Solid-State Electronics 191 (2022) 108258

Contents lists available at ScienceDirect

Solid State Electronics

journal homepage: www.elsevier.com/locate/sse

Curvature based feature detection for hierarchical grid refinement in TCAD topography simulations

Christoph Lenz [a,*], Alexander Toifl [a], Michael Quell [a], Francio Rodrigues [a], Andreas Hössinger [b], Josef Weinbub [a]

[a] Christian Doppler Laboratory for High Performance TCAD, Institute for Microelectronics, TU Wien, Gußhausstraße 27-29, Vienna 1040, Vienna, Austria
[b] Silvaco Europe Ltd, Cambridge, United Kingdom

ARTICLE INFO

Keywords:
Technology computer-aided design
Topography simulation
Level-set method
Curvature
Hierarchical grid
Non-planar epitaxial growth

ABSTRACT

We present a feature detection method for adaptive grid refinement in hierarchical grids used in process technology computer-aided design topography simulations based on the local curvature of the wafer surface. The proposed feature detection method enables high-accuracy simulations whilst significantly reducing the run-time, because the grid is only refined in areas with high curvatures. We evaluate our feature detection method by simulating selective epitaxial growth of silicon-germanium fins in narrow oxide trenches. The performance and accuracy of the simulation is assessed by comparing the results to experimental data showing good agreement.

1. Introduction

Many non-planar semiconductor device geometries (e.g., FinFETs) are fabricated by employing strongly anisotropic processing techniques [1]. One of those techniques is selective epitaxial growth (SEG) which is characterized by crystal facets. The resulting wafer topographies contain high-curvatures and essentially flat areas [2]. Consequently, the underlying grids need to offer high resolutions to accurately resolve the high-curvature features and the material interfaces during a process simulation, while keeping the overall run-time as low as possible to maximize practicability.

The level-set method [3] is well-suited for simulating fabrication processes of semiconductor devices in technology computer-aided design (TCAD) workflows [4–9]. Here, the wafer surface is described by a continuous function ϕ in the simulation domain Ω. The *zero level-set* of ϕ is defined as

$$\{x \in \Omega \mid \phi(x) = 0\}, \tag{1}$$

which is identical to the wafer surface. The function ϕ is named level-set function. The propagation of the wafer surface, i.e., its topographical evolution, is governed by the level-set equation

$$\frac{\partial \phi(x,t)}{\partial t} + V(x)|\nabla \phi(x,t)| = 0, \tag{2}$$

where $V(x)$ is the velocity field which models growth or etch rates during a process step [4]. The level-set equation is typically solved on a regular (Cartesian) grid with spatial resolution Δx using a finite difference scheme [10,11]. It is sufficient to only consider a small band of grid points around the zero level-set that is big enough to calculate finite difference schemes, the so called *narrow-band* [12]. Since the level-set values (ϕ values) are only known on grid points next to the zero level-set, these values have to be propagated further into the domain to enable the calculation of finite difference schemes. This is achieved with a so-called re-distancing step. A widely used method to compute the re-distancing step is the fast marching method which solves the *Eikonal Equation* on a grid [13]. Furthermore, the physical models used to calculate the surface rates (velocities) are only defined on the wafer surface or material interface (zero level-set), thus these rates also have to be propagated into the domain. The propagation is achieved with a velocity extension step [14].

As previously indicated, the need for high grid resolutions to optimally describe regions of interest (i.e., surface curvatures and material interfaces) can be efficiently tackled with hierarchical grids. Hierarchical grids are comprised of a *base* grid covering the entire simulation domain, which is complemented by *sub*-grids with higher resolutions. A natural straightforward approach is to cover the whole narrow-band around the zero level-set with sub-grids [15–17]. Another approach is to detect areas of interest in the simulation domain and cover them with

* Corresponding author.
E-mail address: lenz@iue.tuwien.ac.at (C. Lenz).

https://doi.org/10.1016/j.sse.2022.108258
Received 8 November 2021; Received in revised form 31 January 2022; Accepted 21 February 2022
Available online 26 February 2022

finer sub-grids [18]. However, typical wafer topographies in TCAD topography simulations are composed of large areas which are essentially flat with little to no geometric variation and small areas with pronounced geometric variation. Thus, covering the entire zero level-set or certain material interfaces with sub-grids would lead to large refined areas of the wafer surface that do not benefit from the higher resolution but still drastically increase simulation run-time. Therefore, we propose to employ a hierarchical grid placement algorithm that analyzes the wafer topography and only covers areas of interest with sub-grids (e.g., areas with high curvature) [19,20].

Fig. 1 shows an illustration of a level-set function with three features (i.e., corners) which are resolved with higher spatial accuracy by finer sub-grids. In order to reduce simulation run-time, these sub-grids need to be minimally sized and optimally placed, which, however, requires a feature detection method for automatic guidance.

In this work, we introduce an efficient and automatic feature detection method which is used to guide the sub-grid generation mechanism of hierarchical grid-based process TCAD topography simulations. At the core of the feature detection is the calculation and evaluation of the wafer surface curvature. The curvature of the wafer surface is calculated from the level-set representation of the wafer topography. Furthermore and considering related work, the curvature of the zero level-set has successfully been used in several applications requiring the detection of geometric features in various data sets [21–23] and thus serves as inspiration for this work. We assess our proposed method based on a representative and cutting-edge process simulation, e.g., selective epitaxy. To that end and to showcase integration into process TCAD workflows, our feature detection method has been implemented into Silvaco's *Victory Process* simulator [24].

The thus augmented simulator is used to selectively grow silicon–germanium (SiGe) fins. The results are compared to recent experiments [2]. Furthermore, we analyze the simulation run-time and accuracy of the simulated wafer topographies.

