Solid-State Electronics 200 (2023) 108534

Contents lists available at ScienceDirect

Solid State Electronics

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

Automatic grid refinement for thin material layer etching in process TCAD simulations☆

Christoph Lenz [a,*], Paul Manstetten [b], Luiz Felipe Aguinsky [a], Francio Rodrigues [a], Andreas Hössinger [c], Josef Weinbub [a]

[a] Christian Doppler Laboratory for High Performance TCAD, Institute for Microelectronics, TU Wien, Gußhausstraße 27-29, 1040, Wien, Austria
[b] Institute for Microelectronics, TU Wien, Gußhausstraße 27-29, 1040, Wien, Austria
[c] Silvaco Europe Ltd, Compass Point, St Ives, Cambridge, PE27 5JL, United Kingdom

ARTICLE INFO

Keywords:
Process technology computer-aided design
Topography simulation
Level-set method
Hierarchical grids
Light-emitting diodes

ABSTRACT

Thin material layers are common structures in modern semiconductor device fabrication and are particularly necessary for light-emitting diodes and three-dimensional NAND memory devices. Such layers are not only deposited on the flat wafer surface but are also partially removed during subsequent etching steps. Level-set based process TCAD simulations are capable of representing flat thin material layers, such as those occurring after deposition, with sub-grid accuracy. However, topographical changes during etching processes modeled via Boolean operations expose the low underlying grid resolution, leading to detrimental artifacts. We present a novel algorithm that analyzes the thickness of all material layers and derives a refined target resolution for local regions of thin layers affected by the etching process. This allows to accurately represent topographical changes in thin layers by hierarchically refining the grid without unnecessary refinement in unaffected regions of the domain. We simulate the fabrication of a light-emitting diode device using our algorithm to automatically predict the optimal resolution for all etched material layers. Our algorithm selects efficient refinement factors to obtain the local target resolutions of the hierarchical grids, and achieves a three times faster computation time than a benchmark refinement algorithm based on topographical features.

1. Introduction

The fabrication processes of many modern semiconductor devices – such as light-emitting diodes (LEDs) or *staircase* patterns in three-dimensional NAND flash memories – include process steps in which regions with thin material layers are involved [1,2]. These structures must be accurately simulated in order to advance technology development through process technology computer-aided design (TCAD) workflows. At the core of modern process TCAD approaches for topography simulation of, e.g., film deposition and etching, is the level-set method. In the level-set method the topography of the wafer is described by a function ϕ in the domain Ω, where the zero level-set of ϕ, representing the wafer surface, is defined as [3,4]

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

Typically, the level-set function is discretized on a regular grid with resolution Δx. To define the level-set function in the entire domain, the distance from the zero level-set to each grid point (i.e., the ϕ-value) has to be calculated. This is achieved through a *re-distancing* step, using, e.g., the fast marching method [5]. Thus, the zero level-set describes the isocontour of a volume and the sign of the ϕ-value at a given point informs about its position relative to the zero level-set: If the sign is negative, the grid point is inside the volume; if it is 0 it lies directly on the isocontour (see Eq. (1)); and if it is positive, it lies outside of the volume. Each material layer is represented using an individual level-set function, using a sophisticated layer wrapping approach for robustness [6].

