



A Kinetic Monte Carlo Algorithm to Model the Annealing Process to Compute the Dark Current Non Uniformity

Kevin Lemiére, Christophe Inguimbert, Thierry Nuns

► To cite this version:

Kevin Lemiére, Christophe Inguimbert, Thierry Nuns. A Kinetic Monte Carlo Algorithm to Model the Annealing Process to Compute the Dark Current Non Uniformity. RADECS 2020, Sep 2020, Vannes, France. <hal-03829286>

HAL Id: hal-03829286

<https://hal.science/hal-03829286v1>

Submitted on 25 Oct 2022

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



HAL Authorization

A Kinetic Monte Carlo Algorithm to Model the Annealing Process to Compute the Dark Current Non Uniformity

K. Lemière, C. Inguibert, T. Nuns

Abstract: A simulation chain, composed of GEANT4 and a Kinetic Monte Carlo algorithm, dedicated to study the annealing process of defects produced after energetic particle irradiation is presented. Results will be used to compute Dark Current Non Uniformity.

I. INTRODUCTION

During energetic particle irradiation, like in space or in particle accelerators, electronic devices could suffer from different types of damage. Displacement of atoms causes some specific degradations in semiconductor materials [1]. Displacement Damage Dose (DDD) consists in the mean energy deposited by non-ionising process. It causes displacement of atoms surrounding the incident particle track. Depending on the energy involved in the nuclear reaction, isolated defects or cluster of defects are produced [2]. When incident particle displaces a lattice atom, the latter is called a Primary Knock-on Atom (PKA). If the PKA transmits enough energy to displace a neighboring atom, it is called a Secondary Knock-on Atom (SKA). Damage cascades are created through this process. After a short time, displaced atoms could get back to a lattice position, or be definitively kick out of their lattice position: this is called a Frenkel pair, a pair of an interstitial atom and an associated vacancy [3]. Following this initial phase, primary defects start moving out through the semiconductor lattice thanks to their thermal energy. During this diffusion process, a defect can encounter another defect. Depending on the nature of colliding defects, they could either recombine, or associate in a more complex one. Finally, the small fraction of defects which survived could lead to the introduction of electrically active deep trap levels in the semiconductor band gap [4]. They are responsible of the increasing number of charge carriers produced in the semiconductor bulk observed after an irradiation. It results in increase of the intrinsic parasitic current, the so-called dark current.

In addition, energetic particle irradiation is not uniform. This leads to statistical variation of degradation undergone by each pixel: this is called the Dark Current Non Uniformity (DCNU) [5]. In this paper, a simulation chain composed of different numerical tools to compute the DCNU is presented. This toolkit relies on the assumption of proportionality between the increase in dark current and the DDD by means of a universal damage factor [6]. In order to avoid the use of this empirical parameters that involves annealing processes, we have developed this numerical chain capable to estimate recombination rates and nature of surviving defects. Knowing their thermal emission rate we are able to estimate the increase in the DCNU induced by an irradiation. Generation of damage cascade tracks, simulation of annealing process and DCNU computation are treated separately in order to avoid too complex and time-consuming simulations. Each simulation step is run from

input data calculated from the previous one. Firstly *GEANT4* toolkit [6] is used, in order to have a realistic spatial distribution of Frenkel pairs in the volume. Each track is stored in a database. Then, these defect distributions are used as input for a Kinetic Monte Carlo (KMC) algorithm [8]. In this paper, the KMC algorithm is dedicated to the estimation of annealing process that occurred in a defect distribution. In other words, KMC algorithm is used to estimate which stable defect population could be expected at a given time after irradiation.

A statistical study is made on the annealing factor and the final population of complex defects by executing numerous KMC simulations. Introduction of the results of KMC simulations as input of our Monte Carlo method, dedicated to the prediction of dark current distribution through a pixel array, will be presented in the final version of the paper.

II. MODELING ANNEALING WITH A KINETIC MONTE CARLO METHOD

A. *GEANT4* cascade damage simulations to create KMC input data

When an energetic particle creates a damage cascade in the matter, the shape of the cascade and the number of defects produced are not the same for each incident particle. In the aim of having the most realistic annealing simulation, *GEANT4* code has been used to simulate the damage cascade caused by a PKA. An example of a simulated damage cascade caused by a 10keV silicon PKA is shown in Fig. 1.