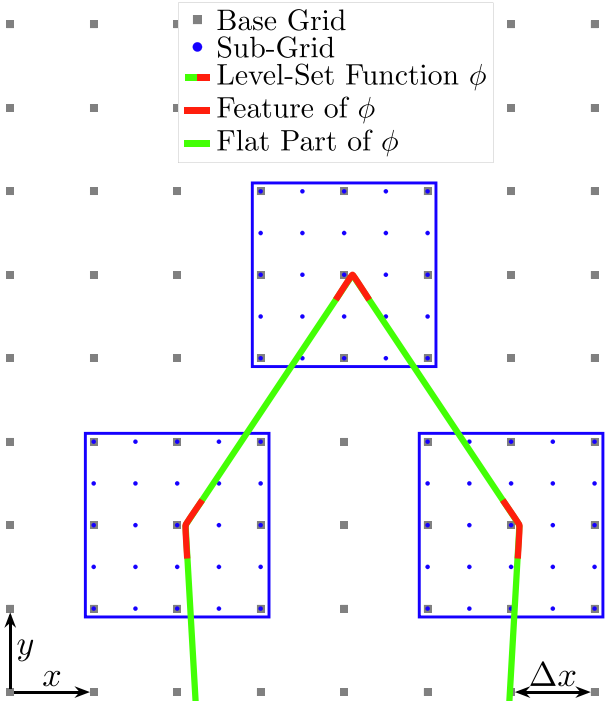

Fig. 1. Illustration of a level-set function ϕ (green/red line segments) with three features (i.e., corners; red line segments) on a hierarchical grid. The base grid has a resolution of Δx, the features of ϕ are covered by sub-grids with a two times higher resolution (blue boxes). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

2. Method

The automatic hierarchical grid placement algorithm is split into two steps. The first step is a feature detection step in which areas of interest on the base grid are detected and marked (flagged) as a feature. The second step consists of an automatic mesh placement algorithm which covers the flagged areas of interest on the base grid with non overlapping sub-grids with a finer resolution. In the example shown in Fig. 1 the resolution of the sub-grids is two times finer.

2.1. Feature detection

Typical topographies used in process TCAD simulations consist of multiple materials which in turn require multiple level-set functions for proper representation. To that end, a sophisticated stacking process of the level-sets using Boolean operations is used [25]. Each of these level-set functions has to be examined to find areas of interest of the topography which benefit from a sub-grid with a finer resolution.

For example: The wafer topography during a fabrication step with pronounced anisotropy is typically characterized by areas of high and low curvature. As previously indicated, the curvature of the level-set function ϕ is calculated with the curvature formula for 2 dimensional implicit curves

$$\kappa = \frac{\phi_y^2\phi_{xx} - 2\phi_x\phi_y\phi_{xy} + \phi_x^2\phi_{yy}}{|\nabla\phi|^3}, \tag{3}$$

where ϕ_i denotes the partial derivative of ϕ with respect to the coordinate $i \in \{x, y\}$ [26]. The partial derivatives are approximated using the second order central finite differences:

$$\phi_x \approx \frac{\phi_{i+1,j} - \phi_{i-1,j}}{2\Delta x}, \tag{4}$$

$$\phi_{xx} \approx \frac{\phi_{i+1,j} - 2\phi_{i,j} + \phi_{i-1,j}}{\Delta x^2}, \tag{5}$$

$$\phi_{xy} \approx \frac{\phi_{i+1,j+1} - \phi_{i-1,j+1} - \phi_{i+1,j-1} + \phi_{i-1,j-1}}{4\Delta x^2}. \tag{6}$$

The calculation of all required central finite differences (4), (5) and (6) to determine the curvature κ requires a 9-point finite difference stencil around each grid point of the wafer surface (see Fig. 2). This is important since, depending on the finite difference scheme used to solve (2), the width of the narrow-band needs to be adjusted.

For the sake of completeness, we need to consider the case where the curvature of a surface may not be defined at a *singular point*, i.e., a surface point where $\nabla\phi = (0, 0)$. However, this special case can be avoided by adding a positive ϵ to the denominator of (3). Thus we can assume that the curvature is defined on each grid point near the wafer surface: The absolute value of the curvature $|\kappa|$ is limited by 0 and $1/\Delta x$ since the maximal curvature a level-set function can describe is bound by the grid resolution [10]. Grid points with a curvature of $|\kappa| = 0$ identify a flat area of the wafer surface. In contrast, grid points with a larger value of $|\kappa|$ indicate a feature on the wafer surface. If the absolute curvature of a grid point exceeds $1/\Delta x$ it indicates that the resolution of the level-set function is not high enough to resolve this feature. Therefore, it is essential that such grid points are flagged as features to improve the simulation quality.

Consequently, topography features are detected based on the curvature threshold parameter $0 < C < 1/\Delta x$, which is simulation specific. If the curvature of a grid point is larger than C the grid point is flagged as a feature. The smaller the threshold parameter C is chosen, the more grid points are detected as a feature. Considering the fact that the calculated curvature is a numerical approximation, it is possible that the entire wafer surface is detected as a feature, because small numerical deviations from a flat plane can be above the threshold. Consequently,

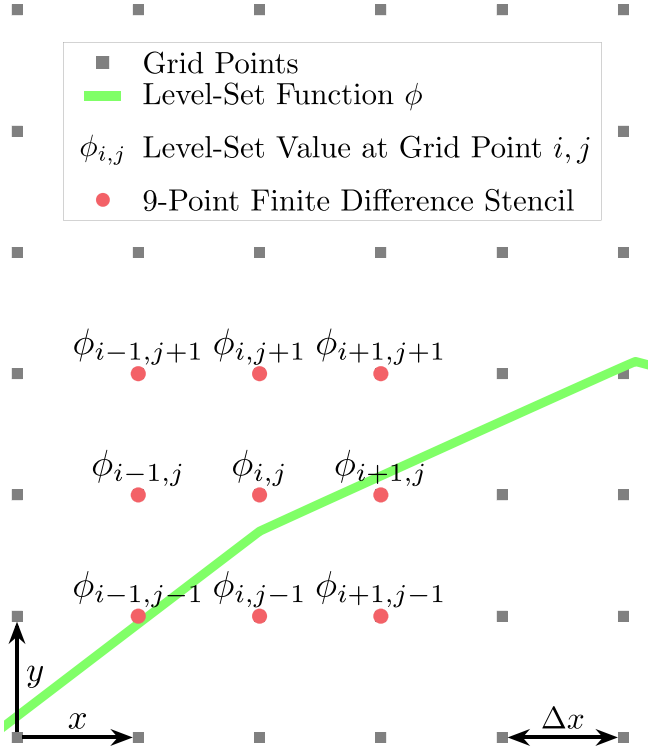

Fig. 2. Illustration of a level-set function and the 9-point stencil which is used to calculate the curvature of the surface at grid point $\phi_{i,j}$.

particularly small values for C should be avoided for a better distinction between features and non-features.

