



MUNDFAB

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 871813.

ICT Project No 871813

MUNDFAB

Modeling Unconventional Nanoscaled
Device FABrication

D6.2: Report on the integration of external KMC/LKMC tools into the TCAD toolchain

C. Rossi (Fraunhofer), E. Bär (Fraunhofer),
P. Pichler (Fraunhofer), G. Calogero (CNR),
A. La Magna (CNR), G. Fisicaro (CNR),
I. Deretzis (CNR), R. Helleboid (ST), D. Rideau (ST),
A. Hemeryck (CNRS), D. Waldhör (TUW),
T. Grasser (TUW)

29.09.2022



Quality management

This deliverable was reviewed by:

Benoit Sklénard, CEA-Leti

Guilhem Larrieu, LAAS-CNRS

Simon Mundinar, Fraunhofer IISB

Contents

Abstract.....	3
1 Introduction	3
2 Coupling of <code>MuLSKIPS</code> and Synopsys TCAD tools.....	4
2.1 Coupling of <code>MuLSKIPS</code> and MSH files	6
2.2 Simulation of topography evolution based on KMC surface node positions	9
3 Coupling of oxidation simulations of WP5 and Synopsys tools	11
4 Conclusions.....	13
References.....	13

Abstract

This deliverable describes the procedure that was developed to achieve the integration of the simulations done using external Kinetic Monte Carlo (KMC)/Lattice Kinetic Monte-Carlo (LKMC) tools into the Synopsys TCAD simulation workflow. The external KMC/LKMC tools include the atomistic models developed in WPs 2 to 5 and need to be interfaced with Synopsys tools to obtain a single simulation toolchain. Results of WPs 2 to 4 are implemented in the `MuLSKIPS` LKMC software package. The structure to be simulated is stored in TDR format within the Synopsys environment. The structure is converted in MSH format, thanks to a series of conversion steps, to be imported in `MuLSKIPS`. After the KMC simulation, a file containing the positions of the surface nodes is given as output from `MuLSKIPS`. The tool DEP3D is then used to construct a triangle-based description of the new topography entity based on the surface nodes provided by `MuLSKIPS`. As last step, the new entity is incorporated into the initial TDR structure, obtaining a final structure resulting from KMC simulations in `MuLSKIPS` that can be further treated within the Synopsys environment.

The defects distribution in the oxide calculated by KMC simulations in WP5, combined with the main results from DFT and molecular dynamics simulations of WP5, namely process induced defects parameters, can be treated without the need of an interface between solvers. Indeed, the defect density statistics in the oxide and at the interface (as a function of process parameters), can be provided as direct inputs to Synopsys Sentaurus Device.

The final workflow will thus include a mix of simulation approaches that expands the capability of standard Sentaurus TCAD flow. The relevance of this methodology and the impact of atomistic simulation steps will be assessed by comparing the simulations output for the electrical characteristics of the device under test, in Deliverable D6.4.

1 Introduction

This document presents the work done to interface external KMC/LKMC tools, which integrate models of WPs 2 to 5, and Synopsys TCAD tools. The goal is to obtain a final simulation toolchain which integrates the modeling results of the MUNDFAB project and the capabilities of Synopsys Sentaurus Process and Device. The toolchain will be later applied to the test applications presented in Deliverable D6.1 and the results will be presented in Deliverables D6.3 and D6.4.

Some of the novel modeling advancements, developed in WPs 2 to 4, cannot be implemented in the LKMC module available within Sentaurus Process of Synopsys. As an alternative, they will be implemented in `MuLSKIPS`, an open-source software package developed by CNR-IMM, as described in Deliverables D2.3, D3.4 and D4.4. Section 2 of this document presents the coupling procedure of `MuLSKIPS` with Synopsys tools. The coupling demonstration reported in this document is based on an example structure extracted from the Vertical Gate-All-Around Nanowire Field Effect Transistor (GAA-NW-FET) demonstrator, described in Deliverable D6.1. The structure consists of a silicon vertical nanowire on a silicon substrate, as shown in Fig. 1b. The nanowire has a diameter of 12 nm and a height of 45 nm, the substrate has a width and length of 31.5 nm. This nanowire has been obtained after the oxidation (wet oxidation at 850 °C for 5 minutes) of the structure in Fig. 1a, followed by the oxide stripping. The oxidation step was simulated in Sentaurus Process, using the continuum solver.

The results regarding oxidation simulations from WP5 will be integrated using a different procedure, which is described in Section 3 of this deliverable. Simulations of oxide layer growth

