



# More Efficient and Accurate Simulations of Primary Radiation Damage in Materials with Nanosized Microstructural Features or Ion Beams

Ju Li, Yang Yang, and Michael P. Short

## Contents

1	Introduction to Primary Radiation Damage	2
2	BCA-MC Simulations of Primary Radiation Damage under Ion Irradiation	4
3	The Necessity of Full-3D BCA-MC Simulations	9
3.1	Examples Where Full 3D BCA Simulations Are Required	9
3.2	3D Geometry Representation and Ray Tracing in Full-3D BCA-MC Simulations	16
3.3	Dynamically Evolving Full-3D BCA Simulation Structures	19
3.4	Maintaining 3D Simulation Efficiency	22
3.5	Open Sourcing and Stopping Power Databases	23
4	Comparisons Between MC and Molecular Dynamics (MD)	25
5	Outlook for the Next 10 Years	27
	References	29

## Abstract

Primary radiation damage, resulting from the direct ballistic collisions of energetic particles with matter, is the starting point for all of radiation damage. It has traditionally been simulated using binary collision approximations (BCA) Monte Carlo (MC) method in bulk or layered materials, which has served the community well until now. The introduction of nanosized features into materials, whether as 0D dispersoids to pin grain boundaries, 1D dispersoids for removal of helium and sinking defects, or other nanosized features, introduces major errors into conventional bulk/multilayer BCA-MC simulations. This is due to an inability to simulate the exchange of ions at internal/external phase boundaries and the creation of struck atoms with ranges larger than or comparable to the

J. Li (✉) · Y. Yang · M. P. Short

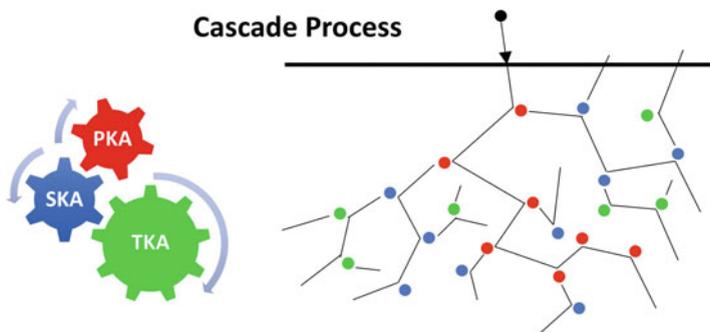
Department of Nuclear Science and Engineering, Massachusetts Institute of Technology, Cambridge, MA, USA

e-mail: [liju@mit.edu](mailto:liju@mit.edu); [yyn@mit.edu](mailto:yyn@mit.edu); [hereiam@mit.edu](mailto:hereiam@mit.edu)

smallest microstructural feature. We first review the fundamentals of BCA-MC simulations with continuous electronic slowing down for energetic ions, followed by identifying where the traditional approaches fail, and ending with a new full-3D simulation capability to correctly model such features. Such simulations drive the planning and interpretation of radiation exposure campaigns, ion implantation, ion modifications, and even the basic definition of radiation damage. A comparison to more accurate but computationally more expensive, molecular dynamics (MD) simulations of radiation damage will be discussed.

## 1 Introduction to Primary Radiation Damage

Radiation damage, by nature, refers to permanent atomic displacements in solid matter caused by energetic particles, including neutrons, charged particles (e.g., electrons, protons, and  $\alpha$  particles), and photons. During the interaction between radiation and solids, a series of atomic displacements may arise, creating damage in materials that impacts their material properties. For example, energetic particles create abundant vacancies and interstitials in materials that may lead to swelling (Short et al. 2015) and embrittlement. Understanding the interplay between radiation and matter is vital not only for designing materials and systems with improved radiation damage tolerance, but also for creating novel devices by defect engineering (Yang et al. 2018a). Atomic displacements start as primary damage, on a timescale of femtoseconds, with the phase known as “primary radiation damage” lasting roughly  $\sim 10$  ps until all atoms ballistically come to rest. Figure 1 illustrates the primary damage process where a series of knock-on atoms are generated: primary knock-on atoms (PKAs), secondary knock-on atoms (SKAs), tertiary knock-on atom (TKAs), and so on. Once the kinetic energies of the incident ions and all the knock-on atoms are low enough (usually much lower than the threshold



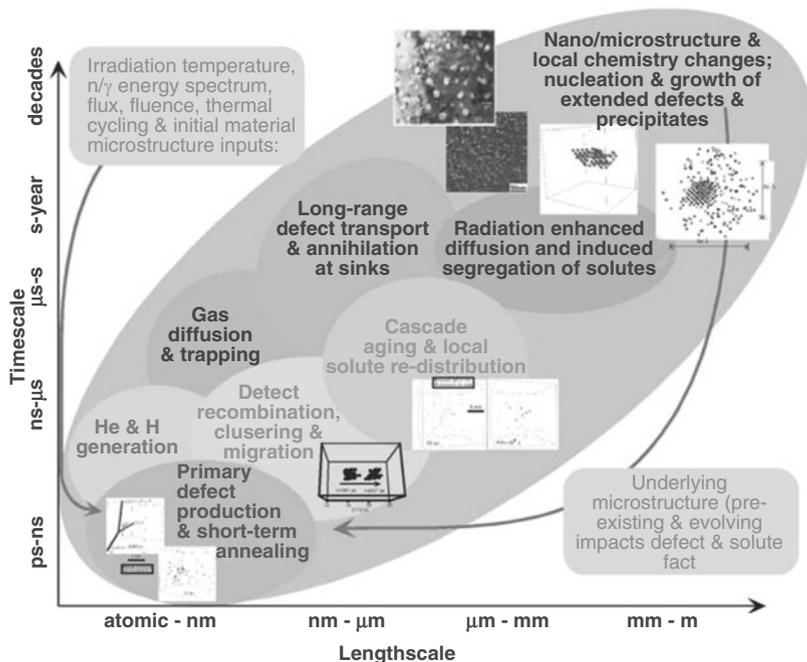
**Fig. 1** Illustration of the PKA/SKA/TKA during the radiation damage cascade process. Once all knock-on atoms come to rest, then the radiation damage cascade, known as “primary radiation damage,” is complete

displacement energy (Crocombette 2005)), then primary radiation damage is said to be complete.

A PKA is born when an incident energetic particle bombards a native atom of the material, transferring kinetic energy  $T$  to the native atom above its threshold displacement energy  $E_d$ . Multiple PKAs (“generation 1”) can be created by the same energetic particle (“generation 0”), and these can be considered the generation-1 leaves of the generation-0 root node of a collision cascade tree. A SKA (“generation 2”)/TKA (“generation 3”)/... is generated by collisions between a PKA/SKA/... and a native atom with the same energy threshold criterion. The whole process is called a “cascade” and terminates when the kinetic energy of every struck atom is below its displacement threshold energy. Rigorously speaking, PKAs, SKAs, TKAs, etc., are not mutually exclusive, since a native atom can in principle be first displaced by an external particle, settle down, and later become displaced by another SKA, although such probabilities are exceedingly small. Along with the cascade process, there is spontaneous recombination (athermal process) which happens below the timescale of picoseconds (Li et al. 2015). Above that, the defects generated will migrate and interact with each other as well as dislocations and different microstructural damage sinks, leading to microstructural evolution and modification of bulk-scale material properties. The time and length scales of these interactions are outlined in Fig. 2. Radiation damage in materials is an inherently multiscale, multiphysics problem (Nordlund and Short 2018).

Neutrons are the main source of atomic displacement damage in nuclear reactors. Although transmutation can create charged particles such as alpha particles in reactors, this is considered as a second order effect (Woo 2005). A neutron may interact with atoms in multiple ways: elastic scattering ( $n, n$ ), radiative capture ( $n, \gamma$ ), inelastic scattering ( $n, n'$ ), fission ( $n, f$ ), and charged particle emission (Yip 2014). Among these mechanisms, elastic and inelastic scattering are the main ways to transfer kinetic energy to target atoms. By contrast, a charged particle may interact with materials in four different ways: (a) elastic collision with electrons, (b) inelastic collision with electrons, (c) elastic collisions with atomic nuclei, and (d) inelastic collisions with atomic nuclei (Yip 2014). (b) and (c) are the main driving forces to stop the transport of charged particles, while (d) is often called “bremsstrahlung” and only becomes prominent for charged particles at very high energy. In BCA-MC simulations, it is often assumed that (a) will only change the kinetic energies of charged particles, while it has no effect on the speed direction. In this context, (c) is the leading term for changing the direction of charged particles and creation of atomic displacements in materials.

In this chapter, we will only focus on the simulations of primary radiation damage. These atomic-scale simulations provide details of the displacement process as well as input parameters for larger scale simulations. These are used as direct inputs for longer timescale models of defect migration, clustering, elemental segregation or dealloying, and evolution of microstructure, mechanical properties, and thermal, electronic, and magnetic properties of irradiated materials. For example, primary radiation damage simulated by BCA-MC codes is used as input for point kinetics simulations of radiation-induced void swelling (Short et al. 2015), MD simulation



**Fig. 2** Illustration of the length and timescales (and inherent feedback) involved in the multiscale processes responsible for microstructural changes in irradiated materials (Odette and Wirth 2005)

of the re-dissolution of Xe fission gas bubbles in uranium dioxide (Schwen et al. 2009), cluster dynamics simulations of nanoscale defect agglomeration (Xu et al. 2012), and so on. Also, it is often required in the planning process of irradiating materials for performance analysis, and its high importance is underscored by being the root of all follow-on simulations. Thus, it is critical that these simulations are as accurate as possible, for as many types and forms of materials as possible. The reader is referred to all other chapters in this section of the Handbook for more information about the subsequent stages of radiation damage and the best practices with which to model them.

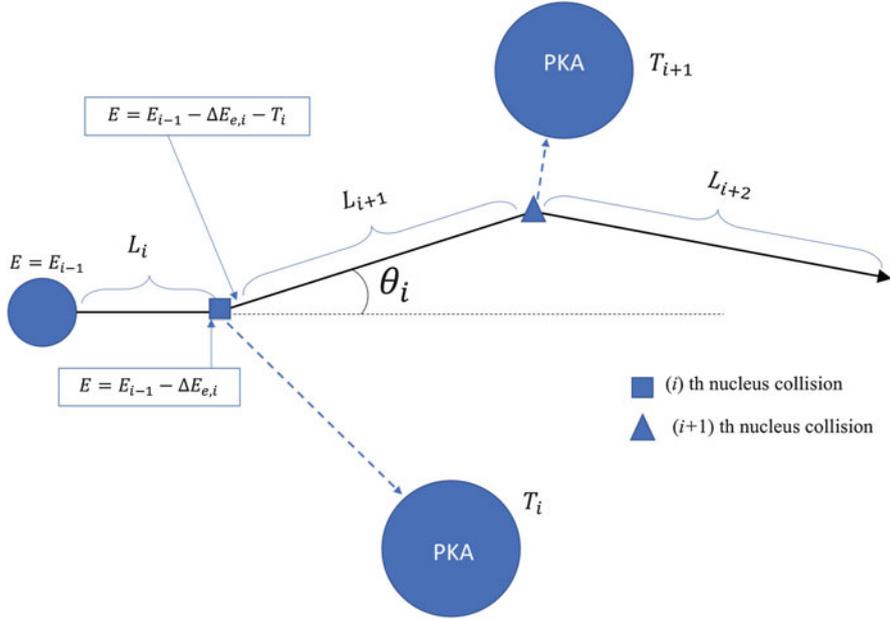
## 2 BCA-MC Simulations of Primary Radiation Damage under Ion Irradiation

In this part of the chapter, we will briefly review what BCA-Monte Carlo (BCA-MC) simulations are, followed by examples of current methods and codes to carry them out, and concluding with the common shortcomings of non-full-3D simulations. This will set the stage for the required modifications to BCA-MC models to accurately model radiation damage in certain types of irradiation scenarios – those

with nanosized features (in target materials or/and incident radiation beam) smaller than or comparable to the mean free path of the ionizing radiation.

Generally, there are two types of BCA simulations for ion irradiation, classified by the assumption of target structure: (1) crysalline structure, simulated by codes such as MARLOWE (Robinson and Torrens 1974); (2) amorphous structure, simulated by codes such as SRIM/TRIM. The latter is simpler and more computationally efficient, and has been more widely used in the past decade. In the following text we will focus on BCA-MC codes based on the random phase approximation (RPA), assuming the target is in amorphous structure. MC simulations based on BCA models are robust methods to understand the detailed cascade process of primary radiation damage (Gilmer and Yip 2005). A BCA-MC simulation tracks all collision events along the trajectories of incident ions. The trajectory of each ion is called a “history.” By computing random trajectories of incident ions (usually about  $10^4$ – $10^8$  atoms), a MC simulation presents statistically meaningful data about the spatial distribution of the injected ions, primary radiation damage in the form of Frenkel pair defects, and energy deposition. The histories of the incoming neutrons and photons (“generation 0”) are often modeled using MCNP (Briesmeister 1986), Geant4 (Agostinelli et al. 2003), and so on, while for incoming ions (generations-1 and above, considered native) the most famous and widely used tool is TRIM/SRIM (Ziegler et al. 2008). This chapter will focus solely on modeling ion-matter interactions, as the damage and defects produced by neutrons and photons are so sparse compared to ions (typically millimeter to centimeter between events, owing to far lower cross sections) that very different methods must be used. Note that in accelerator tests, the external ion accelerated by the accelerator is also considered “generation 0,” whereas the displaced native ions are considered generation 1 and above.