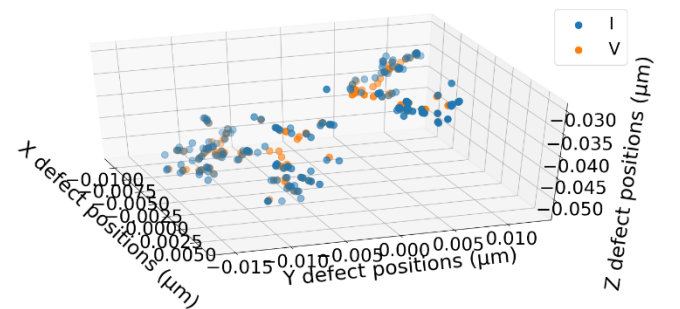


Fig. 1: Initial spatial distribution of Frenkel pairs. This damage cascade results from the simulation of a 10keV silicon PKA in *GEANT4*. Blue points are interstitial defects, orange one are vacancy defects.

The *GEANT4* version used in this work is 9.6 patch 03 version, with the cutoff threshold displacement damage energy set to 21 eV in a simulation box exclusively composed of silicon. The punctual particle source is placed inside the simulation volume.

Position of each vacancy and interstitial created during the simulation is stored. 1000 simulations are executed to have a sufficient database of damage cascades. In order to model the reorganization of the matter after an irradiation, i.e the

annealing of the damage cascade, the Kinetic Monte Carlo (KMC) algorithm has been selected. In the following, the algorithm used in this work is presented, as well as the parameters used to model the diffusion of primary and complex defects in silicon.

B. The Kinetic Monte Carlo Algorithm

The family of KMC algorithms gained his popularity when it successfully explained the Transient Enhanced Diffusion (TED) phenomenon in the 2000's [9]. Since that period, KMC algorithms have been developed and used in different scientific research areas, essentially to model diffusion and reaction processes.

Recently, Jay *et al.* [10][11] proposed a complex chain of simulation to model creation of stable electrically active defects in silicon, coupling molecular dynamics (MD) simulations, a Kinetic Monte Carlo algorithm and *ab initio* calculations. In this paper, the authors investigated the influence of different defect switching states on electrical measurements made after irradiation, and proposed hypothesis to explain the creation of the Random Telegraph Signal (RTS) pixels. This work proposes very accurate simulations at each timescale of the simulation, corresponding to different phases of the physical process. The initial burst is simulated with GEANT4. Then the thermal spike is simulated thanks to molecular simulations in order to get an accurate representation of the process just few picoseconds following the initial damage cascade. MD provides an accurate description of the defects characteristics. Which are then used in a third phase by a KMC algorithm to model the annealing processes for longer time durations.

We focused our work on the KMC simulations in order to get the final population of damages few days up to 1 months after the irradiation. The characteristics of the defects are taken from the literature. However, the migration energy barriers computed in [11] are used in the KMC algorithm developed in this work (see section II.C). The KMC algorithm is used to create a database of annealing factors according to PKA energy, and also to estimate the final population of complex defects. This information will then be used as input for the Monte Carlo based simulation toolkit dedicated to the computation of the Dark Current Non Uniformity on a whole pixel array, and then compared with experimental data.

In this work, the developed KMC method is based on the model described by Nordlund [12]. Firstly, all the primary defects are placed in the simulation box according to a GEANT4 damage cascade simulation.

The defect's kinetic is directly linked to three physical parameters: the migration energy E_m , the intrinsic jump frequency w_0 and the lattice temperature T . Migration energy is equivalent to the potential barrier that a defect must overcome to make a jump in the lattice, i.e move to another location in the lattice. The intrinsic jump frequency $w_{0,i}$ designates the number of attempts made by the defect to overcome the potential barrier by unit of time. From these parameters, the term W_i is deduced, which corresponds to the number of jumps that a defect can make during a unit of time, and computed as follows:

$$W_i = w_{0,i} \exp\left(\frac{-E_{m,i}}{k_B T}\right) \quad (1)$$

The jump process is a thermal activated mechanism, that is why the expression of W_i is similar to an Arrhenius law. It could be associated to a diffusivity or a defect mobility. The KMC algorithm then runs as follow:

1°) The time simulated in the KMC algorithm is initialized. The algorithm checks if some Frenkel pairs are closed enough to recombine directly. It is a first order process, in order to simplify the modeling of the physical process that follows the thermal spike after the crossing of energetic particle. Indeed, a few nanoseconds after that particle deposited its energy, some Frenkel pairs are so closed that they recombine a few moment later, after the thermal energy dissipated.

2°) A random number $u \in [0,1]$ is drawn uniformly and the cumulative function R_i is computed:

$$R_i = \sum_{i=0}^N W_i \quad (2)$$

with N the total number of defects in the system at the beginning of the step. The term R_i equals to the sum of each W_i of the system at the beginning of the step and R_N is noted R for writing simplification. Then the number $u \times R$ is calculated. Finding the index i where $R_i \leq uR < R_{i+1}$ leads to find the defect which will move during the KMC step.