Furthermore, in this work, interfaces between stationary materials are always considered as features, which enables a well-resolved level-set description of SiGe material interfaces.

2.2. Hierarchical grid placement

The employed hierarchical grid placement algorithm is a slight variation of the grid generation algorithm of Berger and Rigoutsos [27]. After the feature detection step (see Section 2.1) the grid is considered as a black and white rectangular patch (or binary image) where a flagged grid point is considered to be black "1" and a non-flagged grid point is considered to be white "0". This patch is then put into a queue. The algorithm starts by measuring the *efficiency* of the first patch in the queue. The efficiency is calculated as

$$\eta(P) = \frac{\text{number of flagged grid points in } P}{\text{total number of grid points in } P}, \tag{7}$$

where P is the currently examined patch. This value is then compared against a patch efficiency threshold parameter E. Additionally it is checked if one of the sides of the rectangular patch has reached a minimum width M. If $\eta(P) \geqslant E$ or one of the patch's sides has reached the minimum width M the hierarchical grid placement algorithm considers this patch as accepted, removes it from the queue and puts the patch into a *pre-grid* list.

If the currently examined patch does not fulfill one of these criteria it is split into two patches. This is achieved by calculating the number of all flagged grid points for each row and column of the patch, these values are called the *signature* of the row or column (grid lines). The signatures are stored in two arrays for rows and columns (signature arrays). If there are elements in the signature array of the patch that have a signature of 0, the patch can simply be split along those grid lines. If there are no grid lines that have a signature of 0, the second derivative of each line in the signature arrays is calculated by using the finite difference formula (5) with $\Delta x = 1$. The thus created array of second derivatives of the signature array is searched for points where the sign of the second derivative changes from one point to the next (zero crossings). The patch is then split according to the biggest numerical change in the second derivatives of all zero crossings. If there is no obvious split, the patch is simply split in half. Afterwards, all grid lines that are on the border of the newly created patches and have a signature of 0 are removed. The thus created patches are put into the queue. If the queue is not empty the first patch in the queue is taken into consideration and the algorithm starts again by measuring the efficiency of this patch.

Fig. 3 shows an example of the above described hierarchical grid placement algorithm. Seven grid points of the level-set function are flagged as features, the minimal efficiency is $E = 0.6$ and the minimum patch side size is $M = 2$. The initial patch that covers the flagged grid points has an efficiency of $\eta(P) = 0.3$. This initial patch is split at the only zero crossing on the x-axis. The thus created patches have several grid lines with a signature of zero which are removed. The two created patches either fulfill the required efficiency (a) or may not shrink further since it would violate the minimum patch side size (b).

New sub-grids with, e.g., a four-times smaller Δx (facilitating locally increased resolutions) are then created according to the patches stored in the *pre-grid* list. The level-set values for the sub-grids are calculated by first calculating temporary level-set values by interpolating the level-set values from the base grid to the sub-grid using trilinear interpolation. The interpolation initializes the position of the zero level-set on each of the newly created sub-grids. The final level-set values of the new sub-grids are then calculated with a hierarchical re-distancing step [28].

3. Results

The proposed curvature based feature detection method is evaluated by simulating the two SEG processes presented by Jang et al. [2].

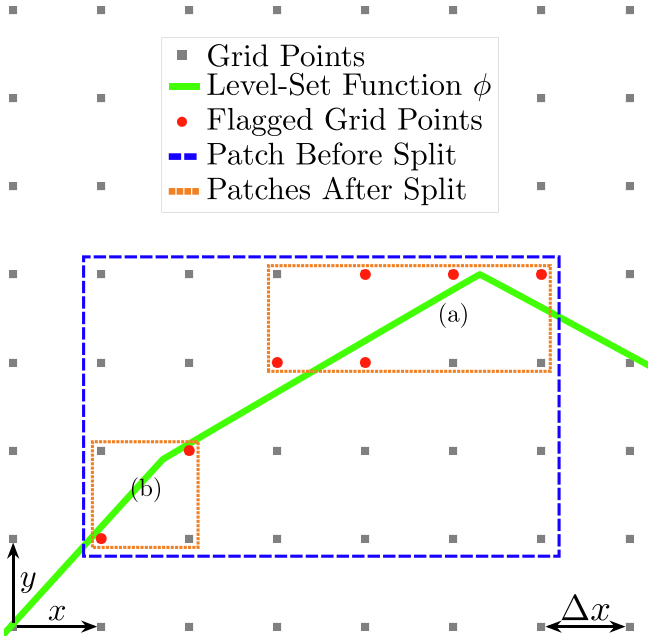

Fig. 3. Illustration of a level-set function on a base-grid with 7 flagged grid points, a minimum patch side size of $M = 2$, and an efficiency of $E = 0.6$. The initial patch (blue) is split into two smaller patches (orange). These patches are not split further since they are considered efficient (a) or would violate the minimal patch side size (b). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

3.1. Simulation setup

SiGe fins are grown inside a SiO_2 trench which is formed in an initial dry etching step on top of a Si substrate. In a cyclic SEG process a deposition and an etch step are repeated. The SiGe growth step consists of 15s of deposition from flowing source gases Si_2H_6 and GeH_4. The etch step consists of 12s of etching by flowing Cl_2, which is necessary to clean nucleated $Si_{1-x}Ge_x$ from the SiO_2 surface. The rate of the GeH_4 influences the desired Ge composition of the produced $Si_{1-x}Ge_x$ alloy. This cyclic process leads to the formation of high-quality $\{100\}, \{111\}$, and $\{311\}$ crystal facets. As previously indicated, this fabrication process has been simulated with Silvaco's *Victory Process* [24] augmented with the here proposed feature detection method: The method is employed during the SEG step, where we utilize a recently developed numerical stability-enabling advection scheme for selective epitaxy using the level-set method [11].

The simulation of the cyclic SEG process is abstracted to a continuous epitaxy process that is perfectly selective, i.e., the deposition rate on the SiO_2 walls is 0. The growth of the SiGe crystal is modeled with a crystal orientation-dependent velocity field V which is constructed from experimentally characterized growth rates [2]. Although the simulation does not directly simulate the cyclic process, we will refer to the simulated profile of the SiGe crystal that is created after one of these cycles as the profile after a *SEG step* or a *SEG cycle*. Note that one SEG cycle may consist of several level-set simulation time steps.