Assumptions are vital for the simplification of the full breadth of interactions to make the physical process more easily computable. Most simulation frameworks to compute primary radiation damage make these same assumptions, so they will be introduced as a common starting point for understanding the simulation process. The cascade processes induced by different ions (generations 0, 1, 2, . . .) are considered to be independent, and only two-body interactions are considered for elastic scattering with target nuclei. The target atom is usually assumed to be stationary before the collision. It is further assumed that ions always travel in straight paths between two binary collisions. The line segment between two successive collisions is called the “free-flight-path,” within which only inelastic scattering with electrons contributes to the change in ion energy, known as electronic energy loss. Therefore, the trajectory of each ion is composed of a series of successive free-flight-paths with elastic nuclear scattering, as depicted in Fig. 3, changing the particle’s energy and direction. The moving direction of ion can only be changed during an elastic scattering event with a nucleus; thus, it is assumed that electronic interactions change the direction of the particle so little that its direction of travel is essentially unchanged. A particle’s history is terminated when it exits the target with zero possibility of reentry, or its energy falls below a pre-defined cutoff energy, which is set depending on the situation being simulated. A proper cutoff energy is



**Fig. 3** Schematic of the modeling process.  $L$  is the free-flight-path length,  $\Delta E_{e,i}$  is the electronic energy loss in  $L_i$ , and  $T_i$  is the kinetic energy loss during the  $i$ -th nuclear collision

important for enhancing the simulation efficiency. The reader is referred to section 6.2.1 in Ref. (Eckstein 1991) for more details about how to set an appropriate cutoff energy. Bremsstrahlung is usually ignored.

A basic algorithm of the BCA-MC simulation modified from (Yang et al. 2018b) is shown in Fig. 4 in the form of pseudocode, to elucidate the general algorithm of BCA simulations while avoiding confusion and unnecessary details. A more detailed program flowchart showing the different treatments of the quick Kinchin-Pease (KP) mode and the full cascade (FC) mode can be found in the documentation of IM3D (Li et al. 2015).

There are four key ingredients which one needs in order to compute in a BCA-MC simulation, which are the free-flight-path length ( $L$ ), the stopping power (determines  $\Delta E_{e,i}/L_i$ ), the scattering angle  $\theta_c$  (determines the direction of  $L$ ), and the nuclear scattering energy loss ( $T_i$ ). The reader is referred to existing references for the calculation of many of these parameters, such as the introduction of the algorithms in TRIM (Biersack and Haggmark 1980), a review of the computer simulation for ion-solid interactions (Eckstein 1991), textbook of SRIM (Ziegler et al. 2008), and recent corrections of stopping powers in the case of high-Z ions damaging low-Z materials (Jin et al. 2014). It is the chosen of the ion-matter interaction models and the numerical methods that differ between legacy and newer BCA-MC simulations. These differences manifest themselves not only in the sampling methods for the free-flight-path length  $L$  and the impact parameter  $P$

```

for all particles do
  Initialization of position, direction and energy;
  /* Take one ion from the 'bank'                                     */
  life←1;
  while life=1 do
    Calculate reduced energy ( $\epsilon$ );
    Calculate the stopping power;
    Sample the free-flight-path length ( $L$ ) and the impact parameter ( $P$ );
    Move to new location and save trajectories;
    Calculate electronic energy loss;
    Reduce energy accordingly ;

    if cross boundary then
      Correct electronic energy loss; Move to boundary; Change material;
      /* Note: The treatment of boundary crossing
         vaires in different codes.                                     */

    else if exit the target then
      life←0;

    else
      /* Collision                                                    */
      Sample direction;
      Calculate nuclear energy loss;
      Reduce energy accordingly ;
      If a recoil is generated, use the KP model to calculate the primary
      damage, or store the recoil in the 'bank' for full-cascade (FC);
      Add tally;
      life←0 if  $E$  is so small;

    end
  end
end

```

**Fig. 4** Basic BCA/MC simulation algorithm in pseudocode, showing general code structure, modified from Ref. (Yang et al. 2018b)

(Biersack and Haggmark 1980) according to the ion and target properties but also in computational approaches to efficiently evaluate scattering angles and stopping powers. Below we briefly review different schemes based on these two aspects, so the reader can appreciate what has evolved in the past decade.

TRIM/SRIM (Biersack and Haggmark 1980; Ziegler et al. 2008), the most widely used tool for BCA simulations, uses semi-empirical relations to compute the electronic and nuclear stopping of ions and struck atoms. These take root in “reduced energies” ( $\epsilon$ ), by which the physical model of the ion-matter interaction is chosen. The ion energy space is further divided into three regions (low, intermediate, and high) or four regions (very low, low, intermediate, and high), with each region

assigned a unique set of semi-empirical equations for the evaluation of the impact parameter  $P$ , the free-flight-path length  $L$ , and the electronic stopping power  $S_e$ . The impact parameter  $P$  is sampled using this reduced energy, which changes its form depending on whether the ion's energy is low, intermediate, or high by assuming hard sphere collisions, a random nonlogarithmic impact parameter, or a random natural log-based impact parameter, respectively. The electronic stopping power in the early version of TRIM (Biersack and Haggmark 1980) is described by two different sets of semi-empirical relations based on different physical models in the low and the high energy regions, while in the intermediate energy region an interpolation method is used. For the ease of computation, fitting formulas for the electronic stopping power are used. A later version of SRIM provides a stopping power database using experimental results, containing over 28,000 stopping values (Ziegler et al. 2010). The scattering angle  $\theta_c$  relies on the chosen interatomic potential and the impact parameter  $P$  as sampled. A key challenge here is the computationally heavy evaluation of the scattering integral (Eckstein 1991), for which various numerical recipes are applied in different codes, such as the so-called "MAGIC" fitting formula in SRIM/TRIM (Biersack and Haggmark 1980), Gauss–Legendre and Gauss–Mehler quadratures reviewed in Ref. (Eckstein 1991), the latest "fast indexing" approach in Corteo (Schiettekatte 2008), and IM3D (Li et al. 2015). In terms of the free-flight-path length  $L$ , a random impact-parameter-dependent  $L(P)$ , a constant  $L$  equal to the mean atomic distance in the target, or a randomly energy-dependent  $L(\varepsilon)$  is applied at low, intermediate, and high energies, respectively. Two newer frameworks, Corteo (Schiettekatte 2008) and IM3D (Li et al. 2015), sample  $L$  over all energies according to a Poisson distribution, with a random inverse-logarithmic sampling of the impact parameter different from that found in SRIM. Otherwise, the basic physics of the older and newer frameworks are quite similar.

One of the major variations among the legacy and newer BCA-MC simulations is the assumption of beam shape and target geometry. SRIM/TRIM, for example, assumes a point incident beam (or "pencil beam") with bulk/multilayer target. The newer codes, however, enable the consideration of arbitrary beam profile (shape and size) as well as 3D target morphologies. The frameworks that is restricted out of the full-3D scope break down in the case of nanoscale microstructural features and radiation beams, or more generally speaking whenever ions encounter features in either the beam or the target that are close to or smaller than the mean free path of the ions. In these cases, codes like SRIM, which can only model bulk or multi-layered materials, fail to account for ions which laterally exit/enter any phase boundary and for a nanoscopically focused beam. This can result in a gross overestimation of primary radiation damage in nanosized features, whether in a radiation mask with pinholes (Bayn et al. 2015), lone carbon nanotubes, microsized mechanical test specimens, oxide dispersion strengthened (ODS) particles, or any other submicron features in materials (Yang et al. 2018a). Thus, more advanced features must be added to BCA codes to accurately simulate primary radiation damage. It should also be noted that complexity often hinders performance, so such features must be added while keeping computational efficiency in mind.

### 3 The Necessity of Full-3D BCA-MC Simulations

While SRIM/TRIM can perform 3D simulations, it is based on several assumptions on the beam profile and target structure. First, the beam source is assumed to be a pencil beam, meaning that ions enter the target via a single point on the surface of target all with identical velocities. Second, the target is at most a multilayered structure with interfaces perpendicular to the longitudinal (i.e., depth) direction, indicating no physical boundaries in the lateral directions. Its 1D output is the most widely used for engineering purposes. If the defect distribution in 3D has a functional form of  $F(x, y, z)$ , then the 1D output along depth direction  $z$  is computed by:

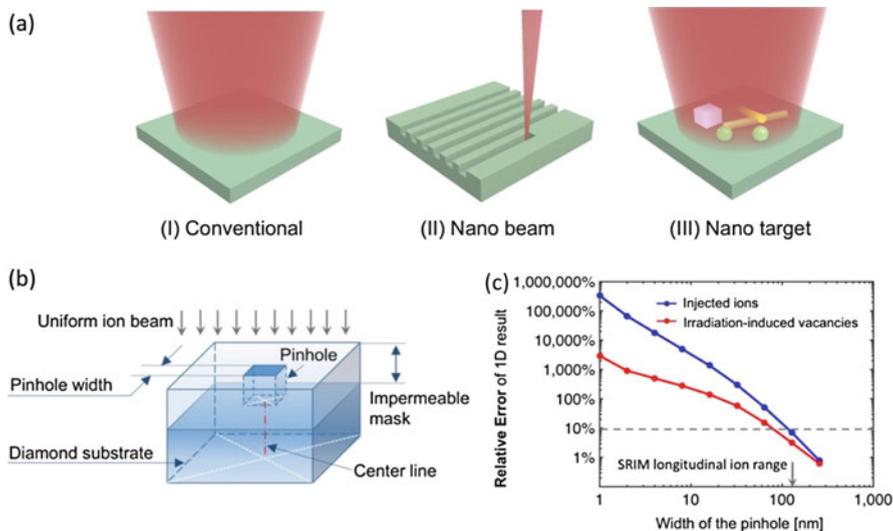
$$f(z) \equiv \iint_{-\infty}^{\infty} F(x, y, z) dx dy \quad (1)$$

$f(z)$  ignores the lateral straggling of ions, which usually ranges from a few nanometers to a few microns. However, it works as a good approximation for the traditional bulk scale experiments, where both the ion beam and the target size are larger than tens of microns (Yang et al. 2018a). Recently, nanoscale ion implantation (nanosized ion beam or target, or both) has drawn expanding interest, facilitating the fabrication of novel devices, such as quantum computers (Shinada et al. 2005; Bayn et al. 2015), magnetometers (Maletinsky et al. 2012), nanowire p-n junctions (Hoffmann et al. 2009). An excellent review about the progress (before 2011) of engineering 0D or 1D nanomaterials using ion radiation has been provided by Krasheninnikov and Nordlund (2010).

#### 3.1 Examples Where Full 3D BCA Simulations Are Required

Because of the breakdown of several key assumptions, these traditional restricted 3D simulations as well as their 1D outputs may be misused, leading to significant errors when predicting defect distributions. Three nanosized effects of ion radiation for MC simulations have been reported, namely, the nano-beam, nano-target (Yang et al. 2018a), and nano-energetic effects (Li et al. 2015), calling for full-3D simulations. The difference between nano-beam and nano-target experiments is illustrated in Fig. 5a.

