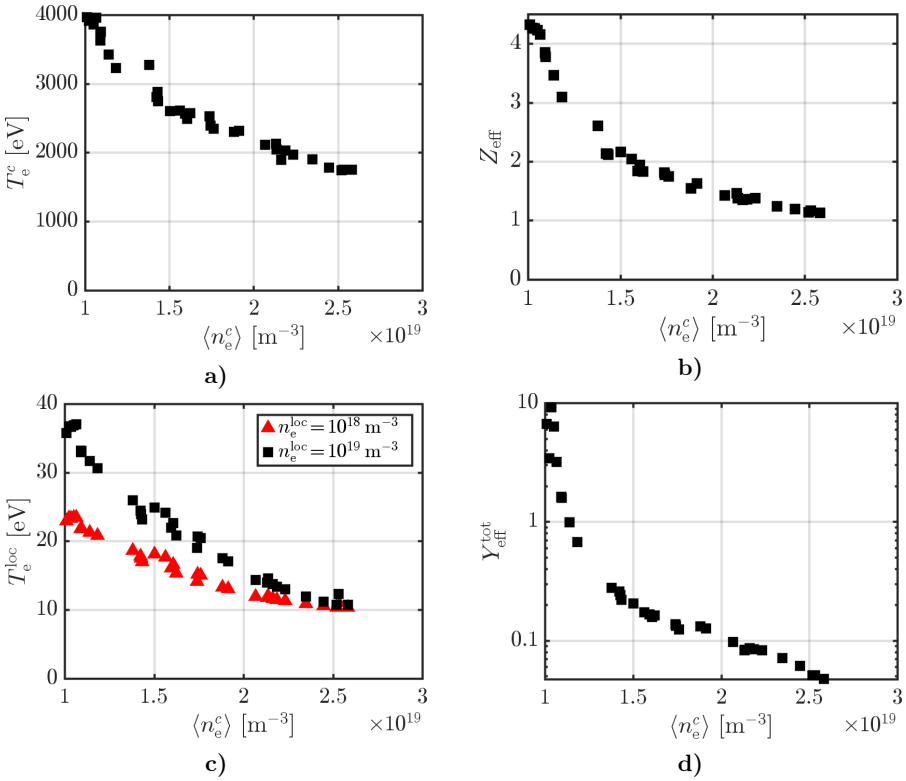


Figure 4.5: Various measurements from the density scan experiment, plotted over the line-averaged density $\langle n_e^c \rangle$. (a) Central electron temperature T_e^c. (b) Effective charge Z_{eff}. (c) Local temperature T_e^{loc} on the spectroscopy OS, determined from Be II line ratios, with two different assumptions of the local electron density n_e^{loc}. (d) Effective yield Y_{eff}^{tot} on the spectroscopy OS (semi-logarithmic plot), calculated from D_γ and Be II line ratios and the S/XB coefficients.

impact, the Be plasma concentration and thus Z_{eff} increases, leading to a rising self-sputtering contribution to erosion in a Be-rich plasma. Starting at about $n_e^{loc} < 1.2 \times 10^{19}\,cm^{-3}$ ($T_e^{loc} > 20$–$30\,eV$) the self-sputtering dominates the erosion: a sharp increase in the ETSY, with $Y_{eff}^{tot} > 1$, is observed.

4.2 Global ERO2.0 modeling

The first global ERO2.0 modeling for JET-ILW presented in this section can be seen as an extension of the modeling efforts from [24], but with the simulation volume increased substantially as illustrated in Figure 4.2(a). While the ERO1.0 volume was limited to the close surroundings of the spectroscopy OS, the ERO2.0

volume covers the entire (in both poloidal and toroidal directions) edge of the plasma. The simulation volume is limited only in radial direction by the wall (outer boundary) and a plasma flux surface that marks the transition to the plasma core (inner boundary) at a given minor radius. The impact of the boundary conditions on the Be transport results is discussed below in section 4.2.4.

The main purpose of this study is to demonstrate the novel aspects of the global modeling that were not possible to study with the local one. The focus is therefore on (1) a general description of the global modeling results, such as the PWI on a larger set of PFCs and the long-range migration of Be, (2) the new possibilities for experimental verification using 2D images from IR and spectroscopic wide-view cameras, and (3) the impact of a self-consistent treatment of Be self-sputtering on the modeling results for the ETSY.

4.2.1 Wall geometry

The polygon mesh for JET wall components that was used in this work is shown in Figure 4.6 and was obtained from the PFCFlux code database[34] [73] and post-processed for ERO2.0 requirements, e.g. redundant polygons on the back of the PFCs were removed. The inset in the red box is a magnification of the three limiter tiles on the spectroscopy OS, which were investigated in the earlier local ERO1.0 modeling [24]. It also shows the edges of the polygons in order to illustrate the typical polygon resolution of ca. 0.7 cm edge length. A finer mesh with more polygons can be used. However, a sensitivity scan showed that the impact on surface-integrated results (e.g. the total number of eroded atoms) is negligible, wherefore the resolution shown in the figure is sufficient. Note also that some finer details such as the castellation of Be tiles [4] are not described by the polygon mesh and are therefore neglected in this work. The effect of castellated structures on impurity deposition has been previously addressed by the MC neutrals transport code 3D-GAPS [127, 128] and may be investigated with ERO2.0 in the future, but is not the focus of the present first ERO2.0 modeling.

The color coding in Figure 4.6 indicates the material composition of the wall elements. The majority of tiles from the IWGLs, which are the main erosion zones in limiter plasmas, are made of bulk Be. In the vicinity of NBI shinethrough areas, tiles from IWGL center sections are recessed. These tiles are made of bulk CFC or Inconel and clad with W and Be, respectively [125]. These are treated as bulk W or Be tiles in the current simulations, so the distinction is made in the figure solely for the sake of a better visibility of the recessed tiles. As wide-view camera

[34] Courtesy Dr. Mehdi Firdaouss from CEA, France.

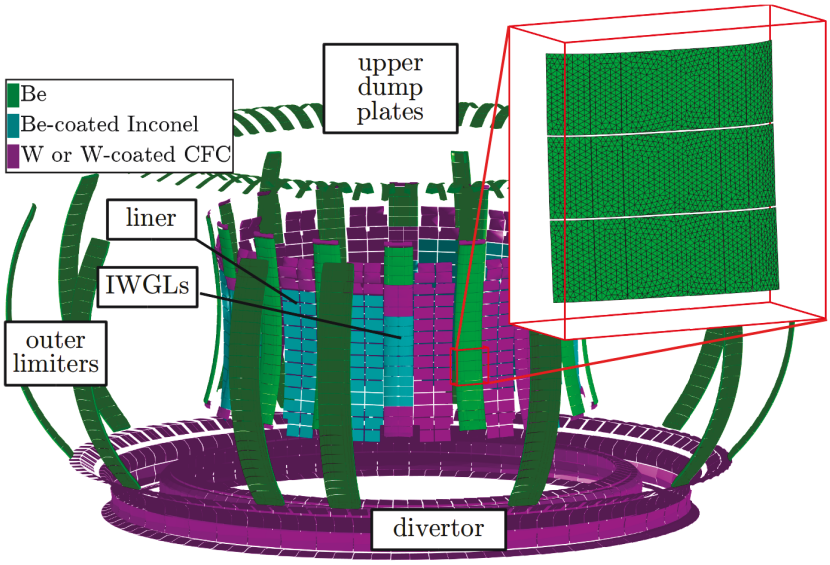


Figure 4.6: 3D view of the polygon mesh used by ERO2.0, representing selected wall components of the JET-ILW. Polygons are color-coded with material composition as indicated in the legend. Note that in the actual ERO2.0 simulations, the coated limiter tiles are treated as solid Be or W tiles. The inset shows the magnified IWGL tiles (octant 7X, tiles 6-8) included in earlier ERO1.0 simulations [24]. It also illustrates the typical polygon resolution.

images indicate (see section 4.2.6), the erosion occurs predominantly at the bulk Be tiles of the IWGLS. Therefore, these are of primary interest in this work, since the erosion of the recessed tiles is negligible.

The upper dump plates, the outer limiters and the divertor are neither expected to affect the general picture of the global modeling, and in particular not the erosion on the spectroscopy OS. The respective 3D polygon mesh parts representing these components are therefore omitted from the modeling to ensure an optimum usage of computational resources. Instead, they are described as being a part of the outer 2D simulation volume boundary as shown in Figure 4.2(a). In this simplified description, the upper dump plater, outer limiters and divertor act as a sink for Be test particles, while their erosion is assumed to be zero.

4.2.2 Plasma backgrounds

The calculation of input plasma backgrounds described in this section largely follows the procedure used for the earlier local ERO1.0 modeling in [24], with two exceptions. First, the plasma backgrounds are now used not just for the vicinity

of the spectroscopy OS, but for the entire (R, z)-plane of JET (see Figure 4.2(a)). Second, plasma backgrounds are created not only by the previously used two-point model, but also using the plasma edge transport code SOLEDGE2D-EIRENE, which gives some different results as discussed below.

For the present modeling, toroidal symmetry of the plasma background (including the magnetic fields) is assumed unless specifically mentioned otherwise, which means that 2D plasma parameter maps in the (R, z)-plane are required.

The magnetic field $\boldsymbol{B}$ is obtained from the EFIT (Equilibrium FITting) code for JET [36]. Note that the magnetic configuration in the density scan experiment is identical for all pulses (in the time window $t = 47$–$52\,\mathrm{s}$ of the analysis), so that taking the magnetic field for a single pulse snapshot (e.g. JPN #80319, $t = 50\,\mathrm{s}$) is sufficient.

The electron density and temperature are at the first stage obtained as 1D radial profiles $n_\mathrm{e}(\rho)$ and $T_\mathrm{e}(\rho)$, where $\rho = a/a_0$ is the effective plasma radius (i.e. the minor radius coordinate normalized to the minor radius of the separatrix at the respective poloidal angle), based on experimental measurements. The profiles are obtained from reciprocating probe [176] (RCP) in the SOL and from high-resolution Thomson scattering (HRTS) in the core.

No RCP measurements were available for the pulses of the density scan experiment. Instead, data were taken from four other pulses (JPN #81261, 80835-80836 and 81015). These pulses have a similar magnetic configuration and plasma conditions to those of the density scan experiment. Each of the four pulses has a different fueling resulting in $\langle n_\mathrm{e}^c \rangle = 1.2$–$3.3 \times 10^{19}\,\mathrm{m}^{-3}$, which mostly covers the electron density range of $\langle n_\mathrm{e}^c \rangle = 1.0$–$2.6 \times 10^{19}\,\mathrm{m}^{-3}$ of the density scan experiment, see Table 4.2. As an example, Figure 4.7 shows the radial profiles from HRTS

JPN	time [s]	$\langle n_\mathrm{e}^c \rangle$ $[10^{19}\mathrm{m}^{-3}]$
81261	48	1.2
80836	56	1.5
80835	56	2.1
81015	56	3.3

Table 4.2: List of pulse snapshots to obtain radial profile measurements of plasma parameters in order to create plasma backgrounds for ERO.

and RCP measurements. These profiles are obtained for JPN #80835 (medium fueling, $\langle n_\mathrm{e}^c \rangle = 2.1 \times 10^{19}\,m^{-3}$). Additionally, the T_e value obtained from the Be II line ratio (as described above in section 4.1) is shown, with ADAS PEC data assuming $n_\mathrm{e}^\mathrm{loc} = 10^{19}\,\mathrm{m}^{-3}$. The maximum of Be II emission is assumed to occur at $\rho = 1.02$ [24], which corresponds to a feasible penetration of Be neutrals into the

plasma before ionization.

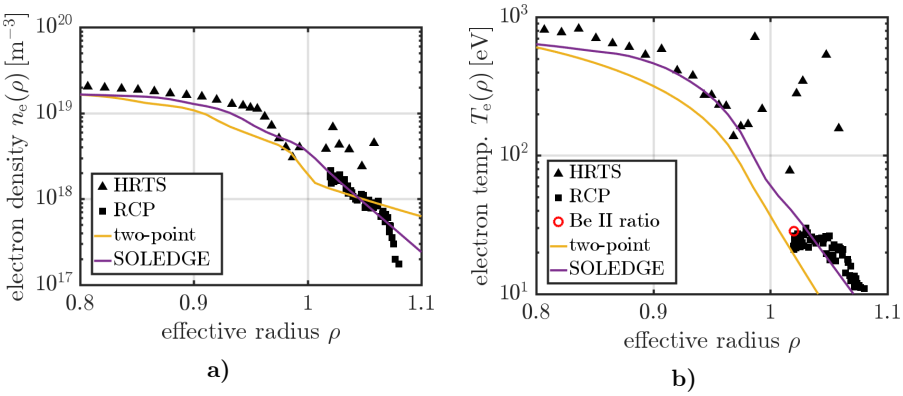


Figure 4.7: Experimental radial profile measurements from HRTS, RCP and Be II line ratio of (a) n_e and (b) T_e, plotted over the effective radius ρ for JPN #80835. The inner midplane profiles (location illustrated in Figure 4.8) obtained with the two-point model and with SOLEDGE2D-EIRENE are also shown. HRTS ist highly scattered for $\rho \geq 1$, which is due to uncertainties in the measurements at lower densities and also in the EFIT equilibrium reconstruction.

In order to obtain 2D plasma backgrounds in the (R,z)-plane from the 1D profiles, two different methods are used:

- *Two-point model*: This model (see [182, p. 224]) provides equations that relate the plasma parameters at a target to those at an upstream location (i.e. a location between two targets along the magnetic field line that connects them). As described in [132], one can solve the equations using the measured upstream profiles to obtain the plasma parameters at each target location. Using interpolation along magnetic field lines, one can obtain the 2D plasma background in the entire (R,z)-plane.
  This technique has already been used in the earlier local ERO1.0 modeling in [24].

- *SOLEDGE2D-EIRENE*: More recently, plasma backgrounds calculated with the SOLEDGE2D-EIRENE code were obtained.[35] This code couples the kinetic Monte-Carlo neutrals transport code EIRENE [162] with SOLEDGE2D, a solver of the Braginskii plasma fluid equations [31]. Unlike in comparable code packages such as B2-EIRENE [162], EDGE2D-EIRENE [178] or (until very recently) SOLPS-ITER [210], SOLEDGE2D-EIRENE allows to extend the computational grid right up to the first wall. However, extrapolation of

[35] Courtesy Dr. Hugo Bufferand from CEA, France.

the plasma parameters is still required for ERO in the shadowed zone behind the limiter ridge. Here, exponential extrapolation is used, where the decay length is determined from an exponential fit, in the direction normal to the flux surface, at the SOLEDGE2D-EIRENE grid boundary.

Most of the ERO2.0 modeling presented in this chapter has been performed using the two-point model plasma backgrounds unless it is explicitly stated otherwise, since SOLEDGE2D-EIRENE plasma backgrounds were not available until very recently.

As an example (again for JPN #80835), Figure 4.8 shows the electron temperature in the (R,z)-plane calculated with the two-point model and with SOLEDGE2D-EIRENE. The separatrix for the density scan experiment (JPN #80319) is also shown.

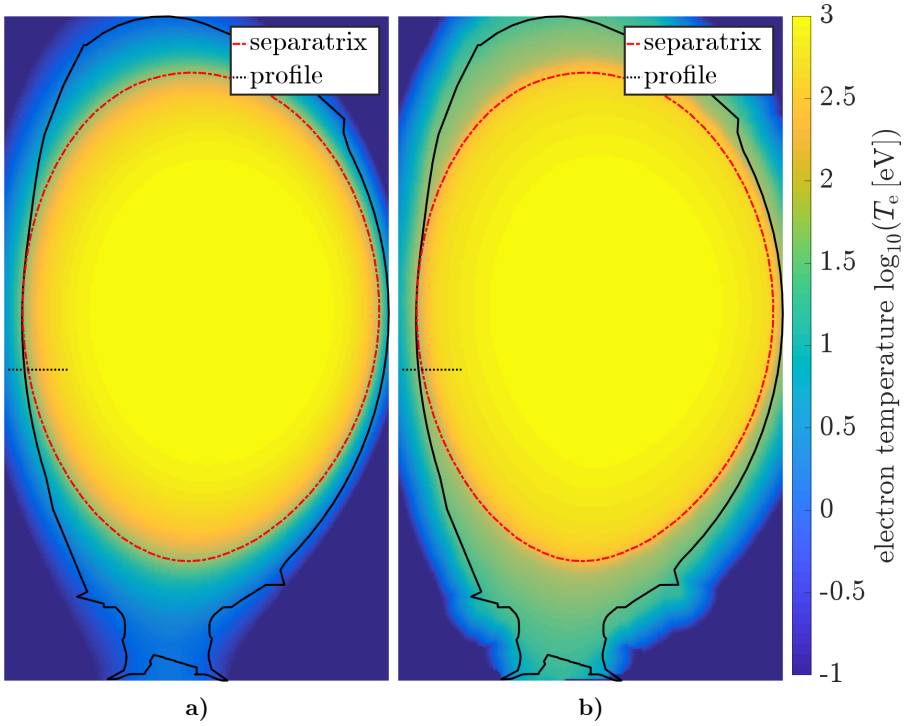


Figure 4.8: Illustrative examples of the 2D plasma backgrounds, which are used as an input for ERO, in the (R,z)-plane. The electron temperature is shown for JPN #80835. (a) Two-point model, (b) SOLEDGE2D-EIRENE. The dotted horizontal line indicates the inner midplane profile location for Figure 4.7.

Figure 4.8 clearly shows that SOLEDGE2D-EIRENE provides larger values of the electron temperature than the two-point model. This is also true for the

electron density at $\rho \lesssim 1.05$, as illustrated above in Figure 4.7, where the inner midplane profiles obtained with the two-point model and SOLEDGE2D-EIRENE are compared with the experimental profiles. Importantly, the two-point model underestimates the electron temperature on the OS at $\rho \approx 1.02$ (compared to experimental values from RCP and Be II line ratio), while SOLEDGE2D-EIRENE overestimates it.[36] Therefore, the two models at hand may be used as low and high estimates for both electron density and temperature in the plasma background.

Two-point model and SOLEDGE2D-EIRENE plasma backgrounds were provided for each of the four limiter pulses with RCP measurements at different fueling. As described in [24], this allows to parametrize the density scan experiment in $\langle n_e^c \rangle$, and create plasma backgrounds for each desired $\langle n_e^c \rangle$ by interpolation.[37] For the current ERO2.0 simulations of the density scan experiment, five interpolated plasma backgrounds were created (for both the two-point model and SOLEDGE2D-EIRENE) which are uniformly spaced within $\langle n_e^c \rangle = 1.5\text{–}2.5 \times 10^{19}\,\mathrm{m}^{-3}$.

Figure 4.9 shows the plasma parameters obtained from these plasma backgrounds on the spectroscopy OS. One can see again that the two-point model and SOLEDGE2D-EIRENE provide low and high estimates for the electron density and temperature. Interestingly, the local electron density from SOLEDGE2D-EIRENE becomes inversely proportional to the line-averaged density for $\langle n_e^c \rangle < 2 \times 10^{19}\,\mathrm{m}^{-3}$, which is not yet understood. Furthermore, the local electron temperature on the OS that is provided by SOLEDGE2D-EIRENE exceeds the experimental estimations from Be II line ratio by 2-3 times.

The two-point model does not provide the background plasma flow velocity as part of a self-consistent simulation like SOLEDGE2D-EIRENE. Therefore, the flow velocity was calculated using the 1D model described in [112, p. 12]. The parallel flow velocity $v_\parallel$ is given by

$$v_\parallel(s) = v_{\parallel,0} \left(\frac{L}{s} - \sqrt{\left(\frac{L}{s}\right)^2 - 1} \right) . \tag{4.2}$$

Here L is the connection length, and s is the distance along an open field line, with

[36] According to private conversation with Dr. Hugo Bufferand, who performed the SOLEDGE2D-EIRENE modeling, two explanations may be considered for the overestimated local electron temperature. First, the cooling effect of radiation by Be was not yet included in the simulations. Second, the recycling of deuterium near the plasma contact point is being underestimated. This has purely geometric reasons: by describing the wall in 2D, the magnetic field incidence angle near the contact point (and therefrom the incident and recycling fluxes) approaches zero in the simulation.

[37] $\langle n_e^c \rangle$ is the interferometer signal (see Table 4.1) giving the average density along the line crossing the plasma close to the plasma center. It is measured automatically with a high time resolution in all JET pulses, and is thus very suitable for the cross-calibration.

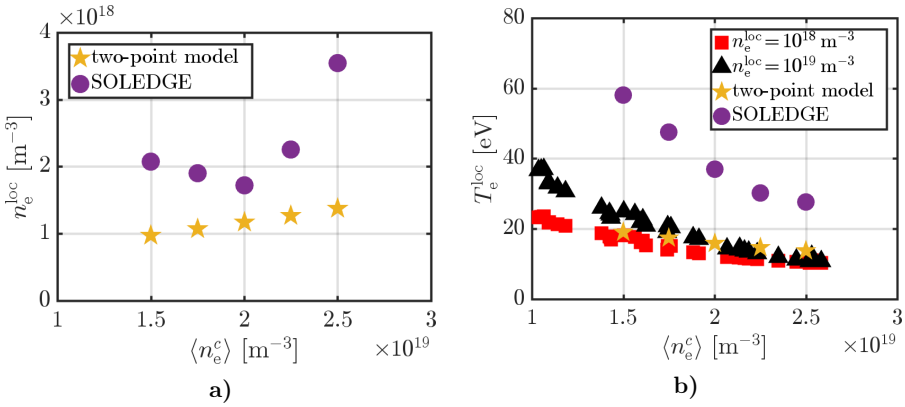


Figure 4.9: Local (a) electron density n_e^{loc} and (b) electron temperature T_e^{loc} in the plasma backgrounds calculated with the two-point model and SOLEDGE2D-EIRENE. The values are obtained by averaging at the 3D limiter surface on the spectroscopy OS, which corresponds to about $\rho = 1.02$ in Figure 4.7. The electron temperature is compared with the estimations based on experimental Be II line intensity ratios, see Figure 4.5(c).