Jang et al. measured the deposition rates for the $\{100\}, \{111\}$ and $\{311\}$ facets for two different SiGe alloys $Si_{0.64}Ge_{0.36}$ (SEG1) and $Si_{0.45}Ge_{0.55}$ (SEG2), the rates for the crystal orientation-dependent velocity field (velocity-parameters) are given in Table 1. To utilize these rates during a simulation using the level-set method, the growth rates need to be available in each direction on the unit sphere. To create such a distribution of the rates given in Table 1 a four-rate Hubbard interpolation is used [29]. The simulation of the SiGe epitaxy with the respective velocity-parameters SEG1 and SEG2 have been performed with three different grid-settings, shown in Table 2.

Fine and *Coarse* grid-settings utilize a single base grid without sub-grids. The number of grid points in the simulations using *Fine* grid-settings is $\approx 3{\cdot}10^5$ and *Coarse* grid-settings is $\approx 1.9{\cdot}10^4$. This establishes a baseline to investigate the performance and accuracy gains of the hierarchical grid approach.

For the sake of completeness, we also considered a simulation with a grid resolution of 0.00125 μm which is between the *Fine* and *Coarse* grid resolutions. However, this grid-setting is not fine enough to accurately resolve the finer features of the SiGe crystal. Thus the accuracy gains in the simulation run with this grid resolution are not worth the performance loss when compared to the results obtained by the *Coarse* grid-setting.

The *Multi-Grid* grid-settings use a base grid (Grid 1) employing the same spatial resolutions as *Coarse* grid-settings complemented by one additional grid hierarchy level (Grid 2) offering a plethora of sub-grids with higher spatial resolutions. The number of grid points using *Multi-Grid* grid-settings is on average $\approx 2.7{\cdot}10^4$. The curvature threshold parameter used for the feature detection method has been empirically chosen as $C = 0.9$, additionally the interfaces of SiO_2 and SiGe are considered as features, see Section 2. The parameters for the hierarchical grid placement algorithm are as follows: The patch efficiency is set to $E = 0.7$ and the minimal patch side size is $M = 6$.

Table 1
Simulation parameters employed for the SEG in trench arrays [2] (velocity-parameters). The number of deposition cycles P_i refers to the number of SEG cycles needed to achieve the topographies in Fig. 4 and Fig. 8.

	Rates [nm/cycle]				Number of deposition cycles for profile P		
Name	R_{100}	R_{110}	R_{311}	R_{111}	P_1	P_2	P_3
SEG1	13	5	3.1	1.6	5	24	47
SEG2	5	3	3.5	1	8	33	55

Table 2
Grid resolutions employed for the SEG in trench arrays (grid-settings).

Simulation	Base Grid Resolution	Sub-Grid Resolution
Coarse	0.002 μm	–
Fine	0.0005 μm	–
Multi-Grid	0.002 μm	0.0005 μm

To ensure the stability of the numerical procedures used to solve (2), the maximal distance the zero level-set can propagate during a simulation time step is bound by the Courant-Friedrichs-Lewy (CFL) condition [10]. The CFL condition is determined by the sub-grids with the finest resolution and the maximal velocity. Thus, the feature detection, hierarchical grid placement and following re-distancing step do not need to be executed in every time step of the simulation (regridding). The regridding only needs to be invoked when a feature is going to be moved outside of a sub-grid. Thus, to further improve the run-time of the simulation, we empirically determined for this particular simulation problem that it is sufficient to perform the regridding every fourth simulation time step.

The results obtained by running the simulation with the *Multi-Grid* grid-settings are compared to the measurement data from the experiment presented by Jang *et al.* [2]. The final simulation results of the simulations using *Fine*, *Coarse* and *Multi-Grid* grid-settings are then analyzed by first comparing the simulated SiGe crystal surfaces generated with the different grid-settings against each other. To gain additional insights in the error that occurs when using the *Coarse* grid-settings the L_2-error of the three differently simulated SiGe crystal surfaces is compared. The simulated surface of the SiGe crystal using the *Fine* grid-settings is taken as the reference surface for determining the L_2-error. Thus, the L_2-error is determined by calculating the smallest distance from each point of the simulated surfaces using *Coarse* or *Multi-Grid* grid-settings to the reference surface. The distance between two surface points is calculated by determining the L_2-norm: $|(x_1, x_2)| = \sqrt{x_1^2 + x_2^2}$.

3.2. Discussion SEG1

Fig. 4 shows the simulation results of the SEG process modeled with the SEG1 velocity-parameters and the *Multi-Grid* grid-settings compared to the experiment. The simulation results are in good agreement with the experiment after all three measured cycles.

Fig. 5 compares the simulation results after the final simulation time step (47 SEG cycles) of the simulation using the different grid-settings shown in Table 2. The *Fine* and *Multi-Grid* simulation results are in good agreement. However, the *Coarse* simulation results do not sufficiently match the results using the *Multi-Grid* or *Fine* grid-settings (Fig. 5) which is evident from the mismatch of the peak positions of the SiGe crystal.

Fig. 6 shows the L_2-error of the simulation results using *Coarse* and *Multi-Grid* grid-settings compared with the simulation results using the *Fine* grid-settings. As discussed previously the *Coarse* simulation results do not sufficiently match the simulation results of the other simulations, as seen by the spike of the L_2-error in Fig. 6. Additionally, the maximum L_2-error of *Multi-Grid* is smaller than the minimum L_2-error of *Coarse*, showing the anticipated increase in accuracy of the hierarchical approach. There is a small L_2-error between the *Multi-Grid* results and the *Fine* results. However, the L_2-error of the *Multi-Grid* simulation is much lower than the base grid resolution. The expected error of a simulation with the level-set method is in the order of one Δx. Thus the simulation using the *Multi-Grid* grid-settings has an error that would be expected from a simulation using the *Fine* grid-settings.