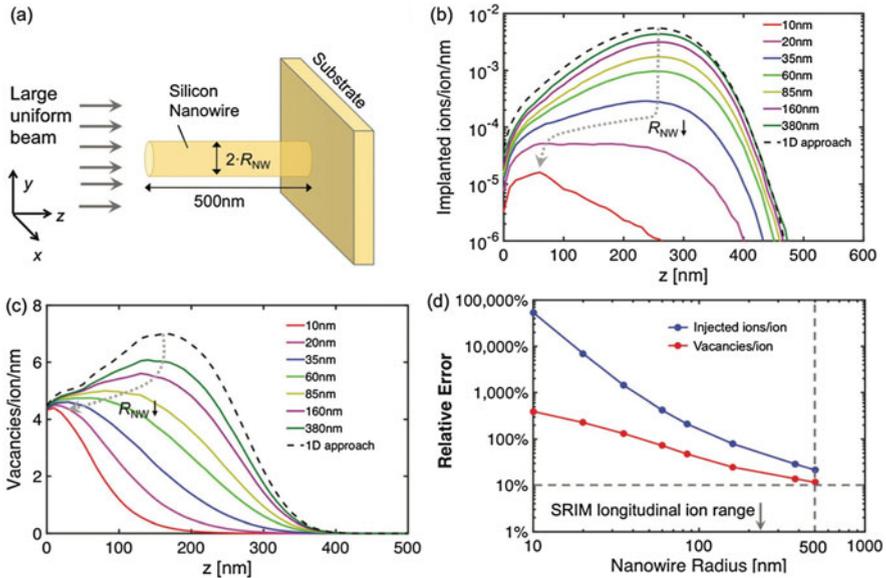
The nano-beam effect arises where the beam size is small and thus the assumptions of 1D output, especially the pencil beam assumption, lose validity. This effect has important implications in nanofabrication, where precision and accuracy relies heavily on beam size. Advanced implantation systems (Watt et al. 2007; Tan et al. 2011; Economou et al. 2012; Yao et al. 2013; PEKIN et al. 2016) and implantation masks with nanoapertures (Pezzagna et al. 2010, 2011; Toyli et al. 2010; Bayn et al. 2015; Scarabelli et al. 2016) have been extensively developed to precisely control the width and position of ion beams. Examples include MeV proton beams with a beam-width below 14 nm (Yao et al. 2013) and silicon masks with pinholes for the



**Fig. 5** (a) Schematic illustration of the difference between a conventional bulk-scale experiment, a nano-beam experiment, and a nano-target experiment. (b) Schematic diagram of the NV center creation experiment. (c) Relative error of the 1D output by SRIM compared to the full-3D simulation. Reproduced from Ref. (Yang et al. 2018a) with permission from The Royal Society of Chemistry

creation of nitrogen-vacancy color centers (Bayn et al. 2015). The relative error of the 1D output of TRIM/SRIM in the application of nanosized ion-beam has been quantified using an example of implanting  $^{15}\text{N}^+$  ions into diamond for the creation of NV centers (Yang et al. 2018a), as shown in Fig. 5b, c. In this case, the effective beam width is equivalent to the width of the square pinhole in the silicon mask. It is shown that the relative error of the traditional 1D output increases dramatically as the width of the pinhole decreases. A full-3D simulation is required to take into account the beam shape information (i.e., the 2D ion density distribution at entrance).

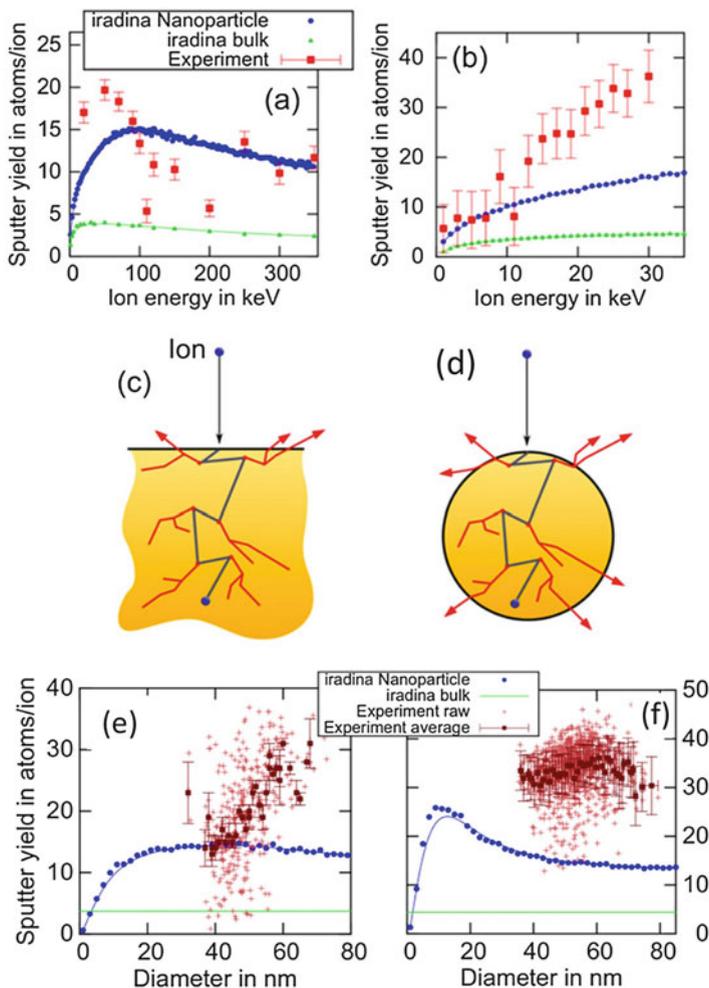
The nano-target effect, on the other hand, becomes prominent when the target size is small, leading to the breakdown of both the pencil beam and multilayer target assumptions. SRIM/TRIM ignores heterogeneity in the lateral directions, failing to consider ion exchange at physical boundaries of the target. Here we take implantation in nanowires as an example (see Fig. 10), which can be used to make nanowire-based devices such as p-n junctions (Hoffmann et al. 2009) and mechanically test ion-irradiated samples (Liontas et al. 2014). The traditional restricted 3D simulations and 1D output are incapable of considering ion leakage from surfaces, overestimating the defects accumulated in the nanowire. Meanwhile, as the target size shrinks, the shape of the defect distribution evolves accordingly, with its peak shifting to a shallower place (Yang et al. 2018a), as shown in Fig. 6b, c.



**Fig. 6** (a) Schematic diagram of the head-on ion implantation experiment. (b, c) Distribution of point defects along  $z$  axis as the nanowire radius decreases. (d) Relative error of the 1D output by SRIM/TRIM. Reproduced from Ref. (Yang et al. 2018a) with permission from The Royal Society of Chemistry

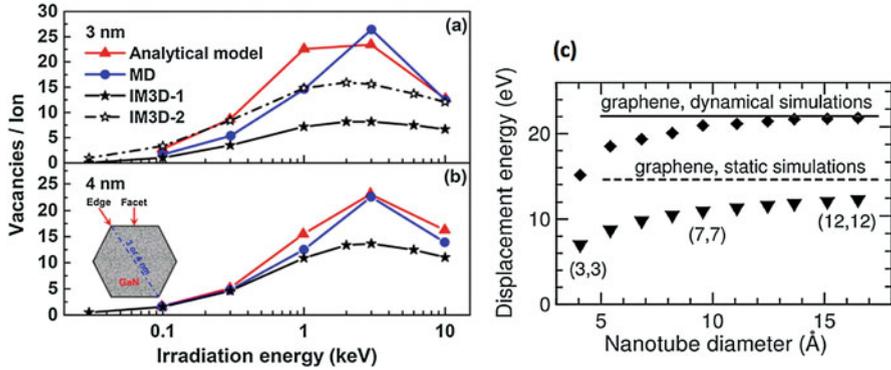
Another example of the nano-target effect is shown in Fig. 7, where the sputtering yield of spherical Au nanoparticles under the irradiation of  $Ar^+$  and  $Ga^+$  is shown to be enhanced when the nanoparticle radius is small (Holland-Moritz et al. 2015) by *iradina* (Borschel and Ronning 2011) simulations, in agreement with experimental results. The enhanced sputtering in nanostructures can be attributed to a large surface-to-volume ratio which allows more ions to exit the target, as illustrated in Fig. 7c, d. Figure 7a, b shows the energy dependence of sputtering yield in bulk and Au nanoparticles, whose diameter follows a Gaussian-shaped distribution with a mean size about 56 nm. Figure 7e, f shows the effect of nanoparticle size on the sputtering yield. The bulk-scale simulation is found to significantly underestimate the sputtering yield, while failing to predict the diameter of the nanoparticle that yields the most sputtering.

The nano-energetic effect is related to the loss of validity of bulk physical parameters at small scales due to quantum confinement, surface stress, and elastic image interactions. These fundamental parameters include electronic stopping power and the threshold energy, which could depend on the sample size. The number of athermal recombinations (Nordlund et al. 2018) could have an even stronger dependency on the size and geometry of the sample, since the sample surface and interfaces provide different venues for knock-out and recombination compared to the bulk. Figure 8 shows an example of the nano-energetic effect for



**Fig. 7** The sputtering yield on bulk and Au nanoparticles as a function of ion energy: (a) Ar<sup>+</sup>; (b) Ga<sup>+</sup>. (c, d) Schematic illustration of the difference between sputtering on bulk sample and nanoparticles. (e-f) The sputtering yield as a function of the diameter of Au nanoparticles: (e) 95 keV Ar<sup>+</sup>; (f) 25 keV Ga<sup>+</sup> (Holland-Moritz et al. 2015)

the ion implantation of Ar into 3–4 nm diameter GaN nanowires. It is known that both the Kinchin–Pease (KP) and full-cascade (FC) modes in SRIM-like MC codes will over-estimate vacancy creation in solids (Stoller et al. 2013); therefore, it is particularly surprising that MC methods underestimate vacancy creation for the nanowire implantation cases, if bulk physical parameters are used (see the difference between MD and IM3D-1 in Fig. 8). By reducing the displacement threshold energy by half, the result of MC simulation (IM3D-2 in Fig. 8) agrees better with that of MD. Comparing Fig. 8a, b, it is also found that the nano-energetic effect is

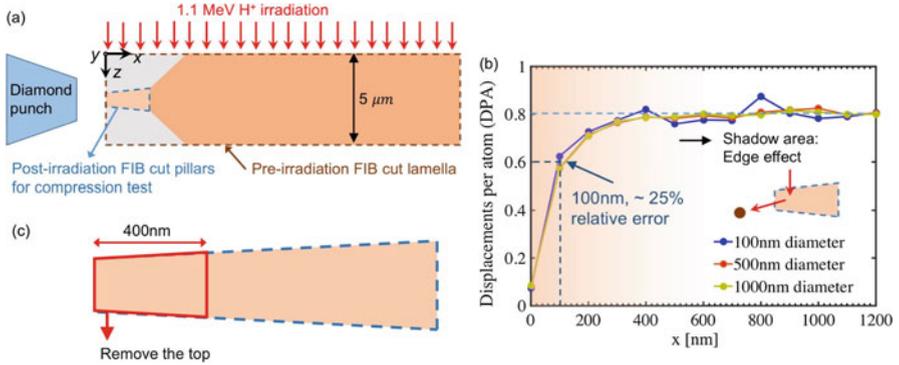


**Fig. 8** Illustration of the nano-energetic effect. (a, b) The difference between IM3D-1 and IM3D-2 is that the former adopts the bulk energy thresholds, while the latter adopts half values of the bulk thresholds. Figure reprinted from (Li et al. 2015) with permission. (c) Dependence of the displacement energy on the carbon nanotube diameter (Krashenninnikov and Nordlund 2010)

less prominent when the size of nanowire increases. The reader is referred to the supplementary document in Ref. (Li et al. 2017) for more details about this nano-energetic effect. Another example is shown in Fig. 8c, where researchers found the displacement energy of carbon nanotubes has a dependence on the diameter (Krashenninnikov and Nordlund 2010).

Although the nano-beam and nano-target effects can be simulated conveniently in full-3D simulation codes such as IM3D, the nano-energetic effect requires corrected physical parameters usually obtained by MD/DFT simulations. It is recommended to consider the nano-energetic effect when the characteristic length of the target is below 20 nm, as the thermodynamic properties vary less sensitively with object sizes above 20 nm (Li et al. 2015 and 2017).

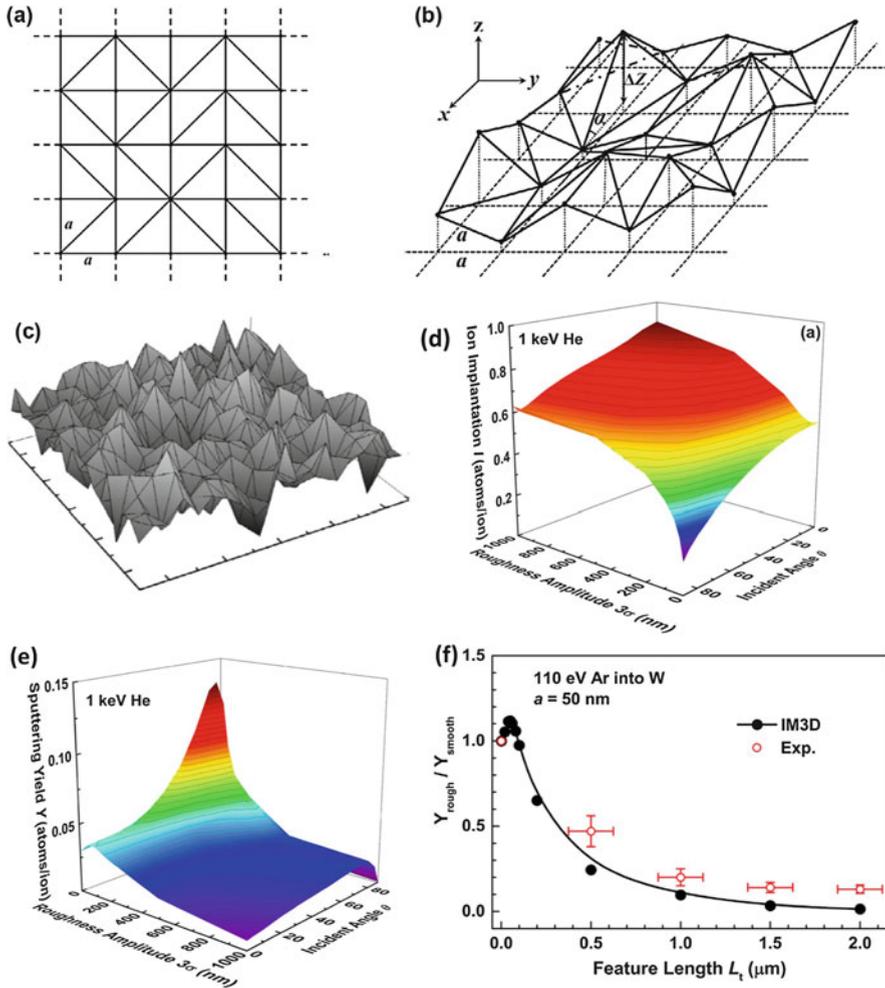
In addition to the three nanosized effects above, researchers also found an “edge-attenuation” effect and a “surface roughness” effect that require full-3D simulations, although their beam/target sizes or surface features are comparatively large. The “edge-attenuation” effect should be considered when the edge of the target is important during a traditional bulk-scale experiment. Figure 9 shows an example (Yang et al. 2018a). Because ion radiation has very limited penetration capability, one has to perform small-scale mechanical experiments to understand the radiation hardening effect of ion irradiated samples. Researchers first irradiated a 5 μm thick Cu thin film by 1.1 MeV protons and then used a focused ion beam (FIB) to cut a small pillar at the edge for small-scale mechanical experiments, as illustrated in Fig. 9a. It is vital for the pillar to have a uniform distribution of radiation damage. However, a full-3D simulation performed in IM3D revealed that there is an “edge attenuation” of radiation damage along the *x* axis from the inner to the edge. Therefore, one should remove the top (i.e., from the *x* = 0 to 400 nm) to avoid uneven radiation dose distributions in the pillar. This is an absolutely critical finding for those undertaking microscale compression pillar irradiation experiments, as