$s = 0$ half-way between the two targets connected by the field line (stagnation point) and $s = \pm L/2$ at the targets. The parameters L and s can be calculated for every point in the SOL using numerical tracing of magnetic field lines as described in section 3.8. Figure 4.10(a) shows the 2D flow distribution resulting from tracing the field lines in 2D, i.e. neglecting the 3D shape of the limiters. Figure 4.10(b) shows the 3D flow distribution in the zone behind the inner wall limiter ridge that results from 3D magnetic field line tracing. Note that the flow is always directed towards the nearest limiter. By friction (see section 2.3.3), the flow may drive eroded Be impurity ions back to the inner wall and increase erosion there. The impact on the modeling results is discussed below in section 4.3.2.

4.2.3 Be erosion of and deposition on the limiters

In this section, a general description of the erosion and deposition patterns on the IWGLs is provided. This is done exemplarily using the following set of simulation conditions:

- The two-point model plasma background with the lowest density of $\langle n_e^c \rangle = 1.5 \times 10^{19}\,\mathrm{m}^{-3}$ is used.
- The Be surface is assumed to have zero D content (i.e. the ERO-max sputtering yields are used, see section 2.2.1).

The influence of the plasma conditions and the D content in the Be surface will be discussed quantitatively in section 4.3 using the ETSY.

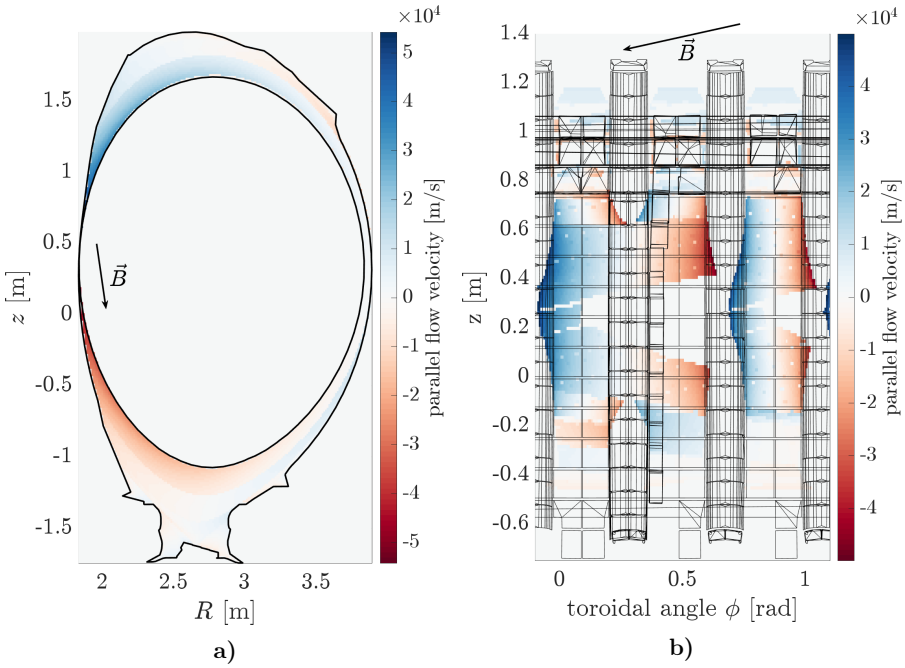


Figure 4.10: Examples of flow velocity patterns calculated with ERO2.0 from two-point model plasma parameters with equation (4.2). (a) 2D flow distribution in the (R, z)-plane, (b) 3D flow distribution (shown as a slice at $R \approx 1.8\,\mathrm{m}$ in the (ϕ, z)-plane) in the area behind the limiter ridges.

Figure 4.11 shows a selection of surface quantities and PWI results for the IWGL in JET octant 7X, which is a non-recessed limiter with bulk Be tiles. Figure 4.11(a)-(c) show the electron density n_e, electron temperature T_e and magnetic angle θ_B at the sheath entrance, respectively. These are obtained simply by interpolating the plasma background values at the surface cell centers. Note that the magnetic angles are shown only within the range $\theta_B = 80$–$90°$. This way, one can see that although the angles are typically very shallow ($\theta_B \approx 85°$ near the limiter ridge), the extreme situation of $\theta_B > 89°$, with ions reaching the surface more quickly than electrons as mentioned in section 2.3.2, does not occur except for the very small triangular areas between the limiter tiles. Figure 4.11(d) shows the connection lengths L (see also Figure 3.21). The pattern with the inversion at the plasma contact point is characteristic for limiter plasmas and is in good agreement with PFCFlux calculations [73] and power flux patterns obtained from infrared (IR) cameras [4]. The shadowing pattern is also visible in the light emission of eroded Be experimentally measured with 2D wide-view cameras (see Figure 4.19(b)) as it is discussed further below.

Figure 4.11(e) shows the incoming D ion flux calculated with equation (2.17), using a 'crude' model for the shadowing factor S:

$$S = \begin{cases} 0 & , L < L_{\mathrm{thr}} \,, \\ 1 & , L = L_{\mathrm{thr}} \,, \end{cases} \tag{4.3}$$

where L is the connection length of a surface cell and L_{thr} is a user-specified maximum field line tracing distance (in this case $L_{\mathrm{thr}} = 6\,\mathrm{m}$). Thus, surface cells are divided into 'plasma-wetted' ones with $\Gamma_{\mathrm{D}}^{\mathrm{in}} = n_{\mathrm{e}} c_{\mathrm{s}} \cos(\alpha)$ and 'shadowed' ones with $\Gamma_{\mathrm{D}}^{\mathrm{in}} = 0$. This is illustrated in Figure 4.12, where the calculated flux is compared before and after multiplication with S. Note that the same shadowing model has already been introduced in ERO1.0 for JET modeling [21], where the connection lengths have been calculated by dedicated PFCFlux code simulations. Possible extensions to the 'crude' shadowing model are discussed in section 4.2.5.

Figure 4.11(f) shows the flux $\Gamma_{\mathrm{Be}\leftarrow\mathrm{Be}}^{\mathrm{ero}}$ of Be particles eroded by D impact, calculated with equation (2.16). Figure 4.11(g) shows the Be ion flux $\Gamma_{\mathrm{Be}}^{\mathrm{in}}$ reaching the surface resulting from following the test particles until they impact on the wall. Figure 4.11(h) shows the flux $\Gamma_{\mathrm{Be}\leftarrow\mathrm{Be}}^{\mathrm{ero}}$ of Be particles eroded by self-sputtering, calculated with equation (2.15) from the impact of simulated Be test particles.

The erosion by self-sputtering in Figure 4.11(f) is somewhat lower but comparable to the erosion by D in Figure 4.11(h). A more quantitative comparison is given below in section 4.3.1 using the ETSY on the spectroscopy OS. Self-sputtering is also very different from sputtering by D in its spatial distribution, being more homogeneously distributed throughout the limiter surface area, including the shadowed area. This pattern cannot be reproduced by simply assuming a fraction of Be in the incident ion flux in Figure 4.11(e), as done in the local modeling that will be further discussed below in section 4.3.1. This highlights the importance of calculating self-sputtering in a self-consistent way using the global transport simulation of test particles.

4.2.4 Be migration

In this section, a general discussion of Be migration is provided. As in the previous section, exemplary results are used for this, which were obtained using the two-point model plasma background of lowest density ($\langle n_{\mathrm{e}}^{c} \rangle = 1.5 \times 10^{19}\,\mathrm{m}^{-3}$) and the assumption of a clean Be surface (ERO-max sputtering yields).

For the transport calculations, an ensemble of 10^6 (in total, i.e. from all surface cells together) eroded Be test particles was launched. This large ensemble is required

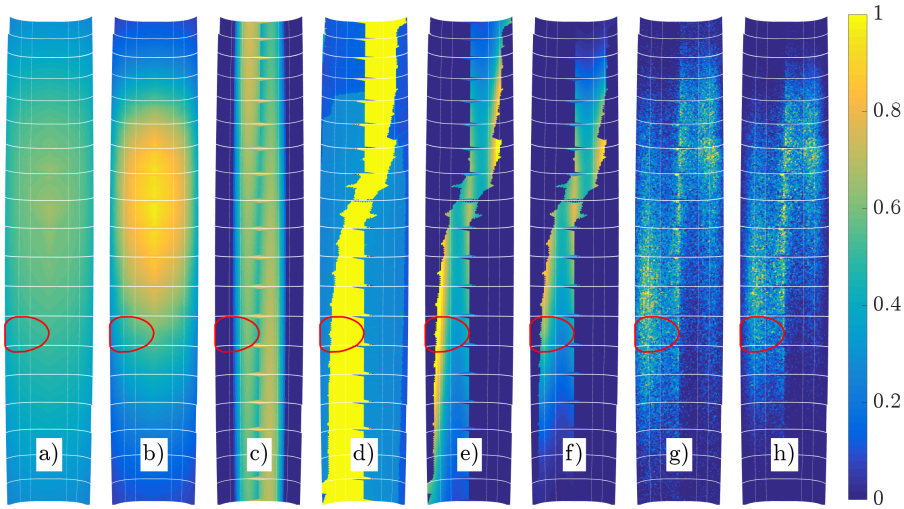


Figure 4.11: Color-maps (normalized to respective maximum value) of selected surface parameters and PWI results for the IWGL in JET octant 7X. The respective quantities are (a) electron density $n_e \,/\, \left(2 \times 10^{12}\,\mathrm{cm}^{-3}\right)$, (b) electron temperature $T_e \,/\, (40\,\mathrm{eV})$, (c) magnetic angle (relative to the surface normal) $(\theta_B - 80°) \,/\, 10°$, (d) connection length $L \,/\, (6\,\mathrm{m})$, (e) incident D flux $\Gamma_D^{in} \,/\, \left(10^{18}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}\right)$, (f) sputtered Be flux $\Gamma_{Be\leftarrow D}^{ero} \,/\, \left(3 \times 10^{17}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}\right)$, (g) incident Be flux $\Gamma_{Be}^{in} \,/\, \left(3 \times 10^{17}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}\right)$, and (h) self-sputtered Be flux $\Gamma_{Be\leftarrow Be}^{ero} \,/\, \left(3 \times 10^{17}\,\mathrm{cm}^{-2}\mathrm{s}^{-1}\right)$. The red oval in (g) shows the spectroscopic system OS used previously for determining the ETSY experimentally [35] and by ERO1.0 modeling [24].
The figure illustrates the workflow of the ERO2.0 modeling with chronological ordering from (a) to (g). (a)-(c) result simply from interpolating the input plasma background at the surface cell locations. (d) results from tracing the magnetic field lines. (e)-(f) result from applying equations (2.17) and (2.16), respectively. (g)-(h) result from following the deposition of and self-sputtering (2.15) by Be test particles eroded by D impact in (f).

to obtain a reasonable statistics for the velocity distributions of sputtered and impacting test particles, in particular on the spectroscopy OS (see Figure 4.11(h)). For each surface cell, between zero and 50 test particles are created. The total number of test particles launched per surface cell is weighted with the erosion rate of this surface cell, so that each test particle corresponds to roughly the same amount of real Be particles.

This simulation (and also the further ones with different plasma backgrounds and sputtering assumptions in section 4.3) requires roughly 3 h on 240 CPU cores. Note that without parallelization, the simulation would require roughly 30 days. Without the octree optimization, it would be a factor 10^4 longer, i.e. roughly 800 years (see appendix A).

In addition to the Lorentz force and friction with the background plasma ions (described by a Fokker-Planck collision term [134]), the test particle movement is

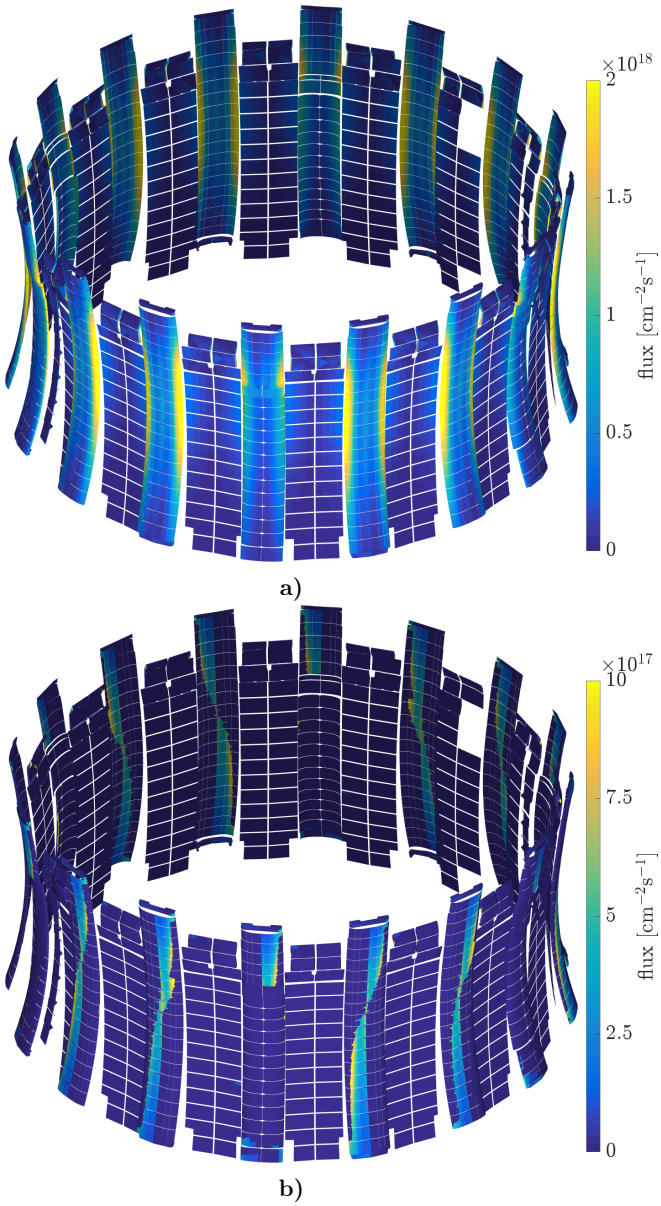


Figure 4.12: Calculated impact deuteron ion flux Γ_D^{in} on the inner wall: (a) without and (b) with consideration of shadowing. The underlying connection length data are shown in Figure 3.21.

affected by anomalous cross-field transport. It is treated in ERO as a diffusion process with a constant coefficient $D_\perp$, which is an input parameter. In the simulations shown here, $D_\perp = 1\,\mathrm{m}^2/\mathrm{s}$ was assumed for Be ions in the entire

plasma edge, which is in the order of the Bohm diffusion coefficient $D_{\perp}^{\mathrm{Bohm}} = (k_{\mathrm{B}}T_{\mathrm{e}})/(16eB) \sim 0.8\,\mathrm{m}^2/\mathrm{s}$ [182, p. 156] for $B \sim 4\,\mathrm{T}$ and $T_{\mathrm{e}} \sim 50\,\mathrm{eV}$ at the LCFS (see Figure 4.7(b)).

Figure 4.13(a)-(e) shows the simulated Be density for charge states $Z = 0$–4 averaged in toroidal direction and time. Note that the test particles do not penetrate deeper into the plasma than allowed by the inner simulation volume boundary, which is set at a closed flux surface at the radius $\rho_c = 0.9$. This selection of the boundary is arbitrary. Setting it at a smaller ρ or even omitting it entirely is possible, as shown in Figure 4.13(f). The possibility to use ERO2.0 for simulating the plasma core impurity transport, with full 3D resolution of the particle orbit, is an entirely new and appealing prospect. However, one should keep in mind that both ERO1.0 and ERO2.0 have been designed with a focus on plasma edge processes (as opposed to dedicated 1D plasma core radial transport codes such as COREDIV [186, 187]). A realistic description of the plasma core would for instance require a model for the radial profile of the diffusion coefficient. Furthermore, ERO2.0 simulations without the boundary are roughly ten times slower due to test particles performing a large number of poloidal rotations in the plasma core. Since for the current application the focus is mainly on the plasma edge physics, the inner boundary was introduced to ensure an optimum usage of computational resources.

The corresponding boundary condition is implemented in such way that a test particle crossing ρ_c is instantly removed and created again at a random poloidal location of the separatrix (since impurities inside the plasma core are homogeneously distributed in poloidal direction, as shown in Figure 4.13(f)). The particle is created with a thermal velocity corresponding to the temperature at this location. This mimics the process of cross-field transport, which should eventually remove an impurity particle from the plasma core after a certain dwell-time therein. This way, the inner boundary preserves the particle balance and does not change the relevant results in the plasma edge. In particular, the deposition and self-sputtering of Be and its impact distributions are unchanged, which was confirmed by comparing the corresponding simulation results with and without a boundary at $\rho_c = 0.9$. The only artifact observed after introducing the boundary is a thin 'red stripe' of reduced Be density just before it, which is visible for Be^{3+} and Be^{4+} in Figure 4.13(c)-(e). Without the boundary, the Be^{4+} density continually increases in radial direction towards the plasma center, as illustrated in Figure 4.13(f).

For Be^0 and Be^+, Figure 4.13 shows that the penetration depth (determined by the ionization probability) is in the order of several cm. Hence, their density is well localized near the erosion sites at the HFS. This shows that Be I and Be II line

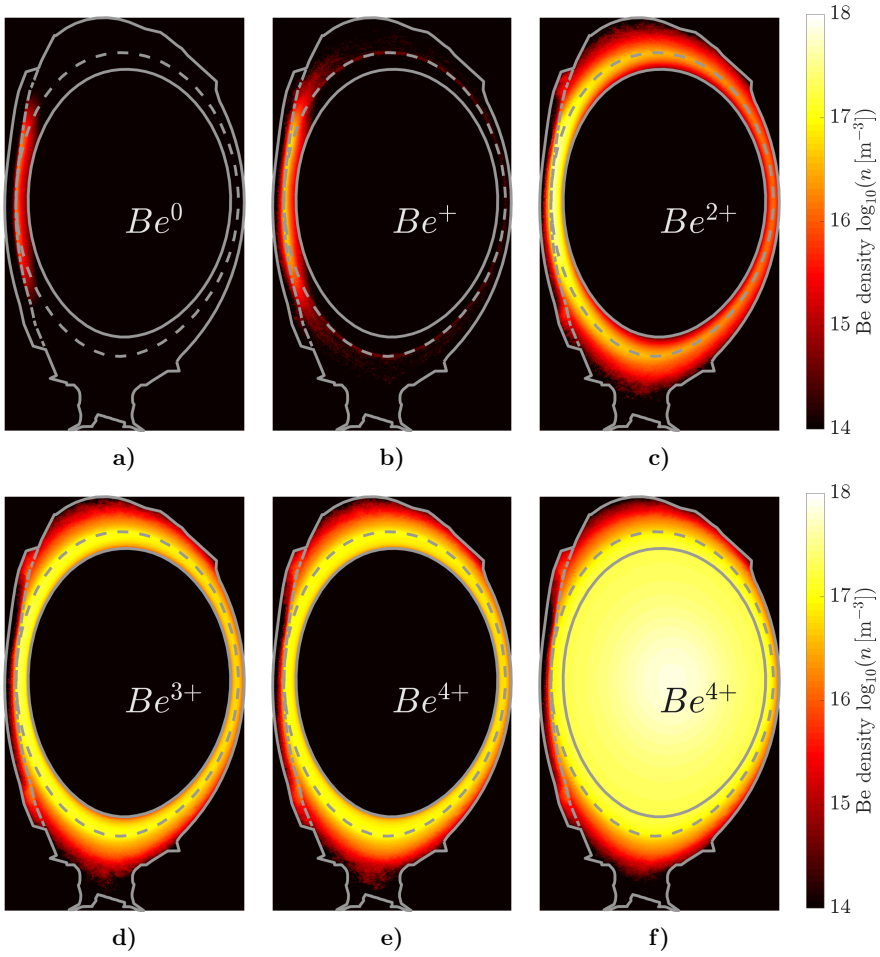


Figure 4.13: Color-maps showing the calculated Be density $\log_{10}(n)$, for different charge states Z, in the (R, z)-plane, averaged over the toroidal angle and time. The solid lines show the simulation volume boundaries, the dashed-dotted line shows the limiter ridge, the dashed line shows the separatrix. (f) shows a simulation in which no inner boundary was used.

emission measurements on the OS are determined mostly by the Be eroded within the same OS or up to a few cm away. This supports previous determination of the ETSY [24, 35], which assumed Be line emission to be originated from particles eroded directly from the OS surface area.