The simulation run-times for different grid-settings are shown in

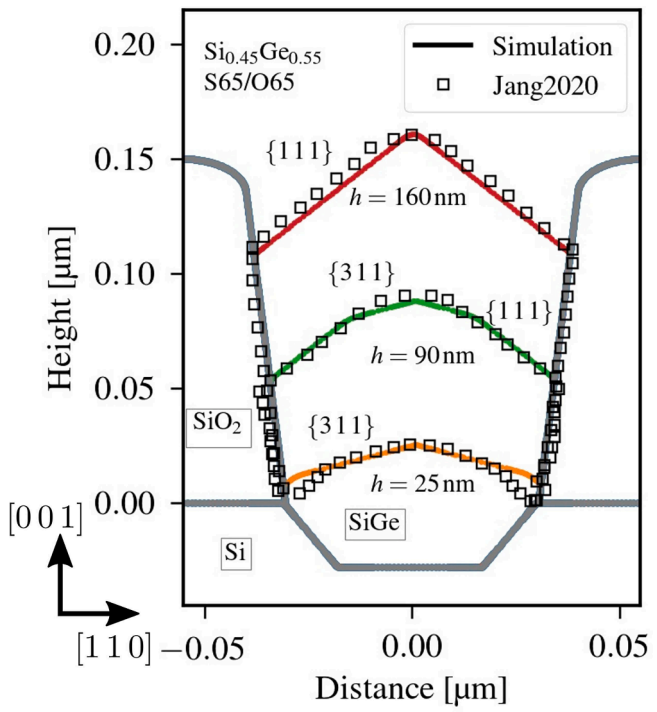

Fig. 4. Simulated surface of the SiGe crystal using the SEG1 velocity-parameters and *Multi-Grid* grid-settings compared with the experimental results from [2] after 5 (orange), 24 (green), and 47 (red) SEG cycles. The simulation results show good agreement with the experimental data. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

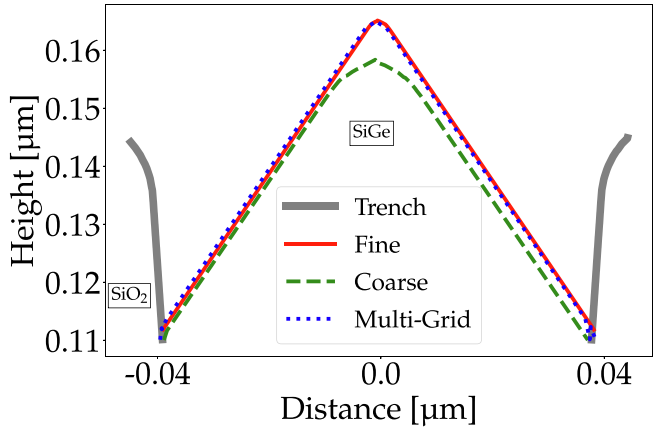

Fig. 5. Surface for the final simulation result of the SEG1 process after 47 SEG cycles using *Coarse*, *Fine*, and *Multi-Grid* grid-settings. The error in the peak of the SiGe crystal using *Coarse* resolution is largest, since the grid resolution is not high enough to properly simulate the SEG process at this feature.

Table 3. The simulation using the *Fine* grid-settings has the disadvantage of significantly increased simulation run-time. This is due to the higher resolution of the base grid, which increases the resolution in many irrelevant flat areas. Although the simulation using *Coarse* grid-settings has by far the fastest run-time, the quality of the final simulation result is not sufficient.

The simulation using the *Multi-Grid* grid-settings enables an excellent agreement with the experimental data and a negligible L_2-error (Fig. 6). The advantage of the *Multi-Grid* grid-settings is considerably improved performance (by 32%) compared to the *Fine* grid-settings.

The flagged grid points and thus generated sub-grids after 24 SEG

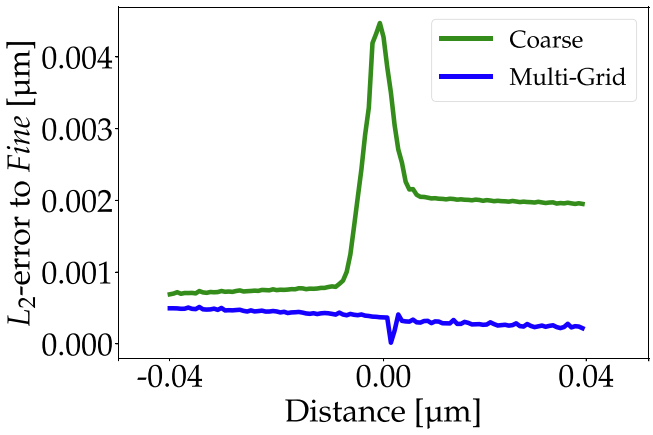

Fig. 6. Smallest L_2-error measured from the surface points of *Multi-Grid* and *Coarse* to the nearest surface point of *Fine*, in the final simulation result of the SEG1 process (see Fig. 5). The error of the simulation using the *Multi-Grid* grid-settings is negligible compared to the error when using the *Coarse* grid-settings.

Table 3
Run-times (Intel Xeon E5-2680v2) for the entire simulation (47/55 SEG cycles) of the SEG in trench arrays with respective grid-settings.

Simulation	Run-Time SEG1	Run-Time SEG2
Coarse	28 s	18 s
Fine	19 m 54 s	9 m 25 s
Multi-Grid	13 m 38 s	3 m 58 s

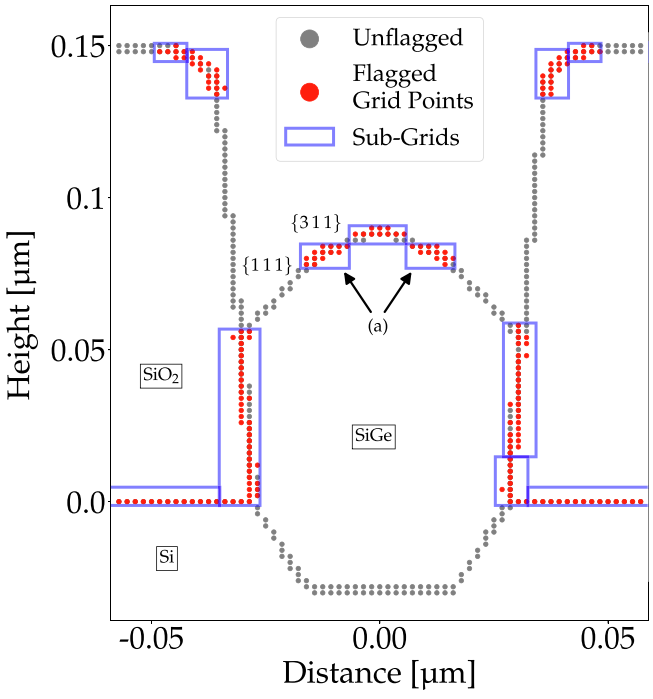

Fig. 7. Grid points near the level-set function for Grid 1 (i.e., base grid) of the simulation using *Multi-Grid* grid-settings after 24 SEG cycles using the SEG1 velocity-parameters. The flagged grid points (red) and generated sub-grids (blue boxes) for this time step are shown. (a) indicates sub-grids over fine features that develop during the SEG1 process. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