**Fig. 9** (a) Schematic drawing of the experimental setup. (b) Radiation damage as a function of the distance to the edge along  $x$  axis. (c) Schematic drawing about the removal of the top of the pillar (Yang et al. 2018a)

without these 3D BCA-MC simulations this effect would not have been discovered and would have continued to alter the consistency of microcompression experiments under irradiation compared to bulk experiments.

The “surface roughness” effect has important implications in the study of ion-surface interactions. For example, in fusion reactors, the plasma facing materials (PFMs) suffer from the impact of low energy (10–1000 eV) and high fluence (up to  $10^{24}$  particles/m<sup>2</sup>-s) D/T/He ions, which lead to sputtering and ion retention (Nordlund et al. 2014). The surface morphology of PFMs can be modified by the plasma. As a result, a series of surface features are formed, including fuzz (Baldwin and Doerner 2010) and surface blisters (Wang et al. 2001). Surface roughness evolution will in turn have a significant effect on sputtering and ion retention. Traditional SRIM/TRIM is only designed for bulk targets with smooth, nonevolving surfaces; thus, it will fail to predict this surface roughness effect. Previously, simulation codes based on fractal surface geometry models, such as VF-TRIM (Ruzic 1990), ACAT (Kenmotsu et al. 2005), and ITMC-F (Hu and Hassanein 2012), have been developed to understand this effect. Recently, a rough-surface geometry model based on finite element triangular mesh (FETM) method has been developed using the IM3D framework (Li et al. 2017). Since the FETM method is simpler and more realistic than the fractal model in terms of validation with experimental atomic force microscopy (AFM) images, here we will only introduce the former. In the FETM model, the surface is first divided into square pixels with a side-length of  $a$  and then subdivided into triangles by randomly selecting one diagonal of each pixel, as shown in Fig. 10a. The height  $Z$  of each point in the mesh is sampled using a truncated Gaussian distribution  $f(Z) \propto \exp(-Z^2/(2\sigma^2))$ , where  $Z \in [-3\sigma, 3\sigma]$ . Therefore, the surface roughness can be adjusted conveniently and intuitively by controlling the surface roughness amplitude  $3\sigma$  and square mesh size  $a$ . The primary ion retention and sputtering yield as a function of surface  $3\sigma$  and ion incident angle are shown in Fig. 10d, e.



**Fig. 10** (a–c) Schematic illustration of the surface roughness generated by the FETM method. Dependence of the ion retention (d) and sputtering yield (e) on the roughness amplitude and incident angle (1 keV He → W). (f) Effect of the surface roughness on the sputtering yield and validation of the IM3D simulations (1 keV He → W) (Li et al. 2017)

It is found that as the surface roughness increases, the primary ion retention will increase while the sputtering yield will decrease. A comparison of the experiment and IM3D simulation is provided in Fig. 10f, showing significant improvement of sputtering yield by making the surface smoother. However, it is surprising that the highest sputtering yield occurs before the surface roughness reaches zero, indicating an optimum surface roughness leading to the highest sputtering yield.

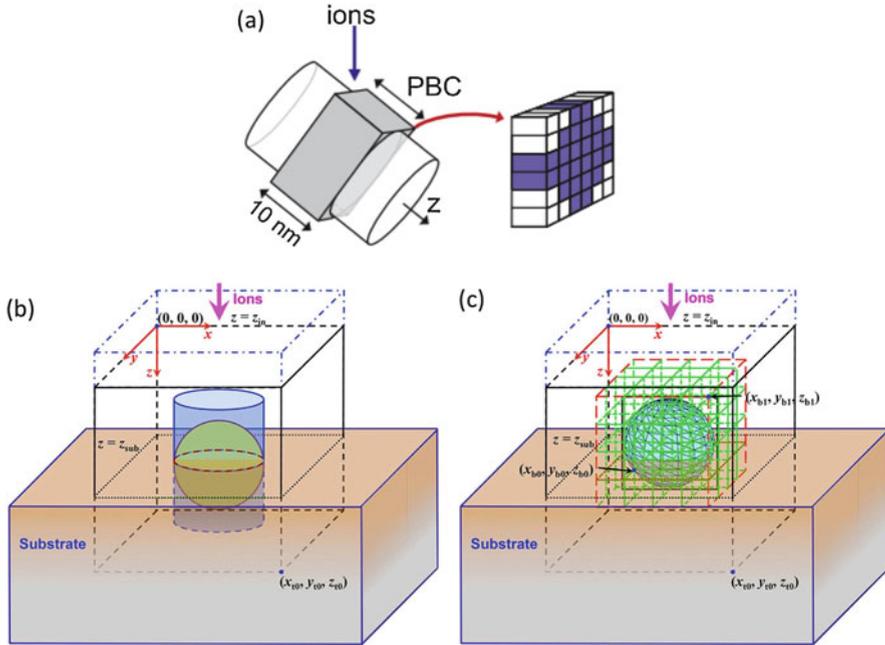
### 3.2 3D Geometry Representation and Ray Tracing in Full-3D BCA-MC Simulations

A key step in full-3D simulation is to build the 3D structure of target. It is significant for the “language” of 3D geometry description to be general and flexible. Also, during the simulation, after sampling the free-flight-path  $L$  to travel in the next step, one should determine if the projectile will enter another material or not during the path  $L$ . If a boundary is crossed, the electronic energy loss will need correction because the free-flight-path  $L$  covers regions of different materials with different stopping capabilities. Such treatment is involved in the process called “particle tracking” or “ray tracing”, and its method varies in different codes. For example, in IM3D (Li et al. 2015), the program follows a similar methodology to CD-SEM (Li et al. 2008). It calculates the sub-free-flight-path length  $L_i$  (where  $L = L_1 + L_2 + \dots + L_n$ , and  $L_1$  is the distance to the closest boundary along the direction of the ion’s path at the starting point of  $L$ ) in each region covered by  $L$ , according to the electronic stopping power ratios between different regions. The total electronic energy loss is corrected accordingly. Besides, the FETM module in IM3D accounts for refraction at each interface, *i.e.*, change of flight path direction due to entering different materials. In *iradina* (Borschel and Ronning 2011), the correction of electronic stopping is ignored because their free-flight-paths and ion energies are so small that nuclear stopping is the dominant mechanism of energy loss. In neutron transport simulation codes such as OpenMC (Romano and Forget 2013; MIT 2011), the free-flight-path is updated to be the distance to the closest boundary, followed by moving the ion to the intersection between the boundary and the free-flight-path, and then re-sampling  $L$  using the new material’s properties. The readers are referred to Chapter 8.3 of Ref. (Bielajew 2001) for details about the treatment in such neutron simulations. The selection of the 3D geometry representation model and the ray-tracing method are closely related. Here we will briefly introduce two kinds of 3D geometry representation methods with some common practices of ray-tracing.

1. The voxel method. First define the whole target space as a rectangular box, then subdivide it into equal-sized rectangular voxels (cells). Each cell is assigned to a pre-defined material, which may be a vacuum. In this case, a curved surface is approximated by a stepped surface. The progression of the ion along a path  $L$  is composed of a sequence of finding the intersection of rays with the next surrounding voxel, followed by moving the ion to the next voxel until the entire path length  $L$  is finished or a “boundary crossing” event occurs. A “boundary crossing” event means that the path  $L$  passes through voxels of different materials. The simulation accuracy of these methods increases with the number of voxels, while the computation load increases at the same time. Note that there is a lower limit of the voxel size in TRI3DRYN because of the consideration of dynamic evolution (Möller 2014). The voxel method has the advantage of easy implementation of dynamic evolution. For example, the formation of voids in

the bulk can be simulated by reducing the density of certain voxels through some preset criteria. However, the voxel method may lead to unavoidable artifacts. For instance, when predicting the sputtering yield of the stepped planes, ions that have exited may re-enter and create further unphysical sputtering/cascades. Also, there is still a lack of understanding about the correction of electronic stopping power at the step edges (Li et al. 2015). Last but not least, domain size adaptivity and scaling issues should also be considered (Borschel and Ronning 2011; Möller 2014), adding to the complexity of simulations.

2. The surface method. The 3D structure is constructed by defining the surfaces of the target in real-space  $(x, y, z)$  using mathematical (vector) descriptors. For example, the vector descriptor for the surface of a spherical nanoparticle centered at the origin with radius  $R$  is  $x^2 + y^2 + z^2 = R^2$ . If an ion is at position  $(x_0, y_0)$ , then  $x_0^2 + y_0^2 + z_0^2 < R^2$ ,  $x_0^2 + y_0^2 + z_0^2 = R^2$  and  $x_0^2 + y_0^2 + z_0^2 > R^2$  means the ion is inside the nanoparticle, at the surface of the nanoparticle and outside the nanoparticle, respectively. Suppose an ion bombards an atom at  $(x_0, y_0)$ , after which the new speed of the ion is  $\vec{v} \equiv (u, v, w)$ . We then sample the distance to travel next with a result of  $L$ . To determine if the ion will enter or exit the surface, one should first compute the minimum distance to the surface of the nanoparticle using  $(x_0, y_0)$  and  $\vec{v} \equiv (u, v, w)$  as  $d = -\Lambda \pm \sqrt{\Lambda^2 - \varpi}$ , where  $\Lambda = x_0u + y_0v + z_0w$ , and  $\varpi = x_0^2 + y_0^2 + z_0^2 - R^2$ . If there is no real solution, it means that the ray will never intersect the sphere, no matter how large  $L$  is. If  $\varpi < 0$ , the ion is inside the sphere at the beginning, so there will be two solutions (one positive and one negative). If  $\varpi > 0$ , the ion is outside the particle; thus, the solutions are both positive or negative. If  $L > d > 0$ , a “boundary crossing” event will occur. When the target structure is more complicated, more surfaces (mathematical descriptors) are required. If there is more than one surface, one should compute the minimum distance to each surface along the ray and sort them in order to identify the sequence of surfaces that the ion will intersect along the path  $L$ . There are several ways to construct arbitrary target structures using the surface method, here we will just briefly introduce two of them: the constructive solid geometry (CSG) method (Li and Ding 2006) and the previously introduced FETM method (see Sect. 3.1) (Zhang et al. 2012). IM3D supports both the CSG and the FETM methods. The CSG method constructs the complicated target surfaces by Boolean operations on simple geometries (such as the sphere, tetrahedron, cuboid, ellipsoid, and taper). These operations include union, difference, and intersection. Therefore, for simple nanostructures such as nanoparticles or nanowires, it provides the best balance between convenience and accuracy. The reader is referred to the documentation webpage of OpenMC (MIT 2011) for details about the CSG method. The FETM method, on the other hand, is more suitable for building arbitrarily complex structures. The triangular mesh for a complicated structure can be easily generated by programs such as Gmsh (Geuzaine and Remacle 2009). Also, triangular mesh enables easier determination of intersection points of a straight line with surfaces. The FETM method is further aided by a spatial subdividing technique (Li et al. 2008) to avoid the computation of intersections with all triangular meshing surfaces. Only



**Fig. 11** Schematic illustrations of different geometry construction method: (a) voxels (Borschel and Ronning 2011); (b) CSG and (c) FETM (Li et al. 2015)

the surfaces contained in the cubes along the path  $L$  are considered for such intersection searching and distance computation. For complicated geometries, the number of times for finding the intersection points with surfaces on a free-flight-path  $L$  using the FETM method is usually much smaller than that using the CSG method, leading to a higher computational efficiency. The FETM method can be extended to the finite element tetrahedron method for easier implementation of dynamic evolution in materials during irradiation.