As Figure 4.13 shows, Be^{2+} is also predominantly located close to the erosion sites at the HFS, although it is already poloidally distributed and can be found e.g. at the LFS as well. Finally, Figure 4.13 shows that Be^{3+} and Be^{4+} penetrate deep

into the plasma and are very homogeneously distributed poloidally. This more highly charged Be with $Z \geq 2$ reaches distant wall areas. Of all Be test particles, 87 % end up on the inner wall, 12 % on the outer wall, and about 1 % on the upper dump plates or in the divertor.[38] Since a part of the Be goes into the spectroscopy OS, it affects the erosion therein by self-sputtering (although Be with $Z \geq 2$ is not observed by the visible spectroscopy and will not 'directly' affect the line emission measurements), which will be further quantified in section 4.3.

Figure 4.14 shows the charge state distribution of Be particles reaching the inner wall (accumulated over the entire surface area). A major particle fraction of 45 % reaches the surface as Be^{2+}, followed by 25 % Be^{3+}, 22 % Be^{4+}, 9 % Be^{+} and <1 % Be^{0}.

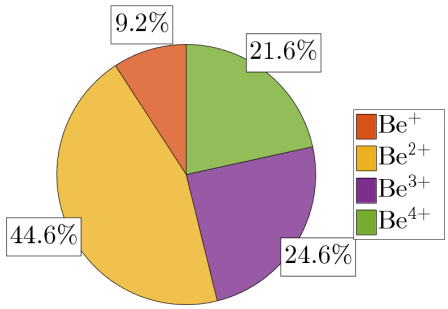


Figure 4.14: Distribution of charge state Z for Be reaching the inner wall, accumulated over the entire surface area.

Figure 4.15 shows the distributions for impact energy and angle, also accumulated over the entire area of the inner wall. Since incoming charged particles obtain a large part $E_{\mathrm{in}} = V_{\mathrm{sheath}} \cdot Z \cdot e$ of their energy in the sheath, the energy distribution (Figure 4.15(a)) is determined by the distribution of charge state Z, and by the local sheath potential V_{sheath}, which is proportional to the electron temperature. ERO2.0 assumes $V_{\mathrm{sheath}} = 3k_{\mathrm{b}}T_{\mathrm{e}}/e$ for the sheath potential. With the maximum of about $T_{\mathrm{e}} = 35\,\mathrm{eV}$ from Figure 4.11(b), the maximum potential is $V_{\mathrm{sheath}} = 105\,\mathrm{V}$. Therefore, each of the energy distributions for an individual charge state Z has a distinctive kink at $E_{\mathrm{in}} = Z \cdot 105\,\mathrm{eV}$. Each kink is followed by a high-energy tail due to the initial thermal energy the particles have before entering the sheath. This tail becomes broader with increasing Z, since higher charged particles have typically travelled longer distances before entering the sheath and are already considerably thermalized with the background plasma. The total energy distribution has a mean

[38] In the current simulation, a test particle is followed only until its first impact on the wall. The possibility and importance of simulating the reflection of test particles from the wall is discussed in section 4.3.2.

at $E_{\text{in}} = 220\,\text{eV}$.

The angular distributions (Figure 4.15(b)) for each charge state show a characteristic shape with a monotonic increase from $\theta_{\text{in}} = 0°$ (normal incidence) to about $\theta_{\text{in}} \approx 50°$, followed by a peak which is more pronounced for low charge states Z. With increasing Z, the peak position is shifted to higher θ_{in} (shallow incidence). The total distribution has a mean at $\theta_{\text{in}} = 56°$. For shallow magnetic field angles relevant for the JET-ILW limiters or ITER blanket modules, similar values are found in literature with ERO1.0 calculations [24] for D ions, and analytic [171] and particle-in-cell calculations [101] for D and carbon (C) ions.

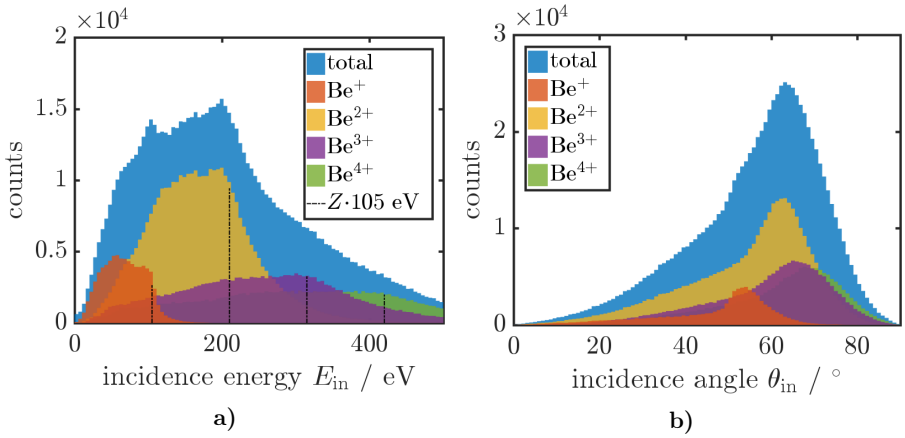


Figure 4.15: Distributions of (a) energy and (b) angle (relative to the surface normal) for Be particles reaching the inner wall, accumulated over the entire surface area. The total counts are shown, and also the contributions from individual charge states. For the energy distributions, vertical lines mark the locations of kinks at $E_{\text{in}} = Z \cdot 105\,\text{eV}$, which corresponds to the maximum sheath potential drop at the limiter surface.

4.2.5 Benchmarking with infrared cameras

Figure 4.16(a) shows an example of an experimental IR camera image (taken from [4]) for JPN #80836. Figure 4.16(b) shows the spatial distribution of the power flux q_{lim} reaching the IWGL in octant 8Z, mapped to the (ϕ, z)-plane (also from [4]). The power flux was calculated from the wall temperature obtained from IR camera images. For this the THEODOR code [88] was used, which numerically solves the 1D heat conduction equation. As JPN #80836 is one of the pulses that are used for constructing the plasma backgrounds (see Table 4.2), the IR measurements provide a valuable tool for validating these plasma backgrounds.

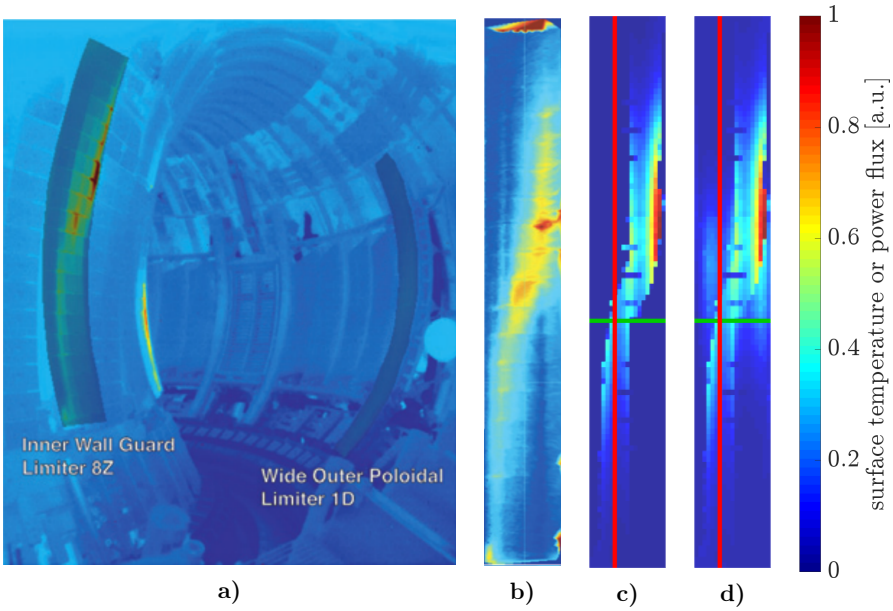


Figure 4.16: (a) Temperature map experimentally measured with a JET IR camera for JPN #80836 (image source: [4]). (b) Power flux pattern on the IWGL in JET octant 8Z, for the same pulse and mapped to the (ϕ, z)-plane, calculated from the IR-measured temperature using the THEODOR code [88] (image source: [4]). (c)-(d) Power flux pattern calculated by ERO2.0, using the two-point model plasma background for JPN #80836 in combination with the 'crude' and 'fine' shadowing models, respectively.

ERO2.0 calculates q_{lim} using the simple model [182, eq. (2.94)]

$$q_{\text{lim}} = \Gamma_{\text{e}}\, \delta\, k_{\text{B}}\, T_{\text{e}}\,. \tag{4.4}$$

Here, $\Gamma_{\text{e}} \approx \Gamma_{\text{D}}^{\text{in}}$ (see equation (2.17)) is the electron flux entering the sheath and $\delta \approx 7$–8 [182, eq. (2.95)] is the sheath heat transmission coefficient. Note that the flux calculated by ERO2.0 using equation (2.17) contains a shadowing factor. Therefore, comparison of the power flux (4.4) with the experiment can be used to validate both the plasma backgrounds and the shadowing model. Note that equation (4.4) does not account for the effect of thin deposits on the limiter. These are visible in the power flux calculated from the IR-measured temperature in Figure 4.16(b) as dark red spots. The spots are explained by the locally increased temperature of the deposits, which have a poor thermal contact with the bulk material [4].

Figure 4.16(c) shows the power flux calculated with equation (4.4) using the two-point model plasma background, the 'crude' shadowing model (4.3) and the

sheath heat transmission coefficient $\delta = 8$. The experimental power flux pattern is qualitatively reproduced. A quantitative comparison is shown in Figure 4.17, where toroidal and poloidal profiles are compared with experimental ones. The power flux calculated with ERO2.0 is in the same order of magnitude as the experimental values but underestimates those, especially at locations far away (toroidally or poloidally) from the plasma contact point at $\phi \approx 1.07\,\text{rad}$, $z \approx 0.3\,\text{m}$. A better

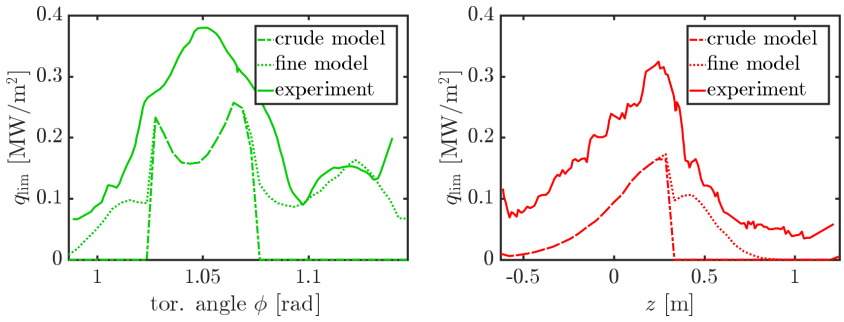


Figure 4.17: (a) Toroidal and (b) poloidal profiles of the power flux on the limiter. The locations of the profiles are shown in Figure 4.16(c)-(d). The calculated profiles are obtained with ERO2.0, using the two-point model plasma background for JPN #80836 in combination with the 'crude' and 'fine' shadowing models, respectively. The experimental profiles are derived from [4].

agreement with the experiment can be achieved in some regions of the limiter using a new, more comprehensive shadowing model

$$S(r) = \frac{n_e(0)\exp(-r/\lambda)}{n_e(r)}\,. \tag{4.5}$$

Here, r is the radial distance of the surface cell from the separatrix, and

$$\lambda = \frac{D_\perp L}{2c_s} \tag{4.6}$$

is a decay length which depends on a diffusion coefficient $D_\perp$, the connection length L and the sound velocity c_s (simple SOL model, see [182, eq. (1.10)]). As c_s is known from equation (2.18) and L is calculated by ERO2.0, the only free parameter is $D_\perp$.

This model is a slightly different formulation of the ones recently investigated with ERO1.0 in [55, 115], which also rely on equation (4.5), however with additional assumptions. In [55], 'zones' of similar connection lengths were defined with constant λ. In [115], the free parameter $D_\perp$ was canceled out using λ_{tip} and L_{tip}

at a limiter tip location. In the present case of a HFS-limited plasma, both these approaches are not applicable, since L diverges near the plasma contact point, so that neither different zones of similar L, nor λ_{tip} and L_{tip} can be easily defined.

The results for the power flux pattern calculated using the 'fine' shadowing model are shown in Figure 4.16(d) and also in Figure 4.17. The distinction between 'shadowed' and 'plasma-wetted' surface cells is now less sharp, which leads to a closer match with the experimental data. A parameter study (not shown) revealed that the flux pattern resulting from the 'fine' shadowing model is very sensitive to the parameter $D_\perp$. The best results shown here were obtained with $D_\perp = 0.1\,\text{m}^2/\text{s}$. A higher $D_\perp$ ($\sim 1\,\text{m}^2/\text{s}$) in the shadowing model leads to $S \approx 1$ throughout the limiter and therefore to flux distributions similar to those obtained without any shadowing model at all. A lower $D_\perp$ ($\sim 0.01\,\text{m}^2/\text{s}$) leads to results almost identical to those obtained with the 'crude' shadowing model.

This means that the benchmarking of ERO2.0 with IR camera images provides a useful way of determining the cross-field diffusion coefficient $D_\perp$. It is yet unclear whether the resulting $D_\perp$ can be directly used as an input for the ERO2.0 Be transport modeling (see section 4.2.4), since (1) $D_\perp$ is not necessarily the same for the main plasma species (which is mainly responsible for the power flux on the wall) and the impurities, and (2) $D_\perp$ has in reality a spatial dependence (in particular in radial direction), which compromises its present treatment as a constant value. In fact, the ERO Be transport modeling outcome has a low sensitivity on $D_\perp$, as was shown earlier by a sensitivity scan for ERO1.0 in [115], with $D_\perp$ being varied over 6 orders of magnitude. This is further discussed in the following section 4.2.6.

Regardless of the choice of the shadowing model, Figure 4.17 shows that equation (4.4) gives lower values for the power flux than the experimental ones. This is an indication that at least one of n_e or T_e is being underestimated in the two-point model plasma background. This is particularly well visible for the region $z < 0.3\,\text{m}$ (where $S \approx 1$ for both shadowing models) in Figure 4.17(b). Similar conclusions are drawn in the following section 4.2.6 from comparison with spectroscopy from wide-angle cameras.

An analogous comparison was also performed using the SOLEDGE2D-EIRENE plasma background for JPN #80836 (Figure 4.18). In contrast to the two-point model, the power flux was in that case strongly overestimated (roughly by a factor of 4), which can be attributed to an overestimation of at least one of n_e or T_e in the plasma background. This is in line with the benchmarking to horizontal LOS-integrated spectroscopy shown in Figure 4.9(b), where it was observed that SOLEDGE2D-EIRENE provides roughly 2-3 times higher electron temperatures compared to the estimations from Be II line ratios.

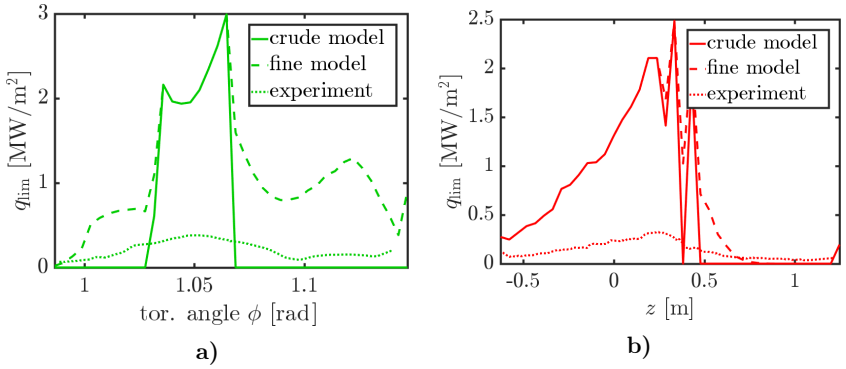


Figure 4.18: Similar profiles are shown as in Figure 4.17, but with the ERO2.0 profiles calculated using the SOLEDGE2D-EIRENE plasma background (for JPN #80836).

4.2.6 Benchmarking with wide-angle cameras

Fusion experiments are often equipped with camera systems (wide-angle cameras, endoscopes, filterscopes) for spectroscopically monitoring the plasma via 2D images. Examples of such diagnostics are found e.g. at JET [81, 93], W7-X [188], EAST [213], DIII-D [40, 48] or PISCES-A [90]. At JET, the wide-angle camera diagnostic KL1 [81] monitors the main chamber, and thus provides an opportunity of validating the Be density therein calculated with ERO2.0 (see section 4.2.4).

For this means, a novel synthetic diagnostic was implemented in ERO2.0, which mimics the respective camera system. This synthetic diagnostic relies on a simple description of the camera by a perspective projection model, illustrated in Figure 4.19. This model requires only a very small set of camera parameters, namely (1) the camera pupil position, (2) the camera view direction, (3) the angle of view and (4) the number of pixels of the charge-coupled device (CCD). This way, a virtual screen (image) subdivided into pixels can be defined. The distance between this virtual screen and the pupil position is irrelevant for the description; for simplicity the screen can be thought of as being placed between the object and the pupil. A pixel on the screen then corresponds to an object point if it is traversed by a ray between this object point and the pupil. Note that the optical aberration effects are neglected by this model.

For rendering an image, two complementary techniques are used:

- ray tracing
- volumetric splatting

The ray tracing technique (see Figure 4.19(a)), motivated by the well-known method from computer graphics science [5], is used for rendering the first wall

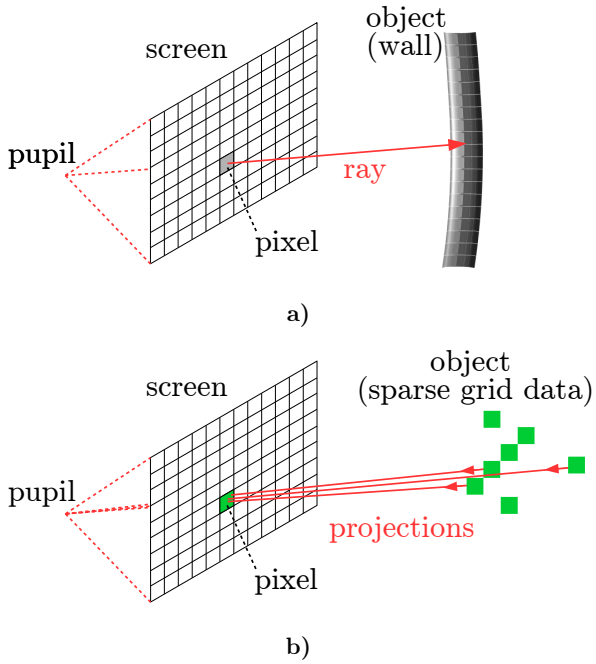


Figure 4.19: Illustration of the ERO2.0 synthetic camera models: (a) ray tracing technique (used for rendering the wall), (b) volumetric splatting (used for rendering gridded sparse data, i.e. impurity density or light emission).

of the fusion device. For each pixel, exactly one ray is cast which goes through the pupil and the pixel center. For each polygon of the 3D polygon mesh in ERO2.0, an intersection search is done for this ray using the algorithms described in section 3.4. The pixel is black if the ray does not hit any object. Otherwise, if an intersection is found, the pixel is colored according to the intensity of light reflection at the intersected polygon. The intensity is calculated using a very simple flat shading model, in which light coming from a source is assumed to be homogeneously reflected on the entire polygon.[39] The intensity is then added up from the ambient and diffuse reflection components. The ambient component is constant for all polygons, while the diffuse component is proportional to the cosine between the polygon normal and the connection vector between the polygon center and the light source. The light source location is arbitrary for the present purposes and is placed at the pupil position for simplicity.

[39]More advanced shading models such as the Gouraud [84] or Phong [149] models use interpolation of the reflection between polygon vertices, which results in more photorealistic images even at lower polygon resolution of the polygon mesh. For the purposes of the present ERO2.0 modeling, the computationally more simple flat shading is sufficient.

Figure 4.20 shows the grayscale image rendered with this model. From this image one can easily extract a transparent wireframe using the isoline technique, i.e. by extracting contour lines along which the grayscale value is constant. The resulting wireframe is shown in Figure 4.21, overlaid with an experimental image (with a Be II 467 nm filter) that is well suitable to indicate the main wall components. This way, it is possible to evaluate the quality of the perspective projection model and the underlying camera parameters. One can see that there is misalignment in some image regions (mostly on the image borders where optical aberrations are the strongest). Nevertheless, the relevant image regions close to the inner wall, where the most Be I and Be II emission occurs in the HFS-limited pulses, are well aligned. For obtaining a better alignment in the entire image, the implementation of correction algorithms that account for the optical aberration can be considered, which however goes beyond the scope of the present work.

For rendering the line emission coming from the plasma, the light intensity (which is stored as a sparse 3D matrix in ERO2.0, see section 3.7) must be integrated along the 'line-of-sight' for each pixel. The ray tracing technique is not feasible for this due to the finite solid angle of each pixel. The rays going through a pixel are diverging with increasing distance from the screen, so that increasing numbers of grid cells (in lateral direction) are projected onto the pixel. Integrating the light would therefore require to follow the ray in finite steps, combined with complex algorithms to detect the grid cells to be projected on the pixel for each step. Furthermore, the sparsity of the 3D data in ERO2.0 would result in a waste of computational effort due to stepping through many empty grid cells.

A more sensible technique for light integration is therefore the volumetric splatting technique [209]. This approach goes the opposite way to the ray tracing (see Figure 4.19(b)): instead of starting at a certain pixel and searching for objects which are hit by the ray going through its center, the algorithm focuses on the individual object points (in this case the non-empty grid cells of the sparse matrix) and finds the corresponding pixels on which they are projected. The projection of a grid cell onto the screen ('splatting') is approximated by a square of size $\mathrm{d}r$ which is proportional to the cell size and inversely proportional to the distance d of the cell to the pupil.