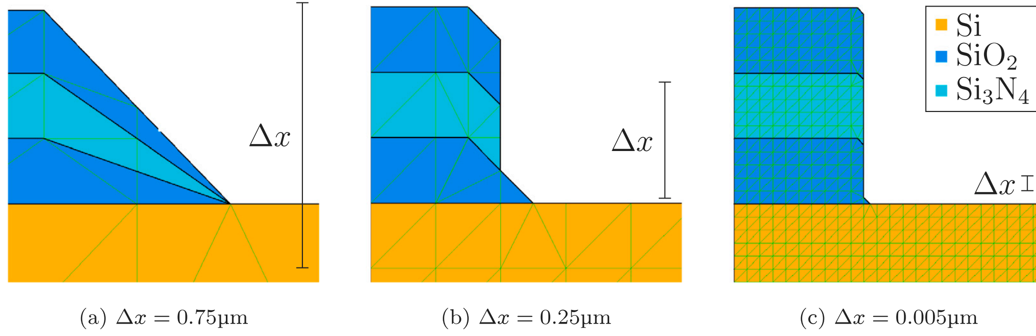
The level-set method is capable of representing flat thin material layers (nm regime), such as those occurring after the deposition of thin films, with a very coarse (μm) resolution. However, when parts of these thin material layers are etched, a process that can be modeled by Boolean operations (see Section 2.1), the low lateral resolution of the underlying simulation domain gets exposed and the material layers meld into each other. This introduces artifacts in the resulting representation of the structure which are shown in Fig. 1, where three stacked thin material layers using three different grid resolutions (Δx) relative to the material layer thickness d are shown. In Fig. 1(a), the

☆ The review of this paper was arranged by Francisco Gamiz.

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

https://doi.org/10.1016/j.sse.2022.108534

Available online 24 November 2022
0038-1101/© 2023 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).



(a) $\Delta x = 0.75\mu m$ (b) $\Delta x = 0.25\mu m$ (c) $\Delta x = 0.005\mu m$

Fig. 1. Thin material layers with a thickness of 0.25 μm after a Boolean operation with different grid resolutions.

resolution is $4d$; in Fig. 1(b), it is $2d$; and in Fig. 1(c), it is $d/6$. Although this indicates that increasing the resolution in the whole simulation domain is sufficient, such an approach is prohibitive due to its severe impact on overall simulation performance.

To alleviate this issue, hierarchical grids can be used to enable a locally increased resolution. A simulation employing hierarchical grids consists of a base-grid that spans the entire simulation domain and several nested sub-grids that cover areas of interest (features) with a higher resolution [7]. Each of the nested sub-grids increases the local resolution of the simulation domain by a given refinement factor. A feature detection algorithm based on the geometrical features (e.g., surface curvatures) of the topography is commonly used to guide the refinements during the evolution of the topography [7] and serves as a benchmark for the here-proposed algorithm. Such an approach is well suited for detecting emerging features in the topography. However, it detects features only after a Boolean operation has been performed, and thus distorts the information about the thickness of the material layers. To maintain the accuracy of the representation after the Boolean operation, the resolution refinement has to be performed before it is applied.

2. Method

Our algorithm for Boolean operations calculates the minimal distance between the closest material layers affected by the Boolean operation. This information is subsequently used to determine the minimal required local resolution (Δx_{tar}) to properly resolve the zero level-set (i.e., the material layers on the wafer) after the Boolean operation is complete. A hierarchical grid refinement algorithm is then employed to achieve the previously calculated target resolution. This algorithm uses a user-supplied refinement factor (F_{ref}) to locally increase the resolution of the level-set function in nested sub-grids placed over features, up to a user-specified maximum refinement level. The proposed algorithm has been implemented into Silvaco's *Victory Process* [8].

2.1. Boolean operations for etching simulation

In the level-set method the level-set function is used to describe a volume, thus, a Boolean operation between two level-set functions can be understood as a Boolean operation between two volumes. All commonly known Boolean operations like the union ($\cup$) or the intersection ($\cap$) between two level-set functions can be defined with simple functions [3]. These functions operate on the values of two level-sets functions (ϕ and χ) in the domain Ω. In this work, the relative complement of two level-set functions is used. The relative complement (also known as the set difference) describes the volume of the level-set function ϕ without the volume of χ and is defined as follows [3]

$$\phi(x)\setminus\chi(x) = \max(\phi(x), -\chi(x));\ \forall x \in \Omega. \quad (2)$$

For discrete level-set functions $\phi(x), \chi(x)$ a Boolean operation is performed by iterating over each grid point in the domain and evaluating the function for the Boolean operation (see Eq. (2)) with the respective values of ϕ and χ. This process results in a new level-set function ψ. Fig. 2 shows an illustration of this process.