The differences between the voxel method and the surface method are illustrated in Fig. 11. Currently, available full-3D MC codes for the simulation of energetic particles in matter include MCNP (LANL 2018), OpenMC (Romano and Forget 2013; MIT 2011), Geant4 (Agostinelli et al. 2003), *iradina* (Borschel and Ronning 2011), TRI3DYN (Möller 2014), IM3D (Li et al. 2015), MyTRIM (Schwen et al. 2009 and 2018), and Corteo 3D (Schiettekatte and Chicoine 2016). The reader is referred to the documentation of these codes for more details about their methods for particle tracing and 3D geometry construction.

### 3.3 Dynamically Evolving Full-3D BCA Simulation Structures

Under high fluence of ion beam radiation, the target can be modified, leading to changes in the morphology and density in material due to sputtering and relocation of atoms by atomic collisions. A “dynamic” code is designed to take into account this evolution of the target during ion implantation, with well-documented application examples where such dynamic evolution is absolutely necessary to correctly simulate experimental results. Examples include the secondary ion mass spectrometry (SIMS) depth profiling technique, plasma–wall interactions in fusion reactors, and magnetron sputtering for thin film processing. TRIDYN (Möller and Eckstein 1984) (1D simulation) and TRI3DYN (Möller 2014) (full-3D simulation) have been developed to model this dynamic evolution under ion radiation. TRI3DYN uses the voxel method (see Section 3.2) for the construction of target 3D structures. Here we will briefly introduce TRI3DYN, for more details please refer to Möller (2014).

In TRI3DYN, the voxels are classified into three types: bulk, surface, and vacuum, as shown in Fig. 12. The neighborhoods of a voxel (say “voxel 1”) are classified by the distance between voxel 1 and its neighborhood voxels. There are 6 first neighbors, 12 second neighbors, and 8 third neighbors. A lower limit for the voxel size is set ( $\lambda < \min(\Delta x, \Delta y, \Delta z)$ ), where  $\lambda$  is the mean free path and  $\Delta x, \Delta y, \Delta z$  are the size of voxel along the  $x, y, z$  directions, respectively, so that the path length  $L$  between two successive collisions always ends within the volume that is composed of the 26 first-to-third nearest neighbors. Suppose there are  $N_c$  types of component in the material, then the partial atomic density of a component  $l$  ( $1 \leq l \leq N_c$ ) in the voxel with spatial index  $(i, j, k)$  is  $n_l(i, j, k)$ . TRI3DYN and TRIDRYN both assume that after the relaxation, a constant atomic volume is reached with the following relation:

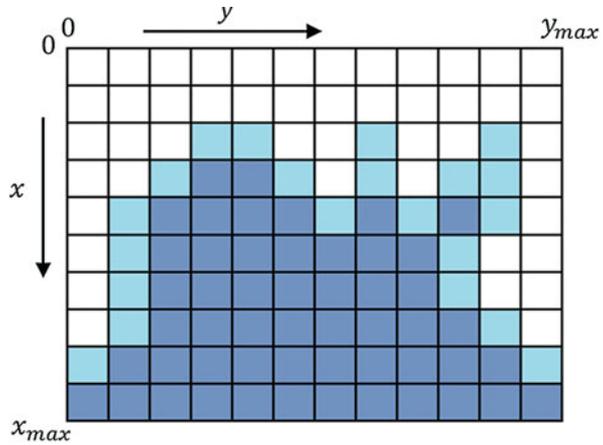
$$\frac{1}{n} = \sum_l \frac{q_l}{n_l^0} \quad (2)$$

where  $n, n_l^0$ , and  $q_l$  denote the total atomic density, a predefined atomic density of component  $l$  and atomic fractions of component  $l$  ( $0 \leq q_l \leq 1$ ), respectively. With this condition at a relaxed state, one can define a reduced partial density:

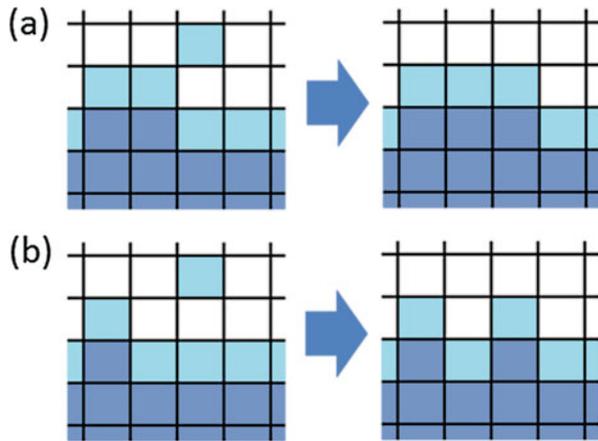
$$d = \sum_l d_l = \sum_l \frac{n_l}{n_l^0} \quad (3)$$

where  $d_l$  is the partial density of the component  $l$ . When  $n_l = q_l \cdot n$ , a “nominal” density is reached such that  $d = 1$ . Based on this,  $d = 0, d > 1$ , and  $0 < d < 1$  represent “vacuum,” “over-dense,” and “under-dense,” respectively. The simulation procedure is an iteration of the following pseudocode steps:

**Fig. 12** Schematic of the voxel types in TRI3DRYN. Dark blue, light blue, and white represent bulk, surface, and vacuum voxels, respectively (Möller 2014)

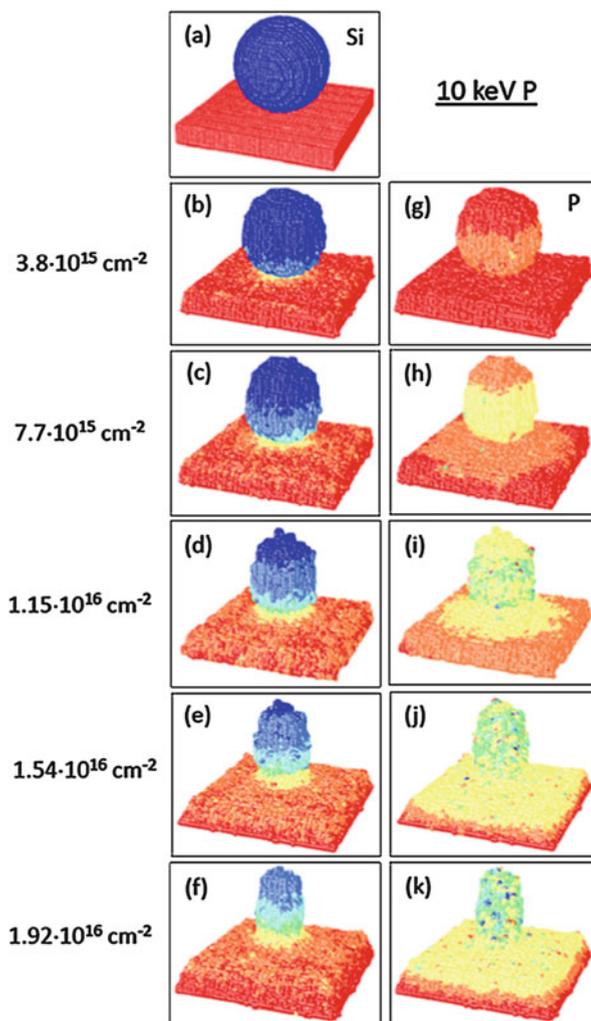


**Fig. 13** Schematic of the “local surface smoothing” in TRI3DRYN. (a) “hanging” or (b) “flying” occurs rarely during the simulations (Möller 2014)



1. Perform a SRIM/TRIM-like BCA simulation for the cascade resulting from one (default) or more incident ions. Record point defect distributions (interstitial atoms and vacancies).
2. Compensation and recombination of interstitials and vacancies. First, the interstitials and vacancies within the same voxel will recombine. Next, the program will search for neighboring voxels within a predefined recombination radius and enable recombination.
3. Incorporation of the remaining point defects from step (2) into the voxels.
4. Relaxation of the voxels, so that nominal density ( $d = 1$ ) is obtained for each bulk voxel, while the surface voxel is permitted to be under-dense ( $d < 1$ ).
5. “Local surface smoothing” will be performed if this kernel is activated. Examples of smoothing methods are illustrated in Fig. 13.
6. Optionally, adjust the surface layers so that the top two layers are kept a vacuum.

**Fig. 14** Evolution of a spherical Si nanoparticle (diameter = 30 nm) on  $\text{SiO}_2$  substrate during the implantation of 10 keV P, whose initial velocity is normal to the substrate surface. (a–f) Si atomic fraction. Color coding: 0.333 (red)  $\rightarrow$  1 (blue). (g–k) P atomic fraction. Color coding: 0 (red)  $\rightarrow$  0.1 (blue) (Möller 2014)

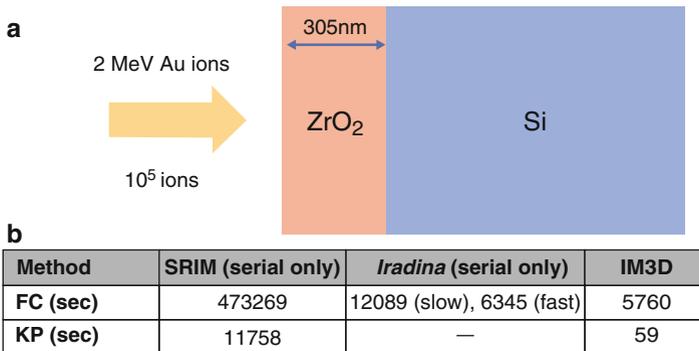


Ion implantation is a popular tool to tailor the composition and morphology of materials (Krasheninnikov and Nordlund 2010; Borschel et al. 2011) and is promising to improve the functional properties of nanostructured materials, such as using  $\text{Ar}^+$  to modify the electronic properties of single-wall carbon nanotube (Gómez-Navarro et al. 2005) or single layer  $\text{MoS}_2$  (Chen et al. 2018). An example of the simulation by TRI3DRYN is shown in Fig. 14, where  $\text{P}^+$  at 10 keV is modifying the morphology and composition gradually. The 3D surface morphology evolution of the nanoparticle (changing from a sphere to pillar) can be well captured by the TRI3DRYN simulation.

### 3.4 Maintaining 3D Simulation Efficiency

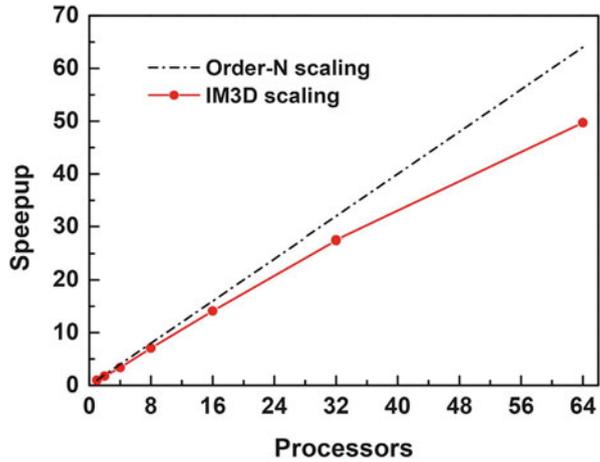
The usage of full-3D simulations and consideration of dynamic evolution will potentially increase the computational load. The increase of computational time can be 10–10,000 $\times$  for a 3D target with a complicated surface morphology. Therefore, it is vital to enhance the efficiency of the simulation codes. Here we will briefly introduce the methods adopted in static BCA-MC simulations (such as Corteo and IM3D) for the improvement of computational efficiency.

