Available online at www.sciencedirect.com

ScienceDirect

Comput. Methods Appl. Mech. Engrg. 386 (2021) 114196

Computer methods in applied mechanics and engineering

www.elsevier.com/locate/cma

Geometric advection and its application in the emulation of high aspect ratio structures

Xaver Klemenschits*, Siegfried Selberherr, Lado Filipovic

Institute for Microelectronics, TU Wien, Gußhausstraße 27-29/E360, Vienna, 1040, Austria

Received 16 March 2021; received in revised form 11 September 2021; accepted 13 September 2021
Available online xxxx

Abstract

The ability to simulate the processes required to fabricate advanced microelectronic structures, commonly referred to as process technology computer aided design (TCAD), is essential for the semiconductor industry. It aids in the design and development of modern integrated circuits at low cost. Ongoing demands of high efficiency and simplicity lead to the development of process emulation techniques, which describe the effect of a fabrication step by using only the geometric properties gathered from experimental data or extracted from physical simulations. However, the accuracy of current emulation models is limited by their strong reliance on explicit surface representations, while traditional algorithms employing implicit surfaces, such as the level set, can frequently suffer from numerical limitations. A novel geometric advection algorithm for microelectronic process emulation is presented, which advances a level set surface over large distances in a single simulation step. It combines the advantages of explicit surface representations, such as fast access to surface data and efficient sorting, and the robustness of implicit surface representations. Furthermore, calculations are performed directly in the level set avoiding costly conversions to explicit surfaces. The performance and applicability of the algorithm to modern processes is demonstrated by emulating the fabrication of several high aspect ratio structures. Due to the almost universal applicability and computational efficiency of the presented geometric advection algorithm, it provides an entirely new way of emulating microelectronic fabrication processes and presents a promising alternative to long-standing and commonly used techniques.
© 2021 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

Keywords: Level set; Process emulation; Geometric advection; Deep reactive ion etching; Design-technology-co-optimization

1. Introduction

The simulation of fabrication processes with process technology computer aided design (TCAD) has become an important tool in the design technology co-optimization (DTCO) cycle used for developing modern microelectronic devices. As these devices become smaller, more capable simulation methods and models are required in terms of accuracy and performance [1,2]. Due to the complex deformations of material interfaces taking place during fabrication, the ability to describe such surfaces robustly and efficiently is essential. Hence, implicit surface representations, such as the LS, are often used as they can robustly describe the separation and merger of material

* Corresponding author.
E-mail addresses: klemenschits@iue.tuwien.ac.at (X. Klemenschits), selberherr@iue.tuwien.ac.at (S. Selberherr), filipovic@iue.tuwien.ac.at (L. Filipovic).

https://doi.org/10.1016/j.cma.2021.114196
0045-7825/© 2021 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

interfaces [3]. Unlike explicit surface representations, implicit methods do not require additional re-meshing steps, making them computationally more efficient [4], which is one of the major concerns for simulations. Due to its robustness and good performance, the LS method has become a standard practice in modern process TCAD to describe the moving material interfaces during the fabrication of advanced microelectronic devices [5,6].

Physical process simulations are oftentimes highly complex and require considerable knowledge about the underlying physics of a process in order to describe the effect they have on the different materials and their interfaces [7]. However, frequently the effect of a process is sufficiently understood through empirical measurements, while the underlying mechanics and surface chemistry might not be fully resolved [8]. In these cases, simply applying the geometric effects of a process can sufficiently describe the resulting geometry with high accuracy. Such models, based solely on geometric considerations found in experiments, are called process emulation models as they do not fully simulate a real process in time, but rather only emulate its outcome [9]. Due to the nature of process emulation models, it is more straight-forward to implement them using explicit surfaces, because these allow for the movement of surfaces and interfaces over large distances. However, several properties of explicit surfaces, such as the need for re-meshing [10] and the discretization of simulation space into voxels [11], make them undesirable for the emulation of many types of microelectronic fabrication processes. Therefore, a solution is desired which can be applied over a broad range of geometric surface descriptions.

In this manuscript, a novel algorithm is presented, combining the advantages of implicit and explicit surface representations to advance a LS surface over large distances in only a single simulation step. Therefore, process emulation models can be applied highly efficiently and performed directly in the LS, without the need for conversions to other surface representations. The presented algorithm can be included easily in workflows using the most common process simulators, as they rely heavily on the LS method and no additional conversion is necessary. As a result, the presented algorithm can be used to generate process-aware initial surfaces for subsequent intricate physical process simulations using standard tools [5,6]. Furthermore, the implementation of this algorithm is described, as provided in ViennaLS [12], an open-source sparse LS library tailored towards microelectronic process simulation.