Etching processes in a level-set based simulation are modeled with Boolean operations on level-set functions [9]. To that end, the material to be removed, or etched, is represented by an additional level-set function χ. The etching is modeled by computing the relative complement of the level-set function χ with each of the material layers which are affected by the etching process (see Fig. 3).

2.2. Required resolution for thin material layers

To determine the minimal required grid resolution Δx_{tar} necessary to accurately represent a thin material layer affected by a Boolean operation, we have to take into account a predefined minimal number of grid points to represent a single layer (N_{min}) and the distance to the closest material layer affected by the Boolean operation ($d_{closest}$). The distance d between two material layers (represented by the level-set functions ϕ and ψ) can be calculated using the ϕ/ψ-values at a grid point (i,j) as $d_{(\phi_{i,j},\psi_{i,j})} = \phi_{i,j} - \psi_{i,j}$. Fig. 4 shows an illustration of this calculation. Furthermore, the sign of the calculated distance holds information about the normal direction. If the sign of $d_{\phi_{i,j},\psi_{i,j}}$ is positive then the level set function ϕ lies in the normal direction of ψ, which is a consequence of the convention that ϕ-values on the inside have a negative sign. In the case that $d_{\phi_{i,j},\psi_{i,j}}$ is equal to zero, the two level-set functions are treated as if ϕ does not lie in normal direction of ψ.

Thus, the distance from a fixed level-set function ψ to the closest level-set function on the grid point (i,j) can be calculated by evaluating

$$d_{closest} = \min_{\phi^{k\in 1..n}}(|d_{(\phi^k,\psi)}|), \quad (3)$$

where n denotes the number of all other k level-set functions with value $\phi^k_{i,j}$. It follows that the minimal required grid resolution for the thinnest material layer is given by:

$$\frac{d_{closest}}{N_{min}} = \Delta x_{tar}. \quad (4)$$

2.3. Detection of affected material layers

To determine if the zero level-set of ϕ is affected by a Boolean operation with a level-set function χ at a grid point (i,j), both the ϕ and χ-values have to be considered. If the absolute ϕ and χ-values at (i,j) are both smaller than Δx_{curr} (the resolution of the currently examined sub-grid) then the zero level-sets of both level-set functions are close to the grid point and it is affected by the Boolean operation. This process also detects grid points of two level-set functions whose zero level-sets run parallel to each other or are part of the wrapping layer. Therefore, an additional check has to be made: If the signs of at least two of the

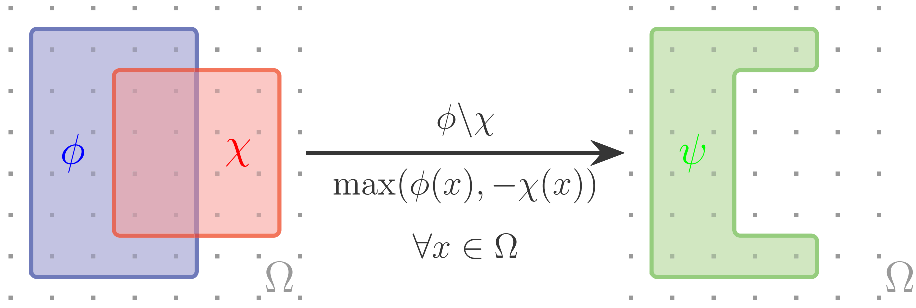


Fig. 2. Illustration of a Boolean operation (relative complement) of two level-set functions ϕ, χ defined on a grid in the domain Ω. The Boolean operation results in a new level-set function ψ describing the volume after the Boolean operation.

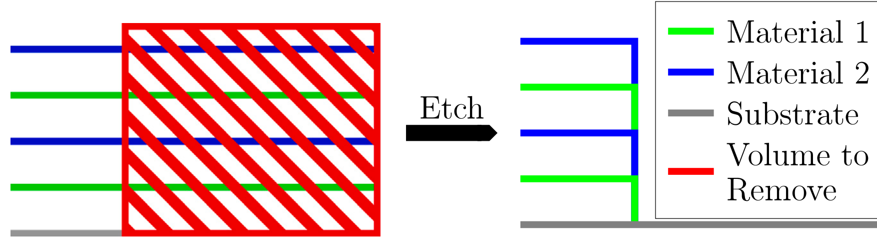


Fig. 3. Illustration of four stacked material layers on a wafer and a simulated etching process using a Boolean operation.