3°) Then comes the migration process, where the defect jumps in a three-dimensional random direction with a radius defined by the user as a KMC input.

4°) At the end of the defect jump, the algorithm tests if there is one or more defects in a specific interaction radius (also entered as KMC input) that could interact with it. If there is more than one candidate in the interaction radius, we suppose in this work that the closest candidate will interact with the defect. Depending on the nature of the candidate, a recombination or association process starts. For example, if a vacancy and an interstitial are within the interaction radius defined by the user, they could recombine and they are both deleted from the system. In the opposite case, if a vacancy is in the interaction radius of another vacancy, they could associate with each other to form a well-known defect: the divacancy.

5°) Finally, the list of defects and their properties is updated, and the migration time step is determined randomly through the KMC characteristic relationship:

$$\Delta t = -\frac{\log u'}{R} \quad (3)$$

where u' a random number $u' \in [0,1]$ drawn uniformly. At the end of the step, the simulated time is updated, the cumulative function is recalculated and a new step begins until the stopping criteria is reached. The stopping criteria can be for example a maximum number of iterations or a maximum simulated time in KMC.

Use of KMC algorithm implies that a defect selected randomly according to its jump probability moves in a random direction during a random migration time, and could interact with a surrounding defect. The most powerful advantage of using KMC algorithm is its adaptability to a large range of timescale. Indeed, as the simulation runs, the time step Δt evolves as function of the number of defects in

the system and their jump frequency. In addition, a great advantage of KMC algorithm is to be able to take into account these different kinetics to make evolve the system. Indeed, at each step of KMC simulation, a defect is selected according to its jump frequency. Hence, the most mobile defects are selected more often than the least mobile defects. These advantages will be detailed more precisely in the final paper.

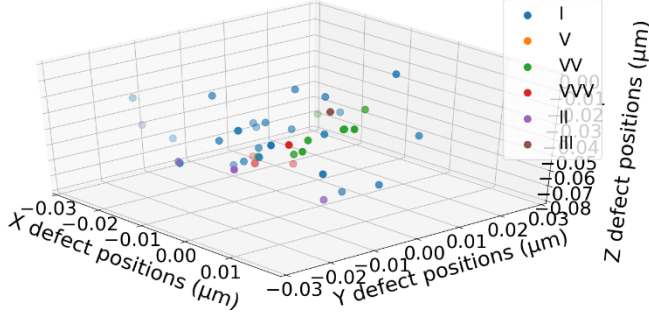


Fig. 2: Final spatial distribution of complex defects. It represents the final state of the Fig. 1 after a KMC simulation of 1h at 300K. Some divacancies and trivacancies have been created (green and red points respectively). Mono-interstitials (blue points) are still present to form di-interstitials (violet points) and tri-interstitials (brown points).

At the end of the simulation, the final distribution of defects that have been produced during the KMC algorithm is stored (see Fig. 2). As the number of each defect species is stored each time an interaction occurred during the KMC execution, the evolution of each species can be observed, as shown in Fig. 3. For example, primary defects, vacancies and interstitials, tend to extinct with time, whereas the number of small complex defects increases with time simulation.

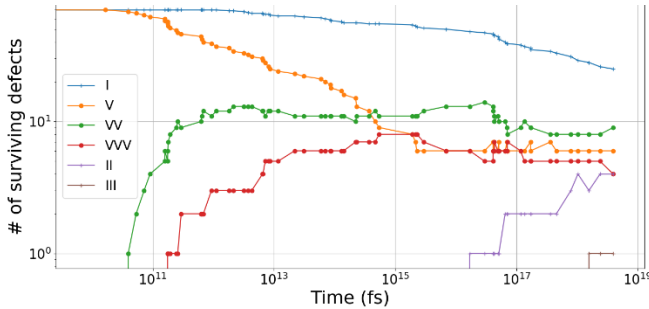


Fig. 3: Evolution of the population of each defect species with time. As the population of primary defects decreases, the population of more complex defects increases. The transition from the initial state (Fig. 1) to the final state (Fig. 2) is observed from this figure.

This KMC algorithm has been validated by comparing its results with those of *Amar et al.* [13]. In [13], the authors investigated a two-species annihilation model to find simple parametric laws which predict the number of surviving defects. These laws are function of initial number of defects and difference in number between two species at initial state. To validate our algorithm, equivalent simulation conditions have been used and similar laws have been found. More validation details will be provided in the final paper.