Several high-aspect ratio structures, which are fabricated using deep reactive ion etching (DRIE)-like processes, are generated with this method. Especially the simulation of DRIE processes may require considerable computational effort using traditional methods, due to the large number of processing cycles which have to be simulated. Since DRIE processes consist of several deposition and etch cycles which are sensitive to small changes, numerical errors stemming from the iterative modeling of discrete time steps may have strong effects on the final geometry.

The presented algorithms avoid these numerical challenges altogether, by essentially drawing the final surface using only geometrical considerations. By describing the geometric effect a process has on a geometry in a generalized way using so-called geometric advection distributions, any modeled process may be applied to any initial structure, producing the expected final geometry with sub-grid accuracy and fast simulation times. The algorithm therefore achieves the accurate modeling of process steps with a comparable computational efficiency as simply drawing the resulting structure. Therefore, subsequent physical process simulations which are to be investigated on complex geometries, such as thin film deposition over the scalloped DRIE-generated sidewalls, can be carried out while avoiding the inclusion of numerical errors introduced in the creation of the scalloped structure.

2. The level set method

The level set method for representing surfaces implicitly relies on storing the signed distance function (SDF) $\phi(\vec{x})$ on a regular grid. The SDF returns the distance of point $\vec{x}$ from the surface S of a material M with its sign denoting whether $\vec{x}$ is inside or outside of M:

$$\phi(\vec{x}) = \begin{cases} -d, & \text{for } \vec{x} \in M \\ 0, & \text{for } \vec{x} \in S \\ d, & \text{for } \vec{x} \notin M \end{cases} \tag{1}$$

It is therefore straight-forward to identify whether a point $\vec{x}$ is inside or outside of M by examining the sign of $\phi(\vec{x})$, where the convention of negative values referring to $\vec{x}$ being inside of M is used here. From Eq. (1), it is clear that all points which are part of the surface S must satisfy $\phi(\vec{x}) = 0$. The set of these points is thus called the zero LS of $\phi(\vec{x})$ and represents the explicit location of S in the LS method.

Numerically, $\phi(\vec{x})$ is represented by storing its value at each point $\vec{p}$ of a regular grid. However, this requires large amounts of memory, because the function needs to be defined everywhere in space and therefore, the memory requirement scales with the volume of the simulation domain, rather than the area of the represented surface, which would be the ideal case. This ideal scaling is only reached by explicit surface representations [13], but methods exist which are able to reduce the memory footprint of implicit representations, such as the narrow band method [14]. The least memory-intensive method is the sparse field LS [15], which is implemented in ViennaLS as described in the next section.

2.1. The sparse field level set

In order to mitigate the large memory requirements of the LS method, it is possible to only store specific points on the full grid, which are close to the surface. Since each grid point has a distance of unity to all of its neighbors, its signed distance can be generated from its neighbor points. Therefore, only points close to S actually hold relevant information about the surface location. These points are referred to as active points $\vec{a}$ and here the convention that $|\phi(\vec{a})| \leq 0.5$ is used. The number of active points scales with the surface area and is therefore close to the optimal solution in terms of memory consumption. All other level set values can be generated from the values of their neighboring points. However, common advection schemes described in the following [16], oftentimes require several neighbor points to advect a LS. The presented algorithm does not have such requirements and, therefore, a sparse field LS is sufficient, without the need for utilizing additional neighbor points.

2.2. Level set advection

In order to model fabrication processes in the semiconductor industry, the movement of surfaces over time due to chemical reactions has to be modeled. Moving a surface described by a LS is called advection and the change of the SDF over time is described by a Hamilton–Jacobi equation, called the LS equation:

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

$V(\vec{x},t)$ is a scalar velocity field, describing how the LS value of each point $\vec{x}$ changes over time. Many numerical finite difference schemes are available to solve Eq. (2), differing in complexity and performance. Simple schemes, such as the Engquist–Osher scheme, require less computational effort, but introduce numerical errors [17]. This might lead to the introduction of unphysical geometries, such as flattened corner geometries during deposition [18]. More complex schemes, such as the Lax–Friedrichs scheme [19], produce more accurate results, but are restricted to certain types of velocity functions $V(\vec{x},t)$ or data structures of the LS [20]. Due to these restrictions, they produce good results for one application, while introducing large numerical discrepancies in others with non-negligible computational overhead compared to simple schemes [21]. These problems also introduce a barrier for usage, as detailed knowledge about the specific process and the mathematical nature of its velocity function is required to choose the most appropriate advection scheme.