cycles, using the SEG1 velocity-parameters, are shown in Fig. 7. This simulation time step is chosen because the SEG process develops an additional geometrical feature between the {111} and {311} crystal facets, in addition to the peak of the crystal. Our feature detection method detects these fine features. The regridding algorithm places the sub-grids accordingly, as indicated by (a) in Fig. 7.

3.3. Discussion SEG2

Fig. 8 shows good agreement of the simulation based on the SEG2 velocity-parameters and *Multi-Grid* grid-settings compared to the experiment.

The simulation results shown in Fig. 9 compare the final surfaces after (55 SEG cycles) of the simulation using the grid-settings shown in Table 2. The simulation results run with *Coarse* and *Fine* grid-settings show a more evenly spaced error between the simulated surfaces compared to the simulation using the SEG1 velocity-parameters. The *Multi-Grid* results are again in good agreement with the *Fine* results.

The L_2-error of the simulations with the *Coarse* and *Multi-Grid* grid-settings is shown in Fig. 10. The maximum of the L_2-error occurs in the *Coarse* simulation results. As already observed in Fig. 9 the L_2-error of the simulation with the SEG2 velocity-parameters is distributed more evenly over the whole crystal surface than for the simulation with SEG1 velocity-parameters. However, it is still significantly larger than the L_2-error of the *Multi-Grid* simulation result. The L_2-error is much smaller than the base grid resolution. For the simulation with the SEG2 velocity-parameters, *Multi-Grid* shows the anticipated increase in accuracy of the hierarchical approach and a negligible L_2-error.

As with the simulation using the SEG1 velocity-parameters the *Coarse* simulation is much faster than the simulations utilizing the *Multi-Grid* and *Fine* grid-settings (see Table 3). However, as discussed in the previous paragraph the simulation results are not as accurate. The *Multi-Grid* grid-settings also show a significant reduction (by 58%) in the simulation run-time compared to the *Fine* parameters.

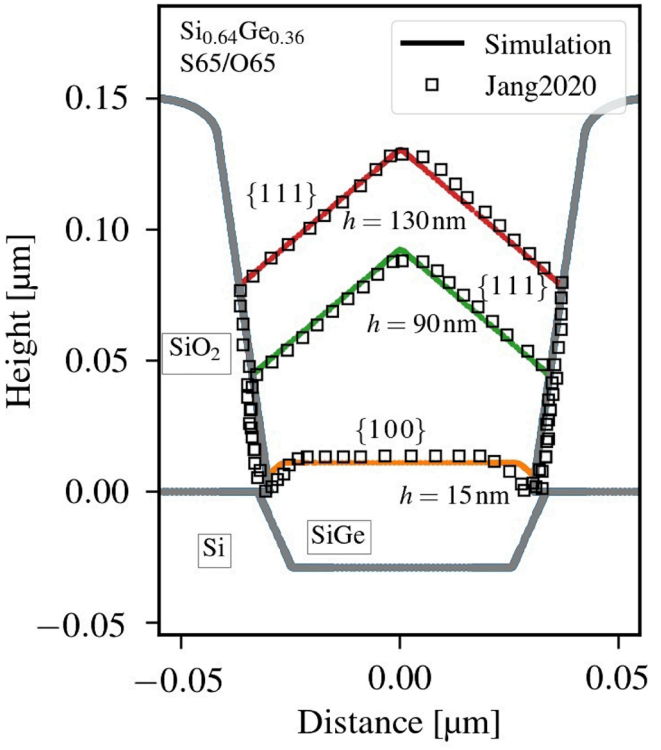

Fig. 8. Simulated surface of the SiGe crystal sing the SEG2 velocity-parameters compared with the experimental results from [2] after 8 (orange), 33 (green), and 55 (red) SEG cycles. The simulation results show good agreement with the experimental data. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

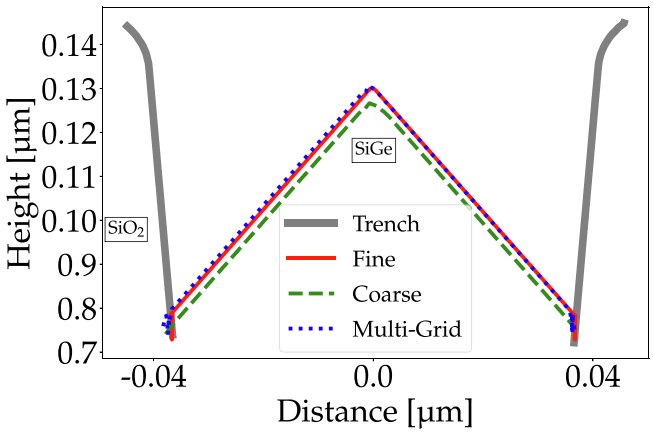

Fig. 9. Surface for the final simulation result of the SEG2 process after 55 SEG cycles using *Coarse*, *Fine*, and *Multi-Grid* grid-settings. The entire SiGe crystal surface using *Coarse* grid-settings is lower. Thus in this example the entire crystal surface has a big error due to the too small resolution.