One of the most computationally expensive parts in the simulation of radiation damage cascade is the sampling of  $\theta_c$ . Traditionally, this requires one to solve the scattering integral equations, leading to a significant computational load. SRIM uses the so-called “MAGIC” formula to sample the scattering and azimuthal angles for better efficiency. Based on it, a “fast indexing” technique using database evaluation was latter proposed (Yuan et al. 1993) to further enhance the efficiency to about 18 times faster than the “MAGIC” formula. Instead of computing directly from the “MAGIC” algorithm, this method utilizes interpolated tables for the scattering angle  $\sin^2(\theta_c)$  and looks up the needed values in the tables. It is worthwhile to note that the table is indexed using the binary representation of floating point numbers so that computing logarithms can be avoided (Schiettekatte 2009). This method also improves accuracy and memory usage. Similarly, one can obtain the electronic stopping power using precalculated tables (generated using “SRModule.exe” (Ziegler 2004)). This “fast indexing” method is now available in Corteo (Schiettekatte 2009), *iradina* (Borschel and Ronning 2011), and IM3D (Li et al. 2015). A comparison of the efficiency is provided in Fig. 15. The readers can refer to Schiettekatte (2009) for more details about this method. In addition, a ray-tracing technique based on the FETM 3D geometry representation and spatial subdivision is introduced in IM3D



**Fig. 15** A comparison of the computational efficiency (single CPU). (a) Schematic illustration of the simulation setup: 2 MeV Au ion irradiation on a dual-layer substrate. The top layer is a ZrO<sub>2</sub> film with a thickness of 305 nm and Si is infinitely thick. (b) Comparison of the time for the simulation of 10<sup>5</sup> ions. Reproduced from Ref. (Li et al. 2015), which is licensed under CC BY 4.0

**Fig. 16** Wall clock scaling of IM3D running Message Passing Interface (MPI), for the system of a 305 nm  $\text{ZrO}_2$  film on Si under a total of 100000 Au ion irradiations with ion energies of 2.0 MeV. Reproduced from Ref. (Li et al. 2015), which is licensed under CC BY 4.0

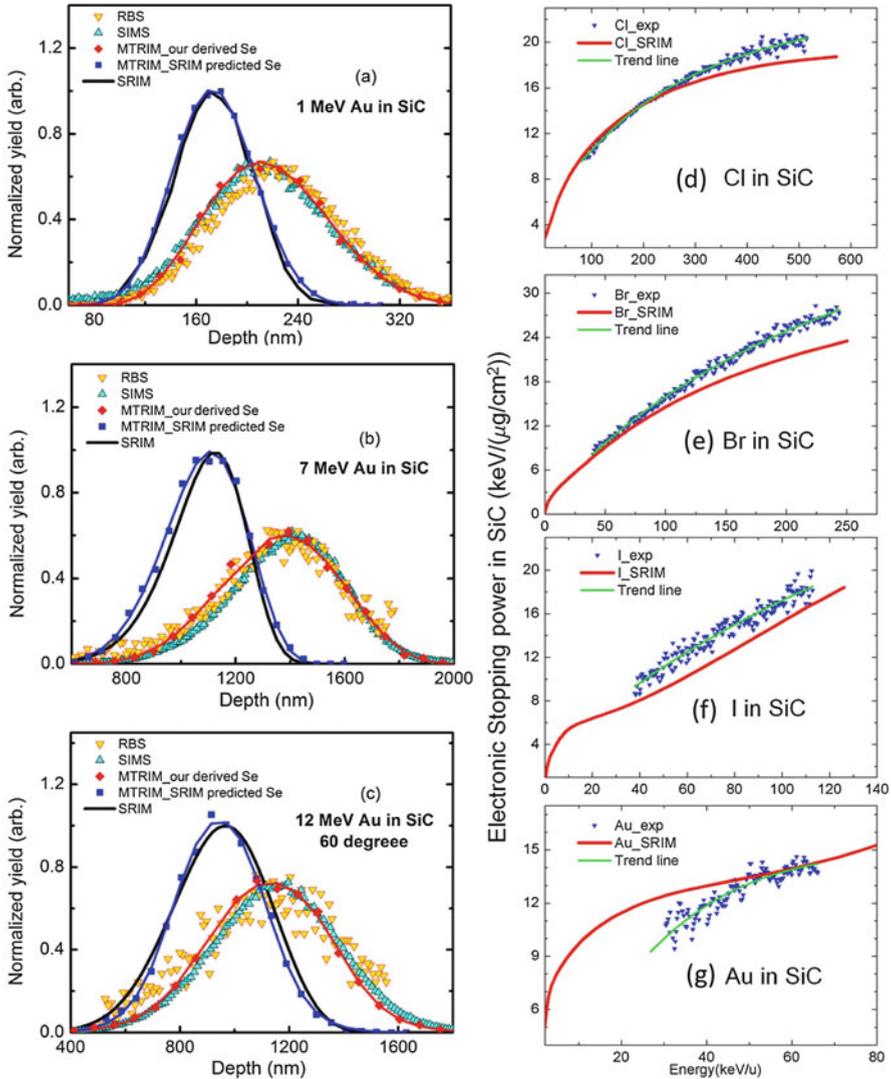


(Li et al. 2015), boosting the computational efficiency of complex 3D geometry simulations to a level similar to that in bulk simulations.

For a static BCA Monte Carlo simulation, the history of each ion is independent. Therefore, it is well suited for parallel computation. IM3D (Li et al. 2015) in particular is implemented with the Message Passing Interface (MPI) library. Its parallel scaling performance, as shown in Fig. 16, shows a greatly enhanced speed, up to  $10^5$  faster than SRIM.

### 3.5 Open Sourcing and Stopping Power Databases

An open source code can greatly facilitate the development of advanced codes, as it allows an open inspection and validation of the underlying algorithms. More and more codes have joined the open-source family, including Corteo (Schiettekatte 2009), *iradina* (Borschel and Ronning 2011), MyTRIM (Schwen 2018), Mat-TRIM (Yang et al. 2018b), and IM3D (Li et al. 2015), etc. One of the main advantages of open-source is that one can easily modify or incorporate new stopping power data into the code. It is known that the accuracy of BCA Monte Carlo simulation codes for ion radiation rely greatly on the accuracy of the stopping power data. Although SRIM has provided a powerful stopping power database (Ziegler 2004), it was recently shown to be inaccurate in several circumstances. For example, for the case of very heavy ions into a light element target, a large discrepancy is shown between the ion ranges predicted by SRIM and experimental results (Behar et al. 1985; Grande et al. 1988; Friedland et al. 1998; Zhang et al. 2009, 2010; Jin et al. 2014) due to errors in the stopping power database. An example of heavy ions into SiC is shown in Fig. 17. Because of this, M-TRIM (Jin et al. 2014) has been developed based on an early version of TRIM (Ziegler et al. 1985) with updated stopping power data for Cl, Br, I, and Au ions into  $\text{SiO}_2$  and SiC. The comparison



**Fig. 17** (a–c) Injected ion distribution of 1 MeV, 7 MeV, and 12 MeV Au into SiC: a comparison between the simulations and experimental results. (d–g) Comparison of the electronic stopping power of Cl, Br, I, and Au into SiC between the experimental results and SRIM database (Jin et al. 2014)

between the ion distribution simulated by SRIM and M-TRIM and the experimental results obtained by Rutherford backscattering spectrometry (RBS) and secondary ion mass spectrometry (SIMS) are shown in Fig. 17a.

Electronic stopping powers for bulk materials may also need to be corrected for nanostructured materials. Recently, the electronic energy loss for graphene (see

**Fig. 18** Comparison of transferred energy as a function of proton energy. The target is graphene and the collision site is point C (Zhao et al. 2015)

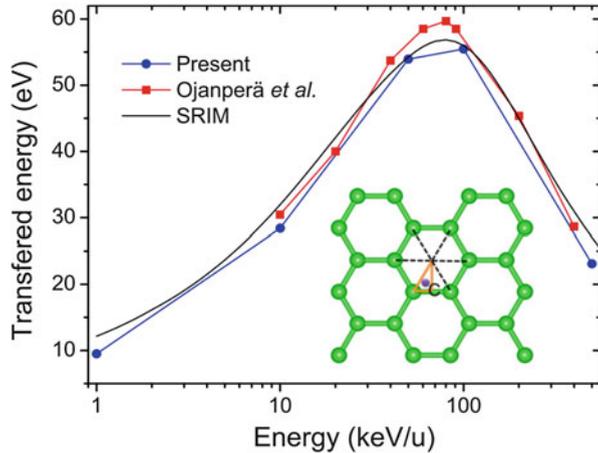


Fig. 18) and boron nitride (BN) sheets have been studied (Zhao et al. 2015) using time-dependent DFT. However, there is still a lack of study of the applicability of bulk stopping power database in nanostructured systems. Currently, Corteo, *iradina*, and IM3D use the same stopping power tables generated by Corteo using the “SRModule.exe” (Ziegler 2004). Since these codes are open-source, in principle it is possible to generate new stopping power tables. However, there is still a need of a friendly user-interface (UI) that can help generate and conveniently manage the stopping power data for these codes.

## 4 Comparisons Between MC and Molecular Dynamics (MD)

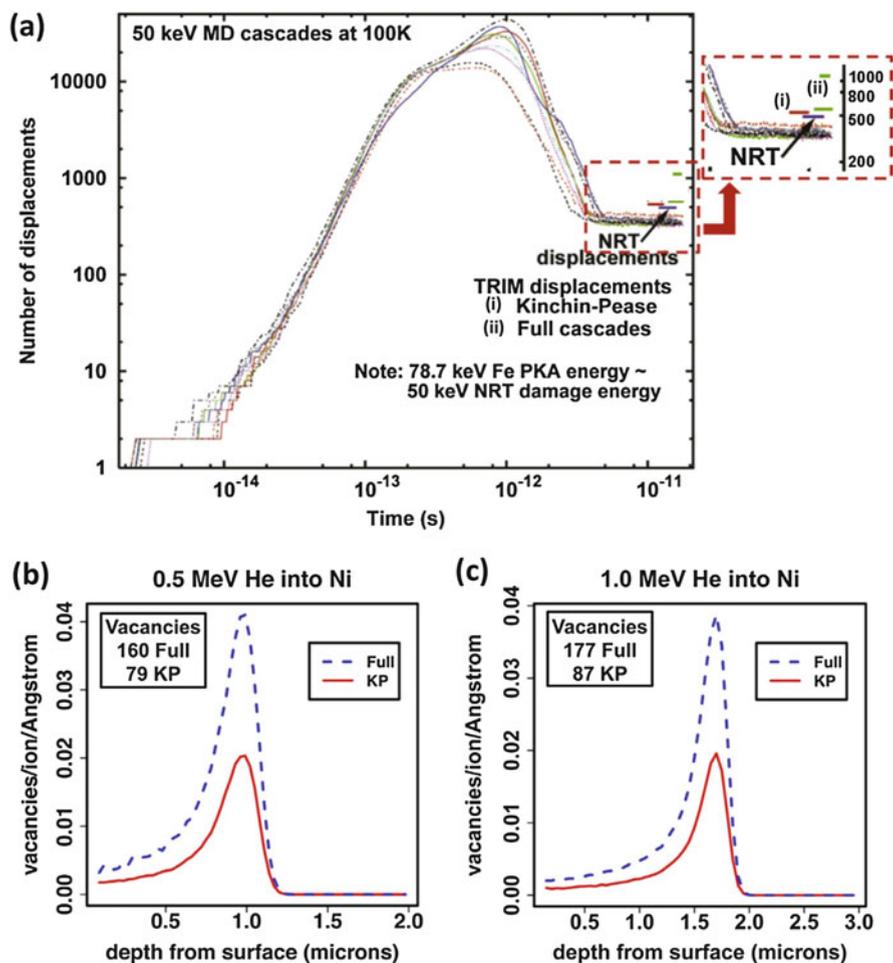
Radiation damage in nanostructured materials can also be simulated by MD (Jin et al. 2018). Here we will briefly review the difference between MC and MD simulations. MC codes are generally based on more assumptions than MD. MC codes usually only consider two-body interactions and use universal potentials in many cases. Because of its high computational efficiency, MC simulations of ion irradiation serve as convenient engineering tools for the prediction of ion ranges and vacancy/energy-deposition distributions. On the other hand, MD simulations are based on solving equations of motion (Li 2005), which is in general more accurate as they consider many-body interactions and can provide more details about the cascade process; however, MD is much more computationally intense than MC, limiting the time and length scales simulated. It should be noted that the accuracy of MD relies on the accuracy of the interatomic potential used. Also, it has recently been shown that the electronic energy loss should be taken into account in MD simulations, while the majority of MD studies on radiation damage cascades ignore this effect (Rutherford and Duffy 2007; Sand 2018). For nanostructured materials whose size is less than the mean-free-path of ions, it was recently shown that

electronic stopping is the dominant energy loss mechanism for incident ions (Yang et al. 2018b), indicating the necessity of considering electronic energy loss in MD simulations. The results of MD also depend on the simulation temperature, which is difficult to consider in BCA-MC codes. Almost no BCA-MC simulations consider defect diffusion and defect-sink interactions (kinetics at longer timescales), while MD simulations more systematically model kinetic processes (especially annealing) in materials if the time and length scales are within computable limits. For instance, previous MD simulations showed that defect production can be greatly enhanced by a large surface-area to volume ratio in nanostructured materials at low incident ion energies (Hoilijoki et al. 2011; Sun et al. 2012), which BCA-MC simulations usually fail to predict. However, the primary radiation damage estimated by MC simulations still has broad and significant engineering implications.

MC simulation is a useful and rapid method to calculate the Norgett–Robinson–Torres (NRT) displacements per atom (DPA), a standardized radiation exposure parameter. The definition of DPA is based on  $T = 0$  K MD simulations, using the following procedure: First, give a selective incident particle an initial energy  $E_0$ . The particle will collide with multiple atoms in the target until its energy is small enough. Every time when it collides an atom in the lattice, if the kinetic energy transferred to the target atom exceeds the threshold energy  $E_d$ , the atom will be knocked out (at  $t = t_1 \sim$  fs). After  $t_1$ , there is spontaneous recombination of the Frenkel pairs if the distance between a vacancy and an interstitial atom is within a certain limit. At  $t = t_F \sim$  ps, this athermal dynamic recombination/transient annealing process is finished. After that, long-term defect evolution occurs, which depends on thermal activation.