The light intensity which is added to the pixel is given by the number of photons per second emitted from this grid cell times $\Omega/(4\pi)$, where $\Omega = \iint_A \mathrm{d}\Omega = \iint_A \mathrm{d}\theta\mathrm{d}\phi \sin\theta$ is the solid angle of the camera entrance slit of area A. For instance, a square camera entrance slit of size $A = \Delta R^2$, with $\Delta R/d \ll 1$ so that $\Delta\theta \ll 1$ and $\Delta\phi \ll 1$, would result in a solid angle $\Omega \approx \Delta R^2/d^2$. For the very first qualitative application of the synthetic diagnostic, the slit size ΔR (which was not known

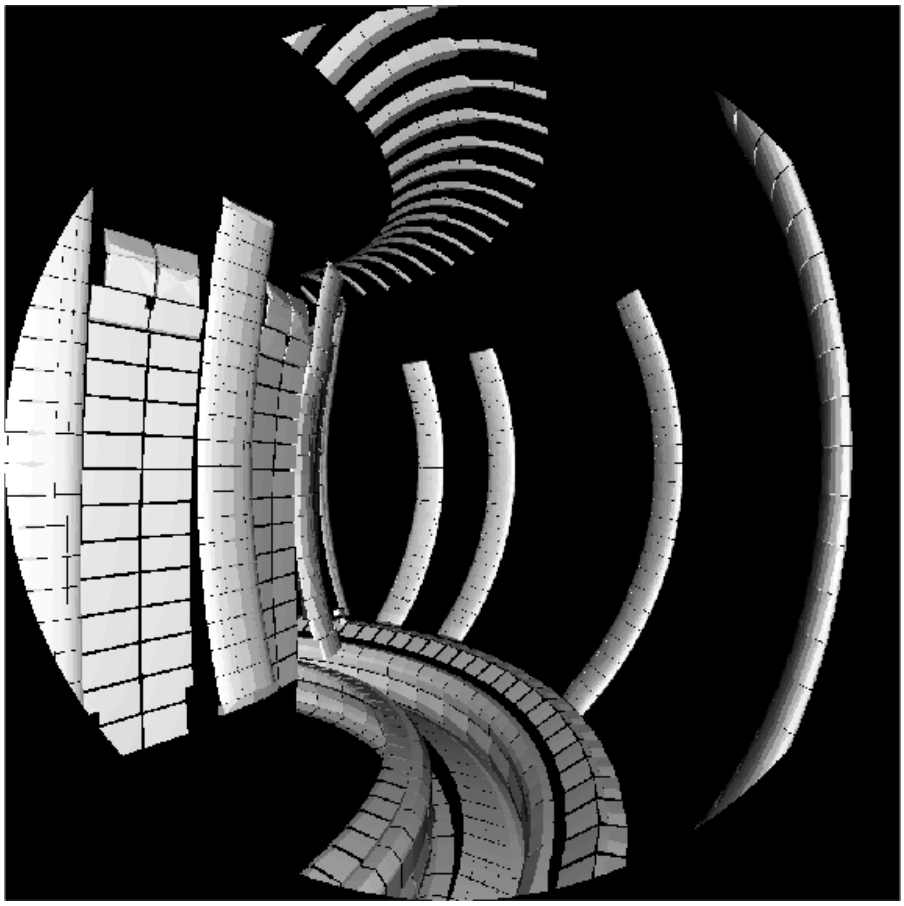


Figure 4.20: Synthetic image of the JET-ILW rendered by ERO2.0 with the ray-tracing technique, using camera parameters of the JET wide-angle camera KL1-E4WC.

at the point of writing this manuscript) was set to unity, therefore the calculated intensity values are given in arbitrary units.

If the grid cell projection covers more than one pixel, the intensity is, for simplicity, equally distributed between the pixels. Due to the small size ($\sim 1\,\mathrm{mm}^3$) of the grid cells, this situation typically occurs only at pixel edges or corners, and does not affect the main results.

Figure 4.22 shows a comparison of an experimental image with a synthetic image created by ERO2.0 using the volumetric splatting technique. Since no Be emission images were available from the density scan experiment, images are used that were measured parasitically in limiter pulses (reference pulse JPN #91140), with the plasma shifted upward compared to JPN #80321. To obtain a

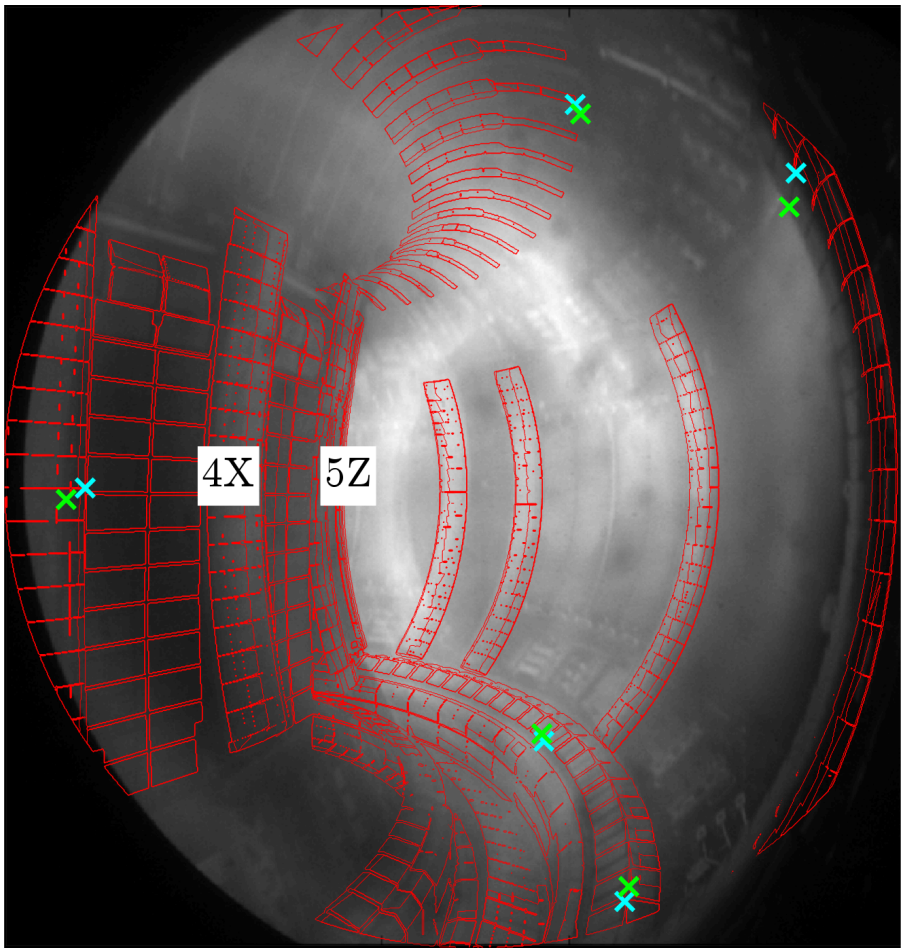


Figure 4.21: Comparison of experimental 2D images from a JET wide-angle camera (KL1-E4WC, Be II 467 nm filter) and the corresponding synthetic images calculated by ERO2.0. The image shows an overlay of an experimental image (grayscale) of a disruption, during which the wall is fairly visible, with a wireframe model of the wall (red) rendered by ERO2.0. Five reference point pairs (green: experiment, cyan: synthetic) are shown in order to illustrate regions where the two images are misaligned. The misalignment occurs mostly at the image borders where optical aberrations are the strongest. Also, there is a misalignment in the divertor region of the image closer to the image center. This suggests a certain inaccuracy in the line-of-sight and pupil position of the synthetic camera model. However, the misalignment in the relevant image parts, where the most Be line emission occurs (inner wall, octants 4X and 5Z), is small enough that it does not affect the general picture and the discussed comparison of synthetic and experimental emission.

plasma background in the (R, z)-plane as shown in Figure 4.8(c), a translation and 'squeezing' transformation was applied to it. Figure 4.22(a) shows an overlay of the wireframe with the experimental image of Be II 467 nm line intensity (colored) from

JPN #91140. Figure 4.22(b) shows a similar overlay for the intensity simulated by ERO2.0. One can see that ERO2.0 can qualitatively well reproduce the emission patterns from the experiment. Since Be^+ is localized at the erosion areas, the image reflects the erosion patterns from Figure 4.11, with the above-described shadowing pattern as the most striking feature. One can observe two distinct emission 'plumes' from the IWGLs in octants 4Z (left side of image) and 5Z (center of image). The emission plume of the IWGL in-between in octant 5X is limited to the five tiles at the top, since the other tiles are recessed and therefore not eroded.

Some subtle differences between experiment and modeling are yet visible, which are marked as regions I-III in the figure:

- In region I, the emission plumes at the IWGLs in the experimental image are more extended in z-direction up to the top IWGL tiles compared to the synthetic image. This might indicate that the T_e in the ERO2.0 plasma background has a too strong gradient in radial direction inside the SOL, so that the upper tiles are not eroded in the model. This is in line with what was observed for the power flux from IR cameras in section 4.2.5.

- In region II, the plume in octant 5Z has a different shape: it is thinner near the plasma contact point and becomes broader near the limiter ends in experiment, while in the modeling the plume has approximately equal thickness everywhere which is also seen in the Be^+ density in Figure 4.13(a). This can also be explained by uncertainties in the plasma background used for the modeling. The plume width is determined mostly by the ionization rate for Be^0 or Be^+, which is calculated as the product of the corresponding ionization rate coefficient (obtained from the ADAS code database [189]) and n_e. While the rate coefficient shows only a slight dependence on n_e and T_e in the parameter range of interest, an underestimation of the electron density n_e in the plasma background may well account for the too broad emission plume. A more thorough benchmarking would require comparison to Be I experimental images, which were not available for JPN #91140 due to technical issues with the camera filter.

- In region III, an emission plume is visible in the experimental image below the IWGL in octant 5Z, which is missing from the synthetic image. This can be attributed to erosion of the Be parallel protection bars [125], which are currently missing in the ERO2.0 ILW geometry.

Apart from the plasma background, the diffusion coefficient $D_\perp = 1\,m^2/s$ used for the transport simulations (see section 4.2.4) was also expected to be a source of

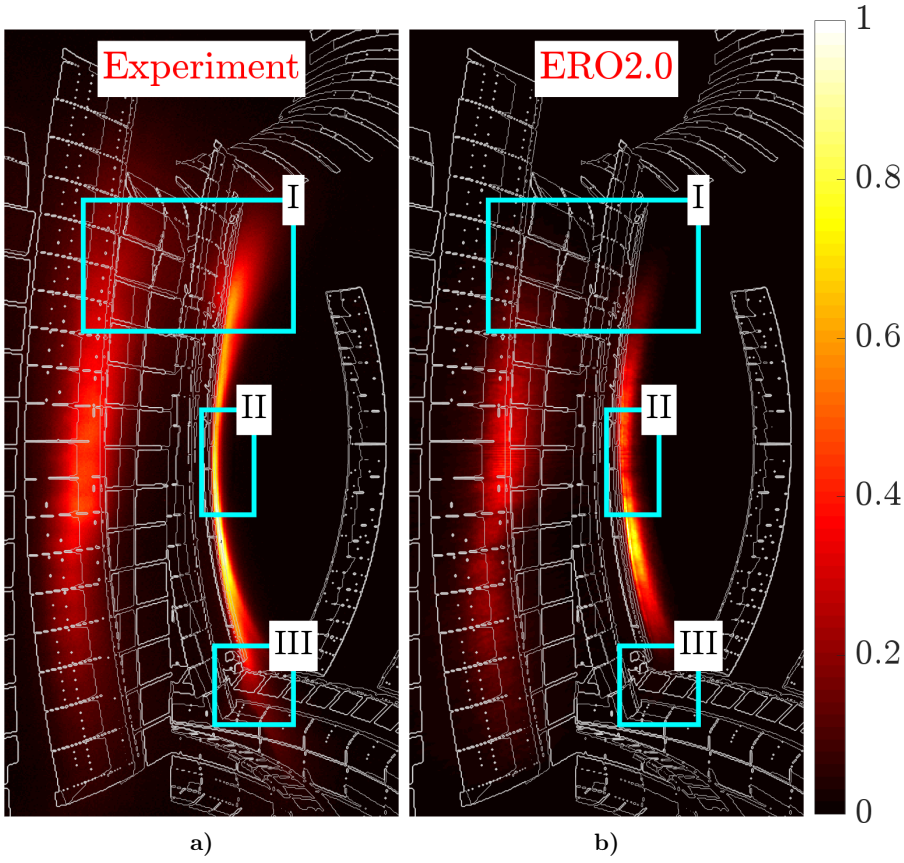


Figure 4.22: Comparison of experimental 2D images from a JET wide-angle camera (KL1-E4WC, Be II 467 nm filter) and the corresponding synthetic images calculated by ERO2.0. (a) Overlay of the wireframe model from Figure 4.21 (white) with an experimental image from JPN #91140 of the same camera showing the Be II emission intensity. (b) The same, with the Be II emission rendered by the ERO2.0 synthetic camera model from the simulated volumetric emission data. The color scale is normalized to the maximum intensity of each image.

uncertainty and affect e.g. the plume width in region II. Therefore the simulations were repeated with two different coefficients $D_\perp = 0.1\,\mathrm{m}^2/\mathrm{s}$ and $D_\perp = 10\,\mathrm{m}^2/\mathrm{s}$. Note that in this sensitivity scan, $D_\perp$ has no impact on the erosion, since the 'crude' shadowing model (4.3) is used, which is independent of $D_\perp$. The impact of $D_\perp$ on erosion using the 'fine' shadowing model (4.5) has been discussed separately in section 4.2.5.

The images resulting from the scan (not shown here) have only a very weak dependence on $D_\perp$. In toroidal direction, the emission plumes are slightly more

stretched for low values and more compressed for high values of $D_\perp$, since cross-field transport obstructs the transport along field lines. However, the poloidal and radial shape of the plumes (for which the main differences to the experimental 2D image are observed) is not affected by $D_\perp$.

In summary, the novel synthetic diagnostic implemented in ERO2.0 for rendering 2D images is a powerful tool for validating the global modeling results, since it contains information about erosion (including shadowing effects), transport, ionization and light emission in the entire poloidal cross-section of a fusion device. The first application for rendering Be II 467 nm line emission shows a reasonable agreement with the corresponding experimental image. This increases confidence in the present ERO2.0 modeling of global Be erosion and deposition in JET-ILW HFS-limited limiter pulses. It must be stressed that these modeling results depend to a high degree on the two-point model plasma background used as an input here. Some differences between the synthetic and experimental emission plumes are found, mainly at the top and bottom parts of the IWGLs. This observation is in line with the benchmarking results for IR camera images discussed in section 4.2.5 and suggests to critically revise the input plasma background.

Up to now, the 2D images are used for a qualitative benchmarking with the experiment, since the synthetic diagnostic does not provide the absolute light intensity. A quantitative comparison with the experiment, including a variation of the plasma backgrounds, is provided in the following section 4.3, using the Be ETSY on the observation spot of the KS3 diagnostic horizontal line-of-sight.

4.3 Effective Be sputtering yield

Two ERO2.0 modeling methods are compared in this section with respect to their results on the calculated Be ETSY:

- In the global modeling presented in section 4.3.2, the simulations which were described in section 4.2 are repeated, with a variation of (1) the plasma backgrounds and (2) the assumptions about D surface content in the Be limiter tiles with respect to the sputtering yields (ERO-min, ERO-max). The aim of the parameter scan is to reproduce the experimentally observed dependence of the ETSY on the plasma conditions, as shown in Figure 4.5. The main focus is to investigate the contribution of Be self-sputtering to the ETSY in the modeling, which is done in a self-consistent way by following the eroded Be test particles in the entire plasma edge and calculating the sputtering occurring due to their impact on Be tiles (including those covered

by the horizontal spectroscopy OS).

- The local modeling presented in section 4.3.1 is similar to the earlier ERO1.0 modeling in [21, 23, 24], using the small simulation volume shown in Figure 4.2(a), which is limited to the horizontal spectroscopy OS and its direct surroundings. The novelty compared to [21, 23, 24] is the additional consideration of SOLEDGE2D-EIRENE plasma backgrounds, which were not available in the earlier work. Since about 90 % of the eroded test particles leave the small simulation volume without striking the surface, and on the other hand the contribution of Be impurities flowing into the simulation volume from outside is not accounted for, self-sputtering cannot be treated self-consistently in the local modeling. Instead, the self-sputtering is treated by assuming a constant Be impurity concentration c_{Be} in the background plasma, so that the ETSY on the spectroscopy OS can be calculated directly (without the need for following the test particles) using equation (2.16). Since the time-consuming following of test particles is omitted[40] and the number of surface cells is much lower than for the global modeling, the simulations are extremely fast ($< 1\,\mathrm{s}$ computational time on a single CPU core).

4.3.1 Local modeling

The calculation of the Be impurity concentrations c_{Be} is discussed in appendix C. Three exemplary cases are considered: no Be impurities, Be^{3+} only and Be^{4+} only. Note that the simple formula presented in appendix C, which is used for estimating c_{Be} from the measured effective charge Z_{eff}, requires the assumption of a single impurity charge state. Note also that lower charged Be is not considered here, even though global ERO2.0 modeling predicts that most Be arrives on the OS as Be^{2+} (see Figure 4.14). This is because $Z_{\mathrm{eff}} = 2$ at $\langle n_{\mathrm{e}}^{c} \rangle = 1.5 \times 10^{19}\,\mathrm{m}^{-3}$ would require the unrealistic assumption of a plasma composed to 100 % of Be^{2+}. This is however not a contradiction to the global modeling results, since the Z_{eff} measurement is averaged along the entire spectroscopy LOS, which covers the plasma core where more highly charged impurities are present, and may therefore be higher than the actual local Z_{eff} on the OS.

For calculating the erosion due to the D and Be components of the background plasma, equation (2.16) is used, with the respective incident ion fluxes calculated

[40] Note that in the original work in [21, 23, 24], the test particle transport was in fact calculated. However, the aim was not to reproduce self-sputtering, but rather a direct comparison of simulated and measured light emission. Such a benchmarking is out of scope of the present work, but should eventually be envisaged within the framework of the global modeling for additional verification.

as

$$\Gamma_{\mathrm{D}}^{\mathrm{in}} = \Gamma_{\mathrm{e}}^{\mathrm{in}} \frac{n_{\mathrm{D}}}{n_{\mathrm{e}}} = \Gamma_{\mathrm{e}}^{\mathrm{in}} c_{\mathrm{D}} \,, \tag{4.7}$$

$$\Gamma_{\mathrm{Be}}^{\mathrm{in}} = \Gamma_{\mathrm{e}}^{\mathrm{in}} \frac{n_{\mathrm{Be}}}{n_{\mathrm{e}}} = \Gamma_{\mathrm{e}}^{\mathrm{in}} c_{\mathrm{Be}} \,. \tag{4.8}$$

Figure 4.23 shows the comparison of the experimental and calculated ETSYs. It is observed that self-sputtering is an important contributor to erosion, in particular at low densities ($\langle n_{\mathrm{e}}^{c} \rangle \sim 1.5 \times 10^{19}\,\mathrm{m}^{-3}$). The assumption of Be^{3+} impurities has a higher effect on $Y_{\mathrm{eff}}^{\mathrm{tot}}$ than Be^{4+} impurities, since the former leads to twice as large impurity concentrations (see Figure C.1), while the sputtering yields are similar for both Be^{3+} and Be^{4+} in the relevant temperature range as shown in Figure 4.24.

For the results obtained with the two-point model plasma backgrounds (Figure 4.23(a)-(b)) and with the Be^{3+} assumption, the curves for ERO-min and ERO-max enclose the experimental one and are therefore valid low and high estimates. In particular, ERO-min gives a close agreement with the experiment at large $\langle n_{\mathrm{e}}^{c} \rangle$, while ERO-max gives a close agreement at low $\langle n_{\mathrm{e}}^{c} \rangle$. This is interpreted in such way that at low $\langle n_{\mathrm{e}}^{c} \rangle$ the limiter surface temperature is higher, which leads to more outgassing of deuterium. In addition, the incident deuterium flux is lower. These two effects decrease the deuterium surface content, so that the ERO-max sputtering yields (which were calculated for a pure Be target) give a closer agreement at low $\langle n_{\mathrm{e}}^{c} \rangle$. However, it must be stressed that the contribution of CAPS to the erosion has been neglected in the entire modeling presented in this chapter. Since it can account for up to 1/3 of the Be erosion [35], the implementation of this erosion mechanism in ERO2.0 is required, which should lead to a somewhat closer agreement for ERO-min with the experiment.