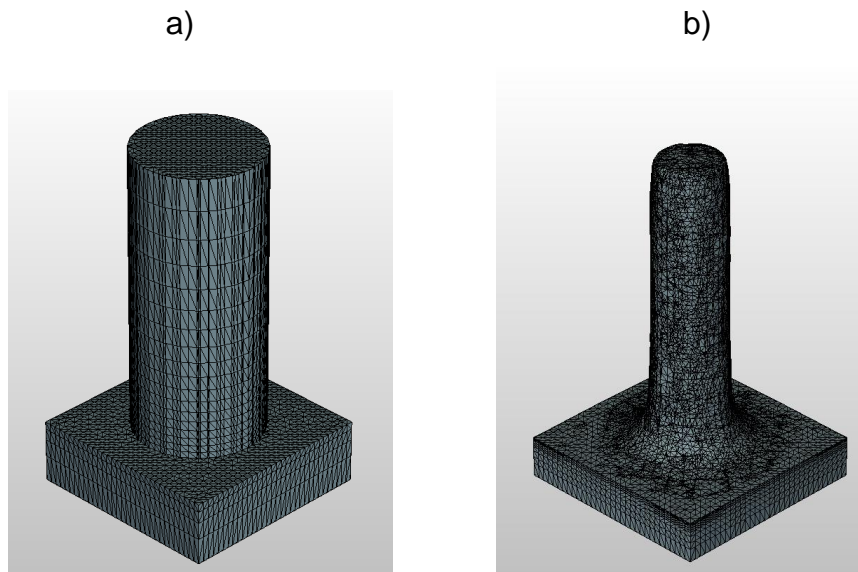


Figure 1: Structures simulated with Synopsys tools: a) silicon cylinder; b) silicon nanowire obtained after oxidation of silicon cylinder. Simulation of oxidation was performed in Sentaurus Process. Structure b) is the structure used as input by MuLSKIPS. The coupling procedure is demonstrated for the structure b).

are performed with a KMC tool at CNRS-LAAS. Then, the obtained structures are analyzed with the Machine Learning (ML) analysis tool developed by TUW and the defect formation mechanisms are inserted back into the KMC tool. The result is the derivation of defects statistics in the oxide and at the interfaces as a function of process parameters (e.g., oxidant pressure, temperature) that are provided as input to the Synopsys tools.

2 Coupling of MuLSKIPS and Synopsys TCAD tools

The coupling procedure starts with an initial structure created with Sentaurus Process. The Synopsys tools employ a custom file format for device geometry, mesh, and physical data (doping concentration, defects, electric field, ...), common to all tools [1], called TDR and with extension “.tdr”. MuLSKIPS cannot read TDR files. However, a coupling procedure to enable MuLSKIPS simulation starting from another widely used mesh file format (MSH format) was developed within the MUNDFAB project and is reported in Section 2.1. MSH is a popular file format for 3D meshes (with extension “.msh”) and it is the custom format used by the open source 3D finite element mesh generator software `gmsh` [2-3]. Like in the TDR format, different geometrical regions (e.g., silicon region, gas region, etc.) can be defined and physical data can be embedded. A conversion strategy from TDR to MSH file is thus needed and was developed in the project. It works as follows:

1. Using the Sentaurus Data Explorer (a Synopsys tool), the TDR file is converted to two ASCII files describing the same structure: a GRD file (extension “.grd”) that contains the mesh and a DAT file (extension “.dat”) that stores the dataset (physical data field).
2. The GRD mesh file is converted to a MSH mesh with an open-source Python converter that was already available on GitHub, a hosting service for software development [4].

3. An additional script was developed to include additional features (indication of physical entities, i.e., silicon, gas, etc., in addition to geometrical entities) needed in the MSH file that were not considered by the open-source converter.
4. The dataset of physical data of the structure stored in TDR is extracted as a function of position. Using Sentaurus Visual (a Synopsys tool), a table of coordinates of all mesh vertices and corresponding data fields is obtained.
5. The dataset is then included in the MSH file with a Python script.

The result of this conversion is an MSH file describing the initial structure, obtained in Synopsys Sentaurus Process. KMC simulations are then performed in `MuLSKIPS` using this MSH file as input. This will be presented in more detail in Section 2.1. After the simulation, the structure must be converted back into the TDR format for further processing in the Synopsys environment. A different approach is used for the “back” conversion due to the possibility of microstructural changes during the simulated process. It works as follows:

1. `MuLSKIPS` gives as output a file containing the positions of the new surface nodes, obtained after the KMC process simulation.
2. This file is used as input by the deposition simulator DEP3D [5], developed by Fraunhofer IISB. Thanks to the information about the new surface nodes and the initial structure file, the final structure is obtained using DEP3D combined with Synopsys tools, in TDR format (and so ready to be further used within the Synopsys framework). This procedure is explained in Section 2.2.
3. If the KMC simulation changes the physical data fields in addition to the geometry, a further step is necessary. A file containing information about the new physical data field, as a function of mesh vertices locations, is extracted from `MuLSKIPS` simulation results. The ASCII file that stores the physical dataset (DAT file) of the final structure is modified accordingly, to correctly describe the new physical data fields. More details will be provided in Section 2.2

The complete coupling procedure between `MuLSKIPS` and Synopsys environments is presented in Fig. 2, and it is described in detail in Sections 2.1 and 2.2. The coupling is