C. Inventory of the different species considered in the KMC algorithm

During the KMC, different defects could interact through recombination process or association process. A recombination process implies the annihilation of the two defects. Taking the example of an interstitial I and a vacancy V , we obtain:



In this work, the association process occurred when two defects of the same species interact together. Here the classical relationship of divacancy creation:



A complex defect can also experience a partial recombination, and this is taken into account in our KMC algorithm. For example, if a tri-vacancy (complex of three vacancies) meets an interstitial, it could also create a divacancy:



When a defect is created, its coordinates are computed as the center of each Cartesian coordinates.

The maximum complex defect size that can be created is limited to 3 elements, i.e V_3 and I_3 are the biggest defect that our KMC can create. The maximum complex defect size is voluntarily limited because the electrical properties of bigger defect, i.e energy level and electron/hole capture cross sections, are not available in the literature. This information is essential to compute the dark current increase induced by stable electrically active defects introduced by irradiation. Also, because we are interested into long term annealing (one month after irradiation), according to the literature, large complex defects seems to be rare on long time period.

All species considered in this work are summarized in the following table. The jump rate at 300K is also shown, in order to identify the most mobile defect at the temperature simulation. The parameters used to make these calculations will be detailed in the final version of the paper, but they are similar to those used in [11].

Species i	W_i (1/fs) ($T = 300K$)
I : mono-interstitial	1.30e-15
V : mono-vacancy	3.47e-12
I_2 : di-interstitial	1.98e-07
V_2 : di-vacancy	1.59e-19
I_3 : tri-interstitial	1.30e-30
V_3 : tri-vacancy	6.68e-25

Table 1: Each species has their own migration properties. The jump frequency is indicated to identify the most mobile defects.

In silicon, interstitial and vacancy do not have the same migration energy and intrinsic frequency factor. From room temperature to approximately 1000K, the vacancy is largely more mobile than the interstitial. As a consequence, their diffusion kinetics are not the same. According to the parameters chosen for the KMC algorithm, di-interstitial defect has the largest jump frequency, followed by vacancy and interstitial. At the beginning of the simulation, the vacancies are so mobile that they start moving in first. It explains the quick decreasing number of vacancies and the

increasing production of based vacancy complexes during first microseconds observed in Fig. 3. A tenth microseconds after, interstitials start jumping, and those that have not recombined with early moving vacancies create interstitial based complexes. Tri-interstitial and tri-vacancy defects are almost motionless, due to the high migration energy barriers that these defects have to overcome to make a jump. This implies that once one of these defects is created, the probability that it makes a jump is very low. The only way for these defects to evolve is that they encounter an opposite species to do a partial recombination and move again.

The role of the impurities like oxygen or carbon atoms is not investigated in this paper. Because of their low concentrations in the silicon, a specific statistical study is needed to take into account properly the contribution of impurities in the computation of dark current increase.

In the following, the KMC algorithm is used to extract annealing information from *GEANT4* proton irradiation simulations.

D. Statistical study of energetic particle damage annealing by simulation with the KMC algorithm

In order to know how many defects on average will survive during the annealing process after irradiation and to have an idea of which species could be expected at the end of the annealing, the KMC algorithm described previously is used. Annealing scenario will not be the same if the experience is repeated many times. It is a stochastic process and as a consequence, a statistical study is needed to extract relevant information about these simulations.

The KMC algorithm is run 100 times, and at the beginning of each simulation, a damage cascade track is randomly selected and the KMC algorithm is applied. Hence, for a given PKA energy, a distribution of annealing factor is obtained. It will serve for the last simulation step: the computation of the Dark Current Non Uniformity. At the end of this computation step, average and standard deviation of the annealing factor distribution are calculated. In addition to this, the number of each species presents at the end of KMC simulation is stored. Thus the information about which species that are the most representative at the end of the simulated annealing process is available. This information will also be used in the next simulation step. This section will be detailed in the final paper.

III. DCNU MODELING USING AN ANNEALING FACTOR DISTRIBUTION COMPUTED FROM KMC SIMULATIONS

A. Methodology

In this section, the last modeling step of our simulation chain, dedicated to the computation of the dark current increase after irradiation, is described. The original simulation Monte Carlo based toolkit described in [14]–[16] is used. Since it has been described in previous work, it will not be detailed in this paper. According to incident particle energy, fluence and pixel geometry, the simulation toolkit is able to compute the mean number of nuclear interactions that occurred in pixels. From this information, a Monte Carlo simulation is made on the whole pixel array to compute the DCNU. For each nuclear interaction that occurred in a pixel, a given number of atomic displacements computed from a *GEANT4* application is added to the total degradation of the pixel. This number of atomic

displacements corresponds to the degradation at the initial state of the damage cascade. A full study on annealing process of PKA damage structures through KMC simulations will allow us to estimate the final number of atomic displacements present in a pixel and use the electrical properties of the final defect population to compute the dark current in each pixel. This study will be presented in the final paper.