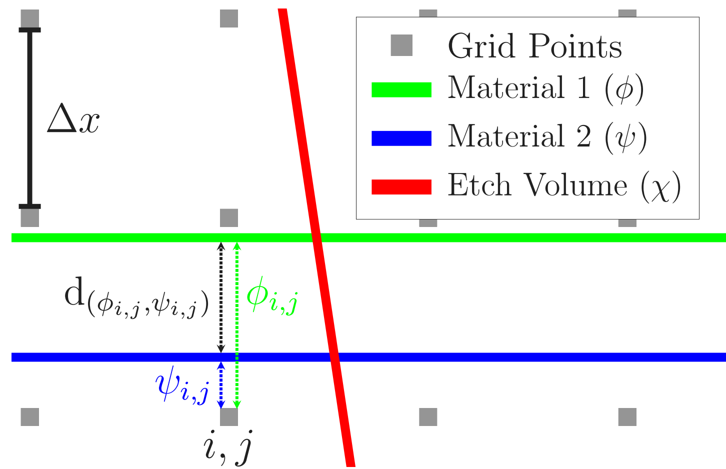


Fig. 4. Level-set functions involved in a Boolean operation: ϕ and ψ represent material layers, and χ represents the material to be removed. The distance between the level-set functions $d_{(\phi_{i,j},\psi_{i,j})}$ is calculated by using the ϕ, ψ-values of the grid point (i, j).

ϕ or χ-values changes in two coordinate directions then the grid point is affected by the Boolean operation. Otherwise, the zero level-sets of the two level-set functions do not intersect each other in the vicinity of this grid point and are thus parallel. Fig. 5 shows an illustration of two parallel level set functions (see Fig. 5(a)) and two intersecting level-set functions (see Fig. 5(b)).

2.4. Full Boolean operation algorithm on hierarchical grids

The algorithm starts with the calculation of the level-set function χ that describes the material that is to be removed on the base grid. Afterward, all affected level-set functions are identified. Next, the grid points at the intersection of the affected level-set functions with the level-set function χ are determined (see Section 2.3). At these intersection points, the minimal distance in the normal direction form the zero level-set that causes the intersection to all other zero level-sets is calculated. This is achieved by using Eq. (3) and additionally checking if the distance has a positive sign (see Section 2.2). If the sign is negative the distance to this zero level-set is ignored and the next level-set function is considered. The thus determined distance to the closest material layer is checked if it fulfills Eq. (4). If not, the grid point is marked for refinement. After all intersection points have been checked, the hierarchical grid placement algorithm is initiated utilizing F_{ref}. The level-set function χ is recalculated where a higher resolution is now available. The above procedure is repeated on the new sub-grids until a predefined number of grid refinement levels is reached (general refinement level). In a concluding refinement step, the required refinement level to properly represent the thinnest material layer (L_{ref}) of the final sub-grid is determined. This is achieved by considering the grid resolution of the currently examined sub-grid (Δx_{curr}), the calculated target resolution (Δx_{tar}), and the refinement factor (F_{ref}):

$$\left\lceil \frac{\log(\frac{\Delta x_{curr}}{\Delta x_{tar}})}{\log(F_{ref})} \right\rceil = L_{ref}. \quad (5)$$

Fig. 6 shows a flowchart of the algorithm described in this section.