Among all MC codes, SRIM (Ziegler et al. 2008) is the most widely used for the estimation of DPA because of its powerful stopping power database and its ease of use. SRIM offers two ways for the calculation of NRT DPA: (1) The quick Kinchin–Pease (K-P) mode, where defect production by generation-1 ions are tallied by using the K-P formula instead of explicitly calculated and (2) the full-cascade (FC) mode, where generation-2, -3, -4, ... ions are also explicitly accounted for by BCA calculations. It has recently been found that the displacements calculated by both modes are significantly larger than the results of MD at ten picoseconds after a PKA when the thermal spike has largely disappeared (Stoller et al. 2013) (Fig. 19a). Also, the displacements by the FC mode are twice of that by K-P mode (Fig. 19a–c). Thus, Stoller et al. questioned the FC mode in SRIM, exposing a fundamental problem in this *closed-source* code. Recently, this discrepancy between K-P mode and FC modes has reproduced by IM3D (Li et al. 2015), a new open-source code.

It is also worthwhile to note that MC codes usually overestimate the number of radiation defects by three times compared to MD, while only capturing 1/30th of the atomic mixing (replacement) events (Nordlund et al. 2018), as shown in Fig. 20a, b. As a result, Nordlund et al. proposed two new complementary displacement production estimators to quantify the radiation damage more realistically: (a) Athermal recombination corrected dpa, namely, “arc-dpa”; and (b) replacements per atom, namely, “rpa.” Both quantities rely on parameters gained by

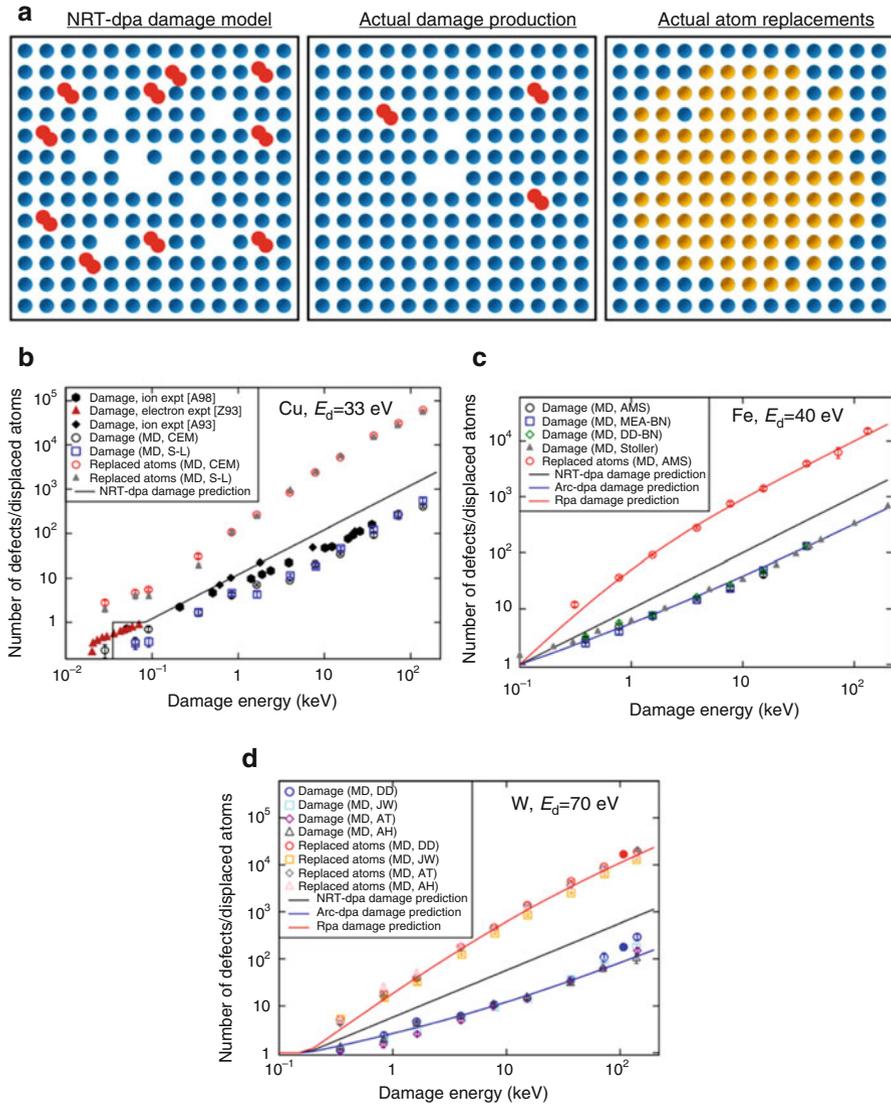


**Fig. 19** (a) Comparison between the displacements calculated by MD, SRIM (K-P mode), and SRIM (FC mode). (b, c) Depth distribution of vacancies for He<sup>+</sup> into Ni (Stoller et al. 2013)

fitting MD simulation results. Helpful discussions about the difference between MC and MD for radiation damage studies can also be found in “Molecular dynamics simulations of nonequilibrium systems” within this Handbook (Djurabekova and Nordlund 2018).

## 5 Outlook for the Next 10 Years

In summary, Monte Carlo simulation of ion radiation based on a BCA model is an efficient tool for the prediction of primary damage in materials. A series of



**Fig. 20** (a) Schematic illustration of the problem of the NRT-dpa damage model. (b) Comparison of MD simulation and NRT-dpa model. (c, d) Two new quantification standards: Arc-DPA and RPA agree well with MD simulations. Reproduced from Ref. (Nordlund et al. 2018), which is licensed under CC BY 4.0

advanced codes have been recently developed, enabling full-3D simulations, which can consider arbitrarily complex target morphologies, arbitrary beam shapes/sizes, and even dynamic evolution. The speed of these codes is also greatly enhanced by the fast indexing method of scattering angle sampling and highly parallel

computation. Also, more and more advanced codes are becoming open-source, allowing verification of the algorithms and the underlying physics engine. What is more, maintaining an open-source code policy will facilitate the incorporation of BCA-MC simulations into the framework of multi-scale and multi-physics frameworks such as MOOSE (Gaston et al. 2009), dynamically providing primary radiation damage data as input for larger time- or length-scale simulations. Besides, it has been shown that legacy codes such as SRIM will lead to significant errors in many applications, such as nano-beams, nanostructured materials, and rough-surface. However, the traditional closed-source code SRIM still enjoys a relative monopoly due to its long history, its relative accuracy for bulk-scale simulation, and especially its simple and intuitive user interface. It is important for new codes to have such a friendly user interface, allowing not only the easy input of complex target morphologies, but also convenient updates of the stopping power database. Last, but not least, increasing doubt exists on the NRT-DPA model when comparing MC to MD simulations, while the applicability of newly proposed damage units is still under exploration.

**Acknowledgments** The authors acknowledge support from the United States National Science Foundation Grant No. DMR-1120901, and support from the U.S. DOE Office of Nuclear Energy under Grant No. DE-NE0008827. M. P. S. specifically acknowledges support from the U.S. National Science Foundation's CAREER award program under Grant No. DMR-1654548. The authors thank Dr. Yong-Gang Li from the Institute of Solid State Physics of the Chinese Academy of Sciences and Prof. Benoit Forget from MIT for helpful discussions.

---

## References

- Agostinelli S, Allison J, Amako K et al (2003) GEANT4 – a simulation toolkit. *Nucl Instrum Methods Phys Res A* 506:250–303. [https://doi.org/10.1016/S0168-9002\(03\)01368-8](https://doi.org/10.1016/S0168-9002(03)01368-8)
- Baldwin MJ, Doerner RP (2010) Formation of helium induced nanostructure “fuzz” on various tungsten grades. *J Nucl Mater* 404:165–173. <https://doi.org/10.1016/j.jnucmat.2010.06.034>
- Bayn I, Chen EH, Trusheim ME et al (2015) Generation of ensembles of individually resolvable nitrogen vacancies using nanometer-scale apertures in ultrahigh-aspect ratio planar implantation masks. *Nano Lett* 15:1751–1758. <https://doi.org/10.1021/nl504441m>
- Behar M, Fichtner PF, Olivieri CA et al (1985) Range profiles of implanted Bi and Au in amorphous silicon. *Nucl Instrum Methods Phys Res B* 6:453–458. [https://doi.org/10.1016/0168-583X\(85\)90002-3](https://doi.org/10.1016/0168-583X(85)90002-3)
- Bielajew AF (2001) Fundamentals of the Monte Carlo method for neutral and charged particle transport. The University of Michigan
- Biersack JP, Haggmark LG (1980) A Monte Carlo computer program for the transport of energetic ions in amorphous targets. *Nucl Inst Methods* 174:257–269. [https://doi.org/10.1016/0029-554X\(80\)90440-1](https://doi.org/10.1016/0029-554X(80)90440-1)
- Borschel C, Ronning C (2011) Ion beam irradiation of nanostructures – a 3D Monte Carlo simulation code. *Nucl Instrum Methods Phys Res B* 269:2133–2138. <https://doi.org/10.1016/j.nimb.2011.07.004>
- Borschel C, Spindler S, Lerose D et al (2011) Permanent bending and alignment of ZnO nanowires. *Nanotechnology* 22:185307. <https://doi.org/10.1088/0957-4484/22/18/185307>
- Briesmeister JF (1986) MCNP – a general Monte Carlo code for neutron and photon transport. Los Alamos National Laboratory, Los Alamos

- Chen Y, Huang S, Ji X et al (2018) Tuning electronic structure of single layer MoS<sub>2</sub> through defect and interface engineering. *ACS Nano* 12(3):2569–2579. <https://doi.org/10.1021/acsnano.7b08418>
- Crocobette JP (2005) Cascade Modeling. In: Yip S. (eds) *Handbook of Materials Modeling*. Springer, Dordrecht [https://doi.org/10.1007/978-1-4020-3286-8\\_49](https://doi.org/10.1007/978-1-4020-3286-8_49)
- Djurabekova F, Nordlund K (2018) Molecular dynamics simulations of non-equilibrium systems. In: Andreoni W, Yip S. (eds) *Handbook of materials modeling*. Springer, Cham [https://doi.org/10.1007/978-3-319-50257-1\\_119-1](https://doi.org/10.1007/978-3-319-50257-1_119-1)
- Eckstein W (1991) *Computer simulation of ion-solid interactions*. Springer, Berlin/Heidelberg
- Economou NP, Notte JA, Thompson WB (2012) The history and development of the helium ion microscope. *Scanning* 34:83–89. <https://doi.org/10.1002/sca.20239>
- Friedland E, Kalbitzer S, Hayes M et al (1998) Range parameters of gold and lead in carbon and carbon in gold at reduced energies of  $10^{-3} < \epsilon < 1$ . *Nucl Instrum Methods Phys Res B* 136–138:147–152. [https://doi.org/10.1016/S0168-583X\(97\)00833-1](https://doi.org/10.1016/S0168-583X(97)00833-1)
- Gaston D, Newman C, Hansen G, Lebrun-Grandié D (2009) MOOSE: A parallel computational framework for coupled systems of nonlinear equations. *Nucl Eng Des* 239:1768–1778. <https://doi.org/10.1016/j.nucengdes.2009.05.021>
- Geuzaine C, Remacle J-F (2009) Gmsh: a 3-D finite element mesh generator with built-in pre- and post-processing facilities. *Int J Numer Methods Eng* 79:1309–1331. <https://doi.org/10.1002/nme.2579>
- Gilmer G, Yip S (2005) Basic Monte Carlo models: equilibrium and kinetics. In: Yip S (ed) *Handbook of materials modeling*. Springer, Dordrecht, pp 613–628
- Gómez-Navarro C, de Pablo PJ, Gómez-Herrero J et al (2005) Tuning the conductance of single-walled carbon nanotubes by ion irradiation in the Anderson localization regime. *Nat Mater* 4:534–539. <https://doi.org/10.1038/nmat1414>
- Grande PL, Fichtner PFP, Behar M, Zawislak FC (1988) Range profiles of medium and heavy ions implanted into SiO<sub>2</sub>. *Nucl Inst Methods Phys Res B* 35:17–20. [https://doi.org/10.1016/0168-583X\(88\)90093-6](https://doi.org/10.1016/0168-583X(88)90093-6)
- Hoffmann S, Bauer J, Ronning C et al (2009) Axial p-n junctions realized in silicon nanowires by ion implantation. *Nano Lett* 9:1341–1344. <https://doi.org/10.1021/nl802977m>
- Hoilijoki S, Holmström E, Nordlund K (2011) Enhancement of irradiation-induced defect production in Si nanowires. *J Appl Phys* 110:43540. <https://doi.org/10.1063/1.3627234>
- Holland-Moritz H, Scheeler S, Stanglmair C et al (2015) Enhanced sputter yields of ion irradiated Au nano particles: energy and size dependence. *Nanotechnology* 26:325301. <https://doi.org/10.1088/0957-4484/26/32/325301>
- Hu A, Hassanein A (2012) How surface roughness affects the angular dependence of the sputtering yield. *Nucl Instrum Methods Phys Res B* 281:15–20. <https://doi.org/10.1016/j.nimb.2012.03.026>
- Jin K, Zhang Y, Zhu Z et al (2014) Electronic stopping powers for heavy ions in SiC and SiO<sub>2</sub>. *J Appl Phys* 115:044903. <https://doi.org/10.1063/1.4861642>
- Jin M, Cao P, Yip S, Short MP (2018) Radiation damage reduction by grain-boundary biased defect migration in nanocrystalline Cu. *Acta Mater* 155:410–417. <https://doi.org/10.1016/j.actamat.2018.05.071>
- Kenmotsu T, Yamamura Y, Muramoto T, Hirotsu N (2005) Simulation studies on sputtering in rough surface. *Nucl Instrum Methods Phys Res B* 228:369–372. <https://doi.org/10.1016/j.nimb.2004.10.072>
- Krashennnikov AV, Nordlund K (2010) Ion and electron irradiation-induced effects in nanostructured materials. *J Appl Phys* 107:71301. <https://doi.org/10.1063/1.3318261>
- Krashennnikov AV, Banhart F, Li JX et al (2005) Stability of carbon nanotubes under electron irradiation: role of tube diameter and chirality. *Phys Rev B Condens Matter Mater Phys* 72:125428. <https://doi.org/10.1103/PhysRevB.72.125428>
- LANL (2018) A general Monte Carlo N-Particle (MCNP) transport code. <https://laws.lanl.gov/vhosts/mcnp.lanl.gov/index.shtml>. Accessed 11 Oct 2018

