UNIVERSIDAD COMPLUTENSE DE MADRID
FACULTAD DE CIENCIAS FÍSICAS

TESIS DOCTORAL

Simulación de la evolución de defectos en materiales irradiados de interés en fusión nuclear mediante un método GPU-OKMC

Simulation of defect evolution in irradiated materials of interest for nuclear fusion using a GPU-OKMC method

MEMORIA PARA OPTAR AL GRADO DE DOCTOR

PRESENTADA POR

Fernando Jiménez Piñero

Director

Christophe Juan Ortiz Boutillier

Madrid

UNIVERSIDAD COMPLUTENSE DE MADRID

FACULTAD DE CIENCIAS FÍSICAS

TESIS DOCTORAL

Simulación de la evolución de defectos en materiales irradiados de interés en fusión nuclear mediante un método GPU-OKMC

Simulation of defect evolution in irradiated materials of interest for nuclear fusion using a GPU-OKMC method

MEMORIA PARA OPTAR AL GRADO DE DOCTOR

PRESENTADA POR

Fernando Jiménez Piñero

DIRECTOR

Christophe Juan Ortiz Boutillier

UNIVERSIDAD
COMPLUTENSE
MADRID

FACULTAD DE CIENCIAS FÍSICAS

Doctorado en Física

Simulación de la evolución de defectos en materiales irradiados de interés en fusión nuclear mediante un método GPU-OKMC

Simulation of defect evolution in irradiated materials of interest for nuclear fusion using a GPU-OKMC method

Autor: Fernando Jiménez Piñero

Director: Christophe Juan Ortiz Boutillier

Laboratorio
Nacional
de Fusión
Ciemat

Agradecimientos

En primer lugar, desearía agradecer a mi director de tesis, Christophe J. Ortiz, su esfuerzo e implicación en este trabajo, nada del cual habría sido posible sin él. El Dr. Ortiz me ha hecho un regalo mucho mayor que la dirección de una tesis: me ha enseñado a pensar como un científico. Cuando empecé a trabajar en este proyecto, yo tan solo era un estudiante recientemente egresado de la universidad; ahora puedo considerarme un hombre de ciencia gracias a las enseñanzas de mi director. Todos los conocimientos que he adquirido, la visión global que me ha dado la ciencia, las ideas en que se fundamenta este trabajo y el hecho de comprender cómo encaja esta tesis en el camino hacia la fusión nuclear comercial, se los debo a él. Espero que este trabajo, aunque sea una gota en el océano, contribuya a la resolución de la cuestión energética mundial.

En segundo lugar, es de justicia agradecer el gran trabajo realizado por mi compañero Carlo L. Guerrero, cuya infinita paciencia y capacidad de sacrificio han sido para mí no solo una valiosa experiencia laboral, sino toda una lección de vida.

Muchas personas han tenido parte de responsabilidad en la consecución de este proyecto. Mis compañeros Julio Calvo, Elisabetta Carella, María Isabel Carmona, Darío A. Cruz, Juan Manuel García, Juan Mauricio García, Víctor Gutiérrez, Begoña Gómez-Ferrer, Edgar León, Montserrat Martín, Fernando Mota, José Antonio Pavón, Juan Antonio Ramírez, Marcelo Roldán, Raquel Román y Rafael Saavedra han sido una constante fuente de apoyo personal y profesional durante todo el periodo que he pasado en el Laboratorio Nacional de Fusión – CIEMAT, de forma que ellos fueron quienes me transmitieron tanto una buena parte de la motivación necesaria para

sacar este proyecto adelante como la respuesta a un sinnúmero de dudas acerca de la física de materiales. Asimismo, mis compañeros Fernando Arranz, Teresa Fernández, María González, M. Pilar Hernández, Ángel Ibarra, Piedad Martín, Joaquín Mollá, Fernando J. Sánchez, Marina Verdú y Rafael A. Vila han estado siempre disponibles para corregir mis errores de principiante y satisfacer mis inquietudes científicas con total claridad y respeto. Dicen que la experiencia es un grado, y todos ellos me lo han demostrado con creces. En general, debo agradecer al Laboratorio Nacional de Fusión - CIEMAT esta oportunidad para desarrollarme como científico.

En cuanto a la programación en tarjetas gráficas relevante para este trabajo, específicamente en cuanto al modelo CUDA, reconozco mi absoluta ignorancia previa al comienzo de esta tesis doctoral. Esto cambió muy pronto gracias a la didáctica dedicación e inestimable ayuda de Miguel Cárdenas y Juan José Rodríguez. Ellos dos provocaron que yo fuera capaz de plasmar las ideas del ámbito de la teoría a la práctica. Gracias a ambos por su tiempo. De la misma manera, agradezco a Nicolas Castin y Dmitry Terentyev muchas de las respuestas a las preguntas que tenía sobre la simulación de experimentos. No querría olvidarme de Christophe Domain en estos agradecimientos, dada su cortesía a la hora de realizar simulaciones por su cuenta para validar nuestro trabajo.

Agradezco sinceramente la ayuda que me prestó Abel F. Paz durante años a la hora de ejecutar las simulaciones en el sistema CETA-CIEMAT. Su paciencia y eficiencia son un modelo para todos.

Más allá de cuestiones profesionales, querría incluir a mis amigos personales en mis agradecimientos, con especial mención a Saúl López, tristemente fallecido durante el desarrollo de esta tesis, en octubre de 2015. Nunca te olvidaremos.

Por último, quiero dedicar esta tesis doctoral a mis padres y a mi hermano Carlos. Su apoyo durante todos estos años de trabajo, y durante toda mi vida, no conoce límites, y su amor y paciencia son infinitos. Confío en que este trabajo los haga sentir orgullosos de mí. Gracias también al resto de mi familia, sobre todo a mis primos Miguel Ángel, Víctor Manuel, Paloma, Javier y David, que tanto se han preocupado por mí sin pedir nada a cambio. No quiero olvidar a María Ángeles ni a Marina, que desde hace pocos años son también parte de mi familia. En particular, Marina me allanó el camino para la impresión de esta tesis.

Este trabajo se ha llevado a cabo haciendo uso de la infraestructura de com-

putación facilitada por el Centro Extremeño de Tecnologías Avanzadas (CETA-CIEMAT), financiado por el Fondo Europeo de Desarrollo Regional (FEDER). El CETA-CIEMAT pertenece al CIEMAT y al Gobierno de España.

This work has been carried out within the framework of the EUROfusion Consortium and has received funding from the Euratom research and training programme 2014-2018 under grant agreement No 633053. The views and opinions expressed herein do not necessarily reflect those of the European Commission.

Madrid, septiembre de 2022

Contents

List of Tables

List of Figures

List of Abbreviations

AKMC	Atomistic Kinetic Monte Carlo
BCA	Binary Collision Approximation
bcc	Body-centred cubic
BKL	Bortz-Kalos-Leibowitz
CPU	Central processing unit
CUDA	Compute Unified Device Architecture
DBTT	Ductile-to-brittle transition temperature
DCLL	Dual-Coolant Lithium Lead
DEMO	Demonstration Power Plant
DL	Dislocation loop
fcc	Face-centred cubic
FLOPs	Floating-point operations per second
FPKMC	First-Passage Kinetic Monte Carlo
GFRD	Green's Function Reaction Dynamics
GPGPU	General-purpose programming on graphics processing units
GPU	Graphics processing unit
HCLL	Helium-Cooled Lithium Lead

HCPB	Helium-Cooled Pebble Bed
HiPER	High Power Laser Energy Research Facility
I	Interstitial
ICF	Inertial confinement fusion
ITER	International Thermonuclear Experimental Reactor
JET	Joint European Torus
KLV	Kalos-Levesque-Verlet
KMC	Kinetic Monte Carlo
LAKIMOCA	Lattice Kinetic Monte Carlo
LLCB	Lithium-Lead Ceramic Breeder
MD	Molecular Dynamics
MEGA-OKMC	Microstructure Evolution GPU-based Accelerated Object Kinetic Monte Carlo
M4F	Multiscale Modelling for Fusion and Fission Materials
MFRT	Mean Field Rate Theory
MPI	Message-Passing Interface
NBI	Neutral beam injection
NIF	National Ignition Facility
ODS	Oxide-dispersion strengthened
OKMC	Object Kinetic Monte Carlo
OpenMP	Open Multi Processing
PKA	Primary knock-on atom
PTLA	Parallel Tau-leaping algorithm
RAFM	Reduced activation ferritic-martensitic
RAM	Random-access memory

RIC	Radiation-induced conductivity
RIED	Radiation-induced electrical degradation
RR	Resistivity recovery
SIA	Self-interstitial atom
SSA	Stochastic simulation algorithm
TBM	Test blanket module
TEM	Transmission electron microscopy
Tokamak	Toroidal'naya kamera s magnitnymi katushkami
V	Vacancy
WCCB	Water-Cooled Ceramic Breeder
WCLL	Water-Cooled Lithium Lead

Summary

As the current world reliance on fossil fuels proves to have catastrophic environmental consequences, which are only exacerbated with a growing world economy and population, a future clean source of energy is required. The scientific community expects nuclear fusion to fulfil this task, in particular magnetically confined fusion. To achieve this, an experimental fusion reactor, the ITER Project, is underway and shall provide the basis for a future demonstration power plant, known as DEMO.

One of the most important challenges in the design of a future nuclear fusion reactor is the choice of materials. Materials are subjected to an intense flux of neutrons and heat in a fusion reactor like ITER or, in a much more pronounced way, DEMO. Under irradiation, a large amount of defects are created and, as a consequence, the properties of materials are severely degraded, and may cause the reactor components to malfunction or break.

Therefore, the choice of materials in fusion reactors depends greatly on their response to neutron irradiation. However, in order to replicate the behaviour of materials under conditions similar to ITER or DEMO, very energetic sources of neutron irradiation are needed, but these are not readily available. Hence, in the last decades, the use of computational simulations has become a fast and inexpensive method to study the behaviour of materials under irradiation, as they help avoid long and costly experiments. In addition, atomistic mechanisms which are not easily accessible to characterisation can be investigated with the help of simulations.

A number of techniques are available to simulate microstructural defect evolution. In this work, our choice was the Object Kinetic Monte Carlo (OKMC) method,

which treats each defect (or particle) as an individual object with a set of parameters. The OKMC method is renowned for its accuracy and flexibility; nevertheless, its computational cost is substantial. In particular, when the number of particles in the system is very large, the resulting time step, which is approximately inversely proportional to the amount of particles, is very small, and many computational steps are needed to finish the simulation.

In order to bypass this problem, in our approach, the time step is first imposed on the particle system, and then the nature of events that are undergone by the particles during this time step is established and their amount calculated. Specifically, it is possible to sample the amount of performed events with a certain event rate from a Poisson distribution. Since the event rates of each particle are, in principle, independent of the rest of particles, these computations can be done for all particles in parallel. For this purpose, an algorithm labelled the parallel Tau-leaping algorithm (PTLA) was developed and implemented on graphics cards. First, the sums of event rates displayed by each particle in the system are calculated, and then the time step is computed as inversely proportional to the maximum of these sums of event rates. Thus, the time step only depends on the maximum sum of event rates available in the system, which is a property of a specific particle, and is therefore independent of the total number of particles. This is a clear advantage over *classic* OKMC models, as the time step independence of the number of particles in the system allows for time steps which are orders of magnitude larger. In turn, this offers better statistical treatment for the study of systems with larger amounts of particles; particularly, as far as particles at low density are concerned.

However, in some cases, the PTLA is not sufficient to offer good performance, so several approximations are required. For instance, when a few particles with very large sums of event rates are present in the system, the resulting time step is very small. If these few particles do not contribute to the evolution of the system as a whole, a large number of computational steps are needed to observe any significant change, which is an obstacle to performance. We included an approximation to solve this problem, in which these few particles do not count when the time step is computed. Therefore, the resulting time step is larger and less computational effort is required to finish the simulation.

An additional technique, known as the Green's Function Reaction Dynamics

(GFRD) method, was applied in this work for systems at low density. In this case, particles are able to perform many migrations before meeting another particle to interact with, such that a diffusion coefficient can be defined. Then, these migrations can be replaced by a single jump, and the time step is advanced accordingly. The GFRD method reduces the number of necessary computational steps by several orders of magnitude, and is appropriate for the study of low-density systems with an important number of very mobile particles. In this dissertation, the application of the GFRD method in our work is detailed, making special emphasis on systems under continuous irradiation.

Our work is validated with a number of different test cases and simulations, including the simulation of ion irradiation experiments in Fe and FeCr, legitimising our choice for a parallel model based on graphics cards and justifying the assumptions that were made for the PTLA and the included approximations. The fact that large amounts of very mobile particles, such as dislocation loops, can be treated with our model in realistic runtimes (from minutes to a few days), makes it a valuable tool for the simulation of microstructural evolution in nuclear materials.

Resumen

Dado que la actual dependencia mundial de los combustibles fósiles muestra ciertas consecuencias catastróficas para el medio ambiente, las cuales son magnificadas a medida que crecen la economía y población mundiales, se necesita una fuente de energía limpia para el futuro. La comunidad científica espera que sea la fusión nuclear la que desempeñe este papel, en particular la fusión por confinamiento magnético. Para ello, un reactor de fusión experimental, el Proyecto ITER, está en marcha y proporcionará las bases para un futuro reactor de demostración llamado DEMO.

Uno de los desafíos principales en el diseño de un futuro reactor de fusión es la elección de los materiales. En efecto, los materiales serán sometidos a un flujo intenso de neutrones y calor en un reactor de fusión como ITER; y, de forma más pronunciada, en uno como DEMO. Esto provocará la creación de una gran cantidad de defectos, por lo que las propiedades de los materiales serán gravemente alteradas, y podrán provocar que los componentes del reactor dejen de funcionar correctamente o, incluso, se quiebren.

Por lo tanto, la elección de materiales para reactores de fusión depende, en gran medida, de su respuesta a la irradiación de neutrones. No obstante, para representar el comportamiento de materiales bajo condiciones similares a ITER o DEMO, se necesitan unas fuentes de irradiación neutrónica muy potentes que habitualmente no están disponibles. Por esta razón, el uso de simulaciones computacionales se ha impuesto en las últimas décadas como un método rápido y económico para investigar el comportamiento de materiales bajo irradiación ya que permiten evitar experimentos largos y costosos y además, permiten estudiar mecanismos atomísticos difícilmente accesibles a la caracterización.

Existen varias técnicas disponibles para simular la evolución microestructural de defectos. En este trabajo, elegimos el método de Montecarlo cinético (OKMC), en el cual se trata cada defecto (o partícula) como un objeto individual con una serie de parámetros. El método OKMC es reconocido por su precisión y flexibilidad, pero es computacionalmente muy costoso ya que el paso de tiempo es aproximadamente inversamente proporcional al número de partículas. Hace falta por lo tanto un número muy grande de pasos para acabar la simulación de un sistema grande de partículas.

Para evitar este problema, en nuestro modelo se impone primero un paso de tiempo en el sistema de partículas, y, a continuación, se determina la naturaleza de los eventos experimentados por las partículas durante este paso de tiempo, así como el número de los mismos. Propiamente hablando, el número de eventos con una cierta tasa de transición llevados a cabo por una partícula se puede determinar tomando muestras de una distribución de Poisson. Dado que las tasas de transición para una partícula son, en principio, independientes del resto de partículas, podemos hacer estos cálculos para todas las partículas en paralelo. Para ello, hemos desarrollado un algoritmo llamado algoritmo Tau-leaping paralelo (PTLA) y lo implementamos en tarjetas gráficas. Primero, se calculan las sumas de tasas características de cada partícula, y después se calcula el paso de tiempo siendo inversamente proporcional al máximo de estas sumas. Por tanto, el paso de tiempo solo depende de la máxima suma de tasas de transición, la cual es una propiedad de una partícula en concreto. Así, el paso de tiempo es independiente del número total de partículas. Esta es una clara ventaja sobre modelos OKMC *clásicos*, ya que la independencia del paso de tiempo del número de partículas en el sistema permite alcanzar pasos de tiempo órdenes de magnitud más grandes y por lo tanto, el estudio de sistemas con un mayor número de partículas, lo cual ofrece un mejor tratamiento estadístico; en particular, en lo relativo a partículas a baja densidad.

Sin embargo, en ciertos casos, el PTLA no es suficiente para ofrecer buen rendimiento computacional y se necesitan aproximaciones. Por ejemplo, cuando hay en el sistema unas pocas partículas con grandes sumas de tasas de transición, el paso de tiempo resultante es muy pequeño. Si dichas partículas no contribuyen significativamente a la evolución del sistema, hacen falta muchos cálculos para observar algún cambio, lo cual afecta negativamente al rendimiento de la simulación. Para resolver este problema, aplicamos una aproximación en la que estas pocas partículas no cuentan para el cálculo del paso de tiempo. Por tanto, el paso de tiempo resultante es mayor, y hace falta menos esfuerzo computacional para terminar la simulación.

Una técnica adicional, conocida como método GFRD (Green's Function Reaction Dynamics), fue aplicada en este trabajo para sistemas a baja densidad. En tal caso, las partículas pueden realizar muchas migraciones antes de encontrar otras partículas con las cuales interactuar, de forma que se puede definir un coeficiente de difusión. Entonces, se pueden reemplazar dichas migraciones por un solo salto, y avanzar el paso de tiempo en consecuencia. El método GFRD reduce el número de pasos de cálculo en varios órdenes de magnitud, y es adecuado para estudiar sistemas a baja densidad con un gran número de partículas muy móviles. En esta tesis, se detalla cómo se aplicó el método GFRD en nuestro trabajo, haciendo especial hincapié en los sistemas bajo irradiación continua.

Nuestro trabajo se valida con varios casos de prueba y distintas simulaciones, incluyendo la simulación de experimentos de irradiación iónica en Fe y FeCr, lo que acredita nuestra elección de un modelo paralelo basado en tarjetas gráficas, y justifica las conjeturas hechas para el PTLA y las aproximaciones incluidas. El hecho de poder tratar grandes poblaciones de partículas muy móviles, tales como bucles de dislocación, en tiempos de ejecución realistas (de minutos a unos pocos días), demuestra que nuestro modelo es una herramienta valiosa para la simulación de la evolución de la microestructura en materiales nucleares.

Chapter 1

Introduction

In this chapter, we present the motivation behind commercial nuclear fusion, the material engineering challenges associated with it, and the need to develop accurate and powerful simulation tools to predict material behaviour in future fusion power plants. In section 1.1, we explore the possibility of nuclear fusion as an energy source and describe several approaches that aim at its viability. Afterwards, section 1.2 is devoted to the materials that are considered as candidates for the construction of nuclear fusion devices, in particular tokamaks, and the processes that take place in these materials when subjected to neutron irradiation. Finally, section 1.3 shall describe the main motivations and objectives of our work, and outline the structure of this doctoral thesis.

1.1 Nuclear fusion

In this section, nuclear fusion is discussed as a future source of energy.

1.1.1 Motivation: the energy question

The demand for energy in the world has been increasing ever since the Industrial Revolution. Due to their availability and inexpensive price, fossil fuels (mainly carbon, oil and methane) have been traditionally chosen as the main source of energy by humans. As world population grows, so do national economies, and more of these fuels are needed for development.

This world dependence on fossil fuels is problematic for two reasons. First, the amount of fossil fuels available on Earth is finite, which indicates that the current reliance on them is not an option for the development of world economy in the future. Second, burning fossil fuels is widely considered an environmental hazard. Indeed, the consumption of fossil fuels leads to an increase of gas emissions to the Earth's atmosphere, particularly carbon dioxide, which negatively impacts the environment in the form of global warming and climate change [93, 162, 219]. Figure 1.1 shows the time evolution of yearly CO_2 emissions in metric tonnes, as studied by Boden et al. [43], and presents an exponential increase thereof since the start of industrialisation, in the middle of the 19th century.

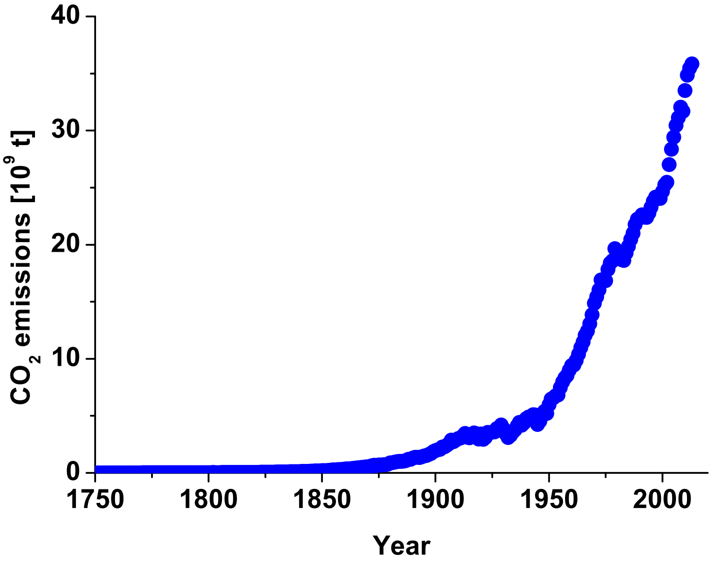


Figure 1.1: Historical record of carbon dioxide emissions [43].

Hence, an alternative to fossil fuels is needed in the midterm. Research on clean energy sources is underway, since the current ones (solar and wind power or hydroelectricity, for example) are unfortunately unable to meet the world demand for energy. Indeed, this demand is expected to increase to 10 TW in the twenty-first century [89] which cannot be covered with these clean energy sources. On the other hand, the long-time environmental hazards that nuclear fission energy is prone to disqualify nuclear fission as a long-term solution. The radioactive waste produced by fission power plants must be handled and disposed of; for instance, some isotopes in spent nuclear fuel are very long-lived, like ^{239}Pu and ^{237}Np. The respective half-lives of these two isotopes are 2.4×10^4 years and 2.1×10^6 years [52]. Therefore, long-lived radioactive elements must be stored for a long time under high-level security conditions in deep underground pools, which must be built at locations that are not prone to natural disasters. Moreover, the impact of a nuclear power plant on the environment may become catastrophic, as evidenced by several accidents in the past. Examples include Kyshtym (1957), where a failure in the cooling system of a storage tank led to a chemical explosion [152]; Chernobyl (1986), where a combination of human errors during a safety test and faulty plant design caused a power spike and a steam explosion [208]; or Fukushima (2011). In Fukushima, an earthquake triggered the automatic shutdown of nuclear reactions. Power generation was derivated to emergency generators, but large tsunami waves flooded the site and damaged key plant systems, including the generators. As a consequence, the cooling system experienced a loss of power and three reactors suffered a meltdown [140].

In order to satisfy world energy demand in the future in a reliable and clean manner, the most promising candidate is controlled nuclear fusion, for several reasons. First, the nuclear fusion reaction, discussed in the next subsection, only requires isotopes of hydrogen as a fuel. Deuterium, for instance, is readily available in water, the main component of the Earth's surface. Tritium, on the other hand, can be gained from neutron bombardment of lithium, which is very abundant in the Earth's crust and in sea water [89]. Thus, nuclear fusion should serve as an energy source for humanity without any concerns about resource depletion. Second, long-term radioactivity concerns are not an issue in nuclear fusion. All potentially activated materials from the fusion reaction chamber have short half-lives, in the order of 100 years, and the half-life of tritium is 12 years [89, 307]. This means that security questions are much easier to resolve than in the case of nuclear fission energy. Finally, no production of greenhouse gases is expected in a nuclear fusion power plant, other than the expected carbon footprint involving construction and operation of the plant [89]. For all these reasons, the prospect of a nuclear fusion power plant seems like the best option for humanity's future as far as energy and economy are concerned.

1.1.2 The nuclear fusion reaction

We shall now review the basics of nuclear fusion. When two atom nuclei with respective masses m_1 and m_2 react, a new nucleus with a mass $m_3 < m_1 + m_2$ is created. The difference in mass $\Delta m = m_1 + m_2 - m_3$ is released as energy E, according to Eq. 1.1 [94, 307]:

$$E = \Delta m \cdot c^2 \tag{1.1}$$

where c is the speed of light.

A number of fusion reactions are possible. For example, in stellar cores, protons react with each other to produce He by way of two known processes, the proton-proton chain and the CNO cycle [2]. In the CNO cycle, which is relevant in the evolution of heavy stars [2, 37], transmutation of C, N and O nuclei occurs through a series of proton capture and beta decay reactions. These elements transmute into other elements at several steps of the CNO cycle but are recreated at other steps, thus acting as catalysts for the fusion of protons into He. As far as the proton-proton reaction is concerned, which governs energy production in lighter stars [2, 47], the necessary conditions to achieve it are very hard to reach in an experiment on Earth. This is a consequence of its small cross-section [47], so other candidate reactions must be considered instead. Figure 1.2 depicts the cross-sections of different nuclear fusion reactions as a function of temperature, calculated with the formulae provided by Bosch and Hale [47].

Clearly, the fusion process that shows the largest cross-section at the lowest temperature is the deuterium-tritium reaction. This fact makes this reaction the most easily achievable fusion reaction in an experiment, and therefore the foremost candidate for a future fusion power plant. The deuterium-tritium reaction yields:

$$^2D + {}^3T \rightarrow {}^4He(3.5\ MeV) + n(14.1\ MeV) \tag{1.2}$$

For a fusion reaction to happen, matter must be in the plasma state. Plasma is the most common state of matter in the visible Universe, present in various forms such as stellar cores or coronas. A plasma is composed of charged particles like negative electrons and positive nuclei. Thus, the nuclear fusion reaction can happen only if Coulomb repulsion between the nuclei is overcome, which requires a very

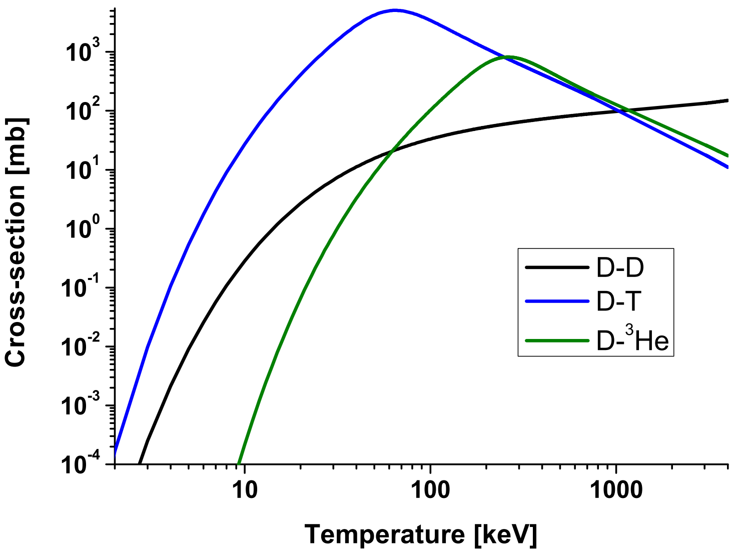


Figure 1.2: Cross-sections of different nuclear fusion reactions [47].

high temperature. An important aspect of nuclear fusion is that the plasma must be confined, i.e. kept to a certain volume and at high temperature, in order to sustain the reaction. If the plasma is not confined, it rapidly loses energy and the necessary conditions to achieve nuclear fusion are no longer satisfied. Confinement time τ_E measures the energy loss rate of a fusion plasma, such that a large τ_E value indicates better confinement [307].

Good confinement is necessary in order to use a fusion plasma as an energy source. When fusion products heat the plasma enough to guarantee its sustenance, without the need of any additional external heating, the plasma is said to have reached ignition. The product of plasma density n, temperature T and confinement time τ_E, called the triple product $nT\tau_E$, is used as a figure of merit to estimate when the conditions for ignition are met. If $nT\tau_E$ exceeds a certain lower limit, the plasma is expected to ignite. In the case of the deuterium-tritium reaction shown in Eq. 1.2, this value is approximately $nT\tau_E > 10^{21}\ m^{-3} \cdot keV \cdot s$ [77]. The goal of a nuclear fusion power plant is to achieve these conditions and ignite the plasma, so that more energy is extracted from the fusion plasma than is invested into producing it.

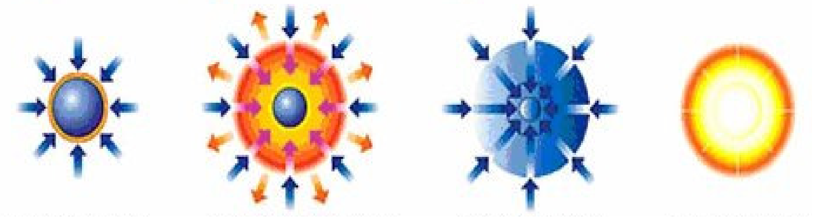


Figure 1.3: The four stages of ICF: heating, blowoff, compression and burn [209].

In stellar cores, for instance, the plasma is confined using the gravity field of the star itself. This type of plasma confinement is labelled gravitational confinement. Gravitational confinement of plasmas is, obviously, not possible on Earth. Thus, other measures are necessary. So far, two techniques have been developed to confine fusion plasmas in laboratory experiments, namely inertial confinement and magnetic confinement. The next two subsections are dedicated to them respectively.

1.1.3 Inertial confinement fusion

One way to produce a fusion plasma in controlled conditions is via inertial confinement fusion (ICF). A frozen deuterium-tritium pellet is rapidly heated by laser or ion beams, such that its outer layer is evaporated. As a reaction, an inward shockwave is created that compresses the pellet core to a very high density plasma in a small time interval. In this state, the ions are not able to move apart during this time frame due to their own inertia (hence the name), and nuclear fusion is accomplished. As far as the triple product is concerned, ICF seeks plasma ignition by increasing density n, since the confinement time τ_E is very small [138].

This method is skizzed in Figure 1.3 [209]. From left to right, the four images portray the four stages of ICF, namely the heating of the pellet, the evaporation of the outer layer (also known as blowoff), the fuel compression, and finally the thermonuclear burn. Blue arrows represent laser beams, whereas orange arrows represent the blowoff of the outer material layer.

The most relevant ICF devices in existence use laser beams to heat the target. In the most intuitive configuration, called the direct drive method, the laser beams focus on the target directly. To achieve the conditions necessary for fusion, a very precise positioning of the beams is necessary. In an alternative approach, labelled indirect drive, the target is located in a cylindrical hohlraum, which absorbs the laser energy. Then, the hohlraum releases X-rays which impact the target. The indirect drive configuration leads to a more uniform target compression than in the direct drive approach, since the hohlraum is designed to provide almost symmetric X-ray radiation, and removes the necessity of extreme precision concerning laser positioning. However, it is less energy-efficient than the direct method, as most of the laser power is spent by heating the hohlraum [166].

The most important facilities that explore ICF include:

1. The National Ignition Facility (Livermore, USA) [138, 172]: In operation since 2009, NIF is equipped with the most powerful laser beams in the world as of 2017. In a 2014 article [137], the NIF team reported a net energy gain by fusion reactions.

2. Laser Mégajoule (Bordeaux, France) [66]: This device started operation in 2014. It is part of the French military *Simulation* programme, and its objective is the study of material behaviour when exposed to simulated nuclear detonations.

3. FIREX-1 Project (Osaka, Japan) [264]: Two laser systems (the older Gekko-XII, completed in 1983, and the newer LFEX, operational since 2009) are used in cooperation for this ICF experiment. The aim is to achieve fast ignition. In a fast ignition experiment, a second energetic, short-pulse laser is fired into the already imploded target core after the first beam has impacted the target.

Additionally, a European project called HiPER (High Power Laser Energy Research Facility), still in the preliminary design phase [112], is a planned European ICF device which seeks to achieve ignition of the fusion plasma with the use of smaller, less energetic laser beams than NIF. If this experiment is successful, the energy gain shall surpass all ICF experiments to date.

1.1.4 Magnetically confined fusion

Another way to confine a fusion plasma is by the use of magnetic fields. Indeed, since fusion plasmas are ensembles of charged particles, they are responsive to magnetism via the Lorentz force, and can be directed with an appropriate use of magnetic fields. Magnetically confined plasmas display low density [307], which makes it challenging to produce a large triple product. For this reason, magnetically confined fusion devices try to reach long confinement times in order to achieve ignition. Two types of configurations are possible to confine the plasma with magnetic fields, namely the magnetic mirror and the toroidal configuration.

Magnetic mirrors pinch the field lines at two ends, such that the magnetic field intensity (or 'density of magnetic field lines') varies between one end and the other. In such a structure, the field is more intense at the ends and weaker in the middle. Charged particles, except those which travel parallel or almost parallel to the magnetic field lines, are thus reflected between both ends and confined in between. Unfortunately, due to the instabilities that are inherent to fusion plasmas, some particles that were initially confined may be scattered to a direction which is parallel to the field lines and escape the device. This leads to a certain leakage of plasma that makes magnetic mirrors a suboptimal choice for a power plant based on magnetically confined fusion [87].

In fact, most modern experiments concerning magnetically confined fusion are based on toroidal confinement. Our work, for instance, is mainly oriented towards the materials used in this kind of devices. In a toroidal configuration, magnetic field lines are closed on themselves, such that no notion of 'ends' exists. These nuclear fusion devices are inspired by two basic designs, the tokamak and the stellarator. We shall now describe both of them.

1.1.4.1 Devices in magnetically confined fusion: tokamaks and stellarators

The initial design of a working device capable to induce nuclear fusion using magnetic confinement is called Tokamak. *Tokamak* is an abbreviation of *Toroidal'naya kamera s magnitnymi katushkami*, which is Russian for *toroidal chamber with magnetic coils*. A tokamak is designed as a toroidal plasma chamber, labelled the vacuum vessel, surrounded by magnetic coils which create a toroidal field. However, this is not enough to guarantee good confinement. Since the magnetic field $\vec{B}$ is not homogeneous, the gradient $\vec{\nabla} B$ generates a particle drift in the vertical direction. Moreover, electrons and ions have separate charges, so, even though they drift in the same direction, they move toward opposite orientations. The resulting separation of charges creates an electric field $\vec{E}$ which induces another drift in $\vec{E} \times \vec{B}$ direction [132]. As a consequence, a poloidal magnetic field is also necessary to offset the drift and confine the plasma.

Tokamaks are equipped with a second set of coils, called poloidal field coils, to induce a toroidal current inside the plasma, which in turn generates the poloidal magnetic field. This strong current is the origin of serious instabilities that can ultimately break the confinement in a process called disruption [307, 323]. Disruptions cause the plasma to violently hit the walls of the plasma chamber. This leads to a rapid deposition of particles and thermal energy that damages wall components and erodes the material. To avoid disruptions, most tokamaks work with pulsed operations rather than in steady state.

In Europe, several tokamaks are in operation such as the following:

1. Joint European Torus (JET) [307]: Located in Culham (United Kingdom), JET started operation in 1983. JET is the foremost tokamak as far as European fusion science is concerned. In 1997, JET was able to achieve a world record 16 MW of fusion power [154]. A sketch of the JET is provided in Figure 1.4, taken from *The science of JET* by J. Wesson [307].

2. ASDEX Upgrade [331]: This tokamak is situated in Garching bei München (Germany) and started operation in 1991. Instead of radioactive tritium, the plasma fed to this device is a mixture of hydrogen (protium) and deuterium.

3. WEST [53, 287]: Having been recently built in Cadarache (France), the first plasma was launched in 2016. WEST is expected to test the limits of tokamak technology for future reactors.

An alternative design to tokamaks, the stellarator, solves the drifting problem by carefully changing the geometry of the torus and the magnetic coils. In a stellarator, no current is induced inside the plasma. Instead, the poloidal field is produced by external currents in the complex magnetic coils. Hence, the instabilities related to plasma interaction with a strong current and potential disruptions are absent, which makes stellarators better suited than tokamaks for steady-state operation. European experiments based on the stellarator configuration include, for example:

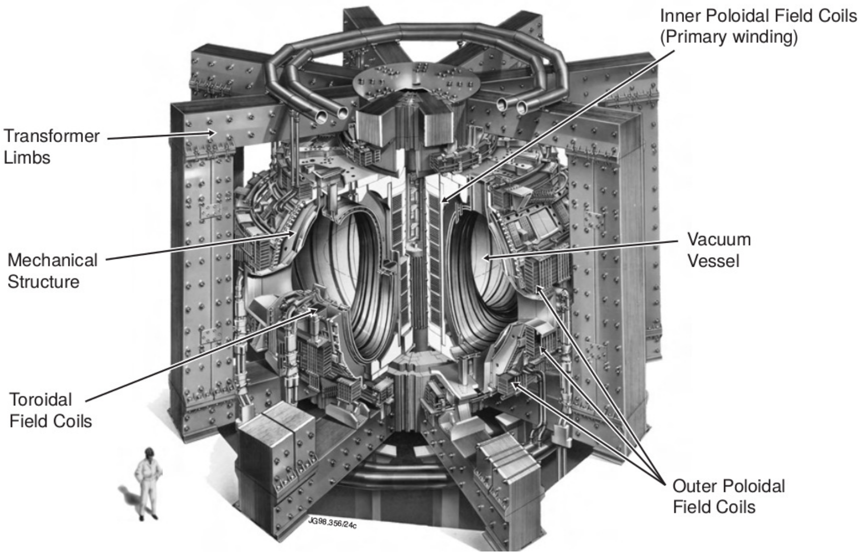


Figure 1.4: Sketch of the JET design [307].

1. Flexible Heliac TJ-II [6]: Located in CIEMAT-Madrid (Spain), the first plasma was launched in 1997. This device is mostly devoted to the study of plasma diagnostics and confinement. A model of the TJ-II stellarator is portrayed in figure 1.5, found in the TJ-II website [198].

2. Wendelstein-7X [224, 35]: As of 2022, this device, located in Greifswald (Germany), is the largest and most advanced stellarator in the world. Wendelstein-7X intends to optimise magnetic confinement for fusion plasmas. Its design is sketched in Figure 1.6, as shown by Pedersen et al. [224]

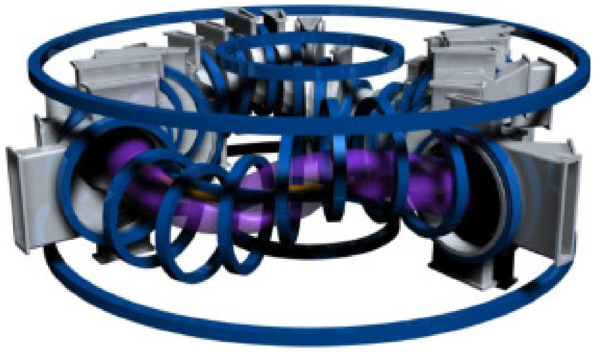


Figure 1.5: Sketch of the stellarator TJ-II [198].

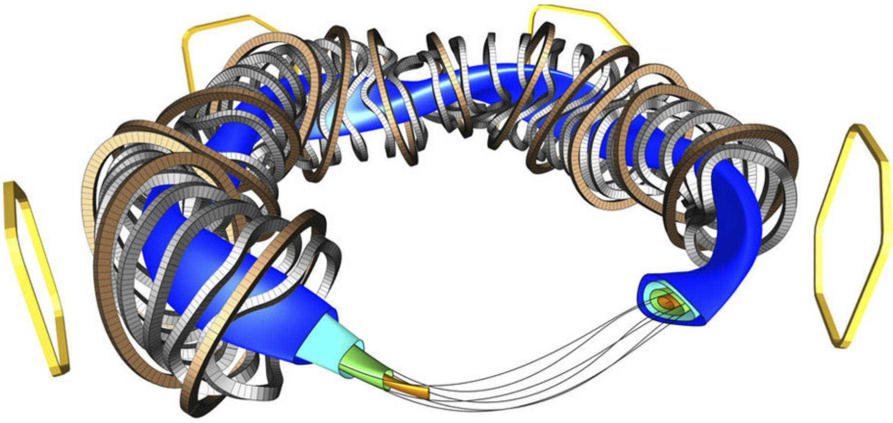


Figure 1.6: Sketch of the stellarator Wendelstein-7X [224].

So far, tokamak physics is better understood than stellarator science. The ambitious project ITER (International Thermonuclear Experimental Reactor) is under construction in Cadarache (France) at the time of writing, and based on the tokamak design. ITER is expected to achieve ignition of a deuterium-tritium plasma and serve as a basis for a future Demonstration Power Station, abbreviated DEMO [89, 99, 30]. These devices are described in the upcoming subsection.

1.1.4.2 Devices for nuclear fusion as a future clean source of energy: ITER and DEMO

ITER is the paramount nuclear fusion project in the world. The ITER project is funded and monitored by seven partners [141], namely China, the European Union, India, Japan, Russia, South Korea, and the USA. ITER is expected to produce a fusion plasma able to sustain itself in long pulses, and eventually achieve steady-state operation. Thus, in the long run, ITER shall provide a physics basis for controlled ignition in a tokamak.

The ITER plasma, first expected in the decade of the 2020s, shall be heated up to a temperature of 1.5×10^8 K by two methods, namely ohmic heating and external heating [141]. Ohmic heating is achieved due to the current induced in the plasma by the poloidal field coils, as the charged particles that respond to this current collide with each other and create heat. However, this heating effect is diminished for higher temperatures, so an additional heating system is needed. Therefore, external heating is provided in two forms to bring the plasma to the desired temperature, namely neutral beam injection (NBI) and high-frequency electromagnetic waves. In NBI, very energetic neutral particles are shot into the plasma, like deuterium atoms [141], which transfer their large kinetic energy to the particles present in the plasma and heat the system as a result. Due to their neutral charge, particles in the neutral beam do not respond to the electric or magnetic fields and do not disturb the plasma. On the other hand, high-frequency electromagnetic waves introduce energy via photons into the plasma. Thus, the kinetic energy of the plasma particles is increased and the plasma is heated.

The ITER project faces serious technological questions, since it is the first device in history for which the feasibility of a self-sufficient plasma by magnetically controlled fusion is under scrutiny. For instance, some components that are necessary for a fusion power plant shall be tested in ITER first, like the materials that shield the magnets from radiation damage or the modules used to breed tritium for the D-T reaction. Furthermore, ITER shall provide the basic data concerning plasma regimes of operation in a commercial fusion power plant, starting by inductive regimes and eventually studying steady-state regimes. Other issues like capital costs, diagnostics and safety features are also part of the investigations that the ITER project must address [89].

DEMO shall be the next step after ITER on the road to nuclear fusion as an energy source. While DEMO is not defined as a commercial fusion power plant per se, but a demonstration one, it aims to produce electricity by fusion and serve as a basis for commercial power plants. DEMO goals include [89, 21, 30]:

1. Guarantee tritium self-sufficiency. The DEMO facility must be able to produce all the tritium needed for the entirety of its operation, without any external input.

2. Achieve net power production. The fusion plasma in DEMO must be self-sustaining, such that more power is extracted from the fusion reactions than is invested in the device.

3. Provide information about technology requirements and availability. Using the experience gained from ITER experiments, DEMO shall test the limits of

materials and plasma technology and their applicability to a future commercial fusion power plant. Indeed, the components used for DEMO should be resilient enough to last for a few years of operation before they are replaced.

Components that are located within the vacuum vessel of a fusion reactor bear the name of in-vessel components. As far as materials physics is concerned, three in-vessel components of ITER, and applicable to DEMO, are of interest to this work, namely the breeding blanket, first wall and divertor.

As we have seen before, the candidate fusion reaction requires tritium to be present in the plasma. However, the amount of tritium available on Earth is limited due to its short half-life (12 years). Therefore, the fusion reactor should be able to produce its own tritium. This is the role of the breeding blanket [42, 48] that, via the presence of lithium, can achieve the production of tritium through the following nuclear reaction:

$$^{6}Li + n \rightarrow {}^{4}He + {}^{3}T \tag{1.3}$$

The necessary neutrons for this reaction are generated by the fusion reaction itself, as shown in Eq. 1.2. Of course, a self-sufficient nuclear fusion device should generate more tritium than it consumes. Therefore, an optimal breeding blanket should capture as many neutrons as possible [21, 42, 48]. For DEMO, the tritium breeding ratio (i.e. the average amount of neutrons produced per neutron consumed) required for self-sufficiency has been defined as 1.1 [42, 48, 99, 103]. To achieve this, a neutron multiplier is usually included in breeding blanket designs. Furthermore, the breeding blanket fulfils the role of neutron shielding, i.e. it protects external elements like magnetic coils from neutron bombardment. As 14 MeV neutrons impact the Li-containing breeding blanket, a large amount of heat is transferred. A coolant is thus needed to ensure safe operation of the breeding blanket, and heating this coolant can be used to generate electricity via turbines.

To protect the breeding blanket from heat, the first wall is in place [21, 99, 190, 195, 298]. This component faces the plasma directly, and is subjected to high particle and heat fluxes. To design a robust first wall, its materials should fulfil the following requirements:

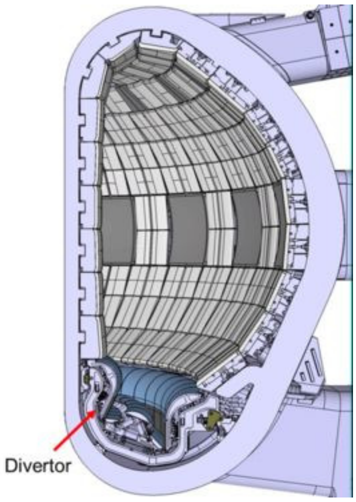


Figure 1.7: Location of the divertor in the vacuum vessel [104].

1. Neutron capture should be kept to a minimum, so that the tritium breeding ratio of the breeding blanket is not disturbed.

2. A high melting point is recommended, due to the large thermal loads faced by the first wall.

3. To avoid problems with radioactivity, neutron activation of the first wall materials should be minimised.

4. The materials should show low erosion yield, in order to prevent wall degradation and plasma contamination.

In case there is an abnormal situation that involves a notable thermal load on the walls, such as a disruption, the first wall must be resistant enough to protect external components.

Finally, the *ash* products must be removed from the vacuum vessel. These include the fusion product He and elements that have been sputtered from the first wall and contaminate the plasma. For this purpose, ITER is equipped with the divertor. Figure 1.7 shows the placement of the divertor in the vacuum vessel, as seen in the Fusion for Energy website [104].

The divertor is a structure that extracts heat and ash produced by the fusion plasma [141]. Thus, divertors help avoid contamination of the plasma and protect other wall components. In case of a plasma disruption, the walls are subjected to high heat fluxes that are orders of magnitude above those in normal operation [89, 141, 195, 298]. The divertor is in place in order to withstand this substantial heat flux, such that damage in the wall structure is avoided. A tokamak divertor consists of a heat sink that is connected to a coolant and protected from the plasma by an armour. In subsection 1.2.1, the considered modules for divertor testing in ITER are listed. The data extracted from ITER divertor operation will afterwards serve as a basis for the DEMO divertor.

As one can see, the materials needed for in-vessel components in future fusion devices face immense engineering challenges. The upcoming section explains the effects of neutron irradiation on these components.

1.2 Neutron irradiation in nuclear fusion devices

In this section, a review of the effects of neutron irradiation in materials relevant for fusion devices is presented. First, in subsection 1.2.1, the main materials used for components in ITER and DEMO are described. More insight to the processes taking place in neutron-irradiated materials is given in subsection 1.2.2. Since materials properties are altered macroscopically as a result [304, 326, 328], subsection 1.2.3 focuses on some of the most important macroscopic effects by which a neutron-irradiated sample might be affected.

1.2.1 Main material candidates for components in nuclear fusion devices

In magnetically-confined nuclear fusion reactors, 14 MeV neutrons are produced, as seen in Eq. 1.2. Therefore, in-vessel components are subjected to a bombardment of these energetic particles [28, 190, 320, 328]. In the case of DEMO, the particle and heat fluxes that materials must sustain exceed all experimental values to date. DEMO materials shall sustain two orders of magnitude of damage above that of ITER materials, as well as a substantially higher flux of gas particles produced in the plasma [196]. For this reason, the materials requirements of DEMO are notably more restrictive than those of ITER.

Operation in ITER since first plasma is planned for about 30 years [28, 89], so materials for the ITER vacuum vessel must be able to withstand the particle and thermal loads for a reasonable lifetime (months to years [161, 196]) before they are replaced. The vacuum vessel armour material needs high crack resistance at very high temperatures and compatibility with plasma-wall interaction effects [241].

For in-vessel components like the first wall, breeding blanket and divertor, materials must be reliable for the whole range of operational temperatures. The main challenge behind the design of in-vessel components is the response of materials to neutron irradiation [28] because the intense neutron flux is known to severely degrade materials properties, as we shall explain in subsection 1.2.3. The heat sink in particular must display good resistance to fracture under high stress, i.e. high fracture toughness. A CuCrZr alloy was chosen for the heat sink of the ITER breeding blanket, as its fracture toughness meets the requirements. In addition, CuCrZr is easily available and inexpensive [28]. Its range of operational temperatures is very limited though, as CuCrZr loses its good mechanical properties under irradiation at moderately high or moderately low temperatures [29, 317], so it is, in principle, not the best option for a DEMO heat sink design.

The ITER first wall is composed of stainless steel 316L(N)-IG with a high-purity Be armour [89, 190]. Be would not be optimal for the requirements of DEMO, however, because of its tendency to tritium retention and transmutation to He, as well as erosion and dust production [115, 328]. Instead, the default DEMO first wall design is a stainless steel (EUROFER-97) structure with W coating [21, 168, 195, 298, 319]. W has a high melting point and features low erosion [1, 190, 240]. Figure 1.8 shows a model of two adjacent cassettes of the DEMO first wall [139]. A W armour protects the EUROFER-97 structure, which is cooled by water flowing in a stainless steel tube with a Cu layer.

There are disadvantages to using W as first wall armour, however. W is a brittle material, which is further embrittled by neutron irradiation [21, 167, 190, 201, 240, 319]. Development of W materials with better ductility under DEMO conditions is a crucial part of DEMO first-wall design. Current research includes the development of W alloys and composites enhanced with W fibres [75, 167, 168, 201].

As far as the breeding blanket is concerned, the chosen primary structural material for ITER is stainless steel 316L(N)-IG. In the case of ITER, six test blan-

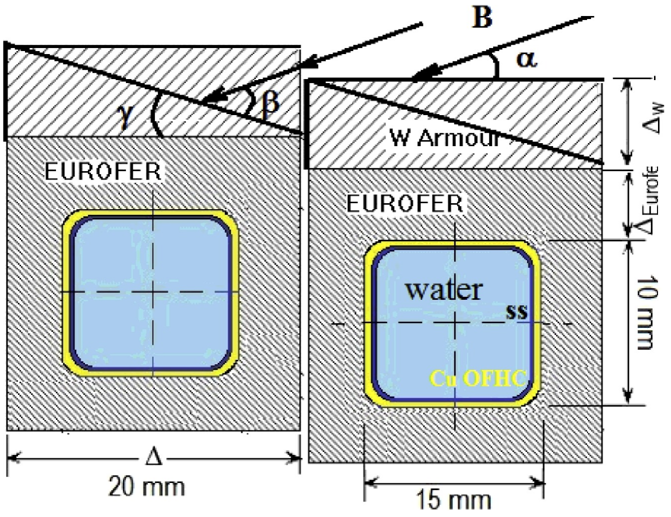


Figure 1.8: Conceptual design of two adjacent cassettes of the DEMO first wall, as shown by Igitkhanov et al [139].

ket modules (TBM) have been planned and are expected to be tested simultaneously [42, 136]:

1. Helium-Cooled Lithium Lead (HCLL) TBM [5], using liquid LiPb as tritium breeder and neutron multiplier, and He as a coolant.

2. Helium-Cooled Pebble Bed (HCPB) TBM [60], using a lithiated ceramic as tritium breeder and Be as neutron multiplier in pebble form, and He as a coolant.

3. Water-Cooled Ceramic Breeder (WCCB) TBM [97], using a lithiated ceramic as tritium breeder in pebble form, and water as a coolant.

4. Dual-Coolant Lithium Lead (DCLL) TBM [266], using liquid LiPb as tritium breeder, neutron multiplier and coolant, and He as a secondary coolant.

5. Water-Cooled Lithium Lead (WCLL) TBM [20], using liquid LiPb as tritium breeder and neutron multiplier, and water as a coolant.

6. Lithium-Lead Ceramic Breeder (LLCB) TBM [156], using LiPb as tritium breeder, neutron multiplier and coolant, and He as a secondary coolant. T breeding is also achieved with a lithiated ceramic in pebble form.

Finally, the divertor is the in-vessel component that is most exposed to high thermal and particle loads. Therefore, its armour structure must be resistant to heat (i.e. have a high melting point) and to sputtering, so that plasma contamination is limited. This is the case of W and carbon fibre composites. However, the latter are prone to erosion and tritium retention, which leaves W as the most viable option and the final choice for the ITER divertor armour [226]. As well as in the case of the breeding blanket heat sink, a CuCrZr alloy is used for the heat sink of the ITER divertor, and steel 316L(N)-IG as a structural material. The DEMO divertor is required to guarantee power extraction capability and reliable for two full power years of operation [318, 319, 320]. In late 2020, the final design for the European DEMO divertor was chosen [318]. In this design, the heat sink is armoured with a 2 mm W coating, and the aforementioned CuCrZr alloy is the selected material for the cooling pipes.

We shall now give more attention to the main structural materials in future fusion reactors, namely structural steels (like stainless steel 316L(N)-IG) and W.

1.2.1.1 Structural steels

The aforementioned austenitic steel 316L(N)-IG is, as we have seen, the main structural material for the ITER vacuum vessel, breeding blanket and divertor [28, 80, 319]. Stainless steel 316L(N)-IG is chosen for its remarkable behaviour like good corrosion resistance, availability and high minimum tensile mechanical properties, which allows for higher stresses than other steels. Its chemical composition takes into account the limits of impurity content (in particular Co and Nb) set by ITER.

Nevertheless, steel 316L(N)-IG does not guarantee safe operation in DEMO. As an austenitic steel, its crystal structure is primarily fcc and therefore prone to swelling. Due to the very high flux of energetic neutrons and the higher than ITER operational temperature that are expected, this steel is not an optimal choice for the DEMO vacuum vessel, as it would be extremely affected by swelling under DEMO conditions [278]. In addition, 316L(N)-IG is a doubtful choice concerning operational safety under the neutron flux of DEMO, due to the high potential for nuclear activation.

The need for structural materials that can resist swelling and nuclear activation has inspired the research of reduced activation ferritic-martensitic (RAFM) steels. Based on a Fe-Cr alloy with about 8% - 10% Cr content, RAFM steels are the main candidates for the structural material of the divertor in ITER and DEMO [40, 57, 161, 276, 278]. Their structure is primarily bcc, and therefore less affected by swelling than austenitic steels. Alloying elements include W and Ta, which are less prone to activation than other typical alloying elements in steels, for example Nb. In addition, impurities that show high activation are minimised in RAFM steels. RAFM steels display good mechanical properties at low temperature, so they are chosen as structural materials for the starting phases of DEMO, which set a low operational temperature of about 620 K [40].

For instance, a very promising RAFM steel for fusion applications is EUROFER-97, the reference steel for nuclear materials in Europe. As mentioned in the previous subsection, the DEMO first wall relies on it as a structural material. In fact, the chosen European DEMO divertor design relies on EUROFER-97 as its structural material [318]. Besides, EUROFER-97 is the main structural material for the European TBM destined for the ITER breeding blanket [161].

Unfortunately, RAFM steels developed to date show worse mechanical properties at high temperature, like resistance to corrosion, than at low temperature. For higher thermal loads expected in later DEMO experiments, alternative structural materials to RAFM steels are better suited. Bearing this in mind, oxide-dispersion strengthened (ODS) steels are being developed and studied [27, 157, 161, 278, 327]. Synthesis of ODS steels involves a steel powder and about 0.3% to 0.5% of a dispersed oxide like Y_2O_3. These oxide particles enhance mechanical properties of this material for operation up to about 950 K [27, 157, 278]. In particular, a class of ODS steels known as nano-structured ODS steels [27, 40] are optimised for good properties at high temperature and resistance to radiation. By being available to operate at high temperatures, ODS steels might be used as a structural material for the DEMO breeding blanket and divertor.

1.2.1.2 Tungsten

W is a strong candidate for the 54 divertors present in ITER [28]. High melting point, thermal conductivity, resistance to sputtering and erosion [114, 115, 168, 170, 230, 240] are among the properties of W that make it attractive to use as a base material for fusion reactor divertors. In general, W or W alloys are usually the main material used in DEMO designs for any regions that are subjected to high thermal fluxes, which is the case of the divertor [115, 168]. There are concerns about the presence of W in a tokamak however. W can melt in case of disruptions or edge-localised modes [54, 230, 240], which are instabilities that appear in tokamak plasmas and are expected in ITER.

Moreover, W is known for its high ductile-to-brittle transition temperature (DBTT), which makes its use as a structural material at low temperature questionable [240, 241]. The DBTT marks the point at which a material loses its ductility and is explained in more detail in subsection 1.2.3.1. Since it is dangerous to operate the fusion device at temperatures below the DBTT, engineering efforts towards W alloys focus on decreasing this value. Several alloys based on W are being explored using Ti, V, Y_2O_3 or Ta as alloying elements, with the hope of enhancing ductility [116, 241].

Another question to keep in mind is transmutation of W into elements like Re or Os [114, 115, 116]. Particularly, large quantities of Os may weaken the structural and mechanical properties of materials [116, 310]. As Os tends to form small precipitates and harden the material [310], this leads to embrittlement, as explained in subsection 1.2.3.1. Although Re is known to increase ductility of W alloys, under the irradiation conditions expected in a fusion reactor like DEMO, the material would be subjected to an unacceptable level of embrittlement [240]. Therefore, the production of these elements under neutron irradiation is unwelcome. Moreover, the production of He by transmutation of the alloying elements, such as Ti or V [240], also leads to degradation of materials properties.

1.2.1.3 Functional materials

Functional materials are those which contribute actively to the sustenance of the fusion plasma, which is the case of the tritium breeder and the neutron multiplier in the breeding blanket, as well as the heating system.

Two of the TBM designs for the ITER and DEMO breeding blanket, namely the HCPB and HCLL TBM, are being developed in Europe [59, 100]. On the one hand, providing the tritium breeder in pebble form, which is the case of the HCPB TBM, is appropriate due to the intrinsic thermal resistance of pebbles [330]. Lithium titanates like Li_2TiO_3 and silicates like Li_4SiO_4 are being considered as breeder materials for the DEMO breeding blanket based on the HCPB TBM [59, 330]. The pebbles must be stable under high heat flux and stress, and activation by neutrons must be reduced as much as possible [330]. In a structure consisting of pebbles, tritium removal is straightforward. Due to the fact that the maximum packing density is about 63% for equally sized pebbles [330], a purge gas may penetrate in the structure and keep tritium retention to a minimum. However, an additional neutron multiplier is needed; in the HCPB TBM, Be is chosen for this purpose. Be is the default neutron multiplier for breeding blanket designs; however, its availability is scarce and new breeding blanket designs aim to reduce the minimal Be requirement for economic reasons, while keeping performance high [263].

On the other hand, a liquid metal breeding blanket, such as the HCLL TBM, is able to provide safe operation even at high operational temperature and is easy to maintain [61]. In addition, the liquid metal LiPb, which contains enriched lithium [42, 19, 100], acts as a neutron multiplier itself, and eliminates the need of expensive Be. The main problems behind a liquid metal breeding blanket are magnetohydrodynamic instabilities, corrosion of the structural material and tritium permeation, which can be mitigated by the use of special coatings [61]. Indeed, corrosion of the structural material, which is a function of its compatibility with the liquid functional material, may become a serious concern as the structure is degraded and the liquid metal is contaminated by impurities and activated corrosion products [19, 42, 61, 100]. In turn, the materials used for the coatings should minimise activation by neutrons. Concerning the HCLL TBM, the liquid metal is the eutectic Pb-15.8%Li alloy (i.e. 84.2% Pb, 15.8% Li) [42, 100]. This alloy is the reference for all liquid metal breeding blanket designs, and is chosen for its low tritium retention and good response to safety concerns and irradiation damage [5, 19, 20, 42, 61, 100].