The results obtained with the SOLEDGE2D-EIRENE plasma backgrounds (Figure 4.23(c)-(d)) show generally larger values of $Y_{\mathrm{eff}}^{\mathrm{tot}}$ due to the higher temperature compared to the two-point model plasma backgrounds. This leads to a slightly better overall agreement with experimental data for ERO-min. Interestingly, the slope of these curves is decreasing at lower density (higher temperature), while the curves in Figure 4.23(a)-(b) and the experimental one show the opposite curvature. Without consideration of self-sputtering (no Be impurities), $Y_{\mathrm{eff}}^{\mathrm{tot}}$ even stagnates or decreases slightly in the low density range. The reason for this is illustrated in Figure 4.24, which shows the sputtering yields from the Eckstein fit (see Figure 2.5) calculated for different electron temperatures and for the average magnetic field angle $\theta_B = 75°$ on the spectroscopy OS. In the temperature range $T_{\mathrm{e}}^{\mathrm{loc}} = 30$–$60\,\mathrm{eV}$ of the SOLEDGE2D-EIRENE plasma backgrounds (see Figure 4.9), the sputtering

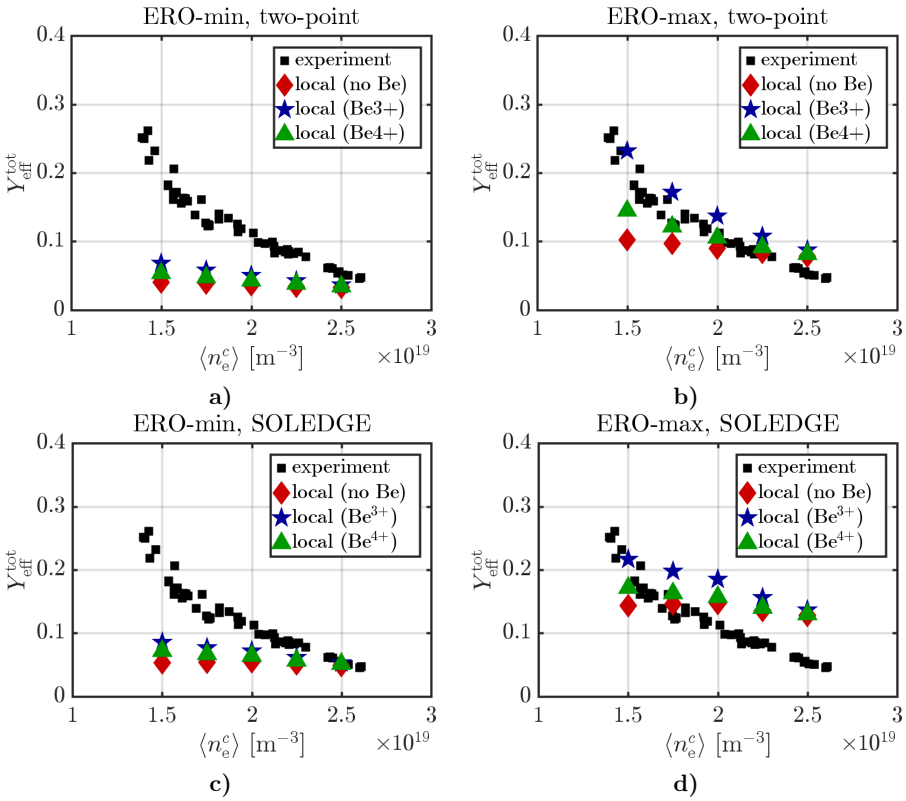


Figure 4.23: Comparison of the ETSY $Y_{\mathrm{eff}}^{\mathrm{tot}}$ from the density scan experiment with values calculated from local ERO2.0 modeling with various assumptions about Be plasma content (no Be, Be^{3+} only, Be^{4+} only). Two different plasma backgrounds are used (two-point model, SOLEDGE2D-EIRENE) as well as two different Be sputtering assumptions (ERO-min, ERO-max), resulting in four different plots: (a) ERO-min and (b) ERO-max with two-point model plasma backgrounds, (c) ERO-min and (d) ERO-max with SOLEDGE2D-EIRENE plasma backgrounds.

yields stagnate near their maximum and start decreasing slowly. Compared to the two-point model, SOLEDGE2D-EIRENE leads to a worse agreement with the experiment in the curvature of $Y_{\mathrm{eff}}^{\mathrm{tot}}$. In summary, the following conclusions can be drawn from the local modeling results for $Y_{\mathrm{eff}}^{\mathrm{tot}}$:

- Self-sputtering is an important contributor to Be erosion in the present HFS-limited pulses, in particular at low densities (high temperatures) of the plasma.
- The assumption of Be^{3+} impurities in the plasma leads to higher Be concentrations in the plasma and higher self-sputtering than the assumption

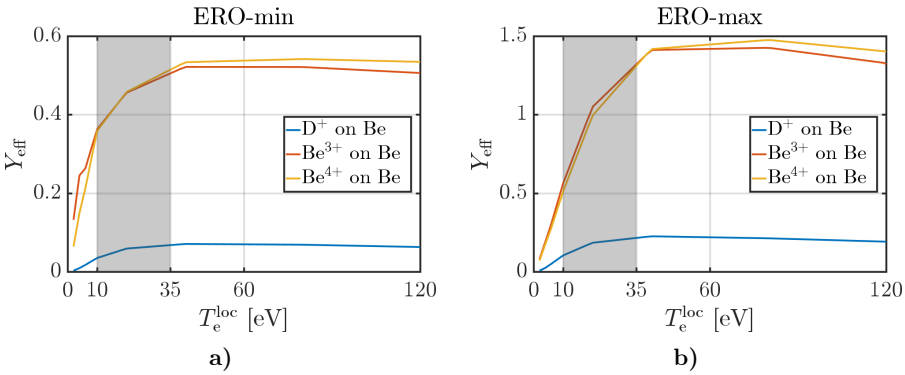


Figure 4.24: Dependence of the Be sputtering yields on the local temperature on the OS for D or Be impact: (a) ERO-min, (b) ERO-max. The yields are calculated using $Y_{\mathrm{eff}}(T_{\mathrm{e}}^{\mathrm{loc}}, \theta_B)$ (see Figure 2.5) using the average magnetic angle $\theta_B = 75°$ on the spectroscopy OS. The gray rectangle marks the approximate value range of $T_{\mathrm{e}}^{\mathrm{loc}} = 10$–$35\,\mathrm{eV}$ in the density scan experiment, see Figure 4.5(c).

of Be^{4+} impurities. This leads to a better match with experimental data in Figure 4.23(a)-(b).

- For the two-point model plasma backgrounds, the values obtained for ERO-min and ERO-max with the Be^{3+} assumption in Figure 4.23(a)-(b) act as valid low and high estimates respectively, which enclose the experimental values. ERO-min, which assumes a 50 % D content in the Be surface, gives a closer agreement at high $\langle n_{\mathrm{e}}^c \rangle$. This confirms the expectation of D presence in the surface and its effect of lowering sputtering yields due to dilution. On the other hand, ERO-max gives a closer agreement at low $\langle n_{\mathrm{e}}^c \rangle$. This can be interpreted in such way that the D is outgassed from the surface for the conditions with higher local plasma temperatures and lower incident fluxes, so that a purer Be target is obtained.

- Due to higher temperatures, the SOLEDGE2D-EIRENE plasma backgrounds lead to generally higher values of $Y_{\mathrm{eff}}^{\mathrm{tot}}$ in Figure 4.23(c)-(d). In both ERO-min and ERO-max cases, the curvature of $Y_{\mathrm{eff}}^{\mathrm{tot}}$ (as a function of $\langle n_{\mathrm{e}}^c \rangle$) is reproduced worse.

4.3.2 Global modeling

In the global simulations, erosion by self-sputtering is calculated in a self-consistent way using the incident Be flux from the transport simulations, rather than treating the Be concentration in the plasma as a free parameter. As a consequence, multiple

iteration steps need to be performed as described in section 2.1. In the 0th step, the test particles represent the Be atoms created by D impact. These are traced until each one has hit the wall (the possibility of considering particle reflection will be discussed shortly). The contribution to erosion by self-sputtering according to equation (2.15) is added up for all test particles. In the 1st step, the test particles represent also the additional atoms created by self-sputtering during the 0th step, etc. It can be shown from simple theoretical considerations (see appendix D) that the iteration converges under certain conditions, which corresponds to a steady state in which the incident and eroded particle fluxes remain constant. Figure 4.25 shows that the simulation results (in this case $Y_{\mathrm{eff}}^{\mathrm{tot}}$) indeed converge after a few steps. Figure 4.26 shows the end points of the curves in Figure 4.25 (i.e.

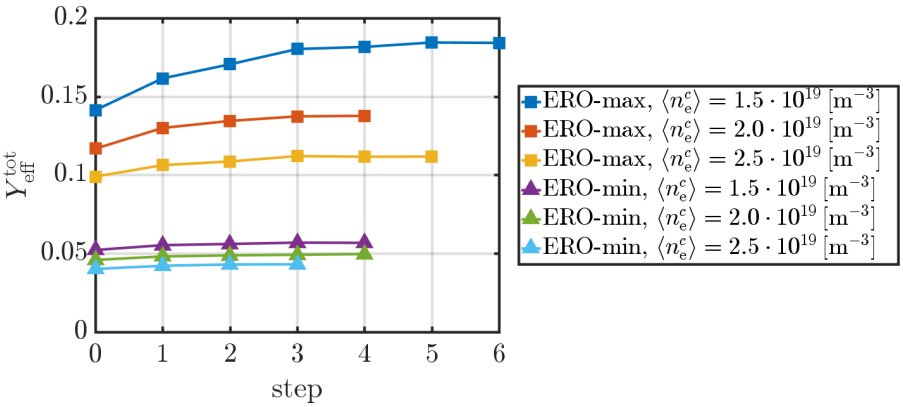


Figure 4.25: Dynamic evolution of the ETSY $Y_{\mathrm{eff}}^{\mathrm{tot}}$ during ERO2.0 iteration steps.

after convergence) plotted over $\langle n_{\mathrm{e}}^{c} \rangle$ together with the experimental values. The values obtained from local modeling without any consideration of self-sputtering (red diamonds in Figure 4.23(a)-(b)) and with the assumption of a certain Be^{3+} background concentration in the plasma (blue stars in Figure 4.23(a)-(b)) are also shown.

The results from global modeling are in a fair agreement with the local one with the Be^{3+} background concentration. The curves for ERO-min and ERO-max enclose the experimental one. In particular, the $Y_{\mathrm{eff}}^{\mathrm{tot}}$ calculated with ERO-min and ERO-max lie on the experimental curve at $\langle n_{\mathrm{e}}^{c} \rangle = 2.5 \times 10^{19}\,\mathrm{m}^{-3}$ and $\langle n_{\mathrm{e}}^{c} \rangle = 1.5 \times 10^{19}\,\mathrm{m}^{-3}$, respectively. However, the slope and curvature of the curves are lower than the ones from local modeling. In particular for ERO-max, the slope and curvature suggest that the experimental values for $\langle n_{\mathrm{e}}^{c} \rangle < 1.5 \times 10^{19}\,\mathrm{m}^{-3}$, where the temperature and Be plasma content become higher, will be underestimated. A possible explanation is that the reflection of Be test particles was neglected in the

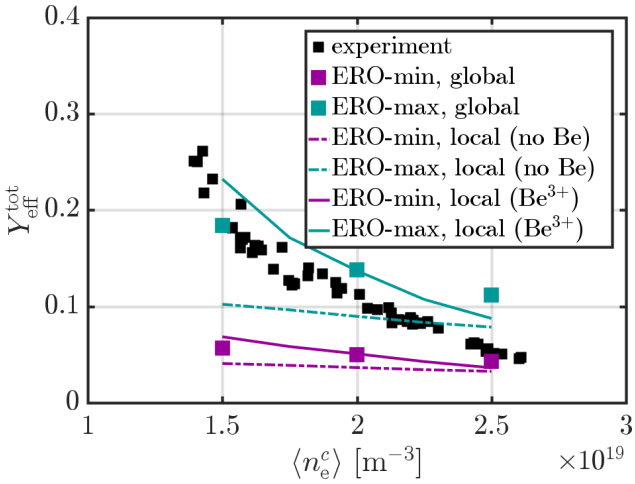


Figure 4.26: Comparison of $Y_{\mathrm{eff}}^{\mathrm{tot}}$ in the global (with self-consistent treatment of Be self-sputtering) and local modeling. The local modeling was performed with either zero Be in the plasma, or with a certain Be^{3+} background concentration, see Figure 4.23(a)-(b).

global modeling. The simple theoretical considerations presented in appendix D, as well as the sensitivity analysis presented further below in this section, indicate that reflection can increase the Be plasma content and thereby self-sputtering significantly.

Figure 4.27 shows a sensitivity analysis for $Y_{\mathrm{eff}}^{\mathrm{tot}}$ on various simulation parameters affecting the Be test particle transport. These simulations are performed with ERO-max and $\langle n_{\mathrm{e}}^{c} \rangle = 1.5 \times 10^{19}\,\mathrm{m}^{-3}$, where self-sputtering is the highest. The curve with the label '2D-flow' shows the reference case used in all previously shown global simulations. In this case, the 2D plasma flow (see Figure 4.10(a)) was included, which leads to a higher transport of eroded Be back towards the inner wall, and consequently to more self-sputtering. In comparison, the 'no flow' model, in which the plasma flow velocity was set to zero, leads to a 30 % lower $Y_{\mathrm{eff}}^{\mathrm{tot}}$.

Next, if the 3D plasma flow (see Figure 4.10(b)) behind the inner wall limiter ridge is included ('3D-flow' curve), the results remain unchanged compared to the reference case. An explanation is that most particles that are already behind the limiter ridge will hit a limiter surface within a short period of time, which is insufficient to attain the plasma flow velocity, and thus to gain sufficient energy to change the self-sputtering rates significantly. This is very convenient from a computational point of view: the toroidal symmetry assumption can be used in the plasma backgrounds, which reduces the memory (because only 2D data in the (R, z)-plane are used) and computation requirements (because bilinear instead of trilinear interpolation is used).

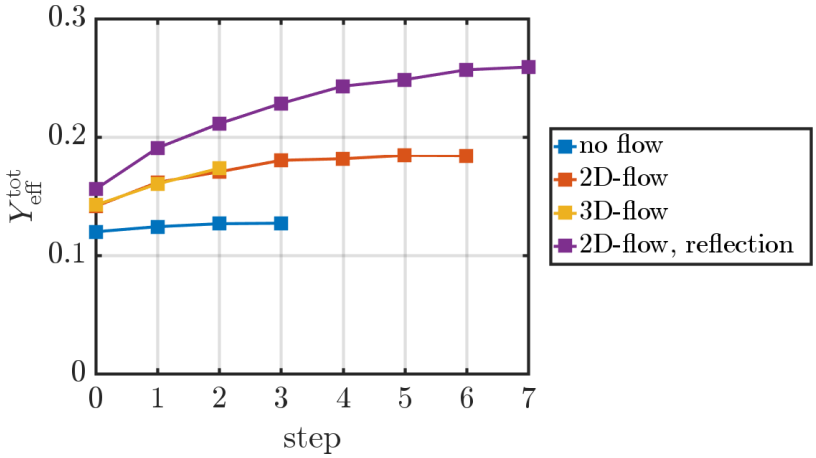


Figure 4.27: Influence of various simulation parameters affecting the Be test particle transport on the ETSY $Y_{\mathrm{eff}}^{\mathrm{tot}}$, illustrated on the example of the simulation using the plasma background with the lowest density $\langle n_{\mathrm{e}}^{c} \rangle = 1.5 \times 10^{19}\,\mathrm{m}^{-3}$ and the ERO-max assumption. Consideration of 2D flows in the plasma background and of Be reflection leads eventually to more frequent Be impact on the OS and to higher erosion by self-sputtering than without consideration of these effects ('no flow' curve).

Finally, the '2D-flow, reflection' curve shows results obtained after considering Be test particle reflection. The preliminary reflection model was implemented only very recently in ERO2.0, which is why it was not used in the other simulations shown in this chapter. The model uses TRIM data for the particle reflection coefficient $R_N(E_{\mathrm{in}}, \theta_{\mathrm{in}})$ as a function of impact energy E_{in} and angle θ_{in}. These data were obtained using a pure Be target (ERO-max); the corresponding data for 50 % D content (ERO-min) are not available at that moment. A reflected particle's angle θ_{out} is initialized using the same cosine distribution as for sputtered neutrals. Its energy is taken as $E_{\mathrm{out}} = R_E E_{\mathrm{in}}$, where R_E is the energy reflection coefficient. A more comprehensive approach to be implemented in ERO2.0 would be to sample the distribution function $f(E_{\mathrm{in}}, \theta_{\mathrm{in}}, E_{\mathrm{out}}, \theta_{\mathrm{out}})$, which is also provided by TRIM, for statistically sampling E_{out} and θ_{out} of a reflected particle. However, due to friction with the plasma during long-range migration, the initial velocity of a reflected test particle is not expected to fundamentally affect the results discussed here. With the current model, one can see that reflection increases $Y_{\mathrm{eff}}^{\mathrm{tot}}$ by about 40 %, and is thus an important mechanism that enhances self-sputtering.

In summary, the local and global ERO2.0 modeling results are in a fair agreement with each other. However, the global modeling method (which was made possible by the improvements introduced with ERO2.0) is self-consistent and therefore the preferred approach, since it does not depend on estimates of the Be plasma

concentration. The two assumptions of a pure Be surface and of a 50 % D surface content provide valid high and low estimates for the Be erosion, respectively, as confirmed by the benchmarking with the experimental ETSY. The proper treatment of Be reflection and CAPS should be implemented in ERO2.0 to round up the model and further increase the confidence in the interpretation of results. As a next step, the implementation of a more comprehensive surface model in ERO2.0 should be envisaged that allows to dynamically calculate the precise D surface content and resulting erosion, thereby overcoming the need for the ERO-min and ERO-max estimates. For this, processes such as D bulk diffusion, trapping/de-trapping at crystal defects and desorption from the surface should be taken into account. This can be done, for instance, in the framework of a reaction-diffusion model [126, 144].

Chapter 5

Summary, conclusions and outlook

5.1 Code development and global modeling results

In the frame of this thesis, the three-dimensional (3D) Monte-Carlo (MC) code ERO for studying the plasma-wall interaction (PWI) and impurity transport has undergone a major redevelopment. The new code ERO2.0 provides a number of technical innovations, which allow increasing the simulation volume substantially, and therefrom a self-consistent modeling of global erosion and deposition in a fusion experiment of ITER-relevant dimension and complexity.

Among the innovations is the massive parallelization, which is used for distributing computational effort and memory among the CPU cores of a supercomputer. Using a reference simulation scenario for JET-ILW, it is demonstrated that a significant code speedup of ∼1000 can be reached if ∼1500 CPU cores are used, providing confidence in the efficiency of the parallelization and the feasibility of even larger-scaled simulations with ERO2.0.

A new and flexible approach to the description of the wall geometry by polygon meshes (which can be obtained e.g. from technical drawings) is implemented. This makes it possible to study wall geometries of arbitrary complexity.

The geometrical algorithms of intersection and distance queries on the polygon mesh, which were identified as code performance bottlenecks, are optimized using an octree method. This leads to an increase in code performance by another factor of 10^3–10^4 when meshes with a large number of polygons ($\sim 10^5$–10^6) are used.

A new data flow and structure is introduced, which relies on the HDF5 format and allows an efficient storage, reading and writing of in- and output files of large size (∼GB) and complex structure.

A domain decomposition of 3D plasma backgrounds is implemented. This allows to use large data sets without exceeding the CPU main memory available.

For the same reason, sparse matrix algorithms are implemented for representing 3D impurity and emission densities.

Finally, a new magnetic field line tracing algorithm allows to calculate the distance of a 3D point to the wall along the magnetic field lines. This is used e.g. for the calculation of connection lengths, which determine the plasma density drop in the SOL and magnetic shadowing of PFCs.

The practical scientific gain due to the mentioned innovations is demonstrated by performing the first ERO2.0 modeling of beryllium (Be) erosion experiments at JET-ILW in limited pulses (i.e. pulses in limiter configuration), extending the similar modeling performed earlier with ERO1.0. The new global modeling, which was made possible with ERO2.0, significantly increases the simulation volume size and the PFC surface area within it. In contrast to the earlier ERO1.0 local modeling at JET-ILW, which focused on the three inner wall limiter tiles targeted by the spectroscopy diagnostic, now all inner wall limiter and liner tiles are considered, increasing the simulated PFC surface area by $\sim$220x from $0.09\,\mathrm{m}^2$ to $20\,\mathrm{m}^2$. Also, the simulation volume for impurity transport was increased by $\sim$770x from $0.06\,\mathrm{m}^3$ to $46\,\mathrm{m}^3$, now covering the entire plasma edge in 3D (spanned in the poloidal plane between the wall and the flux surface at an effective plasma radius $\rho = a/a_0 > 0.9$). Using 240 CPU cores of a supercomputer, the computational time remained below $\sim$1 day, demonstrating the feasibility of large scale simulations with ERO2.0. A further increase in the PFC surface area (e.g. by including the outer limiters and the divertor, which are however less relevant for the JET limited pulses) or the simulation volume size (e.g. by decreasing ρ for the inner boundary) is feasible in the frame of future modeling, in particular after increasing the number of CPU cores involved in the calculation.

A set of 2D poloidal plasma backgrounds, which are representative of the experimental condition and were mapped with the two-point model from experimentally measured plasma profiles, is used as an input to ERO2.0. Compared to previous studies with ERO1.0, the plasma backgrounds are increased in size to cover the global modeling simulation volume. The plasma flow patterns are calculated now using the new functionality of magnetic field line tracing in ERO2.0. Furthermore, a new alternative set of plasma backgrounds, which were calculated with the SOLEDGE2D-EIRENE plasma edge code, is obtained. The two sets show significant differences (in particular, SOLEDGE2D-EIRENE predicts a higher electron temperature) and are used as a parameter scan.