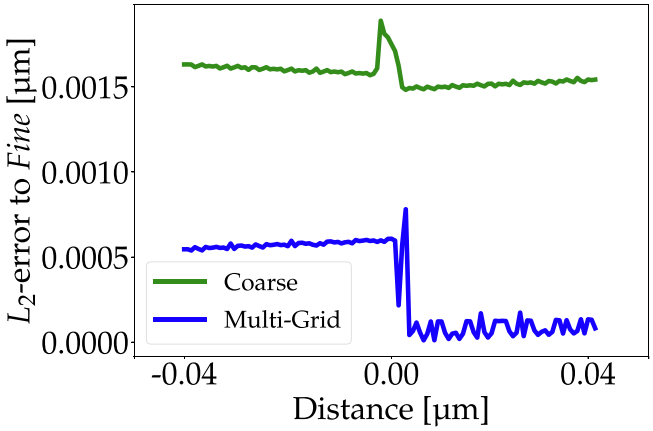

Fig. 10. Smallest L_2-error measured from the surface points of *Multi-Grid* and *Coarse* to the nearest surface point of *Fine*, in the final simulation result of the SEG2 process (see Fig. 9). The largest error is around the peak of the SiGe crystal of the simulation when using *Coarse* and *Multi-Grid* grid-settings.

When comparing Table 2 and Table 3 the SEG1 simulation has fewer SEG cycles than SEG2, however, SEG2 has a much shorter simulation run-time than SEG1. This run-time difference stems from the amplitude and complexity of the velocity field which influences the distance the surface can propagate in each time step of the simulation. The actual time steps the level-set method has to perform with *Multi-Grid* grid-settings to achieve the final simulation result are 2259 for SEG1 and 819 for SEG2. Thus the run-time difference is explained by the fact that the simulation with SEG1 velocity-parameters has to perform more than twice as many time steps than the simulation with the SEG2 velocity-parameters.

Fig. 11 shows the flagged grid points and thus generated sub-grids after 33 SEG cycles using the SEG2 velocity-parameters. In contrast to the simulation using SEG1 velocity-parameters (see Fig. 7) the crystal facets grow in the same direction and thus no additional features appear. Nevertheless the feature detection algorithm detects the peak of the SiGe crystal and places a sub-grid accordingly.

4. Conclusion

An efficient and automatic feature detection method for adaptive

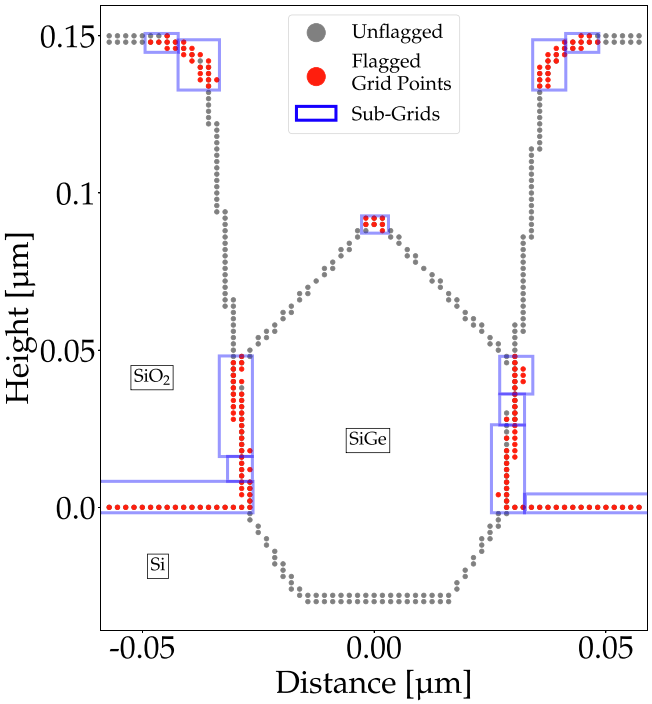

Fig. 11. Grid points near the level-set function for Grid 1 (i.e., base grid) of the simulation using *Multi-Grid* grid-settings after 33 SEG cycles using the SEG2 velocity-parameters. The flagged grid points (red) and generated sub-grids (blue boxes) for this time step are shown. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

grid refinement in hierarchical grids used in process TCAD topography simulations has been introduced. The feature detection method is based on the curvature of the level-set function representing a wafer surface. The efficiency of this method has been demonstrated for representative simulations of selectively grown epitaxial SiGe fins in oxide trenches. Two different growth parameter-sets for the SEG process have been successfully simulated. The developed feature detection, which has been used to optimally create the hierarchical grid, allows to use a low base grid resolution and only introduces sub-grids with higher resolution where crystal facets emerge during the SEG process and at material interfaces. By using our feature detection method the accuracy is maintained and more importantly the simulation run-time is considerably reduced by 32% and 58% for the two discussed simulation scenarios compared to single high resolution reference simulations.

Declaration of Competing Interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Christoph Lenz reports financial support was provided by Christian Doppler Laboratories.

Acknowledgment

The financial support by the *Austrian Federal Ministry for Digital and Economic Affairs*, the *National Foundation for Research, Technology and Development*, and the *Christian Doppler Research Association* is gratefully acknowledged. The authors acknowledge TU Wien Bibliothek for financial support through its Open Access Funding Programme.