Moreover, many of the plasma diagnostics in ITER and DEMO will be conducted by means of the analysis of optical radiation [181]. For this reason, optical components like fibres and diagnostic windows are needed. The materials used for this purpose must offer good optical transmission and electrical insulation [84]. Since they are subjected to an intense flux of neutrons and heat, good mechanical properties are also desired [61, 84]. SiO_2 is the main candidate for transmission components [61, 294] due to its low intensity radiation-induced absorption and luminiscence, compared to other materials. In particular, the fused silicas KU1 and KS-4V are prime candidate materials for optical components due to their mechanical strength [294].

For the heating system by microwaves and NBI, a material that guarantees electrical insulation is necessary. Insulators may become conductive under irradiation in a process known as radiation-induced conductivity (RIC) [135, 294]. RIC is an increase in conductivity as a result of the excitement of valence electrons to the conduction band [84, 135, 294]. Furthermore, radiation-induced electrical degradation (RIED) refers to a permanent increase in volume electrical conductivity caused by defects in an electric field [84, 294]. The physics behind this process is not yet fully understood. Therefore, a material that resists RIC and RIED, as well as surface effects like sputtering or impurity segregation, is necessary for the components of the heating system. The foremost candidate for this purpose is Al_2O_3 [135, 159, 261, 294]. This material offers good resistance to electrical degradation in the forms of RIC, RIED and surface degradation. Research of different radiofrequency window designs with Al_2O_3 as a functional material is underway [135, 159].

Having reviewed the most important structural and functional materials for a nuclear fusion device like ITER or DEMO, we shall now explain the physics behind defect generation in neutron-irradiated materials.

1.2.2 Defect generation in materials under neutron irradiation

Interaction of incident particles such as neutrons with crystal lattices produces microscopic alterations in the lattice, known as defects. This process is called the radiation damage event. In this subsection, we discuss different types of defects that can arise as a result of neutron damage in materials.

1.2.2.1 Interstitials and vacancies

When an incident particle collides elastically with a lattice atom, the latter can be removed from its original position and unable to return there. An atom that has been displaced to an equilibrium position away from the lattice node is called a self-interstitial atom (SIA). Analogously, a lattice node where no crystal atom is present is labelled a *vacancy*. When an atom is displaced from its corresponding lattice node and placed in an interstitial position, the combination of this SIA and the new vacancy is known as a Frenkel pair [120, 304]. Figure 1.9 portrays this defect type. On the left, a sc crystal cell is sketched. On the right, the situation is depicted after an atom is removed from its node and displaced to an interstitial position, i.e. when a Frenkel pair is created.

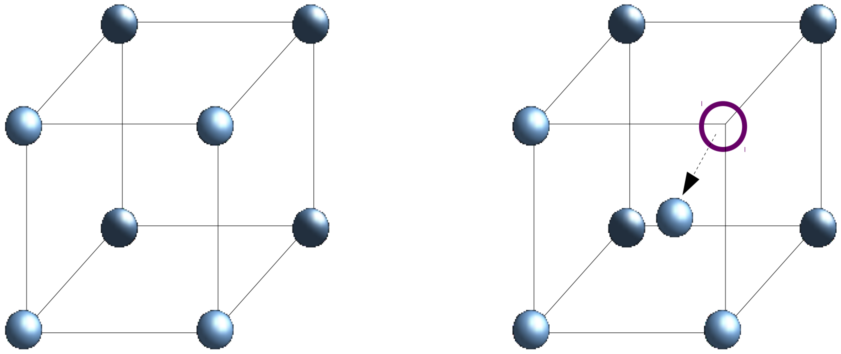


Figure 1.9: Left: sc cell without defects. Right: a Frenkel pair is present in the cell.

In materials with a cubic crystalline structure, SIAs tend to have a high formation energy and a small migration energy, whereas vacancies have low formation energy and high migration energy [304]. SIAs show, in general, a greater mobility than vacancies. For instance, the migration energy of SIAs in α-Fe has been reported as 0.3 eV to 0.35 eV whereas the migration energy of vacancies has been found to be between 0.55 eV and 0.7 eV [31, 105, 174, 187, 275, 285, 282, 303]. Similarly, greater mobility of SIAs over vacancies has been reported in other materials such as W [12, 13, 130, 147, 303] and Cu [117, 148, 262]. At a certain temperature, SIAs become mobile and can diffuse in the material. Hence, they can meet other defects to interact with, like vacancies. Interaction of a SIA and a vacancy typically results in annihilation [210, 304]. Moreover, some features in the microstructure, like grain boundaries, dislocation lines or impurity atoms, act as sinks for point defects. In other words, sinks can attract SIAs or vacancies and immobilise them.

When defects of the same type interact with each other, they are able to form clusters [304]. Interstitial clusters are particularly stable due to the high energy (in the order of a few eV) necessary to dissociate a SIA from the cluster. On the other hand, vacancies can nucleate into vacancy clusters, which are thermally less stable than SIA clusters [194] and can eventually grow into voids. In crystalline materials, voids are three-dimensional imperfections in the lattice which contain no atoms. Voids can grow by subsequent agglomeration of vacancies and vacancy clusters [210, 304]. Vacancy clusters can shrink by recombination with SIAs or grow by the capture of more vacancies. Mechanisms that trap SIAs, such as sinks, contribute to the creation of voids. The presence of such mechanisms allows a net surplus of vacancies which can then nucleate and grow into a void. In particular, dislocations have been shown to trap interstitials preferentially, resulting in a flow of vacancies to voids [90, 169, 179, 210, 304].

1.2.2.2 Displacement cascades

The first atom displaced by the incident particle is labelled primary knock-on atom or PKA. If radiation is sufficiently energetic, the PKA transfers its excess energy to surrounding atoms that proceed to vibrate and eventually reach interstitial positions. In other words, energetic radiation can produce movements of atoms that afterwards set in motion a series of atomic displacements, the *collision cascade* or *displacement cascade*, before equilibrium is reached. The initial phase of neutron-matter energy transfer is known as the *collisional* phase, whereas the subsequent phase until thermalisation is known as the *relaxation* phase [50, 299, 304].

In dense materials such as Fe, Cu or W [83, 249, 250, 299, 303], energy is transferred quickly between the displaced atoms and their neighbours in a small region. This is commonly known as a *thermal spike* [50, 83, 249, 250, 299, 304]. This is a very hot region above the melting point of the material, in the order of thousands of K. As a result, when the thermal spike is produced, the heated region is similar to molten material. In general, a region with a depleted central core, i.e. a region dense in vacancies, is formed with an outer shell of interstitial atoms [50, 83, 304]. Then, heat is transferred to surrounding atoms such that the molten region reaches a temperature lower than the melting point and returns to the state of condensed matter.

A simulation of this phenomenon, studied by Sand et al. [249, 250], is portrayed in Figure 1.10. This simulation shows the effect of a 150 keV cascade on a W crystal, showing interstitials in blue and vacancies in yellow colour. The snapshot on top shows the thermal spike, where the depleted core and surrounding interstitial shell are distinguishable. On the other hand, the relaxation phase results in the creation of new defects, as depicted in the snapshot below.

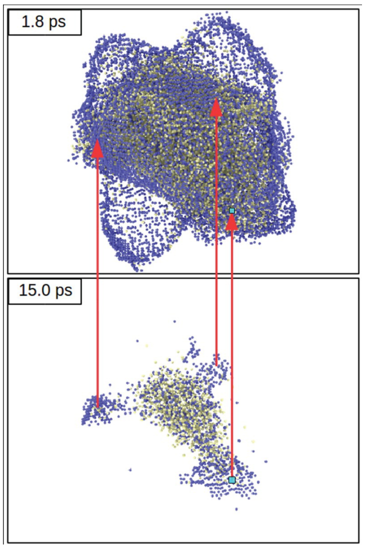


Figure 1.10: Top: Simulation of a heat spike, as depicted by Sand et al. [249, 250]. Bottom: formation of defects in the relaxation phase. Blue spheres depict interstitials, yellow spheres depict vacancies.

SIAs in the shell can travel back to the depleted core as a pressure wave and recombine with the vacancies. Even though some stable defects and defect clusters may be produced, most of the formerly created Frenkel pairs are annihilated [304]. If many heat spikes are present in the material such that a sizable number of stable defects are available, which are able to produce microstructural damage in the material, macrostructural properties can be affected. Figure 1.11 depicts the number of defects produced in a collision cascade as a function of time, as simulated by Vörtler et al. [299] for an irradiation of Fe-10%Cr with a 20 keV PKA. As one can see, the three phases of the collision cascade are clearly distinguishable. The number of defects produced in the collisional phase grows until the thermal spike is achieved, and then cascade relaxation reduces the number of defects until it reaches a stable value.

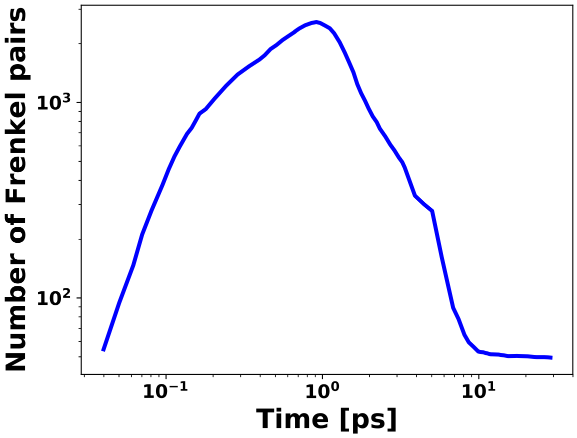


Figure 1.11: Evolution of the number of defects in a 20 keV cascade on Fe-10%Cr in time, as simulated by Vörtler et al. [299]

Once the process of displacement collision is finished, the remaining defects in the system are able to evolve in time and interact with the material structure. Indeed, the diffusion, clustering and annihilation of defects along a time line, also known as the annealing process [161, 241, 281, 304, 322], are crucial to understand the effects of irradiation on material properties in the long run.

1.2.2.3 Dislocation lines and dislocation loops

Distortions in the periodicity of the crystal structure are known as dislocations. The line that forms the boundary between both regions is the dislocation line. There are two main types of dislocations that can be distinguished, i.e. the edge and screw dislocations. In an edge dislocation, this defect can be described as an extra half-plane of atoms, whereas in a screw dislocation, the material looks like a spiral ramp when viewed from a plane perpendicular to the dislocation line [304]. For further clarity, edge and screw dislocations along with their dislocation lines are depicted in Figure 1.12, as found in G. S. Was' book *Fundamentals of Radiation Materials Science* [304]. A third type of dislocations, the mixed dislocations, are a combination of edge and screw dislocations. Dislocations are characterised by the Burgers vector $\vec{b}$, which indicates the difference between the two regions separated by the dislocation line.

Dislocations are affected by stress. In the case of an edge dislocation, atoms are

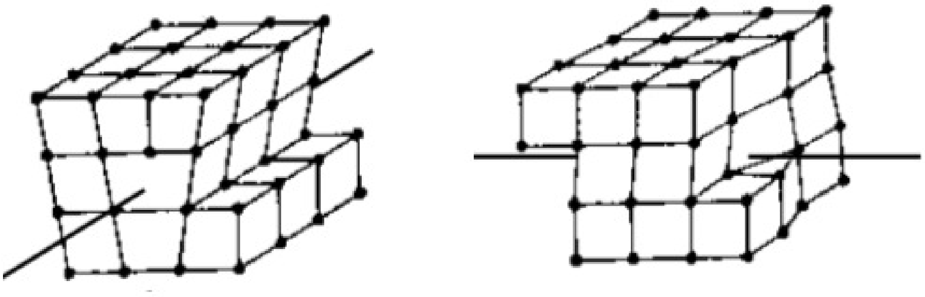


Figure 1.12: Dislocation types, as sketched by Was [304]. Left: edge dislocation. Right: screw dislocation.

shifted in a perpendicular direction to the dislocation line, which moves parallel to the applied stress. In screw dislocations, the atoms are shifted in a parallel direction to the dislocation line, such that the latter moves perpendicular to applied stress. If a dislocation lines moves in a direction that is not perpendicular or parallel to the applied stress, it is a mixed dislocation.

Dislocations are largely responsible for plastic deformation in crystalline materials [210, 218, 304]. At the atomic level, plastic deformation happens when crystal atoms slip on the plane where dislocations move, called the slip plane. The combined effect of many dislocation movements along slip planes gives rise to the macroscopic plastic deformation [210, 304]. Hence, the mobility of dislocations has a direct effect on the ductility of materials, a concept that will be detailed in subsection 1.2.3.1.

The dislocation line can also be a closed loop, labelled dislocation loop (DL). Dislocation loops are of great importance to this work, as they often appear in irradiated structural materials for nuclear fusion [234]. They can appear by clustering of interstitials or vacancies, or they can form in energetic irradiation cascades. Energy minimisation causes defect clusters to acquire a platelet form which gives rise to the DL [192, 304, 328]. A visual example is given in Figure 1.13. As simulated by Osetsky et al. [218], three SIA clusters react to form a DL in a 20 keV cascade in Cu. In this figure, dark spheres and light spheres represent SIAs and vacancies respectively.

DLs migrate very rapidly and are significantly involved in processes that give rise to effects like swelling, as we will see in subsection 1.2.3.

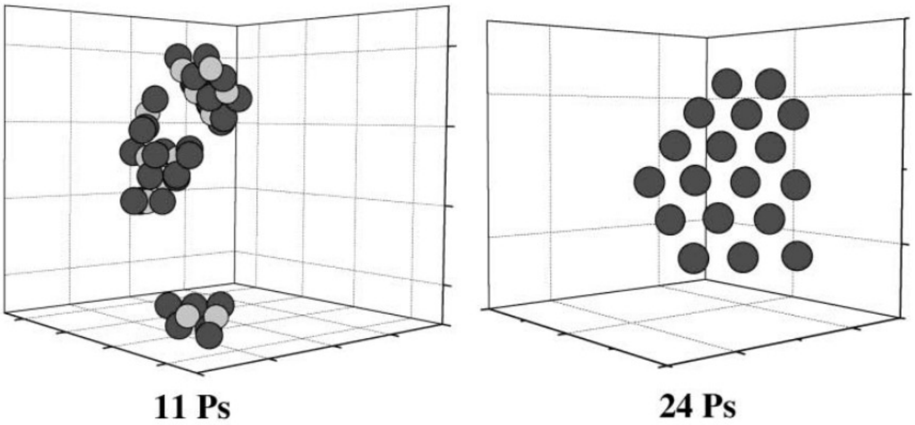


Figure 1.13: A dislocation loop is formed as a reaction of three SIA clusters [218].

1.2.2.4 Helium impurities and bubbles

Transmutation reactions occur in materials under neutron bombardment [115, 114, 106]. In the case of nuclear fusion, the shower of 14 MeV neutrons on metals produces large quantities of H and He. Eq. 1.4 describes the transmutation of ^{56}Fe under neutron irradiation [115].

$$\begin{aligned} &^{56}Fe + n \rightarrow {}^{53}Cr + {}^{4}He \\ &^{56}Fe + n \rightarrow {}^{56}Mn + {}^{1}H \end{aligned} \tag{1.4}$$

He is known to form complexes with point defects like SIAs and vacancies [108, 254]. Particularly, vacancy clusters act as powerful traps for He, which eventually leads to the formation of stable He$_n$V$_m$ clusters. Thus, when He impurities are present, void nucleation is enhanced [192, 193]. Therefore, the presence of He prevents recombination of Frenkel pairs, as vacancies group into He$_n$V$_m$ clusters and less single vacancies are available for SIAs to recombine with. This leads to the enhancement of SIA clustering as well, which induces the creation of DLs. This is why both void and DL formation are increased in the presence of He.

He is also notable for growing into bubbles [106, 115, 193, 194, 260]. When the size of a void exceeds a limit, the void becomes spherical and is termed a bubble [304]. Bubbles nucleate under irradiation. Then, they can grow by absorption of more He and vacancies, or shrink by emission of He, emission of vacancies and absorption of SIAs [193, 194, 260]. The He impurities expelled from $\mathrm{He}_n\mathrm{V}_m$ clusters may act as new nucleation centres for other bubbles. In addition, when gas pressure is very high, such as in the case of plasma-facing materials in fusion reactors [193], He bubbles are able to induce a stress in the neighbouring crystal. In a mechanism labelled dislocation loop punching, SIAs or SIA clusters are emitted as a result, and the bubble grows.

1.2.3 Effects of neutron irradiation on main materials used in nuclear fusion devices

We shall now review the most notable effects of neutron irradiation on materials in fusion reactors, namely hardening, embrittlement and swelling.

1.2.3.1 Hardening and embrittlement

When a stress load is applied to a material, deformation occurs. If the sample changes back to its original shape once the load is removed, this phenomenon is labelled elastic deformation. However, if the change in shape is permanent, it is called plastic deformation. The resistance of a material to plastic deformation is understood as the hardness of this material. If a process makes a material more difficult to deform plastically, the material is said to have been hardened. In general, materials are able to sustain a stress load and show elastic behaviour up to a limit called the elastic limit. When the load is increased, plastic deformation occurs. The stress defining this limit is called the yield stress. Thus, an increase in the yield stress of a material is an indication of hardening.

It has been observed that the motion of dislocations explains hardness in materials. If dislocations are able to move easily, the material is prone to plastic deformation, whereas obstacles to dislocation motion counter plastic deformation and harden the material. In fact, hardening represents the increase in stress required to start a dislocation moving on its glide plane [304]. Neutron irradiation is able to introduce obstacles to dislocation motion in the form of defect clusters like voids, DLs, bubbles or impurities. For this reason, neutron irradiation usually hardens the irradiated material.

The measure of engineering stress as a function of strain is helpful for the characterisation of materials properties. Figure 1.14 portrays a stress-strain curve produced by Dai et al. [79] for two T91 steel samples. One of these (blue curve) has been irradiated with 34 MeV ^{3}He particles and one has not (black curve). In this figure, the concepts of elastic and plastic deformation, yield stress and point of fracture are exemplified for the curve that corresponds to the unirradiated sample. As one can see, the material is hardened by irradiation, as a higher stress is necessary to change the shape of the sample after it has been irradiated.

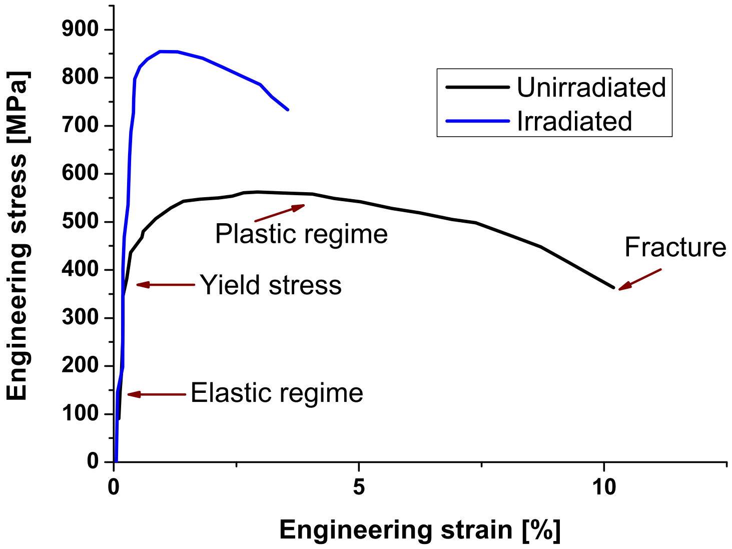


Figure 1.14: Stress-strain curves for two samples of T91 steel, one of which has been irradiated with ^{3}He (blue) and one which has not (black) [79].

However, an associated effect to hardening is embrittlement, which is certainly unwelcome. To understand the concept of embrittlement, the concept of ductility must be first introduced. A material that can withstand a large strain and sustain enough plastic deformation before fracture happens is known as *ductile*, so this property bears the name of *ductility*. The opposite, i.e. a material that is unable to show good ductility and breaks easily before plastic deformation happens, is defined as a *brittle* material. Changes in microstructure that lead a ductile material to become brittle are characterised as *embrittlement*. Embrittlement is an undesirable condition, since plastic deformation is less damaging than material fracture. In Figure 1.14, it is shown that fracture occurs in the irradiated sample at a lower strain than in the unirradiated sample. This is the result of embrittlement.

An ideal nuclear material would be simultaneously hard and ductile, but the processes leading to the hardening of a material are unfortunately correlated to its embrittlement. This is due to the fact that obstacles to dislocation motion lead to embrittlement of the material, as well as being responsible for hardening [304, 326]. Indeed, when dislocations find an obstacle to their motion, they tend to pile up at the location of that obstacle. This agglomeration of dislocations creates a small separation of crystal planes, called a crack, which can grow and propagate, ultimately leading to fracture.

Dislocations, like other defects, move more easily at high temperatures where defect diffusion is likely. Therefore, the ductility or brittleness of a material depends on temperature. If temperature in a ductile material is decreased until a point when dislocations no longer move easily, the material properties change from ductile to brittle. The temperature that characterises this point is known as the DBTT, and sets a lower boundary for operational temperature in nuclear materials [326, 328]. In fact, the embrittlement of a material via external processes like irradiation can be described as an increase in DBTT and a decrease in fracture toughness. In other words, the more serious embrittlement becomes in a material, the higher operational temperature is required, as operation below the DBTT is not adequate. The effect of irradiation hardening and embrittlement is most pronounced at low temperature, below 0.3 T_m, where T_m is the melting point [326].

In order to study the DBTT of a material sample, Charpy impact tests are useful. In a Charpy impact test, a pendulum weight dropped from a certain height collides with a notched sample, such that the energy absorbed by the sample prior to fracture can be determined. This energy is then plotted against the temperature. Figure 1.15 shows a Charpy impact test conducted by Dai et al. [79] for two samples of T91 steel. Again, the results for the sample which has been irradiated with 34 MeV ^{3}He particles to 265 appm (atomic parts per million) is depicted in blue and those for the unirradiated sample in black. Clearly, the unirradiated sample is able to absorb more impact energy at lower temperature before fracture occurs, which shows that the DBTT is increased as a consequence of irradiation.

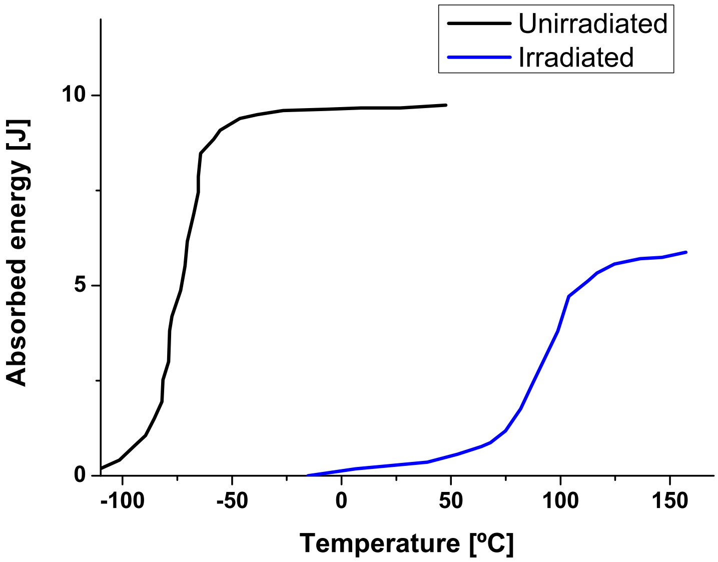


Figure 1.15: Charpy impact test for two samples of T91 steel, one of which has been irradiated with ^{3}He (blue) and one which has not (black) [79].

The presence of DLs is known to harden irradiated materials [17, 180, 308]. It has been found that DLs are sensitive to stress fields induced by dislocations [304, 308]. DLs move in this elastic field and decorate the dislocations, thus inhibiting their motion. As a result, yield strength is increased and fracture toughness is reduced. This leads to an increase in the DBTT, i.e. embrittlement of the material [304].

At high temperature, above 0.5 T_m, embrittlement can also be the effect of neutron irradiation without associated hardening [304, 328]. For instance, voids and bubbles can nucleate and grow at the grain boundary, which creates instabilities between grains. Figure 1.16 shows an underfocus TEM image of He bubbles (white), as obtained by El-Atwani et al. [95], in the matrix of fine grains of a Fe sample irradiated with 10 keV He^{+} ions.

When stress is applied, cracks may propagate between grains as a result. He impurities, which are very mobile, may quickly migrate to the grain boundary and help the growth of intergranular voids and bubbles, decreasing the strength of grain boundaries and further embrittling the material. He migration to grain boundaries sets an upper boundary for operational temperature of the fusion reactor. To mit-

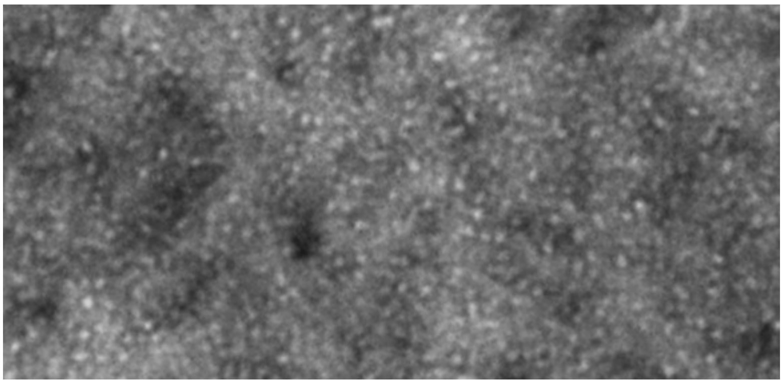


Figure 1.16: He bubbles (white) in a He-irradiated Fe sample [95].

igate this effect, materials under development like nano-structured ODS steels are able to trap He bubbles, thus inhibiting embrittlement at high T [27, 40, 157].

1.2.3.2 Radiation-induced swelling

One of the most catastrophic effects of neutron irradiation on materials is dimensional instability, under which a process known as swelling is classified. Under swelling, we understand the substantial increase in volume of the material sample under irradiation [106, 115, 155]. This is problematic for components in nuclear reactors, as the original sizes and shapes they were designed with are not preserved. Swollen materials are sensitive to gradients in dose, dose rate and temperatures, which can lead to strong distortions and eventually fracture [274].

It is generally accepted that ferritic-martensitic steels suffer less swelling than austenitic steels [111, 155, 328]. Zinkle and Was [329] compared the swelling behaviour of austenitic stainless steels 304L and 316 to that of FeCr ferritic-martensitic steels under neutron irradiation in a fast fission reactor. The swelling rates of austenitic steels were reported an order of magnitude higher than that of ferritic-martensitic steels at doses up to 100 displacements per atom (dpa). Similarly, Garner et al. [111] studied swelling in austenitic and ferritic-martensitic steels at higher neutron doses, up to 2.5×10^{23} cm^{-2} or 150 dpa. The steady-state swelling rates of austenitic steels were reported higher than those of ferritic-martensitic steels by a factor of 4× - 5×. Under DEMO conditions, swelling is expected to reach 5% in RAFM steels at a neutron dose of 10^{23} cm^{-2} [155, 276, 328, 327].

Voids and gas bubbles are responsible for swelling. As vacancies nucleate to form voids, the atoms are displaced outwards. These voids can then develop into small cavities, and as a result, the total volume is increased. Swelling usually occurs at intermediate temperatures, between 0.3 T_m and 0.6 T_m [169, 304, 326, 329]. This is explained by the fact that at lower temperatures, mobility of vacancies is very low, so it is difficult for them to nucleate into voids, whereas at higher temperatures, voids are likely to evaporate by thermal emission of vacancies [49, 155, 169, 274, 304].

In the case of nuclear fusion, the reactor's first wall is subjected to the formation of bubbles, as H and He are present in the plasma, in addition to being released as transmutation products. At high temperatures, He impurities tend to migrate to grain boundaries [255, 328] and form He bubbles [106], as well as contribute to the formation of $\mathrm{He}_n\mathrm{V}_m$ complexes, which was mentioned in subsection 1.2.2.4. Thus, by providing help for the formation of voids and bubbles, gas particles are important contributors to radiation-induced swelling. In fact, it has been proposed that the presence of gas particles is necessary for void nucleation and therefore swelling [304]. Swollen samples display high sensitivity to gradients in dose, dose rate and temperature, which leads to important instabilities and inhibits the good performance of materials. Figure 1.17 shows the difference between an unirradiated sample (left) and a sample that has been affected by swelling (right), from an article by Straalsund et al. [271]

Swelling also leads to embrittlement at high temperature [288]. Void swelling at high temperature can be suppressed by introducing defect sinks in the material that act as recombination centres for interstitial and vacancy clusters. In particular, ODS steels show satisfactory swelling behaviour at high T, as oxide particles prove to be strong sinks for migrating defects [27, 327, 328].

1.3 Motivation and objectives of our work

We are mainly interested in the microstructural processes taking place at irradiated samples under conditions similar to a fusion reactor like DEMO. The intense neutron fluxes that are expected for devices like ITER and, in a more pronounced way, DEMO, require neutron irradiation facilities that are either not readily accessible, or even nonexistent [196, 328].

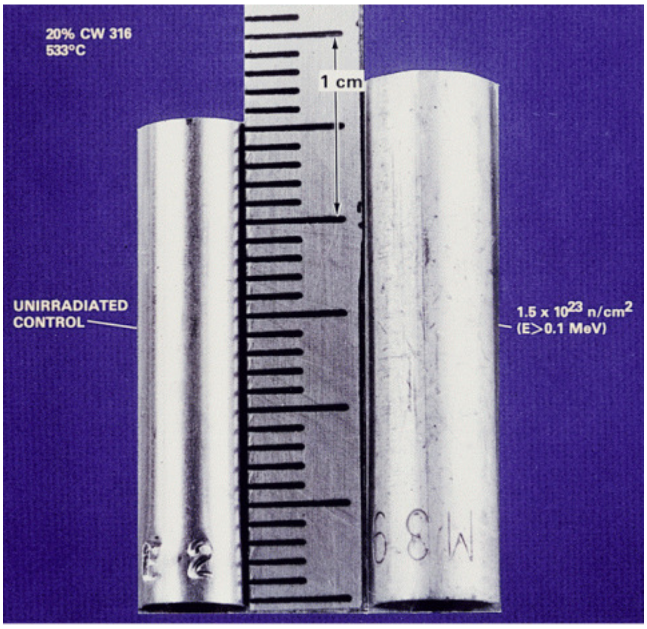


Figure 1.17: Effect of radiation-induced swelling. Left: unirradiated sample. Right: swollen sample [271].

However, it is necessary to predict the behaviour of structural materials under development for fusion reactors. For this reason, accurate and powerful simulation tools are needed. In addition, the physics behind some atomistic mechanisms taking place in the material is uncertain in many cases, and experimental access to clarify these questions is not straightforward. For example, the quantitative effect of gas production (in particular He) via the D-T reaction on structural materials is still an open question [328].

Simulation techniques try to provide an answer for situations where experimental study is difficult, and explain physical developments at space and time scales that cannot be observed directly in the laboratory. For example, a phenomenon like void growth can be quantified by monitoring sink strength, which explains void swelling as observed experimentally by transmission electron microscopy (TEM) [49, 144, 177, 291]. Similarly, the physical mechanisms that explain the stages observed in resistivity recovery (RR) experiments [121, 122, 187, 275] can be interpreted with the help of simulations [82, 105, 151, 285, 204].

The objective of this work is to develop a simulation model able to simulate the evolution of microstructural defects accurately. This model must be able to perform simulations of realistic pieces of material (in the order of grain size) in achievable runtimes, and the predicted results must be in agreement with physical experiments. For this work, the KMC method was chosen as a simulation tool, in particular the subset of Object Kinetic Monte Carlo (OKMC). A new algorithm, called the parallel Tau-leaping algorithm (PTLA), which is optimised for parallel programming, was created for this purpose. Then, an OKMC code based on this algorithm was developed and implemented on graphics cards, also known as graphics processing units (GPUs), called Microstructure Evolution GPU-based Accelerated Object Kinetic Monte Carlo (MEGA-OKMC). The CUDA programming framework was chosen for this purpose, and the open-source Nvidia®C Compiler was used to compile and debug our code [76]. As far as the methodology is concerned, an object-oriented programming approach was selected. Individual properties of single defects such as position or numeric identifier are stored in one class, and common properties to defects of the same type are looked up in an instance of another class. This approach enhances performance from a practical point of view, and helps encapsulation from a programming point of view. The interface of our code was designed with the goal of versatility in mind, such that the user is able to design objects with arbitrary geometries and properties as well as write their own interaction functions. Basic programs were tested for validation against examples found in literature, whereas the validation of more complex programs required coordination with other simulation groups in the fusion community [25, 176].

Different examples are provided to demonstrate the viability of our work as a simulation tool for nuclear materials under irradiation conditions relevant to ITER and DEMO. In this work, we tried to explain the presence of DLs in electron-irradiated Fe as observed by Arakawa et al. using TEM [15] and in self-irradiated Fe observed by Yao et al. [315], using TEM as well. In the future, we expect the MEGA-OKMC model to simulate the trapping of DL by dislocation lines, a phenomenon that has been suggested to induce hardening in steels [86, 91, 223, 279, 290].

This doctoral dissertation is organised as follows. In chapter 2, different simulation techniques for nuclear materials are described, namely MD, KMC and MFRT, and our choice for an OKMC model is justified. Chapter 3 offers a description of general programming on the GPU, which is the chosen implementation of our model.

Afterwards, the main algorithm used in this work, i.e. the Parallel Tau-leaping Algorithm (PTLA), is detailed in chapter 4. Chapter 5 introduces the acceleration method known as Green's Function Reaction Dynamics (GFRD), and how it has been adapted in this work, particularly in the presence of external irradiation. Chapter 6 is devoted to the implementation and interoperability of PTLA and GFRD in our code. Our model is applied to the study of DL evolution in irradiated Fe and Fe-5%Cr in chapter 7, where the benchmarking of our code with OKMC codes produced in similar research groups is laid out as well. Finally, the conclusions of our work are laid out in chapter 8.

Chapter 2

Simulation of defect evolution in irradiated materials

Radiation damage is intrinsically a multiscale problem. From the very small length and time scales related to the creation of the PKA in a cascade, in the order of nm and 10^{-12} s respectively, to the macroscopic sizes (several m) of fusion reactor components and the time scales (a few years) they are designed for, different simulation tools have been developed and must be used to simulate the different phases of radiation damage. The main simulation techniques developed for defect evolution in irradiated materials are Molecular Dynamics (MD), the Binary Collision Approximation (BCA), the Kinetic Monte Carlo (KMC) method and the Mean Field Rate Theory (MFRT). We shall now proceed to explain each of them.

2.1 Molecular Dynamics

MD is a powerful tool to describe the processes taking place in collision cascades. MD simulations follow a classical approach to reproduce the dynamics of the whole particle system. In other words, the Newtonian equations of motion are solved for each of the atoms in the simulation box. The positions and velocities of all atoms in the system are tracked, and this information can be linked via statistical mechanics to macroscopic properties like temperature, pressure or energy [173, 304, 324].

If m_i is the mass of a particle i (like an atom), $\vec{x_i}$ its position vector and $\vec{F_i}$ the forces acting on the particle, Newton's second law of motion is written as:

$$m_i \frac{d^2 \vec{x_i}}{dt^2} = \vec{F_i} \tag{2.1}$$

In a force field, forces can also be expressed as gradients of differentiable potentials U:

$$\vec{F_i} = -\vec{\nabla}_i U \tag{2.2}$$

This yields:

$$m_i \frac{d^2 \vec{x_i}}{dt^2} = -\vec{\nabla}_i U \tag{2.3}$$

If $u_{ij}(\vec{x_i}, \vec{x_j})$ describes the potential between particles i and j due to a pairwise interaction, in the absence of external forces, the system potential U can be expressed as the sum of all pairwise potentials:

$$U = \sum_{i}^{n} \sum_{j}^{i-1} u_{ij}(\vec{x_i}, \vec{x_j}) \tag{2.4}$$

where n is the total number of particles.

Therefore, in general, $U = U(\vec{x_1}, \vec{x_2}, \ldots, \vec{x_n})$ is a function of all atomic positions in the system. Finding a potential that will reproduce the dynamics of the system and being able to integrate the equations of motion numerically is the goal of MD models. In a molecular system, forces are exerted on particles by other particles in their surroundings, in the form of Van der Waals and Coulomb forces. Van der Waals interactions are commonly represented by the Lennard-Jones potential as a function of interatomic distance r [10, 296]:

$$U(r) = 4\epsilon \left[\left(\frac{\sigma}{r}\right)^{12} - \left(\frac{\sigma}{r}\right)^{6} \right] \tag{2.5}$$

Here, σ and ϵ are two constants respectively called collision diameter and potential well depth. The Lennard-Jones potential includes two terms, a repulsive one (proportional to r^{-12}) and an attractive one (proportional to r^{-6}). For short distances, the repulsive term dominates, which is related to Pauli's exclusion principle, as two atoms that are too close to each other tend to separate. However, as

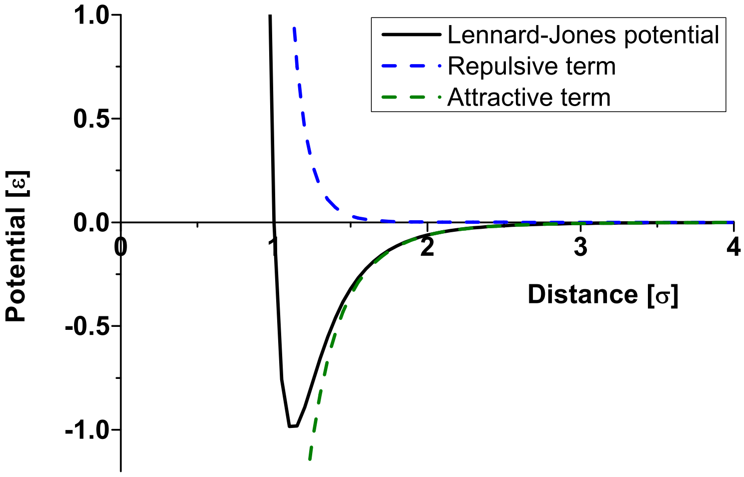


Figure 2.1: Lennard-Jones potential.

separation increases, the attractive term prevails. Interactions between dipoles or induced dipoles, giving stability to the system, are described by this term.

Figure 2.1 shows the Lennard-Jones potential as a function of the distance, in ϵ and σ units respectively, compared to the pure attractive and repulsive terms in Eq. 2.5. As one can see, the function has a minimum at $U(\frac{r}{\sigma}) = -\epsilon$, whereas the collision diameter σ is the distance at which the potential equals zero, as both attractive and repulsive terms cancel each other out.

As far as electrostatic interactions are concerned, the Coulomb potential describes the potential between two charged point-like particles with charges q_1 and q_2, separated by a distance r. If ϵ_0 is the dielectric constant in vacuum and ϵ is the permittivity of the material, the Coulomb potential is written as follows:

$$U(r) = \frac{q_1 q_2}{4\pi\epsilon_0\epsilon} \cdot \frac{1}{r} \tag{2.6}$$

The Coulomb potential is repulsive for particles with equally signed charges and attractive for the interaction between a positively charged and a negatively charged particle. However, most particle systems are too complex to be described by the Lennard-Jones and Coulomb potential only. Usually, for the description of the system potential, ferromagnetic, spin or other terms are included as well [9, 85, 164, 171, 197, 237].

Once the system potential has been defined, the set of coupled equations Eq. 2.3 must be integrated in time. As this equation system is usually too complicated to solve analytically, numerical integration is necessary, which is done by time steps of size Δt. If Δt is small and the positions $\vec{x}(t)$, velocities $\vec{v}(t)$ and accelerations $\vec{a}(t)$ of particles are known at time t, then these values at $t + \Delta t$ can be estimated with a Taylor expansion. The velocity $\vec{v}(t) = \frac{d\vec{x}(t)}{dt}$ and acceleration $\vec{a}(t) = \frac{d^2\vec{x}(t)}{dt^2}$ are the first and second time derivatives of the position, respectively. If $\vec{b}(t)$ is the time derivative of the acceleration with respect to time, i.e. the third time derivative of the position, the Taylor expansion yields:

$$
\begin{aligned}
\vec{x}(t+\Delta t) &= \vec{x}(t) + \vec{v}(t)\cdot\Delta t + \frac{1}{2}\vec{a}(t)\cdot(\Delta t)^2 + \frac{1}{6}\vec{b}(t)\cdot(\Delta t)^3 + O\left((\Delta t)^4\right) \\
\vec{v}(t+\Delta t) &= \vec{v}(t) + \vec{a}(t)\cdot\Delta t + \frac{1}{2}\vec{b}(t)\cdot(\Delta t)^2 + O\left((\Delta t)^3\right) \\
\vec{a}(t+\Delta t) &= \vec{a}(t) + \vec{b}(t)\cdot\Delta t + O\left((\Delta t)^2\right)
\end{aligned}
\tag{2.7}
$$

Time integration algorithms in MD models are based on Eq. 2.7. One of the most popular methods is the Verlet algorithm [188, 296], noted for its accuracy and simplicity. A useful property of Newton's equations of motion is that they are invariant under time reversal. Taking advantage of this, in the original form of the Verlet algorithm, the position $\vec{x}(t + \Delta t)$ in Eq. 2.7 is expressed in forward and backward time steps, i.e. both $+\Delta t$ and $-\Delta t$ are substituted for the time step in this equation. This yields Eq. 2.8.

$$
\begin{aligned}
\vec{x}(t+\Delta t) &= \vec{x}(t) + \vec{v}(t)\cdot\Delta t + \frac{1}{2}\vec{a}(t)\cdot(\Delta t)^2 + \frac{1}{6}\vec{b}(t)\cdot(\Delta t)^3 + O\left((\Delta t)^4\right) \\
\vec{x}(t-\Delta t) &= \vec{x}(t) - \vec{v}(t)\cdot\Delta t + \frac{1}{2}\vec{a}(t)\cdot(\Delta t)^2 - \frac{1}{6}\vec{b}(t)\cdot(\Delta t)^3 + O\left((\Delta t)^4\right) \\
&\Rightarrow \vec{x}(t+\Delta t) + \vec{x}(t-\Delta t) = 2\vec{x}(t) + \vec{a}(t)\cdot(\Delta t)^2 + O\left((\Delta t)^4\right)
\end{aligned}
\tag{2.8}
$$

Substitution of Eq. 2.3 into Eq. 2.8 yields:

$$
\vec{x}(t+\Delta t) = 2\vec{x}(t) - \vec{x}(t-\Delta t) - \frac{\vec{\nabla}U}{m}(\Delta t)^2 + O\left((\Delta t)^4\right) \tag{2.9}
$$

Since the original Verlet algorithm does not calculate velocity explicitly, a more accurate version, the leapfrog Verlet algorithm, has been developed [188]. In the latter, the velocity at $t + \frac{\Delta t}{2}$ is approximated at the midpoint between $\vec{x}(t)$ and $\vec{x}\,(t + \Delta t)$.

$$\begin{aligned} \vec{v}\left(t + \frac{\Delta t}{2}\right) &\approx \frac{\vec{x}(t+\Delta t) - \vec{x}(t)}{\Delta t} \\ \Rightarrow \vec{x}(t+\Delta t) &\approx \vec{x}(t) + \vec{v}\left(t + \frac{\Delta t}{2}\right) \cdot \Delta t \end{aligned} \tag{2.10}$$

Analogously, the velocity at $t - \frac{\Delta t}{2}$ is approximated at the midpoint between $\vec{x}(t - \Delta t)$ and $\vec{x}(t)$:

$$\vec{v}\left(t - \frac{\Delta t}{2}\right) \approx \frac{\vec{x}(t) - \vec{x}(t-\Delta t)}{\Delta t} \tag{2.11}$$

Following the same logic, the acceleration $\vec{a}(t)$ can be approximated between $\vec{v}\left(t - \frac{\Delta t}{2}\right)$ and $\vec{v}\left(t + \frac{\Delta t}{2}\right)$:

$$\begin{aligned} \vec{a}(t) = -\frac{\vec{\nabla}U}{m} &\approx \frac{\vec{v}\left(t + \frac{\Delta t}{2}\right) - \vec{v}\left(t - \frac{\Delta t}{2}\right)}{\Delta t} \\ \Rightarrow \vec{v}\left(t + \frac{\Delta t}{2}\right) &\approx \vec{v}\left(t - \frac{\Delta t}{2}\right) - \frac{\vec{\nabla}U}{m} \cdot \Delta t \end{aligned} \tag{2.12}$$

Eqs. 2.10 and 2.12 give the equations of motion in the leapfrog Verlet algorithm. Other integration mechanisms like the velocity Verlet and Beeman methods are also used in MD models [34, 188].

To be able to describe the system dynamics accurately, the time step Δt must be small enough to resolve the fastest process. In the case of irradiated materials, the smallest time step corresponds to the atomic vibration and to the characteristic time related to the projectile velocity, which can be as small as a fraction of fs.

As far as length scales are concerned, MD simulations scale with the number of atoms, which sets a limitation on the size of the simulation box. In practice, MD can simulate the evolution of atoms in boxes of a few hundreds of nm [8, 9, 39, 171, 197, 324], which involves already several hundred million atoms. This size is usually large enough to describe the radiation damage done by collision cascades, but too small in comparison with the length and depth of materials samples used in the laboratory or characteristic of reactor components. Therefore, final simulation times in these models usually range from ps to ns, in the time frame of collision cascade relaxation [8, 9, 10, 164, 171, 197].

MD is a also useful tool to estimate the properties of defects and defect clusters in irradiated materials. Using MD models, it is possible to extract important information such as formation, migration and binding energies of defects [23, 39, 173], the stability of defect clusters [324] or their response to magnetism [85, 171]. In addition, the interaction mechanisms of defects can be studied with MD [23, 173, 206]. Approximations like the Binary Collision Approximation (BCA), detailed in section 2.2, may be applied to enhance performance under certain conditions. All of this information is helpful for the study of materials at higher length and time scales, with simulation techniques that are able to achieve them, such as BCA, KMC or MFRT. The latter are introduced in sections 2.3 and 2.4 respectively.

2.2 Binary Collision Approximation

In order to model the generation of defects in a crystal under bombardment by energetic particles, it is usually a good approximation to consider the transfer of energy between impinging particles and the closest atoms in their vicinity [215, 243, 244, 245, 246, 247]. The BCA relies on the assumption that the interactions of energetic projectiles with solid crystals can be described by a collection of two-body encounters. The following assumptions are made in the BCA [245, 246]:

1. Collisions between energetic projectiles and their nearest atoms are considered, and these encounters are considered isolated from the rest of the crystal.

2. The trajectories of colliding atoms are straight lines.

3. Interatomic forces are central and conservative; inelastic collisions with electrons are negligible with respect to elastic collisions with atoms.

As the problem with many-body collisions present in MD is reduced to a series of two-body collisions, the BCA is an efficient technique to speed up computation [55, 215, 243, 244, 245, 246, 247]. This is particularly true for high-energy regimes such as sputter erosion, since MD methods become computationally costly at higher energies [51, 55, 113, 215, 251, 252].

For each collision, and a given impact parameter p, the scattering integral is solved between the moving atom and a stationary lattice atom. For a potential $U(r)$, the scattering angle θ is determined by Eq. 2.13 [215]:

$$\theta = \pi - 2p \int_R^\infty \frac{dr}{r^2 g(r)} \tag{2.13}$$

where:

$$g(r) = \sqrt{1 - \left(\frac{p}{r}\right)^2 - \frac{U(r)}{E_r}} \tag{2.14}$$

In Eqs. 2.13 and 2.14, E_r represents the relative kinetic energy of the centre-of-mass and R is the minimum distance between particles, also known as the apsis [215, 246]. At this point, $g(R) = 0$.

As far as the impact parameter is concerned, BCA methods can be roughly divided into two types, namely those intended for amorphous materials – or materials whose structure can be ignored – such as SRIM [325], and those intended for materials with a crystalline structure such as MARLOWE [244, 247]. In BCA codes where the material structure is amorphous, the impact parameter p is computed as a function of the composition and density of the target material [325]. In BCA codes which take the crystal lattice into consideration, the impact parameter may also be computed by examining the trajectory of the moving atom along the crystal, which is the case in MARLOWE [243, 244, 245, 246, 247]. The latter method has the advantage of being able to simulate channeling phenomena [55, 215, 243].

In typical BCA codes like SRIM and MARLOWE, one of the input parameters is the displacement energy E_d [244, 247, 251, 252, 325]. It is assumed that, for recoil energies lower than E_d, an ion will not produce damage, whereas if the recoil energy is greater, radiation damage will be produced. Some codes replace this value with an *effective* displacement energy $E_{d,eff}$, which is usually chosen with a larger value than E_d [51, 113].

The BCA enjoys great popularity in the scientific community for its accuracy and applicability [51, 113, 251, 252]. A more thorough description of the BCA and its ongoing improvements can be found in Ref. [215]. One important drawback of the BCA is its limited usefulness in the range of smaller kinetic energies, as multi-body interactions can become significant [55, 252]. In that scenario, the original assumption that only binary collisions are important breaks down.

We shall now discuss the KMC method, which is of great relevance to the present work.

2.3 Kinetic Monte Carlo

Many processes that are activated thermally, like the migration of interstitials, have got a near zero probability to be undergone in the time scales of atomic vibrations or collision cascades. These processes are known as rare events, and happen in time scales that are unreachable by MD. For longer times (ns to μs and beyond), these rare events are crucial to understand the evolution of particle systems since it corresponds to the time scale during which defects can migrate and interact with the microstructure. Therefore, it is necessary to use an alternative simulation tool to MD in order to predict the evolution of defects.

The KMC method is able to overcome the size and time limitations of MD models. Instead of attempting to simulate the dynamics of all particles in detail, frequent events that happen in the time scales of vibrations and cascade collisions are ignored, and the focus is set on rare events such as migration of defects that occurs in a much longer time scale. The KMC method is stochastic by nature, not deterministic, as random numbers are used to decide the time evolution of the particle system. In the most intuitive KMC algorithm, called the rejection algorithm, the probability of a particle (picked at random) to perform a rare event can be expressed by a number p between 0 and 1. A uniform random number $0 < \xi \leq 1$ is then used to determine whether the particle will or will not perform the event. Specifically, the inequality $\xi \leq \frac{p}{n}$ is checked for n iterations until it holds [70, 256]. In an alternative strategy, the rejection-free algorithm, a cumulative list of all normalised event rates, called frequency line in some publications [184], is set up. Figure 2.2 provides a schematic example of a frequency line. In a particle system with defects of types vacancy, SIA and I_2, four possible types of events are allowed to happen, namely the migrations of each of these three defect types and the dissociation of I_2 clusters. The symbols $r_{V,m}$, $r_{I,m}$ and $r_{I2,m}$ represent the rates of vacancy migration, interstitial migration and I_2 migration respectively, normalised

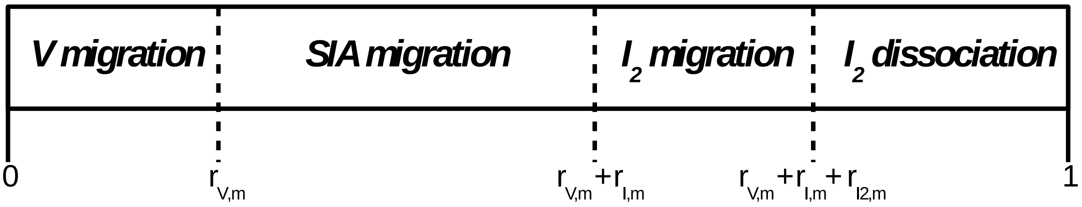


Figure 2.2: Frequency line in a KMC simulation with defects of types V, SIA and I_2.

by the sum of all event rates in the system. Trivially, the rate of I_2 dissociation completes the frequency line in the rejection-free algorithm.

Then, a uniform random number $0 < \xi \leq 1$ is used to sample from this list and select the event that shall be undergone. The rejection-free algorithm, also known as Bortz-Kalos-Leibowitz algorithm [46, 70, 300], is of great importance to our work and is presented in more detail in section 4.1.

After the selected event has been chosen and undergone, the probable time step that may have passed between two events is calculated. As the time step adapts to the nature of the events that may take place at the moment, the evolution of a particle system can be followed with the KMC method for very different time scales (from ns to hours). In principle, this allows for direct comparison of KMC models to experiments, which is unfeasible with MD models due to their short time scales.

Rare events that are activated thermally usually follow a Boltzmann distribution and are characterised by event rates, i.e. the average number of times that a rare event shall trigger per unit time. This shall be explained in more detail in section 4.1. Most KMC models are unable to predict event rates, in contrast to MD, so they must be known a priori [70, 105, 176, 205, 256, 300]. Some KMC models like SEAKMC [312, 314], SLKMC [200] or k-ART [96, 36] are also able to predict the event rates via 'on-the-fly' calculations. In these models, the local neighbourhood of atoms is analysed, transition probabilities are determined as needed and the transitions are computed accordingly. Three important subtypes of KMC methods can be distinguished, namely Atomistic Kinetic Monte Carlo (AKMC), Object

Kinetic Monte Carlo (OKMC) and event-based models, which shall be explained in the following subsections.

2.3.1 Atomistic Kinetic Monte Carlo

The evolution of alloys such as FeCr, of high importance in future fusion reactors [110, 129, 176, 281, 315], at high temperatures is determined by atomistic processes, such as the diffusion of vacancies.

The AKMC method has been developed in order to simulate the evolution of defects in alloys accurately. In an AKMC model, defect diffusion is modelled as follows [45, 64, 176, 267, 282, 297]. A rigid lattice is simulated, and its nodes may be occupied by an atom or unoccupied (vacant). In some models [202, 282], SIAs are included in the simulation as well. A lattice atom neighbouring a vacancy may exchange its lattice position with the latter, giving rise to the vacancy jump. Similarly, a SIA is able to perform a jump which depends on its configuration [202, 282]. For example, an interstitial atom in dilute FeCr alloys occupies a $\langle 110 \rangle$ dumbbell configuration, which may change into a $\langle 111 \rangle$ crowdion configuration with a different migration energy in more concentrated alloys [282].

Defect migration is a thermally activated process characterised by activation energy ΔE, which depends on the local atomic configuration. In fact, the rate of defect migration to a neighbouring site can be calculated as a function of the temperature, attempt frequency (related to atomic vibrations) and ΔE. This is computed for each possible site that the defect may migrate to. Afterwards, a random number is chosen to determine which one of the possible jumps is performed. After the defect has migrated to the chosen site, time is advanced.

The core of the AKMC method resides in the accurate and efficient evaluation of ΔE for all neighbouring sites of defects [45, 64, 297]. To achieve this, a number of techniques have been developed, for example:

1. The energy difference method [202, 297]: First, the energy of the whole lattice before the jump is calculated as E_i, and the energy afterwards is calculated as E_f. Then, the activation energy is computed at the midpoint between E_f and E_i:

$$\Delta E = E_0 + \frac{E_f - E_i}{2} \tag{2.15}$$

where E_0 is an input constant that depends on the exchanged atom.

2. Broken bond models [185, 297]: The activation energy is calculated as the difference between the energy at the stable position of the atom (lattice site) E_{st} and at the saddle point between two lattice sites E_{sp}:

$$\Delta E = E_{sp} - E_{st} \tag{2.16}$$

   In order to find the value of E_{st}, pairwise interactions between nearest neighbours and, in some models, second or further nearest neighbours are considered.

3. Artificial neural networks [63, 64, 281, 282]: The value of ΔE is computed on the fly using an artificial neural network, which is previously set up and trained to predict the values of ΔE. This method has shown to significantly speed up simulations and save runtime.

AKMC simulations are the main computational tool to describe diffusion in alloys, in particular concentrated alloys. Since the lattice is simulated at the atomic level, in the AKMC method, atomistic processes that take place in the alloy such as the formation of complex clusters with solute atoms [202, 267], the response of migrating defects to magnetism [185] or the emission of vacancy-solute pairs from clusters [64] can be explained accurately. Therefore, AKMC is a powerful choice to study the behaviour of alloys at high temperatures for longer times than what is achievable by MD.

Unfortunately, the high computational cost that is inherent to AKMC simulations causes the latter to be usually limited in size. As the volume of the simulated lattice grows, so does the number of lattice sites, and the computational effort increases significantly. Hence, in AKMC models, typical simulation volumes average a few tens of unit cells [267, 281, 297], in the order of MD simulation boxes. In addition, the simulation of several mechanisms such as interstitial clustering, one-dimensional migration of DLs or solute cluster precipitation is difficult to implement and tends to increase the complexity of AKMC models [267, 297], so simplifications and approximations are necessary in many cases. As mentioned by Soisson et al. [267], the inclusion of complex processes that affect reaction kinetics is a challenge for modern AKMC models.

2.3.2 Object Kinetic Monte Carlo

The OKMC method is another widely used variant of KMC simulations in materials science. In OKMC simulations, defects can be designed as objects with a series of properties such as positions, migration and binding energies, dimensionality of motion, and so on. Hence, single point defects, sinks or clusters are modelled as defects with different internal properties, and interactions between defects are defined according to the properties of each defect [26, 33, 70, 109, 176, 270, 313]. External events like continuous irradiation are also considered, as the defects that are created under collision cascades can be implemented with ease. This level of simulation is certainly more abstract than other methods, due to the fact that only the defects themselves are followed, and on-lattice atoms are ignored.

All properties of defects and the events that they may perform must be known beforehand. In other words, it must be known a priori whether a defect is mobile, and if it is the case, in how many dimensions it moves and with what migration energy. Similarly, the dissociation of clusters is considered an event. Hence, the stability of a cluster and its associated binding energies must be provided in advance, since the OKMC method cannot predict them. Interactions between defects are generally described by an instantaneous reaction that occurs within a capture radius. Consequently, the capture radii related to defect interactions have to be deduced in advance as well.

In nuclear materials science, the OKMC method is widely applicable to many different problems like He trapping in W [32], DL diffusion [33, 144, 176] or void and bubble motion in Fe [26, 109] and presents a number of advantages. As in-lattice atoms are ignored, memory constraints in OKMC are not as strict as in AKMC, so larger simulation boxes, comprising a larger portion of the material, can be reproduced. Another difference with AKMC is that as a first approximation, the migration energy is considered constant in most OKMC methods and does not need to be evaluated as a function of the local atomic environment at every time step. As a consequence, the number of computational operations that need to be done in an OKMC simulation is, in general, smaller than in AKMC. For this reason, if the medium can be considered isotropic, the OKMC method offers a precise description of system evolution, and is able to simulate larger volumes than AKMC, which allows for the study of defect evolution at larger space and time scales.

However, even though OKMC simulations offer this computational advantage over the AKMC method, the sample volumes that can be simulated with OKMC are not arbitrarily large. In fact, the achievable simulation box size is limited to several hundreds of nm [26, 33, 270, 313] because computational effort increases with the number of defects present in an OKMC model. When irradiation flux is high or when defects are highly mobile, runtimes in OKMC simulations may also become prohibitive as a consequence of the small time steps imposed by the rate of the different possible events. Hence, this requires a large number of computational steps to achieve the final simulation times. Moreover, phase changes and concentrated alloy modelling are not as straightforward in OKMC as in other methods like AKMC. Due to the fact that lattice atoms are not taken into consideration, there is a lack of information concerning atomistic details [313]. Therefore, an additional effort is needed to model solute atoms in alloys. This increases the complexity of OKMC simulations, and may lead to inconsistencies in case of concentrated alloys, which is why OKMC methods are frequently limited to diluted alloys [26, 88, 110].

2.3.3 Event-based models

A third family of Kinetic Monte Carlo methods exists, which address events themselves to save runtime. Examples of these include event-based Kinetic Monte Carlo (EKMC) and First-Passage Kinetic Monte Carlo (FPKMC).

In many KMC simulations, an important part of runtime is spent on simulating jumps of fast-moving defects [205]. EKMC models like JERK [81] ignore the details of defect motion, which is grouped into 'trajectories', and advance evolution by performing events. In this context, events include defect transitions as well as defect diffusion that results in an interaction with another defect. The JERK model has been used to study the effect of electron irradiation of α-Fe thin foils [81], homogeneous precipitation in Fe-Cu alloys [149] or resistivity recovery (RR) in carbon-doped α-Fe [151].