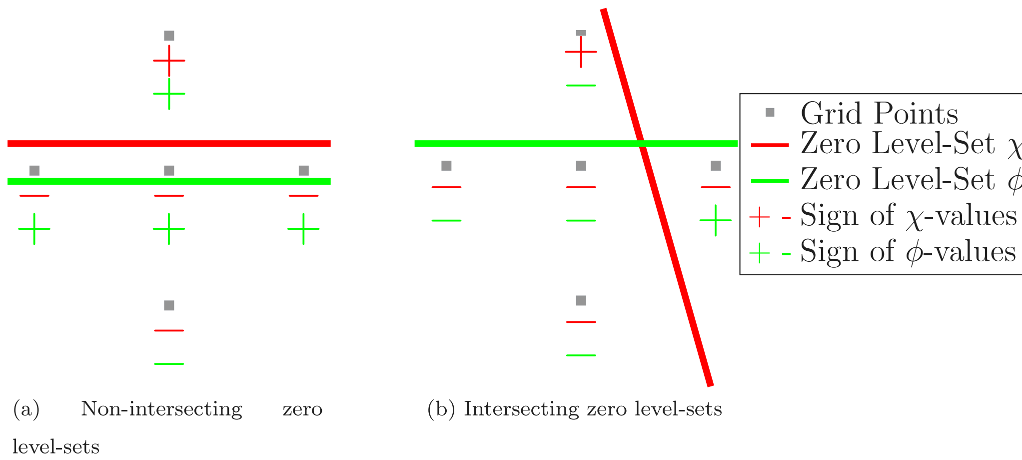


Fig. 5. Two level-set functions and the signs of the ϕ-values for intersecting and non-intersecting zero level-sets.

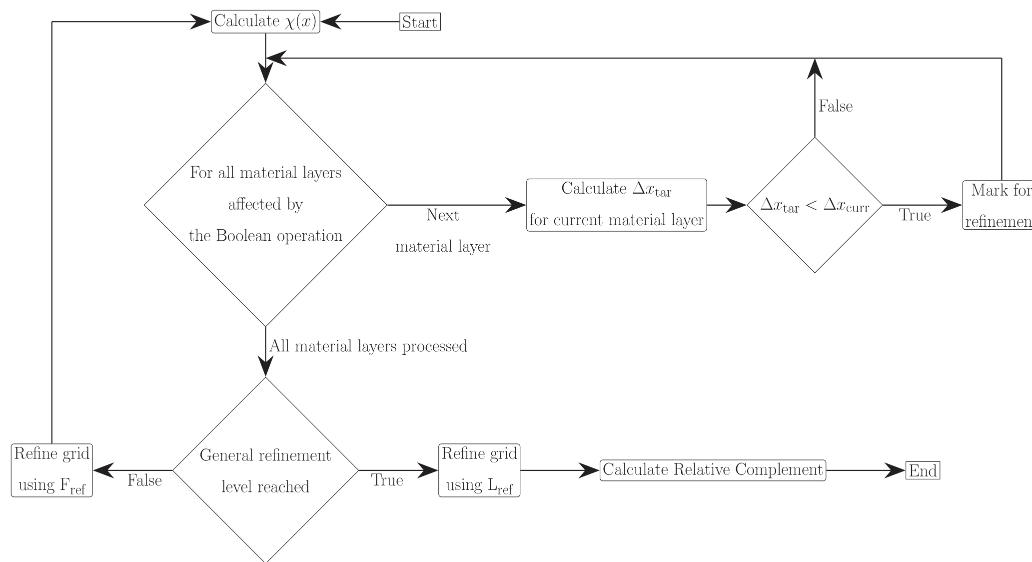


Fig. 6. Flowchart of the thin layer refinement algorithm operating on hierarchical grids.

In contrast, the benchmark feature detection approach lacks any information about the target resolution [7]. Thus, in these methods, a predefined fixed number of refinement levels has to be used. Furthermore, the availability of the target resolution, and thus the required refinement level, allows our algorithm to deviate from a fixed refinement factor for the final sub-grids.

3. Results

To evaluate the here-presented algorithm, we simulate the fabrication of an individual pixel of a LED array reported in the literature [2,10]. The authors report growing a 1.9 μm thick GaN layer on a (0001) sapphire substrate. Afterwards, 10 alternating layers with different thicknesses of InGaN and GaN for a total height of 117.5nm are deposited. The thinnest material layer is a 3nm thick InGaN layer. Subsequently, a 210nm thick p-GaN cap layer is grown on top of the structure [2,10]. To create a singular LED pixel with a diameter of 75 μm, the excess material is etched.