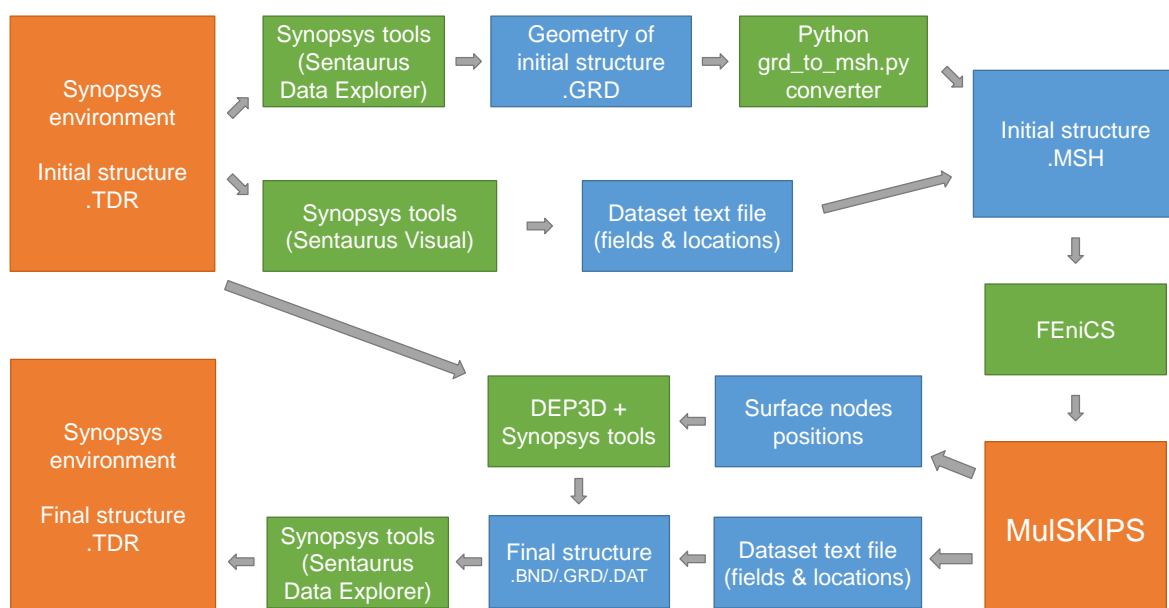


Figure 2: Sequence for coupling `MuLSKIPS` and Synopsys environments.

demonstrated in this deliverable by applying the procedure to a test case: a chemical vapor deposition (CVD) process on a vertical silicon nanowire, using the initial structure as described above. The initial structure is depicted in Fig. 3a, whereas the final structure is visualized in Figs. 6 and 7.

2.1 Coupling of `MuLSKIPS` and MSH files

The procedure to enable the use of MSH mesh files in `MuLSKIPS` consists of a sequence of conversion steps involving three main modelling environments, namely, `gmsh` [2-3], `FEniCS/Dolfin` [6-7] and `MuLSKIPS` [8-9], and two Python libraries, `meshio` [10-11] and `cashocs` [12-13]. The conversions are carried out using a Python script based on four ad-hoc conversion routines implemented in `pyMuLSKIPS`, i.e., the Python interface of `MuLSKIPS` (whose core code is implemented in Fortran). Such routines are named `msh2dolfin`, `dolfin2mulskips`, `mulskips2dolfin` and `dolfin2msh`. In particular, the `msh2dolfin` and `dolfin2msh` routines rely on *ad-hoc* `cashocs` and `meshio` functions able to read/write the native file format of `gmsh` (MSH) and convert it to a mesh object to be handled in `Dolfin`, i.e., the Python interface of `FEniCS`.

In Fig. 3, the full MSH to `MuLSKIPS` coupling procedure is illustrated for the test case considered in this deliverable, i.e., a chemical vapor deposition (CVD) process on a silicon nanowire (SiNW) on top of a silicon substrate. The initial structure is shown in Fig. 3a. The MSH file contains a mesh of 104462 nodes and 642587 mesh elements, overall partitioned into 620102 tetrahedral cells defining two 3D physical entities (gas and silicon) and 22485 triangular elements defining one 2D physical entity (the interface between gas and silicon). Each tetrahedral cell in the mesh is uniquely tagged as either silicon or gas.

The MSH file is subsequently imported in `FEniCS` using the `msh2dolfin` routine of `pyMuLSKIPS`. Specifically, this is done in two subsequent calls to `cashocs` routines: First the `cashocs-convert` command-line tool is executed to convert the MSH file into an intermediate set of XDMF/HDF5 files containing the mesh information in a format compatible with `Dolfin`; these are then readily imported in the Python environment using an *ad-hoc* `import_mesh` `cashocs` routine. We note that at this stage the SiNW direction lies in the positive z direction and the Si substrate lies in the xy plane. This is a requirement for the subsequent coupling with `MuLSKIPS`, which is based on a 3D superlattice with periodicity along the x and y directions, however it will not affect the orientation of the system at the end of the integration procedure. As a check, within `Dolfin`, a node function identifying silicon and gas nodes in the mesh can be written in a VTU file format and inspected using the `Paraview` software [14]. The result is shown in Fig. 3b with silicon nodes shown in red and gas nodes in blue.

Once the mesh is imported in the `Dolfin` framework, the `pyMuLSKIPS` `dolfin2mulskips` routine is used to setup an input superlattice KMC 3D box for `MuLSKIPS` having the same size of the mesh and generate a correspondence map between the sites in the KMC superlattice and the mesh tetrahedral cells indices. The latter is then used to define the occupancy of all KMC particles in the input structure provided to `MuLSKIPS` (KMC sites corresponding to silicon tetrahedral cells will be filled, whereas KMC sites corresponding to gas will be empty). Such occupancy information can either be stored to the disk by `pyMuLSKIPS` and subsequently read by the `MuLSKIPS` Fortran code (which is convenient for debugging), or it can be directly shared via RAM between `Dolfin` and `MuLSKIPS` using `f2py` sockets (see D4.5). We note that such capability of importing a mesh from an MSH file format and using it to setup the initial geometry

for `MuLSKIPS`, has been (and is currently) exploited in the context of other work packages in MUNDFAB (for example, refer to Deliverables D3.4 or D4.4).