Opplestrup et al. proposed [212] and later detailed [213] another possibility to speed up KMC computations by focusing on events, called the FPKMC method. FPKMC is built upon the Kalos-Levesque-Verlet (KLV) algorithm [153], which relies on creating non-overlapping protective environments surrounding all particles, where they can diffuse freely. These protective environments are allowed to grow in size until two of them intersect, and at this point, the time needed by each particle to reach the boundary of its protective space is sampled. In the KLV algorithm, after the shortest time of first arrival τ_{min} has been identified, the associated particle is moved to a point on the surface of its protective environment and time is advanced by τ_{min}. Then, for all the remaining particles, the probability that they have not reached the boundary of their protective environments at time τ_{min} is calculated with the help of Green's functions, in accordance to the theory of first-passage processes [153, 212, 213]. Their positions are updated, the system clock is advanced by τ_{min} and the procedure is repeated. Instead of saving only the minimum of all first arrival times τ_{min}, a list of all first arrival times is preserved. The involved particle is moved to the calculated position and then only its protective environment (instead of all environments) is created again. The time of this particle's first arrival to this new environment is sampled and inserted into the list of first arrival times. Using this method, the original problem involving N particles is reduced to N problems involving 1 particle, which is computationally more efficient [212, 213].

2.4 Mean Field Rate Theory

When the evolution of defects in large volumes must be predicted over long times, the runtime required with KMC techniques becomes unaffordable. Therefore, another more efficient simulation technique is needed. In large volumes, the number of defects in the simulation box becomes very large, which, as we have seen, increases the computational cost of KMC, and tracking their positions may become impractical. Instead, it is computationally easier to work with space and time averages (like particle concentrations), if the physical conditions of the experiment can justify this approximation. This is the mean-field approximation on which MFRT relies. In this method, the simulation box is divided into infinitesimal volumes dV. The premise behind MFRT is that defects are distributed uniformly in these dV, so there is no preference for any type of defect in particular and correlations between defects are neglected [205, 270]. In this context, correlations are instances where the distance between two or more defects is so small that their movement is influenced by their neighbouring defects. MFRT can be viewed as an approximation to the continuum, as the idea of distinct defects with individual properties is replaced by macroscopic averages such as concentrations. As the notion of *simulation volume* disappears, the systems that may be modelled with this technique are in principle unbounded by volume, which is a clear advantage over other simulation methods.

MFRT is a deterministic method, in contrast to KMC. Whereas random numbers are used in the KMC method, and therefore two simulations with the same initial conditions may deliver different results, the final result is identical in two MFRT simulations if their initial conditions are the same.

In MFRT, the concentrations C_j of every defect species j are followed in time in order to describe the evolution of the system. For every species j, a diffusion-reaction equation governing the evolution of its concentration in space and time can be expressed as [214, 216]:

$$\frac{\partial C_j}{\partial t} = D_j \nabla^2 C_j + G_j + \sum_i \sum_k r^+_{ik}(C_i, C_k) - \sum_k r^-_{jk}(C_j, C_k) - r^-_j(C_j) - L_j \tag{2.17}$$

The terms used in this equation correspond to:

- A Fickian term corresponding to the diffusion of particles with a diffusion coefficient D_j.
- The direct generation rate G_j. This term represents the production of defects of type j by mechanisms other than interactions between defects, such as the introduction of this type of defects in a collision cascade.
- The rate of creation $r^+_{ik}(C_i, C_k)$. This is the production rate of defects of type j due to reactions between defects of type i with defects of type k. This would correspond, for instance, to the creation of vacancy clusters V_n via absorption of vacancies by V_{n-1} clusters.
- The rate of destruction $r^-_{jk}(C_j, C_k)$. This is the rate at which defects of type j are lost by interaction with other defects.
- The rate of evaporation $r^-_j(C_j)$. This rate represents the loss of clusters by means of defect emission.

- The direct loss rate L_j. This is the average loss of particles of type j to grain boundaries or other types of sinks that are not taken into account in the term $\sum_k r_{jk}^{-}(C_j, C_k)$.

Other terms, such as the drift of a charged particle in an electric field, can be taken into account as well. Hence, the original problem of tracking the individual position of defects in time, inherent to KMC method, is translated into a set of coupled partial differential equations.

All species that may potentially appear in the simulation must be taken into consideration. Since the number of equations in set Eq. 2.17 cannot be infinite, in the case of defect clustering, a large upper limit for the cluster size is typically defined to ensure that all $A_n + A \rightleftharpoons A_{n+1}$ reactions are accounted for [214, 270].

Like in the case of the KMC method, the input parameters in MFRT like capture radii and activation energies have to be known in advance. In some cases, the coefficients needed for Eq. 2.17 are straightforward to compute. Let us consider the simple example of particle recombination, i.e. a system with particles of types A and B that annihilate each other by the reaction $A + B \rightarrow 0$ with reaction constant k. The respective diffusion coefficients are D_A and D_B. In this case, Eq. 2.17 is written as:

$$\begin{aligned} \frac{\partial C_A}{\partial t} &= D_A \nabla^2 C_A - k C_A C_B \\ \frac{\partial C_B}{\partial t} &= D_B \nabla^2 C_B - k C_B C_A \end{aligned} \tag{2.18}$$

If particles A and B diffuse randomly in three dimensions and r_c is the capture radius defined for the recombination between them, then it can be shown that the reaction constant is calculated as $k = 4\pi r_c (D_A + D_B)$ [301]. The set Eq. 2.18 is then integrated by numerical methods, therefore completely determining the time evolution of the particle system. In general, this process is computationally cheap.Therefore, the time scales that may be reached by MFRT are very long, in the order of the lifetime of a nuclear reactor like ITER and beyond. — In addition, there are in principle no upper limits to concentration in a MFRT simulation, which can achieve large damage doses at high temperatures with little computational effort [81, 82, 214, 216, 270]. MFRT may therefore seem like a very attractive computational tool to model materials under intense irradiation, which, as we have seen in the previous chapter, is the case in DEMO. However, the fact that MFRT does not take correlations into consideration may lead to unphysical results if there are high localised concentrations of defects. — Some MFRT models attempt to minimise this effect by including mechanisms to account for spatial resolution [92, 150, 158, 311], usually with the help of finite difference methods.

Another disadvantage of MFRT is that the terms in Eq. 2.17 are not always evident to compute. If these terms are complex, they (and their associated coefficients) must be derived before executing the simulation. It might be necessary to develop new, complex mathematical methods, which makes the applicability of MFRT certainly not universal. Furthermore, the fact that MFRT does not treat particles as distinct individuals but as collectives means that some MFRT results involve a problematic interpretation. For instance, an inaccurate MFRT simulation could potentially show a negative concentration, which is unphysical.

2.5 Justification for the choice of OKMC in this work

Amongst the objectives of this work, it was mentioned in section 1.3 that the algorithm to be developed ought to reach simulation times close to those achieved experimentally, in the order of seconds and beyond. In addition, the size of the simulation box should be in the order of grain size in an irradiated sample. For these reasons, the MD method is naturally discarded since it only allows reaching physical times in the order of ps to ns, as explained in section 2.1.

Hence, the remaining possible choices for our model are KMC or MFRT. These methods allow for the simulation of longer times, and, in the case of OKMC and MFRT, larger simulation volumes, as described previously. In order to simulate boxes of hundreds of nm in size and beyond, the AKMC method is computationally too expensive. Our intention is to understand physical processes taking place at length scales that are comparable to the grain size, so AKMC was discarded as an option. Instead, spatial isotropy is assumed in OKMC and MFRT, and activation energies are considered constant.

For the sake of accuracy, OKMC was chosen as the basis for our model. Even though MFRT calculations are much faster, the fact that correlations and local inhomogeneities are not taken into account makes MFRT disadvantageous in order to understand atomistic mechanisms. Since the positions of particles are tracked individually in OKMC, local effects are accounted for, in contrast to MFRT.

Moreover, the rate of some interactions such as the reaction between a DL migrating in one dimension and a dislocation line, may be difficult to estimate. In contrast, these interactions are naturally accounted for in the OKMC method, the jump frequency of defects being the only knowledge required. This makes OKMC a more accurate tool than MFRT in order to represent systems with a high density of defects like materials subjected to intense neutron irradiation. Furthermore, in the case of OKMC, there is no need to set a maximum size of particle clusters, which is virtually unlimited, unlike in MFRT where a matrix of a fixed size must be defined in advance. These advantages over MFRT thus support our choice to base this work on the OKMC method.

Finally, one of the consequences of individual particle treatment by the OKMC method is that it maps well to parallel programming, which can be used to speed up simulations. The probabilities of thermally activated events only depend on the activation energies involved, which in the OKMC framework are modelled as properties of each defect. Thus, these probabilities are independent of each other and the problem of following the evolution of a system with many particles can be divided into smaller problems. The latter are then solved simultaneously and independently of each other. In chapter 4, we shall show how the MEGA-OKMC code has been implemented to be efficient even in an inhomogeneous system with an array of different types of defects, as the time step only depends on the nature of the defects with the largest sum of event rates. As one shall see in chapters 4 and 5, it is also possible to enable continuous irradiation in MEGA-OKMC without disturbing performance. Parallel programming is introduced in the following chapter, giving special attention to the implementation used in this work.

Chapter 3

Parallel programming on GPUs

In this chapter, general concepts on parallel programming are presented in section 3.1, with focus on general-purpose programming on graphics processing units (GPGPU). The CUDA programming framework used in this work is introduced in section 3.2.

3.1 Parallel computation

Parallel computing refers to the simultaneous use of multiple compute resources to solve a computational problem [125]. In subsection 3.1.1, this concept is explained. If the computational problem is divided into smaller pieces, such that each computational resource is responsible for one of the pieces, it is possible to reduce the runtime required to solve it. Some examples of machines on which parallel computing is applicable are CPU clusters and graphics processing units (GPUs). The use of GPUs to perform general-purpose computation is known as GPGPU. In particular, we chose GPGPU for the implementation of our model. GPGPU is presented in subsection 3.1.2 and the memory model behind it is explained in subsection 3.1.3.

3.1.1 Concept of parallel programming

Traditional computation, more precisely called serial computation, relies on the partition of a problem into instructions that are introduced one after another, such that only one instruction can be executed at a time. In contrast, a parallel computing model may execute several instructions simultaneously or *in parallel*. An important milestone in the history of parallel computing happened in the 1980s, when the Caltech Concurrent Computation Program built a parallel computer – the *Cosmic Cube* – with 64 interconnected commercial Intel processors [257]. This project demonstrated that it was possible to create a competitive supercomputer using commercially available components. More recently, parallel machines have been developed with the purpose of providing useful tools to scientific research, such as the IBM®*Blue Gene*®project, originally intended to help investigators in biology simulations [38].

A parallel computing model may be beneficial to runtime if the problem maps well to a parallel paradigm, which depends on a number of factors. When solving a problem within a parallel framework, the independence of instructions that are computed in parallel is crucial. Indeed, let us imagine a simple system of two processing units A and B respectively executing two instructions I_A and I_B simultaneously. Since each unit is, in principle, unaware of the data being processed on the other, it is necessary to make sure that I_A does not need an input from unit B and vice versa. Therefore, I_A and I_B must be independent of each other. In a parallel application, the smallest instruction unit is called the *thread*. Thus, threads execute instructions in parallel. In general, these threads are allowed to communicate information with the help of a block of memory called the *shared memory* [78, 125]. Although threads execute instructions in parallel logically, there is no guarantee that these instructions are executed simultaneously in time. For this reason, in a flawed application, it might happen that two parallel threads try to change the output of the same memory address. This is known as a race condition. In order to avoid such conflicts, threads that have access to the same shared memory must wait for the execution of other threads in a process called synchronisation.

With this in mind, it is not difficult to understand why the OKMC method is easily mapped to a parallel model. The defect properties mentioned in subsection 2.3.2 are largely independent of each other. If each thread is responsible for the operations concerning one defect, and many parallel threads (tens of thousands or more) are launched, a big number of defects can be processed in a shorter runtime than if these operations were executed serially. This suggests that larger material sizes may be represented, and realistic time scales achieved, if an OKMC model is implemented using parallel computation in an intelligent way.

Several paradigms in parallel programming exist, such as:

1. Message-Passing Interface (MPI) [127]: This is a message passing standard for C and Fortran applications maintained by a wide array of partners, as a way to ensure portability and flexibility across CPU clusters. The first standard was released in 1994.

2. Open Multi Processing (OpenMP) [78]: Originally released in 1997, this model was originally developed as a portable alternative to MPI. Programming in MPI can prove difficult [78], so OpenMP is equipped with a set of simple directives and routines that provide the desired parallelism. OpenMP supports C, C++ and Fortran on many architectures.

3. General-Purpose Programming on the GPU (GPGPU): As we will explain in subsection 3.1.2, the GPU is used to process data in parallel. GPUs are a cheap and efficient alternative to CPU clusters, as far as parallel computing is concerned. Since GPUs are commonplace whereas access to expensive CPU clusters is usually restricted, GPGPU is increasingly popular. Several implementations of MPI and the newest OpenMP standard support GPU programming [76]. In addition, several programming models are explicitly oriented towards GPGPU, such as CUDA [207] or OpenGL [265].

MPI is suited for both shared memory computers and distributed systems, i.e. parallelism over several nodes, whereas OpenMP is limited to shared memory computers. Thus, an advantage of MPI over OpenMP is that MPI applications are not restricted to one node, unlike OpenMP ones. On the other hand, OpenMP is useful for incremental parallelism, which is the parallelisation of parts of the application with a minimal change in the code. In particular, the execution of a loop in parallel, also known as loop-level parallelism, is optimised in OpenMP. This makes OpenMP applications portable and easier to maintain than MPI ones [78]. As far as GPUs are concerned, their architecture incorporates a series of multiprocessors that perform operations in parallel, such that many parallel threads –in the order of tens of thousands to millions– can be created. The multiprocessors are able to execute tens of thousands of these threads simultaneously [203]. Parallel applications can be run on a single GPU or on multiple ones, if a paradigm such as MPI is applied to distribute memory across several GPU nodes.

Our work has been implemented on GPGPU for the following reasons. Since each defect is able to perform a series of events independently of all other defects, the instructions that affect each defect can be executed simultaneously. Hence, each one of the defects can be assigned to each thread. However, the number of defects present at each stage of the OKMC simulation is not necessarily a constant. The disappearance of defects by reactions with other defects or by recombination at the surface, as well as the introduction of new defects from irradiation cascades or by emission from defect clusters, have to be taken into consideration. In the case of CPU clusters, this could lead to implementation difficulties, as the amount of computational resources used for the processing of events must be calculated at every step. Moreover, the number of parallel threads launched in a CPU cluster application is a function of the number of available cores, which is correlated to the number of processors in use. This number has to be known in advance, and is in general in the order of tens to thousands [125, 184, 233]. This means that only a few hundred or thousands of particles can be processed effectively at the same time, which induces a delay in case a realistic simulation box with millions of defects is simulated.

In contrast, our OKMC model can be implemented on a single GPU, which can create millions of threads and execute tens of thousands of them simultaneously, as stated above. To produce a similar number of threads, a CPU cluster would need vast computational resources, namely thousands of multicore processors. If MPI is used, a distributed memory system is appropriate, whereas if OpenMP is used, a large shared memory computer is needed. In both cases, the economic and energetic cost of maintaining such a system is orders of magnitude higher than the cost of using a GPU.

Finally, one of the most important bottlenecks in parallel programming is the overhead due to communication between processors [125]. In a MPI application for a distributed memory system, minimising this communication is one of the foremost challenges. This is not an issue for programming on a single GPU. GPU hardware is designed to efficiently connect the GPU multiprocessors with the RAM, thus greatly reducing the overhead [207].

For all these reasons, an OKMC model with hundreds of thousands or millions of particles is easier, cheaper and more efficient to implement on a GPU than on a cluster of CPUs, which justifies our choice of GPGPU. The concept of GPGPU shall be detailed further in the following subsection.

3.1.2 GPGPU

In recent years, has been chosen as an inexpensive, reliable alternative to CPU programming [220]. Owing to their architecture, which involves hundreds of parallel processor cores, GPUs offer a high degree of parallelism, as explained in subsection 3.1.1. Thus, they compensate the lack of processing power of each individual processor (compared to modern CPUs) with a substantial display of parallelism. Driven by the demand for rapid rendering of 3D scenes in video games [203], GPU technology continues to evolve at a faster rate than Moore's law [220]. To illustrate the evolution of GPU processing power over the years, Figure 3.1 depicts the theoretical peak processing power in single-precision floating-point operations (FLOPs) per clock cycle that different generations of GPUs are able to calculate. Clearly, the development of GPU technology has been fast and efficient over the last years.

GPUs became more flexible and programmable at the start of the 21st century [203]. In addition to their low price [220], this brought GPUs to be perceived as an attractive possibility to researchers, marking the start of GPGPU. For this reason, GPGPU is used as a paradigm in many fields of research, including MD models such as LAMMPS [232], NAMD [227] or GROMACS [235], particle transport models [123, 124] or Monte Carlo models for many-particle systems [165, 233, 306]. Due to their versatility and processing power, GPUs are routinely featured in supercomputers, including most of those appearing in the supercomputer reference list TOP500 [272]. Moreover, several studies have found GPUs to display excellent energy efficiency when compared to CPUs [189, 259]. For this reason, the supercomputer list Green500 [259], which ranks the top 500 supercomputers in the world by energy efficiency, explicitly mentions the role of GPU computing in the challenges involved in achieving sustainable standards within the high-performance computing community. At the time of writing this dissertation, the second-highest-

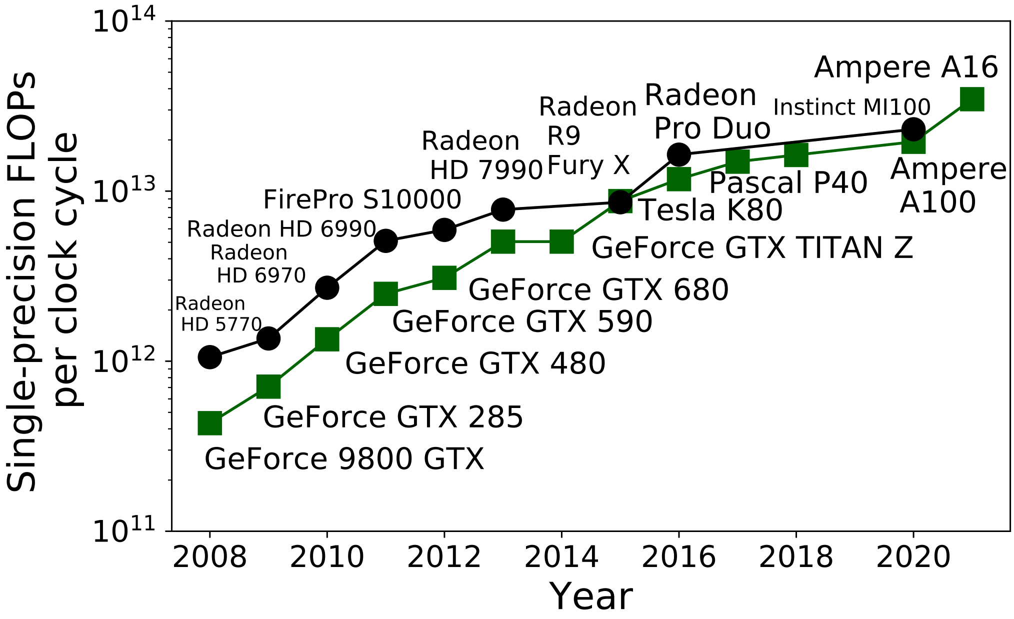


Figure 3.1: Theoretical peak processing power of GPUs in single-precision FLOPs per clock cycle. Source: developer websites [3, 76].

performing supercomputer in the world was *Summit*, located in the Oak Ridge National Laboratory, only behind the *Fugaku* supercomputer in the RIKEN Center for Computational Science [272]. *Summit* is built with 4608 nodes, containing 2 IBM®POWER9™CPUs and 6 Nvidia®Volta®GPUs each [272, 295].

3.1.3 Memory hierarchy in GPGPU

The aspects of the GPU memory hierarchy relevant to our code will be addressed here, as explained by the CUDA programming guide [207].

GPUs (henceforth *devices*) are provided with a *global memory*, which is the device RAM, where data reside and can be up- and downloaded by the CPU (also called the *host*). Each thread is assigned a part of this global memory that can only be accessed by the thread itself, called the local memory. Small variables are stored in thread registers, which are limited to each thread and are the fastest accessible memory regions. Larger variables and arrays are destined to the local memory outside registers. In this case, accessing these variables is computationally slower.

In GPGPU, threads can be grouped into thread blocks, which is another level of parallelism. Blocks are able to carry out operations in parallel, and threads in the same block can share information via the shared memory of that block. GPU shared memory is a fast type of cached memory –i.e. faster than the global memory– that is located on the chip [207] and its scope is limited to the own block. In other words, one block (or, more specifically, threads in the same block) cannot access the shared memory of other blocks. Shared memory is particularly useful in the implementation of the particle interaction algorithm described in the appendix. Since the scope of shared memory is the block it is associated with, shared memory operations carried out in a block are guaranteed not to disturb processes in other blocks and vice versa. Hence, it is straightforward to map our inter-particle reaction algorithm to a GPGPU application. By associating each cell in space to a thread block in the kernel, the interaction algorithm is implemented easily and accurately.

In addition, devices are provided with two types of read-only cached memory, labelled texture and constant memory. Texture memory is optimised for 2D locality, which means that two threads reading from neighbouring memory sites achieve better performance. Constant memory is designed for data that will not change across the application, and may provide a performance advantage over data stored in the global memory.

Figure 3.2 shows a schematic view [76, 207] of the six different types of memory in GPGPU: local memory and registers (with thread scope), shared memory (with block scope) and global, texture and constant memory (with application scope; the last two being read-only memory spaces). In our work, we mainly make use of the global and shared memory. Function types and host-device interoperability in the chosen programming model, namely CUDA, are explained in section 3.2.

3.2 The CUDA programming model

This section is devoted to the traits of the CUDA programming model, which is the paradigm used in this work.

In November 2006, NVIDIA®Corporation launched a new technology named CUDA (Compute Unified Device Architecture) [203, 207]. CUDA, essentially an extension of C/C++, is a parallel computing model designed for NVIDIA graphics

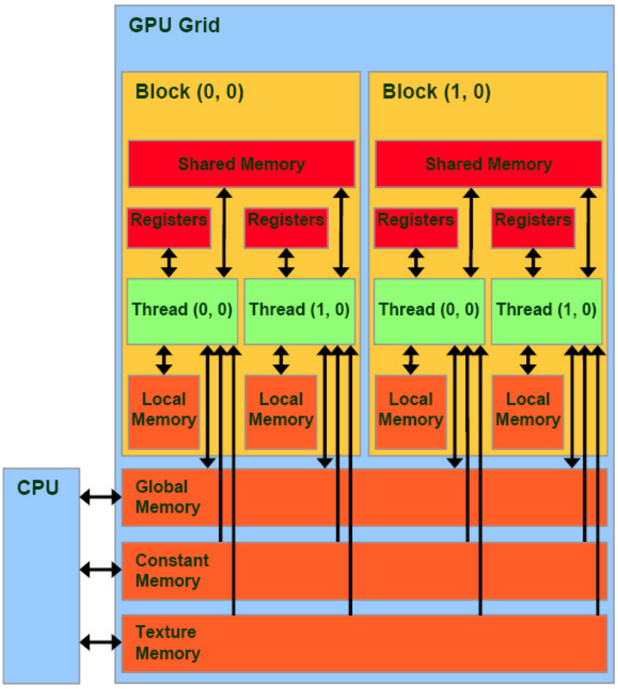


Figure 3.2: Memory types in GPGPU [76, 207].

cards. Here we present the main aspects that ought to be addressed in order to understand the CUDA programming model.

In the CUDA framework [207], a distinction is made between the host and the device. Functions are classified in the CUDA interface into *host functions* (called from and executed on the host), *kernels* (called from the host, but executed on the device) and *device functions* (called from and executed on the device). There is a reserved word for each of these cases, respectively `__host__`, `__global__` and `__device__`, that must be placed in front of the function declaration. If none of these qualifiers are found in the declaration, the function defaults to a host function. This is summarised in Table 3.1.

Called by	Executed on	Name	Qualifier
Host	Host	Host function	`__host__` (default)
Host	Device	Kernel	`__global__`
Device	Device	Device function	`__device__`

Table 3.1: Function types in the CUDA programming model.

A function can be declared with both `__host__` and `__device__` qualifiers, as long as the operation is not illegal on either the host or the device, but kernels must be declared with the `__global__` qualifier exclusively. Kernels in CUDA must be configured by specifying the number of blocks that are launched and the size of these blocks –i.e. the number of threads per block– have to be declared when a kernel is called. Some dedicated high-level libraries for CUDA, such as the Thrust library [134], are designed to manage kernel configuration automatically.

One aspect to keep in mind is that communication between processors is often an issue for implementations of parallel models, as mentioned in subsection 3.1.1. The CPU-GPU interface is no exception, as mentioned in the CUDA programming guide [207], i.e. the transfer of information between host and device should be kept to a minimum to avoid this bottleneck. Therefore, it is usually beneficial to keep the data to be processed on the device as long as possible and retrieve them at the end of the simulation.

We shall now provide a simple example of a problem solved by means of serial computation on the CPU and by parallel computation on a CUDA-capable GPU, namely the sum of two vectors $\vec{c} = \vec{a} + \vec{b}$ of N integer components, where N can take values up to several hundred million. In the case of serial computation, the appropriate operation is a loop that adds the components of vectors $\vec{a}$ and $\vec{b}$ one after another. The function to sum both vectors $\vec{a}$ and $\vec{b}$ in C++ has got this form:

```
void vectorsum_serial(int *a, int *b, int *c){
  int i;
  for (i = 0; i < N; i++){
    c[i] = a[i] + b[i];
  }
}
```

Then, calling this function in the main function is straightforward:

```
vectorsum_serial(a,b,c);
```

The sum of each component $c_i = a_i + b_i$ is independent of the rest of components. Therefore, in parallel computation, each thread can compute one of the components c_i in parallel. In CUDA, it is possible to launch blocks of threads in one, two and three dimensions. Our example involves a sum of vectors, so only one-dimensional blocks are needed. The internal variables `threadIdx.x`, `blockIdx.x` and `blockDim.x` indicate the ID of each thread in its block, the ID of the thread's block and the number of threads per block respectively. The CUDA kernel is written as follows:

```
__global__ void vectorsum_parallel(int *a, int *b, int *c){
  int i = threadIdx.x + blockIdx.x * blockDim.x;
  c[i] = a[i] + b[i];
}
```

In order to call this kernel in the main function, it must be configured. In other words, the number of blocks to be launched and the number of threads per block have to be determined explicitly. We intend to launch as many threads as the number of components in each vector –which is equal to N– due to the fact that each thread is responsible for the addition of one component. We define the size of blocks at 32 threads per block, which is the smallest block size in CUDA to guarantee maximum GPU parallel performance [207], such that `blockDim.x` equals 32. This means that $1 + \left\lfloor \frac{N}{32} \right\rfloor$ blocks must be launched in order to achieve N parallel threads. Here $\lfloor x \rfloor$ represents the floor function, which returns the greatest $n \leq x$ with $n \in \mathbb{Z}$. In CUDA C/C++, integer division automatically applies the floor function. For these reasons, the kernel call in the main function is written as:

```
vectorsum_parallel<<<1 + N/32, 32>>>(a,b,c);
```

The serial addition was performed on an Intel Xeon®CPU ES-1620 v2 @3.70 GHz and the parallel addition was executed on a NVIDIA®GeForce®TITAN Black GPU. Table 3.2 shows the runtime needed by each operation.

As one can see, the gain in runtime –four to five orders of magnitude– offered by the parallel implementation is excellent, which highlights the possibilities of GPGPU. Computational effort is proportional to the number of components in the serial case, such that increasing the vector length by a factor of 25 causes the runtime to increase in a similar factor. However, this is not the case in the parallel operation,

N	CPU runtime	GPU runtime	GPU gain
20×10^6	51.1 ms	0.048 ms	1060×
200×10^6	494.05 ms	0.048 ms	10537×
500×10^6	1246.21 ms	0.059 ms	21122×

Table 3.2: Runtimes needed by the serial (CPU) and parallel (GPU) additions of two vectors of N components.

as runtime increases only slightly when the number of components is increased by an order of magnitude. Moreover, GPU runtime does not change when N is increased from 20×10^6 to 200×10^6. The reason is that the GPU is able to launch millions of parallel threads and optimise memory access in simple operations, such that the addition of very large vectors can be performed effortlessly.

The main concepts behind GPU programming and the CUDA model have been explained in this chapter. This is the chosen implementation for our GPU-OKMC model. With this in mind, chapter 4 is devoted to the main algorithm that we developed in this work.

Chapter 4

Development of a parallel OkMC code based on GPU programming

In this chapter, the basic theory behind MEGA-OkMC is presented. Section 4.1 describes the BKL algorithm, an algorithm of great relevance to OKMC. Section 4.2 shows the PTLA, the algorithm developed by the authors from scratch for the present work. Finally, some simulation results are presented in section 4.3 to validate our algorithm.

4.1 Classic BKL algorithm

This section is devoted to the Bortz-Kalos-Leibowitz (BKL) algorithm, which is widely used in simulations based on the KMC method, like OKMC. First, the concept behind the BKL algorithm is presented in subsection 4.1.1. Later, its advantages and disadvantages are discussed in subsection 4.1.2.

4.1.1 Fundamentals of the BKL algorithm

In a rare-event system [238, 300], transitions or *events* happen from one state σ to another σ' occasionally. If $P(\sigma, t)$ is the probability that the system is in state σ at time t, and $W(\sigma \to \sigma')$ the transition probability that the state will change from state σ to σ' during δt, then the evolution of the probability $P(\sigma, t)$ in time is governed by the master equation [70, 102, 118]:

$$\frac{\partial P(\sigma,t)}{\partial t} = \sum_{\sigma'} W(\sigma' \to \sigma) P(\sigma', t) - \sum_{\sigma'} W(\sigma \to \sigma') P(\sigma, t) \tag{4.1}$$

The idea behind the KMC method is not to provide an exact or analytical solution to this equation [118], which in most cases is not possible, but to compute a realisation of the system trajectory along the time line up to a final time t_{end}. The transition probabilities are assumed independent of each other, and the number of possible transitions must be known a priori and finite [70, 300]. Indeed, since the system evolves in time $\{t_0 < t_1 < \dots\}$ along a series of states $\{\sigma_0, \sigma_1 \dots\}$, the sequence $\{\sigma_k\}$ can be viewed as a Markov chain [238, 256, 300] associated with a Poisson process with transition rate R [102, 256, 300]. R represents the transition probability $W(\sigma_n \to \sigma_{n+1})$ to evolve from state σ_n to state σ_{n+1} if σ_{n+1} is an accessible state, meaning an accessible configuration of the system. Examples of system states are, for instance, the occupation arrays of a cubic crystal [70, 256, 18].

The aim of the KMC method is to produce the sequence $\{\sigma_k\}$ in a time line $\{t_k\}$. It must be noted that a Markov chain is 'memory-less', i.e. the probability of a change of state in a system is independent of the history of the system [102, 119, 300]. Hence, the probable evolution of the system at a given moment can be estimated by the probability that an accessible state will be reached next. As far as the master equation is concerned, we only need to focus on the $W(\sigma \to \sigma')$ values when the states σ' can be accessed from state σ directly. The transition of a state σ_i to an accessible state σ_j, which follows a Poisson process, is called an event and is characterised by the event rate r_j.

The number of possible values that j can take is equal to the number of possible events in the system, i.e. the number of possible states that the system can move to at a certain moment. On average, an event k can occur $r_k \delta t$ times in a time interval $[t, t + \delta t]$. Strictly speaking, we can compute the probability that an event with rate r_k will be triggered n times within $[t, t + \delta t]$ using a Poisson law [44, 119]:

$$P(n; r_k \delta t) = \frac{(r_k \delta t)^n}{n!} \exp(-r_k \delta t) \tag{4.2}$$

Essentially, the systems we want to investigate are ensembles of particles able to perform different events. These events are Poisson processes, and they are independent of each other. For example, the rate of migration or dissociation of a given cluster is a function of the properties of that specific cluster, and independent of all other particles in the particle system. With this premise, we shall take advantage of a useful property of the Poisson distribution, namely the fact that a sum of independent Poisson processes is, itself, a Poisson process. This is easily demonstrated as follows. If $\{X_i\}$ $(i = 1 \dots n)$ is a set of n independent Poisson random variables with parameters $\{\lambda_i\}$, then the variable $X = \sum_i^n X_i$ is a Poisson random variable with parameter $\lambda = \sum_i^n \lambda_i$ [44]. This is clearly the case of the particle systems we intend to study. Indeed, the (total) rate R of the system can be expressed as the sum of all event rates within the system:

$$R = \sum_k^N r_k \tag{4.3}$$

As far as the OKMC method is concerned, in the context of this work, we are mainly interested in thermally activated events [144, 175, 214, 270]. In these cases, the probability that a system will be in a certain state of a given energy at a given temperature usually follows a Boltzmann distribution [88, 118, 300]. If k_B is the Boltzmann constant, T the temperature, and ν_0 the attempt frequency –i.e. the vibration frequency of the atom [70, 88]–, the rate of the event characterised by energy barrier E is given by Eq. 4.4:

$$r_k = \nu_0 \exp\left(-\frac{E}{k_B T}\right) \tag{4.4}$$

It is now clear how to calculate the event rates in the system in order to generate the sequence of system states $\{\sigma_k\}$. The question remains of how to generate $\{t_k\}$, i.e. how to advance time step by step in a rational way. In order to do this, let us consider a system with only one particle able to perform exactly one event with event rate r_k. This is, as mentioned, a Poisson process, and therefore has exponential decay statistics [118, 119, 300]. The probability that the particle has not yet performed the event at time t is [300]:

$$p_{no\,event} = \exp\left(-r_k t\right) \tag{4.5}$$

Therefore, the probability that the event has been performed at t is:

$$p_{event}(t) = 1 - p_{no\,event} = 1 - \exp\left(-r_k t\right) \tag{4.6}$$

Let us now introduce the transition probability density function $f(t)$. This function gives the probability density of times between successive events [102] and is defined as the time derivative of $p_{event}(t)$. This yields Eq. 4.7:

$$f(t) = \frac{dp_{event}(t)}{dt} = r_k \exp\left(-r_k t\right) \tag{4.7}$$

Trivially, the cumulative distribution function $F(t) = \int_0^t f(x)dx = p_{event}(t)$ as given by Eq. 4.6. The inverse transform method [44] is useful to sample the probable time interval t_e between two events in succession. Trivially, by letting $t = F^{-1}(u)$ we get:

$$u = 1 - \exp\left(-r_k t\right) \Rightarrow t = -\frac{\ln\left(1-u\right)}{r_k} \tag{4.8}$$

If $0 \leq u < 1$ is a uniformly distributed random variable, and $\eta = 1 - u$, then $0 < \eta \leq 1$ is also a uniformly distributed variable. By substitution of η into Eq. 4.8, Eq. 4.9 yields the probable time interval t_e between two consecutive events with rate r_k.

$$t_e = -\frac{\ln(\eta)}{r_k} \tag{4.9}$$

Finally, we shall find out how to sample the time interval between two successive events in a rare-event system in general, also known as the time step δt. Due to the fact that the sum of a number of independent Poisson processes is a Poisson process, substitution of the total system rate R into Eq. 4.7 yields the transition probability density function of the whole system:

$$f(t) = R\exp\left(-Rt\right) \tag{4.10}$$

Thus, Eq. 4.11 gives the probable time interval that passes between two consecutive events in the system, i.e. the time step δt.

$$\delta t = -\frac{\ln\left(\eta\right)}{R} \tag{4.11}$$

There are a number of KMC methods to generate trajectories of rare-event systems along a time line. Here, we will focus on one of these techniques that is relevant to this work. Named the residence-time algorithm or BKL algorithm, this method involves two steps [46, 70, 118, 300], namely the computation of all possible event rates in the system and the selection of the process to be undergone.

The residence-time Monte Carlo algorithm was described by Young and Elcock [321] as a way to study vacancy migration in binary alloys. In their 1975 paper, Bortz, Kalos and Leibowitz (BKL) [46] applied their version, initially called the *n-Fold Way*, to the evolution of Ising spin systems. In this algorithm, usually referred to as Bortz-Kalos-Leibowitz or BKL algorithm, all possible event rates r_i in the system must be given a priori. It is also equivalent to the stochastic simulation algorithm (SSA) detailed by Gillespie [118]. The BKL algorithm can be described along these lines:

1. Time is initialised to $t = 0$.

2. All transitions j available to particles i in the system are identified and their individual event rates $r_{i,j}$ computed.

3. The sum $R_j = \sum_{i=1}^{N_j} r_{i,j}$ is calculated, where N_j is the number of particles that can undergo the event j.

4. The sum of all event rates corresponding to the N possible types of events is computed as $R = \sum_{i=1}^{N} R_i$. This is the total rate of the system.

5. A uniform random number ξ such that $0 < \xi \leq 1$ is produced to choose the event that will be undergone during this time step.

6. Event i such that $\frac{R_{i-1}}{R} < \xi \leq \frac{R_i}{R}$ is chosen.

7. One of the particles able to perform an event of type i is chosen randomly, and the event is carried out.

8. The system state is updated after the event has been undergone by the selected particle.

9. Another uniform random number η such that $0 < \eta \leq 1$ is produced in order to advance time.

10. Time is advanced by $\delta t = -\frac{\ln(\eta)}{R}$, such that $t' = t + \delta t$.

11. Steps 2 to 10 are repeated until the final simulation time t_{end} has been reached.

It should be mentioned that the average of $-\ln(\eta)$ calculated in step 10 is exactly 1, so it is common to compute the time step as $\delta t = \frac{1}{R}$ instead, in particular if the total number of time steps is large. [31, 70, 88]

4.1.2 Advantages and disadvantages of the classic BKL algorithm

The BKL algorithm has been extensively used in OKMC models for decades, particularly in the research field of materials physics [31, 33, 88, 175, 176, 177, 214]. An important advantage is that the time step conforms dynamically to the situation, explained by the fact that it depends on the event rates existing at the moment. Indeed, if certain particles vanish from the system and/or others appear in it, the time step is adjusted automatically to reflect this fact, as the calculation of R in Eq. 4.3 yields a different result. As a consequence, δt, which is inversely proportional to R as shown in Eq. 4.11, is updated as well. Therefore, the BKL algorithm gives an idea about the time scales associated with the evolution of the particle system under scrutiny.

However, the inversely proportional dependence of the time step δt on the total system rate R also reflects a limitation of the BKL algorithm when the simulation box consists of a large amount of particles. As shown by Eq. 4.3, R is calculated as the sum of event rates in the system. When many particles able to perform events are present, their event rates contribute to R, which may take a large value. Trivially, Eq. 4.11 indicates that large values for R imply small δt values. If the time step is very small, the number of computational steps needed to reach a final simulation time will be very large, which implies a long runtime to achieve the simulation.

In addition, as seen in step 7 of the BKL algorithm detailed in subsection 4.1.1, only one event is performed in the whole particle system per time step. Hence, all particles in the simulation box except one –the particle that performs the chosen event– are idle during δt. Many computational steps are thus necessary to notice significant changes in microstructural evolution of the system as a whole, which becomes a major problem for simulation boxes with hundreds of thousands or millions of particles.

Due to these drawbacks, simulations based on the BKL algorithm must restrict themselves to simulation boxes with a limited number of particles in order to avoid prohibitive runtimes. To do so, the size of the simulation box is reduced, which is equivalent to reducing the size of the sample, often to unrealistic sizes [270].

For instance, typical simulation box sizes average a few hundred lattice parameters in depth [31, 39, 175, 176, 270, 306, 313]. Clearly, this strong restriction seriously hinders the study of defect evolution in realistic pieces of materials that can contain polycrystals and grain boundaries, which play an important role in the evolution of defects and/or impurities and thus, cannot be neglected [11, 24, 56, 283, 305]. In fact, experimental grains are in general in the order of μm [129, 191, 315]. Small sizes of simulation boxes become particularly problematic for defects which are typically present in low concentrations, such as DLs in Fe. For instance, in the experiment of Arakawa et al. [16] where Fe was electron-irradiated, DLs were found with densities in the order $10^{16} - 10^{17}$ cm^{-3}. Considering a typical OKMC simulation box of $(200a_0)^3$ in size, this implies that it is only possible to follow the evolution of 1 to 20 DL, which is clearly not enough statistical information for the prediction of defect evolution and their influence on a macroscopic level.

A number of acceleration mechanisms for the OKMC method have been proposed in the scientific community [69, 70, 119, 286], but the size of the problem remains an issue for large simulation boxes with hundreds of thousands or millions of particles. Our work tries to overcome these limitations, as we shall see in section 4.2.

4.1.3 Previous attempts to accelerate the BKL algorithm: Parallelisation

One possible way to accelerate the BKL algorithm is by means of parallelisation. Previous work has been done on OKMC models based on parallel codes [182, 184]. Notably, Martínez et al. [184] developed a parallel KMC code, based on the BKL algorithm, that relies on the partition of the computational cell into K subdomains Ω_k $(k = 1, \ldots, K)$, each of which is responsible for n_k particles. In their algorithm, each of these subdomains is then assigned to one of the processing units present in the parallel machine. The rates of events taking place in each subdomain are calculated, including *null-event functions* $r_{0,k}$, i.e. the probability that an event will not take place in subdomain k. A uniform random number is produced per subdomain and used to determine which event will take place in each one of them. Assuming that no null-event was chosen in subdomain k, the event is undergone and the particle system is updated globally. Communication between processors must take place, since it is possible that an event taking place in subdomain k might affect the state of subdomain k'. This is the case, for instance, for particle migrations from subdomain k to subdomain k' or interactions across subdomain boundaries. Global time is advanced and the whole process is repeated until the final simulation time t_{end} has been reached.

Space decomposition in this algorithm is, in principle, arbitrary, so that the user is free to choose the desired implementation. Clearly, a space division intended to minimise $r_{0,k}$ for most subdomains is optimal [184], but this may not be a straightforward task and depends on the problem. A priori, a simulation box with N particles that are approximately uniformly distributed may be divided into K equally sized subdomains and implemented on a parallel machine with K processing units, with one unit being responsible for each subdomain. It follows that there would be approximately $\langle n_k \rangle = \frac{N}{K}$ particles per subdomain on average, so that the size of the problem to be solved by each processor decreases linearly with K.

Unfortunately, Martínez et al., who implemented this algorithm on a parallel CPU cluster, found a sublinear computational gain which would peak at $K = 64$. The main issue slowing down simulations was found to be the communication overhead between processors, which takes up too much runtime for the simulations to be efficient. This is a well-known issue in parallel programming [4, 125, 182, 258], as mentioned in subsection 3.1.1, so, whereas Martínez et al.'s approach is an encouraging one, we found that a different perspective is required in order to exploit parallel programming as a tool to accelerate OKMC.

We shall now present our solution, which is inspired by Gillespie's Tau-leaping algorithm [119] and based on parallel programming.

4.2 Our approach: parallel Tau-leaping algorithm

In this section, our proposal to accelerate the KMC method is discussed. First, the Tau-leaping method, which serves as inspiration for our work, is detailed. Afterwards, the parallel Tau-leaping algorithm (PTLA) developed during this work is presented, with specific emphasis given to the selection of the time step δt.

4.2.1 The Tau-leaping method

The Tau-leaping method described by Gillespie [119] is used to accelerate OKMC models without a significant loss in accuracy. Instead of calculating the time step δt as described in Eq. 4.11, a fixed time step τ is set and the system is advanced in concordance.

Let us describe the history of the system from $t = 0$ to $t = t_{end}$ as a collection of consecutive instants $t_0 = 0, t_1, t_2, \ldots, t_{end}$ such that the time interval between two consecutive instants $t_{\alpha-1}$ and t_α is, for the sake of simplicity, a constant τ larger than the standard δt. Then, as described by Gillespie [119], if it were possible to determine the number of times a certain event was undergone within said interval, it would not be necessary to know the exact moments when each of the events were carried out. Under these conditions, it would be possible to explore the system evolution in time steps of size τ rather than δt, which implies a smaller amount of computational steps than the standard, due to the condition $\tau > \delta t$. This is certainly an interesting possibility to save runtime. As described in the original article [119], the value for τ must be large enough to accelerate computation but small enough to preserve accuracy; in other words, the system is not expected to change significantly between instants t and $t + \tau$.

By definition, the number of events of a Poisson process to occur within an interval is a Poisson random variable $X \sim \mathcal{P}(\lambda)$ [119]. Recalling Eq. 4.2, we know that the Poisson probability mass function with parameter λ has the form:

$$P(n; \lambda) = \frac{\lambda^n}{n!} \exp(-\lambda) \tag{4.12}$$

where n is the number of occurrences of the event and λ is the average number of occurrences expected during a given time interval.

In particular, the average number of times that an event j with rate r_j shall be triggered during an infinitesimal dt is $r_j \cdot dt$. If τ, though certainly not infinitesimal, is small enough to estimate that the system does not change notably before and after the interval (a situation called the *leap condition* of the Tau-leaping method), the mean number of times that event j is going to fire between t and $t + \tau$ is approximately $r_j \cdot \tau$. With this in mind, we can substitute the parameter λ by $r_j\tau$ and write:

$$P\left(n; r_j\tau\right) = \frac{\left(r_j\tau\right)^n}{n!} \exp\left(-r_j\tau\right) \tag{4.13}$$

Therefore, if τ has got a fixed value, it is possible to estimate the number of occurrences of event j in time interval τ, which we call N_j. In fact, this number is a Poisson random variable: $N_j \sim \mathcal{P}\left(r_j\tau\right)$. After the system has been updated, it is advanced in time by τ and the process is repeated. Certainly, the choice of a suitable value for τ is the main challenge behind this approximation, which is discussed by Gillespie in detail [119]. An appropriate τ may help us investigate larger than usual systems in achievable runtimes accurately. In principle, the largest τ that satisfies the leap condition is the obvious choice for the time step.

4.2.2 Description of our parallel Tau-leaping algorithm

Starting from the main idea behind the Tau-leaping algorithm described in the previous section, we developed a parallel Tau-leaping algorithm (PTLA), which can be found in our article [146].

Similarly to the Tau-leaping method, the time step in the PTLA is imposed beforehand. In our approach, the time step τ is evaluated taking into account the rates inside the system and the rate of external events. Then, the number of events happening in the system is calculated according to a Poisson law. However, there is a difference between our work and Gillespie's Tau-leaping method. In the original article [119], particles of the same type are treated as a collective, and the evolution of their population is tracked. Instead, the PTLA treats particles as individuals, each of which is potentially capable of performing an event during the time step. Instead of attempting a space division like Martínez et al., as mentioned in section 4.1.3, each particle in the system is treated separately in order to ensure a higher degree of parallelism.

We shall now describe the PTLA step by step. After time has been initialised, the following operations are performed:

1. For each particle p_i able to undergo different events with rate $r_{k,i}$, the sum of event rates is computed, as described by:

$$R_i = \sum_{k=1}^{K_i} r_{k,i} \tag{4.14}$$

   with K_i being the total number of events that particle p_i can perform. All sums of event rates are computed in parallel, due to the fact that the calculation of R_i is independent of particle p_j if $j \neq i$.

2. The maximum rate R_{max} out of the R_i of all particles is determined in a parallel manner:

$$R_{max} = \max\{R_i\} \tag{4.15}$$

3. Let us assume that the rate of external events (such as irradiation flux) is R_{ext}. Then, the time step is calculated as a function of R_{max} and R_{ext}:

$$\delta t = f(R_{max}, R_{ext}) \tag{4.16}$$

4. For each particle p_i, the number of times $N_{k,i}$ that each event with rate $r_{k,i}$ will occur during the time step δt is determined using the Poisson distribution:

$$P(N_{k,i}; r_{k,i}\delta t) = \frac{1}{N_{k,i}!}\left(r_{k,i}\delta t\right)^{N_{k,i}} \exp(-r_{k,i}\delta t) \tag{4.17}$$

   This operation is applied to all particles in parallel.

5. Analogously, the number N_{ext} of external events to happen in the system is chosen using a Poisson law $\mathcal{P}\left(R_{ext}\delta t\right)$.

6. Particles undergo the selected events in parallel, provided $N_{k,i} > 0$. External events are performed as well if $N_{ext} > 0$.

7. The surrounding of particles that have moved is analysed. This applies to particles that have migrated, dissociated from a cluster or belong to a cascade that was introduced to the system in this time step.
8. Particles that lie within a specified distance interact, provided interaction is allowed.
9. Time is advanced by δt.
10. Repeat steps 1 to 9 until the final simulation time t_{end} has been reached.

As well as in the serial Tau-leaping algorithm shown in subsection 4.2.1, in the PTLA it is important to find a suitable time step to keep a sensible balance between accuracy and performance. This question shall now be investigated.

4.2.3 Selection of the time step

A possibility to choose τ in an intelligent way is suggested in Gillespie's original article [119]. First, $a_j = r_j h_j$ is defined as the 'propensity function' of event j, with r_j being the event rate associated with it and h_j the number of particles that can perform it [118]. Then, the criterion to decide whether the system changes significantly in a time interval between t and $t+\tau$ is that the change in a_j before and after the interval is not noticeable, for any j. Taking into account all the particles that can perform event j, the average number of times that it is triggered in this interval is given by $a_j\tau$. Therefore, it is possible to compute the expected change Δa_j from t to $t+\tau$ in each function a_j easily. These values Δa_j are then compared to a fraction of $\sum_j a_j$ and the largest value of τ that satisfies this condition for all τ is taken. Compared to the standard BKL algorithm, the number of computational steps was reported to be reduced by two to three orders of magnitude [119].

Our approach is different though. Since the PTLA does not treat particles as collectives but as individuals, our work does not consider propensity functions. Instead, the maximum sum of event rates R_{max} and the rate of external events R_{ext} are used to determine δt, as shown in step 3 of the algorithm described above. Specifically, the function 4.16 chosen to calculate δt is inversely proportional to R_{max}. As explained in subsection 4.1.2, this allows the time step to adapt to the situation of the particle system dynamically. In the absence of continuous irradiation, Eq. 4.18 is the function used for the calculation of δt in our model:

$$\delta t = \frac{\omega}{R_{max}} \tag{4.18}$$

where $\omega > 0$ is a constant.

However, if the sample is under irradiation, the external rate of irradiation R_{ext} must be taken into account. Indeed, if we consider a sample which is initially without any defects, Eq. 4.18 without correction would lead to an infinite time step.

In addition, we should take into consideration the case when irradiation is particularly intense, such that R_{ext} is very large. In case that the rate of external events is much larger than the largest sum of event rates of the particles that are inside the simulation box, i.e. if $R_{ext} >> R_{max}$, using Eq. 4.18 may lead to inconsistencies, as we will explain now. If δt is computed using this equation, the average number of cascades introduced at each time step is $R_{ext} \cdot \frac{\omega}{R_{max}}$, as seen in step 5 of the algorithm detailed in the previous subsection. Therefore, if $R_{ext} >> R_{max}$, this number is very large, and many cascades (and therefore many particles) enter the simulation box during the time step. It may happen that several of these particles display a higher sum of event rates than the current R_{max}, such that many events may take place in the box in the interval between the insertions of one cascade and the next cascade. In practice, all selected cascades are introduced simultaneously, so in this case, the evolution of the system is rendered inaccurate. Obviously, this inaccuracy is more pronounced if the total number of particles entering the simulation box that fulfil this condition is high. However, since the cascades are chosen randomly, there is no possibility to know a priori which type of particles enter the sample at each δt.

One way to mitigate this problem is to restrict the average number of cascades that may be introduced at each time step. By introducing a constant $C > 0$ that sets an upper limit on the average number of cascades per time step, $C \cdot \frac{\omega}{R_{ext}}$ gives the average number of cascades introduced in a time interval equal to $\frac{\omega}{R_{ext}}$. This number can then be compared to $\frac{\omega}{R_{max}}$. If the latter is larger than $C \cdot \frac{\omega}{R_{ext}}$, the limit to the average number of introduced cascades is triggered, and the time step is calculated using the expression $\delta t = C \cdot \frac{\omega}{R_{ext}}$. Otherwise, we can consider that continuous irradiation is weak enough to guarantee that the computation of δt using Eq. 4.18

gives accurate results. Summing up, the time step in the PTLA is calculated as:

$$\begin{aligned} \delta t_{PTLA} &= \min\left\{C \cdot \frac{\omega}{R_{ext}}, \frac{\omega}{R_{max}}\right\} \quad (R_{max} > 0) \\ \delta t_{PTLA} &= C \cdot \frac{\omega}{R_{ext}} \quad (R_{max} = 0) \end{aligned} \tag{4.19}$$

Trivially, setting $R_{ext} = 0$, $R_{max} > 0$ yields Eq. 4.18.

Eq. 4.19 shows that the time step calculated by the PTLA is independent of the number of particles in the system. To highlight this advantage, let us examine a simple example from the perspective of the models described in this chapter, namely the BKL algorithm [46], the Tau-leaping method [119], and Martínez et al's parallel model [184]. We shall apply these models to a simulation box with N_{part} equal particles able to perform one event with rate r and compare their behaviour with that of the PTLA.

In the standard BKL algorithm [46] described in subsection 4.1.1, the total rate of this system is simply $R = N_{part} \cdot r$. Considering the average of the random exponential distribution, and using Eq. 4.11, the time step in this case yields:

$$\delta t_{BKL} = -\frac{\ln(\eta)}{R} = -\frac{\ln(\eta)}{N_{part} \cdot r} \approx \frac{1}{N_{part} \cdot r} \tag{4.20}$$

One of the particles is chosen, which performs the event, while the rest (i.e. $N_{part} - 1$ particles) are idle during the time step. If N_{part} increases, the total rate R is increased and δt is decreased. Thus, many computational steps are necessary if this system comprises millions of particles.

If the Tau-leaping method [119] is used instead, the time step τ is about two to four orders of magnitude times larger than δt_{BKL}. Gillespie was able to simulate the irreversible isomerization reaction accurately with a set of 10^5 molecules in only 305 computational steps, whereas a simulation with the BKL algorithm would need 10^5 steps [119]. In this case, the propensity function is calculated as $a_k = N_{part} \cdot r$. If this event changes the nature of the particles involved (such as a dissociation or, in the context of Gillespie's work, a chemical reaction), the change in the propensity function between times t and $t + \tau$ is calculated as:

$$\Delta a_k = (N_{part} - N'_{part}) \cdot r \tag{4.21}$$

where N'_{part} represents the number of particles for which the event does not trigger during time interval τ. To satisfy the leap condition, $\frac{\Delta a_k}{a_k}$ must be small [118, 119]. As mentioned above, the leap condition is usually satisfied when τ is about two orders of magnitude larger than δt_{BKL}. Since δt_{BKL} decreases when N_{part} is increased, this is also the case for τ in the Tau-leaping method.

Finally, in the Martínez et al.'s work [184], the simulation box is divided into K subdomains. Clearly, if there is a higher concentration of particles in one of the subdomains than in the others, this is the subdomain that defines R_{max}. In the optimal case, R_{max} is minimised so that the time step $\delta t_{Mart.}$ is maximised. To do this, each of the subdomains must contain exactly $\frac{N_{part}}{K}$ particles. Then, the rate in each subdomain k is $R_k = \frac{N_{part}}{K} \cdot r$, such that $R_k = R_{max}$. Hence, the time step in the best-case scenario verifies:

$$\delta t_{Mart.} = -\frac{\ln(\eta)}{R_{max}} = -\frac{\ln(\eta)}{\frac{N_{part}}{K} \cdot r} \approx \frac{K}{N_{part} \cdot r} \tag{4.22}$$

For a network of $N = 64$ processors, a realistic configuration [184], and a SIA migration rate of $r = 2.7 \times 10^4\ s^{-1}$ in α-Fe at $T = 200\ K$ [105, 31, 146], this yields a time step of $\delta t_{Mart.} = 2.4 \times 10^{-7}\ s$ for a system with $N_{part} = 10000$ particles and $\delta t_{Mart.} = 2.4 \times 10^{-8}\ s$ for a system with $N_{part} = 100000$ particles. Again, the time step is inversely proportional to the number of particles in the simulation box with proportionality constant $\frac{K}{r}$, assuming the density of particles is the same in all subdomains. In general, particle density varies across subdomains. There is usually a higher concentration of particles than $\frac{N_{part}}{K}$ in at least one of them, yielding a larger value for R_k. As a consequence, the time step δt is decreased, such that a greater computational effort to finish the simulation is necessary. In this model, K events are carried out simultaneously at most, assuming that no null-events were chosen. However, in the general case, there is a nonzero probability to choose a null-event in each of the subdomains for which $R_k < R_{max}$.

In practice, the time step in all of the three mentioned methods is decreased by the number of particles in the system. This makes those algorithms impractical to use for systems with several hundred thousand to millions of particles, which, as we discussed in subsection 4.1.2, are closer to the experimental size of grains in irradiated materials that this work intends to simulate.

However, our PTLA is not constrained by these conditions. Instead, all particles are considered individually capable of performing events during the time step, the probability of which is calculated in Eq. 4.17. In the present example, $R_{max} = r$ and $R_{ext} = 0$. Hence, Eq. 4.19 yields:

$$\delta t_{PTLA} = \frac{\omega}{r} \tag{4.23}$$

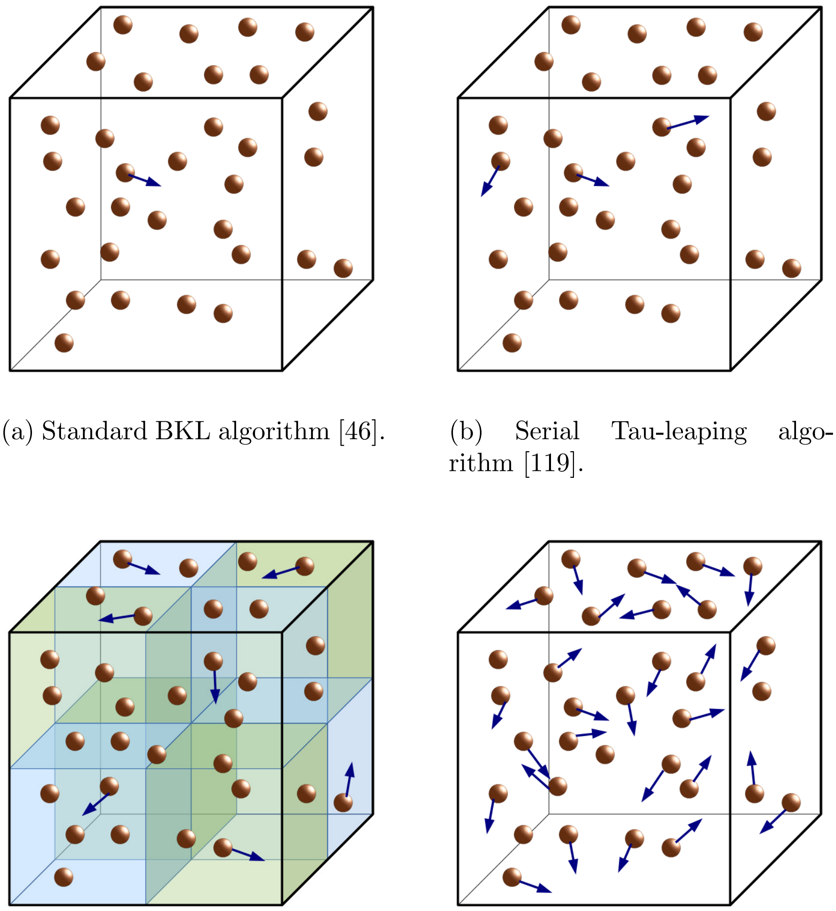
The PTLA time step δt_{PTLA} is thus $(\omega \cdot N_{part})$ times larger than δt_{BKL} and at least $\left(\omega \cdot \frac{N_{part}}{K}\right)$ times larger than $\delta t_{Mart.}$. In the case $\omega = 1$, our model reduces the number of computational steps by a factor of N_{part} compared to the standard BKL algorithm, and by a factor of $\frac{N_{part}}{K}$ or greater compared to Martínez et al.'s model. In other words, for systems with several hundred thousand or millions of particles, the time step achieved with the PTLA increases by five or more orders of magnitude compared to δt_{BKL}. This strongly reduces the number of computational steps that are necessary to reach a specified physical time, which speeds up simulations significantly. This indicates that the PTLA is a promising technique to simulate particle systems which are representative of real samples.