However, no matter what scheme is chosen for the solution of the LS equation, the Courant–Friedrichs–Lewy (CFL) condition must be satisfied [22]. In the case of the LS, it governs how much a single LS value can be changed, without first updating all other LS values as well. It therefore limits how far the surface can be moved with one solution of Eq. (2). For a general LS, this maximum distance, according to the CFL condition, is given by [19]:

$$\max|\phi(\vec{x},t+\Delta t) - \phi(\vec{x},t)| \leq 1 \quad , \tag{3}$$

where the left-hand side of this equation is equal to the second term of Eq. (2), the Hamiltonian of the LS equation. It follows that the surface must not be moved more than one grid spacing in a single simulation step.

However, in the case of a sparse field LS, only grid points with absolute values lower than 0.5 are stored. Therefore, the maximum value an active point $\vec{a}$ may have after advection is 1.5, meaning that the surface is no longer located between this active point and one of its neighbors. Hence, the new set of active points cannot be generated from the old set of active points. Thus, the maximum change applied to a single LS value should be smaller for sparse field level sets and is given by [23]:

$$\max|\phi(\vec{x},t+\Delta t) - \phi(\vec{x},t)| \leq 0.5 \quad . \tag{4}$$

In practice, Eqs. (3) and (4) are satisfied by lowering the time increment Δt. If a fabrication step moves the surface by several grid spacings, several iterations must be performed in order to advance the simulation time enough to correspond to the duration of the applied process. Each iteration requires that all LS values are updated and the LS equation is solved anew, resulting in increased computational effort. Additionally, numerical errors introduced in an early step propagate and often result in large discrepancies after several iterations, even though the numerical errors introduced by an advection scheme may have been small at first, creating the need for additional, computationally expensive error correction [24].

More complex advection schemes often need to satisfy even stronger restrictions than the CFL condition [25], meaning that the LS equation must be solved more often. Therefore, the choice of the numerical scheme used to solve Eq. (2) is usually a trade-off between accuracy and computational efficiency. Due to the lack of a universally robust numerical scheme and the propagation of errors, this type of advection, referred to as iterative advection, is not commonly used in the emulation of fabrication processes [26].

3. Geometric advection

Moving a surface described by a LS requires solving the LS equation numerous times, which leads to substantial computational effort and the introduction of numerical errors [27]. However, during process emulations, where the final result depends only on geometric considerations, the new surface can essentially be “drawn” and does not need to be generated by solving the LS equation. This method is referred to as geometric advection.

This approach has already been applied successfully with voxel-based material representations for several years [26,28]. Usually, voxels would be defined on a regular grid, just like the LS, and given a numerical identifier to denote which material they represent. However, each voxel is usually only set to filled or empty without the possibility for partially filled voxels [29], as this would highly complicate the surface representation and lead to increased computational effort [30]. A sample deposition process can be modeled using only the distances between initial voxels and empty voxels which might become part of the new material. A visual explanation of isotropic deposition modeling is provided in Fig. 1. For each voxel which has a neighboring empty voxel, a circle is drawn. All empty voxels which are inside this circle are set to filled. After this is performed for all circles, a new material is created which mimics an isotropically deposited layer. Fig. 1 already highlights some of the disadvantages of this voxel approach. The resolution is limited by the size of the voxels, meaning that materials with sub-grid thicknesses and smooth corners cannot be represented appropriately [31]. Furthermore, if a voxel’s center is not inside any circle, it will be marked as empty, even if most of its volume is inside the circles. Therefore, the surface of the new material is not always advanced as far as it should have been and these errors, stemming from the limited accuracy of the voxels, are propagated to the next processing step. These inaccuracies, while offering computational efficiency and simplicity, make the voxel approach unfavorable for many applications [32].