IV. CONCLUSIONS

A complete simulation toolkit dedicated to the computation of the Dark Current Non Uniformity in irradiated pixel arrays is proposed. Its originality consists in the use of a Kinetic Monte Carlo algorithm to estimates the annealing process that occurs after irradiation. A KMC algorithm is used to estimates the distribution of annealing factors for Primary Knock-on Atoms of different energies. The final paper will present a full statistical study And the implementation of this annealing phase within our Monte Carlo based method devoted to the computation of the DCNU.

REFERENCES

- [1] J. R. Srour et J. W. Palko, « Displacement Damage Effects in Irradiated Semiconductor Devices », *IEEE Trans. Nucl. Sci.*, vol. 60, n° 3, p. 1740-1766, june 2013
- [2] J. R. Srour, C. J. Marshall, et P. W. Marshall, « Review of displacement damage effects in silicon devices », *IEEE Trans. Nucl. Sci.*, vol. 50, n° 3, p. 653-670, june 2003
- [3] P. M. Fahey, P. B. Griffin, et J. D. Plummer, « Point defects and dopant diffusion in silicon », *Rev. Mod. Phys.*, vol. 61, n° 2, p. 289-384, apr. 1989
- [4] G. D. Watkins, « Intrinsic defects in silicon », *Mater. Sci. Semicond. Process.*, vol. 3, n° 4, p. 227-235, aug. 2000
- [5] P. W. Marshall, C. J. Dale, et E. A. Burke, « Proton-induced displacement damage distributions and extremes in silicon microvolumes charge injection device », *IEEE Trans. Nucl. Sci.*, vol. 37, n° 6, p. 1776-1783, dec. 1990
- [6] J. R. Srour et D. H. Lo, « Universal damage factor for radiation-induced dark current in silicon devices », *IEEE Trans. Nucl. Sci.*, vol. 47, n° 6, p. 2451-2459, 2000.
- [7] « Overview | geant4.web.cern.ch ». <https://geant4.web.cern.ch/>
- [8] M. Andersen, C. Panosetti, et K. Reuter, « A Practical Guide to Surface Kinetic Monte Carlo Simulations », *Front. Chem.*, vol. 7, 2019
- [9] M. Jung, R. Gunawan, R. Braatz, et E. Seebauer, « A Simplified Picture for Transient Enhanced Diffusion of Boron in Silicon », *J. Electrochem. Soc. - J ELECTROCHEM SOC*, vol. 151, jan. 2004,
- [10] M. Raine et al., « Simulation of Single Particle Displacement Damage in Silicon; Part I: Global Approach and Primary Interaction Simulation », *IEEE Trans. Nucl. Sci.*, vol. 64, n° 1, p. 133-140, jan. 2017
- [11] A. Jay et al., « Simulation of Single Particle Displacement Damage in Silicon; Part II: Generation and Long-Time Relaxation of Damage Structure », *IEEE Trans. Nucl. Sci.*, vol. 64, n° 1, p. 141-148, jan. 2017,
- [12] K. Nordlund, « Kinetic Monte Carlo ». 2006, <http://beam.acclab.helsinki.fi/~knordlun/mc/mc8nc.pdf>.
- [13] J. G. Amar, E. Ben-Naim, S. M. Davis, et P. L. Krapivsky, « Extinction and Survival in Two-Species Annihilation », nov. 2017,
- [14] C. Inguibert, T. Nuns, E. Martin, D. Falguère, et O. Gilard, « Monte Carlo based dcnu modeling », in *2012 13th European Conference on Radiation and Its Effects on Components and Systems (RADECS)*, Biarritz, France, sept. 2012.
- [15] C. Inguibert et al., « Modeling the Dark Current Non-Uniformity of Image Sensors With GEANT4 », *IEEE Trans. Nucl. Sci.*, vol. 61, n° 6, p. 3323-3330, dec. 2014
- [16] M. C. Ursule, T. Nuns, C. Inguibert, H. Bugnet, F. Mayer, et J. Pratlong, « Impact of the Electric Field in the Modelling of the Dark Current Non Uniformity in Pixel Arrays », in *2017 17th European Conference on Radiation and Its Effects on Components and Systems (RADECS)*, Geneva, Switzerland, oct. 2017, p. 1-7