The behaviour of the PTLA compared to the other three methods is sketched in Figure 4.1. All particles are assumed equal and able to perform a migration with rate r. If $\omega = 1$, in our model, the time step takes the value $\delta t_{PTLA} = \frac{1}{r}$. Then, the number of times that a particle i is going to migrate is given by $N_{mig,i} \sim \mathcal{P}(1)$. In other words, even though individually, each particle performs a number of migrations during δt_{PTLA} that follows a Poisson distribution, the average number of migrations per particle when all particles are considered is 1, regardless of the total number of particles. For this reason, it seems natural to set $\omega = 1$ as the standard, and this will be the case in forthcoming examples unless otherwise stated.

Having described the PTLA used in this work and its potential advantages, we shall now focus on a number of test cases of relevance in materials physics.

4.3 Validation of the parallel Tau-leaping algorithm

In this section, some results produced by our implementation of the PTLA on the programming framework CUDA version 6.5 are presented. Since our OKMC model



(a) Standard BKL algorithm [46].

(b) Serial Tau-leaping algorithm [119].

(c) Martínez et al. algorithm [184].

(d) PTLA [146].

Figure 4.1: Sketch of the four methods mentioned in section 4.2: serial BKL [46], serial Tau-leaping [119], Martínez et al. [184] and PTLA [146].

was implemented on GPGPU, it is referred to as GPU-OKMC. The device used for this purpose was a GPU NVIDIA®GeForce®TITAN Black with 6 GB of global memory. These results were published in Comp. Mat. Sci. **113** 178-186 (2016) [146].

4.3.1 Test case: Random walks of non-interacting particles

Many processes in materials physics are governed by the random walks of defects or impurities in solids [229, 248, 304]. The test case presented in this subsection is a simple example related to this problem. In order to validate our GPU-OKMC model, we simulated the random walk of particles that interact neither with each other nor with any surface. In this particular case, it is possible to compare simulations to analytical results, which is an appropriate way to validate our results.

As an example, we considered the case of particles diffusing in pure Fe. This is a system of interest in fission and future fusion reactors as these defects form by collision cascades in steels irradiated by energetic neutrons [98, 234, 267, 304]. First, we consider the evolution of five million non-interacting random walkers in Fe, in a $(36a_0)^2 \times 216a_0$ simulation box. Initially located in a plane $z = \frac{z_{max}}{2}$ with random $0 \leq x < x_{max}$ and $0 \leq y < y_{max}$ coordinates, the particles are allowed to perform one-dimensional random walks in z direction for a time interval from $t = 0$ to $t = t_{end}$. In principle, after $t = t_{end}$, if the total number of jumps is sufficiently high, the profile in z direction describes a Gaussian function in agreement with Fick's laws of diffusion [67, 229].

If a particle performs enough jumps by means of random walks, a macroscopic diffusion coefficient D may be defined [67, 229]. For particles migrating in n dimensions with migration energy E_m at temperature T, the diffusion coefficient can be expressed by Eq. 4.24:

$$D = \frac{\nu_0 \lambda^2}{2n} \exp\left(-\frac{E_m}{k_B T}\right) \tag{4.24}$$

where ν_0 and λ represent the attempt frequency and the jump length, respectively. In the case of diffusion in one dimension, $n = 1$. The time-dependent concentration per unit length of particles migrating in one dimension is given by Eq. 4.25 [67, 229].

$$\rho(x,t) = \frac{N}{\sqrt{4\pi D t}} \exp\left(-\frac{(x - x_0)^2}{4Dt}\right) \tag{4.25}$$

where N is the initial number of particles at $x = x_0$.

Considering this, simulations were performed at three different temperatures $T = 130\ K$, $T = 135\ K$ and $T = 140\ K$ for $t_{end} = 30\ s$ and compared to equivalent Gaussians.

The values $\nu_0 = 10^{13}\ s^{-1}$ for the attempt frequency and $E_m = 0.34\ eV$ for the migration energy of walkers were used, which corresponds to the parameters related to SIA migration in pure Fe found in literature [105]. The results are shown in Figure 4.2 and show excellent agreement with the analytical solution.

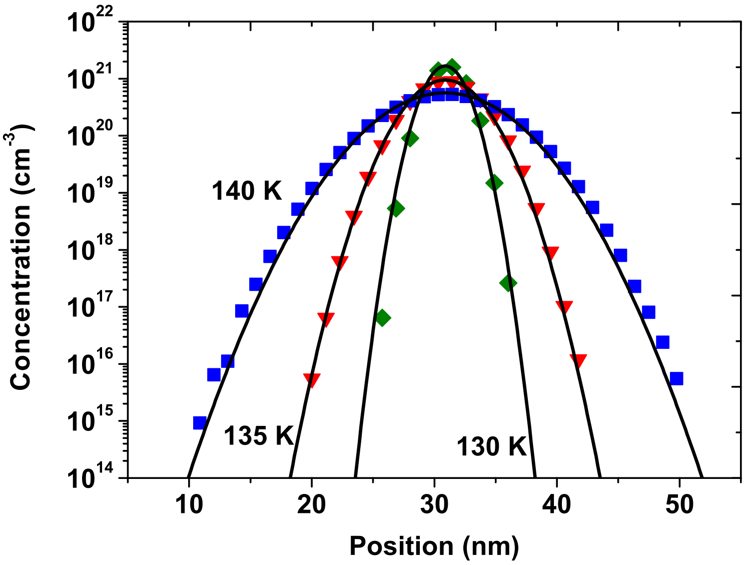


Figure 4.2: Profiles at $t = 30\ s$ of 1D random walkers in pure Fe at three different temperatures, initially located in a plane $z = \frac{z_{max}}{2}$. The results of this code (symbols) are compared to those obtained with the analytical expression based on Fick's laws of diffusion (lines).

In order to compare performance of this GPU-OKMC model over a hypothetical solution on one single processor (a *classic OKMC* solution), a similar case described by Martínez et al. [184] was simulated. In their article, Martínez et al. investigated the evolution of 2^{22} (about 4 million) random walkers. Using 64 parallel processors, a speedup of circa 30× over the standard BKL on a single processor was achieved. Here, we simulated the migration of four million SIAs at a temperature of 135 K for 10 s. The same test case was run on a CPU AMD Opteron 6174@2.2 GHz using code LAKIMOCA [88] version o128p. LAKIMOCA is an established code in the scientific community based on the standard BKL algorithm. Both codes, the present GPU-OKMC model and the *classic* LAKIMOCA model, achieved the same result in 1.9 s and 563 s respectively. This corresponds to a remarkable speedup of 296×. Encouraged by this result, more complex test cases that were simulated with our GPU-OKMC code are discussed in the next subsections.

4.3.2 Test case: Diffusion of non-interacting particles in the presence of absorbing surfaces

In irradiated materials, point defects like SIAs and vacancies that are created in irradiation cascades tend to recombine when they migrate to the surface [129, 304, 315]. For this reason, it is interesting to examine the diffusion of defects in the presence of an absorbing surface.

In the present work, the evolution of five million initial SIAs in Fe, randomly distributed in a $(400a_0)^3$ simulation box at the beginning, was simulated. This corresponds to an initial density of $\rho = 3.34 \times 10^{21}\ cm^{-3}$. Then, particles were allowed to perform random walks in three dimensions during 200 s for three different temperatures, namely 130 K, 140 K and 150 K. Absorbing boundary conditions were set for planes $z = 0$ and $z = 400a_0$, whereas periodic boundary conditions were assumed in x and y directions. The values for ν_0 and E_m were set equal to those assumed in subsection 4.3.1.

In order to validate our GPU-OKMC model, simulations with classical diffusion equations and Dirichlet boundary conditions were also performed. The time evolution of the number of particles under the aforementioned conditions was monitored and depicted in Figure 4.3 along with the results obtained with diffusion equations.

The agreement between our GPU-OKMC model and the results obtained with diffusion equations is excellent. Similarly, Figure 4.4 shows the depth profiles of random walkers under these conditions, which are also found to agree very well with the depth profiles calculated with the help of diffusion equations. Hence, our GPU-OKMC model reproduces the recombination of randomly walking particles at absorbing surfaces accurately.

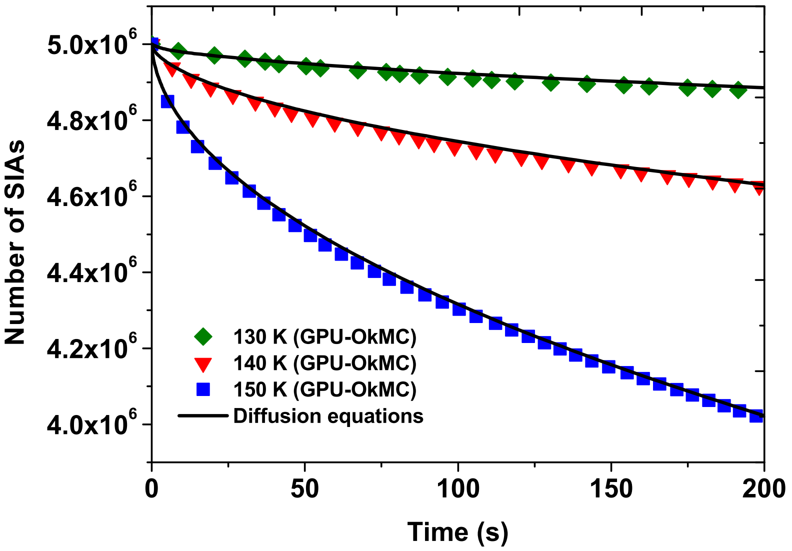


Figure 4.3: Time evolution of the total amount of Fe interstitials for 200 s at three different temperatures in the presence of absorbing surfaces at $z = 0$ and $z = 400a_0$. Results of GPU-OKMC simulations (symbols) are compared to those obtained with diffusion equations (lines).

4.3.3 Test case: Recombination of interacting particles

In this subsection, we shall examine a relevant case of interest, namely the evolution of a system of particles of different types that annihilate each other, as the evolution of many systems is governed by the diffusion-limited reaction $A + B \to AB$ [72, 128, 146, 177]. For instance, this process is commonly found in materials relevant for nuclear fusion like the agglomeration of helium and vacancies in W [32] or in Fe [107, 214, 288], and the nucleation of point defect clusters in α-Fe [131, 180, 289]. As a practical example to test the accuracy and performance of our GPU-OKMC model, we investigated the annihilation of Frenkel pairs in Fe, i.e. $I + V \to 0$. These point defects form in Fe under irradiation [16, 105, 237, 275] by atomic displacement cascades –and, more generally, in any crystal material under irradiation– and tend to annihilate when they are close enough.

For vacancies, the value found in literature for the migration energy is $E_m = 0.67$ eV [105, 175, 214], and the values for SIA migration energy and attempt fre-

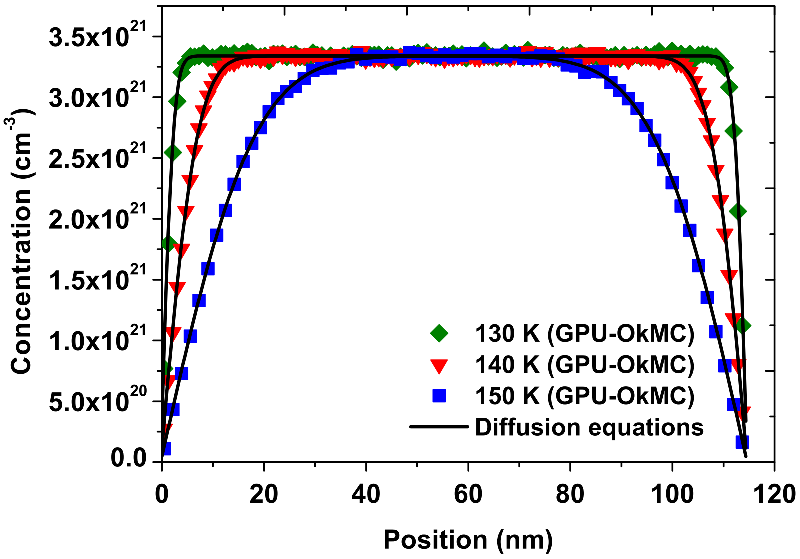


Figure 4.4: Depth profiles of Fe interstitials in z direction for three different temperatures in the presence of absorbing surfaces at $z = 0$ and $z = 400a_0$. Results of GPU-OKMC simulations (symbols) are compared to those obtained with diffusion equations (lines).

quency were set equal to those used in subsections 4.3.1 and 4.3.2. Recombination between SIAs and vacancies takes place within a capture radius instantaneously, as described in subsection 2.2.2, i.e. the reaction takes place if the distance between both particles is less than the defined capture radius for this reaction. A value of $r_c = 3.3a_0$ was used for the recombination radius between SIAs and vacancies, as accepted by the community [105, 175, 214].

The time evolution of 4 million SIAs and 4 million vacancies in a simulation box of $(200a_0)^2 \times 70000a_0$ was monitored. This corresponds to an initial Frenkel pair density of $\rho = 6.11 \times 10^{19}\ cm^{-3}$ in a simulation box of 20 μm depth, which is already in the order of grain sizes and close to material thickness in some realistic physical experiments. In order to ensure that particles may only disappear by recombination, periodic boundary conditions were set in all directions. Then, simulations were performed for 500 s at three different temperatures (130 K, 135 K and 140 K) and compared to results obtained with LAKIMOCA [88] version o128p, a *classic* OKMC. For this case, the GPU-OKMC simulations were produced with a NVIDIA®GeForce®TITAN Black graphics card, whereas LAKIMOCA simulations were performed on a CPU AMD Opteron 6174@2.2 GHz. A value $\omega = 1$ was used to reproduce these results in the GPU-OKMC model. Figure 4.5 depicts the time evolution of SIAs as calculated by both GPU-OKMC and LAKIMOCA models.

The agreement between both approaches is clearly excellent, which again validates the accuracy of our GPU-OKMC model.

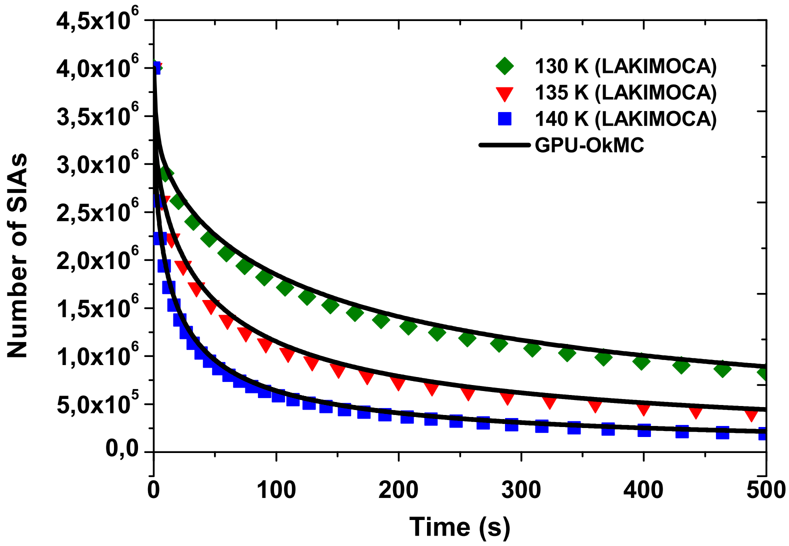


Figure 4.5: Time evolution of 4 million SIAs at three different temperatures. Results of GPU-OKMC simulations (lines) are compared to those obtained with LAKIMOCA (symbols), courtesy of C. Domain.

4.3.3.1 Speedup and accuracy

We have shown that the results for the case of recombination of particles are accurate for $\omega = 1$. Still, it might be useful to check if other values for ω might as well lead to satisfactory results. As one can see in Eq. 4.18, the δt value is proportional to ω. Therefore, choosing a larger value for ω leads to larger time steps, which might reduce the necessary computational effort to achieve simulations. However, choosing a time step that is too large could affect the physical accuracy of the simulation, which means that a compromise is necessary. We shall now discuss the impact of ω on accuracy and performance.

Due to the large values that the time step can take in our algorithm, there exists a significant probability that a particle undergoes the same event several times during the same time step. Assuming $\omega = 1$ and a system of 2×10^5 particles, which corresponds to a density of $\rho = 6.84 \times 10^{19}\ cm^{-3}$ in a $(500a_0)^3$ simulation box, the probability that one particle will perform a certain number of events is shown in Figure 4.6. As one can see, in the standard BKL algorithm, the probability that a particle undergoes no event at all during a time step is close to 1, whereas the probability that a single particle of the whole system undergoes one event is N_{part}^{-1}, that is, five orders of magnitude smaller. Furthermore, the probability that a particle undergoes the same event n times during the same time step is proportional to N_{part}^{-n}, which explains why, in the standard BKL algorithm, it can be safely assumed that one particle in the whole simulation box will perform only one event at each time step.

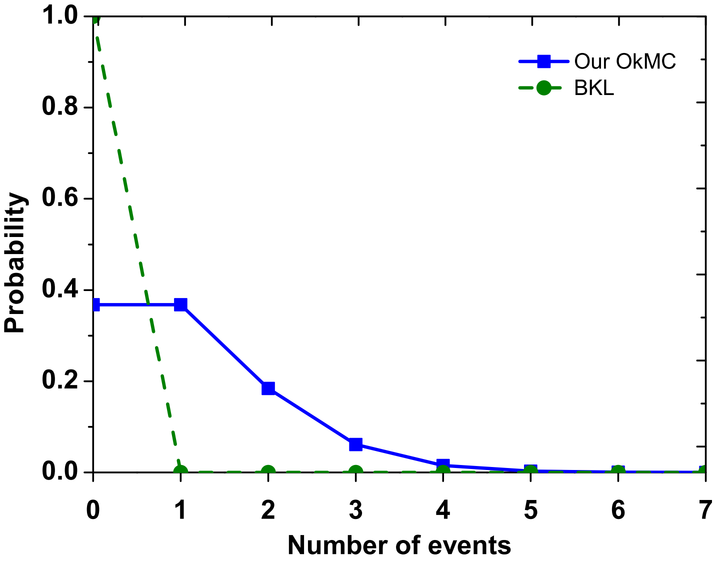


Figure 4.6: Probability that a particle performs a given number of events during a time step according to the PTLA (squares) and to the BKL algorithm (circles).

Clearly, this is not the case in the PTLA. The probability that each particle carries out at least one event in the same time step is relatively high, as seen in Figure 4.6. Setting $N_{k,i} = 1$ and $r_{k,i}\delta t = 1$ in Eq. 4.17 yields the probability that a particle will perform exactly one event, namely $P(1) \approx 0.37$. This means that a large number of events can take place simultaneously in the system. Figure 4.6 also evidences that the probability that a particle performs more than one event during the same time step is not negligible. With the help of Eq. 4.17, it is trivial to compute that the probability to perform more than one event in δt, considering $r_{k,i}\delta t = 1$:

$$P(N_{k,i} > 1) = 1 - P(0) - P(1) \approx 0.26 \tag{4.26}$$

However, due to the fact that the surrounding of particles is only analysed at the end of each time step, in several instances, this may affect the accuracy of the solution. For example, if a particle is chosen to undergo N_m migrations in the same time step, its surrounding is only analysed after the N_m^{th} event, and hence, possible interactions are neglected on $N_m - 1$ occasions. This might affect accuracy at large concentrations for which the distance between particles can be relatively small.

In order to achieve a physically realistic approximation, this occurrence, which is also an issue in Gillespie's Tau-leaping method [119] discussed in subsection 4.2.1, ought to be minimised. One possible solution is to choose a small value for ω, and therefore an arbitrarily small time step, such that the probability that more than one event is carried out for the same particle in the same time step is attenuated. To illustrate this point, in Figure 4.7 we plotted this probability for values of $\omega \leq 1$ against the given number of events.

Evidently, this figure shows that the probability that an event will be carried out once or more than once decreases for smaller values of ω, i.e. for smaller values of δt. For the curve associated with the smallest value considered ($\omega = 0.1$), the probability that a particle will carry out two events during the same time step is already negligible. Nevertheless, a small $\omega < 1$ value such as $\omega = 0.1$ implies that a larger number of computational steps will be needed to achieve the final simulation time t_{end}. With this in mind, we should be able to find a sensible compromise between accuracy and performance.

In order to examine the influence of the particle density and choice of time step interval on accuracy and speedup, several simulations were performed for $T = 135\ K$ by varying values for ω, ranging from $\omega = 0.1$ to $\omega = 5$, and initial Frenkel pair

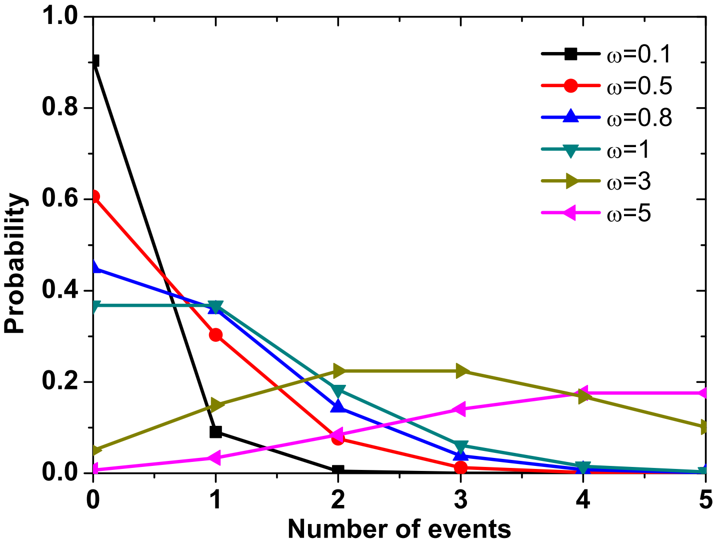


Figure 4.7: Influence of ω on the probability that a particle performs a number of events during the same time step, for values $\omega \leq 1$.

density, ranging from $\rho = 6.11 \times 10^{17}\ cm^{-3}$ to $\rho = 6.11 \times 10^{20}\ cm^{-3}$. In order to perform simulations for different initial densities while keeping good statistics, the initial total number of Frenkel pairs (4 million) was kept constant and the simulation box volume was changed. Then, the gain in performance was compared to simulations under the same conditions performed with LAKIMOCA.

The time evolution of SIAs obtained with both models is depicted in Figure 4.8. As we can see, the agreement is excellent in all cases, as only a small deviation of the results is observed for the case with the largest time step ($\omega = 5$), in the cases for medium and high density. This is expected, since for large values of ω, i.e. for large values of the time step, particles are able to perform several jumps in the same time step, as seen in Figure 4.7.

Due to the fact that interactions are only checked at the end of each time step, it is possible that some interactions are missed in such cases. However, in the test cases presented here, a good compromise between performance and accuracy for values of $\omega > 1$ was found.

Table 4.1 shows the speedups achieved by our code based on the PTLA over the

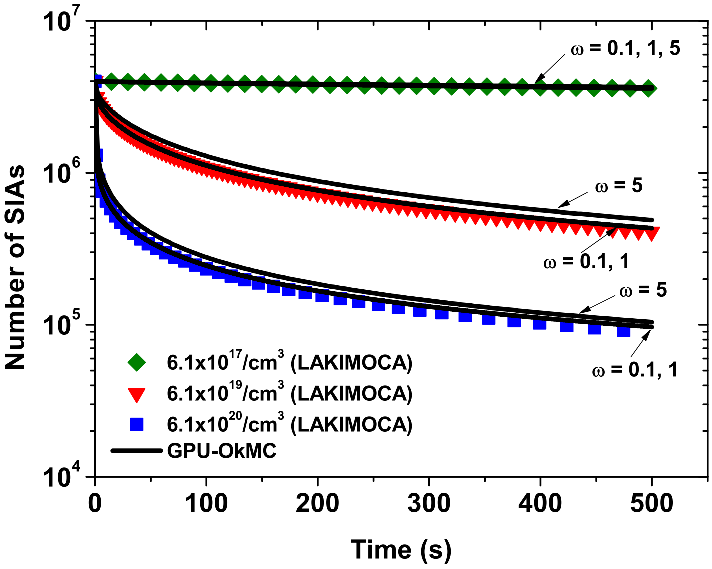


Figure 4.8: Evolution of the number of SIAs in time at $T = 135\ K$ for different initial densities and values of ω. GPU-OKMC results (lines) are compared to those obtained with LAKIMOCA (symbols), courtesy of C. Domain.

LAKIMOCA simulations based on the *classic* BKL algorithm. As expected, better speedups are obtained for larger values of ω. Indeed, larger values of the time step imply less computational steps and hence, shorter simulation runtimes.

ω / $\rho\,(cm^{-3})$	0.1	0.5	1.0	2.0	5.0
6.1×10^{17}	12.9×	51.6×	88.4×	157.7×	375.6×
6.1×10^{19}	4.5×	17.8×	32.2×	51.1×	115.6×
6.1×10^{20}	3.3×	12.7×	20.5×	33.3×	62.9×

Table 4.1: Speedup achieved with our GPU-OkMC algorithm for different particle densities and values of ω in comparison to LAKIMOCA runtimes for the case $T = 135\ K$.

Additionally, in order to distinguish between the gain achieved by the PTLA from the one achieved by the implementation, we performed additional simulations where the standard BKL algorithm was used on the GPU. To do so, we calculated the time step corresponding to the BKL algorithm, as defined in Eq. 4.11. This leads to a much smaller time step and consequently, on average, only one event is carried out per time step, as it is expected in the BKL algorithm. As an example, we performed a simulation at 130 K for a time of 10 s in a volume of $200a_0 \times 200a_0 \times 700a_0$ with 40000 Frenkel pairs, which corresponds to a density of 6.11×10^{20} cm^{-3} for SIAs and vacancies alike. Under these conditions, the time step predicted by the BKL is approximately equal to 3.79×10^{-5} s. To achieve the simulation using the BKL algorithm on the GPU, a runtime of 1493 s was necessary. In contrast, when the PTLA is used (with $\omega = 1$), a time step of 1.51 s is obtained using Eq. 4.18. In this case, only 87 ms were necessary to achieve the final simulation time. This evidences that when both algorithms are used on the GPU, an important speedup of about 17100× is achieved by the PTLA over the *classic* BKL algorithm.

Actually, this is not surprising. As mentioned in subsection 3.1.2, GPUs are inherently able to launch a large number of threads in parallel. In particular, the GPU used in this example, NVIDIA®GeForce®TITAN Black is able to launch approximately 3×10^5 threads in parallel. This means that, on average, approximately 300000 particles can move simultaneously with our implementation of the PTLA. However, if the *classic* BKL is used instead, only one particle is able to move per time step, i.e., about 300000 times fewer particles (five orders of magnitude) than what the PTLA allows. Hence, it is not surprising that we get a speedup of 17100× between the BKL and the PTLA when they are both used on the GPU. The speedup obtained when comparing the performance of our model with the serial code LAKIMOCA is of course not so high since the implementation of the BKL algorithm is likely optimised for a CPU use. Moreover, as mentioned in subsection 3.1.2, GPU processors are significantly slower than CPUs.

Our work accounts for some approximations that are appropriate, and sometimes necessary, for the study of defect evolution. These are addressed in the next chapter.

Chapter 5

Adaptation of Green's Function Reaction Dynamics to the MEGA-OkMC code

In this chapter, the Green's Function Reaction Dynamics (GFRD) method is introduced and its application to our work is explained. Special attention is given to the use of GFRD in diffusion-reaction systems with an external source of particles, and examples related to the study of nuclear materials under irradiation are shown.

5.1 Introduction

In the previous chapter, the PTLA was explained and shown as an efficient OkMC technique to study diffusion-reaction systems. However, despite its advantages, the PTLA, as any OKMC method, becomes highly inefficient when the system is dilute, i.e., when the concentration of species is low. In such systems, the average distance between particles is large and hence, they can evolve without interacting during long periods of time. Consequently, most of the computational effort is wasted in propagating the random walkers and reactions that could lead to some significant evolution of the system occur very rarely. At higher temperature, this is particularly damaging to performance, due to the fact that the computed time step becomes very small.

To circumvent this problem and efficiently simulate the evolution of dilute systems, Van Zon, ten Wolde et al. developed the Green's Function Reaction Dynamics method [292, 293]. It is based on the assumption that a diffusion coefficient can be defined for particles that are able to perform a large number of migrations before finding an obstacle, such as an open surface or another particle with which they may react [293, 304]. In the GFRD method, the space containing the N particles is partitioned into domains that contain one or, at most, two particles. After breaking up the N-body problem into one-body and two-body problems, the reaction-diffusion problem of each domain is solved analytically and the particle positions are updated. This method allows to propagate the particles by large jumps in time and space while keeping the spatial-stochasticity of the diffusion process, and taking into account reactions between particles. The GFRD method has showed to be very efficient in the case of systems where the particles are initially distributed in space and only evolve by random walks and mutual interactions [292, 293].

However, to our knowledge, there is another class of systems where GFRD has not been applied so far, namely systems with an external continuous source of particles. Such systems can be found, for instance, in materials under irradiation. Defects are continuously created in the material due to displacement cascades generated by collisions between energetic particles and the lattice atoms. Simultaneously, the defects formed at a previous time migrate by performing random walks and interact with other defects. As we shall see, applying the GFRD method naïvely to simulate the evolution of such systems gives rise to two problems. In this chapter, we propose a novel scheme that circumvents these issues and allows to simulate the evolution of systems with an external source of particles with the GFRD.

The chapter is organised as follows. In Section 5.2 we remember the fundamental concepts of GFRD, which will allow us to demonstrate why the original GFRD algorithm cannot be applied in the case of systems with a continuous source of particles. In Section 5.3 we present the solution to the two issues mentioned above and describe a novel GFRD scheme that takes into consideration the external source of particles.

All the simulations presented in this chapter were performed on the GALILEO supercomputer at the CINECA supercomputing centre. More specifically, the simulations were performed on a machine with a NVIDIA® Tesla K80 GPU and Intel®

Xeon® E5-2697 v4 processors at 2.30 GHz.

5.2 Description of the original Green's Function Reaction Dynamics method for dilute systems

Before we present our GFRD algorithm adapted to the case of materials under irradiation, we believe it is instructive to remind various concepts and mathematical expressions related to the GFRD method developed by Van Zon, ten Wolde et al [292, 293]. This will allow the reader to understand why the original GFRD method is not well-suited to the case of systems with a continuous source or particles and how it should be adapted.

In an infinity medium where a random walker is free to evolve without any interaction, the probability $p(\vec{r}, t|\vec{r_0})$ of finding the particle at the position vector $\vec{r}$ at time t given that its initial position is $\vec{r} = \vec{r_0}$ at $t = 0$, i.e., with an initial probability distribution $p(\vec{r}, t) = \delta(\vec{r} - \vec{r_0})$, is governed by the classical diffusion equation [67, 231, 248]:

$$\frac{\partial}{\partial t} p(\vec{r}, t|\vec{r_0}) = D\nabla_r^2 p(\vec{r}, t|\vec{r_0}) \tag{5.1}$$

where D is the diffusion coefficient of the particle. Under these conditions, an analytical solution of $p(\vec{r}, t|\vec{r_0})$ can be found and is known as the Green's function [228, 248, 292, 293].

Similarly to an infinite medium, in a dilute system composed of random walkers, the average distance between particles is large and hence, they can evolve without interacting during long periods of time. In such conditions, it is clear that it is significantly more efficient from the computational point of view to determine the probable position of a particle after a time t using the analytical solution of the diffusion equation (Eq. 5.1) than performing many individual jumps or random walks. This is the main advantage of the GFRD method.

However, in a realistic system, though it may be dilute, particles are at a finite distance one from each other and are likely to experience encounters with other particles, leading to a physico-chemical process such as the formation of a new product or the mutual anihilation of the particles. Hence, in this case, Eq. 5.1 is no longer valid and appropriate modifications of the diffusion equation, accounting for reaction with other particles, should be taken into account. Unfortunately, the general diffusion-reaction problem of N interacting random walkers is very difficult, not to say impossible, to solve analytically [292, 268]. Nevertheless, in the particular case where two particles diffuse and interact, isolated from the rest of the other particles, the diffusion-reaction problem can be formulated such that an analytical solution can be found [292, 293]. This suggests that, if a realistic system of N interacting particles can be broken up into independent problems involving only one or at most two particles, the diffusion equation could be analytically solved for each of them. This is precisely the original idea of the GFRD method developed by Van Zon, ten Wolde et al [292, 293]. In the GFRD method, the system is divided spatially into different independent protective environments of simple geometric shape (originally introduced by Oppelstrup and co-workers [211]), which only contain one or two particles. The domains containing only one particle are named *Single* domains whereas domains containing two particles are called *Pair* domains. The size of these protective domains is determined by allowing them to grow geometrically until a contact with another one is found. An illustration of how protective domains are built in practice around isolated particles and pairs of particles is shown in Fig. 5.1. For the sake of simplicity, here we considered that all particles are random walkers that migrate in 3D. Hence, the protective domains in this case are spheres.

5.2.1 Solution for Single domains

Here, the calculations for particles moving in Single domains are detailed. In this case, particles diffuse freely without any interaction inside their protective domain. We shall focus, in particular, on the cases of particles moving in one dimension, like dislocation loops (DL) in Fe [16, 17, 131, 234, 284, 289, 309, 308], and three dimensions, like SIAs and vacancies [33, 105, 284, 304, 313].

The movement of a randomly walking particle follows Bernoulli statistics [67, 229, 248]. Let us first focus on the case of a particle that performs random walks in one dimension along an axis of motion. The probability $W(m, N)$ that the particle starting at the origin will be found at position m (with $m \in \mathbb{Z}$) after N jumps is given by Eq. 5.2.

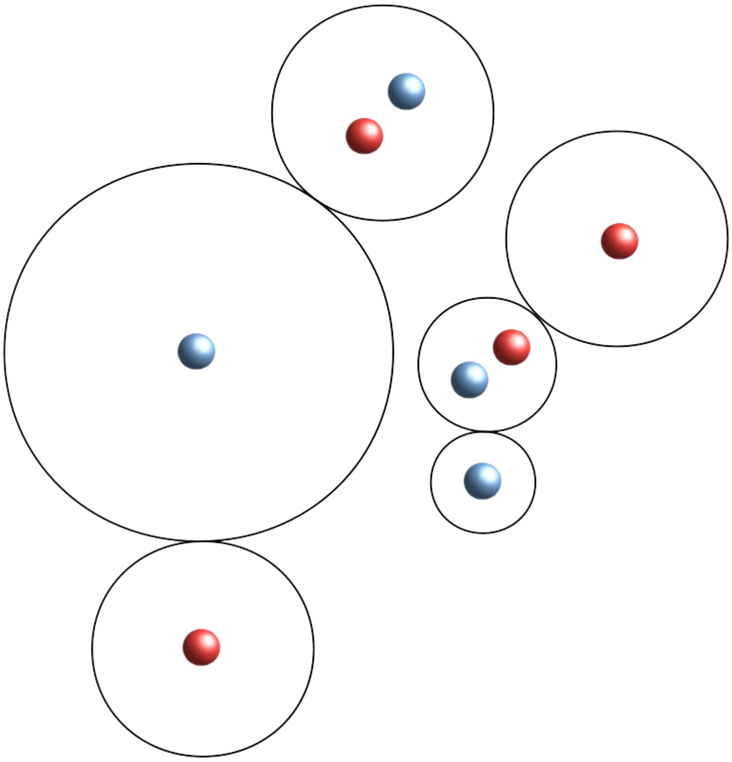


Figure 5.1: Example of spherical protective domains in the case of particles migrating in 3D. Domains containing only one particle are named Singles whereas domains containing two particles are named Pairs.

$$W(m,N) = \frac{1}{2^N}\binom{N}{\frac{N+m}{2}} \tag{5.2}$$

For a large enough number of free jumps, i.e. for a large value of N, this formula can be approximated as Eq. 5.3.

$$W(m,N) = \sqrt{\frac{2}{N\pi}}\exp\left(-\frac{m^2}{2N}\right) \tag{5.3}$$

If λ is the jump length, $x = m\lambda$ yields the total displacement from the origin. Then, the probability $W(x, \delta x; N)$ to find the particle in the interval $[x, \delta x]$ is given by Eq. 5.4.

$$W(x,\delta x;N) = \frac{1}{\sqrt{2\pi N\lambda^2}}\exp\left(-\frac{x^2}{2N\lambda^2}\right)\delta x \tag{5.4}$$

The fact that the number of jumps N takes a large value implies that a diffusion coefficient $D = \frac{N \cdot \lambda^2}{2t}$ can be defined. The diffusion coefficient is thus related to the migration rate $r_{mig} = \frac{N}{t}$, i.e. the average number of jumps that a particle performs per unit time. In particular, in one dimension, the diffusion coefficient is expressed as [67, 229, 248, 304]:

$$D = r_{mig} \cdot \frac{\lambda^2}{2} \tag{5.5}$$

Substitution into Eq. 5.4 yields the Green's function $p(x, t|x_0)$ for the one-dimensional motion of a particle with diffusion coefficient D starting at position x_0, given by Eq. 5.6. This function is a Gaussian probability distribution function with mean x_0 and standard deviation $\sqrt{2Dt}$ [67, 229, 248].

$$p(x, t|x_0) = \frac{1}{\sqrt{4\pi Dt}} \exp\left(-\frac{(x - x_0)^2}{4Dt}\right) \tag{5.6}$$

However, since domains around particles are of finite size, appropriate boundary conditions must be imposed to solve Eq. 5.1. In the common case where particles diffuse in three dimensions, the domain is of spherical shape with radius R and the boundary conditions are simply written as:

$$p(\|\vec{r} - \vec{r_0}\| = R, t|\vec{r_0}) = 0 \tag{5.7}$$

The Green's function for this case is the solution of Eq. 5.1 with boundary conditions given by Eq. 5.7. In Cartesian coordinates, the function $p(\vec{r}, t|\vec{r_0})$ can be rewritten as the product of three independent functions:

$$p(\vec{r}, t|\vec{r_0}) = p_x(x, t|x_0)p_y(y, t|y_0)p_z(z, t|z_0) \tag{5.8}$$

Since $\frac{\partial p_{x_i}}{\partial x_j} = 0$ for $i \neq j$, Eq. 5.1 can be written as:

$$\frac{\partial p(\vec{r}, t|\vec{r_0})}{\partial t} = D\left[\frac{\partial p_x(x, t|x_0)}{\partial x}\frac{\partial p_y(y, t|y_0)}{\partial y}\frac{\partial p_z(z, t|z_0)}{\partial z}\right] \tag{5.9}$$

Integration of Eq. 5.9 yields:

$$p(\vec{r}, t|\vec{r_0}) = \frac{1}{(4\pi Dt)^{\frac{3}{2}}} \exp\left(-\frac{(x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2)}{4Dt}\right) \tag{5.10}$$

which in spherical coordinates yields [229]:

$$p(\vec{r},t|\vec{r_0}) = \frac{1}{(4\pi Dt)^{\frac{3}{2}}} \exp\left(-\frac{(\vec{r}-\vec{r_0})^2}{4Dt}\right) \tag{5.11}$$

In this case, the diffusion coefficient in three dimensions is given by Eq. 5.12 [231, 229].

$$D = r_{mig} \cdot \frac{\lambda^2}{6} \tag{5.12}$$

5.2.2 Solution for Pair domains

As mentioned above, when two particles diffuse and such as they may interact during an interval of time δt, an analytical solution can still be found. However, in this case, solving the diffusion equation is a more complex task than for Single domains.

In the one-dimensional case, a particle moving with diffusion coefficient D along an axis of motion has a chance to migrate towards its neighbour in a Pair domain and interact. Let x_c be the point of contact, i.e. the position at which the particle reacts with its neighbour. This occurs when the distance between both particles equals the capture radius r_c between them, as defined in subsection 2.2.2. This scenario is analogous to a particle moving in one dimension with an absorbing boundary at x_c. In this case, the Green's function takes the form [67, 231]:

$$p(x,t|x_0) = \frac{1}{\sqrt{4\pi Dt}} \left[\exp\left(-\frac{(x-(x_0-x_c))^2}{4Dt}\right) - \exp\left(-\frac{(x+(x_0-x_c))^2}{4Dt}\right)\right] \tag{5.13}$$

In the case of three dimensions, the probability density of finding the particles p and q at time t at $\vec{r_p}$ and $\vec{r_q}$ knowing they are initially located at $\vec{r}_{0,p}$ and $\vec{r}_{0,q}$ at $t=0$ is governed by the following equation [231]:

$$\frac{\partial}{\partial t} p(\vec{r_p},\vec{r_q},t|\vec{r}_{0,p},\vec{r}_{0,q}) = [D_p \nabla^2_{r_p} + D_q \nabla^2_{r_q}] p(\vec{r_p},\vec{r_q},t|\vec{r}_{0,p},\vec{r}_{0,q}) \tag{5.14}$$

The equation above must be solved for $\|\vec{r_p}-\vec{r_q}\| \geq r_c$, where r_c is the capture radius defined for their interaction. As it is often assumed in the OkMC approach [33, 88, 146, 214], we consider here that the time of reaction between two particles is much shorter that the characteristic time of the diffusion process. The reaction can thus be considered as instantaneous.

As suggested by Van Zon and ten Wolde [293], the problem of solving Eq. 5.14 can be simplified by a convenient transformation of coordinates:

$$\begin{aligned}\vec{r} &= \vec{r_p} - \vec{r_q} \\ \vec{R} &= \sqrt{D_q/D_p}\vec{r_p} + \sqrt{D_p/D_q}\vec{r_q}\end{aligned} \tag{5.15}$$

where $\vec{r} = \vec{r_p} - \vec{r_q}$ is the distance between the two particles and $\vec{R}$ is the weighted centre-of-mass of the two particles. This change of coordinates allows to split Eq. 5.14 into two uncoupled diffusion equations in $\vec{r}$ and $\vec{R}$, with $p(\vec{r}, \vec{R}, t|\vec{r_0}, \vec{R_0}) = p_r(\vec{r}, t|\vec{r_0})p_R(\vec{R}, t|\vec{R_0})$:

$$\frac{\partial}{\partial t} p_R(\vec{R}, t|\vec{R_0}) = D_{tot}\nabla^2_R p_R(\vec{R}, t|\vec{R_0}) \tag{5.16}$$

$$\frac{\partial}{\partial t} p_r(\vec{r}, t|\vec{r_0}) = D_{tot}\nabla^2_r p_r(\vec{r}, t|\vec{r_0}) \tag{5.17}$$

where $D_{tot} = D_p + D_q$.

Eq. 5.16 that governs the evolution of $p_R(\vec{R}, t|\vec{R_0})$ corresponds to the free diffusion of the centre-of-mass. The solution of this equation is thus similar to that of Eq. 5.11:

$$p_R(\vec{R}, t|\vec{R_0}) = \frac{1}{(4\pi D_{tot}t)^{\frac{3}{2}}} \exp\left(-\frac{(\vec{R} - \vec{R_0})^2}{4D_{tot}t}\right) \tag{5.18}$$

In turn, Eq. 5.17 describes the diffusion of two interacting particles, initially at a mutual distance r_0. As mentioned above, a reaction between the two particles is supposed to occur instantaneously if their distance is smaller than the capture radius r_c. This corresponds to the following boundary conditions (Dirichlet):

$$p_r(\|\vec{r}\| = r_c, t) = 0 \tag{5.19}$$

The derivation of the Green's function corresponding to these conditions can be found in literature [228, 231].

$$p_r(r,t) = \frac{r}{r_0\sqrt{4\pi D_{tot}t}}\left(\exp\left[-\frac{(r-r_0)^2}{4D_{tot}t}\right] - \exp\left[-\frac{(r+r_0-2r_c)^2}{4D_{tot}t}\right]\right) \tag{5.20}$$

As can be verified, this solution predicts that there is a region of space where the probability of finding both particles at the same time becomes negative when their distance is $r \leq r_c$. This is the region where particles react.

5.2.3 Concept of Survival probability

Once all the domains have been defined, in the original GFRD method, it is necessary to estimate the next-time event for each domain individually [292, 293]. For instance, for Single domains, the next event simply corresponds to hitting the outer absorbing domain boundary by the particle. In the case of a Pair domain, next-event can be the reaction between both particles, when the centre-of-mass reaches the outer boundary, or when the particles reach the outer boundary conditions. For each domain, the next-event time is sampled using the corresponding analytical probability solution of the diffusion (Single domains) or diffusion-reaction (Pair domains) equation. In Pair domains, this can be done using the concept of survival probability, which is the probability for the particle(s) to still remain in their domain Ω at time t and that is defined as follows [292, 293]:

$$S(t) = \int_\Omega p(\vec{r}, t|\vec{r_0})d\vec{r} \tag{5.21}$$

For instance, for the case of two particles moving in three dimensions and interacting, the survival probability, i.e., the probability that the particles have not recombined by time t, is obtained by integration of Eq. 5.20, which yields [228]:

$$S(t|\vec{r_0}) = 1 - \frac{r_c}{r_0}\text{erfc}\left(\frac{r_0 - r_c}{\sqrt{4Dt}}\right) \tag{5.22}$$

If particles move in one dimension, the survival probability of a Pair after time step δt is equal to the survival probability of a particle moving along a straight line in the presence of an absorbing wall. This survival probability is calculated as [41, 231, 269]:

$$S(t) = \operatorname{erf}\left(\frac{x_0 - x_c}{\sqrt{4Dt}}\right) \tag{5.23}$$

Knowing the expression of $S(t)$ for each domain, it is thereby possible in principle to estimate the next-time event for a given domain with:

$$\tau = S^{-1}(1 - \xi) \tag{5.24}$$

where $\xi \in (0, 1]$ is a uniformly distributed random number.

Hence, it is necessary to derive the analytical expression of S^{-1} to calculate the next-time event associated to a domain. However, this can be difficult since S does not have a simple form, as can be seen in Eq. 5.22. Alternatively, the next-time event τ can be obtained by solving $S(\tau) - \xi = 0$ with a numerical rootfinder.

Once the event times are obtained for all the different domains, they are ordered chronologically and the events are performed in the same order. Since the event times associated to the domains are, *a priori*, different, each particle evolves with its own local time and the original GFRD method is therefore an asynchronous algorithm.

5.2.4 Construction of Singles and Pairs

We believe it might be useful to explain the construction of Single and Pair domains in MEGA-OkMC. In their original work [292, 293], Van Zon, ten Wolde et al. developed the GFRD method with a particle system composed of identical particles, i.e. particles with the same diffusion coefficient. In contrast, in the present work, we assume that there are different species of particles in the system which, in general, have different mobilities. In our method, it is necessary to identify the *fastest particles* in the system as a first step, meaning the particles with the largest migration rate. We make the assumption that all other particles can be considered immobile with respect to them. This assumption is justified because a small difference in migration en- ergy between two different particles induces a big difference between migration rates. For instance, the ratio between migration rates of SIA and I_2 in Fe takes the value $\frac{r_{mig}(SIA)}{r_{mig}(I_2)} = 185.3$ at $T = 200\ K$ and $\frac{r_{mig}(SIA)}{r_{mig}(I_2)} = 32.5$ at $T = 300\ K$, using the values found in Ref. [105]. In other words, at room temperature, the number of jumps performed by an SIA is, on average, 32 times larger than the number of jumps performed by an I_2 in the same time interval. Afterwards, non-overlapping protective environments are created around mobile particles in such a way that only one or two particles are found within them.

5.2.4.1 Fastest particles moving in three dimensions

As mentioned in subsection 5.2.1 and shown in Fig. 5.1, the protective environments around particles that migrate in three dimensions are spheres. The sphere radius represents the mean distance covered by the particle after time δt_{GFRD}.

Green's functions for Single domains are calculated with Eq. 5.11. If two particles are located inside a Pair domain, their survival probability is given by Eq. 5.22, and the motion of their centre of mass is given by Eq. 5.18.

5.2.4.2 Fastest particles moving in one dimension

To isolate the fast 1D-moving particle p and its closest neighbour q_1 in a Pair domain, it must be guaranteed that motion in the opposite direction is not perturbed by the presence of another close neighbour. In other words, if the position vector to p is $\vec{x_0}$ and the vector to the point of contact with its closest neighbour q_1 is $\vec{x_c}$, this pair can only be isolated if movement in direction $(\vec{x_0} - \vec{x_c})$ is free. If a second close neighbour q_2 is found along the axis of motion of p in said direction, this assumption is false. This situation is illustrated in Fig. 5.2. In the top example, the distance between particles p and q_2 is notably larger than the distance between p and q_1. Therefore, it can be assumed that they form an isolated Pair, so the movement of p can be described by Eq. 5.13 and the survival probability of the Pair is expressed by Eq. 5.23.

The bottom example shows a situation where the distance between p and $\vec{q_2}$ is comparable to the distance between p and q_1. Hence, $\vec{q_2}$ would act as a second absorbing boundary for p. In this case, Eq. 5.13 cannot describe the motion of p accurately and the pair (p, q_1) cannot be considered as isolated. The movement of a particle migrating in one dimension with absorbing boundaries x_{c1} and x_{c2} takes the form of a Fourier series given by Eq. 5.25 [231, 248]:

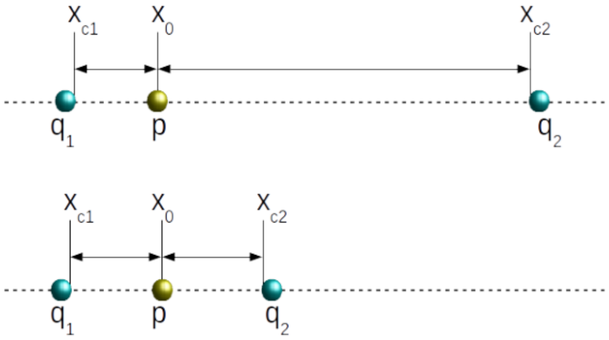


Figure 5.2: Top: Particles p and q_1 form a Pair. Bottom: Particles p and q_1 may not be isolated due to the close presence of q_2

$$p(x,t|x_0) = \frac{2}{L}\sum_{n=1}^{\infty} \exp\left(-\left[\frac{n\pi}{L}\right]^2 Dt\right) \sin\left(\frac{n\pi}{L}x\right) \sin\left(\frac{n\pi}{L}x_0\right) \tag{5.25}$$

where $L = |x_{c2} - x_{c1}|$.

It follows that, when particle q_1 is found, the line of motion has to be traversed in the opposite direction until a certain point of validity to decide whether p and q_1 form a Pair. To find the point of validity, the solutions given by Eq. 5.13 and 5.25 were compared to each other. By fixing x_{c1} and varying the position of x_{c2}, it is possible to find the point at which both solutions are similar enough, such that the Pair (p, q_1) is considered isolated.

The Green's functions $p(x,t|x_0)$ are portrayed in Fig. 5.3 for a value $Dt = 1$ and different positions of x_{c2}. Equivalent Green's functions are portrayed in Fig. 5.4 for $Dt = 2$. Jump length was set at $\lambda = 1$ and the position of x_0 was fixed at $x_0 = \sqrt{5}$.

As one can see, both solutions approach each other the bigger the separation between p and q_2 is, to the point that they are barely distinguishable when $|x_{c2} - x_0| \geq 2 \cdot |x_0 - x_{c1}|$. Thus, we used the following criterion to determine whether p and q_1 form a Pair or p is included in a Single domain. If the closest neighbour q_1 is found and the point of contact x_{c1} is determined, the axis of motion of p has to be inspected from x_0 to $x_0 + 2 \cdot (x_0 - x_{c1})$. If no particles are found in this segment, then (p, q_1) form a Pair. On the other hand, if a particle q_2 is found in the segment between x_0 and $x_0 + 2 \cdot (x_0 - x_{c1})$, particle p must be isolated in a Single domain.

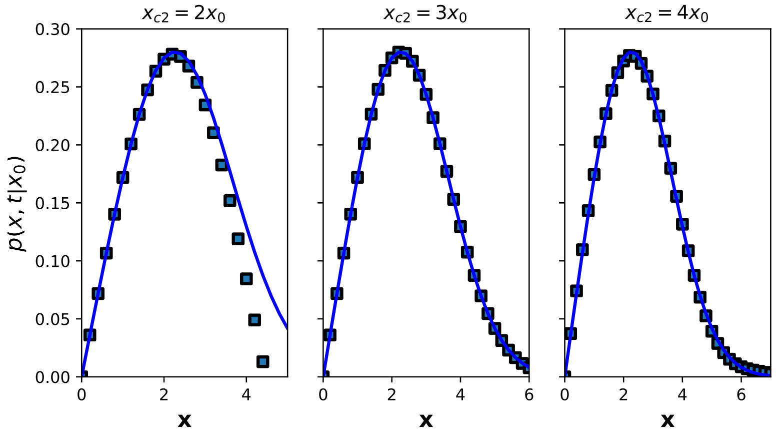


Figure 5.3: Green's functions for particles moving in one dimension in the presence of one absorbing wall (solution to Eq. 5.13, depicted in blue lines) and two absorbing walls (solution to Eq. 5.25, depicted in black squares). Here, $Dt = 1$.

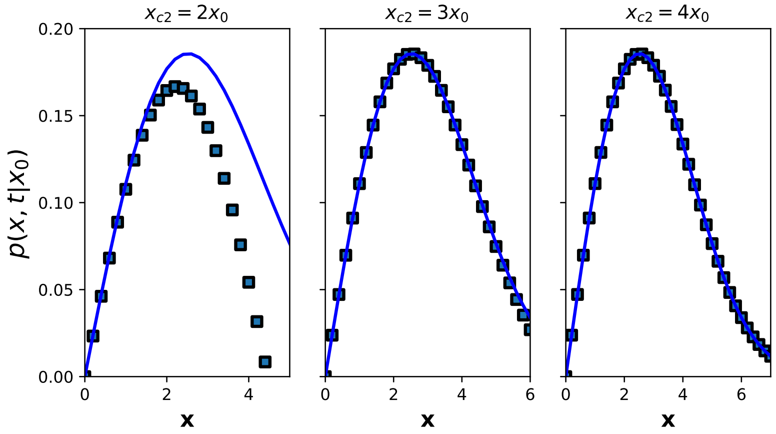


Figure 5.4: Green's functions for particles moving in one dimension in the pres- ence of one absorbing wall (solution to Eq. 5.13, depicted in blue lines) and two absorbing walls (solution to Eq. 5.25, depicted in black squares). Here, $Dt = 2$.

Moreover, it is possible that no particle q_1 will be found at all along the axis of motion of p. For this reason, it is helpful to establish an arbitrary maximum time step t_{max} so that the neighbour search algorithm is not executed indefinitely, but stopped at a cutoff distance that is a function of t_{max}.

With this in mind, we shall now outline the algorithm used for the construction of Single and Pair domains when the fastest particles migrate in one dimension along an axis of motion.

Algorithm 1 Pseudocode that shows the computations performed in order to create Single and Pair domains, when the fastest particles move in one dimension.

```
for all particles $p_i$ in positions $\vec{x}_{0,i}$ do
    traverse the axis of motion of $p_i$ from $\vec{x}_{0,i}$ until a particle $q_i$ is found
    if no particles are found then
        stop search at distance $\sigma_{max} = \sqrt{2D\delta t_{max}}$
    else
        identify neighbour $q_{1,i}$ and store distance $\sigma_i = \vec{x}_{0,i} - \vec{x}_{c1,i}$
    end if
    traverse the axis of motion of $p_i$ from $\vec{x}_{0,i}$ until position $\vec{x}_{0,i} + 2 \cdot (\vec{x}_{0,i} - \vec{x}_{c1,i})$
    if no particles are found in the segment $[\vec{x}_{0,i}, \vec{x}_{0,i} + 2 \cdot (\vec{x}_{0,i} - \vec{x}_{c1,i})$ then
        identify $p_i$ and $q_{1,i}$ as a Pair
    else
        isolate $p_i$ in a Single cylindrical domain with length $\sigma_i = \frac{1}{2}\|\vec{x}_{0,i} - \vec{x}_{c1,i}\|$
    end if
end for
```

Having explained the construction of Single and Pair domains when the fastest particles move in one or three dimensions, we shall now detail one of the main achievements in this work, namely the application of the GFRD method in the presence of an external source of particles.

5.3 Application of the GFRD method to systems with continuous introduction of particles

Although the original GFRD algorithm is in principle much faster than a scheme using only discrete random walks, in particular in the case of a dilute system, it can become computationally expensive under certain conditions. For instance, we can anticipate that the GFRD method requires an increased computational effort as the number of particles grows in the system. This is the situation when a material is subjected to continuous irradiation. Indeed, the more particles the system contains, the larger the number of domains to be calculated and the higher the computational effort necessary to sample the next-time events in all the protective domains.

On the other hand, the fact that the GFRD algorithm is inherently asynchronous makes it suboptimal to simulate the evolution of a system where particles are continuously introduced. Indeed, the number of particles to be introduced by irradiation in the system is by definition proportional to the time step, which is common to all particles. In addition, one can anticipate the difficulty to define protective domains around particles during a given time interval at the same time that others are being introduced. One expects that some of the newly introduced particles will be located into some of the protective domains previously defined. In this situation, some domains could contain more than two particles, condition for which it is no longer possible to find an analytical solution for the diffusion-reaction equation, as we have seen before.

In the current section, we describe how these issues can be circumvented and we present the new GFRD algorithm that we specifically developed to simulate the evolution of systems of interacting with an external source.

5.3.1 Synchronous scheme: Selection of the time step

As it was mentioned above, it is desirable to find a method that can significantly reduce the computational cost associated to the sampling of the next-time events in the original GFRD method. In their work, Van Zon, ten Wolde et al. [292] created a list of possible events which were associated to their survival probability functions. Then, these were used to sample next-reaction times and select the event to undergo, if an event occurs within the maximum allowed time step. Instead of using a sampling method – that can be computationally costly – to determine the next-time event of each domain, in this work we propose to use a more direct method.

To calculate the time step associated to an event in a given protective domain, one has to keep in mind that during the time interval that particles will evolve, they must not cross the outer boundary of their protective domain. This is one of the principles of the GFRD method. If this is not guaranteed, some particles could penetrate other domains and the analytical solution of the diffusion or diffusion-reaction problem would no longer be valid.

To prevent that particles reach the outer boundary of their protective domain and penetrate another domain during a given δt, it is useful to calculate the probability to find a particle outside its domain Ω. If we assume, for instance, that a particle performs random walks in three dimensions and that it starts at the origin, we can calculate the probability to find the particle at a distance $r \leq R$ by integrating Eq. 5.11 in spherical coordinates, which yields (see Ref. [239]):

$$P(r \leq R) = \frac{1}{\sqrt{\pi}}(\sqrt{\pi}\mathrm{erf}(u) - 2u\exp(-u^2)) \tag{5.26}$$

with $u^2 = R^2/4D\delta t$.

On the other hand, it is useful to remember a well-known property of random walks, which goes back to Einstein[94, 229] and according to which the mean square distance σ^2 travelled by a random walker is proportional to the time interval δt. For a particle that performs random walks in three dimensions, it is given by:

$$\sigma^2 = 6D\delta t \tag{5.27}$$

With this in mind, one might calculate, for instance, what is the fraction of particles that go beyond $r = \sigma$ during an interval of time δt. Using Eq. 5.26 and $R = \sigma$, we obtain $P(r \leq \sigma) \simeq 0.608$. This means that, if we choose a δt for a given spherical domain such that the mean distance σ travelled by the particles corresponds to the radius R of the domain, around 40% of the particles would cross the outer boundary of the domain and could penetrate other domains. On the other hand, one can verify that $P(r \leq 2\sigma) \simeq 0.992$. In other words, if particles are allowed to travel during a time interval δt such that $2\sigma = R$, then it would be very unlikely that a particle crossed the outer boundary of its protective domain. Thereby, for a given domain of radius R, it is possible to calculate in a simple manner a conservative time step that minimises the chance that a particle will cross the outer boundary of the domain. It is given by the following condition:

$$6D\delta t_{max} = (R/2)^2 \tag{5.28}$$

Clearly, this method allows to determine the maximum time interval δt_{max} that particles can travel inside a domain of a given radius R without reaching its outer boundary in an analytical and thus efficient manner.