A well calibrated `MuLSKIPS` KMC simulation of Si CVD growth with SiH_2Cl_2 , HCl and H_2 precursors at 1253 K is then carried out, starting from the generated input geometry. All details on the specific CVD model implementation and calibration will be provided elsewhere in the context of MUNDFAB WP3 (Deliverable D3.6). Here we show the surface silicon atoms within the KMC box at the beginning of the `MuLSKIPS` simulation (Fig. 3c) and at the end of the CVD growth (Fig. 3d), where H and Cl atoms are also shown. In Fig. 4 we report a close view of the silicon nanowire top region after the `MuLSKIPS` CVD simulation, where both crystal and coverage species are visible (H and Cl atoms are shown in white and light gray, respectively), and expected faceting along various crystallographic directions can also be noticed.

The output file containing the coordinates of surface nodes for the next step of the coupling procedure with DEP3D is extracted after the `MuLSKIPS` simulation (Fig. 5). The output file is a text file with three columns containing the $\{xyz\}$ coordinates of all the silicon surface atoms and all the coverage atoms, i.e., H and Cl: the surface atoms and the coverage atoms determine the solid-gas interface and so the new surface of the structure after the KMC simulation.

Even though it is not needed for the coupling procedure developed in MUNDFAB, it is also possible to convert the structure in Fig. 3d back to MSH format. In this way, a full coupling

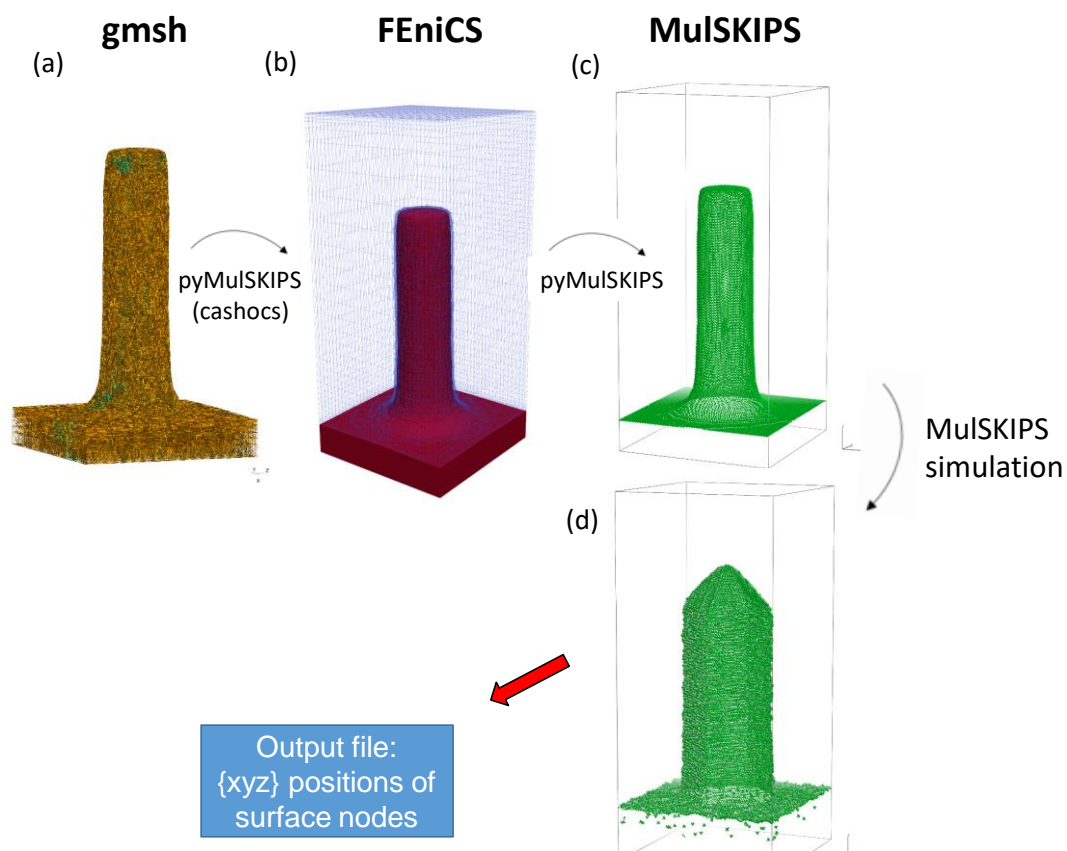


Figure 3: Schematic illustration of MSH - `MuLSKIPS` integration procedure. The relevant conversion tools used to interface the four different software environments are indicated under each arrow. Panels (a) with the `gmsh` viewer; (b) with `Paraview` [14]; (c) and (d) with `v_sim` [15].