First unique results are obtained for the erosion and deposition fluxes on the 3D wall and for the distribution of Be impurities in the plasma (up to now using the two-point model plasma backgrounds only). The global modeling predicts that

most (45 %) Be particles are impacting the wall as Be^{2+} and with a mean energy of ∼220 eV and mean angle of ∼56°.

The global modeling offers a novel and powerful synthetic diagnostic, which allows comparison of the simulated Be line emission with experimental 2D images from wide-angle cameras. A first benchmarking (using a two-point model plasma background) has been performed for Be II line emission, which shows a reasonable agreement between experiment and modeling. As a result, certain deviations from experiment suggest that the temperature might be underestimated in the plasma backgrounds.

Another novel means of code validation is the comparison of the simulated power flux with the one obtained from IR camera images. A first benchmarking shows a good agreement if a two-point model plasma background is used, with certain deviations consistent with those observed for Be II line emission. A similar benchmarking is performed with SOLEDGE2D-EIRENE plasma backgrounds, which shows an overestimation of the power flux due to the higher electron temperature. As a conclusion from the cross-checking of ERO2.0 results with IR and spectroscopic wide-view camera images, both plasma background sets should be critically revised.

The comparison between modeling and experiment of the total effective sputtering yield (ETSY) $Y_{\mathrm{eff}}^{\mathrm{tot}}$, measured at a spectroscopy observation spot, was performed using both local and global modeling approaches. These show a reasonable agreement with each other, and both provide valid high and low estimates of the erosion using the assumptions of either a clean Be surface ('ERO-max') or 50 % D surface content ('ERO-min'), respectively. This result increases the confidence in the sputtering data from binary collision approximation (BCA) and molecular dynamics (MD), which are underlying the present ERO2.0 modeling.

Self-sputtering is shown to be an important contributor to Be erosion for the low density (high temperature) range of the experimental plasma conditions, which was pointed out before in [21, 24, 35]. The new global modeling provides a reliable and self-consistent way (i.e. without the need for assumptions about the plasma impurity content) of predicting the self-sputtering in such conditions.

5.2 Possible modeling improvements and future applications

As an outlook, the global ERO2.0 modeling of Be erosion and transport in JET limited pulses should be continued by performing several parameter studies and

under consideration of certain model improvements:

- A sensitivity analysis shows that SOL plasma flows and Be reflection are important mechanisms that enhance self-sputtering. The implementation of a more realistic reflection model in ERO2.0 is ongoing and should be used to repeat the analysis.
- Parameter studies are required to investigate the effect of the anomalous cross-field diffusion coefficient, which is currently treated as a constant model parameter.
- The study should be repeated using the above-mentioned refined shadowing model.
- Chemically assisted physical sputtering (CAPS) of Be under release of BeD molecules should be included in ERO2.0 as an additional erosion mechanism (which is strongly dependent on the substrate temperature), since it is known from experiments and MD modeling to account for up to 1/3 of the Be erosion in limited pulses [35].

These model improvements and parameter studies for JET limited pulses will lead to a further decrease in the uncertainties related to the erosion of Be components of the ITER-like wall. As a next step, it is furthermore desirable to implement (directly or by coupling with suitable codes or models) a dynamic surface model in ERO2.0, for instance using the reaction-diffusion model [126, 144]. This should allow to calculate the D content in the surface interaction layer, thereby overcoming the requirement for low and high estimates. Such a model would not only reduce the uncertainties related to Be sputtering yields in this present ERO2.0 application, but is also important for precise predictions of tritium retention.

The modeling of tritium retention in JET-ILW (and extrapolation to ITER) is an important task that should be envisaged with ERO2.0. This should be done in the framework of diverted pulses, since these are more relevant for the ITER baseline operation scenario [25, 43]. In JET-ILW diverted pulses, the Be eroded from the main chamber limiters migrates mainly into the inner divertor, where net deposition and fuel retention take place [34]. This has previously been studied with the WallDYN code [34, 170, 173], however in 2D without taking into account the 3D dependence of the erosion sources on the Be limiters. Preliminary ERO1.0 modeling of Be migration from the main chamber into the inner divertor has also been performed (see Figure 5.1(a)). However, due to the lack of a flexible

wall description, ERO1.0 could not take the 3D shape of the erosion sources into account. "Instead, the source was approximated by an artificial distribution that would reproduce the observed emission" [1]. Furthermore, reflection of Be at the inner divertor target plates and subsequent transport into remote divertor areas, such as the louvre or the area below the divertor tiles 5 (see Figure 5.1(b)), has been modeled with ERO1.0 [108]. The lack of a flexible wall description made it necessary to couple ERO1.0 to the 3D-GAPS code [127, 128] (which is a MC code for neutral particle transport in complex 3D wall structures, e.g. gaps and castellated surfaces, and thereby suited for handling the complex geometry shown in Figure 5.1(b)). ERO2.0 makes it possible to combine and extend the above-

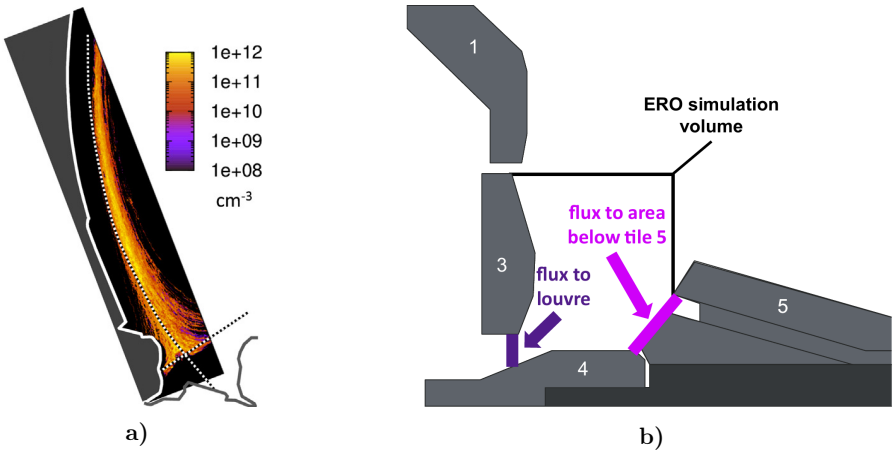


Figure 5.1: Earlier ERO1.0 modeling of Be migration in JET-ILW. (a) Migration of Be from the main chamber (with test particles created by artificial sources to match the observed emission) into the inner divertor [1]. The density of Be^{4+} averaged in the poloidal plane is shown. (b) ERO1.0 simulation volume for studying the transport of Be, which is reflected from the inner divertor target plate, into the remote areas (with coupling to 3D-GAPS) [108].

mentioned coupled ERO1.0-3D-GAPS approach in a seamless way. This would allow to self-consistently simulate the Be main chamber erosion and consequent Be migration into the divertor (including remote divertor areas) in 3D, leading to more precise estimates of fuel retention. However, this requires implementing the sputtering by charge exchange neutrals in the code, since it is the dominant main chamber erosion mechanism in diverted pulses (due to the lower plasma temperature and incident ion fluxes, resulting from to the larger gap between the limiters and the separatrix) [35]. Furthermore, the effects of ELMs in H-mode pulses and of Be/W material mixing in the divertor should also be included in the modeling.

After verifying that ERO2.0 can reproduce the experimental observations of Be erosion and transport in JET limited and diverted pulses, the code should be applied for the predictive modeling of ITER baseline scenario pulses. Thereby, the earlier ERO1.0 [25] predictions of blanket module lifetimes and fuel retention rates should be critically revised and updated if necessary. Furthermore, ERO2.0 allows predictions for the concentration of Be impurities in the core plasma, as well as for their impact on retention and tungsten sputtering in the divertor.

Aside from applications to JET and ITER, the flexible code structure of ERO2.0 facilitates the modeling application to other fusion experiments, including not only tokamaks and linear devices but also stellarators. In particular in the case of the W7-X stellarator with the complex HELIAS configuration, ERO2.0 is a perfect and currently unique candidate for PWI and impurity transport modeling, due to its high computational performance and its capability of handling complex 3D plasma backgrounds and wall geometries.

Furthermore, it should be noted that while Be and W are important for ITER, advanced high heat flux and plasma-facing materials are being investigated for usage in future fusion power plants such as DEMO [119]. These include for instance self-passivating smart alloys [121, 207] or the reduced activation ferritic-martensitic steel EUROFER [123]. Modeling the performance of such materials, with consideration of dynamic effects as for instance preferential sputtering, should be envisaged with ERO2.0. This can be done using e.g. the homogeneous mixing model (HMM) or by a dynamic coupling with the SDTrimSP code. Both these approaches have been successfully applied earlier with ERO1.0 [61, 62] for investigating the formation of mixed graphite-tungsten surface layers and their effect on tritium retention, and can act as a starting point for modeling advanced materials with ERO2.0.

Appendices

Appendix A

ERO2.0 code profiling

The open-source instrumentation software *Valgrind*[41] was used for profiling the ERO2.0 code. *Valgrind* was used for two different purposes:

- The command `valgrind --tool=massif` was used for determining the memory that is dynamically allocated on the so-called heap area of a CPU's random-access memory (RAM). The results have been discussed in section 3.6.

- The command `valgrind --tool=callgrind` was used for performance profiling, i.e. for measuring the execution time spent in individual functions of ERO2.0. This is important for identifying and optimizing bottlenecks for the code performance. As mentioned in section 3.4, the geometric operations on the polygon mesh (i.e. the distance and intersection queries) were identified from early on as the main performance bottlenecks. This is why much effort has been spent on optimizing these operations using the octree. The results of the performance profiling after the optimization is discussed in this section.

For the performance profiling, a serial simulation (i.e. using a single CPU core) has been performed, using the simulation conditions from section 4.2.4. In contrast to those simulations, only 100 instead of 10^6 test particles were used (which is sufficient for the statistics required for the profiling), since without the parallelization and with the additional runtime overhead due to *Valgrind*, the profiling would be too time-consuming.

Figure A.1(a) and (b) show the so-called callgraphs obtained from the profiling and visualized with the open-source tool *Graphviz*[42]. A callgraph displays functions as nodes of a graph, with the function names as labels. Arrows (edges of the

[41] `http://valgrind.org/`
[42] `http://www.graphviz.org/`

directed graph) indicate the calling of a function by a parent function. The arrow labels indicate the percentage of time spent relative to the root function (top node of the graph) and the number of times it was called. For clarity, the maximum depth of function calls is cropped to 2 in Figure A.1(a) and to 4 in Figure A.1(b). Also, functions in which less than 1 % of time was spent were omitted. In Figure A.1(a), the root function is *main()*, which is the first function called in the code. The graph shows that over 95 % of the total simulation time is spent in the calculation of test particle trajectories (*Ero2Simulation::transportLoop()*). The next callgraph in Figure A.1(b) starts with this function as the root function. For each of the 100 test particles in the simulation, *Ero2Simulation::transportParticleLoop()* is called, which calculates a single trajectory. A total of about 1.5 Mio. trajectory iteration steps (*Ero2Simulation::transportParticleStep()*) is performed for the particles.

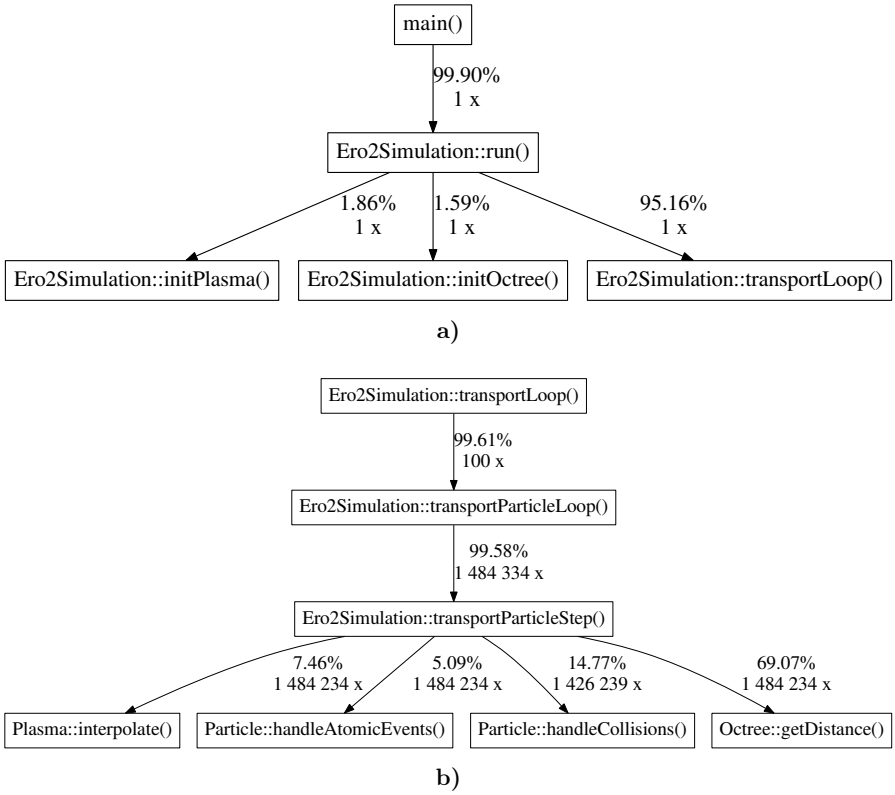


Figure A.1: Performance profiling: callgraphs of (a) the `main()` function and (b) the `Ero2Simulation::transportLoop()` function in ERO2.0, measured with `valgrind --tool=callgrind`. The graphs were produced using the *Graphviz* software.

For each time step, most time (69 %) is spent in the polygon mesh distance

query using the octree (*Octree::getDistance()*). The remaining 31 % are mostly spent for calculating the drift and diffusion in the Fokker-Planck term (*Particle::handleCollisions()*), the interpolation of plasma parameters (*Plasma::interpolate()*) and the calculation of atomic event probabilities such as ionization and recombination (*Particle::handleAtomicEvents()*). This means that the distance query is the main bottleneck for ERO2.0 performance (for the simulation conditions from chapter 4). Note that the value of 69 % is obtained after including the optimizations described in section 3.4. If the linear method is used instead of the octree method (for the distance and intersection queries), the total simulation time was found to increase by orders of magnitude. This is not unexpected, since the polygon mesh used in the simulation contains a large number of about 5×10^5 polygons. In fact, the linear method makes the code so slow that no profiling could be performed in that case. However, using 10 test particles and following these for only a few trajectory steps, it could be estimated that a simulation using the linear method would require roughly 10^4 longer than the one using the octree method. This is in line with the results shown in Figure 3.12(a), where the performance of the octree and linear methods is compared. This highlights the importance of the optimization that was described in section 3.4.

Appendix B

Classification of parallel computer systems

Parallelization refers to the concurrent execution of computing instructions (processes). For instance, modern central processing units (CPUs) are often mutli-core processors consisting of two or more independent processing units (CPU cores), which allows running instructions in parallel and thus increasing overall speed. According to Flynn's taxonomy [74], computer architectures may be classified based on the available number of concurrent instruction and data streams:

- Single-Instruction, Single-Data (SISD)
- Multiple-Instruction, Single-Data (MISD)
- Single-Instruction, Multiple-Data (SIMD)
- Multiple-Instruction, Multiple-Data (MIMD)

SISD is the conventional sequential computer (von Neumann model [200]).

In MISD, multiple processes operate on the same set of data. According to [158] this execution model is very restrictive and no commercial parallel computer of this type has ever been built.

SIMD is a highly efficient execution model for applications which involve data parallelism, i.e. when the same instruction is applied on different pieces of data. This is often the case in multimedia applications (digital image, audio and video) [158]. Computer graphics algorithms for instance, such as the rendering of 3D images, often involve vector and matrix operations. In the ideal case, an operation like adding two vectors can be performed within one instruction for all vector elements (vectorization), instead of performing it piecewise to the individual elements. On modern desktop computers, such applications are often outsourced from the more

general-purpose CPUs to the graphics processing units (GPUs) of the graphics card, which are specifically designed to perform efficient data-parallel computation. However, modern CPUs often allow a certain degree of data-parallelism as well (compiler vectorization).

A restriction of SIMD is that since there is only one program flow, conditional statements (`if-else`) have to be executed in two steps (first the processing elements in the `if`-branch, then those in the `else`-branch). This reduces code performance if a large number of conditionals are involved, which means not all applications can be efficiently parallelized using SIMD. This is specifically the case for ERO2.0, where different instructions need to be performed for different test particles (e.g. depending on whether they are ionized or neutral).

Therefore the parallelization concept of ERO2.0 relies on MIMD, which is more flexible than SIMD. MIMD involves multiple processes which operate independently on the same or on different sets of data (task parallelism). Thus, the program flow does not need to be synchronous. MIMD systems can be further classified by their memory organization:

- A shared memory system is a computer with a multi-core processor and local memory, which is physically shared among the CPU cores. The task parallelism is typically implemented by threads, which are independent control flows executed by different cores. An example of a programming interface for shared memory multiprocessing is OpenMP, which is supported by a wide range of C, C++ and Fortran compilers [194].

- A distributed memory system is a cluster of computers (nodes), each of which has its own processor and local memory. The data exchange between the individual nodes is implemented by message passing. The message passing interface (MPI) defines a standard for C, C++ and Fortran libraries which provide routines for message passing [193].

A supercomputer like JURECA, which was used for the simulations in the present work (see section 3.2.3), is a distributed memory system, where each node is internally a shared memory system. For the simulations on JURECA, ERO2.0 uses a combined (hybrid) parallelization, with MPI communication between the nodes, and MPI or OpenMP communication within the nodes. This approach is very flexible, and can be easily transferred to comparable computer systems. For instance, ERO2.0 was found to operate efficiently in parallel on the multi-core CPU of a local PC, as well as on the computer cluster of the FZJ IEK-4 institute.

Appendix C

Z_{eff} and Be plasma concentration

The electron density is given by

$$n_{\mathrm{e}} = \sum_i n_i Z_i \tag{C.1}$$

and the effective charge by [157]

$$Z_{\mathrm{eff}} = \frac{\sum_i n_i Z_i^2}{n_{\mathrm{e}}}\,, \tag{C.2}$$

where n_i are densities and Z_i are charge states of an ion species i. If Z_{eff} is measured, the equation system (C.1)-(C.2) can be solved for the individual ion concentrations if a two-component plasma is considered. In this case, we assume the presence of deuterium with $Z_{\mathrm{D}} = 1$ and beryllium of a single charge state Z_{Be}. Using equation (C.1) one can write n_{D} as

$$n_{\mathrm{D}} = n_{\mathrm{e}} - n_{\mathrm{Be}} Z_{\mathrm{Be}}\,. \tag{C.3}$$

Inserting this in equation (C.2), one can solve for the ion concentrations relative to the electron density:

$$c_{\mathrm{Be}} = \frac{n_{\mathrm{Be}}}{n_{\mathrm{e}}} = \frac{Z_{\mathrm{eff}} - 1}{Z_{\mathrm{Be}}(Z_{\mathrm{Be}} - 1)} \tag{C.4}$$

$$c_{\mathrm{D}} = \frac{n_{\mathrm{D}}}{n_{\mathrm{e}}} = \frac{Z_{\mathrm{Be}} - Z_{\mathrm{eff}}}{Z_{\mathrm{Be}} - 1} \tag{C.5}$$

This can also be transformed into the Be concentration relative to the total ion concentration:

$$f_{\mathrm{Be}} = \frac{c_{\mathrm{Be}}}{c_{\mathrm{D}} + c_{\mathrm{Be}}}\,. \tag{C.6}$$

Figure C.1(a) shows the five Z_{eff} points (fitted to the experimental values) at the $\langle n_{\mathrm{e}}^{c} \rangle$ values corresponding to the plasma backgrounds used in the modeling. Figure C.1(b) shows the corresponding concentrations calculated with equation (C.4) for $Z_{\mathrm{Be}} = 3$ and $Z_{\mathrm{Be}} = 4$. Note that following from equation (C.4), c_{Be} for $Z_{\mathrm{Be}} = 3$ is

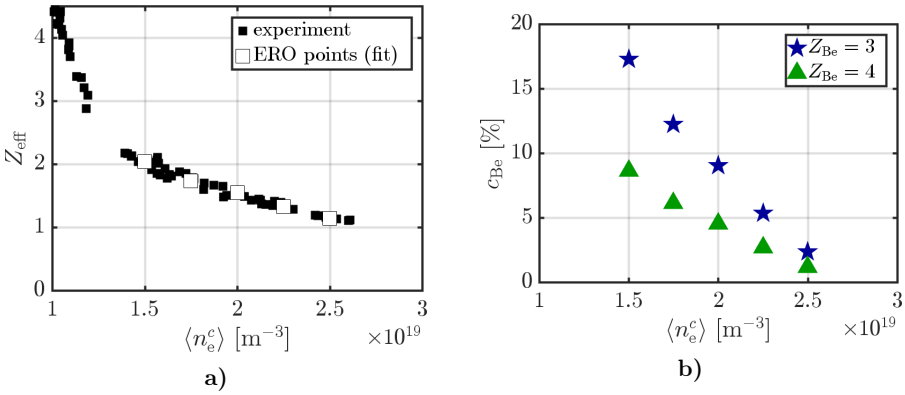


Figure C.1: (a) Effective charge Z_{eff}. (b) Estimations of the Be concentration, used for the local ERO2.0 modeling shown in Figure 4.23, from the effective charge.

exactly twice as large as for $Z_{\mathrm{Be}} = 4$. For the low density case $\langle n_{\mathrm{e}}^{c} \rangle = 1.5 \times 10^{19}\,\mathrm{m}^{-3}$ with $Z_{\mathrm{eff}} \approx 2$, a very high concentration of $c_{\mathrm{Be}} \approx 17\,\%$ ($f_{\mathrm{Be}} \approx 26\,\%$) is obtained with $Z_{\mathrm{Be}} = 3$.