Now, since our aim is to use the GFRD method with a synchronous scheme, i.e., a scheme where all particles will evolve with the same time step, it is necessary to select an appropriate and unique time step value to evolve all the particles during a given update. Though in this scheme, the time step is the same for all the particles during an update, it does not mean that it is constant throughout the simulation. Since the density of particles changes with time, the domains must be recalculated and thus, the time step must adapt to the situation.

A conservative approach to ensure that no particle will cross the outer boundary of its protective domain is to select the minimum of the time steps obtained with Eq. 5.28 from all the domains. In order to determine the accuracy and the efficiency achieved with the GFRD method using a synchronous scheme and with the minimum of the time steps found among all the domains in the system, the GFRD algorithm was implemented in our MEGA-OkMC code. As a practical case, we simulated the evolution of SIAs and vacancies in Fe. The values for the attempt frequency, migration energies and capture radius were equal to those used in Section 4.3. These parameters were used to calculate their diffusion coefficients, which are used in the diffusion-reaction equations and in Eq. 5.28.

We considered a hypothetical case where SIAs and Vs are initially randomly distributed in space at a given concentration and where the system evolves at different temperatures for different times. In Fig. 5.5 we can see the evolution of the SIA concentration obtained with the GFRD using the minimum time step (blue line) for an annealing at 250 K and for initial SIA/V concentrations of 10^{19} and 10^{20} cm^{-3}. For comparison, we also show the results obtained using only the PTLA (green squares). The latter does not use a geometrical approximation like the GFRD and is thus, by definition, more accurate. The agreement between both methods is very good. A similar good agreement was found for the other conditions of temperature and time explored in this work.

Since SIAs have a migration energy smaller than that of vacancies, the characteristic time step of the system is imposed by the SIAs. For example, at a temperature of 250 K, a SIA performs on average one jump in approximately 7.14×10^{-7} s. Therefore, the simulations performed with the PTLA for the temperature of 250 K during 1 s required about 1.4 million computational steps. As expected, the GFRD method requires fewer computational steps, which results in a reduction of the computational effort.

In order to quantify the reduction of computational effort achieved by the GFRD using the minimum time step, we calculated the speedup factor, which is defined as

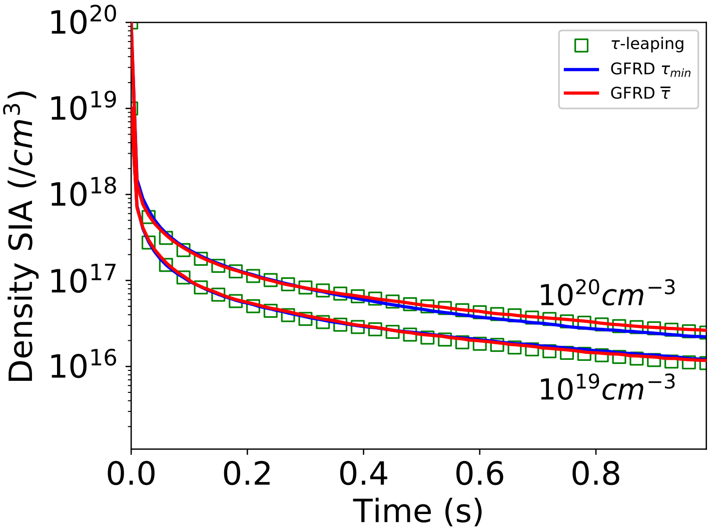


Figure 5.5: Comparison between results obtained with discrete random walks (green squares) and the GFRD with a synchronous scheme using the minimum time step (blue line) and the average time step (red line) for initial concentrations of 10^{19} and 10^{20} cm^{-3}.

SIA/V conc.	Temperature	t_{end}	Nb. Steps	Runtime
(cm^{-3})	(K)	(s)		(s)
10^{19}	200	1	27063	492.3
10^{19}	250	1	1399267	13706.2
10^{19}	300	0.1	1942007	17530.8
10^{20}	200	1	27063	371.36
10^{20}	250	1	1399267	10128.1
10^{20}	300	0.1	1942007	20227

Table 5.1: Number of computational steps and runtime corresponding to the simulations performed using the τ-leaping method for each of the conditions considered in this work.

the ratio of the runtime achieved with the PTLA to the runtime achieved with the GFRD method. This was done for each condition considered in this work and was reported in the last column of Table 5.2. For the first condition, we can notice that the GFRD method is even slower than the PTLA as the speedup factor is lower than 1. This is due to the overhead necessary to the calculation of the protective domains, which is higher than the gain achieved with the GFRD method in this particular condition. Otherwise, as a general trend, we noticed that the gain achieved by GFRD is enhanced at higher temperatures. This can be explained by the fact that the characteristic time step associated to the jump of SIAs gets smaller as the temperature increases. Hence, the number of computational steps increases significantly with temperature when using only discrete random walks. This is where the advantage of the GFRD method is exploited since much fewer computational steps are required to reach a same physical time, as can be seen by comparing Tables 5.1 and 5.2. However, for most conditions explored in this work, we can see that the speedup factor achieved by the GFRD method and a conservative time step remains modest. Such modest gain can be explained by the fact that the time step is always limited by the occurrence of particles that are close to each other, since the minimum time step is taken. Their protective domain has thus a small extension and consequently, the corresponding time step is small.

SIA/V conc.	Temperature	t_{end}	Nb. Steps	Runtime	Acceleration factor
(cm^{-3})	(K)	(s)		(s)	
10^{19}	200	1	23825	593.18	0.82
10^{19}	250	1	114103	1440.77	9.51
10^{19}	300	0.1	113018	1409.23	12.43
10^{20}	200	1	20958	248.14	1.49
10^{20}	250	1	65800	570.90	17.74
10^{20}	300	0.1	63288	561.83	36.0

Table 5.2: Number of computational steps and runtime corresponding to the simulations performed using the GFRD method and the minimum time step for each of the conditions considered in this work.

Since such reduction of the computational effort was achieved with the minimum time step in the system, one expects thus that with larger time steps the number of computational steps could be reduced even further. Since we are dealing with a system with a large number of particles, from the statistical point of view it seems reasonable to describe its behaviour with the average time step $\overline{\tau}$, which can be calculated at the beginning of each update from the time step distribution obtained from all the domains in the system.

The evolution of the SIA concentration with time obtained with the GFRD method and the average time step (red line) for a temperature of 250 K is illus-

SIA/V conc.	Temperature	t_{end}	Nb. Steps	Runtime	Acceleration factor
(cm^{-3})	(K)	(s)		(s)	
10^{19}	200	1	129	5.10	96.52
10^{19}	250	1	817	11.57	1184.63
10^{19}	300	0.1	934	12.03	1457.25
10^{20}	200	1	231	4.40	84.4
10^{20}	250	1	1353	22.76	444.99
10^{20}	300	0.1	1608	32.45	623.32

Table 5.3: Number of computational steps and runtime corresponding to the simulations performed using the GFRD and the average time step for each of the conditions considered in this work.

trated in Fig. 5.5 along with the results discussed above. Clearly, the agreement with the results obtained with discrete random walks is excellent. Table 5.3 quantifies the computational effort achieved using the GFRD method and the average time step for the different conditions considered in this work. As one can see by comparing Tables 5.2 and 5.3, simulations performed with the average time step require remarkably fewer steps than those performed using the minimum time step. As a result, simulations are considerably accelerated. For some conditions, the acceleration factor with respect to simulations performed with the PTLA is larger than three orders of magnitude.

We have thus evidenced that the results produced with the GFRD method using the average time step are in excellent agreement with those obtained with the PTLA. Hence, although the chosen time step –the average time step– is relatively large, the resulting loss of accuracy is negligible. At first glance, one could indeed expect a large deviation, since a certain amount of particles are located in domains with an associated time step which is smaller than the average time step $\overline{\tau}$. Thus, these particles are able, in principle, to cross the outer boundary of their respective domains and intersect the domains of other particles. In order to explain why the loss of accuracy is negligible in spite of this, we did the following. For each particle in the system, we used Eq. 5.26 to compute the probability $P(r \leq R)$ that the particle remains inside its protective domain of radius R during an interval of time equal to $\overline{\tau}$. For the sake of simplicity, we assumed that all particles are isolated in Single domains. In Fig. 5.6 we plotted the probability distribution (red histogram), i.e.,

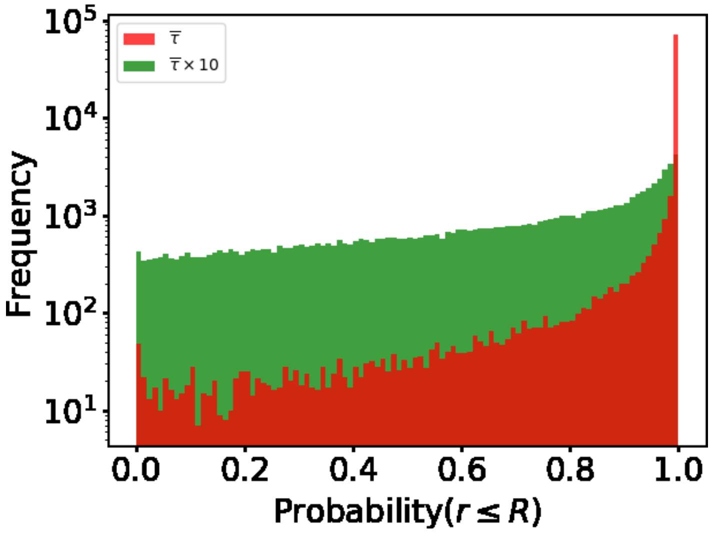


Figure 5.6: Distribution of the probability $P(r \leq R)$ that a particle has to remain inside its protective domain during a time interval $\overline{\tau}$ (red histogram) and during a time interval $\overline{\tau} \times 10$ (green histogram). Results are shown for a temperature of 250 K at time $t = 1.75 \times 10^{-2}s$.

the number of particles that have a probability between P and $P + \Delta P$ to remain inside their protective domain. This was done for a temperature of 250 K and an initial concentration of 10^{19} cm^{-3}, i.e., one of the conditions showed in Fig. 5.5. The distribution represented in Fig. 5.6 was calculated at the 100th time step of the simulation, which corresponds approximately to $t = 1.75 \times 10^{-2}s$. At this time, the average time step $\overline{\tau}$ calculated from the time step distribution is about $3.7 \times 10^{-4}s$. As we can see in the figure, the probability histogram exhibits a stiff peak around a value of 1. This means that a significant amount of the particles in the system have a high probability to remain inside their domain during the time step $\overline{\tau}$. Nevertheless, the figure also evidences that there are a certain amount of particles that have a non-negligible probability to escape from their protective domain. In order to quantify the amount of particles in the system that could reach the outer boundary of their domain with a significant probability during the time step $\overline{\tau}$, we calculated the fraction of particles that have a probability $P(r \leq R) \leq 0.90$, i.e., the fraction of particles that have more than 10% probability to escape from their domain. Our calculations show that only about 5% of the particles in the system fulfill this condition. Instead, if we calculate the fraction of particles that have a probability higher than 20% to escape from their domain ($P(r \leq R) \leq 0.80$) then our calculations show that only 3.4% of the particles in the system would be affected. In other words, only a small fraction of the particles have a non-negligible probability to exit their domain and interact with other particles in adjacent domains during the time interval $\overline{\tau}$. This explains why the loss of accuracy is negligible when a large time step such as $\overline{\tau}$ is used to evolve all the particles synchronously.

However, we observed that the probability distribution shifts more and more to the left as the time step used to evolve the particles increases. As expected, the number of particles that can reach the boundary of their domain with a significant probability rapidly increases when the time step becomes too large. To illustrate this, we calculated the probability distribution in the same conditions as before but for a time step equal to $10 \times \overline{\tau}$ (see green histogram in Fig. 5.6). Our results show that for this time step, approximately 71% of the particles in the system would have more than 10% probability to escape from their domain. In this case, we observed indeed that results obtained with such large time step significantly deviate from the results calculated with discrete random walks.

Using the GFRD method with a synchronous scheme and the average time step $\overline{\tau}$ seems thus a good choice that allows for simulating the evolution of systems in a very efficient manner and with a very good accuracy.

5.3.2 Development of a new scheme for systems with continuous introduction of particles

In the previous subsection, we established that the GFRD method can be used with a synchronous scheme. It was shown that, when all particles in the system are evolved with the average time step computed from all the protective domains, GFRD displays a very good accuracy when compared to simulations performed only with discrete random walks. This method achieves a considerable reduction of the computational effort, reaching for some conditions an acceleration factor larger than three orders of magnitude. These results were obtained in a specific case, namely a system with an initial distribution of defects.

Another interesting example of a diffusion-reaction system is the case where a continuous external source of particles is present. Indeed, this is the situation in materials under continuous irradiation. In this case, defects are continuously introduced in the system at the same time that those that are already present migrate and interact. *A priori*, this situation makes the use of the GFRD method difficult, not to say impossible. Indeed, in a conventional system where GFRD is typically used, there is no external source of particles. The system is defined by an initial number of particles distributed in space and at the beginning of each update, the number of particles is known as well as their position, which allows for the definition of a protective domain around each particle. The spatial extension of these domains defines the time during which particles will travel isolated from the rest (Single domains) or interacting with another particle (Pair domains). In the case we are considering now, new particles are introduced during the interval of time that the already present particles are supposed to travel in isolation, within the limits of their protective domains. Consequently, these new particles might either fall into some of the protective domains corresponding to the particles already present in the system or, in a more general way, their protective domains might intersect with some of the protective environments that were already defined. In both cases, the solution of the diffusion equation calculated for the domains where this occurs would no longer be valid. For instance, if a new particle is introduced during a time step into a Single domain, this latter would become a Pair domain and the solution given by Eq. 5.11 would not be applicable. Similarly, if a new particle is introduced such that its domain intersects with a Pair domain that was already defined, one of the fundamental principles of the GFRD method would be violated, since the problem of diffusion can only be solved analytically for domains that contain, at most, two particles.

In this section, we propose a new GFRD algorithm that circumvents this problem and allows treating the case where particles are continuously introduced in the system. For the sake of simplicity, we present the algorithm for the case of particles that perform random walks in three dimensions, i.e., for spherical protective domains. The same reasoning can of course be extended to other dimensions.

Let us assume that after some time t, the system already contains some particles and that we wish to evolve the system by a certain time step δt. During this time step δt, the particles already present in the system will migrate and possibly interact with others while new particles will be introduced. In the new GFRD algorithm that

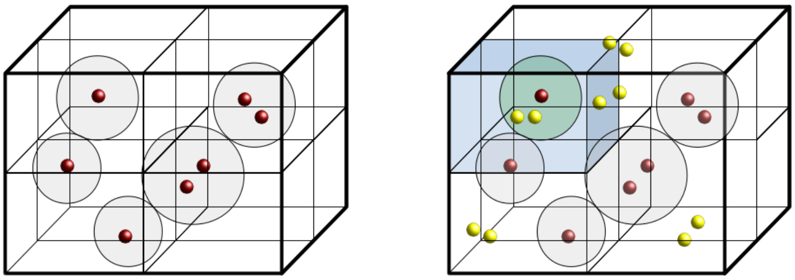


Figure 5.7: Left: Single and Pair domains in the simulation box before new particles are introduced. Right: Single and Pair domains after new particles are introduced. Newly introduced particles are coloured yellow. The protective sphere highlighted in green cannot be solved analytically since it contains more than two particles. The highlighted cell (blue) where the domain with more than two particles appears is labelled as unsolvable cell.

we propose, we first consider the particles that are already present in the system. As in the original GFRD method, protective environments (spheres) are created around the particles and domains are identified as Single and Pair domains, as shown in Fig. 5.7 on the left. Then, as described in the previous subsection, the average time step $\overline{\tau}$ is calculated from all the protective domains. This will be the time step used to evolve all the particles synchronously. The time step $\overline{\tau}$ thereby calculated is also used to estimate the number of new particles to be introduced in the system during the update owing to the rate of the external source. The new particles are then placed in the simulation box and the extension of their protective domains is calculated assuming a time step $\overline{\tau}$. When the protective domain of one of the newly introduced particles intersects with some of the domains already defined, the cell in which it is located is identified and labelled as *unsolvable cell*. This denomination refers to the fact that no analytical solution can be found in this case. The rest of the cells, i.e, where no domains intersect, are labelled as *solvable cells*. Fig. 5.7 on the right schematically illustrates the simulation box after the new particles have been introduced by irradiation. Here, some particles have been introduced inside one of the protective environments, highlighted in green colour. Since there are more than two particles in the protective sphere, in principle, the problem of diffusion cannot be solved analytically. The cell in which this domain is located, highlighted in blue, is therefore labelled as an unsolvable cell.

Once this classification has been done, the particles located in Single and Pair domains in solvable cells, i.e., those that have not been altered by new particles, are first evolved with the time step $\overline{\tau}$ calculated previously. For the unsolvable cells, i.e., where the protective domains of the new particles intersect with other domains, we proceed as follows. For a given unsolvable cell i, the protective domains around all the particles are reconstructed, taking into account the new particles introduced by irradiation. This leads to a local redefinition of the Single and Pair domains. The minimum of the time steps corresponding to these new protective domains, denoted $\delta t_{min}^{i,1}$, is determined and is, by definition, such that $\delta t_{min}^{i,1} < \overline{\tau}$. Particles in the new Single and Pair domains in the unsolvable cell i are then evolved using the time step $\delta t_{min}^{i,1}$. Since the sub-time steps *a priori* vary from one unsolvable cell to another, the particles that have evolved in different unsolvable cells are no longer synchronized with the rest of the particles. In order to synchronize again with the rest of the system, i.e., with the particles in solvable cells, they must continue evolving by sub-time steps until they reach the time step $\overline{\tau}$. Each unsolvable cell i is then examined again to define the new protective domains and to determine a new sub-time step $\delta t_{min}^{i,2}$, as before. The process is repeated until $\sum_j \delta t_{min}^{i,j} = \overline{\tau}$ is reached in each cell. Once all the particles in the unsolvable cells have reached the time step $\overline{\tau}$, all the particles can be considered synchronized and a new global time step $\overline{\tau}$ can be determined to evolve all the particles and to introduce new ones. Our GFRD algorithm is thus a combination of synchronous and asynchronous schemes. The following pseudocode summarises our GFRD algorithm.

In order to demonstrate the accuracy and the efficiency of our GFRD algorithm to simulate the evolution of systems with an external source of particles, we simulated an experiment that was carried out by Arakawa et al [16] in which Fe was irradiated with 1 MeV electrons. In these experimental conditions, only SIA-V pairs (Frenkel pairs) are expected to be created. Indeed, the calculation of the average of energy spectrum of collisions [163] shows that electrons displace Fe lattice atoms with, on average, a kinetic energy of approximately 11 eV. This implies that Fe atoms are only displaced by a small distance from their lattice site. As a result, only a SIA-V pair is formed when an electron succeeds to displace a Fe atom from its site. Thus, it is necessary to calculate how many SIA-V pairs are introduced in the simulation box during a time step δt. To do so, we must calculate how many

Algorithm 2 Pseudocode that summarises our GFRD algorithm developed to solve the problem of particle evolution under continuous irradiation.

```
while $t_{sim} < t_{end}$ do
    divide space into cells
    for all cells do
        define protective domains around particles
        calculate time step associated to each domain
    end for
    calculate average time step $\overline{\tau}$
    calculate number of new particles to be introduced by irradiation during $\overline{\tau}$
    introduce the new particles in simulation box
    calculate extension of domains of new particles
    identify unsolvable cells
    evolve particles in solvable cells with time step $\overline{\tau}$
    for all unsolvable cells $i$ do
        $j \leftarrow 1$
        $\delta t^{i}_{local} \leftarrow 0$
        while $\delta t^{i}_{local} < \overline{\tau}$ do
            recalculate protective domains
            determine minimum time step $\delta t^{i,j}_{min}$
            if $\sum_j \delta t^{i,j}_{min} < \overline{\tau}$ then
                evolve particles with $\delta t^{i,j}_{min}$
                $\delta t^{i}_{local} \leftarrow \delta t^{i}_{local} + \delta t^{i,j}_{min}$
                $j \leftarrow j + 1$
            else
                evolve particles with $\overline{\tau} - \delta t^{i}_{local}$
                $\delta t^{i}_{local} \leftarrow \overline{\tau}$
            end if
        end while
    end for
    $t_{sim} \leftarrow t_{sim} + \overline{\tau}$
end while
```

electrons hit the front surface of the material during this time step and how many SIA-V pairs are generated by each electron on average.

The number of electrons that impinge on the surface during a time step δt is simply given by Eq. 5.29:

$$nb_e = \phi \cdot S \cdot \delta t \tag{5.29}$$

where ϕ is the irradiation flux per unit surface and unit time, $9.2\times 10^{18}cm^{-2}s^{-1}$ in the experiment considered here [16], and S is the surface of the sample hit by electrons.

On the other hand, the average number of displacements that an electron can generate when going through the sample can be calculated as follows:

$$nb_{disp} = \sigma \cdot \rho \cdot z \tag{5.30}$$

where z is the thickness and ρ is the density of the material, i.e., the number of Fe atom per unit volume ($8.46 \times 10^{22}cm^{-3}$). In the experiment, Arakawa et al irradiated Fe samples with a thickness of 150 nm approximately. Using the relativistic expression for the differential cross-section [163], we calculated that for this energy (1 MeV) the total cross-section for displacements σ between electrons and Fe lattice atoms is equal to $1.46 \times 10^{-21}\ cm^2$. The average number of SIA-V pairs to be introduced in the system during a time step δt is thus merely $nb_e \cdot nb_{disp}$.

The following atomistic model was considered to simulate the evolution of this system of defects in Fe. Once the SIAs and the Vs are created by atomic displacements in Fe, they start performing random walks at a frequency determined by their migration energies, the same that were used in Sections 4.3 and 5.3.1. As described in Section 5.3.1, SIAs and Vs can annihilate upon encounter. In addition, we took into consideration that when two SIA defects approach at a sufficiently small distance, they can agglomerate into a cluster following the first order chemical reaction SIA$_n$ + SIA $\longrightarrow$ SIA$_{n+1}$. These SIA clusters can also migrate, as it was determined by first principle calculations [105, 175]. In this work, for the sake of simplicity, we consider that only small clusters of up to four SIAs can migrate and their migration energies are those calculated in Ref. [105]. Similarily to SIAs and Vs, SIA clusters can also annihilate with vacancies upon encounter following the reaction SIA$_n$ + V $\longrightarrow$ SIA$_{n-1}$. This is of course a simple model to describe the evolution of defects in Fe under irradiation. The atomistic mechanisms that can fully describe what occurs in Fe under irradiation are certainly more complex and are in fact, still under debate [175, 214, 234, 31, 105]. Our goal here is to demonstrate that our GFRD algorithm is able to efficiently simulate the evolution of a system of particles under realistic irradiation conditions and we believe that a simple atomistic model will suffice.

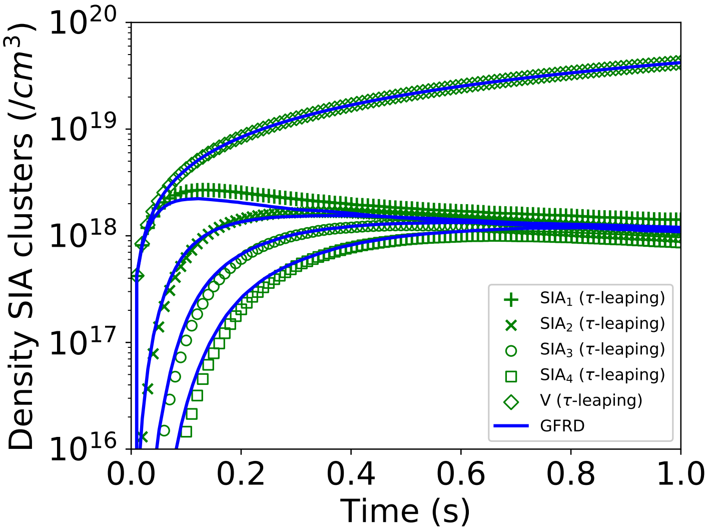


Figure 5.8: Comparison between the PTLA (green symbols) and GFRD with our synchronous-asynchronous scheme (blue line). Results were obtained for an irradiation at 200 K for 1 s. Experimental parameters of the irradiation conditions were taken from Ref. [16].

In order to test our GFRD algorithm in a wide range of conditions, we simulated the irradiation of Fe using the model mentioned above and the experimental parameters given in Ref. [16] for different temperatures and irradiation times. As in the previous subsection, results obtained with our GFRD algorithm were compared to those obtained with a scheme using only discrete random walks, i.e., without the approximation.

In Fig. 5.8, the evolution of the density of different SIA clusters (up to 4 SIAs) and of vacancies obtained with our GFRD algorithm is represented for a temperature of 200 K and an irradiation time of 1 s. The results obtained using discrete random walks (τ-leaping) are also plotted for comparison. As we can see, our GFRD algorithm is able to predict the evolution of different species that form under irradiation with a very good accuracy. For this condition, a comparison of the runtimes shows that the simulation performed with our GFRD algorithm requires 17 times less effort than that performed with only discrete random walks. The number of computational steps and the runtimes necessary to achieve the simulations that were performed for different conditions are summarised in Tables 5.4 and 5.5 for the PTLA and for our GFRD algorithm, respectively. Similarily to what we observe in

Temperature (K)	t_{end} (s)	Nb. Steps	Runtime (s)
200	1	27036	1161.9
250	1	1399275	32515.0
300	0.1	1942002	13265.1

Table 5.4: Number of computational steps and runtime corresponding to the simulations with continuous irradiation using the τ-leaping method.

Temperature (K)	t_{end} (s)	Nb. Steps	Runtime (s)	Acceleration factor
200	1	583	66.13	17.57
250	1	24757	1508.23	21.55
300	0.1	8290	247.62	53.57

Table 5.5: Number of computational steps, runtime and acceleration factor corresponding to the simulations with continuous irradiation using the GFRD and the average time step with our synchronous-asynchronous scheme.

conditions where particles are initially distributed without an external source, the gain achieved by our GFRD algorithm is better for higher temperatures.

We would like to point out that additional simulations were performed with the conventional GFRD algorithm, i.e., without identifying unsolvable cells and without using the synchronous-asynchronous scheme. The results (not shown here) obtained with the conventional GFRD algorithm strongly deviate from those showed in Fig. 5.8. This evidences that it is crucial to identify the domains where new particles have been introduced and to redefine the domains locally, taking into account the new particles.

5.4 Conclusions

The GFRD method developed by Van Zon and ten Wolde [292, 293] was adapted to our MEGA-OKMC code. GFRD is useful to accelerate low-density diffusion-reaction systems where random walkers perform many migrations before interactions happen.

As a first step, we have proposed an alternative way to estimate the time steps associated to Single and Pair protective domains to evolve particles. Using the probability that a free random walker has to be found beyond a certain distance after a certain time, we could establish the maximum allowed time step that a particle can travel without reaching the outer boundary of its protective domain. The method proposed to calculate the time step associated to a protective domain is analytical and thus, significantly reduces the computational effort that is in general necessary in the original GFRD method to sample time steps through numerical root finders.

Then, we have demonstrated that the GFRD can be used in a synchronous manner, i.e., that particles can be evolved with the same time step during the same update. To do so, we have evidenced that the varying time step must be carefully selected. On the one hand, we have showed that when the GFRD method is used with a conservative time step, i.e., the minimum time step found among all the domains in the system, it achieves an interesting reduction of the computational effort in comparison to simulations performed with only discrete random walks. On the other hand, we also have demonstrated that results obtained with the GFRD and the average time step $\overline{\tau}$ are in excellent agreement with those obtained with discrete random walks. We have showed that in this case, the GFRD method requires much fewer computational steps and hence, achieves a considerable reduction of the computational effort. For some conditions, simulations were accelerated by a factor larger than three orders of magnitude. The accuracy of the results obtained with the average time step evidence that, despite the fact that some particles belong to domains with an associated time step smaller than $\overline{\tau}$ and hence, might cross the outer boundary of their protective domains, the loss of accuracy is negligible. To explain this, we have calculated for each particle the probability $P(r \leq R)$ it has to remain in its protective domain during a time interval $\overline{\tau}$. Our calculations show that in fact, only a small fraction of the particles have a non-negligible probability to escape from their domain.

The feasibility of using the GFRD method in the presence of continuous irradiation has been demonstrated. Initially, the following circular problem was found. The GFRD time step has to be calculated with the help of Singles and Pairs in protective environments, such that the number of particles produced in cascades is computed accordingly; however, some of the particles created in cascades may enter the protective environments used to calculate this time step, which would violate the principle that Singles and Pairs are isolated. Solutions were offered when the fastest particles move in one or three dimensions. The GFRD algorithm that we propose consists in a hybrid scheme that combines a step during which particles are evolved synchronously and another step where the rest of the particles is evolved asynchronously by sub-time steps. In three dimensions, the cells which the simulation box are divided into were classified as solvable cells or unsolvable cells. Singles and Pairs in solvable cells are solved in one time step δt_{GFRD}, whereas unsolvable cells are treated separately. In the latter, Singles and Pairs are redefined and solved by sub-time steps $\delta t_{i,1}$, $\delta t_{i,2} \ldots$ until the final time step δt_{GFRD} is reached. In one dimension, it is sufficient to redefine Singles and Pairs and solve these problems in one time step δt_{GFRD}.

Our implementation of the GFRD method for materials under continuous irradiation was tested and compared to an equivalent simulation using only the PTLA, yielding promising results. In order to validate our GFRD algorithm we have simulated, taking the parameters of a realistic experiment, the formation and evolution of self-interstitial clusters and of vacancies in Fe under electron irradiation for different temperatures and times. The results obtained with our GFRD algorithm are in very good agreement with those obtained with simulations performed with only discrete random walks. The reduction of the computational effort achieved by our GFRD algorithm is enhanced for higher temperatures.

In the following chapter, the full implementation of the PTLA and GFRD methods into our MEGA-OkMC code is detailed.

Chapter 6

Integration of PTLA and GFRD and additional approximations in the MEGA-OkMC code

As we have seen in the previous chapter, the GFRD method is useful to accelerate OkMC simulations in low density conditions. Obviously, GFRD is not always applicable. For example, this method breaks down when particles are not able to perform many jumps before interactions take place, such that a diffusion coefficient D cannot be defined. In order to decide whether this assumption is correct at a particular time step, it is necessary to establish a criterion.

Thus, it is essential to provide the code with a mechanism that is able to choose the correct option (PTLA or GFRD) according to a criterion that depends on the physical situation. In particular, the calculations which are responsible for the creation of protective environments in GFRD are computationally expensive, which means that they are best avoided if GFRD cannot be used. In this chapter, the mechanism used in MEGA-OkMC to make the decision between PTLA and GFRD is explained. We refer to computational steps where the PTLA is applied as PTLA steps. Computational steps where the GFRD method is applied are labelled GFRD steps. Section 6.1 reviews the considerations that must be accounted for, such that GFRD is applicable in a given time step. In addition, some approximations to reduce runtime by means of choosing an appropriate time step shall be presented in Section 6.2. In Section 6.3, we show how MEGA-OkMC has been adapted to include the GFRD method in the presence of continuous irradiation. In the rest of this chapter, δt_{PTLA} and δt_{GFRD} refer to the time step calculated with PTLA and the GFRD method, respectively.

6.1 Considerations and assumptions in the application of GFRD to realistic diffusion-reaction systems

Assumption: migration dominates

As shown in chapter 5, GFRD is a geometric method. In our implementation, this method relies on the assumption that only the fastest particles are considered mobile, and all others can be considered immobile. This assumption is justified because a small difference in migration energy between two different particles induces a big difference between migration rates. For instance, the ratio between migration rates of I_2 and SIA in Fe takes the value $\frac{r_{mig}(I_2)}{r_{mig}(SIA)} = 185.3$ at $T = 200\ K$ and $\frac{r_{mig}(I_2)}{r_{mig}(SIA)} = 32.5$ at $T = 300\ K$. A long sequence of PTLA steps producing discrete particle jumps is replaced by one GFRD calculation, saving a considerable amount of runtime. This is correct since in this case, it is justified to define a diffusion coefficient for the fastest particles. In other words, the GFRD method is exact if the only event that happens within a time step is the migration of the fastest particles.

Naturally, this works well for simple models where a single particle species is so fast that migrations of other species can be neglected. In general, in a realistic diffusion-reaction system [65, 174, 267, 281, 284], there is an array of different particle species with varying mobilities, and it is not guaranteed that during the whole simulation, one specific species is substantially more mobile than the rest. In fact, the existence of different mobile species means that there is a non-zero possibility that not only the fastest particles are able to migrate within the time step. In addition, there are other events like dissociation of clusters or introduction of particles via continuous irradiation that should be considered.

For these reasons, we tried to implement the GFRD method in our work in a manner that is both accurate and compatible with particle systems in which different types of particles are present. As a first step, the MEGA-OkMC code checks whether the most probable event in the system is the migration of fastest particles. The fastest particles are identified and their migration rate $R_{max,mig}$ is determined. This is compared to the rate of external irradiation, R_{ext}, and the maximum rate of other events per particle, $R_{max,other}$. If $R_{max,mig} \geq R_{ext}$ or $R_{max,mig} \geq R_{max,other}$, the migration of fastest particles is not the likeliest event, and therefore GFRD cannot be applied in this step. The pertinent calculations to GFRD are avoided and a PTLA step is produced instead.

Assumption: second fastest particle can be considered immobile

It is possible that the computed time step δt_{GFRD} is large enough so that particles other than the fastest ones can perform one or more jumps during this time step. This might be the case, for instance, in α-Fe under irradiation. As small SIA clusters are formed during system evolution, a few DL $\langle 111 \rangle$ may appear through clustering, which coexist with mobile SIAs and I_2, I_3 and I_4 clusters. Due to the fact that these DLs migrate in one dimension [16, 17, 131, 234, 284, 289, 308, 309], they may cover a long distance until they find a neighbour to interact with. Consequently, the protective environments around them (cylinders) may become quite long until a collision between protective environments is found, and the computed time step becomes large enough for other particles (for example, the SIAs) to move as well.

Let $\delta t_{2nd} = \frac{\omega}{R_2}$ be the time step associated to the sum of event rates R_2 of the second fastest particles, i.e. the particles with the second largest sum of event rates in the system. If it is less than or equal to the time step δt_{GFRD} computed by the GFRD method, using the fastest particles in the simulation box, then it is incorrect to assume that the second fastest particles may be considered immobile during the time step.

Considering this possibility, we have implemented a solution that avoids this inaccuracy. Namely, we verify whether the possible events undergone by second fastest particles are likely or unlikely to trigger during the chosen time step. As mentioned above, if R_2 is the sum of event rates of the second fastest particles, these are expected to perform 1 event on average in the characteristic time step $\delta t_{2nd} = \frac{1}{R_2}$. Consequently, to minimise the chance that any events are undergone by the second fastest particles during the GFRD step, it is advisable to check whether the characteristic time step δt_{2nd} is large enough with respect to the selected time step δt_{GFRD}, computed via the GFRD method. In our code, δt_{2nd} is computed and compared to δt_{GFRD}. If $\delta t_{2nd} > \delta t_{GFRD}$, we assume that the GFRD method is accurate, therefore, a GFRD step with time step δt_{GFRD} is produced.

On the other hand, if $\delta t_{2nd} \leq \delta t_{GFRD}$, the GFRD method may not be applied in this computational step. In the latter case, a PTLA step is produced. It is in our interest to choose the largest possible time step that keeps the simulation accurate, in order to reduce the number of computational steps. One can simply check that the applicable time step is δt_{2nd} in this case. The question remains whether the choice this time step describes the behaviour of the fastest particles with accuracy. Indeed, we can assume that it does. Due to the fact that $\delta t_{2nd} \leq \delta t_{GFRD}$, the time needed by the fastest particles to leave their protective environments is larger than the selected δt_{2nd}, which justifies our choice.

Summing up, if $\delta t_{2nd} > \delta t_{GFRD}$, a GFRD step is produced with time step δt_{GFRD}, whereas if $\delta t_{2nd} \leq \delta t_{GFRD}$, a PTLA step is produced with time step δt_{2nd}.

Algorithm 3 sums up the computational steps taken to decide between a GFRD and a PTLA step, in pseudocode.

Algorithm 3 Pseudocode that shows the first computational steps performed to execute or ignore GFRD calculations.

```
do_GFRD_step ← false
do_PTLA_step ← false
determine $\delta t_{GFRD}$
$\delta t_{2nd} \leftarrow \frac{\omega}{R_2}$
if $\delta t_{2nd} \leq \delta t_{GFRD}$ then
    do_GFRD_step ← true
    $\delta t \leftarrow \delta t_{GFRD}$
else
    do_PTLA_step ← true
    $\delta t \leftarrow \delta t_{2nd}$
end if
```

6.2 Additions for the acceleration of simulations in MEGA-OkMC: approximation for very mobile particles at low density

Since the time step tends to decrease as T increases, it is desirable to minimise the amount of computational steps needed to complete the simulation. One possibility to achieve this is by applying convenient approximations. We shall now present some mechanisms included in MEGA-OkMC that accelerate computation by choosing a time step such that meaningless calculations are avoided, yet accuracy is still preserved.

6.2.1 Approximation for the PTLA

As described in section 4.2, in the absence of continuous irradiation, the particle with the highest sum of event rates determines R_{max} and thus, the time step δt. In particular, if it happens that R_{max} takes a large value –for instance, when there are very mobile particles in the system– then the time step can take very small values. In heterogeneous systems where many defects of different types are present, it may occur that the density of these particles is very small in comparison to the total system density, and that they are not relevant for the physical processes that are under scrutiny in the present simulation. If R_{max} is very large compared to the sums of event rates for the remaining particles, it implies that the other particles –those with a lower event rate– will likely not perform any event during the time step δt. This is simply evidenced using Eq. 4.13.

Under these conditions, the system as a whole is not going to evolve significantly until a large number of computational steps have been undergone, which is clearly detrimental to performance. For instance, this situation arises in the investigation of defect evolution in Fe under neutron irradiation where a large amount of He atoms form by nuclear transmutation. Interstitial He impurities in Fe display a high mobility due to their small migration energy $E_m = 0.06\ eV$ [106, 108], which is much smaller than the migration energies of single Fe SIAs or vacancies. If the system contains millions of SIA and vacancies and there is only one interstitial He impurity, then the time step δt would take a very small value. Indeed, by the nature of our PTLA, the time step would be determined by the interstitial He impurity, the particle with the highest sum of event rates in the system. In the case given as an example, this would imply a time step of about $10^{-12}s$ at room temperature. It is easy to understand that in this case, a very large amount of computational steps should to be performed before any event involving SIAs or vacancies is triggered. Consequently, it is crucial to circumvent this issue to keep the runtime of the simulation as small as possible.

It is reasonable to assume that if the density of the particles with the highest sum of event rates is low enough, they should not significantly contribute to the evolution of the whole system. We can thus define a threshold ϵ, such that, if the density of the particles with the highest event rate is lower than $(\rho_{tot} \cdot \epsilon)$ –where ρ_{tot} represents the total density of the system– they can be neglected. It is important to note here that by 'neglected', we mean that these particles do not count concerning the computation of the time step, as described by Eq. 4.18. Instead, particles with a lower sum of event rates are considered to compute the value of the time step.

In our implementation, the list of particles present in the system is ordered in decreasing sums of event rates R_i. The fraction f_i is defined as the density of particles of type i with the sum of event rates R_i divided by the total density ρ_{tot}. Then, the cumulative function $F_i = \sum_k^i f_k$ is defined as the sum of particle fractions of higher and equal rates to those of type i. In other words, particles of type k must verify $R_k \geq R_i$. If $F_i \leq \epsilon$, the density of 'fast' particles up to type i is considered significant enough to define R_{max}. Otherwise, R_i is ignored for the calculation of R_{max}. The ordered list of particles is navigated until a large enough cumulative function $F_j \geq \epsilon$ is found for particles of type j. Then, the corresponding sum of event rates R_j is used to determine R_{max}.

This is summed up in Algorithm 4.

Algorithm 4 Choice of R_{max} in the PTLA. ϵ is the threshold defined for the significant fraction of particles with a large sum of event rates.

```
define $\epsilon$ such that $0 \leq \epsilon < 1$
**while** $F_j \geq \epsilon$ **do**
    identify event rates $R_i$ of each particle $i$
    order list of particles in decreasing $R_i$
    $f_i \leftarrow \frac{\rho_i}{\rho_{tot}}$
    $F_i \leftarrow \sum_k^i f_k$ s.t. $R_k \geq R_i$
**end while**
$R_{max} \leftarrow R_j$
```

As one can see, the further the list of particles is explored if the condition $F_i \leq \epsilon$ is not reached, the smaller R_{max} and therefore the larger δt becomes.

In order to demonstrate the usefulness of this approximation, the evolution of He in the presence of vacancies and the formation of $\mathrm{He}_n\mathrm{V}_m$ clusters in a Fe sample was investigated. Helium is known to be produced by nuclear transmutation in stainless steels subjected to neutron irradiation, as described in subsection 1.2.4. Once created, interstitial He atoms quickly migrate and get trapped by vacancies or recombine at the surface in a short time. As a result, the concentration of interstitial He in the system decreases rapidly.

A collection of collision cascades of He ions implanted in Fe at 5 keV were simulated with the MARLOWE code [242, 243, 244, 247]. A 3×10^{14} cm^{-2} dose was applied to a simulation box with $300a_0 \times 300a_0$ surface area and $7000a_0$ depth, equivalent to 2 μm. Periodic boundary conditions in x and y directions and absorbing boundaries in z direction were set for all species in the system. Then, the system was allowed to evolve for $t_{end} = 60$ s at 300 K (room temperature). Simulations with different threshold values ($\epsilon = 5\%$, $\epsilon = 2\%$, $\epsilon = 1\%$, $\epsilon = 0.5\%$ and $\epsilon = 0.2\%$) were performed on a NVIDIA®GeForce®GTX 1060 GPU and compared.

Figure 6.1 depicts the depth profiles of $\mathrm{He}_n\mathrm{V}_m$ clusters for the five values of ϵ. Notably, all profiles almost overlap, showing that the simulations are accurate even for higher values of ϵ. The runtimes that were necessary to achieve each simulation are reported in Table 6.1. In fact, it was not possible to achieve the reference simulation for which the approximation was not used ($\epsilon = 0$). The authors wish to emphasise that after seven days of calculations, only 2.2 μs of physical

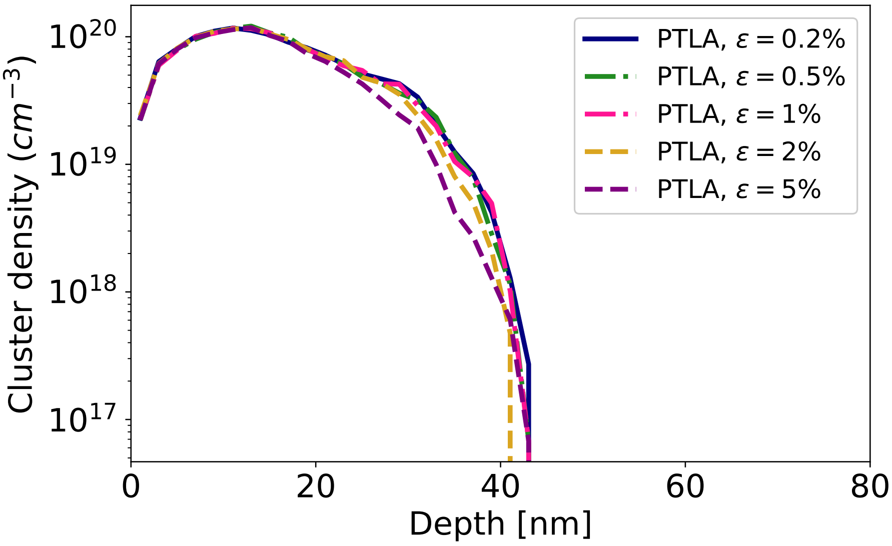


Figure 6.1: Depth profiles of $\mathrm{He}_n\mathrm{V}_m$ clusters for five different ϵ values in a simulation box of $7000a_0$ depth at 300 K.

time could be reached. In other words, millions of days of calculations would be required to achieve a few seconds of physical time. We estimate that since 7 days of runtime were needed to achieve 2.2 μs of simulation, the expected runtime required to achieve the final time $t_{end} = 60\ s$ would be $1.65 \times 10^{13}\ s$, i.e. 1.9×10^8 days or 5.2 million years. Obviously, it would be unfeasible to complete even a small fraction of the simulation within a sensible time frame. Our approximation reveals itself as a necessary addition to the MEGA-OkMC code in order to perform simulations of particle systems which include very mobile particles.

Indeed, at room temperature, the time step associated with the migration of interstitial He atoms is $\delta t = 10^{-12}\ s$. Clearly, a simulation without the approximation proposed here is not feasible since it would imply more than 10^{14} computational steps. With our approximation, the time step is calculated according to the migration of vacancies ($E_m = 0.67\ eV$) when the relative concentration of interstitial He atoms falls below the pre-established threshold ϵ. In practice, this leads to a time step of 0.018 s, which is ten orders of magnitude higher than the time step associated to the migration of He atoms. The increase in performance is noteworthy, without a noticeable loss of accuracy. Therefore, the introduction of the ϵ threshold

ϵ	Runtime (Tau-Leaping)	Speedup (Tau-Leaping)
0% (estimated)	1.65×10^{13} s (estimated)	1.00×
0.2%	4320.9 s	$3.82 \times 10^{9}\times$
0.5%	1357.3 s	$1.22 \times 10^{10}\times$
1.0%	790.7 s	$2.09 \times 10^{10}\times$
2.0%	665.8 s	$2.48 \times 10^{10}\times$
5.0%	636.4 s	$2.59 \times 10^{10}\times$

Table 6.1: Speedup comparison between simulations with different ϵ values, as depicted in Figure 6.1.

is justified as an acceptable, and sometimes necessary, approximation.

6.2.2 Approximation for GFRD

The first step to decide whether GFRD is valid is to check what particles have the largest sum of event rates. As explained in 6.1, the GFRD method does not offer any significant advantages if the migration of fastest particles does not dominate in comparison to other events. Thus, the maximum rate of migration $R_{mig,max}$ is compared to $R_{other,max}$ such that a PTLA is produced if $R_{other,max} \geq R_{mig,max}$. In case δt_{GFRD} has a smaller value than δt_{PTLA}, GFRD is not applicable, and a PTLA step is produced.

If GFRD is applicable, it might be possible to choose a larger time step than the δt_{GFRD} computed using the GFRD method if the corresponding particles are at low density. In this case, we defined the threshold η as follows. The ratio of the number of fastest particles N_f with respect to the total number of particles N is compared to η. If $\frac{N_f}{N} < \eta$, we consider that the presence of these particles is not significant enough to determine the evolution of the particle system. Thus, a larger time step may be chosen without a negative effect on the accuracy of the simulation.

To make sure that the algorithm keeps the assumptions behind GFRD valid, it is advisable to ensure that the second fastest particles may be considered immobile during the GFRD time step. More importantly, we need to make sure that the configuration of Pairs and Singles does not change within the GFRD step. In other words, when a Pair is defined with a fastest particle and a second fastest particle

–which is considered immobile in the construction of protective environments– the time step must be chosen in such a way that the second fastest particle does not move.

As mentioned above, δt_{2nd} is the characteristic time step for the events performed by the second fastest particles. In our approximation, this means that, whereas the factor η determines whether the time step δt may be selected such that $\delta t > \delta t_{GFRD}$, the characteristic time step δt_{2nd} is an upper threshold to δt. The time step may be chosen larger than δt_{GFRD} to accelerate the simulation, but not so large that the second fastest particles can evolve during that time step, and therefore violate the premise that they may be considered immobile. In summary, time step δt needs to verify $\delta t_{GFRD} \leq \delta t < \delta t_{2nd}$.

In MEGA-OkMC, the chosen time step in this situation is $\delta t = \frac{\delta t_{2nd}}{10}$. The probability that the second fastest particles will not perform an event within this time step $\frac{1}{10R_2}$ is 91%, according to the Poisson law shown in Eq. 4.2., and 9% of the second fastest particles are expected to perform one event. Even though at first glance, this seems like a considerable error which would accumulate with increasing simulation time, in practice it is an acceptable approximation, as we shall demonstrate with the following example.

Simulations were performed with the same initial conditions as in the previous subsection, i.e. a He dose of 3×10^{14} cm^{-2} dose was applied to a simulation box with $300a_0 \times 300a_0$ surface area and $7000a_0$ depth. We allowed the GFRD method to be triggered automatically if the situation allowed it. Since it was not possible to finish a reference PTLA simulation with $\epsilon = 0$, as explained previously, the new (GFRD) simulations were compared to the slowest PTLA simulation with $\epsilon = 0.2\%$ and depicted in Fig. 6.2. The values of η ranged from $\eta = 0.2\%$ to $\eta = 5\%$.

Table 6.2 shows the runtimes needed for each simulation and their speedup with respect to the estimated runtime of the PTLA simulation at $\epsilon = 0$.

6.2.3 Algorithm in the absence of continuous irradiation

Now that both approximations for PTLA steps and GFRD steps have been presented, we shall describe the execution of MEGA-OkMC when continuous irradiation is disabled. After δt_{GFRD} has been determined, it is compared to δt_{2nd}, the

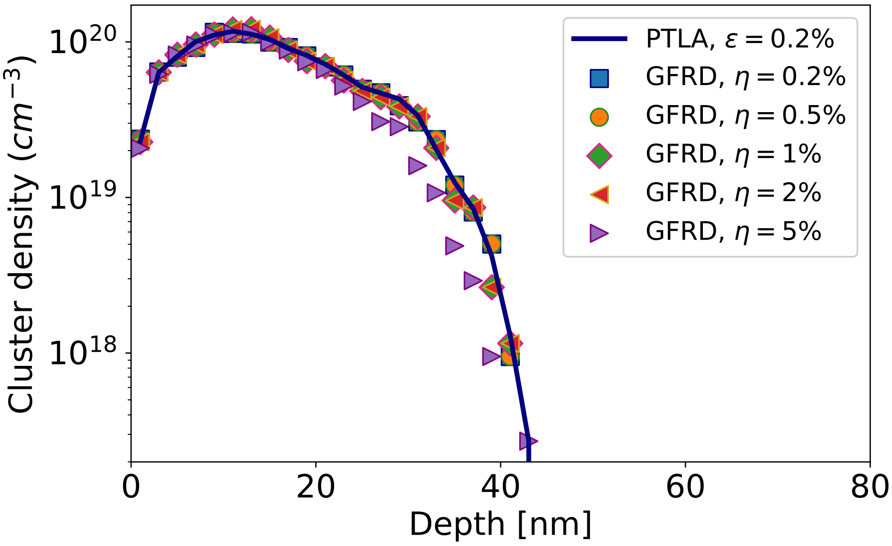


Figure 6.2: Depth profiles of $\mathrm{He}_n\mathrm{V}_m$ clusters for five different ϵ values in a simulation box of $7000a_0$ depth at 300 K. GFRD simulations (symbols) are compared to the slowest PTLA simulation (line) that could be performed, namely $\epsilon = 0.2\%$.

η	Runtime (GFRD)	Speedup (GFRD)
$\epsilon = 0\%$ (PTLA, estimated)	1.65×10^{13} s (estimated)	1.00×
0.2%	7264.7 s	$2.27 \times 10^9\times$
0.5%	1436.7 s	$1.15 \times 10^{10}\times$
1.0%	726.7 s	$2.27 \times 10^{10}\times$
2.0%	557.3 s	$2.96 \times 10^{10}\times$
5.0%	467.9 s	$3.53 \times 10^{10}\times$

Table 6.2: Speedup comparison between simulations with different ϵ values, as depicted in Figure 6.2.

time step associated to the sum of event rates of the second fastest particles. If $\delta t_{GFRD} > \delta t_{2nd}$, it is not possible to consider the second fastest particles immobile during time step δt_{GFRD}. In this case, GFRD calculations are repeated, focusing on the second fastest particles. The time step is recalculated, yielding $\delta t_{GFRD,2nd}$, and δt_{GFRD} is rewritten with this value.

Algorithm 5 Pseudocode that shows the first steps executed in MEGA-OkMC.

```
determine $R_{mig,max}$, $R_{other,max}$, $\delta t_{2nd}$
if $R_{mig,max} \leq R_{otner,max}$ then
    return do_PTLA_step
end if
compute $\delta t_{GFRD}$
if $\delta t_{GFRD} \geq \delta t_{2nd}$ then
    redraw protective environments, considering second fastest particles mobile
    compute $\delta t_{GFRD,2nd}$
    $\delta t_{GFRD} \leftarrow \delta t_{GFRD,2nd}$
end if
```

Afterwards, the criterion explained in section 6.2 is checked. Namely, if a GFRD step is produced, the fraction of fastest particles is compared to the threshold defined for the minimum significant fraction η. If the fraction of fastest particles happens to be below the established threshold η, a PTLA step is produced, and the chosen time step is $\frac{\delta t_{2nd}}{10}$. Instead, in case a PTLA step is produced, the list of particles is iterated through until the cumulative function F_i is greater than or equal to the threshold ϵ_i, as shown in Algorithm 4. The chosen time step is $\delta t = \frac{\omega}{R_i}$.

Algorithm 6 Pseudocode that shows our implementation of the approximations laid out in this section.

```
if $\frac{N_f}{N} \leq \eta$ then
    $\delta t \leftarrow \frac{\delta t_{2nd}}{10}$
    do_PTLA_step $\leftarrow$ true
else
    do_GFRD_step $\leftarrow$ true
end if
if do_PTLA_step then
    execute PTLA algorithm with chosen $\epsilon$
end if
if do_GFRD_step then
    execute GFRD algorithm with chosen $\eta$
end if
```

Finally, if a PTLA step has been produced, the interactions are performed using the space division algorithm explained in the Appendix. In the code, this is calculated by comparing the number of particles $n_{p,i}$ in each cell i to the number of particles allowed by the shared memory $n_{p,max}$, which depends on the hardware.

In case a GFRD step has been produced instead, the Single and Pair problems are solved in parallel. As described in Section 5.2.3, the two particles in a Pair problem have a chance to react that is given by $1 - S(\delta t)$ where $S(\delta t)$ is the survival probability and δt is the time step. A uniform random number $\xi \in (0, 1]$ is used to determine whether the particles interact.

Algorithm 7 Pseudocode that shows the final commands in the execution of PTLA or GFRD steps, as particles interact with each other.

```
if do_PTLA_step then
    execute interaction algorithm
end if
if do_GFRD_step then
    if Pair then
        generate $\xi \in (0, 1]$
        compute $S(\delta t)$
        if $\xi > S(\delta t)$ then
            execute interaction
        else
            execute events
        end if
    else
        if Single then
            execute events
        end if
    end if
end if
```

6.3 Choice of the GFRD time step in the presence of continuous irradiation: special case of electron irradiation

6.3.1 Solving the problem with continuous irradiation and the GFRD method

As mentioned in previous section 5.3, MEGA-OkMC includes the possibility of introducing particles into the sample during the simulation via continuous irradiation. Clearly, the choice of the time step affects how many particles enter the sample at each computational step.

Special care has to be taken when continuous irradiation is electronic. Irradiation flux in typical electron irradiation experiments are usually large and in the range $(10^{18} - 10^{21})cm^{-2}s^{-1}$ [16, 68, 221, 253, 302]. If only one electron on average were allowed to impact the sample on average per time step, the latter would become very small. Thus, the total number of computational steps –and, consequently, the runtime– would become prohibitively large. To avoid this problem, in our algorithm, many electrons are allowed to impact the sample at any given time step. Therefore, many defects –in general, Frenkel pairs– may be created inside protective environments at a time step when electron irradiation is enabled.

To check the validity of this choice, we simulated an electron irradiation experiment on Fe. Electron irradiation parameters were chosen equal to the conditions in the experiment by Arakawa et al. [16]. The energy of impinging electrons was 1 MeV and the irradiation flux was $\Phi = 9.2 \times 10^{18}\ cm^{-2}$. We chose an irradiation face of area $A = 1000a_0 \times 1000a_0$ which corresponds to $A = 8.18 \times 10^{-10}\ cm^2$. Material thickness was chosen equal to the experimental thickness, $z = 150\ nm \approx 520a_0$.

Under these conditions, if only one electron were allowed to strike the sample on average per time step, the corresponding rate of irradiation would be $R_{ext} = \Phi \cdot A = 7.53 \times 10^9\ s^{-1}$. Using the energy parameters calculated by Fu et al. [105], this rate is 2.78×10^5 times higher than the migration rate of SIAs in Fe at $T = 200\ K$ and 388 times higher at $T = 300\ K$. In fact, this rate is comparable to the rate of migration of prismatic DL $\langle 111 \rangle$ in Fe.

In addition, the cross-section was calculated as $\sigma_0 = 1.545 \times 10^{-21}\ cm^2$. Since the atomic density of α-Fe is $\rho = 8.46 \times 10^{22}\ cm^{-3}$, the mean number of PKA per electron is $n_{PKA} = \rho \cdot z \cdot \sigma_0 = 1.9 \times 10^{-3}$. This shows that only 0.19% of the electrons actually produce damage in the chosen sample.

The necessary calculations to create the protective environments around particles in the simulation box are performed. These are allowed to grow until two environments intersect, as explained in the previous chapter. The average times that particles need to arrive to the edge of their respective domains are then sampled. The time step δt_{GFRD} is thus determined. The following subsection details this process.

6.3.2 Algorithm in the presence of continuous irradiation

If continuous irradiation is enabled, and a PTLA step is produced, cascades are introduced into the sample according to a Poisson distribution in the chosen time step δt. Should there be any massive particles in the simulation box, then the interactions between massive particles with each other are performed, followed by the interactions of massive particles with other particles in their vicinity. Finally, the neighbour search algorithm from the Appendix is applied to all particles except the massive ones, and the interactions between them are done.

However, if a GFRD step is produced instead, the solution is more complex. The cascades are introduced in the sample in time step $\delta t = \delta t_{GFRD}$. Some of the new particles may appear inside the Single or Pair environments defined before. Here, we apply our solution shown in subsection 5.2.4. As we have seen, if the fastest particles move in one dimension, it is sufficient to redefine Singles and Pairs and solve these in δt, due to the fact that movement in one dimension is recurrent. On the other hand, if the fastest particles move in three dimensions, the code identifies unsolvable and solvable cells. Solvable cells are those where all protective environments contain one or two bodies in the new situation – that is, after the introduction of irradiation cascades – and unsolvable cells are those that contain at least one protective environment with three or more bodies. The protective environments are recalculated, taking into account the new particles created by irradiation.

Singles and Pairs in solvable cells are solved in parallel with time step δt. In each unsolvable cell i, Singles and Pairs are solved in sub-time steps $\delta t^{i,j}_{min}$ until the time step δt is reached, as described in Algorithm 2 presented in subsection 5.3.2. At the end of the GFRD step, events other than migrations are executed, such as dissociations. These steps are summarised in pseudocode in this Algorithm:

Algorithm 8 Pseudocode showing the execution of events and interactions in MEGA-OkMC, depending on whether a PTLA or GFRD step was produced.

```
introduce cascades in $\delta t$
if do_PTLA_step then
    execute events
    execute interactions
end if
if do_GFRD_step then
    if movement_fastest == 1D then
        redefine Single, Pair problems
        Create Single, Pair domains in 1D
    else
        if movement_fastest == 3D then
            Create Single, Pair domains in 3D
        end if
    end if
    execute events other than migrations
end if
```

The authors wish to emphasise the adaptive nature of our code. As one can see, the MEGA-OKMC code is able to decide which method to apply (PTLA or GFRD) in each time step, depending on the physical conditions of the simulation. In the following chapter, we shall present the experimental validation of MEGA-OkMC.