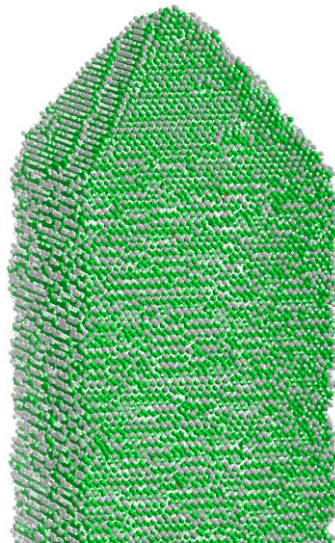


Figure 4: Close view of the silicon nanowire top region after the CVD growth, highlighting the presence of coverage species at the surface and facets formation along various crystallographic directions.

procedure between MSH files and `MuLSKIPS`, is obtained and can be applied to interface `MuLSKIPS` to other simulators that can work with MSH files. The back conversion works as follows. The `pyMuLSKIPS mulskips2dofin` routine takes care of post-processing the results of the `MuLSKIPS` simulation. It reads the updated set of occupied KMC sites and uses it, in conjunction with the aforementioned correspondence map between KMC superlattice and mesh tetrahedral cells, to generate a new `Dofin` mesh function which labels every mesh cell with an integer (e.g., 1 and 2) depending on whether it corresponds to empty (gas) or occupied sites after the `MuLSKIPS` CVD process simulation. Finally, the `pyMuLSKIPS dofing2msh`

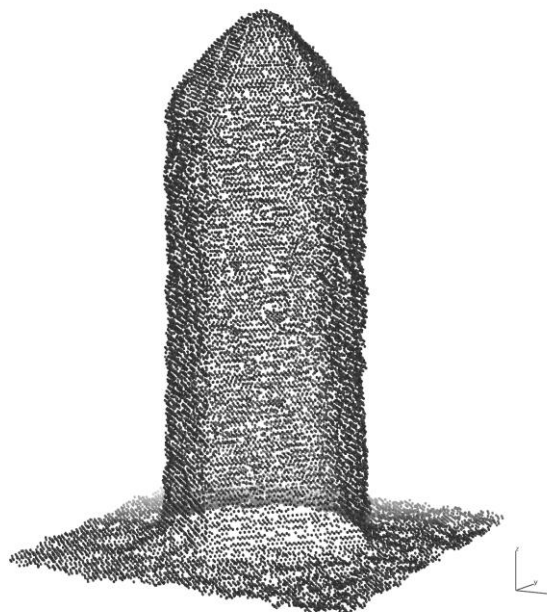


Figure 5: Surface nodes after the KMC simulation with `MuLSKIPS`.

routine exploits the `meshio Mesh` Python class and its I/O routines to export a new MSH file, identical to the input one, but which now includes the new data on the tetrahedral cells (stored in the `Dolfin` mesh function constructed in the previous step) as an additional *Element Data* block. It should be noted that in such a *Element Data* block of the MSH file only cell indices are used to mark the mesh regions modified by the atomistic process, therefore the $\{x,y,z\}$ node coordinates from the original mesh are preserved.

2.2 Simulation of topography evolution based on KMC surface node positions

The procedure to couple the `MuLSKIPS` output file, containing the position of surface nodes obtained from the LKMC simulation, and the TDR file (Synopsys format) containing the initial structure operates as follows:

1. `MuLSKIPS` provides the positions of the surface nodes as result of the KMC simulation (Fig. 5).
2. Using the deposition simulator DEP3D [5], the node positions from `MuLSKIPS` are used to determine a polyhedron describing the new topography entity with a triangle-based description instead of the node-wise description provided by `MuLSKIPS`. DEP3D is background work of Fraunhofer IISB. As foreground work within the project, the feature allowing this operation has been implemented in DEP3D.
3. The new region, represented by this polyhedron, is added to the regions of the structure before deposition. To this end, the vertices, edges, and faces constituting the new region are merged with the vertices, edges, faces, and elements of the already existing regions. The hierarchical dataset structure Vertices \Rightarrow Edges \Rightarrow Faces \Rightarrow Elements \Rightarrow Regions is updated accordingly to have a consistent dataset for the structure including the new region. Using DEP3D and Synopsys tools, the final structure is obtained in TDR format (through the boundary ASCII file in BND format). This structure can then be handled by the Synopsys process and device simulation tools in further simulation steps.
4. The physical data fields are included in the final structure. If only the geometry was modified by the LKMC simulations, the dataset from the initial TDR can be directly imported in the new TDR file with commands available in Synopsys tools. If the dataset of physical properties is modified, a further step is necessary. This is done before the conversion to TDR, by modifying the ASCII file that stores the physical dataset (DAT file). A new DAT file, which contains the information about the data fields defined at mesh vertices or elements (according to the type of field) is generated. This new DAT file contains the dataset of the final structure after KMC simulation, and it is written according to the dataset text file (which contains fields as a function of coordinates) given as output by `MuLSKIPS`. The procedure ends by converting the DAT and the corresponding ASCII GRD mesh file to a TDR file, using the Sentaurus Data Explorer.
5. As result, a continuum description of the structure is obtained which can be further treated within the Synopsys software by process or device simulation.

In the following, the procedure is demonstrated for the simulation example described above, that is, the CVD growth of a silicon layer on a vertical nanowire. Fig. 4 shows the structure atoms after the KMC simulation. The positions of the surface nodes (as described above in Section 2.1), visualized in Fig. 5, are imported in DEP3D. The polyhedron generated by DEP3D

after adding it on top of the initial surface of the nanowire is shown in Figs. 6 and 7, as visualized with the Synopsys visualization tool. The faceting which is predicted by the KMC deposition simulation is visible, particularly in the horizontal cross section shown in Fig. 6.