3.1. Geometric distributions

In order to develop a model which can emulate a real process, the properties of this process must be expressed in terms of the geometric effect it has on the initial surface to generate the final surface. The form in which this effect, independent of the initial surface, is expressed is referred to as geometric distribution. One of the simplest processes which can be emulated this way is isotropic deposition of one material on top of another. The required geometric distribution to model such a process has already been presented in Fig. 1 and is simply a circle or a sphere in two-dimensional (2D) or three-dimensional (3D) representations, respectively. A spherical distribution implies that the material is growing equally in all spatial directions. The radius of the sphere gives the growth distance of the material during the modeled process time. Therefore, every geometric distribution can be expressed as a function of spherical angles θ_{inc} and ϕ_{az}: $D(\theta_{inc}, \phi_{az})$. In the case of isotropic deposition, this function is constant in all directions.

$$D_{iso}(\theta_{inc}, \phi_{az}) = R_{depo} \tag{5}$$

R_{depo} is the growth distance of the isotropic deposition process. If the growth distance is not the same in every direction, the geometric distribution will have a non-spherical shape. There are no limitations to the shape of a geometric distribution other than it being single-valued in every direction. This must be the case since $D(\theta_{inc}, \phi_{az})$ describes by which distance the initial surface is advected in each direction, which can only be one single value. For

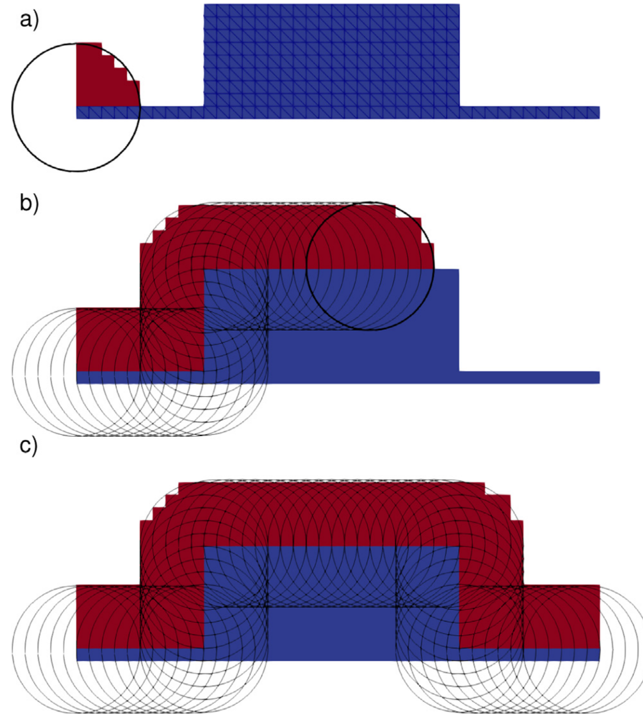

Fig. 1. Geometric advection in a voxel-based material representation: (a) For a voxel on the surface of the initial material (blue), a circle is drawn. All previously empty voxels inside this circle are filled with new material (red). (c) Circles are drawn for all voxels along the initial surface and a new material (red) is created. (c) After all circles have been filled, the new material (red) resembles the result of isotropic deposition. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

some distributions it might be more natural to express the geometric distribution in terms of Cartesian coordinates. However, spherical coordinates were used here, because they are the most natural choice for a spherical distribution.

The voxel approach of drawing the geometric distribution centered at each surface point, shown in Fig. 1, can also be applied to the LS method, as we have shown in [33]. A LS describing the geometric distribution could be created for each surface point and their union applied to the initial surface using Boolean operations, which are easily applied using SDFs [34]. However, creating and combining many LSs would be inefficient, as it is also possible to compute the resulting LS values directly.

The computation of the signed distance d_s of a possible new surface point $\vec{P}_{cand}$, called candidate point, is independent of all other possible new surface points. Therefore, Fig. 2 shows what the d_s of $\vec{P}_{cand}$ would be, if only one point on the initial surface $\vec{P}_{cont}$, called contribute point, is considered, similarly to only using one voxel to draw each circle in Fig. 1. In this case, there is only one geometric distribution centered at $\vec{P}_{cont}$.