References

[1] Peng J, Qi Y, Lo H-C, Zhao P, Yong C, Yan J, Dou X, Zhan H, Shen Y, Regonda S, Hu O, Yu H, Joshi M, Adams C, Carter R, Samavedam S. Source/Drain eSiGe Engineering for FinFET Technology. Semicond. Sci. Technol. 2017;32(9):094004. https://doi.org/10.1088/1361-6641/aa7d3f.
[2] Jang H, Koo S, Byeon D-S, Choi Y, Ko D-H. Facet Evolution of Selectively Grown Epitaxial $Si_{1-x}Ge_x$ Fin Layers in sub-100 nm Trench Arrays. J. Cryst. Growth 2020; 532:125429. https://doi.org/10.1016/j.jcrysgro.2019.125429.
[3] Osher J, Stanley J, Sethian. Fronts Propagating with Curvature Dependent Speed. J. Comput. Phys. 1988;79(1):12–49. https://doi.org/10.1016/0021-9991(88)90002-2.
[4] Sethian JA. Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision, and Materials Science. Cambridge University Press; 1999.
[5] F. Rodrigues, L.F. Aguinsky, A. Toifl, A. Hössinger, J. Weinbub, Feature Scale Modeling of Fluorocarbon Plasma Etching for Via Structures including Faceting Phenomena, in: Book of Abstracts of the International Workshop on Computational Nanotechnology (IWCN), vol. 254, 2021, pp. 101–102.
[6] Toifl A, Rodrigues F, Aguinsky LF, Hössinger A, Weinbub J. Continuum level-set model for anisotropic wet etching of patterned sapphire substrates. Semicond. Sci. Technol. 2021;36(4). https://doi.org/10.1088/1361-6641/abe49b.
[7] Aguinsky LF, Wachter G, Manstetten P, Rodrigues F, Trupke M, Schmid U, Hössinger A, Weinbub J. Modeling and Analysis of Sulfur Hexafluoride Plasma Etching for Silicon Microcavity Resonators. J. Micromech. Microeng. 2021. https://doi.org/10.1088/1361-6439/ac2bad.
[8] Salem MS, Saif OM, Shaker A, Abouelatta M, Alzahrani AJ, Alanazi A, Elsaid MK, Ramadan RA. Performance Optimization of the InGaP/GaAs Dual-Junction Solar Cell Using SILVACO TCAD. Int. J. Photoenergy 2021;2021. https://doi.org/10.1155/2021/8842975.
[9] Sehra K, Kumari V, Gupta M, Mishra M, Rawal DS, Saxena M. Impact of Heavy Ion Particle Strike Induced Single Event Transients on Conventional and p-Gate AlGaN/GaN HEMTs. Semicond. Sci. Technol. 2021;36(3). https://doi.org/10.1088/1361-6641/abdba3.
[10] Osher S, Fedkiw R. Level Set Methods and Dynamic Implicit Surfaces. Springer; 2003. https://doi.org/10.1007/b98879.
[11] Toifl A, Quell M, Klemenschits X, Manstetten P, Hössinger A, Selberherr S, Weinbub J. The Level-Set Method for Multi-Material Wet Etching and Non-Planar Selective Epitaxy. IEEE Access 2020;8:115406–22. https://doi.org/10.1109/ACCESS.2020.3004136.
[12] Chopp DL. Computing Minimal Surfaces via Level Set Curvature Flow. J. Comput. Phys. 1993;106(1):77–91. https://doi.org/10.1006/JCPH.1993.1092.
[13] Sethian JA. A Fast Marching Level Set Method for Monotonically Advancing Fronts. Proc. Nat. Acad. Sci. 1996;93(4):1591–5. https://doi.org/10.1073/PNAS.93.4.1591.
[14] Adalsteinsson D, Sethian JA. The Fast Construction of Extension Velocities in Level Set Methods. J. Comput. Phys. 1999;148(1):2–22. https://doi.org/10.1006/JCPH.1998.6090.
[15] Hubbard ME. Adaptive Mesh Refinement for Three-Dimensional Off-Line Tracer Advection over the Sphere. Int. J. Numer. Meth. Fluids 2002;40(3–4):369–77. https://doi.org/10.1002/fld.320.
[16] Cornford SL, Martin DF, Lee V, Payne AJ, Ng EG. Adaptive Mesh Refinement Versus Subgrid Friction Interpolation in Simulations of Antarctic Ice Dynamics. Ann. Glaciol. 2016;57(73):1–9. https://doi.org/10.1017/aog.2016.13.
[17] Löffler F, Cao Z, Brandt SR, Du Z. A new Parallelization Scheme for Adaptive Mesh Refinement. J. Comput. Sci. 2016;16:79–88. https://doi.org/10.1016/j.jocs.2016.05.003.
[18] A. Talpaert, Direct Numerical Simulation of Bubbles with Adaptive Mesh Refinement with Distributed Algorithms, Ph.D. thesis, Université Paris Sacla, 2017.
[19] Trompert RA, Verwer JG. A Static-Regridding Method for Two-Dimensional Parabolic Partial Differential Equations. Appl. Numer. Math. 1991;8(1):65–90. https://doi.org/10.1016/0168-9274(91)90098-K.
[20] J.G. Trompert, R.A., Verwer, J.G., Blom, Computing Brine Transport in Porous Media with an Adaptive-Grid Method, Int. J. Numer. Methods Fluids 16 (1) (1993) 43–63. doi: 10.1002/fld.1650160104.
[21] Liu Y, Kong F, Yan F. Level Set Based Shape Model for Automatic Linear Feature Extraction from Satellite Imagery. Sens. Transducers 2013;159(11):39–45.
[22] Beddad B, Hachemi K. Brain Tumor Detection by Using a Modified FCM and Level Set Algorithms, in. In: Proceedings of the International Conference on Control Engineering Information Technology (CEIT); 2016. p. 1–5. https://doi.org/10.1109/CEIT.2016.7929114.
[23] N. Christoff, A. Manolova, L. Jorda, S. Viseur, S. Bouley, J.-L. Mari, Level-Set Based Algorithm for Automatic Feature Extraction on 3D Meshes: Application to Crater Detection on Mars, in: Computer Vision and Graphics, Springer International Publishing, Cham, 2018, pp. 103–114.
[24] Silvaco, Victory Process (2021).www.silvaco.com/tcad/victory-process-3d/.
[25] Ertl O, Selberherr S. A Fast Level Set Framework for Large Three-Dimensional Topography Simulations. Comput. Phys. Commun. 2009;180(8):1242–50. https://doi.org/10.1016/j.cpc.2009.02.002.
[26] Goldman R. Curvature Formulas for Implicit Curves and Surfaces. Comput. Aided Geometric Des. 2005;22(7):632–58. https://doi.org/10.1016/j.cagd.2005.06.005.

[27] Berger M, Rigoutsos I. An Algorithm for Point Clustering and Grid Generation. IEEE Trans. Syst., Man Cybern. 1991;21(5):1278–86. https://doi.org/10.1109/21.120081.
[28] Quell M, Diamantopoulos G, Hössinger A, Weinbub J. Shared-memory block-based fast marching method for hierarchical meshes. J. Comput. Appl. Math. 2021;392:113488. https://doi.org/10.1016/j.cam.2021.113488.
[29] Ted J. Hubbard, MEMS Design: The Geometry of Silicon Micromachining, Ph.D. thesis, California Institute of Technology (1994).https://doi.org/10.7907/TK4C-M144.

Christoph Lenz was born in 1988 in Vienna, Austria. He studied Technical Mathematics at the Technische Universität Wien where he received his Bachelor's degree (2017) and the degree of Diplomingenieur (2019). After finishing his thesis which was written in collaboration with the Institute for Microelectronics, he decided to join the research team of the institute, where he is currently working on his doctoral degree. His research interests include high performance algorithms and data structures and he is working within the scope of the Christian Doppler Laboratory for High Performance TCAD.