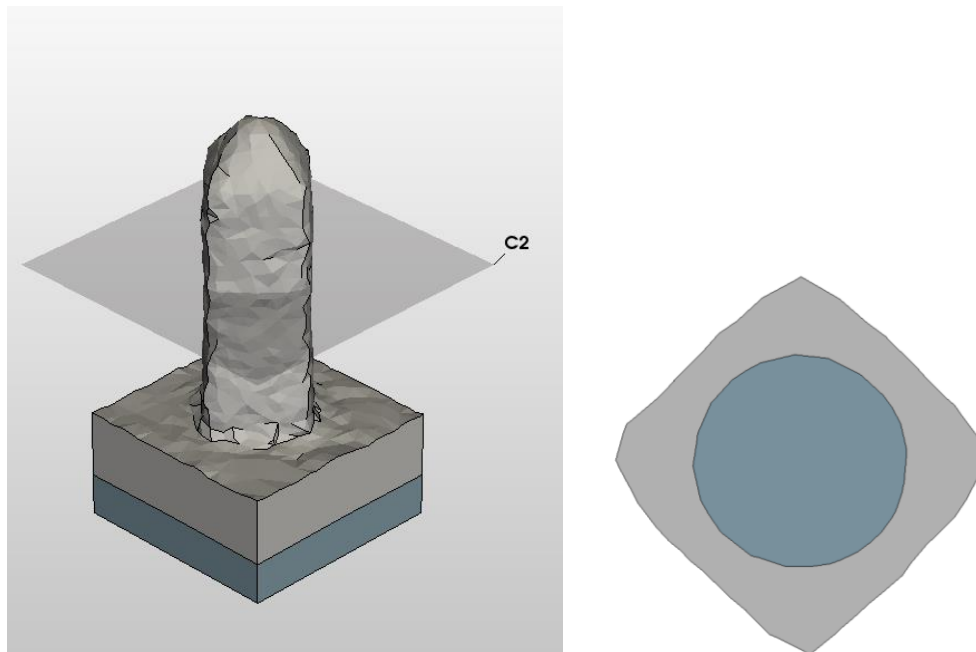


Figure 6: Simulation result obtained by generating a polyhedron that has been constructed with DEP3D based on the surface node positions delivered by MULSKIPS. The original silicon material is shown in dark gray, the new silicon material deposited by CVD is shown in light gray. On the left, the 3D structure is shown (Synopsys visualization tool). On the right, a horizontal cross section according to the plane shown on the left is depicted.

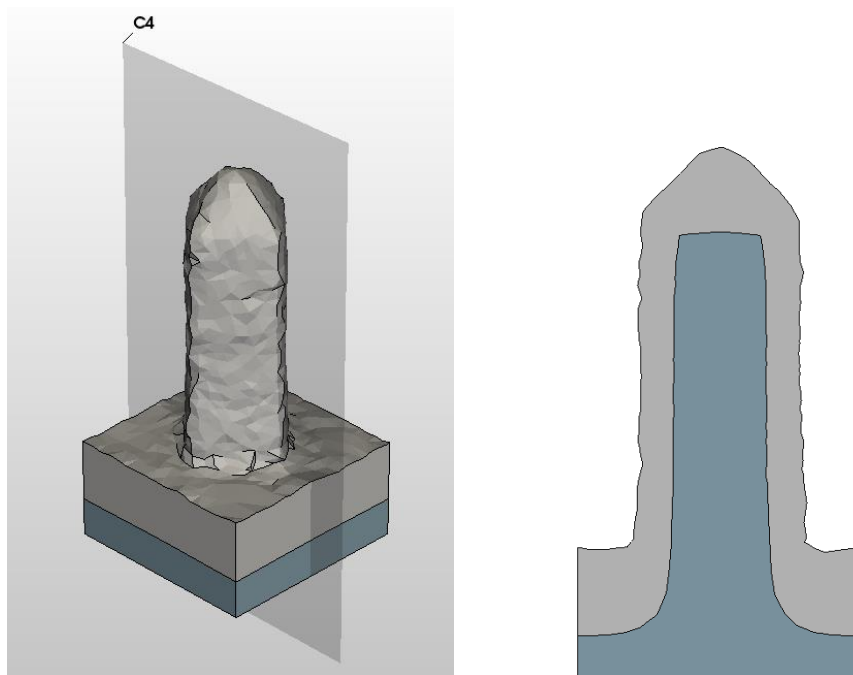


Figure 7: Same simulation result as shown in Fig. 6 (Synopsys visualization tool). This time a vertical cross section according to the plane shown on the left is shown on the right.

It should be noted that the number of surface discretization elements is significantly reduced by this procedure: The number of surface nodes of the structure from `MuLSKIPS` shown in Figure 5 is 24127. The number of surface triangles, that is, the triangles open to the gas volume, for the structure shown in Figures 5 and 6 is 2505. This also results in a smoothing of the surface. DEP3D includes algorithms that perform thin triangle deletion and refinement of large triangles to allow for a more efficient treatment of the surface. The operation of this algorithm can be controlled by parameters for the corresponding tolerances. E.g., for the minimum height of the triangle a value can be specified. The larger this value, the more effective thin triangle removal will be. The smoothing of the atomistic surface obtained from LKMC simulations is of paramount importance. Without the smoothing, the surface in the continuum structure (in TDR format) would be very rough and it would be very difficult, or even impossible, to further use the resulting structure in Sentaurus Process or Device. The careful node reduction performed in this procedure is beneficial for the efficiency of further simulation steps to follow.