The distance vector $\vec{v}$ from $\vec{P}_{cont}$ to $\vec{P}_{cand}$ is then computed and used to find the growth distance $D(\vec{v})$ in that direction:

$$D(\vec{v}) = D\left(\theta_{inc}(\vec{v}), \phi_{az}(\vec{v})\right) \tag{6}$$

For simplicity, the dependence of $\vec{v}$ on $\vec{P}_{cont}$ and $\vec{P}_{cand}$ is not written explicitly. The difference between this growth distance and the length of $\vec{v}$ gives the new signed distance value d_s for the point $\vec{P}_{cand}$:

$$d_s\left(\vec{v}, D(\vec{v})\right) = |\vec{v}| - D(\vec{v}) \tag{7}$$

Therefore, d_s only depends on the distance and direction from the original surface, but not on the absolute position. Since the LS value $\phi(\vec{x})$ is always normalized to the grid spacing Δx, the new LS value at the grid point $\vec{P}_{cand}$, generated from the initial surface point $\vec{P}_{cont}$, is given by

$$\phi\left(\vec{P}_{cand}, \vec{P}_{cont}\right) = \frac{d_s\left(\vec{v}, D(\vec{v})\right)}{\Delta x} \quad . \tag{8}$$

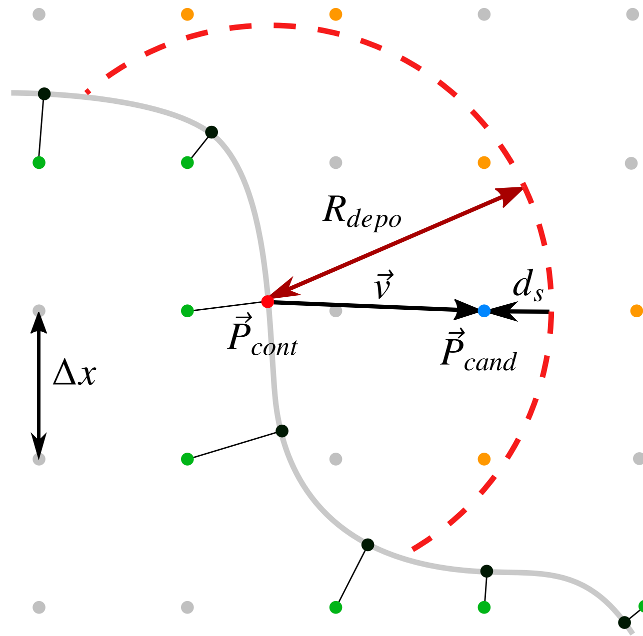

Fig. 2. Calculation of the signed distance d_s for one candidate point $\vec{P}_{cand}$ (blue) of the set of candidate points (orange) from one point on the original surface $\vec{P}_{cont}$ (red). The isotropic geometric distribution with growth distance R_{depo} is shown as a dashed red line and the grid spacing is Δx. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

In case of isotropic deposition, the LS value is thus

$$\phi_{iso}\left(\vec{P}_{cand}, \vec{P}_{cont}\right) = \frac{d_{s,iso}(\vec{v}, D_{iso})}{\Delta x} = \frac{|\vec{v}| - R_{depo}}{\Delta x} \quad , \tag{9}$$

which depends only on the distance from the original surface $|\vec{v}|$, as one would obviously expect for isotropic growth.

3.2. The geometric advection algorithm

The purpose of the presented algorithm is to find the smallest signed distance using Eq. (7) for a specific grid point $\vec{P}_{cand}$ and one $\vec{P}_{cont}$ out of the set of all initial surface points. This signed distance is then used to compute the new LS value $\phi(\vec{P}_{cand})$ for this grid point. Once the LS values for all relevant grid points have been computed, they automatically form a robust and complete LS, describing the final surface based on the given geometric distribution. The most important steps are depicted in the flowchart in Fig. 3.

The following listing describes these steps in detail, referring to Fig. 4 as a visual guide.

1. Generate explicit points on the initial surface. As shown in Fig. 4a, this is achieved by shifting active points (green) along the direction of the surface normal by their signed distance value [35]. The resulting point cloud (black) consists of points directly on the explicit surface, each separated from its neighbors by roughly one grid spacing. Therefore, a geometric distribution can be centered on each of them with the union of distributions robustly forming the final surface.
2. Identify which grid points are likely to be active LS points of the final surface, highlighted as orange and blue points in Fig. 4a. The smaller the number of these candidate points, the fewer time-consuming calculations of d_s have to be performed. Therefore, identifying the minimum number of candidate points is key to a performant algorithm.
3. Repeat the following for each candidate point:
   (a) Similarly to the number of candidate points, the number of contribute points should be small to avoid unnecessary calculations of d_s. Therefore, the grid points which lie within the bounding box of the