Appendix D

Simple model for the steady state erosion

In this section, a simple model is presented, which explains qualitatively under which conditions multi-step ERO2.0 simulations will converge (steady state). For this model a Be surface, irradiated by a constant flux of deuterons, is considered. Under certain conditions discussed below, a steady state can be reached in which the fluxes of incoming and eroded particles remain constant. This requires the strong assumption that the surface properties, such as the morphology and material composition (e.g. due to erosion, deposition and implantation, outgassing of hydrogenic isotopes), are not affected by the irradiation. Furthermore, the presence of impurities is assumed to leave the plasma parameters unchanged. Finally, the surface geometry and spatial inhomogeneities in the plasma conditions are neglected here (0D model).

As illustrated in Figure D.1, a number N_D of deuterons hitting the Be surface will erode $N_\mathrm{D} Y_{\mathrm{Be}\leftarrow\mathrm{D}}$ Be atoms. A number $N_0 = N_\mathrm{D} Y_{\mathrm{Be}\leftarrow\mathrm{D}} P_\mathrm{ret}$ of Be particles return to the surface again. The factor $P_\mathrm{ret} \leq 1$ accounts for the fact that some particles reach remote locations where they have no further impact on the system. In the following steps,

$$\begin{aligned} N_1 &= N_0 P\,, \\ N_2 &= N_0 P^2\,, \\ N_3 &= N_0 P^3\,, \\ &\dots \\ N_k &= N_0 P^k \end{aligned} \tag{D.1}$$

particles return to the surface. The coefficient P is defined by

$$P = (R + Y_{\mathrm{Be}\leftarrow\mathrm{Be}})\, P_{\mathrm{ret}} \tag{D.2}$$

where self-sputtering with the yield $Y_{\mathrm{Be}\leftarrow\mathrm{Be}}$ and reflection with the coefficient R are considered (the energy and angle distributions of particles are neglected here, so that both $Y_{\mathrm{Be}\leftarrow\mathrm{Be}}$ and R are constants). The total number of Be atoms hitting

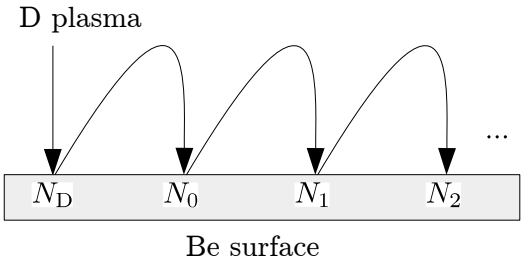


Figure D.1: Illustration of a simple model for the convergence of Be self-sputtering.

the surface is given by a geometric series, which converges if $P < 1$:

$$N_{\mathrm{Be}} = \sum_{i=0}^{\infty} N_i = \frac{N_0}{1-P} = \frac{N_{\mathrm{D}} Y_{\mathrm{Be}\leftarrow\mathrm{D}} P_{\mathrm{ret}}}{1-P}\,. \tag{D.3}$$

This corresponds to a steady state of the system. Otherwise if $P \geq 1$, the geometric series does not converge. This has the physical meaning that on average each sputtered Be particle sputters more than one other particle before being deposited again, so that the Be concentration in the plasma increases with time and no steady state is reached.

Equation (D.3) may be used to define a simple model for the total effective sputtering yield (ETSY) discussed in chapter 4 after the steady state is achieved:

$$\begin{aligned} Y_{\mathrm{eff}}^{\mathrm{tot}} &= Y_{\mathrm{Be}\leftarrow\mathrm{D}} + \frac{N_{\mathrm{Be}} Y_{\mathrm{Be}\leftarrow\mathrm{Be}}}{N_{\mathrm{D}}} = Y_{\mathrm{Be}\leftarrow\mathrm{D}} + \frac{Y_{\mathrm{Be}\leftarrow\mathrm{D}} Y_{\mathrm{Be}\leftarrow\mathrm{Be}} P_{\mathrm{ret}}}{1-P} \\ &= Y_{\mathrm{Be}\leftarrow\mathrm{D}} + \frac{Y_{\mathrm{Be}\leftarrow\mathrm{D}} Y_{\mathrm{Be}\leftarrow\mathrm{Be}} P_{\mathrm{ret}}}{1-(R+Y_{\mathrm{Be}\leftarrow\mathrm{Be}})\, P_{\mathrm{ret}}}\,. \end{aligned} \tag{D.4}$$

Figure D.2 shows as an example the results of equation (D.4) obtained with $R = 0.1$–0.4 and $P_{\mathrm{ret}} = 0.2$–0.8. $R = 0.1$–0.4 are typical values for the reflection coefficient of Be on Be, if the mean impact angles $\sim 50^\circ$ and energies $\sim 100\,\mathrm{eV}$ for particles hitting the limiter surface in the JET-ILW HFS-limited pulses (see

Figure 4.15) are assumed.[43] P_{ret} is more difficult to estimate, since the actual global migration patterns of test particles between different zones of the inner wall are complex and out of scope of the present 0D model. The selected values $P_{\mathrm{ret}} = 0.2$–0.8 are therefore somewhat arbitrary. Note that for the global ERO2.0 simulations in chapter 4, typically 80–90 % of test particles return to the inner wall (see section 4.2.4), while the rest 'vanishes' in remote regions (e.g. the divertor or the outer wall). $Y_{\mathrm{Be}\leftarrow\mathrm{D}}$ and $Y_{\mathrm{Be}\leftarrow\mathrm{Be}}$ in Figure D.2 were obtained for the temperature range of the density scan experiment $T_{\mathrm{e}}^{\mathrm{loc}} = 10$–$35\,\mathrm{eV}$ and the average magnetic angle $\theta_B = 75°$ at the observation spot (see Figure 4.24). The results are compared to the case where no self-sputtering is considered ($Y_{\mathrm{eff}}^{\mathrm{tot}} = Y_{\mathrm{Be}\leftarrow\mathrm{D}}$). Note that there is a kink at $T_{\mathrm{e}}^{\mathrm{loc}} = 20\,\mathrm{eV}$. This is a numerical artifact which comes from the coarse resolution of the available sputtering yield data, which is interpolated in $T_{\mathrm{e}}^{\mathrm{loc}}$ and θ_B using bilinear interpolation. The simple model demonstrates the following: (1) by

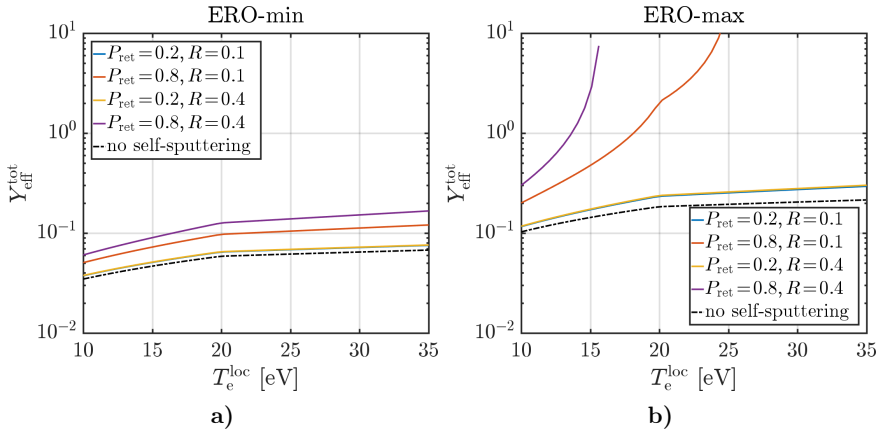


Figure D.2: Results of the simple model (D.4) for (a) ERO-min and (b) ERO-max, using different values of the parameters R and P_{ret}. $Y_{\mathrm{Be}\leftarrow\mathrm{D}}$ and $Y_{\mathrm{Be}\leftarrow\mathrm{Be}}$ are calculated as a function of temperature $T_{\mathrm{e}}^{\mathrm{loc}}$ and for the average magnetic angle $\theta_B = 75°$.

combining ERO-min at low temperatures and ERO-max at high temperatures, the experimentally found value range of $Y_{\mathrm{eff}}^{\mathrm{tot}} \approx 0.4$–$10$ (see Figure 4.5(d)) is covered, (2) self-sputtering can contribute significantly to $Y_{\mathrm{eff}}^{\mathrm{tot}}$ and even be the dominant erosion mechanism (in the case of ERO-max at high temperatures, where values > 10 are achieved), and (3) the importance of self-sputtering is determined not only by the self-sputtering yield but also by the reflection coefficient and the probability of Be particles returning to the surface.

[43] See [64, p. 28] or `http://www.eirene.de/html/beryllium.html` for the reflection coefficient of Be on Be as a function of the impact energy and angle.

Bibliography

1. M. I. Airila *et al.*, *J. Nucl. Mater.* **463**, 800–804 (2015).
2. T. Akenine-Möller, presented at the ACM SIGGRAPH 2005 Courses - SIGGRAPH '05, p. 8.
3. P. Alliez *et al.*, presented at the ACM SIGGRAPH 2005 Pap. - SIGGRAPH '05, vol. 24, p. 617.
4. G. Arnoux *et al.*, *Nucl. Fusion* **53**, 73016 (2013).
5. J. A. Bærentzen, J. Gravesen, F. Anton, H. Aanæs, *Guide to Computational Geometry Processing* (Springer London, London, 2012).
6. J. Beal, A. Widdowson, K. Heinola, K. J. Gibson (2015).
7. J. Behrens *et al.*, *Ocean Model.* **10**, 171–183 (2005).
8. R. Behrisch, W. Eckstein, *Top. Appl. Phys.* **110**, 1–526 (2007).
9. A. Benoit, L. Marchal, J.-F. Pineau, Y. Robert, F. Vivien, *IEEE Trans. Comput.* **59**, 202–217 (2010).
10. J. P. Biersack, W. Eckstein, *Appl. Phys. A Solids Surfaces* **34**, 73–94 (1984).
11. J. Biersack, L. Haggmark, *Nucl. Instruments Methods* **174**, 257–269 (1980).
12. J. A. Bittencourt, *Fundamentals of Plasma Physics* (Springer New York, New York, NY, 2004).
13. C. Björkas, K. Nordlund, *J. Nucl. Mater.* **439**, 174–179 (2013).
14. C. Björkas *et al.*, *J. Nucl. Mater.* **438**, S276–S279 (2013).
15. C. Björkas, K. Vörtler, K. Nordlund, D. Nishijima, R. Doerner, *New J. Phys.* **11**, 123017 (2009).
16. C. Björkas *et al.*, *Plasma Phys. Control. Fusion* **55**, 074004 (2013).
17. E. Böhm-Vitense, *Introduction to stellar astrophysics Volume 3* (Cambridge University Press, 1989).
18. B. Bondarenko, *Uspekhi Fiz. Nauk* **171**, 886 (2001).

19. J. P. Boris, R. A. Shanny, U. S. O. of Naval Research, N. R. Laboratory, *Proceedings: Fourth Conference on Numerical Simulation of Plasmas, November 2, 3, 1970* (Naval Research Laboratory, 1972).

20. D. Borodin *et al.*, *Nucl. Mater. Energy* (2017).

21. D. Borodin *et al.*, *Nucl. Mater. Energy* **0**, 1–6 (2016).

22. D. Borodin *et al.*, *J. Nucl. Mater.* **390-391**, 106–109 (2009).

23. D. Borodin *et al.*, *J. Nucl. Mater.* **438** (2013).

24. D. Borodin *et al.*, *Phys. Scr.* **T159**, 014057 (2014).

25. D. Borodin *et al.*, *Phys. Scr.* **T145**, 014008 (2011).

26. I. E. Borodkina, M. Komm, I. V. Tsvetkov, *Russ. Phys. J.* **58**, 438–445 (2015).

27. I. Borodkina *et al.*, *Nucl. Mater. Energy* **0**, 1–5 (2017).

28. I. Borodkina *et al.*, *Contrib. to Plasma Phys.* **56**, 640–645 (2016).

29. M. Botsch, *Polygon mesh processing* (A K Peters, 2010).

30. A. M. Bradshaw, T. Hamacher, U. Fischer, *Fusion Eng. Des.* **86**, 2770–2773 (2011).

31. S. I. Braginskii, *Transport processes in a plasma*, 1965.

32. D. W. Brenner, *MRS Bull.* **21**, 36–41 (1996).

33. S. Brezinsek, *J. Nucl. Mater.* **463**, 11–21 (2015).

34. S. Brezinsek *et al.*, *Nucl. Fusion* **55**, 63021 (2015).

35. S. Brezinsek *et al.*, *Nucl. Fusion* **54**, 103001 (2014).

36. M. Brix *et al.*, *Rev. Sci. Instrum.* **79**, 10F325 (2008).

37. R. A. Brizzolara, C. Cooper, T. K. Olson, *Nucl. Instruments Methods Phys. Res. Sect. B Beam Interact. with Mater. Atoms* **35**, 36–42 (1988).

38. J. N. Brooks, *Phys. Fluids B Plasma Phys.* **2**, 1858–1863 (1990).

39. J. N. Brooks, J. P. Allain, D. A. Alman, D. N. Ruzic, *Fusion Eng. Des.* **72**, 363–375 (2005).

40. N. H. Brooks *et al.*, *Rev. Sci. Instrum.* **79**, 10F330 (2008).

41. H. Bufferand *et al.*, *J. Nucl. Mater.* **438**, S445–S448 (2013).

42. D. Campbell, *Europhys. News* **47**, 28–31 (2016).

43. S. Carpentier *et al.*, *J. Nucl. Mater.* **415**, S165–S169 (2011).

44. R. Chodura, *Phys. Fluids* **25**, 1628 (1982).

45. R. E. H. Clark, D. Reiter, *Nuclear Fusion Research*, ed. by R. E. Clark, D. H. Reiter (Springer-Verlag, Berlin/Heidelberg, 2005), vol. 78.

46. J. W. Coenen *et al.* (2016).

47. J. W. Coenen *et al.*, *Nucl. Fusion* **53**, 73043 (2013).

48. R. J. Colchin, D. L. Hillis, R. Maingi, C. C. Klepper, N. H. Brooks, *Rev. Sci. Instrum.* **74**, 2068–2070 (2003).

49. J. W. Connor, H. R. Wilson, *Plasma Phys. Control. Fusion* **42**, R1–R74 (2000).

50. J. Cooper, *Reports Prog. Phys.* **29**, 302 (1966).

51. P. C. de Vries *et al.*, *Nucl. Fusion* **51**, 053018 (2011).

52. L. Devroye, *Non-uniform random variate generation* (Springer-Verlag, 1986), p. 843.

53. R. Ding *et al.*, *Nucl. Fusion* **57**, 056016 (2017).

54. R. Ding *et al.*, *Nucl. Mater. Energy* (2017).

55. R. Ding *et al.*, *Nucl. Fusion* **55**, 23013 (2015).

56. R. Ding *et al.*, *J. Nucl. Mater.* **415**, S270–S273 (2011).

57. M. Djomehri, R. Biswas, M. Potsdam, R. Strawn, presented at the Proc. Int. Parallel Distrib. Process. Symp. P. 9.

58. R. Doerner, M. Baldwin, D. Buchenauer, G. De Temmerman, D. Nishijima, *J. Nucl. Mater.* **390-391**, 681–684 (2009).

59. R. Doerner, M. Baldwin, D. Nishijima, *J. Nucl. Mater.* **455**, 1–4 (2014).

60. R. Doerner, C. Björkas, D. Nishijima, T. Schwarz-Selinger, *J. Nucl. Mater.* **438**, S272–S275 (2013).

61. S. Droste *et al.*, *Contrib. to Plasma Phys.* **46**, 628–634 (2006).

62. S. Droste *et al.*, *Plasma Phys. Control. Fusion* **50**, 015006 (2008).

63. D. Duffy, *Mater. Today* **12**, 38–44 (2009).

64. W. Eckstein (2002).

65. W. Eckstein, *Vacuum* **82**, 930–934 (2008).

66. W. Eckstein, in *Sputtering by Part. Bombard.* (Springer Berlin Heidelberg, Berlin, Heidelberg, 2007), pp. 33–187.

67. W. Eckstein, H. M. Urbassek, in *Sputtering by Part. Bombard.* (Springer Berlin Heidelberg, Berlin, Heidelberg, 2007), pp. 21–31.

68. A. Eksaeva *et al.*, *Nucl. Mater. Energy* **000**, 1–8 (2017).

69. C. Ericson, *Real-time collision detection* (Elsevier, 2005), vol. 1.

70. European Fusion Development Agreement (EFDA), "Fusion Electricity: A roadmap to the realisation of fusion energy", tech. rep.

71. D. Fasel, M. Q. Tran, *Fusion Eng. Des.* **75-79**, 1163–1168 (2005).

72. Y. Feng *et al.*, *Contrib. to Plasma Phys.* **44**, 57–69 (2004).

73. M. Firdaouss, V. Riccardo, V. Martin, G. Arnoux, C. Reux, *J. Nucl. Mater.* **438**, S536–S539 (2013).

74. M. J. Flynn, *IEEE Trans. Comput.* **C-21**, 948–960 (1972).

75. J. P. Freidberg, *Plasma Physics and Fusion Energy* (Cambridge University Press, Cambridge, 2007).

76. H. G. Frerichs, PhD thesis, 2010.

77. W. Fundamenski, R. A. Pitts, J. E. Contributors, *Plasma Phys. Control. Fusion* **48**, 109–156 (2005).

78. M. G. Ganchenkova, V. A. Borodin, *Phys. Rev. B* **75**, 054108 (2007).

79. M. G. Ganchenkova, V. A. Borodin, R. M. Nieminen, *Phys. Rev. B* **79**, 134101 (2009).

80. M. Ganchenkova, P. Vladimirov, V. Borodin, *J. Nucl. Mater.* **386-388**, 79–81 (2009).

81. P. Gaudio *et al.*, presented at the Fusion Eng. Des. Vol. 88, pp. 1293–1296.

82. S. H. Glenzer *et al.*, *Science (80-.).* **327**, 1228–1231 (2010).

83. C. Gormezano *et al.*, *Phys. Rev. Lett.* **80**, 5544–5547 (1998).

84. H. Gouraud, *IEEE Trans. Comput.* **C-20**, 623–629 (1971).

85. H. Hagen, H. Müller, G. M. Nielson, *Focus on Scientific Visualization*, ed. by H. Hagen, H. Müller, G. M. Nielson (Springer Berlin Heidelberg, Berlin, Heidelberg, 1993).

86. A. G. Harmsen, M. D. Hoover, F. A. Seiler, *J. Nucl. Mater.* **122**, 821–826 (1984).

87. P. Helander *et al.*, *Plasma Phys. Control. Fusion* **54**, 124009 (2012).

88. A. Herrmann *et al.*, *Plasma Phys. Control. Fusion* **37**, 17–29 (1995).

89. B. R. L. Hirsch, G. Kulcinski, R. Shanny, *Issues Sci. Technol.* **13**, 1–8 (1997).

90. E. M. Hollmann *et al.*, presented at the J. Nucl. Mater. Vol. 415, S425–S429.

91. L. Horton, presented at the Fusion Eng. Des. Vol. 88, pp. 434–439.

92. J. D. Huba, "NRL: Plasma Formulary", tech. rep. (Naval Research Laboratory Washington, 2004).

93. A. Huber *et al.*, presented at the Fusion Eng. Des. Vol. 88, pp. 1361–1365.

94. A. Iiyoshi *et al.*, *Nucl. Fusion* **39**, 1245–1256 (1999).

95. F. Imbeaux *et al.*, *Comput. Phys. Commun.* **181**, 987–998 (2010).

96. JET KS3 spectroscopy wiki page, *https://users.euro-fusion.org/openwiki/-index.php/KS3/documents*.

97. H. Jin, R. Van der Wijngaart, presented at the Proc. 18th Int. Parallel Distrib. Process. Symp. Pp. 6–15.

98. Jülich Supercomputing Centre, *JLSRF* **2** (2016).

99. A. Kallenbach *et al.*, *Plasma Phys. Control. Fusion* **55**, 124041 (2013).

100. E. Källne, J. Källne, *Phys. Scr.* **T17**, 152–165 (1987).

101. G. Kawamura, Y. Tomita, A. Kirschner, *J. Nucl. Mater.* **438**, S909–S912 (2013).

102. A. Kirschner, *Fusion Sci. Technol.* **61**, 230–245 (2012).

103. A. Kirschner *et al.*, *J. Nucl. Mater.* **390-391**, 152–155 (2009).

104. A. Kirschner *et al.*, *J. Nucl. Mater.* **328**, 62–66 (2004).

105. A. Kirschner *et al.*, *J. Nucl. Mater.* **290-293**, 238–244 (2001).

106. A. Kirschner *et al.*, *Contrib. to Plasma Phys.* **56**, 622–627 (2016).

107. A. Kirschner *et al.*, *to be Submitt. to Plasma Phys. Control. Fusion* (2017).

108. A. Kirschner *et al.*, *J. Nucl. Mater.* **463**, 116–122 (2015).

109. A. Kirschner, V. Philipps, J. Winter, *Nucl. Fusion* **40**, 989 (2000).

110. T. Klinger, in *Magn. Fusion Energy* (Elsevier, 2016), pp. 493–505.

111. U. Kögler, F. Weschenfelder, J. Winter, *J. Nucl. Mater.* **243**, 816–820 (1997).

112. U. Kögler, J. Winter, PhD thesis.

113. A. Kukushkin, H. Pacher, V. Kotov, G. Pacher, D. Reiter, *Fusion Eng. Des.* **86**, 2865–2873 (2011).

114. A. Lasa, K. Heinola, K. Nordlund, *Nucl. Fusion* **54**, 123021 (2014).

115. A. Lasa *et al.*, *Nucl. Fusion* **58**, 016046 (2018).

116. A. Lasa, K. Schmid, K. Nordlund, *Phys. Scr.* **T159**, 014059 (2014).

117. J. D. Lawson, *Proc. Phys. Soc. Sect. B* **70**, 6–10 (1957).

118. J. Linke, *Phys. Scr.* **T123**, 45–53 (2006).

119. C. Linsmeier *et al.*, *Nucl. Fusion* **57**, 092007 (2017).

120. X. Litaudon *et al.*, *Nucl. Fusion* **57**, 102001 (2017).

121. A. Litnovsky *et al.*, *Nucl. Fusion* **57**, 066020 (2017).

122. L. Lucas, M. Unterweger, *J. Res. Natl. Inst. Stand. Technol.* **105**, 541 (2000).

123. E. Lucon *et al.*, *Fusion Eng. Des.* **81**, 917–923 (2006).

124. J. Luxon, *Nucl. Fusion* **42**, 313 (2002).

125. G. F. Matthews *et al.*, *Phys. Scr.* **T145**, 14001 (2011).

126. D. Matveev *et al.*, *Nucl. Instruments Methods Phys. Res. Sect. B* **(submitted** (2018).

127. D. Matveev *et al.*, *Phys. Scr.* **T159**, 014063 (2014).

128. D. Matveev *et al.*, *Plasma Phys. Control. Fusion* **52**, 075007 (2010).

129. M. Mayer *et al.*, *J. Nucl. Mater.* **390-391**, 538–543 (2009).

130. M. D. McCool, J. Reinders, A. Robison, *Structured parallel programming: patterns for efficient computation* (Elsevier/Morgan Kaufmann, 2012).