Chapter 7

Application of the MEGA-OkMC code to the simulation of experiments

Having developed a number of different computational techniques which were presented throughout this manuscript, in the last part of the present dissertation, we focused on the simulation of experiments with the MEGA-OkMC code.

In this chapter, the simulation of a resistivity recovery (RR) experiment is presented in Section 7.1. In Section 7.2.1, a Fe irradiation experiment carried out by Yao et al. [315] is introduced. This experiment is further described in subsection 7.2.2. Inputs used in the MEGA-OkMC code are presented in subsection 7.2.3 and simulation results are shown and discussed in subsections 7.2.4 and 7.2.5. Finally, a summary of benchmark results of MEGA-OKMC and two other OKMC codes is shown in section 7.3 in the context of the Multiscale Modelling for Fusion and Fission Materials (M4F) project.

7.1 Simulation of a resistivity recovery experiment

In order to demonstrate the efficiency of our model to simulate the evolution of a large amount of interacting particles in realistic conditions, we simulated a typical RR experiment in a Fe sample with realistic dimensions. This type of experiment is of importance as it allows to investigate the kinetics of defects in metals [121, 122, 204, 275, 285]. In a typical RR experiment, the sample is first cooled down to a very low temperature ($4 - 77\ K$) and is then irradiated with electrons, such that only Frenkel pairs (SIA-V) are produced. At these temperatures, SIAs and vacancies are immobile and thus their recombination is inhibited. Temperature is then increased by δT and kept constant for a fixed amount of time, in general in the order of a few minutes. Resistivity ρ is measured at the end of the isothermal annealing and the process is repeated until a preset temperature is reached. Since resistivity reflects the total number of defects that affect electron diffusion in the metal, RR experiments are a valuable tool to study the evolution of defects and the thermally-activated mechanisms they undergo. In practice, a typical RR curve exhibits several drops, reflecting the activation of recombination processes. Indeed, as temperature increases, the migration of the different defects gets thermally activated, allowing them to diffuse and recombine with defects of the opposite type. As a result, if the derivative of the total number of defects is plotted with respect to temperature, a typical RR spectrum exhibits several peaks, also called recovery stages, corresponding to the different thermally-activated mechanisms [275, 285, 214] that take place in the sample.

To simulate a typical RR experiment in Fe, we considered the evolution of SIAs and Vs in pure Fe, assuming that they both migrate and can recombine, as described in Section 4.3. The formation of SIA or V clusters or other mechanisms were neglected here. We emphasise that our goal is not to propose a physically-based model to explain the different features of the kinetics of defects in Fe but to show that, taking into account basic mechanisms and Fe parameters, our MEGA-OkMC code can be used to reproduce typical experiments using realistic simulation boxes representative of real samples with a large amount of particles and in short runtimes.

Here we consider a simulation box similar to that of a typical sample thickness in a RR experiment, with a thickness of about 20 μm ($70000a_0$). For the sake of simplicity, the effect of electron irradiation, i.e. the formation of Frenkel pairs during irradiation, was emulated by distributing randomly an equal number of interstitials and vacancies in the simulation box. In the present case we used 10 million SIAs and 10 million vacancies. The evolution of the system was then simulated for temperatures starting from 77 K up to 140 K with increase steps of $\frac{\delta T}{T} = 0.008$, and an annealing time $t_{end} = 300\ s$ at each temperature. Each RR curve simulated here consists thus of 77 simulations at different temperatures. In a similar fashion to the test cases shown in chapter 4, the size of the simulation box was changed in x and y directions in order to vary the initial concentration of particles, as their initial number (10 million SIAs and 10 million vacancies) was kept constant. Values for the initial density of Frenkel pairs range from $6.11 \times 10^{18}\ cm^{-3}$ to $6.11 \times 10^{20}\ cm^{-3}$. Like in previous cases, time step values were computed with $\omega = 1$. The GPU which was chosen in this example was a NVIDIA®GeForce®TITAN Black graphics card.

The derivative of the total number of defects as a function of temperature, for different initial concentrations, is shown in Figure 7.1. Here, two different peaks (recovery stages) can be observed, depending on the initial concentration. When the initial concentration is sufficiently high, our model predicts a first peak at about 107 K, which corresponds to the so-called I_D stage reported in literature [105, 275] for Fe. This stage corresponds to the recombination of correlated interstitials and vacancies, i.e. that are spatially close. When the migration of SIAs is thermally activated, the probability that they recombine with their vacancy after the first jump is relatively high, giving rise to a recombination at relatively low temperature. The temperature at which our model predicts this stage to appear is in perfect agreement with what is observed experimentally [275] and predicted theoretically [105].

Our model also predicts a second peak at a temperature of about $120 - 130\ K$, in very good agreement to what is reported in the literature [105, 275]. In contrast to previous stage I_D, this stage, commonly denoted stage I_E, corresponds to the recombination of uncorrelated SIA-V pairs. It is related to the recombination of interstitials that must perform several jumps before they can find a vacancy and recombine. As one can see in Figure 7.1, this peak shifts towards higher temperatures as initial concentration decreases. This is expected and experimentally observed [187] since for lower particle densities, the mean distance between particles increases and thus, interstitials need to perform more jumps to find a vacancy to recombine with. As a result, it becomes more difficult for interstitials to find a vacancy in a limited amount of time as the initial concentration decreases. Hence, higher temperatures are necessary for them to perform the required number of jumps before they can recombine with a vacancy, leading to the shift in temperature of stage I_E.

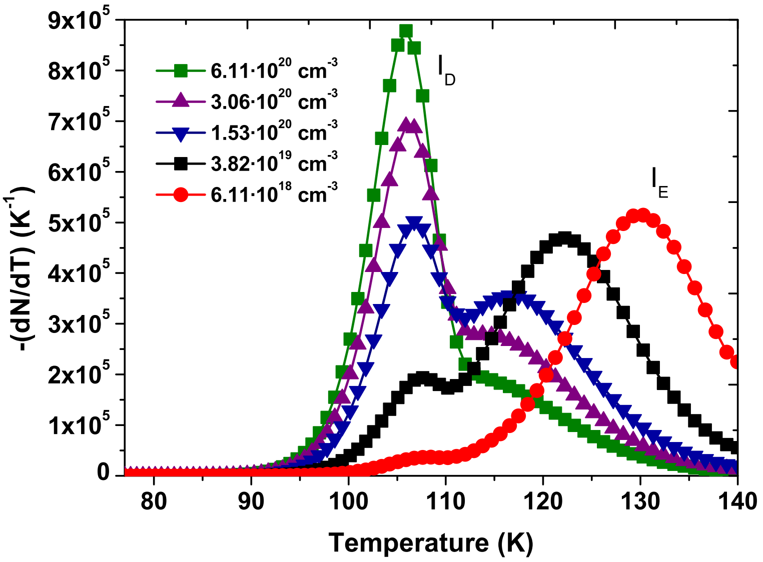


Figure 7.1: Resistivity recovery spectra in Fe simulated with MEGA-OkMC, corresponding to different initial SIA and V densities.

The runtimes needed to finish the simulations shown in Figure 7.1 are reported in Table 7.1. In each case the simulation was performed with 10 million SIAs and 10 million vacancies.

Simulation box	Initial density of Frenkel pairs (cm^{-3})	Runtime (s)
$100a_0 \times 100a_0 \times 70000a_0$	6.11×10^{20}	400.7
$100a_0 \times 200a_0 \times 70000a_0$	3.05×10^{20}	476.6
$200a_0 \times 200a_0 \times 70000a_0$	1.53×10^{20}	631.4
$400a_0 \times 400a_0 \times 70000a_0$	3.82×10^{19}	1270
$1000a_0 \times 1000a_0 \times 70000a_0$	6.11×10^{18}	3481

Table 7.1: Size of simulation box, initial concentration of Frenkel pairs and respective runtimes corresponding to Figure 7.1.

As one can see, our model, based on the PTLA and implemented on GPGPU, allows to simulate the evolution of a large number of particles and reproduce a realistic experiment in very reasonable runtimes. Where an OKMC model based on the standard BKL algorithm would likely require days of calculations, the PTLA algorithm that we developed and implemented using a GPU programming approach only requires computational times between 6 min and slightly less than 1 h in the worst case (lowest concentration).

7.2 Simulation of a self-ion irradiation experiment

7.2.1 Motivation to simulate a self-ion irradiation experiment

Understanding the effects of irradiation on steels is an important step in the development of novel nuclear materials. In the design of new fusion and Generation IV fission reactors, steels are the main structural material of choice, as seen in section 1.2.1. In the case of ITER [89, 141] and DEMO [21, 75, 101], the study of steels is of particular interest to predict embrittlement, hardening and creep in future fusion reactors [21, 75, 79, 167, 190, 201, 240, 304, 326, 328].

Self-ion irradiation experiments are performed to emulate the effect of neutron irradiation in materials [129, 222, 315]. The accessibility of ion sources compared to neutron sources makes self-ion irradiation an attractive alternative to monitor radiation damage in nuclear materials, as described in Section 1.3. Even though the fluence in typical ion irradiation experiments is higher than in neutron irradiation experiments [22, 23, 105, 204, 254], self-ion irradiation avoids the introduction of foreign ions that may interact with the defects in the sample and distort the results. Moreover, the investigation of irradiation effects in pure Fe and Fe-5%Cr is a case study of great interest to the MEGA-OkMC code, as it is possible to efficiently investigate a number of physical mechanisms and monitor their impact on the evolution of defects, making the study of complex diffusion-reaction systems in realistic conditions viable.

We shall now describe the Yao et al. experiment that was chosen as a study case for MEGA-OkMC.

7.2.2 Description of the Yao et al. experiment

The chosen experiment [315] was conducted at the IVEM-Tandem Facility at Argonne National Laboratory, USA.

In this experiment [315], thin Fe and Fe-5%Cr foils were irradiated with 100 keV Fe$^+$ self-ions. The experiment was carried out at room temperature and a constant flux $\Phi = 8 \times 10^{10} cm^{-2} s^{-1}$. The areal density of DLs was monitored with respect to dose, until a final dose of $d = 3 \times 10^{14} cm^{-2}$ in the case of Fe and $d = 10^{14} cm^{-2}$ in the case of Fe-5%Cr.

The impurity content of the Fe sample was 0.0013 $wt\%$, which is equivalent to a density of 2.38×10^{18} cm^{-3}. Eq. 7.1 [304] shows the equation used to convert impurity content from $wt\%$ to cm^{-3}, where ρ_C represents the density of carbon impurities, x_C the impurity content of the sample in percentage, ρ_s the number density of the sample (i.e. 8.46×10^{22} cm^{-3} for α-Fe [304]), and n_{Fe} and n_C are the respective atomic weights of Fe and C. It must be noted that there was an erratum in the original article [315] regarding the impurity content, which was corrected in a private communication. The Fe-5%Cr sample was reported to possess a higher purity than the α-Fe sample.

$$\rho_C = x_C \cdot \rho_s \cdot \frac{n_{Fe}}{n_C} \tag{7.1}$$

7.2.3 Input parameters in MEGA

The evolution of defects in OKMC simulations is a function of the input parameters. The most important parameters that govern the evolution of defects are:

1. Properties of the simulation box: crystal structure (sc, bcc or fcc), lattice parameter, box size, Debye frequency, sample orientation and material density. In the case of simulations with alloys, the solute content is also input as a property.

2. Radii of the various defects, expressed as either a constant or a function of defect type and cluster size.

3. Defect mobility: migration energies and dimensionality of motion.

4. Binding energies to defect clusters.

5. Interactions between defects and capture radii of said interactions.

In this work we focused on studying the mechanisms that may explain the development of DLs under self-ion irradiation of Fe and Fe-5%Cr as observed by Yao et al. using TEM. The 100 keV Fe$^+$ on Fe cascades were simulated with MARLOWE [244, 247], a BCA code.

We took a number of defects that may appear in the sample into consideration. In addition to single SIAs and vacancies, we considered the clusters that could be formed by these defects, i.e. I_n and V_n clusters. When I_n clusters reach a certain size, they may transform into prismatic DLs moving in one dimension. As is commonly accepted in literature [16, 17, 131, 234, 284, 289, 308, 309], here it is assumed that I_n clusters in Fe adopt a DL configuration for sizes $n \geq 5$.

As far as the interactions between defects are concerned, we introduced interstitial clustering and vacancy clustering, shown in Eq. 7.2 and 7.3 respectively, as well as recombination between interstitials and vacancies, shown in Eq. 7.4.

$$I_n + I_m \rightarrow I_{n+m}, n + m < 5 \tag{7.2}$$

Vacancy clustering:

$$V_n + V_m \rightarrow V_{n+m} \tag{7.3}$$

Recombination:

$$I_n + V_m \rightarrow \begin{cases} 0 \quad (n = m) \\ I_{n-m} \quad (n > m) \\ V_{m-n} \quad (n < m) \end{cases} \tag{7.4}$$

For our parameterisation of defect properties, we adopted the data provided in literature and used in the M4F project [176]. The migration (E_m) and binding energies (E_b) of defects were chosen as a compromise between the values provided by a number of different studies [58, 105, 143, 142, 176, 174, 178, 199, 284]. The migration energies of single SIAs and vacancies were selected as $E_m(SIA) = 0.3\ eV$ and $E_m(V) = 0.6\ eV$ respectively.

Binding energies of V_n clusters were computed according to Eq. 7.5, an extrapolation [176] of AKMC data provided by Castin et al [64].

$$E_b\left(V_n\right) = \left[1.71 - 2.76 \cdot \left((n+1)^{0.73} - n^{0.73}\right)\right]\ eV \tag{7.5}$$

Defect radii were calculated according to the following criteria. For vacancy clusters, Eq. 7.6 was used, which is the standard formula found in literature [65, 88, 176, 199] used to compute the radius of a spherical defect.

$$r_{X_n} = \left(\frac{3a_0^3}{8\pi}n\right)^{1/3} \qquad (X = V, C) \tag{7.6}$$

The radius of an interstitial cluster is calculated in this manner as well for single SIAs and clusters that are smaller than I_5, as shown in Eq. 7.7.

$$r_{I_n} = \left(\frac{3a_0^3}{8\pi}n\right)^{1/3} \qquad (n < 5) \tag{7.7}$$

As in other examples seen in chapters 4-6, each defined interaction between two defects has a capture radius associated to it, such that the interaction only occurs if the distance between both defects is less than the capture radius defined for them. Using data from simulations with SEAKMC code [312, 314], Nakashima et al. [199] proposed a recombination radius between Frenkel pairs of $r_{FP} = 2.26a_0$. In the same work, Eq. 7.8 was provided for the capture radius between a SIA and a vacancy cluster. These are the values that were used in our work.

$$r_c\left(I_1, V_m\right) = \left(0.86 \cdot m^{1/3} + 1.41\right) \cdot a_0 \tag{7.8}$$

As far as clustering between defects of the same type is concerned – i.e. interstitial clustering and vacancy clustering –, we chose the capture radii proposed by Jansson et al. [142, 143, 176] which take into consideration the geometry of each defect and the strain field around them. Eq. 7.9 yields the capture radius for interstitial clustering and Eq. 7.10 yields the one for vacancy clustering, where r_{I_j} and r_{V_j} $(j = n, m)$ are the geometrical radii of interstitial and vacancy clusters respectively, given by Eq. 7.7 and Eq. 7.6, with bias $\gamma = 1.2$ for interstitials and $\gamma = 1$ for vacancies.

$$r_c\left(I_n, I_m\right) = 2 \cdot \frac{\gamma \cdot r_{FP}}{\gamma + 1} + r_{I_{n-1}} + r_{I_{m-1}} \tag{7.9}$$

$$r_c\left(V_n, V_m\right) = 2 \cdot \frac{r_{FP}}{\gamma + 1} + r_{V_{n-1}} + r_{V_{m-1}} \tag{7.10}$$

For I_n and V_m clusters with $n > 1$, Eq. 7.11 applies.

$$r_c\left(I_n, V_m\right) = r_{FP} + r_{I_n} + r_{V_n} \tag{7.11}$$

For the simulation of FeCr alloys, our model follows a *grey alloy* approach [73, 74, 142, 143]. This means that the atoms of Cr solute are not represented specifically. Instead, in a *grey alloy*, the solute is assumed to be distributed uniformly across the simulation box. Hence, the effect of solute atoms on defect mobility is represented as a multiplicative prefactor $\nu = \frac{D_{FeCr}}{D_{Fe}}$ on the diffusion coefficient of each defect. It must be said that this *grey alloy* model does not allow for the portrayal of precipitates or distinct phases, since it is assumed that the uniform distribution of the solute is kept constant throughout the simulation.

In order to establish the prefactor ν needed to compute the diffusivities of SIAs and I_n clusters, the data provided by Chiapetto et al. [73] were taken into consideration. We performed a cubic spline interpolation of the available data given for a set of I_n clusters in order to calculate migration and binding energy values for the rest of interstitial clusters. Eq. 7.12 shows the function $\nu(n)$ obtained by interpolation. This function is shown in Fig. 7.2.

$$\nu(n) = \begin{cases} 8.3503 \cdot 10^{-4} \cdot x^3 - 2.5051 \cdot 10^{-3} \cdot x^2 - \\ \qquad - 1.9411 \cdot 10^{-1} \cdot x + 1.1958 & \text{if } x \in [1, 7] \\ -5.0793 \cdot 10^{-4} \cdot x^3 + 2.5697 \cdot 10^{-2} \cdot x^2 - \\ \qquad - 3.9152 \cdot 10^{-1} \cdot x + 1.6564 & \text{if } x \in (7, 19] \\ 7.4063 \cdot 10^{-5} \cdot x^3 - 7.4766 \cdot 10^{-3} \cdot x^2 + \\ \qquad + 2.3877 \cdot 10^{-1} \cdot x - 2.3355 & \text{if } x \in (19, 37] \\ -1.1300 \cdot 10^{-5} \cdot x^3 + 1.9988 \cdot 10^{-3} \cdot x^2 - \\ \qquad - 1.1182 \cdot 10^{-1} \cdot x + 1.9885 & \text{if } x \in (37, 61] \\ 7.6900 \cdot 10^{-7} \cdot x^3 - 2.0994 \cdot 10^{-4} \cdot x^2 + \\ \qquad + 2.2915 \cdot 10^{-2} \cdot x - 7.5107 \cdot 10^{-1} & \text{if } x \in (61, 91] \end{cases} \tag{7.12}$$

The formation of interstitial DLs may occur through a number of mechanisms, which we shall discuss now. DLs can be formed by growth of interstitial clusters or

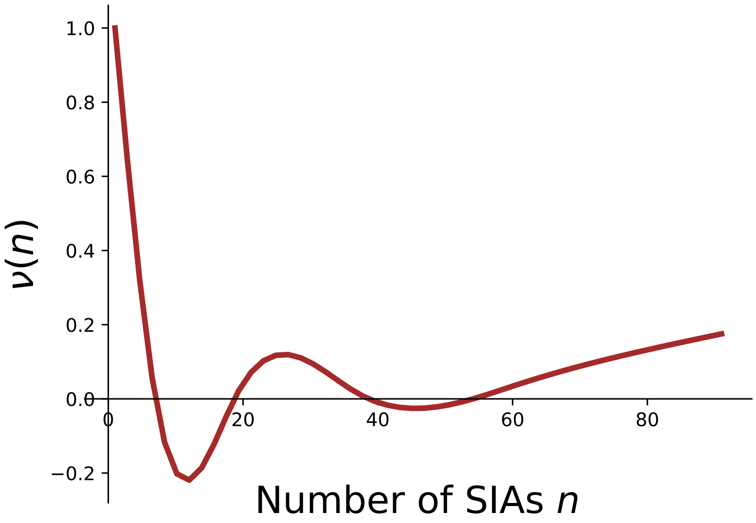


Figure 7.2: Function $\nu(n)$ shown in Eq. 7.12, obtained by cubic spline interpolation using the data given by Chiapetto et al. [73]

nucleation at impurities, as mentioned above [14, 16, 23, 71, 131, 142, 143, 176, 180, 217, 253, 277, 280, 284, 313].

DL formation through interstitial clustering:

$$I_n + I_m \to I_{n+m}\langle 111\rangle, n, m < 5, n+m \geq 5 \tag{7.13}$$

DL capture of small interstitial clusters:

$$\begin{aligned} I_n\langle 111\rangle + I_m \to I_{n+m}\langle 111\rangle, n \geq 5, m < 5 \\ I_n\langle 100\rangle + I_m \to I_{n+m}\langle 100\rangle, n \geq 5, m < 5 \end{aligned} \tag{7.14}$$

Clustering of two DLs:

$$I_n\langle\vec{b_1}\rangle + I_m\langle\vec{b_2}\rangle \to \begin{cases} I_{n+m}\langle\vec{b_1}\rangle(n \geq m) \\ I_{n+m}\langle\vec{b_2}\rangle(n < m) \end{cases} \tag{7.15}$$

Capture of DLs by impurities:

$$I_n\langle 111\rangle + C_k \to I_n\langle 111\rangle C_k \tag{7.16}$$

The binding energy of interstitial-type DLs was computed with the parameters provided by Alexander et al. [7] using ab initio calculations. The respective expressions for the binding energy of $I_n\langle 111\rangle$ and $I_n\langle 100\rangle$ $(n \geq 5)$ are given in Eq. 7.17 and Eq. 7.18.

$$\begin{aligned} E_{f,\langle 111\rangle}(n) &= \left(1.60485 \cdot \ln(n)\sqrt{n} + 5.35226\sqrt{n} - 0.147319\right)\ eV \\ E_{b,\langle 111\rangle}(n) &= E_{f,\langle 111\rangle}(n-1) + E_{f,SIA} - E_{f,\langle 111\rangle}(n) \end{aligned} \tag{7.17}$$

$$\begin{aligned} E_{f,\langle 100\rangle}(n) &= \left(1.77677 \cdot \ln(n)\sqrt{n} + 7.15951\sqrt{n} - 5.81801\right)\ eV \\ E_{b,\langle 100\rangle}(n) &= E_{f,\langle 100\rangle}(n-1) + E_{f,SIA} - E_{f,\langle 100\rangle}(n) \end{aligned} \tag{7.18}$$

When two SIA clusters $I_m, m < 5$ and $I_n, n < 5$ interact, the resulting cluster $I_{m+n}, m + n \geq 5$ is a DL which is mobile in one dimension with Burgers vector $\langle 111\rangle$ [105, 280, 284].

In case an interstitial cluster grows to DL size, namely I_n with $n \geq 5$, the formula in Eq. 7.19 is applied [142, 143, 176], with lattice parameter a_0.

$$r_{DL}(n) = a_0 \frac{\sqrt{n} - 1}{\sqrt{\pi\sqrt{3}}} \tag{7.19}$$

In case two DLs with different Burgers vectors react, the Burgers vector of the product is equal to the Burgers vector of the largest reactant. In other words, if two DLs I_m with Burgers vector $\vec{b_1}$ and I_n with Burgers vector $\vec{b_2}$ react, the product is a DL I_{m+n} with Burgers vector $\vec{b_1}$ if $m > n$ or Burgers vector $\vec{b_2}$ if $m < n$. In the particular case when two DLs of similar sizes with Burgers vector $\langle 111\rangle$ react, the resulting product may become a DL with Burgers vector $\langle 100\rangle$ [284]. This requires both interacting DLs to be of size I_{15} or larger. This is shown in Eq. 7.20.

$$I_n\langle 111\rangle + I_m\langle 111\rangle \rightarrow I_{n+m}\langle 100\rangle \quad \left(n, m \geq 15; \frac{|n-m|}{n} < 0.1\right) \tag{7.20}$$

7.2.4 Results and discussion: Fe$^+$ on Fe

We shall now present the simulation results. In order to monitor the concentration of DLs with respect to the dose, the number of DLs was recorded at different times.

DLs were expected to be found as I_n complexes with $n \geq 5$, either alone or trapped by carbon impurities or single vacancies. The smallest DLs which are visible to TEM are around the sizes I_{50} to I_{60} [133], i.e. those with a radius of about 0.8 nm as calculated with Eq. 7.19.

Case 1

In the first case, we shall consider that the only available mechanism for the formation of DLs is interstitial clustering. In order to help the readers visualize the physical situation of the particle system, Fig. 7.3 portrays the simulation box in MEGA-OkMC. Vacancies and their clusters are depicted in red, SIAs and interstitial clusters in white, carbon impurities in blue and vacancy-carbon complexes in yellow colour.

No DLs appeared in the sample other than a few small I_6 and I_7 clusters. In fact, during the course of the simulation, we noticed that once DLs were formed, they tended to be rapidly absorbed by the surface. All SIA clusters found in the sample were small and present on a residual level. Fig. 7.6 shows the evolution of SIAs and SIA clusters in this simulation. As one can see, the population of these defects was very small throughout the whole simulation – as their density was always lower than 10^{17} cm^{-3} – which confirms that the chosen conditions were not conducive to the formation of observable DLs. Fig. 7.5 displays the size distribution of SIA clusters for the final dose.

Closer inspection of individual defects shows that the conditions that would allow for a substantial population of DLs are not met. Fig. 7.7 shows the evolution of small interstitial clusters in time. As one can see, small interstitial clusters (I_2 to I_4) tend to disappear rapidly, i.e. the population of interstitial clusters never reaches a level which would be conducive to the formation of I_5 and visible DLs.

Fig. 7.8 shows the time evolution of small vacancy clusters in Case 1. One may notice that their density tends to decrease for higher doses, showing that vacancies gradually coalesce into larger vacancy clusters.

Clearly, this small population of SIA clusters does not correspond to the experimental data, as Yao et al. [315] were able to observe a population of DLs which grows with irradiation dose. Instead, in our simulation, DLs with Burgers vector

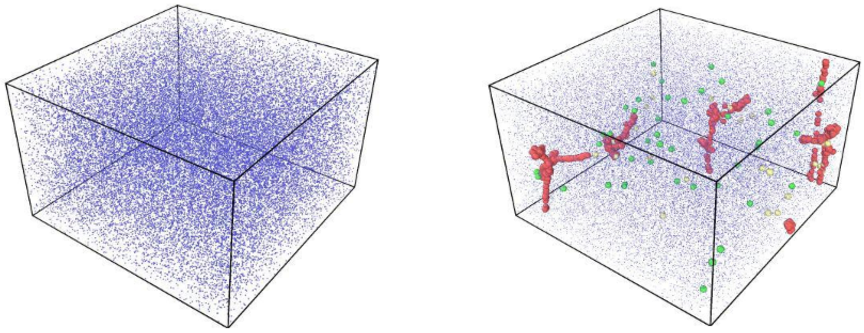


Figure 7.3: On the left, MEGA-OkMC simulation box at the start of the simulation. On the right, MEGA-OkMC box after 60000 computational steps. The top plane of the figure is plane (100). Vacancies and V_n clusters are coloured red, SIAs and I_n clusters are coloured green. C impurities and VC complexes are depicted in blue and yellow colour respectively. OVITO software [273] was used to display the simulation box in this and all following instances.

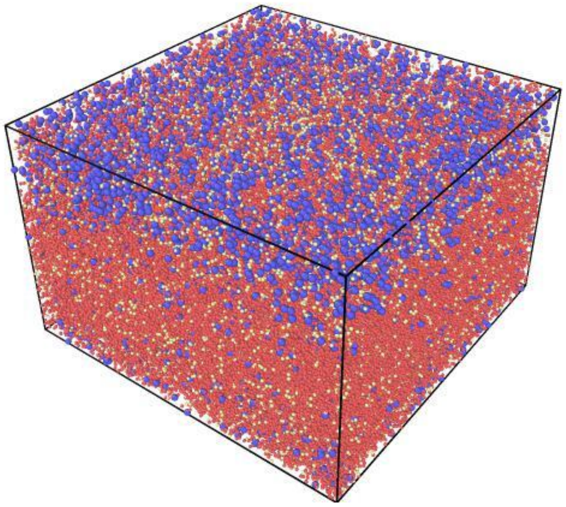


Figure 7.4: Simulation box at the end of the Case 1 simulation, i.e. for the final dose $d = 8 \times 10^{14}\ cm^{-2}$, as depicted with OVITO [273]. Most of the particles present at the end of the simulation are vacancy clusters (red). Interstitial clusters (green) and carbon impurities (blue) can be found as well.

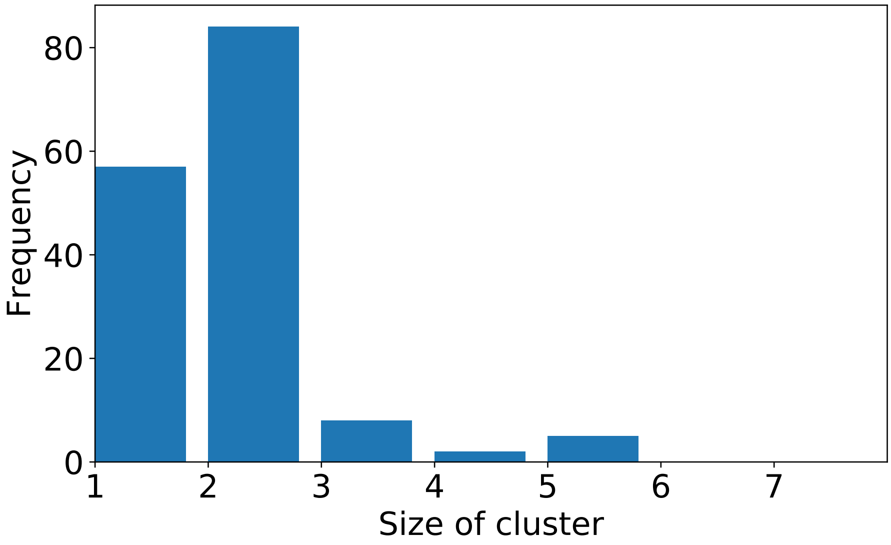


Figure 7.5: Histogram showing the size distribution of SIA clusters at the end of the Case 1 simulation. Only small I_n clusters were found, since prismatic DL loops tended to quickly recombine at the surface.

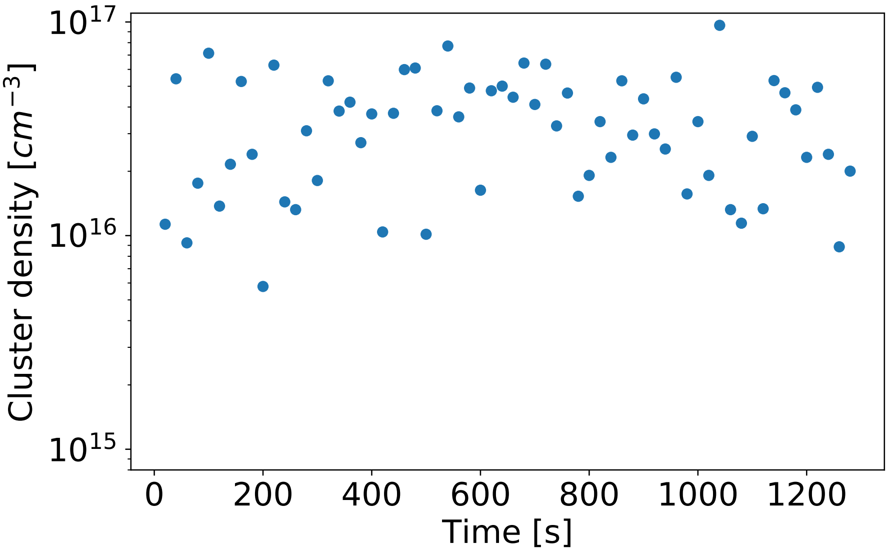


Figure 7.6: Evolution of SIA clusters in the Case 1 simulation. Under these conditions, SIAs and SIA clusters were only present on a residual level.

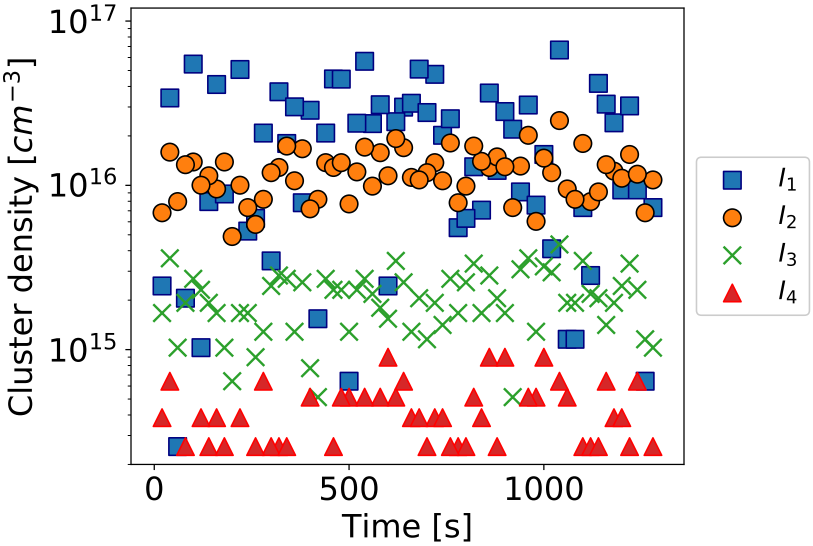


Figure 7.7: Evolution of SIAs and small I_2 to I_4 clusters in Case 1.

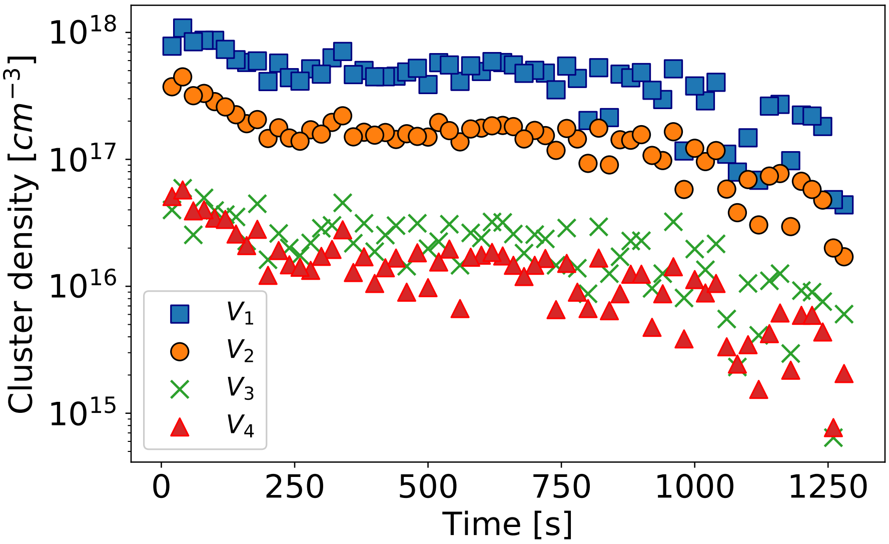


Figure 7.8: Evolution of vacancies and small V_2 to V_4 clusters in Case 1.

$\langle 111 \rangle$ leave the sample and do not get trapped or immobilised. Prismatic $I_n\langle 111 \rangle$ migrate to the absorbing surfaces where they disappear by recombination, and $I_n\langle 100 \rangle$ are not found at any point of the simulation, indicating that the formation of these DLs $\langle 100 \rangle$ does not occur. This suggests that a mechanism that temporarily immobilises $I_n\langle 111 \rangle$ DLs is necessary, as interstitial clustering alone may not explain their presence in the sample as the dose increases. One such mechanism may be the trapping of DLs by single vacancies, which takes us to case 2.

Case 2

Based on previous theoretical studies and MD calculations [14, 225, 236], single vacancies have been theorised to act as weak traps for interstitial DLs in bcc metals under certain conditions for a short time, provided the sizes of these DLs are large enough. In the case of Fe, this trapping effect has been calculated for I_{19} and larger SIA clusters [14, 225, 236].

Eq. 7.21 shows DL immobilisation by a single vacancy:

$$I_n\langle 111 \rangle + V \rightarrow \begin{cases} V \cdot I_n \text{ (under right conditions)} \\ I_{n-1}\langle 111 \rangle \text{ (otherwise)} \end{cases} \tag{7.21}$$

This mechanism was modelled according to the computations described by Refs. [14, 225, 236]. Vacancies are only able to trap DLs if the vacancy is located inside the cylinder described by the motion of the DL with a radius equal to the DL radius, otherwise both particles recombine if the distance between them is lower than the recombination radius between vacancy and DL. This is illustrated in Fig. 7.9. The DL is depicted in green colour, whereas the vacancy is depicted in red. On the left, the vacancy is able to trap the DL because of its location inside the cylinder; on the right, the vacancy is located on the edge of the cylinder, so recombination will happen.

For our second study case, trapping of DLs by single vacancies was enabled. With the introduction of this mechanism, our results show no formation of $V \cdot I_n$ complexes at any stage of the simulation. Furthermore, we searched the simulation box for DL which may have been formed as I_n clusters, and none were reported. As in Case 1, all SIA clusters disappeared from the sample quickly. In other words, vacancies were

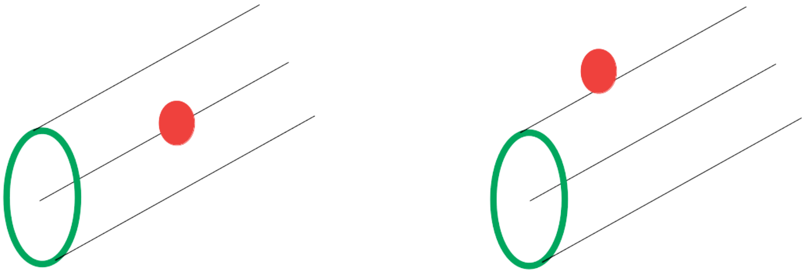


Figure 7.9: Sketch of a vacancy-DL interaction. On the left, the vacancy (red) is located inside the cylinder described by the radius of the DL (green) and its axis of motion (in the middle), and trapping may occur [14, 225, 236]. On the right, the vacancy is located on the edge of the cylinder, therefore both defects will recombine.

not able to trap prismatic DLs before these escaped to the surface. The evolution of SIAs and SIA clusters in this second case is illustrated in Fig. 7.10. Clearly, our results suggest that the presence of single vacancies does not enhance the nucleation of DLs. On the contrary, we found that the availability of vacancies hinders the formation of SIA clusters, due to the fact that single SIAs and small clusters tend to recombine with them. The necessary conditions shown in literature [14, 225, 236] proved too restrictive to result in any significant trapping of DLs – namely, the fact that only vacancies located inside the cylinder described in Fig. 7.9 can act as traps for DLs. Instead, vacancies are likelier to destroy SIA clusters by recombination than trap them.

It is of interest to observe the behaviour of vacancy clusters, as we have noticed that all SIA clusters vanish from the sample rapidly, whereas vacancy clusters remain. Fig. 7.12 displays the size distribution of vacancies and vacancy clusters at two different times, $t = 200\ s$ and $t = 1300\ s$, which is the end of the simulation. Under the present conditions, many vacancy clusters are formed under these conditions which evolve into larger clusters for higher doses, although most of them are small in size. This is best evidenced by monitoring the time evolution of small V_n and V_n clusters in Case 2, shown up to size V_4 in Fig. 7.13. These vacancy clusters act as a barrier to SIA cluster formation, and thus to DL formation.

We can safely conclude from this simulation that single vacancies acting as traps for DLs do not explain the experimental results obtained by Yao et al. [315] such

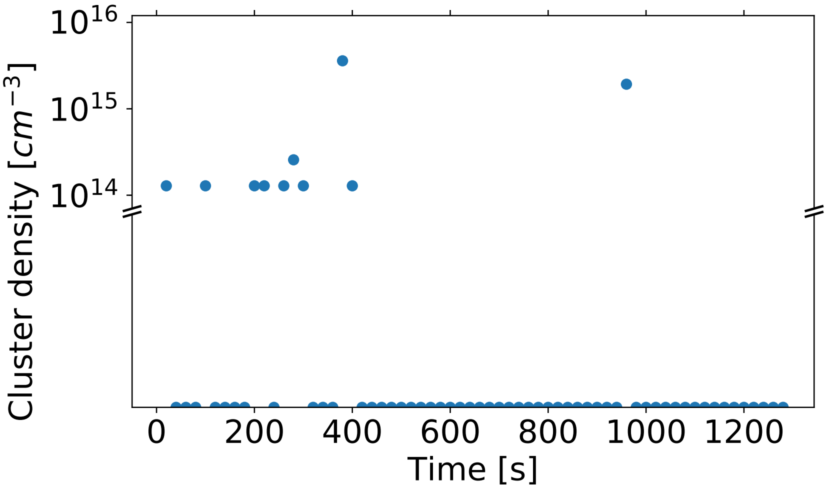


Figure 7.10: Evolution of SIAs and SIA clusters in Case 2. All defects of this type disappeared quickly by recombination at the surface or with vacancies.

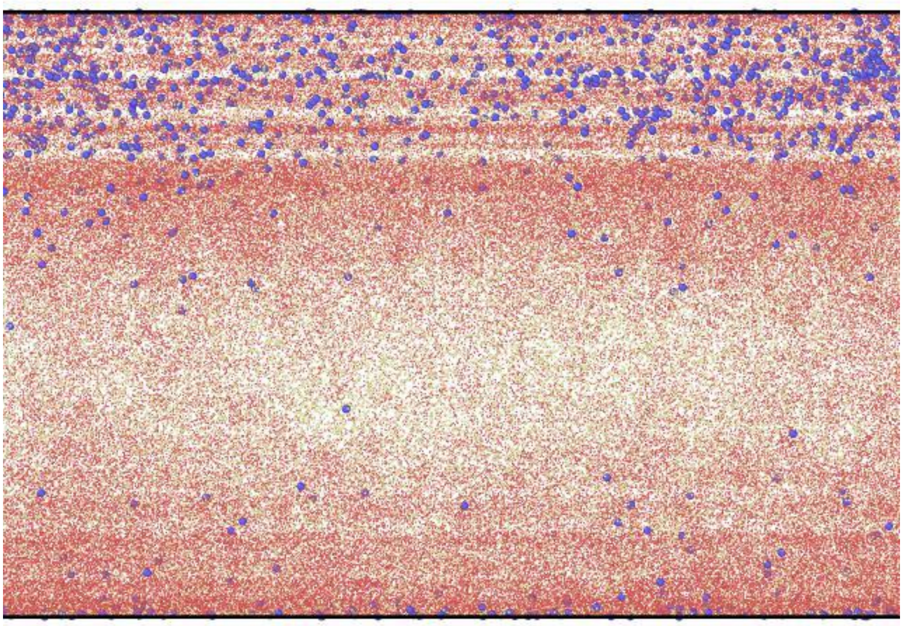
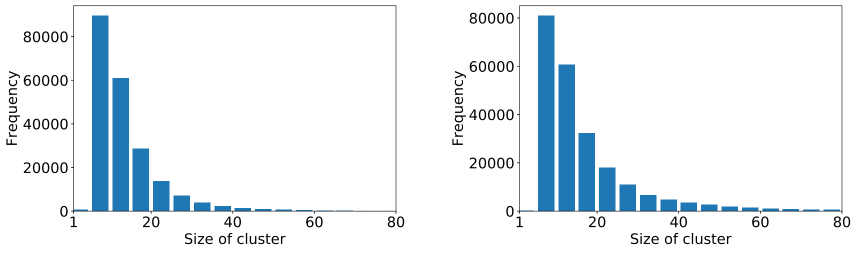


Figure 7.11: End state of the simulation box in Case 2, depicted with OVITO [273], as viewed from the plane (100).



(a) $t = 200\ s$ (b) $t = 1300\ s$

Figure 7.12: Size distribution showing V_n clusters at $t = 200\ s$ (left) and $t = 1300\ s$ (right) in Case 2.

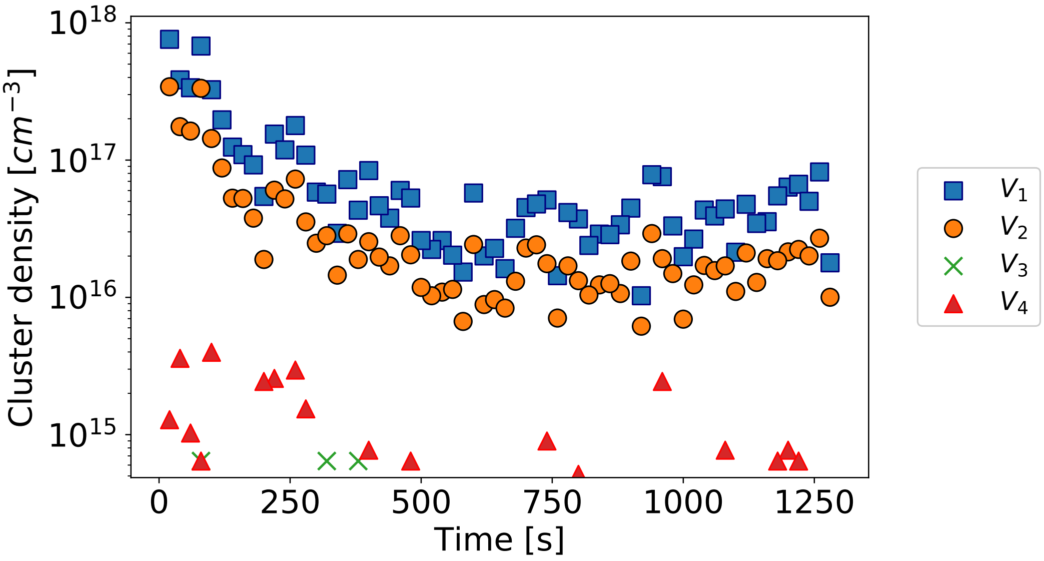


Figure 7.13: Evolution of vacancies and small V_2 to V_4 clusters in Case 2.

that a different mechanism is needed. This brings us to Case 3.

Case 3

Another mechanism that could explain the experimental observations by Yao et al. is the interaction of defects with impurities. As mentioned in subsection 7.2.2, the sample contains an impurity content of 0.0013 $wt\%$. Impurities are known to act as nucleation points for interstitial and vacancy clusters [14, 142, 143, 176, 253, 277, 280, 284].

Hence, we considered the following interactions between impurities and SIAs or vacancies:

Interstitial trapping by impurity:

$$I_n + C_m \rightarrow I_nC_m \tag{7.22}$$

Vacancy trapping by impurity:

$$V_n + C_m \rightarrow V_nC_m \tag{7.23}$$

Interstitial clustering at impurities:

$$I_n + I_kC_m \rightarrow I_{n+k}C_m \tag{7.24}$$

Vacancy clustering at impurities:

$$V_n + V_kC_m \rightarrow V_{n+k}C_m \tag{7.25}$$

The radii of impurity clusters were calculated with Eq. 7.6, which is, as mentioned above, the formula that is typically used to calculate the radius of a spherical defect.

As far as the capture of DL by impurities is concerned, impurities may contribute to the trapping of DLs, as shown in literature [14, 16, 23, 71, 131, 142, 143, 176, 180, 217, 253, 277, 280, 284, 313]. This mechanism was studied by Jansson and Malerba [143] in Fe-C alloys. In their work, the effect of C impurities and VC complexes on the motion of vacancy and interstitial clusters was simulated with the introduction of traps. Instead, in the present work, C impurities are introduced explicitly in the sample, as particles that may interact with defects in their vicinity. With this method, C atoms are able to migrate and react, resulting in the creation of C_n clusters, as well as leave the sample by recombination at the surface.

The capture radius between an impurity and an interstitial cluster is given by Eq. 7.26 which is equal to the radii of both defects plus one nearest-neighbour jump. This is the standard formula used for impurity trapping in literature [142, 176, 199].

$$\begin{aligned} r_c(I_n, C_k) &= r_{I_n} + r_{C_k} + \frac{\sqrt{3}}{2}a_0 \\ r_c(V_n, C_k) &= r_{V_n} + r_{C_k} + \frac{\sqrt{3}}{2}a_0 \end{aligned} \tag{7.26}$$

Using this mechanism, large DLs were found in the simulation box, which in all cases existed in the form of $I_n - C$ complexes.

Fig. 7.14 shows the simulation box in MEGA at the end of Case 3.

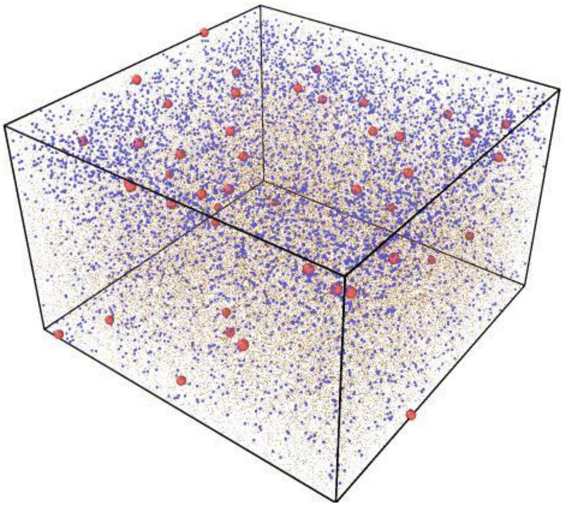


Figure 7.14: Simulation box at the end of the Case 3 simulation, depicted with OVITO [273].

Our simulation shows an overestimation of the DLs found experimentally, as evidenced by the fact that the curves showing $I_n - C$ complexes for $n \geq 50$ and $n \geq 80$ in Fig. 7.15 are clearly above the curve depicting the experimental results. In fact, vacancies were shown to coalesce into a few vacancy clusters, which may explain the large number of $I_n - C$ complexes found, as SIAs and their clusters find no available vacancy clusters to recombine with. Fig. 7.16 shows the time evolution of vacancies and vacancy clusters in Case 3.

In fact, this tendency that vacancies have to form larger vacancy clusters was observed in Cases 1 and 2. However, unlike in those two cases, the presence of carbon impurities was able to assist the formation of DLs. This effect can be explained by the fact that small $I_n - C$ defects are created by trapping of small DLs by carbon impurities, which are afterwards able to grow into larger $I_n - C$ clusters. Fig. 7.17 illustrates the time evolution of small $I_{10} - C$, $I_{15} - C$ and $I_{20} - C$ clusters in Case 3. A small population of $I_n - C$ clusters forms at low dose which then grows to

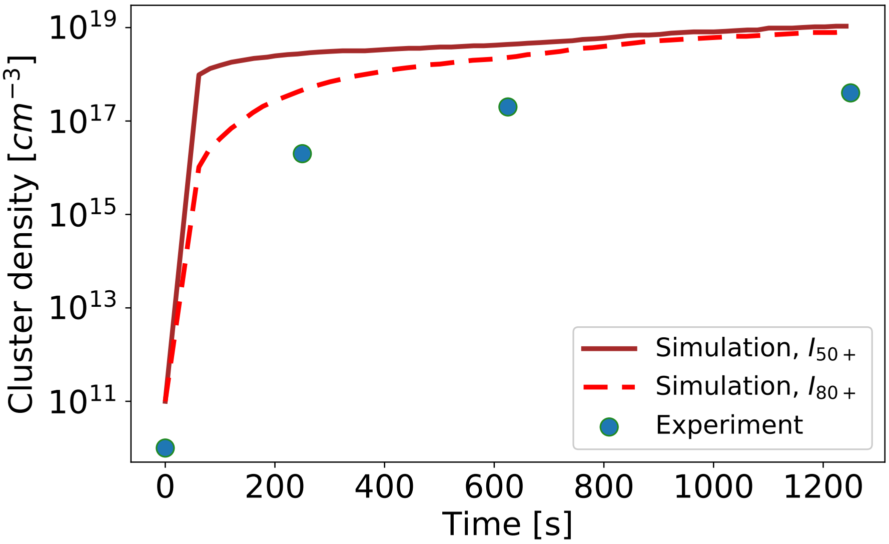


Figure 7.15: MEGA-OkMC results for Case 3, compared to the experimental results [315]. All DLs were found in the form of $I_n - C$ complexes.

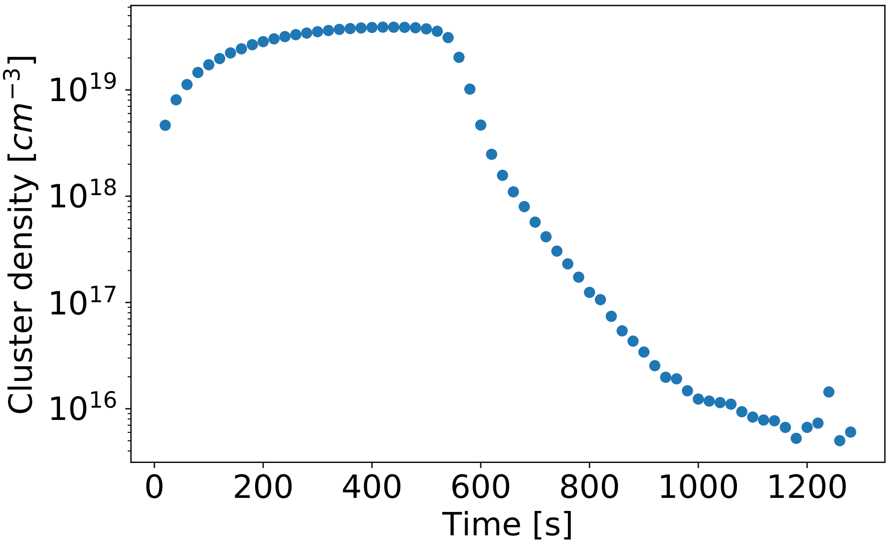


Figure 7.16: Evolution of vacancies and vacancy clusters, Case 3. Vacancy clusters are shown to coalesce into a few large V_n clusters.

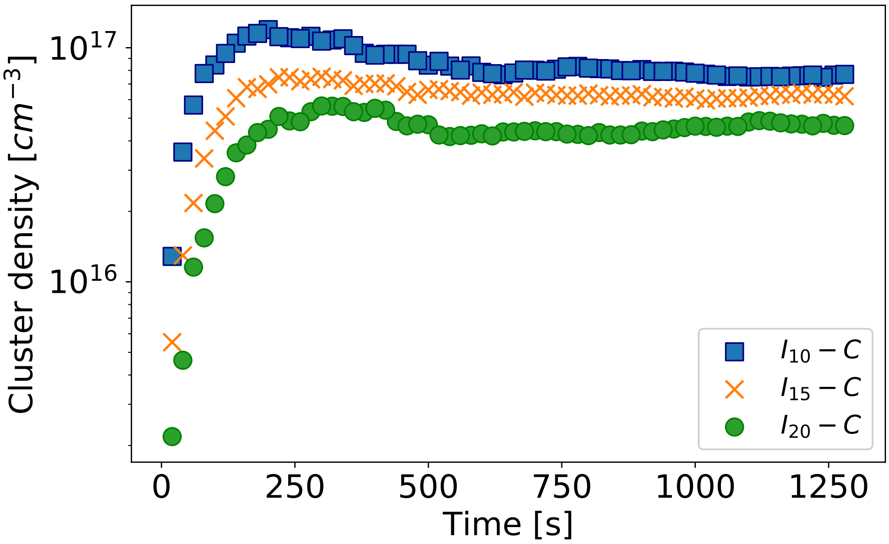


Figure 7.17: Evolution of small $I_n - C$ clusters in Case 3.

a stable level. Larger clusters are thus able to form as these small $I_n - C$ defects trap more DLs. Hence, the result of introducing DL trapping by C impurities is consistent with established theoretical literature [14, 253, 277, 280], which predicts that $I_n - C$ clusters can act as nucleation points for larger clusters.

In the fourth and last case we considered for Fe^+ on Fe irradiation, Case 4, we took both trapping mechanisms into account, namely trapping of DLs by single vacancies and trapping by impurities.

Case 4

In our final study case for Fe^+ on Fe irradiation, all mechanisms present in Cases 2 and 3 were included. In other words, we allowed DLs to be formed by interstitial clustering, as well as by clustering at carbon impurities and single vacancies.

We noticed a remarkable similarity with the results of Case 3, which we shall discuss now. As well as in Case 3, DLs were also only found attached to C atoms, as I_nC complexes. Fig. 7.18 shows the evolution of said I_nC complexes in time.

In addition, the evolution of vacancy-type defects, shown in Fig. 7.19, was also very similar to their behaviour in Case 2.

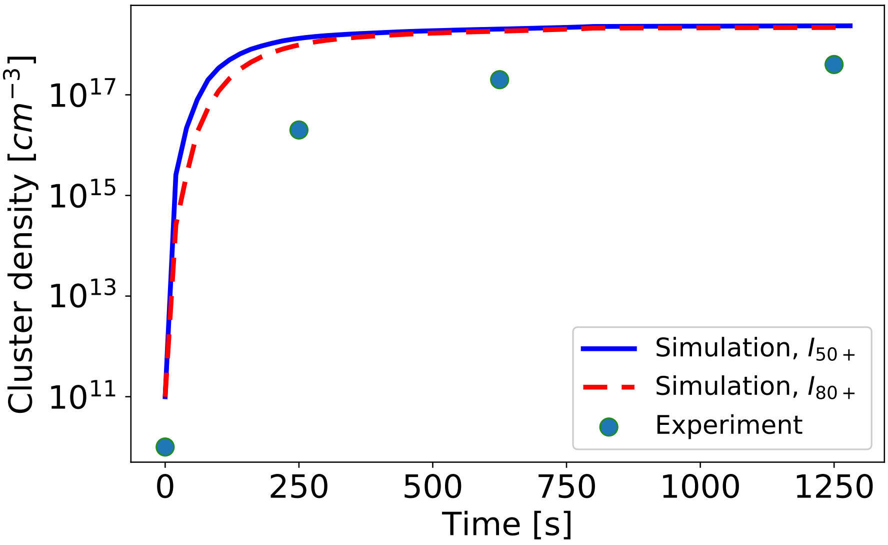


Figure 7.18: Evolution of I_nC complexes in Case 4, $n \geq 50$.

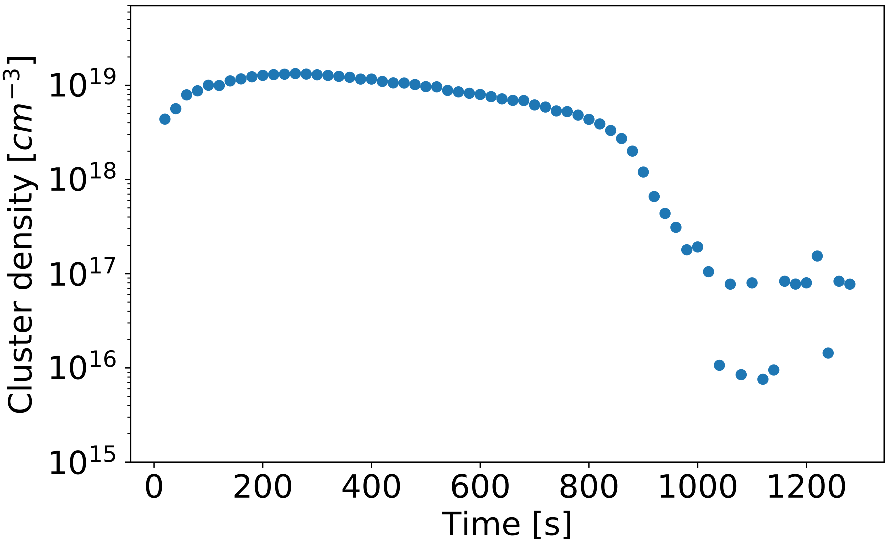


Figure 7.19: Evolution of vacancy-type defects in Case 4.