In the following, the entire simulation flow is considered. We use a base-grid resolution of 0.125 μm, $N_{min} = 6$, $F_{ref} = 4$, and require a minimum of two grid refinement levels (4-4), to guarantee an accurate representation of corners in the entire domain. For the simulation of the final etching process, the algorithm presented in this work is used. It follows from Eq. (5) that the finest sub-grid needs a resolution 256 times finer than the base-grid to accurately resolve the thinnest material layers.

We assess four different simulation configurations. The first configuration is a simulation utilizing only three fixed refinement levels (4-4-4) (e.g., the finest sub-grid has a 64 times finer resolution). The second and third configurations use four refinement levels to achieve the previously calculated minimal resolution (4-4-4-4). Moreover, the second configuration applies the benchmark geometrical feature detection algorithm and the third configuration applies our algorithm without the last step that dynamically changes the resolution of the final sub-grid. The fourth configuration uses our algorithm with three refinement levels and a final sub-grid with a 16 times finer resolution (4-4-16). Fig. 7 shows the LED device after etching through the thin material layers (active region).

Fig. 7(b) shows that choosing only 3 grid refinement levels (4-4-4) is not enough to accurately represent the thin layers after the etching process (see visible kinks between the material layers). Fig. 7(c) shows the results using a 4-4-16 and a 4-4-4-4 refinement which produce the same final topography. Table 1 shows the run times of the etching simulation. We observe that the simulation run with the benchmark geometric feature detection algorithm is the slowest, since it creates unnecessary hierarchical grids. Our algorithm is approximately one third faster than the benchmark when it uses the same four refinement