3 Coupling of oxidation simulations of WP5 and Synopsys tools

The final objective is the simulation of various processing steps, including the thermal oxidation of Si. A key output of these computations is the formation of defects relevant for the device reliability.

The work carried out on the understanding of the oxidation step by the TUW and LAAS-CNRS teams (see Deliverables D5.1 to D5.3) have revealed an early transition from a crystalline silicon lattice to an amorphous oxide. This rapid amorphization is facilitated by multiple possible pathways upon O₂ adsorption and dissociation. To account for the real structure of the oxide (amorphous) and the interfacial zone (non-crystalline) as well as possible seeds for defect formation, it is crucial to represent the real structure of the oxide and the interface in the KMC in order to reliably predict defect formation.

To this end, we have chosen to develop an off-lattice kinetic Monte Carlo method. However, the development of this off-lattice kinetic Monte Carlo method requires the knowledge of a catalog of events in a traditional way, whether this catalog is predefined in input data (from DFT calculations for instance) at the start of the simulation, or created on-the-fly using a computationally cheap but physically valid energy model (such as an interatomic potential). To overcome the lack of accuracy and completeness commonly associated with an interatomic potential, we develop a two-track strategy for the predictive simulation of oxide growth by implementing defect formation.

Thus two implementation tracks will be applied to integrate the DFT results into the existing Synopsys TCAD tools (Fig. 8): for track 1/ physically reasonable electronic properties of defects are inferred based on a collection of defects acquired by TUW, for track 2/ a study using KMC developed in LAAS-CNRS simulations is conducted in order to link the process parameters with the formation of defects according to the local atomic structure of the oxide and at the interface as obtained during the growth with the goal to predict the defects most likely to exist.

TRACK 1. The DFT and molecular dynamics simulations in combination with ML tools, provide information and quantification of the defects in the oxide, in particular on the activation and formation energy of the defects in close connection with their configuration and local chemistry. It is necessary to exploit these results obtained. With the help of this constructed defect

database, we provide theoretical statistics on the different reaction barriers concerning the oxidation of Si and the formation of defects in SiO₂. This new information will be compared against the currently available model within Synopsys tools. The gathered information on the formation energies will be used to estimate defect concentrations as a function of temperature. In combination with the other available defect information like trap levels and relaxation energies, the electronic response of those defects can be computed for arbitrary operating conditions.

TRACK 2. At this stage, the link with the atomic structure obtained during growth and by varying the oxidation technological parameters (mainly temperature and O₂ flow) will be established. The off-lattice KMC allows to simulate a local structure of the oxide and the interface as formed by the technological processes. This KMC uses an interatomic potential to simulate the formation of defects during growth. The formation of these defects will then be studied as a function of the process parameters. The resulting structures of the growing oxide layer will be provided to TUW, to establish a link with the already available theoretical defect database and the local atomic structures obtained in KMC. In addition, the TUW DFT database will be used as a characterization basis for the electronic properties of the defects thus formed. If some defects, which have been identified during the theoretical study in track 1, are of interest but are not obtained in KMC, i.e., they are not present in the KMC catalog, it will be possible to generate new KMC events, using the existing defect catalog from TUW. For this, the strategy will be to make statistics of the growth of the oxide layer with the KMC tool at LAAS-CNRS, to analyze the obtained structures with a ML potential particularly trained on Si oxidation developed by TUW, and to capture the mechanisms of defect formation thanks to the TUW work in the KMC from LAAS-CNRS. As a result, the KMC will provide to Synopsys TCAD tools a list of the most likely defects to be formed during growth of the oxide and their localization at the interface and in the oxide layer.

At the end of the project, we will be able to derive statistics of defect formation in the oxide and at the interface during growth. We will be able to provide this information as a function of the process parameters as input to the Synopsys Sentaurus Device simulator. In particular, a file with a list of effective single defects and their respective charge trapping parameters will be generated. Here, a nonradiative multi-phonon (NMP) model will be used to describe the behavior of oxide defects under different operating device conditions. The defects are