Our results demonstrate that impurities play a crucial role in the creation of DLs in Fe. This observation supports the results found in literature [142, 143, 253, 277, 280] describing the role of impurities in DL trapping. On the other hand, the effect of single vacancies as weak traps does not contribute to the formation of DLs;

Defect	E_m [eV]	E_b [eV]	Defect radius [nm]	Dimensionality of motion
I_1	0.30	N/A	0.14	3D
I_2	0.40	0.80	0.18	3D
I_3	0.40	0.92	0.20	3D
I_4	0.40	1.64	0.22	3D
$I_n\langle 111\rangle, n \geq 5$	0.10	Eq. 7.17	Eq. 7.19	1D
V_1	0.60	N/A	0.14	3D
V_2	0.60	0.13	0.18	3D
V_3	0.40	0.27	0.20	3D
V_4	0.50	0.37	0.22	3D
V_5	0.70	0.44	0.24	3D
V_n	1.00	Eq. 7.5	Eq. 7.6	3D
C_1	0.86	N/A	0.14	3D
$C_n, n \geq 2$	Immobile	Stable	Eq. 7.6	N/A
$I_nC_1, n < 5$	Immobile	Unstable		N/A
$I_nC_1, 5 \leq n \leq 19$	Immobile	0.40	Eq. 7.7	N/A
$I_nC_1, 20 \leq n \leq 50$	Immobile	0.70	Eq. 7.7	N/A
$I_nC_1, 51 \leq n \leq 90$	Immobile	0.80	Eq. 7.7	N/A
$I_nC_1, 91 \leq n \leq 150$	Immobile	0.80	Eq. 7.7	N/A
$V \cdot I_n\langle 111\rangle, n \geq 5$	Immobile	Ref. [14]	Eq. 7.7	N/A
$V \cdot I_n\langle 100\rangle, n \geq 5$	Immobile	Ref. [14]	Eq. 7.7	N/A

Table 7.2: Energies and defect radii of the defects considered in the MEGA-OkMC simulations of the Yao et al. experiment [315].

rather, the abundance of vacancies acts as an obstacle to interstitial clustering. Therefore, we may conclude that impurities act as nucleation points for DLs in self-ion irradiation of pure Fe.

Table 7.2 shows a summary of the energies and defect radii used in our input.

7.2.5 Results and discussion: Fe$^+$ on Fe-5%Cr

As in subsection 7.2.3, we included interstitial clustering, vacancy clustering and recombination between SIA and vacancy clusters, represented by Eqs. 7.2, 7.3 and 7.4 respectively.

Two cases were contemplated, similar to Cases 1 and 3 in subsection 7.2.4.

Case 1

As well as in the first case of subsection 7.2.4, impurities were ignored and the only available method for the production of DLs was interstitial clustering, described by Eq. 7.13.

We noticed that, as well as in Case 1 of the simulations with α-Fe, SIAs did not tend to cluster and form DLs. Instead, they recombined quickly with vacancies and their clusters. Fig. 7.20 depicts the evolution of vacancies and vacancy clusters in time.

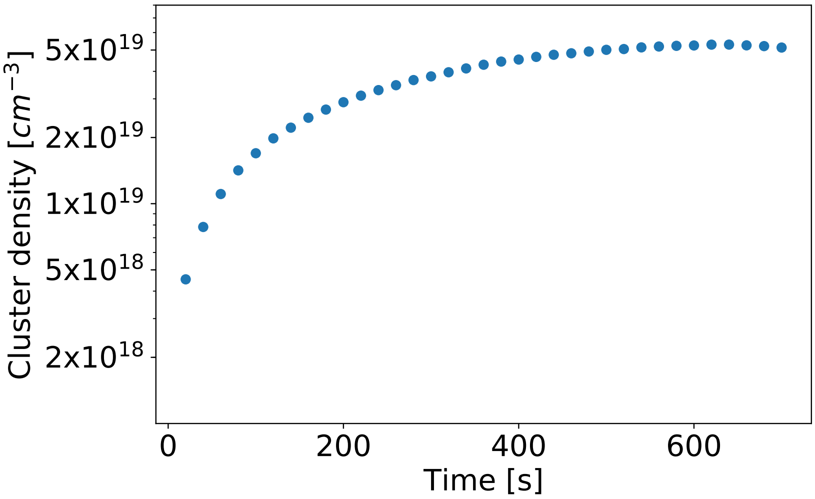


Figure 7.20: Time evolution of vacancies and their clusters in Case 1.

The time evolution of small interstitial and vacancy clusters may help us understand why DLs were not found at any point of the simulation. These are shown respectively in Fig. 7.21 and Fig. 7.22 up to I_4 and V_4. It is easy to notice that the formation of large interstitial and vacancy clusters is not brought about by the present conditions, as even the population of I_3 and V_3 defects is too small. These small clusters are eventually destroyed by recombination, such that in fact, I_4 clusters were not found at any point of the simulation, and V_4 clusters were only formed at the start.

These results we obtained for Case 1 contradict the experimental results and demonstrate the necessity of a trapping mechanism that allows SIA clusters to stay in the sample, and thus allows for a population of DLs to be built up over time.

Since trapping of DLs by impurities could explain the fact that DLs were present in the simulation box in α-Fe, as shown in the previous section, we chose to introduce this mechanism in Case 2.

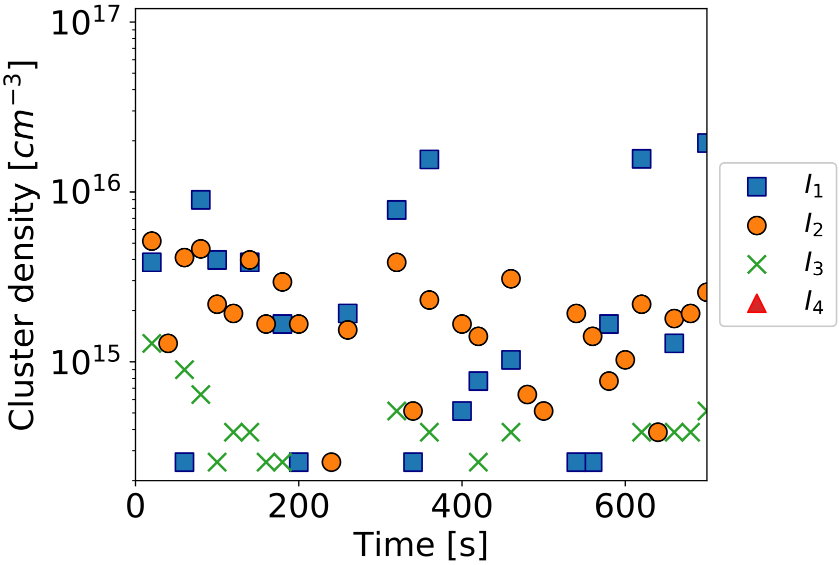


Figure 7.21: Evolution of SIAs and small interstitial clusters in Case 1.

Case 2

In our second case, impurities were allowed to act as traps for DLs. The purity of the FeCr sample was described as higher than the Fe sample studied in Section 7.2.1. To reflect this fact, an impurity content of 0.001% was introduced in the simulation box, lower than the impurity of the previously studied Fe sample.

In this example, all DLs were found in the form of $I_n - C$ complexes, as in Cases 3 and 4 of the α-Fe simulations. We depicted the time evolution of DLs which were visible to TEM – i.e. DLs of size I_{50} and larger – in Fig. 7.23 and found our results to be in fair agreement with the experiment.

Indeed, similarly to Case 3 of the α-Fe simulations, the presence of DLs may be explained by C impurities. As in the previous cases, we decided to monitor several small $I_n - C$ clusters – namely, I_{10}, I_{15}, I_{20} – whose evolution is shown in Fig. 7.24. Once again, we noticed that the populations of these small clusters build up rapidly

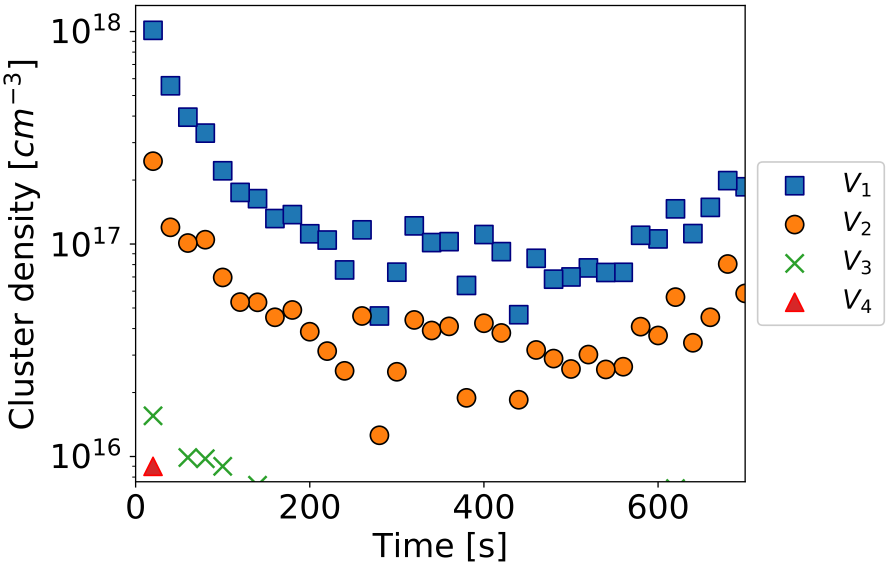


Figure 7.22: Evolution of single vacancies and small vacancy clusters in Case 1.

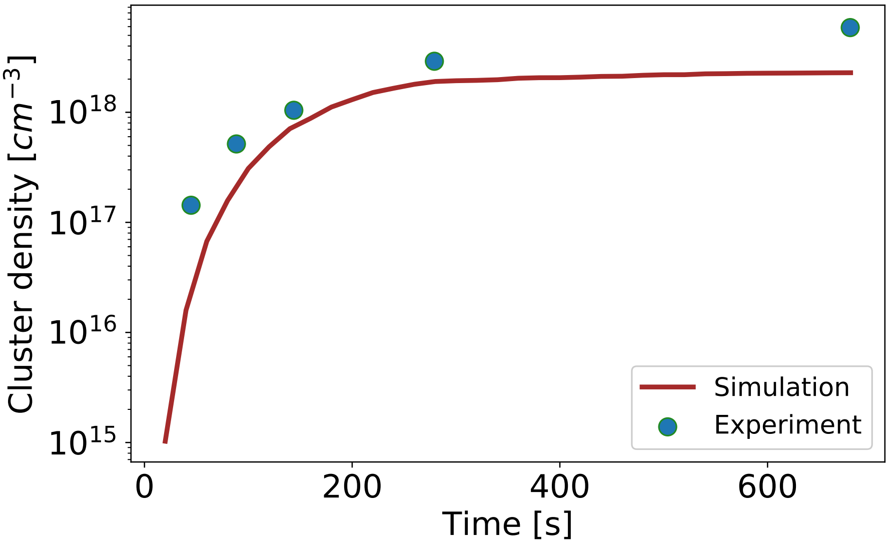


Figure 7.23: Time evolution of DLs in Case 2, size I_{50} and larger. All DLs were found in the form of $I_n - C$ complexes.

at low dose until they reach a stable level that allows larger DLs to be formed by trapping. As we have seen, this result is consistent with the known properties of carbon impurities in steels [14, 73, 86, 142, 277]. We can therefore confidently conclude that carbon impurities play an important role in DL trapping, both in α-Fe and Fe-5%Cr.

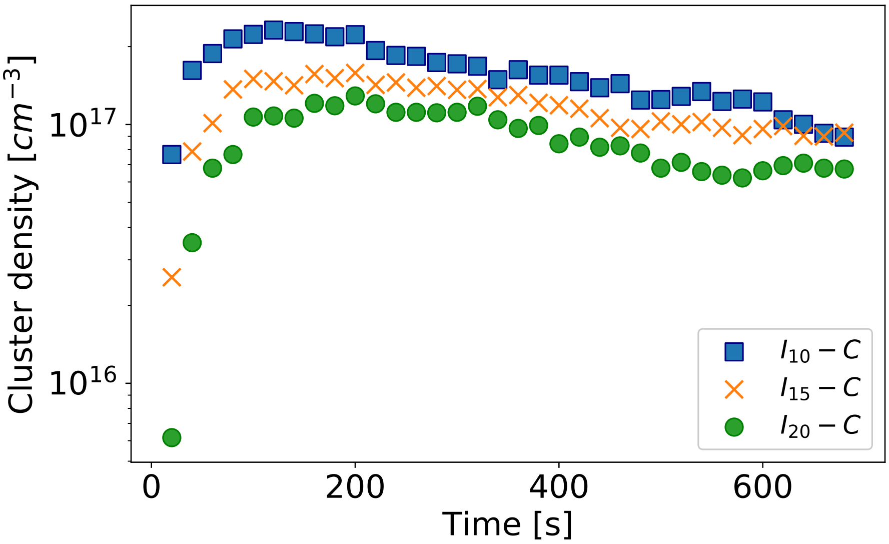


Figure 7.24: Time evolution of DLs in Case 2. All DLs were found in the form of $I_n - C$ complexes.

7.2.6 Summary

As a final comment on these results, the authors believe that a summary of the mechanisms taken into account for each of the simulation cases would be helpful to the reader. In this work, different mechanisms that could lead to the formation of DLs were taken into consideration, which were grouped into four cases in the case of α-Fe:

1. As a first approach, we only allowed DLs to be formed by clustering of SIAs and small SIA clusters, according to Eq. 7.13.

2. In the second case, the effect of vacancies as weak traps for DLs was monitored. In our simulation, this mechanism was simulated through the implementation

of $V \cdot I_n$ complexes with a small binding energy which decreases with n [14]. Hence, the second case includes interstitial clustering and DL trapping by vacancies as possible mechanisms for DL growth.

3. In the third case, we included the trapping of DLs by carbon impurities, in addition to interstitial clustering. For this purpose, we added an initial distribution of carbon impurities.

4. In the fourth and final case, both trapping mechanisms, namely carbon impurities and single vacancies, were considered, in addition to interstitial clustering. Our purpose was to explore the possibility of a synergy between DL formation by nucleation at impurities and nucleation at vacancies.

In Fe-5%Cr simulations, two cases were considered:

1. In Case 1, the only mechanism that would potentially lead to the formation of DLs was clustering of SIAs and small SIA clusters. This is similar to the conditions present in Case 1 for α-Fe.

2. In Case 2, carbon impurities were also allowed to act as traps. For this purpose, an initial distribution of carbon impurities was created in the simulation box. This is similar to the conditions present in Case 3 for α-Fe.

The summary of interactions for α-Fe and Fe-5%Cr are shown in Table 7.3 and Table 7.4 respectively.

7.3 Benchmarking of MEGA-OKMC with similar codes

Our work was presented as one of the modelling tools in the M4F project [176], in the same Work Package as two other OKMC codes, namely MATEO and MMonCa [176, 183]. In order to demonstrate that all three codes are consistent with each other regarding defect migration, defect interaction and the advancement of time [25, 176], efforts were made in this Work Package in the form of simple benchmark cases. Whereas a more comprehensive description of these cases may be found in Ref. [25],

Case	Interactions	References
Case 1	Interstitial clustering $I_n + I_m \rightarrow I_{n+m}$ DL immobilisation $I_n\langle 111\rangle + I_m\langle 111\rangle \rightarrow I_{n+m}\langle 100\rangle$	[105, 199, 280, 284] [284]
Case 2	Interstitial clustering $I_n + I_m \rightarrow I_{n+m}$ DL capture by C impurities $I_n + C \rightarrow I_n - C$ DL immobilisation $I_n\langle 111\rangle + I_m\langle 111\rangle \rightarrow I_{n+m}\langle 100\rangle$	[105, 199, 280, 284] [142, 143, 176] [284]
Case 3	Interstitial clustering $I_n + I_m \rightarrow I_{n+m}$ DL capture by single vacancies $I_n + V \rightarrow V \cdot I_n$ DL immobilisation $I_n\langle 111\rangle + I_m\langle 111\rangle \rightarrow I_{n+m}\langle 100\rangle$	[105, 199, 280, 284] [14, 225, 236] [284]
Case 4	Interstitial clustering $I_n + I_m \rightarrow I_{n+m}$ DL capture by C impurities $I_n + C \rightarrow I_n - C$ DL capture by single vacancies $I_n + V \rightarrow V \cdot I_n$ DL immobilisation $I_n\langle 111\rangle + I_m\langle 111\rangle \rightarrow I_{n+m}\langle 100\rangle$	[105, 199, 280, 284] [142, 143, 176] [14, 225, 236] [284]

Table 7.3: Defect reactions considered in each of our inputs for α-Fe.

Case	Interactions	References
Case 1	Interstitial clustering $I_n + I_m \rightarrow I_{n+m}$ DL immobilisation $I_n\langle 111\rangle + I_m\langle 111\rangle \rightarrow I_{n+m}\langle 100\rangle$	[105, 199, 280, 284] [284]
Case 2	Interstitial clustering $I_n + I_m \rightarrow I_{n+m}$ DL capture by single vacancies $I_n + V \rightarrow V \cdot I_n$ DL immobilisation $I_n\langle 111\rangle + I_m\langle 111\rangle \rightarrow I_{n+m}\langle 100\rangle$	[105, 199, 280, 284] [14, 225, 236] [284]

Table 7.4: Defect reactions considered in each of our inputs for Fe-5%Cr.

in this section, we shall show a summary of some of these benchmark results. All figures shown in this section have been taken from Ref. [25].

To demonstrate that all three codes deliver equivalent results when their input parameters and simulation conditions are equal, several cases involving the evolution of DLs in α-Fe were considered. In all cases, the chosen simulation box was a cubic $(1000a_0)^3$ box with periodic boundary conditions in all dimensions. Temperature was fixed to $T = 300\ K$ and the final simulation time was set to $t = 1\ s$. As in the previous section, two types of DLs were considered, namely mobile $\langle 111 \rangle$ and immobile $\langle 100 \rangle$ loops. $\langle 111 \rangle$ loops were permitted to move in one dimension along a direction of the $\langle 111 \rangle$ family. For a DL with n SIA, the migration attempt frequency was calculated as $\nu_0 = 2.4 \times 10^{10} n^{-0.64}\ s^{-1}$ and the DL radius as $r = a_0 \sqrt{\frac{n}{\sqrt{2}\pi}}$. The capture radius between two DLs of radii r_1 and r_2 was set to $r_c = r_1 + r_2 + 2a_0$ [25].

Case 1

In the first case, the simulation box was initially filled with a concentration of $10^{17}\ cm^{-3}$ $I_{50}\langle 111 \rangle$ and randomly chosen positions. Each interaction between two DLs was defined as the formation of an immobile DL $\langle 100 \rangle$ as follows:

$$\begin{aligned} I_n\langle 111 \rangle + I_m\langle 111 \rangle &\rightarrow I_{n+m}\langle 100 \rangle \\ I_n\langle 111 \rangle + I_m\langle 100 \rangle &\rightarrow I_{n+m}\langle 100 \rangle \end{aligned} \tag{7.27}$$

The time evolution of DLs in benchmark case 1 is depicted in Fig. 7.25 [25]. In all three codes, the density of $\langle 111 \rangle$ loops rapidly decreases as $\langle 100 \rangle$ loops are formed. As expected, the final density of DL $\langle 100 \rangle$ is approximately equal to one half of the initial density of DL $\langle 111 \rangle$, as two DLs $\langle 111 \rangle$ are needed to produce one DL $\langle 100 \rangle$. The remaining nonzero density of mobile DLs was shown to be an effect of the cubic geometry of the simulation box [25]. The fact that the three codes predict the same evolution of defects was evaluated positively in the M4F project [176].

Case 2

Once again, the initial concentration of mobile DLs in the simulation box was set to $10^{17}\ cm^{-3}$, but in the second benchmark case, their size was defined as I_{10}, i.e., the initial population of defects consisted of $I_{10}\langle 111 \rangle$ loops. In this case, the interaction

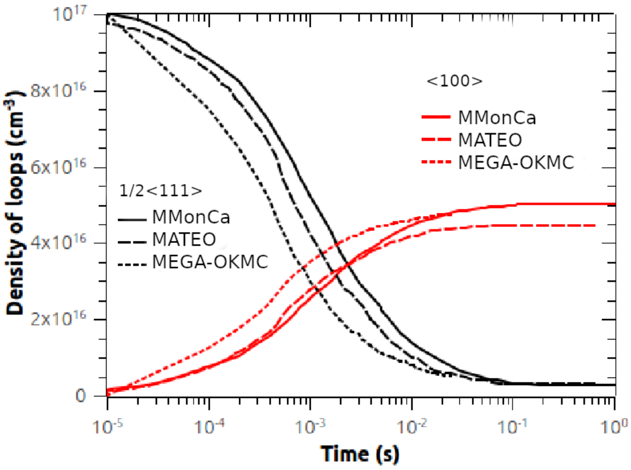


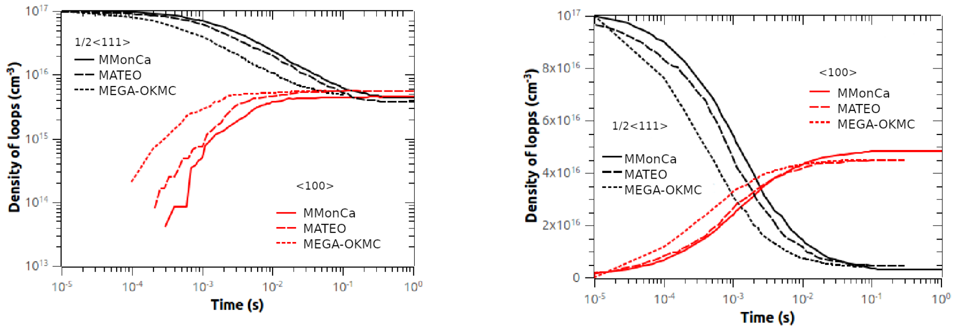
Figure 7.25: Time evolution of DLs in benchmark case 1 [25].

rules between DLs were inspired by Ref. [284]. Thus, interactions between DLs of similar size were determined as described in Eq. 7.20, whereas in interactions between DLs of unequal sizes, the resulting product took the Burgers vector of the bigger reactant.

The time evolution of DLs in benchmark case 2 is shown on the left in Fig. 7.26 [25]. The results predicted by all codes are very similar, showing that the creation of DLs $\langle 100 \rangle$ only begins circa $t = 10^{-4}\ s$. Final densities of $\langle 111 \rangle$ and $\langle 100 \rangle$ loops are roughly equal. Balbuena et al. [25] explained this phenomenon by the fact that the size criterion established for the formation of a DL $\langle 100 \rangle$ in Ref. [284] is not fulfilled at the very start of the simulation. This was confirmed with a complementary benchmark case, in which the initial DL population was replaced with $I_{30}\langle 111 \rangle$ loops, shown on the right in Fig. 7.26.

Case 3

In the next benchmark case, both types of DLs – namely, mobile and immobile DLs – were introduced in the local population. The initial density of DLs was set to $10^{17}\ cm^{-3}$, of which half were $I_{10}\langle 111 \rangle$ and half were $I_{20}\langle 100 \rangle$. The same interaction rules were established as in the previous case, i.e., two DLs of similar sizes would coalesce into a bigger DL following Eq. 7.20 as studied by Terentyev and Martín-



(a) Initial population of $I_{10}\langle 111\rangle$ loops. (b) Initial population of $I_{30}\langle 111\rangle$ loops.

Figure 7.26: Time evolution of DLs in benchmark case 2 [25].

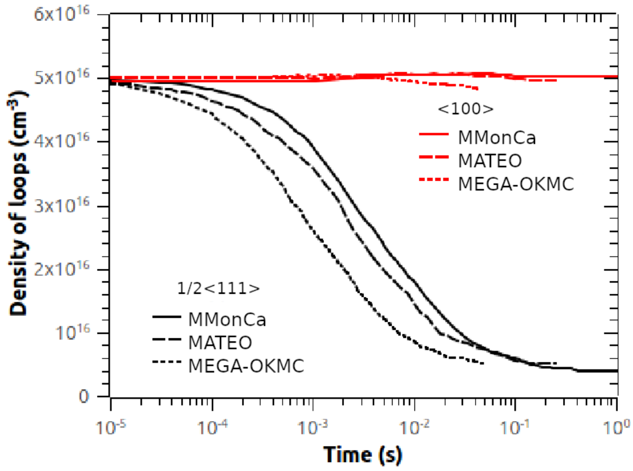


Figure 7.27: Time evolution of DLs in benchmark case 3 [25].

Bragado [284], and when the sizes of the reactants were not similar, the reaction product would inherit its Burgers vector from the bigger reactant.

Again, the results produced by all three codes are in agreement, shown in Fig. 7.27 [25]. Whereas the population of $\langle 111\rangle$ loops decreases to a residual level, the density of $\langle 100\rangle$ loops is approximately constant. Balbuena et al. [25] found that most $\langle 111\rangle$ loops are simply absorbed by bigger immobile $\langle 100\rangle$ loops, as their size is too small to fit the size requirement established in Ref. [284] for the formation of a new immobile $\langle 100\rangle$ loop. A small decrease of DL $\langle 100\rangle$ density was reported as a consequence of the absorption of small $\langle 100\rangle$ loops by large $\langle 111\rangle$ [25].

In this chapter, we have used MEGA-OKMC as a tool to understand physical experiments. The benchmarking of all three codes involved in this Work Package of the M4F project – MEGA-OKMC, MATEO and MMonCa – was determined successful [176].

We shall now finish the current dissertation with some conclusions.

Chapter 8

Summary and conclusions

An OKMC model to study defect evolution in irradiated materials of interest in nuclear fusion was developed from the ground up, named MEGA-OKMC. Due to the fact that the OKMC method maps well to a parallel model, a GPGPU paradigm was considered for the implementation of our work. The programming model used in our GPGPU implementation was CUDA.

The main problem behind *classic* OKMC models is their high computational cost when the number of defects is large. Indeed, due to the fact that the time step is approximately inversely proportional to this number, many computational steps are necessary to bring the simulation to a final simulation time, which increases the necessary runtime. For this reason, most OKMC models are restricted to small simulation boxes, in the order of a few hundred lattice parameters, which does not allow for enough statistics when the density of defects is low. In order to create an efficient GPU-OKMC model that solves this problem, the parallel Tau-leaping algorithm was developed in this work. In the PTLA, the time step is independent of the number of defects in the system, and consequently, the number of computational steps necessary to finish the simulation is smaller –in the order of the number of defects– than in an equivalent *classic* OKMC simulation.

To test our model for accuracy and performance, different test cases were considered. In order to demonstrate that this work is suited for particle systems of realistic sizes, millions of particles were simulated in realistic simulation boxes, in the order of typical grain size (20 μm). Our test cases showed excellent agreement with theory and equivalent *classic* OKMC simulations, and the performance of MEGA-OKMC over the serial OKMC –up to a 375× reduction in runtime– shows that our model is appropriate and efficient for the simulation of defect evolution in realistic boxes. For instance, we succeeded to reproduce resistivity recovery simulations in relatively short runtimes (less than one hour) where serial OKMC would require days of calculations to achieve the same result.

However, it was shown in this work that, under certain conditions, application of the PTLA is not enough to achieve simulations in a realistic runtime, such that approximations are necessary. In the case of dilute systems, where particles must perform many jumps before they can interact with other defects, the OKMC method becomes inefficient as most of the runtime is invested into the propagation of random walkers, the PTLA being no exception. Van Zon, ten Wolde et al. [292, 293] developed a solution to this problem, namely the GFRD method. In the GFRD method, the space containing N particles is partitioned into subspaces (protective domains) that contain one or two particles, and then, these 1-body and 2-body problems, identified as *Singles* and *Pairs* respectively, are solved independently of each other. This method was implemented in MEGA-OKMC as a complement to the PTLA, in situations where the density of defects was low. It was proven that using the GFRD method with a synchronous scheme in MEGA-OKMC, coupled with a sensible choice of the time step, provided accurate results efficiently.

One aspect that needed to be addressed was the behaviour of the GFRD method under continuous introduction of particles, which is the case in the simulation of a material under continuous irradiation. Since it is possible that several protective domains are entered by some of the newly introduced particles, the chance that some domains contain three or more particles at the end of the time step may not be ignored. These domains are not solvable analytically, so a new solution is required. In MEGA-OKMC, we developed a scheme that circumvents this problem. After creating the protective environments in the simulation box, cascades are introduced according to the rate of external irradiation, following a Poisson law. New protective environments are created around the new particles, and the simulation box is scanned in order to find where the new and old environments intersect. Cells where an intersection occurs are labelled *unsolvable cells* whereas all other cells are labelled *solvable cells*. First, the solvable cells are evolved with the selected time step. Then, the protective environments around particles in unsolvable cells are reconstructed, such that only 1-body and 2-body problems are defined. These are evolved in sub-time steps until the originally selected time step has been reached. Our solution was exemplified with the simulation of an electron irradiation experiment done by Arakawa et al. [16], and shown to be efficient and precise.

Further approximations were added to MEGA-OKMC. Some situations may arise in which many calculations are needed, resulting in a long runtime, which justifies the need for approximations. This is the case, for instance, when a few particles with very large sums of event rates are present in the system. Then, the associated time step is very small. If the contribution of these few particles to the evolution of the system is negligible, the number of computational steps needed to notice any change of state in the system becomes too large, which is detrimental to runtime. Hence, we introduced an approximation in which these few particles are not considered for the calculation of the time step, described in relation to the PTLA and the GFRD method. This approximation proved to be accurate and helped finish the simulations, which would have not been possible otherwise, or they would have required a prohibitive runtime. Specifically, with this approximation we were able to achieve good speedup factors of 9.2×, while keeping satisfactory precision. We deduce that this approximation provides an important increase in performance without a significant decrease in accuracy.

In addition, we noticed that there are limitations to the space division algorithm used in MEGA-OKMC to calculate interactions between particles and perform the computations needed for the GFRD method. Indeed, the growth of defect clusters may result in the appearance of some particles with a large radius, and the need of larger cells in order to do the desired calculations. As spatial cells may only contain a certain number of particles due to the upper threshold to the available shared memory per thread block on the GPU, regions of space with a large density of particles may translate into a hardware problem. We were able to overcome this limitation by developing a space division algorithm. Cells assigned to thread blocks whose shared memory limit is surpassed are subdivided into 8 new cells, giving rise to a new subdivision level. This operation is applied recursively if needed – up to 7 times in the current MEGA-OKMC version – and each subdivision level is processed separately.

The algorithm showing the integration of the PTLA and the GFRD method into MEGA-OKMC was explained. MEGA-OKMC is able to decide whether the GFRD method is applicable in any given time step by analysing the state of particles in the simulation box. Different parameters that influence the decision to produce a PTLA or GFRD step include the presence of massive particles, the prevalence of fast particle migration over other events, the mobility of second-fastest particles, the population of fast particles with respect to other populations, or the existence of continuous irradiation. The adaptive nature of MEGA-OKMC was demonstrated.

Last, our MEGA-OKMC model was used to study DL evolution in Fe and Fe-5%Cr. For this purpose, an experiment conducted by Yao et al. [315] in which an ultra-high purity Fe sample and a Fe-5%Cr sample were irradiated at 300 K with 150 keV self-ions was chosen. Under these conditions, a number of hypotheses were tested, and our simulation results show that the formation of DLs may be explained by trapping by impurities. Since we overestimated the presence of DLs compared to the experimental TEM observations, we explored several ideas that may clarify our results. On the one hand, the availability of many vacancies and vacancy clusters was shown as an obstacle to the formation of DLs. On the other hand, the artificial reduction of DL mobility resulted in unphysical behaviour. Future work with MEGA-OKMC may be concerned with the study of the capture radii used for SIA and vacancy clustering. It must be noted that, due to the efficiency of the algorithms developed in this work and the implementation using GPU programming, it was possible to study the evolution of thousands of DLs. This contrasts with previous studies conducted with sequential OKMCs, where only few DLs can be tackled due to the small size of the simulation boxes.

In conclusion, we have developed a GPU-OKMC model able to simulate the evolution of defects in pieces of material with sizes close to those achieved experimentally in realistic runtimes. As a consequence, our model can provide enough data for statistical treatment of particles that are usually present in low density, such as DLs, in principle related to the hardening of steels under irradiation. With this model, experimental time scales can be reached, which would be very difficult with *classic* OKMC models as a consequence of the long runtimes required.

MEGA-OKMC has been used and tested as a benchmark in the framework of the M4F project [176] by comparison with codes MMonCa [183] and MATEO [62], and found to be usable and accurate. In addition, MEGA-OKMC has been recently applied to neutron irradiation, delivering promising results for a future publication.

Future work includes the application of MEGA-OKMC to defects in alloys, such as FeCr, which are of high interest for structural materials in nuclear fusion [121, 145, 281, 315]. For example, in the case of FeCr, our model needs to take into account the possible migration energies of SIAs that depend on the local Cr arrangement. This remains a challenge for our model that shall be investigated in the future.

In addition, long-range effects between particles, such as elasticity, are important for the evolution of many defects like DLs in the surrounding of dislocation lines [186, 316]. MEGA-OKMC expects to represent materials in the presence of ITER-like or DEMO-like neutron irradiation environments, where the production of many of these defects is expected. Hence, our work must take long-range effects into account, but our work on this matter is only tentative so far. A possibility to account for long-range effects in an OKMC model is to calculate the field potential created by each particle in space, and then calculate the potential difference sustained by each particle when a jump is performed. The effects of elasticity and other field potentials on defect evolution must be implemented and validated as part of our future work.

Finally, the introduction of one-dimensional defects in this GPU-OKMC model, such as dislocations, or two-dimensional ones, like grain boundaries, will be a next step in our modelling in order to simulate the evolution of defects in realistic materials.

Appendix A

Appendix

A.1 Inter-particle reaction algorithm

As mentioned in subsection 4.2, the reactions between defects are the driving force behind many processes in materials physics. One of the most important features of any OKMC model is the treatment of interactions between particles. Indeed, in order to investigate diffusion-reaction systems relevant for nuclear fusion like helium and vacancy clustering in W [32] and the nucleation of point defects in α-Fe [131, 180, 289], it is necessary to search for inter-particle reactions at every step. Clearly, a brute force search is inadvisable for the million-particle systems that this work is expected to represent. Moreover, since local phenomena like particle interactions are confined to a small region in space, a neighbour search is appropriate. By dividing space into subspaces and treating these in parallel, the interactions between particles can be studied without the computational effort that a brute force search across the whole simulation box would entail.

The inter-particle reaction algorithm chosen in this GPU-OKMC model consists of two parts, namely the neighbour search algorithm and the function that performs interactions between particles. These are explained in section A.2 and section A.3 respectively.

A.2 Neighbour search algorithm

The neighbour search algorithm used for interactions in this code is based on the space division method described by S. Le Grand in chapter 32 of the book *GPU Gems 3* [126]. However, there are a few differences. For instance, we do not strictly regard particles as hard spheres, but as point objects surrounded by a sphere whose radius is related to the capture radius between two objects. In order to clarify this, a description of the neighbour search algorithm follows.

In order to exploit the power of device parallelism, our intention is to divide the simulation box into subdomains or cells such that particle interactions may be performed in each of these cells in parallel. With this in mind, space is divided into a three-dimensional uniform grid of n cells with the shape of a rectangular prism whose sides are parallel to the three axes. The cells are characterised by integer indices i, j and k according to the position of their left, bottom, front vertex in the X, Y and Z axes respectively. Thus, a particle p_α whose position is given by $\vec{x_\alpha} = (x, y, z)$ is located in cell α with indices (i, j, k). These indices are easily computed with Eq. A.1:

$$i = \left\lfloor \frac{x}{d_x} \right\rfloor \quad j = \left\lfloor \frac{y}{d_y} \right\rfloor \quad k = \left\lfloor \frac{z}{d_z} \right\rfloor \tag{A.1}$$

Here, d_x, d_y and d_z are the cell dimensions along each axis and the operator $\lfloor x \rfloor$ represents the floor function (i.e. rounding down of a real number). The cell α (i, j, k) is called the *home cell* of particle p_α.

It may be the case that two possibly interacting particles are located close to each other in space but along a boundary between their home cells, as shown in Figure A.1. If all cells were processed in parallel –in this figure, cells α and β– without regard for this situation, the algorithm would be fundamentally flawed.

For this reason, we need to account for *phantom cells*, a concept which is described by S. Le Grand in the cited book [126] and shall be explained next.

First, a spherical environment with radius R_c is drawn around each particle. The cells that are intersected by this sphere and that are not identical to the home cell of the particle are labelled phantom cells. Special care is taken if the sphere around particle p_α crosses a periodic boundary, since the cell on the opposite side that would be equivalent to the intersected cell beyond the boundary has to be counted as a

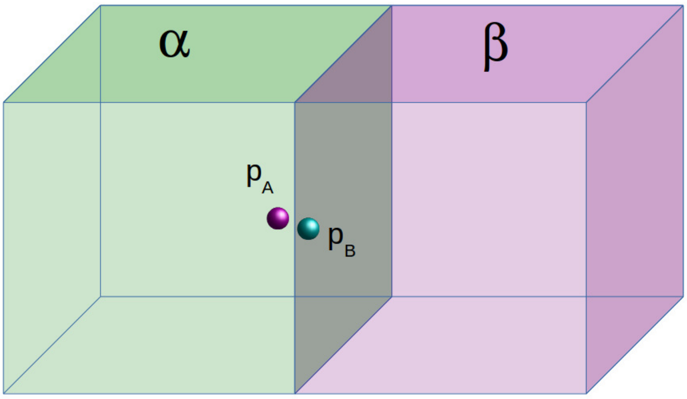


Figure A.1: Close particles in two different home cells.

phantom cell of p_α. To clarify this, a visual example involving home and phantom cells is depicted in Figure A.2. The point-like particle p_α is located in its home cell α. A sphere with radius R_c also intersects cell β, which is counted as a phantom cell. Considering periodic boundary conditions, cell γ is also intersected by the sphere. For easier visualisation, the simulation box is divided into only 8 cells, although our model usually creates thousands of cells to take full advantage of GPU parallelism.

The phantom cells are computed with help of the box-sphere collision test algorithm described by Larsson et al. in their 2007 article [160]. For each particle p_α with position $\vec{x_\alpha}$, collision tests are performed between the sphere with centre in $\vec{x_\alpha}$ and radius R_c and all 26 cells surrounding the home cell (i, j, k) of p_α, i.e. those cells with indices (i', j', k') which satisfy Eq. A.2. Interactions in phantom cells as well as those in home cells must be taken into consideration in order to avoid inaccuracies in the final result.

$$\begin{aligned} (|i - i'| \leq 1) \text{ and } (|j - j'| \leq 1) \text{ and } (|k - k'| \leq 1) \\ \text{and } (\text{not } [(i = i') \text{ and } (j = j') \text{ and } (k = k')]) \end{aligned} \tag{A.2}$$

The question remains of how to determine the size of sphere radius R_c and cell dimensions (d_x, d_y, d_z) in a sensible manner. The value of R_c must be large enough

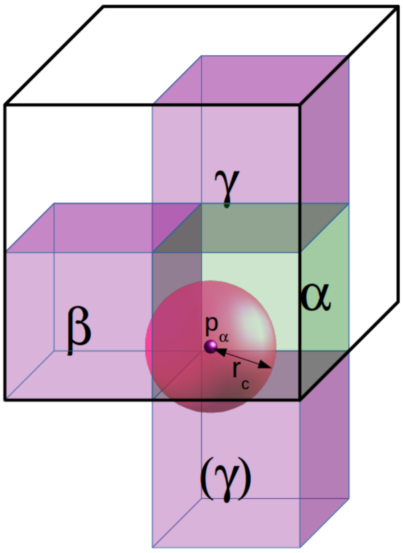


Figure A.2: Home and phantom cells of particle p_α. The simulation box borders are depicted in black.

to ensure that no particle interactions shall be overlooked. In our GPU-OKMC model, R_c must be at least as large as the largest possible capture radius defined in the possible reactions that can take place in the system.

Considering this, it is clear that the cell dimensions cannot be arbitrarily small, since their minimal size depends on R_c. Furthermore, although the interaction algorithm must take phantom cells into account, as we have shown, this may pose a problem if done naïvely. Let us, for instance, take a look at the situation depicted in Figure A.3. Assume that a particle p_A, whose home cell is cell α, is located close to the border between cells α and β, such that cell β is one of its phantom cells. If particle p_B is found in cell α and particle p_C is found in cell β, both located within the capture radii between them and p_A, it is certainly possible for p_A to perform an interaction with both p_B and p_C. Hence, if both cells α and β are processed in parallel, both interactions take place simultaneously, which is unphysical and may lead to wrong simulation results.

As a solution to this problem, we need a way to ensure that *either* the (p_A, p_B) interaction *or* the (p_A, p_C) one shall be performed. The idea suggested in [126] and applied to our GPU-OKMC model is to classify all grid cells by a type, such that

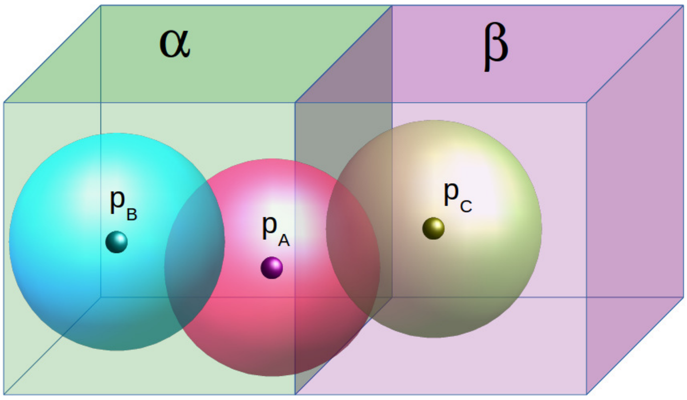


Figure A.3: Particles p_A and p_B are meant to interact in cell α (green), whereas an interaction between particles p_A and p_C is expected to take place in cell β (purple).

no two contiguous cells belong to the same type. Then, cells of the same type are processed in parallel, and this mechanism is repeated for all defined cell types. In the case of a three-dimensional grid, it is self-evident that 8 cell types are enough to fully characterise the cell system. Hence, the cells have to be created in groups of 8; strictly speaking, the total number of cells that the simulation box is divided into is a multiple of 8.

Figure A.4 illustrates this point. If all interactions in cells of type i are performed in parallel, and the particle system is updated before cells $j \neq i$ are processed, no conflicts arise from contradictions like those described in Figure A.3.

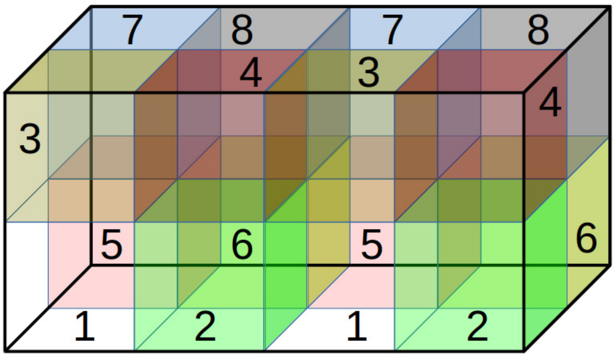


Figure A.4: 3D grid with 8 cell types, numbered 1-8.

At the same time, this means that the separation between cells of the same type should be large enough to avoid conflicts in parallel operations. Let p_A and p_B be two particles whose home cells, respectively α and $\alpha' \neq \alpha$, have the same cell type i. Let β be a phantom cell of both p_A and p_B. Then, assume that a third particle p_C is present in cell β, and placed in such a way that its bounding sphere overlaps both spheres around p_A and p_B. In other words, when the cells of type i are processed in parallel, particle p_C will be updated simultaneously in cell α and cell α'. This creates an undesirable race condition and a physically unreliable result.

This issue vanishes trivially if the cell dimensions are at least $4R_c$ in size. If both particles p_A and p_B are separated by a distance greater or equal to $4R_c$, the spheres with radius R_c around them cannot overlap simultaneously with a sphere with the same radius centred in p_C. Both situations are exemplified in Figure A.5. The top figure shows the contradictory situation where particles p_A and p_B are within interacting distance of p_C. Since the home cells α and α' are different but have the same type, they are processed in parallel, so the state of p_C is updated twice. Our solution is depicted in the bottom figure. p_A and p_B are separated by a distance of at least $4R_c$, i.e. the lower bound of the dimensions that shape the cell β between α and α'. By construction, the spheres around p_A and p_B cannot overlap the sphere around p_C at the same time. Therefore, the lower limit that defines the size of cell dimensions in each direction is equal to $4R_c$.

This construction has got an important consequence. Due to the existence of this lower bound, two particles may only interact in a cell if it is the home cell for at least one of them. If two particles appear in the same cell β which is a phantom cell for both, but neither of them are present in each other's home cell, it is safe to assume that they are too far apart to interact. The calculation can thus be neglected, saving runtime.

A.3 Particle interaction function

In order to perform the interactions, parallel operations are performed across all cells of the same type. Hence, eight function calls must be made per time step in order to sweep through the whole simulation box. Consequently, since two contiguous cells are of different types by construction, the operations performed in one cell are ensured to disturb no processes taking place in neighbouring cells.

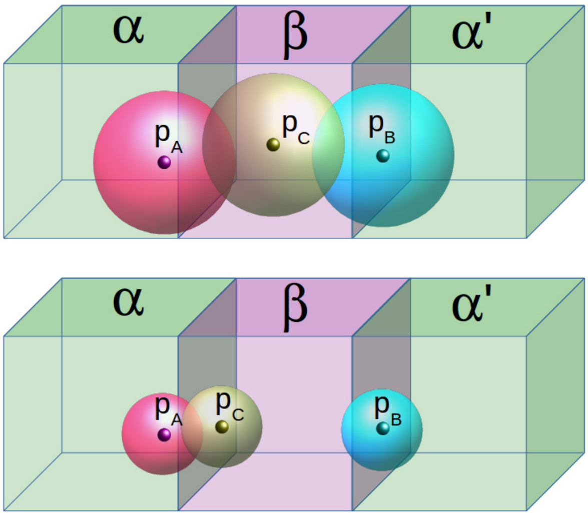


Figure A.5: Top: Contradictory situation that leads to the double update of particle p_C. Bottom: Our solution.

At any time step in the simulation, it may happen that several particle pairs in the same cell are in the right conditions to interact. Thus, a criterion is necessary to establish what pair to choose to interact. At this point, several criteria can be considered. On the one hand, it would seem logical and intuitive to discriminate in favour of the closest pair in the cell, i.e. to choose the pair with the shortest distance between both particles; however, from the physical point of view, there is no evidence that reactions actually take place in this way. In fact, this criterion fails to consider other factors such as the energy landscape, i.e. the shape of the interaction potential between the particles involved. On the other hand, the algorithm could simply choose one of the pairs randomly or according to a non-physical, arbitrary criterion: for example, choosing the first pair that appears on the list. This somehow takes into account the stochasticity of reactions and the fact that the energy landscape between particles cannot be known a priori. The fact that the inter-particle reaction function operates in parallel for cells of the same type, which is a completely arbitrary construction, contributes to this randomness, since there is no guarantee that pairs processed in cells of type α are closer than those processed in cells of type $\beta \neq \alpha$.

In this work, we present a combination of both criteria. Particles present in a cell are iterated through in order to check for possible interactions with other particles in the same cell. When the aforementioned occurrence presents itself, the closest neighbour of the chosen particle takes precedence over all others.

The following algorithm describes our particle interaction function:

1. One of the particles in the cell is chosen. The particle must satisfy all of the following conditions:

   - The particle must have moved in this time step. In other words, it must have migrated, been dissociated from a cluster or been introduced in a cascade.
   - The present cell must be the home cell of this particle.
   - The particle must not be flagged as having already interacted in this time step (see step 5).

   Otherwise, the particle is discarded.

2. The distances between the particles in the cell and the chosen particle are computed in parallel and stored. Reactions across periodic boundaries are also taken into consideration.

3. A parallel reduction takes place in order to find the closest neighbour of the chosen particle, determined by the minimum of the distances computed in step 2.

4. The function checks if an interaction is possible between the chosen particle and its closest neighbour. For this to occur, an interaction between both particles must have been defined in the input, and the distance between them must be smaller than the capture radius defined for them.

5. If there is an interaction taking place between the chosen particle and its neighbour, said interaction is performed. Both particles are then flagged as having interacted in this time step to avoid confusion.

6. If no interaction took place however, the particle's closest neighbour is ignored and steps 2 to 5 are repeated with the remaining neighbours.

7. For all particles in this cell, steps 1 to 6 are repeated.

Summing up, the inter-particle reaction algorithm used in our GPU-OKMC model has got the following form:

1. Choose the appropriate cell dimensions (d_x, d_y, d_z) automatically, such that the simulation box can be divided into groups of 8 cells. The sides of grid cells must be at least as large as $4R_c$.

2. Identify the home and phantom cells of each particle.

3. Call the particle interaction function to process all cells of type α in parallel.

4. Repeat step 3 for all seven remaining cell types.

Bibliography

[1] R. G. Abernethy. Predicting the performance of tungsten in a fusion environment: a literature review. *Mat. Sci. and Tech.*, 33:388–399, 2017.

[2] E. G. Adelberger et al. Solar fusion cross sections. II. the pp chain and CNO cycles. *Rev. Mod. Phys.*, 83:195–245, 2011.

[3] Advanced Micro Devices, Inc. AMD website. http://www.amd.com/, 2016.

[4] R. Agrawal and J. C. Shafer. Parallel mining of association rules. *IEEE TKDE*, 8:962–969, 1996.

[5] G. Aiello, J. Aubert, N. Jonquères, A. Li Puma, A. Morin, and G. Rampal. Development of the helium cooled lithium lead blanket for DEMO. *Fus. Eng. and Des.*, 89:1444–1450, 2014.

[6] C. Alejaldre et al. First plasmas in the TJ-II flexible Heliac. *Plasma Phys. Control. Fusion*, 41:A539, 1999.

[7] R. Alexander, M. C. Marinica, L. Proville, F. Willaime, K. Arakawa, M. R. Gilbert, and S. L. Dudarev. Ab initio scaling laws for the formation energy of nanosized interstitial defectclusters in iron, tungsten, and vanadium. *Phys. Rev. B*, 94:024103, 2016.

[8] M. J. Aliaga, A. Prokhodtseva, R. Schäublin, and M. J. Caturla. Molecular dynamics simulations of irradiation of α-Fe thin films with energetic Fe irons under channeling conditions. *J. Nucl. Mater.*, 452:453–456, 2014.

[9] M. J. Aliaga, R. Schäublin, J. F. Löffler, and M. J. Caturla. Surface-induced vacancy loops and damage dispersion in irradiated Fe thin films. *Acta Mater.*, 101:22–30, 2015.

[10] M. P. Allen. Introduction to Molecular Dynamics simulation. In N. Attig, K. Binder, H. Grubmüller, and K. Kremer, editors, *Computational Soft Matter: from Synthetic Polymers to Proteins: Lecture notes*, chapter 1, pages 1–28. John von Neumann Institute for Computing, Jülich, Germany, 2004.

[11] T. R. Allen, J. I. Cole, J. Gan, G. S. Was, R. Dropek, and E. A. Kenik. Swelling and radiation-induced segregation in austenitic alloys. *J. Nucl. Mater.*, 342:90–100, 2005.

[12] T. Amino, K. Arakawa, and H. Mori. Activation energy for long-range migration of self-interstitial atoms in tungsten obtained by direct measurement of radiation-induced point-defect clusters. *Phil. Mag. Lett.*, 91:86–96, 2011.

[13] T. Amino, K. Arakawa, and H. Mori. Detection of one-dimensional migration of single self-interstitial atoms in tungsten using high-voltage electron microscopy. *Sci. Rep.*, 6:26099, 2016.

[14] N. Anento and A. Serra. Carbon–vacancy complexes as traps for self-interstitial clusters in Fe–C alloys. *J. Nucl. Mater.*, 440:236–242, 2013.

[15] K. Arakawa, M. Hatanaka, E. Kuramoto, K. Ono, and H. Mori. Changes in the burgers vector of perfect dislocation loops without contact with the external dislocations. *Phys. Rev. Lett.*, 96:125506, 2006.

[16] K. Arakawa, H. Mori, and K. Ono. Formation process of dislocation loops in iron under irradiations with low-energy helium, hydrogen ions or high-energy electrons. *J. Nucl. Mater.*, 307:272–277, 2002.

[17] K. Arakawa, K. Ono, M. Isshiki, K. Mimura, M. Uchikoshi, and H. Mori. Observation of the one-dimensional diffusion of nanometer-sized dislocation loops. *Science*, 318:956–959, 2007.

[18] G. Arampatzis, M. A. Katsoulakis, P. Plecháč, M. Taufer, and L. Xu. Hierarchical fractional-step approximations and parallel kinetic Monte Carlo algorithms. *J. Comp. Phys.*, 231:7795–7814, 2012.

[19] J. Aubert, G. Aiello, J.-C. Jaboulay, B. Kiss, and A. Morin. Status on DEMO Helium Cooled Lithium Lead breeding blanket thermo-mechanical analyses. *Fus. Eng. and Des.*, 109-111:991–995, 2016.

[20] J. Aubert, G. Aiello, N. Jonquères, A. Li Puma, A. Morin, and G. Rampal. Development of the water cooled lithium lead blanket for DEMO. *Fus. Eng. and Des.*, 89:1386–1391, 2014.

[21] C. Bachmann, F. Arbeiter, L. V. Bocaccini, et al. Issues and strategies for DEMO in-vessel component integration. *Fus. Eng. and Des.*, 112:527–534, 2016.

[22] D. J. Bacon, Yu. N. Osetsky, and Z. Rong. Computer simulation of reactions between an edge dislocation and glissile self-interstitial clusters in iron. *Phil. Mag. A*, 86:3921–3936, 2006.

[23] D. J. Bacon, Yu. N. Osetsky, R. Stoller, and R. E. Voskoboinikov. MD description of damage production in displacement cascades in copper and α-iron. *J. Nucl. Mater.*, 323:152–162, 2003.

[24] X.-M. Bai, A. F. Voter, R. G. Hoagland, M. Nastasi, and B. P. Uberuaga. Efficient annealing of radiation damage near grain boundaries via interstitial emission. *Science*, 327:1631–1634, 2010.

[25] J. P. Balbuena, N. Castin, M. J. Caturla, C. Guerrero, L. Malerba, and C. Ortiz. Validation of microstructure evolution models: limitations and applicability. Technical Report Deliverable 3.2/D13, EURATOM, November 2020.

[26] J.P. Balbuena, L. Malerba, N. Castin, G. Bonny, and M.J. Caturla. An object kinetic Monte Carlo method to model precipitation and segregation in alloys under irradiation. *J. Nucl. Mater.*, 557:153236, 2021.

[27] N. Baluc, J. L. Boutard, S. L. Dudarev, M. Rieth, J. Brito Correia, B. Fournier, J. Henry, F. Legendre, T. Leguey, M. Lewandowska, R. Lindau, E. Marquis, A. Muñoz, B. Radiguet, and Z. Oksiuta. Review on the EFDA work programme on nano-structured ODS RAF steels. *J. Nucl. Mater.*, 417:149–153, 2011.

[28] V. Barabash, The ITER International Team, A. Peacock, S. Fabritsiev, G. Kalinin, S. Zinkle, A. Rowcliffe, J.-W. Rensman, A. A. Tavassoli, P. Marmy, P. J. Karditsas, F. Gillemot, and M. Akiba. Materials challenges for ITER – current status and future activities. *J. Nucl. Mater.*, 367-370:21–32, 2007.

[29] T. R. Barrett, G. Ellwood, et al. Progress in the engineering design and assessment of the European DEMO first wall and divertor plasma facing components. *Fus. Eng. and Des.*, 109-111:917–924, 2016.

[30] L. Barucca, E. Bubelis, et al. Pre-conceptual design of EU DEMO balance of plant systems: Objectives and challenges. *Fus. Eng. and Des.*, 169:112504, 2021.

[31] C. S. Becquart, A. Barbu, J. L. Bocquet, M. J. Caturla, C. Domain, C.-C. Fu, S. I. Golubov, M. Hou, L. Malerba, C. J. Ortiz, A. Souidi, and R. E. Stoller. Modeling the long-term evolution of the primary damage in ferritic alloys using coarse-grained methods. *J. Nucl. Mater.*, 406:39–54, 2010.

[32] C. S. Becquart and C. Domain. An object kinetic Monte Carlo simulation of the dynamics of helium and point defects in tungsten. *J. Nucl. Mater.*, 385:223–227, 2009.

[33] C. S. Becquart, C. Domain, U. Sarkar, A. DeBacker, and M. Hou. Microstructural evolution of irradiated tungsten: Ab initio parameterisation of an OKMC model. *J. Nucl. Mater.*, 403:75–88, 2010.

[34] D. Beeman. Some multistep methods for use in molecular dynamics calculations. *J. Comp. Phys.*, 20:130–139, 1976.

[35] C. D. Beidler, H. M. Smith, A. Alonso, et al. Demonstration of reduced neoclassical energy transport in Wendelstein 7-X. *Nature*, 596:221–226, 2021.

[36] L. K. Béland, P. Brommer, F. El-Mellouhi, J.-F. Joly, and N. Mousseau. Kinetic activation-relaxation technique. *Phys. Rev. E*, 84:046704, 2011.

[37] H. A. Bethe. Energy production in stars. *Phys. Rev.*, 55:103, 1939.

[38] G. Bhanot, D. Chen, A. Gara, and P. Vranas. The bluegene/l supercomputer. *Nucl. Phys. B (Proc. Suppl.)*, 119:114–121, 2003.

[39] C. Björkas, K. Nordlund, and M. J. Caturla. Influence of the picosecond defect distribution on damage accumulation in irradiated α-Fe. *Phys. Rev. B*, 85:024105, 2012.

[40] M. Roldán Blanco. *Caracterización de los efectos del He implantado en los materiales de fusión EUROFER97 y EU-ODS EUROFER mediante nanoindentación y microscopía electrónica de transmisión*. PhD thesis, Universidad Rey Juan Carlos, Madrid, Spain, 2015.

[41] R. A. Blythe and A. J. Bray. Survival probability of a diffusing particle in the presence of Poisson-distributed mobile traps. *Phys. Rev. E*, 67:041101, 2003.

[42] L. V. Bocaccini, G. Aiello, C. Bachmann, T. Barrett, A. del Nevo, D. Demange, L. Forest, F. Hernández, P. Norajitra, G. Porempović, D. Rapisarda, P. Sardain, M. Utili, and L. Vala. Objectives and status of EUROfusion DEMO blanket studies. *Fus. Eng. and Des.*, 109-111:1199–1206, 2016.

[43] T. A. Boden, G. Marland, and R. J. Andres. Global, regional, and national fossil-fuel CO_2 emissions. Technical report, Carbon Dioxide Information Analysis Center, Oak Ridge, USA, 2017.

[44] G. Bohm and G. Zech. *Introduction to statistics and data analysis for physicists*. Verlag Deutsches Elektronen-Synchrotron, Hamburg, Germany, 1st edition, 2010.

[45] G. Bonny, N. Castin, C. Domain, P. Olsson, B. Verreyken, M. I. Pascuet, and D. Terentyev. Density functional theory-based cluster expansion to simulate thermal annealing in FeCrW alloys. *Phil. Mag.*, 97:299–317, 2017.

[46] A. B. Bortz, M. H. Kalos, and J. L. Leibowitz. A new algorithm for Monte Carlo simulation of Ising spin systems. *J. Comp. Phys.*, 17:10–18, 1975.

[47] H.-S. Bosch and G. M. Hale. Improved formulas for fusion cross-sections and thermal reactivities. *Nucl. Fusion*, 32:611, 1992.

[48] R. Boullon, J.-C. Jaboulay, and J. Aubert. Molten salt breeding blanket: Investigations and proposals of pre-conceptual design options for testing in DEMO. *Fus. Eng. and Des.*, 171:112707, 2021.

[49] A. D. Brailsford, R. Bullough, and M. R. Hayns. Point defect sink strengths and void-swelling. *J. Nucl. Mater.*, 60:246–256, 1976.

[50] J. A. Brinkman. Production of atomic displacements by high-energy particles. *Amer. J. Phys.*, 24:246, 1956.

[51] C. H. M. Broeders and A. Y. Konobeyev. Defect production efficiency in metals under neutron irradiation. *J. Nucl. Mater.*, 328:197–214, 2004.