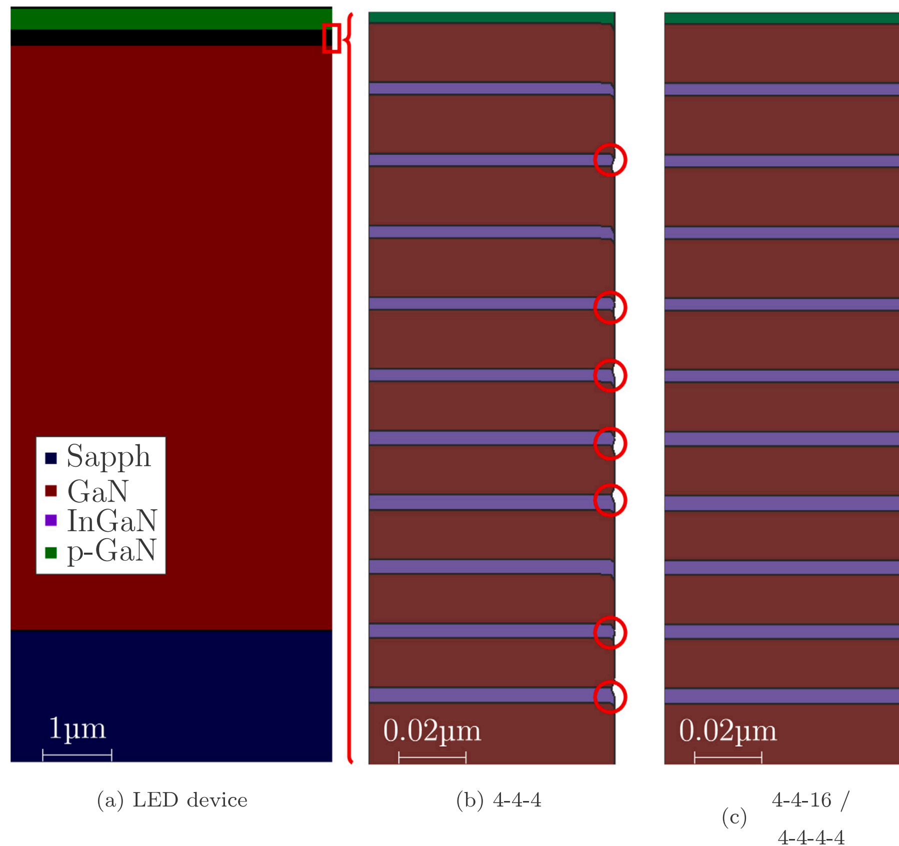


Fig. 7. LED device: (a) entire device, (b) active region with 3 grid levels (fixed refinement factor) — red circles highlight kinks resulting from the too low resolution, (c) active region with 4 grid levels. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 1
Etching simulation run times for different refinement level configurations (Intel Xeon E5-2680v2).

Feature detection method	Refinement factors	Run time
Our method	4-4-16	4 min 22 s
Our method	4-4-4-4	8 min 45 s
Benchmark, geometrical	4-4-4-4	11 min 45 s

levels (4-4-4-4). However, when our algorithm is allowed to deviate from a fixed refinement factor, thus taking advantage of information on the thinnest material layers, it achieves a three times faster run time than the benchmark.

4. Summary

We present an automatic grid refinement algorithm for thin material layers affected by a Boolean operation simulating an etching process. Our algorithm is able to determine the thickness of the material layers affected by the Boolean operation and dynamically uses a required minimal refinement level. This allows our algorithm to prevent the formation of artifacts between the affected material layers after the Boolean operation. Furthermore, the ability of our algorithm to dynamically determine the minimal required refinement level improves the computation time of the Boolean operation since, it can avoid the creation of additional unnecessary sub-grids. Thus, our algorithm enables a two times faster computation time when using dynamic refinement levels and a three times faster computation time than an algorithm based on the geometric features of the wafer topography.

Declaration of competing interest

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

Data availability

Data will be made available on request.

Acknowledgments

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.

References

[1] Hong P, Zhao Z, Luo J, Xia Z, Su X, Zhang L, et al. An improved dimensional measurement method of staircase patterns with higher precision in 3D NAND. IEEE Access 2020;8:140054–61. http://dx.doi.org/10.1109/ACCESS.2020.3012012.

[2] Zhou X, Tian P, Sher CW, Wu J, Liu H, Liu R, et al. Growth, transfer printing and colour conversion techniques towards full-colour micro-LED display. Progr Quantum Electron 2020;71:100263. http://dx.doi.org/10.1016/j.pquantelec.2020.100263.

[3] Sethian JA. Level set methods and fast marching methods: evolving interfaces in computational geometry, fluid mechanics, computer vision, and materials science. Cambridge, England: Cambridge University Press; 1999.

[4] Osher, Stanley J. and Sethian J. Fronts propagating with curvature dependent speed. J Comput Phys 1988;79(1):12–49. http://dx.doi.org/10.1016/0021-9991(88)90002-2.

[5] Sethian JA. A fast marching level set method for monotonically advancing fronts. Proc Natl Acad Sci 1996;93(4):1591–5. http://dx.doi.org/10.1073/PNAS.93.4.1591.

[6] Ertl O, Selberherr S. A fast level set framework for large three-dimensional topography simulations. Comput Phys Comm 2009;180(8):1242–50. http://dx.doi.org/10.1016/j.cpc.2009.02.002.

[7] Lenz C, Toifl A, Quell M, Rodrigues F, Hössinger A, Weinbub J. Curvature based feature detection for hierarchical grid refinement in TCAD topography simulations. Solid State Electron 2022;191:108258. http://dx.doi.org/10.1016/j.sse.2022.108258.

[8] Silvaco, victory process. 2022, URL www.silvaco.com/tcad/victory-process-3d/.

[9] Ertl O. Numerical methods for topography simulation (Ph.D. thesis), TU Wien; 2010.

[10] Zhang S, Gong Z, McKendry JJ, Watson S, Cogman A, Xie E, et al. CMOS-controlled color-tunable smart display. IEEE Photonics J 2012;4:1639–46. http://dx.doi.org/10.1109/JPHOT.2012.2212181.