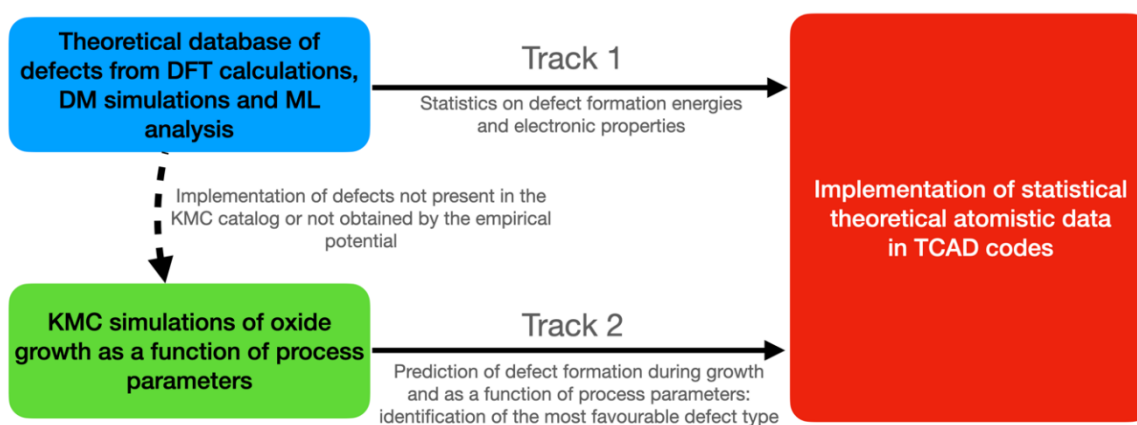


Figure 8: The two implementation steps from atomic scale simulations to Synopsys TCAD tools.

assumed to have two internal states (neutral and charged) with the associated defect parameters governing the transition rate between both states. Synopsys Sentaurus Device employs the extended NMP model. The extended NMP model considers a special trap having four internal states: neutral stable state, positive stable state, neutral metastable state and positive metastable state [16-17], as shown in Fig. 9. To make the generated 2-state data useful within Sentaurus Device, it has to be embedded within the implemented 4-state extended NMP model. To do so, the transition barrier between the metastable and its corresponding stable state will be set to a very small value, allowing for an instant transition between them. In this way, the two states effectively behave like a single state, resulting in a total of two states, as provided by the input model. This extension will ensure that the fundamental defect physics is conserved and the final model in TCAD will be consistent with the results previously obtained.

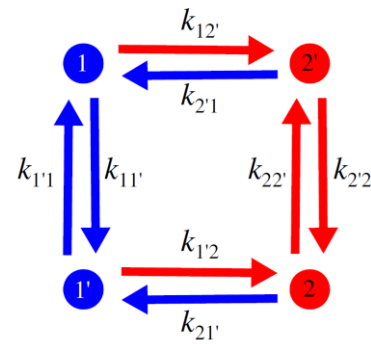


Figure 9: State diagram for the eNMP model, adapted from [17].

4 Conclusions

The results of the work done for interfacing external KMC/LKMC tools into the Sentaurus TCAD workflow have been presented. The `MuLSKIPS` software package was successfully coupled to the Synopsys environment and a consistent simulation toolchain was demonstrated. The full procedure involves different tools, namely Synopsys, `MuLSKIPS`, `gmsh`, `FEniCS` and `DEP3D`. The sequence of conversions steps and file exchange strategy among the different environments have been detailed. As a result, starting from a structure simulated in Synopsys TCAD, one or more process steps can be simulated by KMC in `MuLSKIPS` and then the resulting structure can be imported back in Synopsys TCAD for further process or device simulation. Next, the coupling strategy for the oxidation simulations have been described. This strategy, different from the previous one, involves the calculation of defect formation statistics in the oxide and at the interface as a function of process parameters with a KMC approach. The obtained parameters are then provided as input to the Synopsys TCAD tools.

References

- [1] Sentaurus Process of Synopsys TCAD, release 2021.06, <https://www.synopsys.com/silicon/tcad.html>
- [2] `gmsh` v4.4.1, available at <https://gmsh.info/>
- [3] C. Geuzaine & J.-F. Remacle, *Gmsh, A 3-d finite element mesh generator with built-in pre- and post-processing facilities*, Int. J. Numer. Methods Eng. **79**, 1309–1331 (2009).
- [4] `grd_to_msh`, available at https://github.com/hsahasra/grd_to_msh
- [5] `DEP3D`, physical deposition simulator and geometry editor, release 0.7.0, May 2022, Fraunhofer IISB, Erlangen, Germany

- [6] Alnæs, M. et al., *The FEniCS project version 1.5*, Archive of Numerical Software **3** (2015).
- [7] FEniCS 2019.2.0, available at <https://fenicsproject.org/>
- [8] G. Calogero et al., *Multiscale modeling of ultrafast melting phenomena*, npj Computational Materials **8**:36 (2022)
- [9] MulSKIPS, available at <https://github.com/MulSKIPS/MulSKIPS>
- [10] N. Schlömer, *meshio: Tools for mesh files*, doi: 10.5281/zenodo.1173115
- [11] meshio v5.0.2, available at <https://github.com/nschloe/meshio>
- [12] S. Blauth, *cashocs: A Computational, Adjoint-Based Shape Optimization and Optimal Control Software*, SoftwareX, **13**, 100646 (2021), doi: 10.1016/j.softx.2020.100646
- [13] cashocs v1.4.0, available at <https://github.com/sblauth/cashocs>
- [14] Paraview, available at <https://www.Paraview.org/>
- [15] CEA. V_{sim} , available at: https://www.mem-lab.fr/en/Pages/L_SIM/Softwares/V_sim.aspx
- [16] W. Goes, Y. Wimmer, A.-M. El-Sayed, G. Rzepa, M. Jech, A.L. Shluger, T. Grasser, *Identification of oxide defects in semiconductor devices: A systematic approach linking DFT to rate equations and experimental evidence*, Microelectronics Reliability **87**, 286-320 (2018).
- [17] Synopsys Sentaurus Device User Guide, Version S-2021.06, June 2021