- Li J (2005) Basic molecular dynamics. In: Yip S (ed) Handbook of materials modeling. Springer, Dordrecht, pp 565–588
- Li HM, Ding ZJ (2006) Monte Carlo simulation of secondary electron and backscattered electron images in scanning electron microscopy for specimen with complex geometric structure. *Scanning* 27:254–267. <https://doi.org/10.1002/sca.4950270506>
- Li YG, Mao SF, Li HM, et al (2008) Monte Carlo simulation study of scanning electron microscopy images of rough surfaces. *J Appl Phys* 104:064901. <https://doi.org/10.1063/1.2977745>
- Li YG, Yang Y, Short MP et al (2015) IM3D: a parallel Monte Carlo code for efficient simulations of primary radiation displacements and damage in 3D geometry. *Sci Rep* 5:1–13. <https://doi.org/10.1038/srep18130>
- Li Y, Yang Y, Short MP et al (2017) Ion radiation albedo effect: influence of surface roughness on ion implantation and sputtering of materials. *Nucl Fusion* 57:16038. <https://doi.org/10.1088/1741-4326/57/1/016038>
- Liontas R, Gu XW, Fu E et al (2014) Effects of helium implantation on the tensile properties and microstructure of Ni73P27 metallic glass nanostructures. *Nano Lett* 14:5176–5183. <https://doi.org/10.1021/nl502074d>
- Maletinsky P, Hong S, Grinolds MS et al (2012) A robust scanning diamond sensor for nanoscale imaging with single nitrogen-vacancy centres. *Nat Nanotechnol* 7:320–324. <https://doi.org/10.1038/nnano.2012.50>
- MIT (2011) OpenMC: theory and methodology. <https://openmc.readthedocs.io/en/stable/methods/index.html>. Accessed 11 Oct 2018
- Möller W (2014) TRI3DYN – collisional computer simulation of the dynamic evolution of 3-dimensional nanostructures under ion irradiation. *Nucl Instrum Methods Phys Res B* 322:23–33. <https://doi.org/10.1016/j.nimb.2013.12.027>
- Möller W, Eckstein W (1984) Tridyn – a TRIM simulation code including dynamic composition changes. *Nucl Inst Methods Phys Res B* 2:814–818. [https://doi.org/10.1016/0168-583X\(84\)90321-5](https://doi.org/10.1016/0168-583X(84)90321-5)
- Nordlund K, Björkas C, Ahlgren T et al (2014) Multiscale modelling of plasma-wall interactions in fusion reactor conditions. *J Phys D Appl Phys* 47:224018. <https://doi.org/10.1088/0022-3727/47/22/224018>
- Nordlund K, Zinkle SJ, Sand AE et al (2018) Improving atomic displacement and replacement calculations with physically realistic damage models. *Nat Commun* 9:1084. <https://doi.org/10.1038/s41467-018-03415-5>
- Nordlund K, Short MP (2018) Modelling of radiation damage in materials: best practices and future directions. In: Andreoni W, Yip S (eds) Handbook of materials modeling. Volume 2 Applications: Current and Emerging Materials. Springer
- Odette GR, Wirth BD (2005) Radiation effects in fission and fusion reactors. In: Yip S (ed) Handbook of materials modeling. Springer, Dordrecht, pp 999–1037
- Pekin TC, Allen FI, Minor AM (2016) Evaluation of neon focused ion beam milling for TEM sample preparation. *J Microsc* 264:59–63. <https://doi.org/10.1111/jmi.12416>
- Pezzagna S, Wildanger D, Mazarov P et al (2010) Nanoscale engineering and optical addressing of single spins in diamond. *Small* 6:2117–2121. <https://doi.org/10.1002/sml.201000902>
- Pezzagna S, Rogalla D, Becker HW et al (2011) Creation of colour centres in diamond by collimated ion-implantation through nano-channels in mica. *Phys Status Solidi A* 208:2017–2022. <https://doi.org/10.1002/pssa.201100455>
- Romano PK, Forget B (2013) The OpenMC monte carlo particle transport code. *Annals of Nuclear Energy*, 51:274–81
- Rutherford AM, Duffy DM (2007) The effect of electron-ion interactions on radiation damage simulations. *J Phys Condens Matter* 19:496201. <https://doi.org/10.1088/0953-8984/19/49/496201>
- Ruzic DN (1990) The effects of surface roughness characterized by fractal geometry on sputtering. *Nucl Inst Methods Phys Res B* 47:118–125. [https://doi.org/10.1016/0168-583X\(90\)90019-Q](https://doi.org/10.1016/0168-583X(90)90019-Q)
- Sand AE (2018) Incorporating electronic effects in molecular dynamics simulations of neutron and ion-induced collision cascades. In: Andreoni W., Yip S. (eds) Handbook of Materials Modeling. Springer, Cham. [https://doi.org/10.1007/978-3-319-50257-1\\_135-1](https://doi.org/10.1007/978-3-319-50257-1_135-1)

- Scarabelli D, Trusheim M, Gaathon O et al (2016) Nanoscale engineering of closely-spaced electronic spins in diamond. *Nano Lett* 16:4982–4990. <https://doi.org/10.1021/acs.nanolett.6b01692>
- Schiettekatte F (2008) Fast Monte Carlo for ion beam analysis simulations. *Nucl Instrum Methods Phys Res B* 266:1880–1885. <https://doi.org/10.1016/j.nimb.2007.11.075>
- Schiettekatte F (2009) Monte Carlo simulation of ion beam analysis spectra using Corteo. <https://www.lps.umontreal.ca/~schiette/index.php?n=Recherche.Corteo>
- Schiettekatte F, Chicoine M (2016) Spectrum simulation of rough and nanostructured targets from their 2D and 3D image by Monte Carlo methods. *Nucl Instrum Methods Phys Res B* 371:106–110. <https://doi.org/10.1016/j.nimb.2015.09.089>
- Schwen D (2018) MyTRIM. <https://github.com/idaholab/mytrim>. Accessed 11 Oct 2018
- Schwen D, Huang M, Bellon P, Averback RS (2009) Molecular dynamics simulation of intragranular Xe bubble re-resolution in UO<sub>2</sub>. *J Nucl Mater* 392:35–39. <https://doi.org/10.1016/j.jnucmat.2009.03.037>
- Shinada T, Okamoto S, Kobayashi T, Ohdomari I (2005) Enhancing semiconductor device performance using ordered dopant arrays. *Nature* 437:1128–1131. <https://doi.org/10.1038/nature04086>
- Short MP, Gaston DR, Jin M et al (2015) Modeling injected interstitial effects on void swelling in self-ion irradiation experiments. *J Nucl Mater* 471:200–207. <https://doi.org/10.1016/j.jnucmat.2015.10.002>
- Stoller RE, Toloczko MB, Was GS et al (2013) On the use of SRIM for computing radiation damage exposure. *Nucl Instrum Methods Phys Res B* 310:75–80. <https://doi.org/10.1016/j.nimb.2013.05.008>
- Sun L, Lan C, Zhao S et al (2012) Self-irradiation of thin SiC nanowires with low-energy ions: a molecular dynamics study. *J Phys D Appl Phys* 45:135403. <https://doi.org/10.1088/0022-3727/45/13/135403>
- Tan S, Livengood R, Hack P et al (2011) Nanomachining with a focused neon beam: a preliminary investigation for semiconductor circuit editing and failure analysis. *J Vac Sci Technol B* 29:06F604. <https://doi.org/10.1116/1.3660797>
- Toyli DM, Weis CD, Fuchs GD et al (2010) Chip-scale nanofabrication of single spins and spin arrays in diamond. *Nano Lett* 10:3168–3172. <https://doi.org/10.1021/nl102066g>
- Wang W, Roth J, Lindig S, Wu CH (2001) Blister formation of tungsten due to ion bombardment. *J Nucl Mater* 299:124–131. [https://doi.org/10.1016/S0022-3115\(01\)00679-1](https://doi.org/10.1016/S0022-3115(01)00679-1)
- Watt F, Breese MBH, Bettiol AA, van Kan JA (2007) Proton beam writing. *Mater Today* 10:20–29
- Woo CH (2005) Modeling irradiation damage accumulation in crystals. In: Yip S (ed) *Handbook of Materials Modeling*. Springer, Dordrecht, pp 959–986
- Xu D, Wirth BD, Li M, Kirk MA (2012) Combining in situ transmission electron microscopy irradiation experiments with cluster dynamics modeling to study nanoscale defect agglomeration in structural metals. *Acta Mater* 60:4286–4302. <https://doi.org/10.1016/j.actamat.2012.03.055>
- Yang Y, Li YG, Short MP et al (2018a) Nano-beam and nano-target effects in ion radiation. *Nanoscale* 10(4):1598–1606. <https://doi.org/10.1039/C7NR08116B>
- Yang Y, Short MP, Li J (2018b) Monte Carlo simulation of PKA distribution along nanowires under ion radiation. *Nucl Eng Des* 340:300–307. <https://doi.org/10.1016/j.nucengdes.2018.09.001>. Accessed 11 Oct 2018
- Yao Y, van Mourik MW, Santhana Raman P, van Kan JA (2013) Improved beam spot measurements in the 2nd generation proton beam writing system. *Nucl Instrum Methods Phys Res B* 306:265–270. <https://doi.org/10.1016/j.nimb.2012.10.033>
- Yip S (2014) *Nuclear radiation interactions*. World Scientific Publishing, Singapore
- Yuan B, Yu PC, Tang SM (1993) A database method for binary atomic scattering angle calculation. *Nucl Instrum Methods Phys Res B* 83:413–418. [https://doi.org/10.1016/0168-583X\(93\)95864-2](https://doi.org/10.1016/0168-583X(93)95864-2)
- Zhang Y, Bae IT, Sun K et al (2009) Damage profile and ion distribution of slow heavy ions in compounds. *J Appl Phys* 105:104901. <https://doi.org/10.1063/1.3118582>

- Zhang Y, Ishimaru M, Jagielski J et al (2010) Damage and microstructure evolution in GaN under Au ion irradiation. *J Phys D Appl Phys* 43:85303. <https://doi.org/10.1088/0022-3727/43/8/085303>
- Zhang P, Wang HY, Li YG et al (2012) Monte Carlo simulation of secondary electron images for real sample structures in scanning electron microscopy. *Scanning* 34:145–150. <https://doi.org/10.1002/sca.20288>
- Zhao S, Kang W, Xue J et al (2015) Comparison of electronic energy loss in graphene and BN sheet by means of time-dependent density functional theory. *J Phys Condens Matter* 27:025401. <https://doi.org/10.1088/0953-8984/27/2/025401>
- Ziegler JF (2004) Srim-2003. *Nucl Instrum Methods Phys Res B* 219–220:1027–1036. <https://doi.org/10.1016/j.nimb.2004.01.208>
- Ziegler JF, Biersack JP, Littmark U (1985) *The stopping and range of ions in matter*, vol 1. Pergamon Press, New York
- Ziegler JF, Biersack JP, Ziegler MD (2008) *SRIM: the stopping and range of ions in matter*. Cadence Design Systems. ISBN-10: 096542071X; ISBN-13: 978-0965420716
- Ziegler JF, Biersack JP, Ziegler MD (2010) *SRIM - The stopping and range of ions in matter*. *Nucl Instruments Methods Phys Res Sect B Beam Interact with Mater Atoms* 268:1818–1823. <https://doi.org/10.1016/j.nimb.2010.02.091>