131. S. Meyers, *Effective C++: 55 Specific Ways to Improve Your Programs and Designs* (Addison Wesley, 3rd, 2005), vol. 43, arXiv: `arXiv:1011.1669v3`.

132. J. Miettunen *et al.*, *J. Nucl. Mater.* **438**, S612–S615 (2013).

133. W. Möller, W. Eckstein, *Nucl. Instruments Methods Phys. Res. Sect. B Beam Interact. with Mater. Atoms* **2**, 814–818 (1984).

134. D. C. Montgomery, D. A. Tidman, *Plasma kinetic theory* (McGraw-Hill, New York, 1964).

135. A. Mutzke, W. Eckstein, *Nucl. Instruments Methods Phys. Res. Sect. B Beam Interact. with Mater. Atoms* **266**, 872–876 (2008).

136. D. Naujoks, R. Behrisch, 843–846 (1992).

137. D. Naujoks, J. Roth, K. Krieger, G. Lieder, M. Laux, *J. Nucl. Mater.* **210**, 43–50 (1994).

138. D. Naujoks, R. Behrisch, J. P. Coad, L. D. Kock, *Nucl. Fusion* **33**, 581–590 (1993).

139. D. Naujoks, R. Behrisch, V. Philipps, B. Schweer, *Plasma Phys. {...}* **36**, 2021–2030 (1994).

140. R. Neu *et al.*, *Fusion Eng. Des.* **109-111**, 1046–1052 (2016).

141. D. Nishijima, R. Doerner, M. Baldwin, G. De Temmerman, *J. Nucl. Mater.* **390-391**, 132–135 (2009).

142. D. Nishijima, R. P. Doerner, M. J. Baldwin, G. De Temmerman, E. M. Hollmann, *Plasma Phys. Control. Fusion* **50**, 125007 (2008).

143. K. Nordlund *et al.*, *Phys. Rev. B* **57**, 7556–7570 (1998).

144. M. Oberkofler, M. Reinelt, C. Linsmeier, *Nucl. Instruments Methods Phys. Res. Sect. B Beam Interact. with Mater. Atoms* **269**, 1266–1270 (2011).

145. J. Ongena, R. Koch, R. Wolf, H. Zohm, *Nat. Phys.* **12**, 398–410 (2016).

146. J. Pamela, E. Solano, *From JET to ITER. Preparing the next step in fusion research*, 2001.

147. N. J. Peacock, D. C. Robinson, M. J. Forrest, P. D. Wilcock, V. V. Sannikov, *Nature* **224**, 488–490 (1969).

148. V. Philipps, P. Wienhold, A. Kirschner, M. Rubel, *Vacuum* **67**, 399–408 (2002).

149. B. T. Phong, B. Tuong, *Commun. ACM* **18**, 311–317 (1975).

150. A. Piel, *Plasma physics: An introduction to laboratory, space, and fusion plasmas*, arXiv: `arXiv:1011.1669v3`.

151. R. A. Pitts *et al.*, *J. Nucl. Mater.* **438**, S48–S56 (2013).

152. R. A. Pitts *et al.*, *Nucl. Fusion* **47**, 1437–1448 (2007).

153. R. A. Pitts *et al.*, *J. Nucl. Mater.* **415**, S957–S964 (2011).

154. A. Portone, M. Roccella, R. Roccella, F. Lucca, G. Ramogida, *Fusion Eng. Des.* **83**, 1619–1624 (2008).

155. H. Qin *et al.*, *Phys. Plasmas* **20**, 84503 (2013).

156. R. Rabenseifner, G. Hager, G. Jost, presented at the 2009 17th Euromicro Int. Conf. Parallel, Distrib. Network-based Process. Pp. 427–436.

157. S. K. Rathgeber *et al.*, *Plasma Phys. Control. Fusion* **52**, 095008 (2010).

158. T. Rauber, G. Rünger, *Parallel programming : for multicore and cluster systems.* (Springer, 2014).

159. M. H. Redi, R. V. Budny, D. C. McCune, C. O. Miller, R. B. White, *Phys. Plasmas* **3**, 3037 (1996).

160. D. Reiser, D. Reiter, M. Tokar', *Nucl. Fusion* **38**, 165–177 (1998).

161. D. Reiser, "Zur Anwendung der driftkinetischen Theorie in Monte-Carlo-Studien zum Verunreinigungstransport in Tokamak-Plasmen", report.

162. D. Reiter, M. Baelmans, P. Börner, *Fusion Sci. Technol.* **47**, 172–186 (2005).

163. F. L. Ribe, *Rev. Mod. Phys.* **47**, 7–41 (1975).

164. J. Romazanov *et al.*, *Phys. Scr.* **T170**, 014018 (2017).

165. J. Roth, J. Bohdansky, W. Ottenberger, "Data on Low Energy Light Ion Sputtering", tech. rep.

166. J. Roth *et al.*, *Plasma Phys. Control. Fusion* **50**, 103001 (2008).

167. E. Safi *et al.*, *J. Nucl. Mater.* **463**, 805–809 (2015).

168. E. Safi, G. Valles, A. Lasa, K. Nordlund, *J. Phys. D. Appl. Phys.* **204003**, 204003 (2017).

169. U. Samm, *Fusion Sci. Technol.* **47**, 73–75 (2005).

170. K. Schmid, K. Krieger, S. W. Lisgo, G. Meisl, S. Brezinsek, *J. Nucl. Mater.* **463**, 66–72 (2015).

171. K. Schmid, M. Mayer, C. Adelhelm, M. Balden, S. Lindig, *Nucl. Fusion* **50**, 105004 (2010).

172. K. Schmid, M. Reinelt, K. Krieger, *J. Nucl. Mater.* **415**, S284–S288 (2011).

173. K. Schmid *et al.*, *Nucl. Fusion* **55**, 53015 (2015).

174. P. J. Schneider, D. H. Eberly, *Geometric tools for computer graphics* (Boston, 2003).

175. K. R. Schultz, *IEEE Control Syst.* **26**, 32–34 (2006).

176. C. Silva *et al.*, *Nucl. Fusion* **54**, 83022 (2014).

177. C. Silva *et al.*, *Plasma Phys. Control. Fusion* **51**, 105001 (2009).

178. R. Simonini, G. Corrigan, G. Radford, J. Spence, A. Taroni, *Contrib. to Plasma Phys.* **34**, 368–373 (1994).

179. L. Spitzer, *Phys. Fluids* **1**, 253 (1958).

180. W. M. Stacey, Wiley InterScience (Online service), *Fusion : an introduction to the physics and technology of magnetic confinement fusion* (Wiley-VCH, 2010).

181. M. F. Stamp, K. Krieger, S. Brezinsek, *J. Nucl. Mater.* **415**, S170–S173 (2011).

182. P. C. Stangeby, *Plasma Phys. Control. Fusion* **43**, 223–224 (2000).

183. P. C. Stangeby, *J. Nucl. Mater.* **176-177**, 51–64 (1990).

184. P. C. Stangeby, *Nucl. Fusion* **52**, 83012 (2012).

185. P. C. Stangeby, J. Elder, *J. Nucl. Mater.* **196-198**, 258–263 (1992).

186. R. Stankiewicz, R. Zagórski, *J. Nucl. Mater.* **313-316**, 899–903 (2003).

187. R. Stankiewicz, R. Zagórski, *J. Nucl. Mater.* **337-339**, 191–195 (2005).

188. L. Stephey *et al.*, *Rev. Sci. Instrum.* **87**, 11D606 (2016).

189. H. P. Summers, M. G. O'Mullane, in *Nucl. Fusion Res. Underst. Plasma-Surface Interact.* Ed. by R. E. H. Clark, D. H. Reiter (Springer Berlin Heidelberg, Berlin, Heidelberg, 2005), pp. 399–413.

190. H. P. Summers *et al.*, *Plasma Phys. Control. Fusion* **44**, B323–B338 (2002).

191. H. P. Summers, *The ADAS User Manual version 2.6*, 2004.

192. The HDF Group webpage, *www.hdfgroup.org/HDF5*.

193. The MPI Forum webpage, *http://mpi-forum.org*.

194. The OpenMP API specification, *http://www.openmp.org*.

195. P. R. Thomas *et al.*, *Phys. Rev. Lett.* **80**, 5548–5551 (1998).

196. M. W. Thompson, *Philos. Mag.* **18**, 377–414 (1968).

197. M. Z. Tokar, M. Koltunov, *Nucl. Fusion* **53**, 093014 (2013).

198. M. Z. Tokar, R. Ding, M. Koltunov, *Plasma Phys. Control. Fusion* **52**, 075003 (2010).

199. G. A. van Swaaij *et al.*, *Plasma Phys. Control. Fusion* **54**, 095013 (2012).

200. J. von Neumann, *IEEE Ann. Hist. Comput.* **15**, 27–75 (1993).

201. F. Wagner *et al.*, *Phys. Rev. Lett.* **53**, 1453–1456 (1984).

202. F. Wagner *et al.*, *Phys. Rev. Lett.* **49**, 1408–1412 (1982).

203. M. Wakatani, *Stellarator and heliotron devices* (Oxford University Press, 1998).

204. A. A. Ware, *Phys. Rev. Lett.* **25**, 15–17 (1970).

205. F. Warmer *et al.*, *Fusion Eng. Des.* **123**, 47–53 (2016).

206. A. Weckmann *et al.*, *Phys. Scr.* **T167**, 014058 (2016).

207. T. Wegener *et al.*, *Nucl. Mater. Energy* **9**, 394–398 (2016).

208. J. Wesson, D. J. Campbell, *Tokamaks* (Oxford University Press, 2011).

209. L. A. Westover, PhD thesis, University of North Carolina, 1991.

210. S. Wiesen *et al.*, *J. Nucl. Mater.* **463**, 480–484 (2015).

211. W. D. Wilson, L. G. Haggmark, J. P. Biersack, *Phys. Rev. B* **15**, 2458–2468 (1977).

212. G. Z. Wu, J. Gu, J. Dai, in (Springer, Singapore, 2018), pp. 67–77.

213. Z. Xu *et al.*, *Rev. Sci. Instrum.* **87**, 11D429 (2016).

214. W. Yuanxi, L. Jiangang, W. Peide, E. Team, *Plasma Sci. Technol.* **8**, 253–254 (2006).

215. R. Zagórski, R. Ivanova-Stanik, R. Stankiewicz, *Nucl. Fusion* **53**, 073030 (2013).

216. J. F. Ziegler, J. P. Biersack, in *Treatise Heavy-Ion Sci.* (Springer US, Boston, MA, 1985), pp. 93–129.

Danksagung (Acknowledgements)

I gratefully acknowledge the computing time granted by the JARA-HPC Vergabegremium and provided on the JARA-HPC Partition part of the supercomputer JURECA (Project ID JIEK43) at Forschungszentrum Jülich.

Diese Promotion ist am Institut für Energie- und Klimaforschung – Plasmaphysik (IEK-4) des Forschungszentrums Jülich entstanden. Ich möchte mich gerne bei Allen bedanken, die mich bei der Promotion unterstützten und zu ihrem Gelingen beitrugen.

Zunächst danke ich meinem Doktorvater, Herrn Prof. Dr. Christian Linsmeier, für die freundliche Aufnahme ins Institut, die Übernahme der Betreuung, die Unterstützung während der gesamten Promotion, sowie das Probelesen und die Begutachtung des Manuskripts.

Ich danke meinem Zweitbetreuer, Herrn Prof. Dr. Achim von Keudell, für die hilfsbereite Übernahme der Begutachtung und für die positive Resonanz im Arbeitsgruppenseminar an der Ruhr-Universität Bochum.

Mein großer Dank geht an Herrn Dr. Dmitriy Borodin, durch dessen Betreuung meine Promotion ermöglicht wurde, für die unzähligen wissenschaftlichen Diskussionen, die mich sehr voranbrachten und bei denen ich viel lernte. Seine Ideen, insbesondere zur Parallelisierung und zum Benutzen von 2D-Kamerabildern für die Validierung, haben maßgeblich zum Gelingen der Arbeit beigetragen. Mein Dank gilt auch dem sorgfältigen Korrekturlesen dieses Manuskripts, sowie der immer hilfsbereiten Unterstützung bei JET-Experimenten, Konferenzen und anderen Gegebenheiten.

Ich danke herzlich Herrn Dr. Andreas Kirschner, ohne dessen professionelle Unterstützung das Gelingen der Promotion ebenfalls undenkbar wäre. Ich danke für seinen Humor und seine aufmunternden Worte, die mich in schwierigen Zeiten aufgebaut haben. Ich danke auch für die Unterstützung bei der Einarbeitung in den ERO-Code und für zahlreiche hilfreiche Ratschläge. Speziell möchte ich danken für das überaus sorgfältige und hilfreiche Korrekturlesen dieses Manuskripts.

Ich danke Herrn Dr. Dirk Brömmel vom Jülich Supercomputing Centre (JSC) für seine Ratschläge zur Code-Parallelisierung, zum Performance-Profiling und

zum Supercomputer JURECA, sowie für das Korrekturlesen meines Zeitschriftenartikels. Herrn Prof. Dr. Paul Gibbon und Herrn Benedikt Steinbusch vom JSC danke ich für die Hilfe bei der Planung der Code-Entwicklung zu Beginn und das Feedback während der Promotion. Weiterhin danke ich für die überaus freundliche Bereitstellung von JURECA-Rechenzeit zur Entwicklung von ERO2.0.

Meinem (Büro-)Nachbarn, Herrn Dr. Dmitry Matveev, danke ich für unsere Freundschaft und seinen menschlichen Halt, sowie für Diskussionen zum Thema 3D-Wandgeometrien, die mir sehr bei der Code-Entwicklung geholfen haben.

Ich danke Herrn Dr. Sebastijan Brezinsek, dessen Vorlesung zur Plasmaphysik an meiner ehemaligen Universität in Duisburg-Essen mich dazu ermunterte, in die Fusionsforschung zu gehen. In seiner Tätigkeit als Koordinator unserer Arbeitsgruppe trug er maßgeblich zum Gelingen dieser Promotion bei. Er nahm sich außerdem viel Zeit für meine Fragen zu JET-Experimenten, und half enorm bei der Planung, Durchführung und Auswertung von solchen.

Herrn Dr. Dirk Reiser danke ich, dass er sich Zeit nahm für meine vielen Fragen zur Plasma-Theorie. Ich habe bei diesen Gesprächen unglaublich viel gelernt (und eines davon hat sogar eine gemeinsame Veröffentlichung angestoßen), außerdem war er mit seiner wunderbar geschriebenen Dissertation ein großes Vorbild und Motivation für mich.

Frau Prof. Dr. Taina Kurki-Suonio und ihren Kollegen von der Aalto-Universität in Espoo, Finnland, danke ich für die Gastfreundschaft während meines Besuches und für interessante wissenschaftliche Diskussionen zu Simulationen des Plasma-Verunreinigungstransports.

Herrn Dr. Alexander Huber, Frau Dr. Valentina Huber und Herrn Dr. Scott Silburn vom CCFE in Culham, England, danke ich für die Hilfe bei der Analyse von experimentellen 2D-Kamerabildern.

Herrn Dr. Mehdi Firdaouss und Herrn Dr. Hugo Bufferand vom CEA in Aix-en-Provence, Frankreich, danke ich jeweils für die Bereitstellung von JET-Wandgeometriedaten und SOLEDGE2D-Plasmahintergründen.

Frau Angelika Hallmanns, Frau Judith Schönbrenner und Frau Marlene Hoffmann danke ich für die liebenswürdige Hilfe bei Reiseplanungen und vielen anderen organisatorischen Belangen.

Ich danke allen weiteren Kollegen vom IEK-4, speziell den Doktoranden, für die tolle Arbeitsatmosphäre.

Schließlich gilt mein besonderer Dank meiner Familie, insbesondere meinen Eltern, Schwiegereltern und meiner Schwester, für ihre liebevolle Unterstützung. Meiner Frau Ola und meiner Tochter Xenia kann ich nie genug danken für den großen Beistand und die Inspiration, die sie mir jeden Tag geben.

Band / Volume 434
Adhäsionsverhalten von wässrigen Nafion-Lösungen an dispersen Phasengrenzen
A. Schulz (2018), xii, 129 pp
ISBN: 978-3-95806-354-9

Band / Volume 435
Alterungs- und fehlertolerante optimale Betriebsführung eines Direktmethanol-Brennstoffzellensystems
R. Keller (2018), XX, 175 pp
ISBN: 978-3-95806-355-6

Band / Volume 436
Chamber study of biogenic volatile organic compounds: plant emission, oxidation products and their OH reactivity
Y. Zhujun (2018), ix, 139 pp
ISBN: 978-3-95806-356-3

Band / Volume 437
Characterization of High Temperature Polymer Electrolyte Fuel Cells
Y. Rahim (2018), iii, 162 pp
ISBN: 978-3-95806-359-4

Band / Volume 438
Lattice Boltzmann Simulation in Components of Polymer Electrolyte Fuel Cell
J. Yu (2018), ii, 173 pp
ISBN: 978-3-95806-360-0

Band / Volume 439
Quantitative Luminescence Imaging of Solar Cells
V. Huhn (2018), 155 pp
ISBN: 978-3-95806-363-1

Band / Volume 440
Characterization of Phosphoric Acid Doped Polybenzimidazole Membranes
Y. Lin (2018), II, IV, 140 pp
ISBN: 978-3-95806-364-8

Band / Volume 441
Degradation Study of SOC Stacks with Impedance Spectroscopy
Y. Yan (2018), 135 pp
ISBN: 978-3-95806-367-9

Band / Volume 442
Future Grid Load of the Residential Building Sector
L. Kotzur (2018), xxi, 213 pp
ISBN: 978-3-95806-370-9

Band / Volume 443
Yttriumoxid-Dünnschichten als Tritium-Permeationsbarriere
J. Engels (2018), 252 pp
ISBN: 978-3-95806-371-6

Band / Volume 444
Inverse conditioning of a high resolution integrated terrestrial model at the hillslope scale: the role of input data quality and model structural errors
S. Gebler (2018), xxii, 160 pp
ISBN: 978-3-95806-372-3

Band / Volume 445
Cathode Stability and Processing in Inert Substrate-Supported Solid Oxide Fuel Cells
E. Matte (2018), viii, 178 pp
ISBN: 978-3-95806-373-0

Band / Volume 446
Aging and Degradation Behavior of Electrode Materials in Solid Oxide Fuel Cells (SOFCs)
X. Yin (2018), x, 103 pp
ISBN: 978-3-95806-374-7

Band / Volume 447
TRENDS 2017
Transition to Renewable Energy Devices and Systems
D. Stolten, R. Peters (Eds.) (2018), 206 pp
ISBN: 978-3-95806-376-1

Band / Volume 448
3D simulation of impurity transport in a fusion edge plasma using a massively parallel Monte-Carlo code
J. Romazanov (2018), xvi, 149 pp
ISBN: 978-3-95806-377-8

Weitere ***Schriften des Verlags im Forschungszentrum Jülich*** unter
http://wwwzb1.fz-juelich.de/verlagextern1/index.asp

Energie & Umwelt / Energy & Environment
Band / Volume 448
ISBN 978-3-95806-377-8

Mitglied der Helmholtz-Gemeinschaft

JÜLICH
Forschungszentrum