[52] J. Bruno and R. C. Ewing. Spent nuclear fuel. *Elements*, 2:343–349, 2006.

[53] J. Bucalossi et al. The WEST project: Testing ITER divertor high heat flux component technology in a steady state tokamak environment. *Fusion Eng. and Des.*, 89:907–912, 2014.

[54] V. P. Budaev, Yu. V. Martynenko, A. V. Karpov, N. E. Belova, A. M. Zhitlukhin, N. S. Klimov, V. L. Podkovyrov, V. A. Barsuk, A. B. Putrik, A. D. Yaroshevskaya, R. N. Giniyatulin, V. M. Safronov, and L. N. Khimchenko. Tungsten recrystallization and cracking under ITER-relevant heat loads. *J. Nucl. Mater.*, 463:237–240, 2015.

[55] L. Bukonte, F. Djurabekova, J. Samela, K. Nordlund, S. A. Norris, and M. J. Aziz. Comparison of molecular dynamics and binary collision approximation simulations for atom displacement analysis. *Nucl. Instrum. Methods Phys. Res. B*, 297:23–28, 2013.

[56] J. T. Busby, G. S. Was, and E. A. Kenik. Isolating the effect of radiation-induced segregation in irradiation-assisted stress corrosion cracking of austenitic stainless steels. *J. Nucl. Mater.*, 302:20–40, 2002.

[57] C. Cabet, F. Dalle, E. Gaganidze, J. Henry, and H. Tanigawa. Ferritic-martensitic steels for fission and fusion applications. *J. Nucl. Mat.*, 523:510–537, 2019.

[58] R. Candela, N. Mousseau, R. G. A. Veiga, C. Domain, and C. S. Becquart. Interaction between interstitial carbon atoms and a $1/2 \langle 111 \rangle$ self-interstitial atoms loop in an iron matrix: a combined DFT, off lattice KMC and MD study. *J. Phys.: Condens. Matter*, 30:335901, 2018.

[59] E. Carella and M. González. ^{3}He behaviour in Li_2TiO_3 ceramics for fusion breeding blanket applications. *Energy Procedia*, 41:26–33, 2013.

[60] D. Carloni, L. V. Boccaccini, F. Franza, and s. Kecskes. Requirements for helium cooled pebble bed blanket and R&D activities. *Fus. Eng. and Des.*, 89:1341–1345, 2014.

[61] N. Casal, F. Sordo, F. Mota, J. Jordanova, A. García, A. Ibarra, R. Vila, D. Rapisarda, V. Queral, and M. Perlado. IFMIF suitability for evaluation of fusion functional materials. *J. Nucl. Mater.*, 417:1316–1320, 2011.

[62] N. Castin, A. Bakaev, G. Bonny, et al. On the onset of void swelling in pure tungsten under neutron irradiation: an object kinetic Monte Carlo approach. *J. Nucl. Mater.*, 493:280–293, 2017.

[63] N. Castin, J. R. Fernández, and R. C. Pasianot. Predicting vacancy migration energies in lattice-free environments using artificial neural networks. *Comp. Mat. Sci.*, 84:217–225, 2014.

[64] N. Castin, M. I. Pascuet, and L. Malerba. Mobility and stability of large vacancy and vacancy-copper clusters in iron: An atomistic kinetic Monte Carlo study. *J. Nucl. Mater.*, 429:315–324, 2012.

[65] M. J. Caturla, N. Soneda, E. Alonso, B. D. Wirth, T. Díaz de la Rubia, and J. M. Perlado. Comparative study of radiation damage accumulation in Cu and Fe. *J. Nucl. Mater.*, 276:13–21, 2000.

[66] CEA. Laser Mégajoule website. http://www-lmj.cea.fr/index.htm, 2017.

[67] S. Chandrasekhar. Stochastic problems in physics and astronomy. *Rev. Mod. Phys.*, 43:1–89, 1943.

[68] A. Chartier and M.-C. Marinica. Rearrangement of interstitial defects in alpha-Fe under extreme condition. *Acta Mater.*, 180:141–148, 2019.

[69] A. Chatterjee and D. G. Vlachos. Temporal acceleration of spatially distributed kinetic Monte Carlo simulations. *J. Comp. Phys.*, 211:596–615, 2006.

[70] A. Chatterjee and D. G. Vlachos. An overview of spatial microscopic and accelerated kinetic Monte Carlo methods. *J. Computer-Aided Mater. Des.*, 14:253–308, 2007.

[71] J. Chen, N. Gao, P. Jung, and T. Sauvage. A new mechanism of loop formation and transformation in bcc iron without dislocation reaction. *J. Nucl. Mater.*, 441:216–221, 2013.

[72] M. Chertkov and V. Lebedev. Boundary effects on chaotic advection-diffusion chemical reactions. *Phys. Rev. Lett.*, 90:134501, 2003.

[73] M. Chiapetto, L. Malerba, and C. S. Becquart. Effect of Cr content on the nanostructural evolution of irradiated ferritic/martensitic alloys: An object kinetic Monte Carlo model. *J. Nucl. Mater.*, 465:326–336, 2015.

[74] M. Chiapetto, L. Malerba, and C. S. Becquart. Nanostructure evolution under irradiation in FeMnNi alloys: A “grey alloy” object kinetic monte carlo model. *J. Nucl. Mater.*, 462:91–99, 2015.

[75] J. W. Coenen et al. Materials for DEMO and reactor applications - boundary conditions and new concepts. *Phys. Scr.*, T167:014002, 2016.

[76] NVIDIA Corporation. NVIDIA website. http://www.nvidia.com/, 2017.

[77] A. E. Costley. On the fusion triple product and fusion power gain of tokamak pilot plants and reactors. *Nucl. Fusion*, 56:066003, 2016.

[78] L. Dagum and R. Menon. OpenMP: An industry-standard API for shared-memory programming. *IEEE Comp. Sci. & Eng.*, 5:46–55, 1998.

[79] Y. Dai, J. Henry, Z. Tong, X. Averty, J. Malaplate, and B. Long. Neutron/proton irradiation and He effects on the microstructure and mechanical properties of ferritic/martensitic steels T91 and EM10. *J. Nucl. Mater.*, 415:306–310, 2011.

[80] M. Dalla Palma. Modelling of cyclic plasticity for austenitic stainless steels 304L, 316L, 316L(N)-IG. *Fus. Eng. and Des.*, 109-111:20–25, 2016.

[81] J. Dalla Torre, J.-L. Bocquet, N. V. Doan, E. Adam, and A. Barbu. Jerk, an event-based Kinetic Monte Carlo model to predict microstructure evolution of materials under irradiation. *Phil. Mag.*, 85:549–558, 2005.

[82] J. Dalla Torre, C.-C. Fu, F. Willaime, A. Barbu, and J.-L. Bocquet. Resistivity recovery simulations of electron-irradiated iron: Kinetic Monte Carlo versus cluster dynamics. *J. Nucl. Mater.*, 352:42–49, 2006.

[83] T. Díaz de la Rubia, R. S. Averback, R. Benedek, and W. E. King. Role of thermal spikes in energetic displacement cascades. *Phys. Rev. Lett.*, 59:1930–1933, 1987.

[84] M. Decreton, T. Shikama, and E. R. Hodgson. Performance of functional materials and components in a fusion reactor: the issue of radiation effects in ceramics and glass materials for diagnostics. *J. Nucl. Mater.*, 329-333, Part A:125–132, 2004.

[85] P. M. Derlet and S. L. Dudarev. Million-atom molecular dynamics simulations of magnetic iron. *Prog. Mat. Sci.*, 52:299–318, 2007.

[86] C. Dethloff, E. Gaganidze, and J. Aktaa. Review and critical assessment of dislocation loop analyses on EUROFER 97. *Nucl. Mat. and Energy*, 15:23–26, 2018.

[87] T. J. Dolan. Magnetic electrostatic plasma confinement. *Plasma Phys. Control. Fusion*, 36:1539–1593, 1994.

[88] C. Domain, C. S. Becquart, and L. Malerba. Simulation of radiation damage in Fe alloys: an object kinetic Monte Carlo approach. *J. Nucl. Mater.*, 335:121–145, 2004.

[89] T. Donné and W. Morris and others. European research roadmap to the realisation of fusion energy. Technical report, EUROfusion, Garching, Germany, 2018.

[90] V. I. Dubinko. Temporal and spatial evolution of dislocation and void structures under cascade damage production. *Nucl. Instrum. Methods Phys. Res. B*, 153:116–121, 1999.

[91] A. Dunn, R. Dingreville, and L. Capolungo. Multi-scale simulation of radiation damage accumulation and subsequent hardening in neutron-irradiated α-fe. *Modelling Simul. Mater. Sci. Eng.*, 24:015005, 2016.

[92] A. Y. Dunn, L. Capolungo, E. Martínez, and M. Cherkaoui. Spatially resolved stochastic cluster dynamics for radiation damage evolution in nanostructured metals. *J. Nucl. Mater.*, 443:128–139, 2013.

[93] O. Edenhofer et al., editors. *Climate Change 2014: Mitigation of Climate Change. Contribution of Working Group III to the Fifth Assessment Report of the Intergovernmental Panel on Climate Change.* Cambridge University Press, Cambridge, UK and New York, USA, 2014.

[94] A. Einstein. Zur Elektrodynamik bewegter Körper. *Ann. Phys.*, 17:891–921, 1905.

[95] O. El-Atwani, J.E. Nathaniel II, A.C. Leff, B.R. Muntifering, J.K. Baldwin, K. Hattar, and M.L. Taheri. The role of grain size in He bubble formation: Implications for swelling resistance. *J. Nucl. Mater.*, 484:236–244, 2017.

[96] F. El-Mellouhi, N. Mousseau, and L. J. Lewis. Kinetic activation-relaxation technique: An off-lattice self-learning kinetic Monte Carlo algorithm. *Phys. Rev. B*, 78:153202, 2008.

[97] M. Enoeda, H. Tanigawa, et al. Development of the Water Cooled Ceramic Breeder Test Blanket Module in Japan. *Fus. Eng. and Des.*, 87:1363–1369, 2012.

[98] Y. Fan, A. Kushima, S. Yip, and B. Yildiz. Mechanism of void nucleation and growth in bcc Fe: Atomistic simulations at experimental time scales. *Phys. Rev. Lett.*, 106:125501, 2011.

[99] G. Federici, C. Bachmann, et al. Overview of the design approach and prioritization of R&D activities towards an eu DEMO. *Fus. Eng. and Des.*, 109-111:1464–1474, 2016.

[100] G. Federici, L. Bocaccini, et al. An overview of the EU breeding blanket design strategy as an integral part of the DEMO design effort. *Fus. Eng. and Des.*, 141:30–42, 2019.

[101] G. Federici et al. DEMO design activity in europe: Progress and updates. *Fus. Eng. and Des.*, 136:729–741, 2018.

[102] K. A. Fichthorn and W. H. Weinberg. Theoretical foundations of dynamical Monte Carlo simulations. *J. Chem. Phys.*, 95:1090–1096, 1991.

[103] U. Fischer, C. Bachmann, J.-C. Jaboulay, F. Moro, I. Palermo, P. Pereslavtsev, and R. Villari. Neutronic performance issues of the breeding blanket options for the European DEMO fusion power plant. *Fus. Eng. and Des.*, 109-111:1458–1463, 2016.

[104] Fusion for Energy. F4E website. http://fusionforenergy.europa.eu/, 2017.

[105] C.-C. Fu, J. Dalla Torre, F. Willaime, J.-L. Bocquet, and A. Barbu. Multiscale modelling of defect kinetics in irradiated iron. *Nature Materials*, 4:68–74, 2005.

[106] C.-C. Fu and F. Willaime. Ab initio study of helium in α-Fe: Dissolution, migration, and clustering with vacancies. *Phys. Rev. B*, 72:064117, 2005.

[107] C.-C. Fu and F. Willaime. Interaction between helium and self-defects in α-iron from first principles. *J. Nucl. Mater.*, 367-370:244–250, 2007.

[108] X. Gai, T. Lazauskas, R. Smith, and S. D. Kenny. Helium bubbles in bcc fe and their interactions with irradiation. *J. Nucl. Mater.*, 462:382–390, 2015.

[109] G. J. Galloway and G. J. Ackland. Molecular dynamics and object kinetic Monte Carlo study of radiation-induced motion of voids and He bubbles in bcc iron. *Phys. Rev. B*, 87:104106, 2013.

[110] L. Gámez, B. Gámez, M. J. Caturla, D. Terentyev, and J. M. Perlado. Object Kinetic Monte Carlo calculations of irradiated Fe–Cr dilute alloys: The effect of the interaction radius between substitutional Cr and self-interstitial Fe. *Nucl. Instrum. Methods Phys. Res. B*, 269:1684–1688, 2011.

[111] F. A. Garner, M. B. Toloczko, and B. H. Sencer. Comparison of swelling and irradiation creep behavior of fcc-austenitic and bcc-ferritic/martensitic alloys at high neutron exposure. *J. Nucl. Mater.*, 276:123–142, 2000.

[112] D. Garoz, A. R. Páramo, A. Rivera, J. M. Perlado, and R. González-Arrabal. Modelling the thermomechanical behaviour of the tungsten first wall in HiPER laser fusion scenarios. *Nucl. Fusion*, 56:126014, 2016.

[113] N. M. Ghoniem and S. P. Chou. Binary collision Monte Carlo simulations of cascades in polyatomic ceramics. *J. Nucl. Mater.*, 155-157:1263–1267, 1988.

[114] M. R. Gilbert, S. L. Dudarev, D. Nguyen-Manh, S. Zheng, L. W. Packer, and J.-Ch. Sublet. Neutron-induced dpa, transmutations, gas production, and helium embrittlement of fusion materials. *J. Nucl. Mater.*, 442:S755–S760, 2013.

[115] M. R. Gilbert, S. L. Dudarev, S. Zheng, L. W. Packer, and J.-Ch. Sublet. An integrated model for materials in a fusion power plant: transmutation, gas production, and helium embrittlement under neutron irradiation. *Nucl. Fusion*, 52:083019, 2012.

[116] M. R. Gilbert and J.-Ch. Sublet. Neutron-induced transmutation effects in W and W-alloys in a fusion environment. *Nucl. Fusion*, 51:043005, 2011.

[117] H. Gilhaus and W. Schüle. Dynamic crowdions, interstitials and vacancies in copper. *Rad. Eff.*, 138:29–38, 1996.

[118] D. T. Gillespie. A general method for numerically simulating the stochastic time evolution of coupled chemical reactions. *J. Nucl. Mater.*, 22:403–434, 1976.

[119] D. T. Gillespie. Approximate accelerated stochastic simulation of chemically reacting systems. *J. Chem. Phys.*, 115:1716, 2001.

[120] F. Goded Echeverría. *Teoría de Reactores y Elementos de Ingeniería Nuclear*. Sección de Publicaciones de la J. E. N., Madrid, Spain, 1958.

[121] B. Gómez-Ferrer, I. García-Cortés, J. F. Marco, D. Jiménez-Rey, and R. Vila. Decoupling of defect and short-range order contributions to resistivity recovery measurements in binary alloys. *Phys. Rev. B*, 90:220102, 2014.

[122] B. Gómez-Ferrer, R. Vila, D. Jiménez-Rey, C. J. Ortiz, F. Mota, J. M. García, and A. Rodríguez. In situ resistivity measurements of RAFM base alloys at cryogenic temperatures: The effect of proton irradiation. *J. Nucl. Mater.*, 447:225–232, 2014.

[123] C. Gong, J. Liu, L. Chi, H. Huang, J. Fang, and Z. Gong. GPU accelerated simulations of 3D deterministic particle transport using discrete ordinates method. *J. Comp. Phys.*, 230:6010–6022, 2011.

[124] C. Gong, J. Liu, H. Huang, and Z. Gong. Particle transport with unstructured grid on GPU. *Comp. Phys. Comm.*, 183:588–593, 2012.

[125] A. Grama, A. Gupta, G. Karypis, and V. Kumar. *Introduction to Parallel Computing*. Addison Wesley, Harlow, England, 2nd edition, January 2003.

[126] S. Le Grand. *Broad-phase collision detection with CUDA*, chapter 32. NVIDIA Corporation, 2008.

[127] W. Gropp, E. Lusk, N. Doss, and A. Skjellum. A high-performance, portable implementation of the MPI message passing interface standard. *Parallel Computing*, 22:789–828, 1996.

[128] A. M. Gusak, G. V. Lutsenko, and K. N. Tu. Ostwald ripening with non-equilibrium vacancies. *Acta Mater.*, 54:785–791, 2006.

[129] C. D. Hardie, C. A. Williams, S. Xu, and S. G. Roberts. Effects of irradiation temperature and dose rate on the mechanical properties of self-ion implanted Fe and Fe–Cr alloys. *J. Nucl. Mater.*, 439:33–40, 2013.

[130] J. Heikinheimo, K. Mizohata, et al. Direct observation of mono-vacancy and self-interstitial recovery in tungsten. *APL Mater.*, 7:021103, 2019.

[131] H. L. Heinisch, H. Trinkaus, and B. N. Singh. Kinetic Monte Carlo studies of the reaction kinetics of crystal defects that diffuse one-dimensionally with occasional transverse migration. *J. Nucl. Mater.*, 367-370:332–337, 2007.

[132] P. Helander, C. D. Beidler, T. M. Bird, M. Drevlak, Y. Feng, R. Hatzky, F. Jenko, R. Kleiber, J. H. E. Proll, Yu. Turkin, and P. Xanthopoulos. Stellarator and tokamak plasmas: a comparison. *Plasma Phys. Control. Fusion*, 54:124009, 2012.

[133] M. Hernández-Mayoral and D. Gómez-Berceño. Transmission electron microscopy study on neutron irradiated pure iron and RPV model alloys. *J. Nucl. Mater.*, 399:146–153, 2010.

[134] J. Hoberock and N. Bell. Thrust: A parallel template library, 2010. Version 1.8.1.

[135] E. R. Hodgson and T. Shikama. Radiation effects on the physical properties of dielectric insulators for fusion reactors. In R. J. M. Konings, editor, *Comprehensive Nuclear Materials*. Elsevier, Amsterdam, Netherlands, 2012.

[136] J. How and R. Reichle. PD - Plant Description. Technical report, ITER Organization, 2009.

[137] O. A. Hurricane, D. A. Callahan, et al. Fuel gain exceeding unity in an inertially confined fusion implosion. *Nature*, 506:343–348, 2014.

[138] O. A. Hurricane et al. The high-foot implosion campaign on the National Ignition Facility. *Phys. Plasmas*, 21:056314, 2014.

[139] Yu. Igitkhanov, R. Fetzer, and B. Bazylev. Effect of design geometry of the demo first wall on the plasma heat load. *Nuc. Mater. and Energy*, 9:560–564, 2016.

[140] International Atomic Energy Agency. The fukushima daiichi accident. technical volume 1: Description and context of the accident. Technical report, International Atomic Energy Agency, Vienna, Austria, 2015.

[141] ITER. ITER website. https://www.iter.org/, 2017.

[142] V. Jansson, M. Chiapetto, and L. Malerba. The nanostructure evolution in Fe–C systems under irradiation at 560 K. *J. Nucl. Mater.*, 442:341–349, 2013.

[143] V. Jansson and L. Malerba. Simulation of the nanostructure evolution under irradiation in Fe–C alloys. *J. Nucl. Mater.*, 443:274–285, 2013.

[144] V. Jansson, L. Malerba, A. De Backer, C. S. Becquart, and C. Domain. Sink strength calculations of dislocations and loops using OKMC. *J. Nucl. Mater.*, 442:218–226, 2013.

[145] M. L. Jenkins, Z. Yao, M. Hernández-Mayoral, and M. A. Kirk. Dynamic observations of heavy-ion damage in Fe and Fe–Cr alloys. *J. Nucl. Mater.*, 389:197–202, 2009.

[146] F. Jiménez and C. J. Ortiz. A GPU-based parallel Object kinetic Monte Carlo algorithm for the evolution of defects in irradiated materials. *Comp. Mat. Sci.*, 113:178–186, 2016.

[147] R. A. Johnson. Point-defect calculations for tungsten. *Phys. Rev. B*, 27:2014–2018, 1983.

[148] R. A. Johnson and E. Brown. Point defects in copper. *Phys. Rev.*, 127:446–454, 1962.

[149] T. Jourdan, J.-L. Bocquet, and F. Soisson. Modeling homogeneous precipitation with an event-based Monte Carlo method: Application to the case of Fe–Cu. *Acta Mater.*, 58:3295–3302, 2010.

[150] T. Jourdan and J.-P. Crocombette. Rate theory cluster dynamics simulations including spatial correlations within displacement cascades. *Phys. Rev. B*, 86:054113, 2012.

[151] T. Jourdan, C.-C. Fu, L. Joly, J. L. Bouquet, M. J. Caturla, and F. Willaime. Direct simulation of resistivity recovery experiments in carbon-doped α-iron. *Phys. Scr.*, T145:014049, 2011.

[152] S. A. Kabakchi, A. V. Putilov, and Ye. R. Nazin. Data analysis and physicochemical modeling of the radiation accident in the southern urals in 1957. *Atomnaya Energiya*, 78:46–50, 1995.

[153] M. H. Kalos, D. Levesque, and L. Verlet. Helium at zero temperature with hard-sphere and other forces. *Phys. Rev. A*, 9:2178–2195, 1974.

[154] M. Keilhacker et al. High fusion performance from deuterium-tritium plasmas in JET. *Nucl. Fusion*, 39:209, 1999.

[155] H. Kim, J. G. Gigax, C. J. Rietema, et al. Void swelling of conventional and composition engineered HT9 alloys after high-dose self-ion irradiation. *J. Nucl. Mater.*, 560:153492, 2022.

[156] I. R. Kirillov, D. M. Obukhov, D. A. Pertsev, I. A. Kartashev, A. Yu. Leshukov, and M. N. Sviridenko. Results of lead-lithium ceramic breeder TBM conceptual design optimization. *Fus. Eng. and Des.*, 89:1421–1425, 2014.

[157] M. Klimenkov, U. Jäntsch, M. Rieth, et al. Post-irradiation microstructural examination of EUROFER-ODS steel irradiated at 300° C and 400° C. *J. Nucl. Mat.*, 557:153259, 2021.

[158] A. A. Kohnert, B. D. Wirth, and L. Capolungo. Modeling microstructural evolution in irradiated materials with cluster dynamics methods: A review. *Comp. Mat. Sci.*, 149:442–459, 2018.

[159] E. A. Kotomin, V. N. Kuzovkov, A. I. Popov, and R. Vila. Kinetics of F center annealing and colloid formation in al_2o_3. *Nucl. Instrum. Methods Phys. Res. B*, 374:107–110, 2016.

[160] T. Larsson, T. Akenine-Möller, and E. Lengyel. On faster sphere-box overlap testing. *J. Graphics Tools*, 12:3–6, 2007.

[161] R. Lässer, N. Baluc, J.-L. Boutard, E. Diegele, S. Dudarev, M. Gasparotto, A. Möslang, R. Pippan, B. Riccardi, and B. van der Schaaf. Structural materials for DEMO: The EU development, strategy, testing and modelling. *Fus. Eng. and Des.*, 82:511–520, 2007.

[162] T. S. Ledley, E. T. Sundquist, S. E. Schwartz, D. K. Hall, J. D. Fellows, and T. L. Killeen. Climate change and greenhouse gases. *Eos*, 80:453–454, 457–458, 1999.

[163] C. Lehmann. *Interaction of Radiation with Solids and Elementary Defect Production.* North-Holland, Amsterdam, Netherlands, 1977.

[164] K. Li, R. Khanna, Z. Jianliang, G. Li, et al. Determination of the accuracy and reliability of molecular dynamics simulations in estimating the melting point of iron: Roles of interaction potentials and initial system configurations. *J. Mol. Liq.*, 290:111204, 2019.

[165] Y. Liang, X. Xing, and Y. Li. A GPU-based large-scale Monte Carlo simulation method for systems with long-range interactions. *J. Comp. Phys.*, 338:252–268, 2017.

[166] J. Lindl. Development of the indirect-drive approach to inertial confinement fusion and the target physics basis for ignition and gain. *Phys. Plasmas*, 2:3933–4024, 1995.

[167] C. Linsmaier, M. Rieth, et al. Development of advanced high heat flux and plasma-facing materials. *Nucl. Fusion*, 57:092007, 2017.

[168] A. Litnovsky, J. Schmitz, et al. Smart tungsten-based alloys for a first wall of DEMO. *Fus. Eng. and Des.*, 159:111742, 2020.

[169] E. A. Little. Void-swelling in irons and ferritic steels: I. mechanisms of swelling suppression. *J. Nucl. Mater.*, 87:11–24, 1979.

[170] L. Liu, D. Liu abd Y. Hong, H. Fan, W. Ni, Q. Yang, Z. Bi, G. Benstetter, and S. Li. High-flux He^+ irradiation effects on surface damages of tungsten under ITER relevant conditions. *J. Nucl. Mater.*, 471:1–7, 2016.

[171] P.-W. Ma, S. L. Dudarev, and C. H. Woo. SPILADY: A parallel CPU and GPU code for spin-lattice magnetic molecular dynamics simulations. *Comp. Phys. Comm.*, 207:350–361, 2016.

[172] T. Ma, O. A. Hurricane, D. A. Callahan, et al. Thin shell, high velocity inertial confinement fusion implosions on the National Ignition Facility. *Phys. Rev. Lett.*, 114:145004, 2015.

[173] L. Malerba. Molecular dynamics simulation of displacement cascades in α-Fe: A critical review. *J. Nucl. Mater.*, 351:28–38, 2006.

[174] L. Malerba, G. J. Ackland, C. S. Becquart, G. Bonny, C. Domain, S. L. Dudarev, C.-C. Fu, D. Hepburn, M. C. Marinica, P. Olsson, R. C. Pasianot, J. M. Raulot, F. Soisson, D. Terentyev, E. Vincent, and F. Willaime. Ab initio calculations and interatomic potentials for iron and iron alloys: Achievements within the Perfect project. *J. Nucl. Mater.*, 406:7–18, 2010.

[175] L. Malerba, N. Anento, J. P. Balbuena, et al. Physical mechanisms and parameters for models of microstructure evolution under irradiation in Fe alloys - part I: Pure Fe. *Nucl. Mat. and Energy*, 29:101069, 2021.

[176] L. Malerba, N. Anento, et al. Multiscale modelling for fusion and fission materials: The M4F project. *Nucl. Mat. and Energy*, 29:101051, 2021.

[177] L. Malerba, C. S. Becquart, and C. Domain. Object kinetic Monte Carlo study of sink strengths. *J. Nucl. Mater.*, 360:159–169, 2007.

[178] L. Malerba, M. C. Marinica, N. Anento, C. Björkas, H. Nguyen, C. Domain, et al. Comparison of empirical interatomic potentials for iron applied to radiation damage studies. *J. Nucl. Mater.*, 406:19–38, 2010.

[179] L. K. Mansur. Theory and experimental background on dimensional changes in irradiated alloys. *J. Nucl. Mater.*, 216:97–121, 1994.

[180] J. Marian, B. D. Wirth, R. Schäublin, J. M. Perlado, and T. Díaz de la Rubia. $\langle 100 \rangle$-loop characterization in α-Fe: comparison between experiments and modeling. *J. Nucl. Mater.*, 307-311:871–875, 2002.

[181] P. Martín, D. Jiménez-Rey, R. Vila, F. Sánchez, and R. Saavedra. Optical absorption defects created in SiO_2 by Si, O and He ion irradiation. *Fus. Eng. and Des.*, 89:1679–1683, 2014.

[182] I. Martín-Bragado, J. Abujas, P. L. Galindo, and J. Pizarro. Synchronous parallel Kinetic Monte Carlo: Implementation and results for object and lattice approaches. *Nucl. Instrum. Methods Phys. Res. B*, 352:27–30, 2015.

[183] Ignacio Martín-Bragado, Antonio Rivera, Gonzalo Valles, Jose Luis Gomez-Selles, and M. J. Caturla. MMonCa: An Object Kinetic Monte Carlo simulator for damage irradiation evolution and defect diffusion. *Comp. Phys. Comm.*, 184:27032710, 2013.

[184] E. Martínez, J. Marian, M. H. Kalos, and M. Preciado. Synchronous parallel kinetic Monte Carlo for continuum diffusion-reaction systems. *J. Comp. Phys.*, 227:3804–3823, 2008.

[185] E. Martínez, O. Senninger, C.-C. Fu, and F. Soisson. Decomposition kinetics of fe-cr solid solutions during thermal aging. *Phys. Rev. B*, 86:224109, 2012.

[186] D. R. Mason, X. Yi, M. A. Kirk, and S. L. Dudarev. Elastic trapping of dislocation loops in cascades in ion-irradiated tungsten foils. *J. Phys.: Condens. Matter*, 26:375701, 2014.

[187] H. Matsui, S. Takehana, and M. W. Guinan. Resistivity recovery in high purity iron after fission- and fusion- neutron irradiation. *J. Nucl. Mater.*, 155-157:1284–1289, 1988.

[188] A. K. Mazur. Common Molecular Dynamics algorithms revisited: Accuracy and optimal time steps of Störmer-Leapfrog integrators. *J. Comp. Phys.*, 136:354–365, 1997.

[189] S. Mittal and J. S. Vetter. A survey of methods for analyzing and improving GPU energy efficiency. *ACM Comput. Surv.*, 47:19:1–22, 2014.

[190] R. Mitteau, B. Calcagno, P. Chappuis, R. Eaton, S. Gicquel, J. Chen, A. Labusov, A. Martin, M. Merola, R. Raffray, M. Ulrickson, and F. Zacchia. The design of the ITER first wall panels. *Fus. Eng. and Des.*, 88:568–570, 2013.

[191] W. Mohammed, B. Miller, D. Porter, and K. Murty. The role of grain size on neutron irradiation response of nanocrystalline copper. *Materials*, 9:144, 2016.

[192] S. Moll, T. Jourdan, and H. Lefaix-Jeuland. Direct observation of interstitial dislocation loop coarsening in α-iron. *Phys. Rev. Lett.*, 111:015503, 2013.

[193] K. Morishita and R. Sugano. Mechanism map for nucleation and growth of helium bubbles in metals. *J. Nucl. Mater.*, 353:52–65, 2006.

[194] K. Morishita, R. Sugano, B. D. Wirth, and T. Díaz de la Rubia. Thermal stability of helium-vacancy clusters in iron. *Nucl. Instrum. Methods Phys. Res. B*, 202:76–81, 2003.

[195] M. Moscheni, M. Carr, et al. Radiative heat load distribution on the EU-DEMO first wall due to mitigated disruptions. *Nucl. Mat. and Energy*, 25:100824, 2020.

[196] A. Möslang. IFMIF: the intense neutron source to qualify materials for fusion reactors. *C. R. Physique*, 9:457–468, 2008.

[197] F. Mouvet, J. Villard, V. Bolnykh, and U. Rothlisberger. Recent advances in First-Principles based Molecular Dynamics. *Acc. Chem. Res.*, 55:221–230, 2022.

[198] Laboratorio Nacional de Fusión CIEMAT. El Proyecto TJII: Heliac Flexible. http://www-fusion.ciemat.es/, 2017.

[199] K. Nakashima, R. E. Stoller, and H. Xu. Recombination radius of a Frenkel pair and capture radius of a self-interstitial atom by vacancy clusters in bcc Fe. *J. Phys.: Condens. Matter*, 27:335401, 2015.

[200] G. Nandipati, N. Govind, A. Andersen, and A. Rohatgi. Self-learning kinetic Monte Carlo simulations of Al diffusion in Mg. *J. Phys.: Condens. Matter*, 28:155001, 2016.

[201] R. Neu, J. Riesch, et al. Advanced tungsten materials for plasma-facing components of DEMO and fusion power plants. *Fus. Eng. and Des.*, 109-111:1046–1052, 2017.

[202] R. Ngayam-Happy, C. S. Becquart, and C. Domain. First principle-based AKMC modelling of the formation and medium-term evolution of point defect and solute-rich clusters in a neutron irradiated complex Fe-CuMnNiSiP alloy representative of reactor pressure vessel steels. *J. Nucl. Mater.*, 440:143–152, 2013.

[203] J. Nickolls and W. J. Dally. The GPU computing era. *IEEE Micro*, 30:56–68, 2010.

[204] A. L. Nikolaev and T. E. Kurennykh. On the interaction between radiation-induced defects and foreign interstitial atoms in α-iron. *J. Nucl. Mater.*, 414:374–381, 2011.

[205] K. Nordlund. Historical review of computer simulation of radiation effects in materials. *J. Nucl. Mater.*, 520:273–295, 2019.

[206] K. Nordlund, M. Ghaly, R. S. Averback, M. Caturla, T. Díaz de la Rubia, and J. Tarus. Defect production in collision cascades in elemental semiconductors and fcc metals. *Phys. Rev. B*, 57:7556–7570, 1998.

[207] NVIDIA Corporation. *NVIDIA CUDA C Programming Guide*, March 2017.

[208] OECD Nuclear Energy Agency. Chernobyl: Assessment of radiological and health impacts. Technical report, Nuclear Energy Agency, Issy-les-Moulineaux, France, 2002.

[209] University of York Department of Physics. Inertial Confinement Fusion - York Plasma Institute. https://www.york.ac.uk/physics/ypi/research/laser/icf/, 2017.

[210] D. R. Olander. *Fundamental Aspects of Nuclear Reactor Fuel Elements.* Technical Information Center, Office of Public Affairs - Energy Research and Development Administration, Springfield, Virginia, USA, 1976.

[211] T. Oppelstrup, V. V. Bulatov, A. Donev, M. H. Kalos, G. H. Gilmer, and B. Sadigh. First-passage kinetic monte carlo method. *Phys. Rev. E*, 80:066701, 2009.

[212] T. Opplestrup, V. V. Bulatov, G. H. Gilmer, M. H. Kalos, and B. Sadigh. First-passage Monte Carlo algorithm: Diffusion without all the hops. *Phys. Rev. Lett.*, 97:230602, 2006.

[213] T. Opplestrup, V. V. Bulatov, G. H. Gilmer, M. H. Kalos, and B. Sadigh. First-passage kinetic Monte Carlo method. *Phys. Rev. E*, 80:066701, 2009.

[214] C. J. Ortiz and M. J. Caturla. Simulation of defect evolution in irradiated materials: Role of intracascade clustering and correlated recombination. *Phys. Rev. B*, 75:184101, 2007.

[215] C. J. Ortiz, L. Luneville, and D. Simeone. Binary Collision Approximation. In R. J. M. Konings and R. E. Stoller, editors, *Comprehensive Nuclear Materials.* Elsevier, Oxford, UK, 2nd edition, 2020.

[216] C. J. Ortiz, P. Pichler, T. Fühner, F. Cristiano, B. Colombeau, N. E. B. Cowern, and A. Claverie. A physically based model for the spatial and temporal evolution of self-interstitial agglomerates in ion-implanted silicon. *J. Appl. Phys.*, 96:4866–4877, 2004.

[217] Yu. N. Osetsky, D. J. Bacon, A. Serra, B. N. Singh, and S. I. Golubov. Stability and mobility of defect clusters and dislocation loops in metals. *J. Nucl. Mater.*, 276:65–77, 2000.

[218] Yu. N. Osetsky, D. J. Bacon, B. N. Singh, and B. Wirth. Atomistic study of the generation, interaction, accumulation and annihilation of cascade-induced defect clusters. *J. Nucl. Mater.*, 307-311, Part 2:852–861, 2002.

[219] M. Osuch, T. Wawrzyniak, and M. Majerska. Changes in hydrological regime in High Arctic non-glaciated catchment in 1979-2020 using a multimodel approach. *Adv. Clim. Chang. Res.*, 13:517–530, 2022.

[220] J. D. Owens, D. Luebke, N. Govindaraju, M. Harris, J. Krüger, A. E. Lefohn, and T. E. Purcell. A survey of general-purpose computation on graphics hardware. *Comp. Graph. Forum*, 26:80–113, 2007.

[221] S. Pantousa, K. Mergia, A. Ionescu, et al. Fe^{+} ion irradiation effects in Fe-10at%Cr films irradiated at 300 ° c. *Nucl. Mat. and Energy*, 30:101147, 2022.

[222] K. Papamihail, K. Mergia, F. Ott, Y. Serruys, T. Speliotis, G. Apostolopoulos, and S. Messoloras. Magnetic effects induced by self-ion irradiation of Fe films. *Phys. Rev. B*, 93:100404(R), 2016.

[223] S. Papanikolaou, Y. Cui, and N. Ghoniem. Avalanches and plastic flow in crystal plasticity: an overview. *Modelling Simul. Mater. Sci. Eng.*, 26:013001, 2018.

[224] T. S. Pedersen, M. Otte, S. Lazerson, P. Helander, S. Bozhenkov, C. Biedermann, T. Klinger, R. C. Wolf, and H.-S. Bosch. Confirmation of the topology of the Wendelstein 7-X magnetic field to better than 1:100,000. *Nature Comm.*, 7:13493, 2016.

[225] M. Pelfort, Yu. N. Osetsky, and A. Serra. Vacancy interaction with glissile interstitial clusters in bcc metals. *Phil. Mag. Lett.*, 81:803–811, 2001.

[226] V. Philipps. Tungsten as material for plasma-facing components in fusion devices. *J. Nucl. Mater.*, 415:S2–S9, 2011.

[227] J. C. Phillips, R. Braun, W. Wang, J. Gumbart, E. Tajkhorshid, E. Villa, C. Chipot, R. D. Skeel, Laxmikant K., and K. Schulten. Scalable molecular dynamics with NAMD. *J. Comp. Chem.*, 26:1781–1802, 2005.

[228] Francesco Piazza, Giuseppe Foffi, and Cristiano De Michele. Irreversible bimolecular reactions with inertia: from the trapping to the target setting at finite densities. *J. Phys. Condens. Matter*, 25:245101, 2013.

[229] P. Pichler. *Intrinsic Point Defects, Impurities, and Their Diffusion in Silicon.* Springer-Verlag Wien, Vienna, Austria, 1st edition, 2004.

[230] R. A. Pitts et al. A full tungsten divertor for ITER: Physics issues and design status. *J. Nucl. Mater.*, 438:S48–S56, 2013.

[231] Ianik Plante and Francis A. Cucinotta. Monte-Carlo simulation of particle diffusion in various geometries and application to chemistry and biology. In *Defects in Crystalline Solids S.* InTechOpen, 2013.

[232] S. Plimpton. Fast parallel algorithms for short-range molecular dynamics. *J. Comp. Phys.*, 117:1–19, 1995.

[233] S. Plimpton, C. Battaile, M. Chandross, L. Holm, A. Thompson, V. Tikare, G. Wagner, X. Zhou, C. García Cardona, and A. Slepoy. Crossing the Mesoscale No-Man's Land via Parallel Kinetic Monte Carlo. Unlimited Release, 2009.

[234] A. Prokhodtseva, B. Décamps, and R. Schäublin. Comparison between bulk and thin foil ion irradiation of ultra high purity Fe. *J. Nucl. Mater.*, 442:S786–S789, 2013.

[235] S. Pronk, Sz. Páll, R. Schulz, P. Larsson, P. Bjelkmar, R. Apostolov, M. R. Shirts, J. C. Smith, P. M. Kasson, D. van der Spoel, B. Hess, and E. Lindahl. GROMACS 4.5: a high-throughput and highly parallel open source molecular simulation toolkit. *Bioinf.*, 29:845–854, 2013.

[236] M. A. Puigví, N. de Diego, A. Serra, Yu. N. Osetsky, and D. J. Bacon. On the interaction between a vacancy and self-interstitial atom clusters in metals. *Phil. Mag.*, 87:3501–3517, 2007.

[237] M. Rajput, P. V. Subhash, and R. Srinivasan. Displacement damage study in tungsten and iron for fusion neutron irradiation. *Fus. Eng. and Des.*, 150:111370, 2020.

[238] K. Reuter. *First-Principles Kinetic Monte Carlo Simulations for Heterogeneous Catalysis: Concepts, Status and Frontiers*, chapter 3. Wiley-VCH Verlag, Weinheim, Germany, 2009.

[239] S. A. Rice. Diffusion-limited Reactions. In *Comprehensive Chemical Kinetics*, volume 25. Elsevier Science, 1985.

[240] M. Rieth, S. L. Dudarev, et al. Recent progress in research on tungsten materials for nuclear fusion applications in Europe. *J. Nucl. Mater.*, 432(1-3):482–500, 2013.

[241] M. Rieth et al. Review on the EFDA programme on tungsten materials technology and science. *J. Nucl. Mater.*, 417:463–467, 2011.

[242] M. T. Robinson. Slowing-down time of energetic atomic in solids. *Phys. Rev. B*, 40:10717–10726, 1989.

[243] M. T. Robinson. The temporal development of collision cascades in the Binary-Collision Approximation. *Nucl. Instrum. Methods in Phys. Rev. B*, 48:408–413, 1990.

[244] M. T. Robinson. Computer simulation studies of high-energy collision cascades. *Nucl. Instr. and M. in Phys. Res. B*, 67:396–400, 1992.

[245] M. T. Robinson. The binary collision approximation: Background and introduction. *Rad. Eff.*, 130-131:3–20, 1994.

[246] M. T. Robinson. Attractive interaction potentials and the binary collision approximation. *Rad. Eff.*, 141:1–19, 1997.

[247] M. T. Robinson and I. M. Torrens. Computer simulation of atomic-displacement cascades in solids in the binary-collision approximation. *Phys. Rev. B*, 9:5008, 1974.

[248] J. Rudnick and G. Gaspari. *Elements of the Random Walk: an introduction for advanced students and researchers.* Cambridge University Press, New York, USA, 1st edition, 2004.

[249] A. E. Sand, S. L. Dudarev, and K. Nordlund. High-energy collision cascades in tungsten: Dislocation loops structure and clustering scaling laws. *EPL*, 103:46003, 2013.

[250] A. E. Sand, D. R. Mason, A. De Backer, X. Yi, S. L. Dudarev, and K. Nordlund. Cascade fragmentation: deviation from power law in primary radiation damage. *Mat. Res. Lett.*, 5:357–363, 2017.

[251] I. Santos, L. A. Marqués, L. Pelaz, and P. López. Molecular dynamics study of amorphous pocket formation in Si at low energies and its application to improve binary collision models. *Nucl. Instrum. Methods Phys. Res. B*, 255:110–113, 2007.

[252] I. Santos, L. A. Marqués, L. Pelaz, and P. López. Improved atomistic damage generation model for binary collision simulations. *J. Appl. Phys.*, 105:083530, 2009.

[253] Y. Satoh, H. Matsui, and T. Hamaoka. Effects of impurities on one-dimensional migration of interstitial clusters in iron under electron irradiation. *Phys. Rev. B*, 77:094135, 2008.

[254] R. Schäublin and Y. Chiu. Effect of helium on irradiation-induced hardening of iron: A simulation point of view. *J. Nucl. Mater.*, 362:152–160, 2007.

[255] H. Schroeder and H. Ullmaier. Helium and hydrogen effects on the embrittlement of iron- and nickel-based alloys. *J. Nucl. Mater.*, 179-181:118–124, 1991.

[256] T. P. Schulze. Efficient kinetic Monte Carlo simulation. *J. Comp. Phys.*, 227:2455–2462, 2008.

[257] C. L. Seitz. The cosmic cube. *Comm. ACM*, 28:22–33, 1985.

[258] R. O. Selvitopi, M. M. Ozdal, and C. Aykanat. A novel method for scaling iterative solvers: Avoiding latency overhead of parallel sparse-matrix vector multiplies. *IEEE TPDS*, 26:632–645, 2015.

[259] S. Sharma, C.-H. Hsu, and W.-C. Feng. Making a case for a Green500 list. In *2nd IEEE IPDPS Workshop on High-Performance, Power-Aware Computing*, April 2006.

[260] J. Shi, L. Li, L. Peng, F. Gao, and J. Huang. Atomistic study on helium-to-vacancy ratio of neutron irradiation induced helium bubbles during nucleation and growth in α-Fe. *Nucl. Mat. and Energy*, 26:100940, 2021.

[261] K. Shiiyama, M. M. R. Howlader, S. J. Zinkle, T. Shikama, M. Kutsuwada, S. Matsumura, and C. Kinoshita. Electrical conductivity and current-voltage characteristics of alumina with or without neutron and electron irradiation. *J. Nucl. Mater.*, 258-263, Part 2:1848–1855, 1998.

[262] Y. Shimomura. Point defects and their clusters in f.c.c. metals studied by computer simulations. *Mat. Chem. and Phys.*, 50:139–151, 1997.

[263] J. Shimwell, S. Lilley, L. Morgan, L. Packer, M. Kovari, S. Zheng, and J. McMillan. Reducing beryllium content in mixed bed solid-type breeder blankets. *Fus. Eng. and Des. B*, 109-111:1564–1568, 2016.

[264] H. Shiraga, S. Fujioka, et al. Fast ignition integrated experiments with Gekko and LFEX lasers. *Plasma Phys. Contr. Fusion*, 53:124029, 2011.

[265] D. Shreiner, G. Sellers, J. Kessenich, and B. Licea-Kane. *OpenGL® programming guide: The official guide for learning OpenGL, Version 4.3.* Addison-Wesley, Ann Arbor, USA, 8th edition, 2013.

[266] S. Smolentsev, N. B. Morley, M. A. Abdou, and S. Malang. Dual-coolant lead-lithium (DCLL) blanket status and R&D needs. *Fus. Eng. and Des.*, 100:44–54, 2015.

[267] F. Soisson, C. S. Becquart, N. Castin, C. Domain, L. Malerba, and E. Vincent. Atomistic Kinetic Monte Carlo studies of microchemical evolutions driven by diffusion processes under irradiation. *J. Nucl. Mater.*, 406:55–67, 2010.

[268] Thomas Sokolowski and Pieter Rein ten Wolde. Spatial-Stochastic Simulation of Reaction-Diffusion Systems. In Lev S. Tsimring Brian Munsky, William S. Hlavacek, editor, *Quantitative Biology: Theory, Computational Methods, and Models*, pages 149–178. MIT Press, 2018.

[269] John L. Spouge. Exact solutions for a diffusion-reaction process in one dimension. *Phys. Rev. Lett.*, 60:871–874, 1988.

[270] R. E. Stoller, S. I. Golubov, C. Domain, and C. S. Becquart. Mean field rate theory and object kinetic Monte Carlo: A comparison of kinetic models. *J. Nucl. Mater.*, 382:77–90, 2008.

[271] J. L. Straalsund, R. W. Powell, and B. A. Chin. An overview of neutron irradiation effects in LMFBR materials. *J. Nucl. Mater.*, 108:299–305, 1982.

[272] E. Strohmaier, J. Dongarra, H. Simon, and M. Meuer. Top500: TOP 500 supercomputer sites. https://www.top500.org/, 2016.

[273] A. Stukowski. Visualization and analysis of atomistic simulation data with OVITO - the Open Visualization Tool. *Modelling Simul. Mater. Sci. Eng.*, 18:015012, 2009.

[274] M. P. Surh, J. B. Sturgeon, and W. G. Wolfer. Radiation swelling behavior and its dependence on temperature, dose rate, and dislocation structure evolution. *J. Nucl. Mater.*, 336:217–224, 2005.

[275] S. Takaki, J. Fuss, H. Kugler, U. Dedek, and H. Schultz. The resistivity recovery of high purity and carbon doped iron following low temperature electron irradiation. *Radiat. Eff.*, 79:87–122, 1983.

[276] H. Tanigawa, K. Shiba, A. Möslang, R.E. Stoller, R. Lindau, M. A. Sokolov, G. R. Odette, R. J. Kurtz, and S. Jitsukawa. Status and key issues of reduced activation ferritic/martensitic steels as the structural material for a DEMO blanket. *J. Nucl. Mater.*, 417:9–15, 2011.

[277] K. Tapasa, A. V. Barashev, D. J. Bacon, and Yu. N. Osetsky. Computer simulation of the interaction of carbon atoms with self-interstitial clusters in α-iron. *J. Nucl. Mater.*, 361:52–61, 2007.

[278] A.-A. F. Tavassoli, E. Diegele, R. Lindau, N. Luzginova, and H. Tanigawa. Current status and recent research achievements in ferritic/martensitic steels. *J. Nucl. Mater.*, 455:269–276, 2014.

[279] D. Terentyev, N. Anento, and A. Serra. Interaction of dislocations with carbon-decorated dislocation loops in bcc Fe: an atomistic study. *J. Phys.: Condens. Matter*, 24:455402, 2012.

[280] D. Terentyev, N. Anento, A. Serra, V. Jansson, H. Khater, and G. Bonny. Interaction of carbon with vacancy and self-interstitial atom clusters in α-iron studied using metallic–covalent interatomic potential. *J. Nucl. Mater.*, 408:272–284, 2011.

[281] D. Terentyev, G. Bonny, N. Castin, C. Domain, L. Malerba, P. Olsson, V. Moloddtsov, and R.C. Pasianot. Further development of large-scale atomistic modelling techniques for Fe-Cr alloys. *J. Nucl. Mater.*, 409:167–175, 2011.

[282] D. Terentyev, N. Castin, and C. J. Ortiz. Correlated recombination and annealing of point defects in dilute and concentrated Fe-Cr alloys. *J. Phys.: Condens. Matter*, 24:475404, 2012.

[283] D. Terentyev, X. He, E. Zhurkin, and A. Bakaev. Segregation of Cr at tilt grain boundaries in Fe–Cr alloys: A Metropolis Monte Carlo study. *J. Nucl. Mater.*, 408:161–170, 2011.

[284] D. Terentyev and I. Martín-Bragado. Evolution of dislocation loops in iron under irradiation: The impact of carbon. *Scripta Mater.*, 97:5–8, 2015.

[285] D. Terentyev, P. Olsson, T. P. C. Klaver, and L. Malerba. On the migration and trapping of single self-interstitial atoms in dilute and concentrated Fe–Cr alloys: Atomistic study and comparison with resistivity recovery experiments. *Comp. Mat. Sci.*, 43:1183–1192, 2008.

[286] V. I. Tokar and H. Dreyssé. Accelerated kinetic Monte Carlo algorithm for diffusion-limited kinetics. *Phys. Rev. E*, 77:066705, 2008.

[287] S. Tokunaga, P. Moreau, J. Signoret, et al. Remote experiment with WEST from ITER Remote Experimentation Centre. *Fus. Eng. and Des.*, 154:111554, 2020.

[288] H. Trinkaus. Modeling of helium effects in metals: high temperature embrittlement. *J. Nucl. Mater.*, 133-134:105–112, 2005.

[289] H. Trinkaus, H. L. Heinisch, A. V. Barashev, S. I. Golubov, and B. N. Singh. 1d to 3d diffusion-reaction kinetics of defects in crystals. *Phys. Rev. B*, 66:060105(R), 2002.

[290] H. Trinkaus, B. N. Singh, and A. J. E. Foreman. Mechanisms for decoration of dislocations by small dislocation loops under cascade damage conditions. *J. Nucl. Mater.*, 249:91–102, 1997.

[291] M. A. Tschopp, K. N. Solanki, F. Gao, X. Sun, M. A. Khaleel, and M. F. Horstemeyer. Probing grain boundary sink strength at the nanoscale: Energetics and length scales of vacancy and interstitial absorption by grain boundaries in α-fe. *Phys. Rev. B*, 85:064108, 2012.

[292] Jeroen S. van Zon and Pieter Rein ten Wolde. Green's-function reaction dynamics: A particle-based approach for simulating biochemical networks in time and space. *J. Chem. Phys.*, 123:234910, 2005.

[293] Jeroen S. van Zon and Pieter Rein ten Wolde. Simulating biochemical networks at the particle level and in time and space: Green's function reaction dynamics. *Phys. Rev. Lett.*, 94:128103, 2005.

[294] G. Vayakis, E. R. Hodgson, V. Voltsenya, and C. I. Walker. Chapter 12: Generic diagnostic issues for a burning plasma experiment. *Fus. Sci. Tech.*, 53:699–750, 2008.

[295] S. S. Vazhkudal, B. R. de Supinski, et al. The design, deployment, and evaluation of the CORAL pre-exascale systems. *Proceedings of SC18*, 52:1–12, 2018.

[296] L. Verlet. Computer "experiments" on classical fluids. I. Thermodynamical properties of Lennard-Jones molecules. *Phys. Rev.*, 159:98, 1967.

[297] E. Vincent, C. S. Becquart, C. Pareige, P. Pareige, and C. Domain. Precipitation of the FeCu system: A critical review of atomic kinetic Monte Carlo simulations. *J. Nucl. Mater.*, 373:387–401, 2008.

[298] Z. Vizvary, W. Arter, et al. European DEMO first wall shaping and limiters design and analysis status. *Fus. Eng. and Des.*, 158:111676, 2020.

[299] K. Vörtler, C. Björkas, D. Terentyev, L. Malerba, and K. Nordlund. The effect of Cr concentration on radiation damage in Fe-Cr alloys. *J. Nucl. Mater.*, 382:24–30, 2008.

[300] A. F. Voter. Introduction to the kinetic Monte Carlo method. In Kurt E. Sickafus, Eugene A. Kotomin, and Blas P. Uberuaga, editors, *Radiation Effects in Solids*. Springer, Dordrecht, Netherlands, 2007.

[301] T. R. Waite. Theoretical treatment of the kinetics of diffusion-limited reactions. *Phys. Rev.*, 107:463–470, 1957.

[302] F. Wan, Q. Zhan, Y. Long, et al. The behavior of vacancy-type dislocation loops under electron irradiation in iron. *J. Nucl. Mater.*, 455:253–257, 2014.

[303] J. Wang, Q. Hou, and B. L. Zhang. Migration behavior of self-interstitial defects in tungsten and iron. *Solid State Commun.*, 325:114158, 2021.

[304] G. S. Was. *Fundamentals of Radiation Materials Science.* Springer-Verlag Berlin Heidelberg, Heidelberg, Germany, 1st edition, 2010.

[305] G. S. Was, J. P. Wharry, B. Frisbie, B. D. Wirth, D. Morgan, J. D. Tucker, and T. R. Allen. Assessment of radiation-induced segregation mechanisms in austenitic and ferritic–martensitic alloys. *J. Nucl. Mater.*, 411:41–50, 2011.

[306] J. Wei and F.E. Kruis. GPU-accelerated Monte Carlo simulation of particle coagulation based on the inverse method. *J. Comp. Phys.*, 249:67–79, 2013.

[307] J. Wesson. *The science of JET.* JET Joint Undertaking, Abindgon, UK, 2nd edition, 2006.

[308] B. D. Wirth, G. R. Odette, D. Maroudas, and G. E. Lucas. Dislocation loop structure, energy and mobility of self-interstitial atom clusters in bcc iron. *J. Nucl. Mater.*, 276:33–40, 2000.

[309] B.D. Wirth, G.R. Odette, J. Marian, L. Ventelon, J.A. Young-Vandersall, and L.A. Zepeda-Ruiz. Multiscale modeling of radiation damage in Fe-based alloys in the fusion environment. *J. Nucl. Mater.*, 329-333:103–111, 2004.

[310] A. Xu, C. Beck, D. E. J. Armstrong, K. Rajan, G. D. W. Smith, Paul A. J. B., and S. G. Roberts. Ion-irradiation-induced clustering in W–Re and W–Re–Os alloys: A comparative study using atom probe tomography and nanoindentation measurements. *Acta Mater.*, 87:121–127, 2015.

[311] D. Xu and B. E. Wirth. Modeling spatially dependent kinetics of helium desorption in BCC iron following He ion implantation. *J. Nucl. Mater.*, 403:184–190, 2010.

[312] H. Xu, Yu. N. Osetsky, and R. E. Stoller. Simulating complex atomistic processes: On-the-fly kinetic monte carlo scheme with selective active volumes. *Phys. Rev. B*, 84:132103, 2011.

[313] H. Xu, Yu. N. Osetsky, and R. E. Stoller. Cascade annealing simulations of bcc iron using object kinetic Monte Carlo. *J. Nucl. Mater.*, 423:102–109, 2012.

[314] H. Xu, Yu. N. Osetsky, and R. E. Stoller. Self-evolving atomistic kinetic Monte Carlo: fundamentals and applications. *J. Phys.: Condens. Matter*, 24:375402, 2012.

[315] Z. Yao, M. Hernández-Mayoral, M. L. Jenkins, and M. A. Kirk. Heavy-ion irradiations of Fe and Fe-Cr model alloys part 1: Damage evolution in thin-foils at lower doses. *Phil. Mag.*, 88:2851–2880, 2008.

[316] X. Yi, A. E. Sand, D. R. Mason, M. A. Kirk, S. G. Roberts, K. Nordlund, and S. L. Dudarev. Direct observation of size scaling and elastic interaction between nano-scale defects in collision cascades. *EPL*, 110:36001, 2015.

[317] J. H. You. A review on two previous divertor target concepts for DEMO: mutual impact between structural design requirements and materials performance. *Nucl. Fusion*, 55:113026, 2015.

[318] J. H. You, G. Mazzone, E. Visca, et al. Divertor of the European DEMO: Engineering and technologies for power exhaust. *Fus. Eng. and Des.*, 175:113010, 2022.

[319] J. H. You, R. Villari, D. Flammini, D. Marzullo, and G. Mazzone. Nuclear loads and nuclear shielding performance of EU DEMO divertor: A comparative neutronics evaluation of two interim design options. *Nucl. Mat. and Energy*, 23:100745, 2020.

[320] J.H. You, E. Visca, Ch. Bachmann, T. Barrett, F. Crescenzi, M. Fursdon, H. Greuner, D. Guilhem, P. Languille, M. Li, S. McIntosh, A.V. Müller, J. Reiser, M. Richou, and M. Rieth. European DEMO divertor target: Operational requirements and material-design interface. *Nucl. Mat. and Energy*, 9:171–176, 2016.

[321] W. M. Young and E. W. Elcock. Monte Carlo studies of vacancy migration in binary ordered alloys: I. *Proc. Phys. Soc.*, 89:735–746, 1966.

[322] Q. Yuan, A. Chauhan, E. Gaganidze, and J. Aktaa. Direct observation of dislocation loops shrinkage upon annealing neutron-irradiated Fe–9Cr alloy. *J. Nucl. Mater.*, 542:152401, 2020.

[323] L. E. Zakharov, S. A. Galkin, S. N. Gerasimov, et al. Understanding disruptions in tokamaks. *Phys. Plasmas*, 19:055703, 2012.

[324] E. Zarkadoula, S. L. Daraszewicz, D. M. Duffy, M. A. Seaton, I. T. Todorov, K. Nordlund, M. T. Dove, and K. Trachenko. The nature of high-energy radiation damage in iron. *J. Phys.: Condens. Matter*, 25:125402, 2013.

[325] J. F. Ziegler, J. P. Biersack, and U. Littmark. *The Stopping and Range of Ions in Matter.* Pergamon, New York, USA, 1st edition, 1985.

[326] S. J. Zinkle and J. T. Busby. Structural materials for fission and fusion energy. *Mater. Today*, 12:12–19, 2009.

[327] S. J. Zinkle, A. Möslang, T. Muroga, and H. Tanigawa. Multimodal options for materials research to advance the basis for fusion energy in the ITER era. *Nucl. Fusion*, 53:104024, 2013.

[328] S. J. Zinkle and L. L. Snead. Designing radiation resistance in materials for fusion energy. *Annu. Rev. Mater. Res.*, 44:88.1 – 88.27, 2014.

[329] S. J. Zinkle and G. S. Was. Materials challenges in nuclear energy. *Acta Mater.*, 61:735–758, 2013.

[330] M. Zmitko, Y. Poitevin, L. Boccaccini, J.-F. Salavy, R. Knitter, A. Möslang, A. J. Magielsen, J. B. J. Hegeman, and R. Lässer. Development and qualification of functional materials for the EU Test Blanket Modules: Strategy and R&D activities. *J. Nucl. Mater.*, 417:678–683, 2011.

[331] H. Zohm. Recent ASDEX Upgrade research in support of ITER and DEMO. *Nucl. Fusion*, 55:104010, 2015.