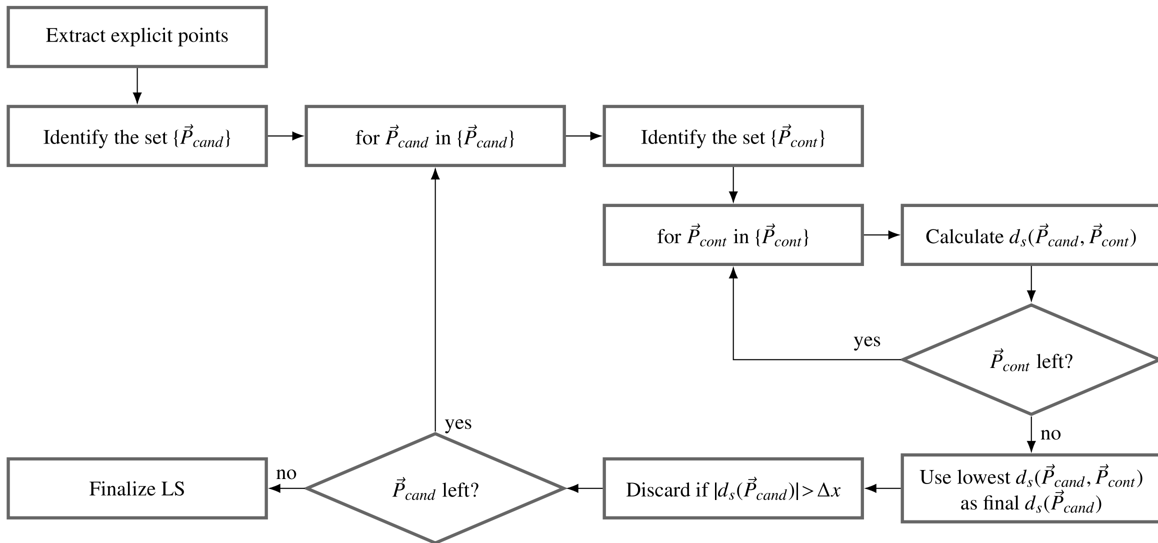

Fig. 3. Flowchart of the full geometric advection algorithm for a deposition process.

geometric distribution are chosen as the set of contribute points. The bounding box is defined as the Cartesian extent of the geometric distribution and is represented with a dashed blue line in Fig. 4b. Although this set of contribute points is not minimal, this approach can be applied using any geometric distribution and is nevertheless computationally highly efficient, as grid points can be identified solely by checking their coordinates without any further calculations.

(b) As shown in Fig. 4c, each contribute point and the current candidate point are used to find $d_s(\vec{P}_{cand}, \vec{P}_{cont})$ with Eq. (7). Then, the lowest of these values is selected as the final LS value for the current $\vec{P}_{cand}$, because the lowest value represents a point which is closest to the final surface. If $|d_s(\vec{P}_{cand})| > \Delta x$, the candidate point is discarded, as it will not form a part of the final sparse field LS.

4. The final values of all not discarded candidate points already form a valid LS. This LS still includes values larger than 0.5, as shown in Fig. 4d. However, values above 0.5 are not strictly required for sparse field LSs and are used in the presented framework only to minimize numerical errors in areas of high surface curvature. Therefore, a valid LS could also be constructed only from the new active points $\phi(\vec{a}) \leq 0.5$, albeit a potentially less robust one.

3.2.1. Differences between deposition and etching

As presented, the algorithm only applies to deposition, as the criterion for choosing the correct final d_s in Item 3b of the algorithm is only correct for outwards movements of the surface, because outwards movement of a LS means, that all LS values become smaller. Hence, choosing the smallest d_s leads to the correct result. However, when moving the surface inwards, as is the case during etching, all LS values become larger, which means the largest d_s must be chosen in Item 3b to yield the correct result. Therefore, it must be defined in advance, whether the emulated process is entirely a deposition process or entirely an etching process. In the presented implementation, this is indicated by the sign of the supplied geometric distribution, where a negative sign indicates etching. Therefore, a process which includes both etching and deposition cannot be modeled in a single geometric advection step. However, such a process can still be emulated using two separate geometric advection steps, one to describe etching followed by a second one to describe deposition.

3.2.2. Masked geometric advection

Lithography and etching are the corner stones of complementary metal oxide semiconductor (CMOS) fabrication processes [36]. It is imperative to be able to etch defined regions of a surface, not covered by a mask, and to model

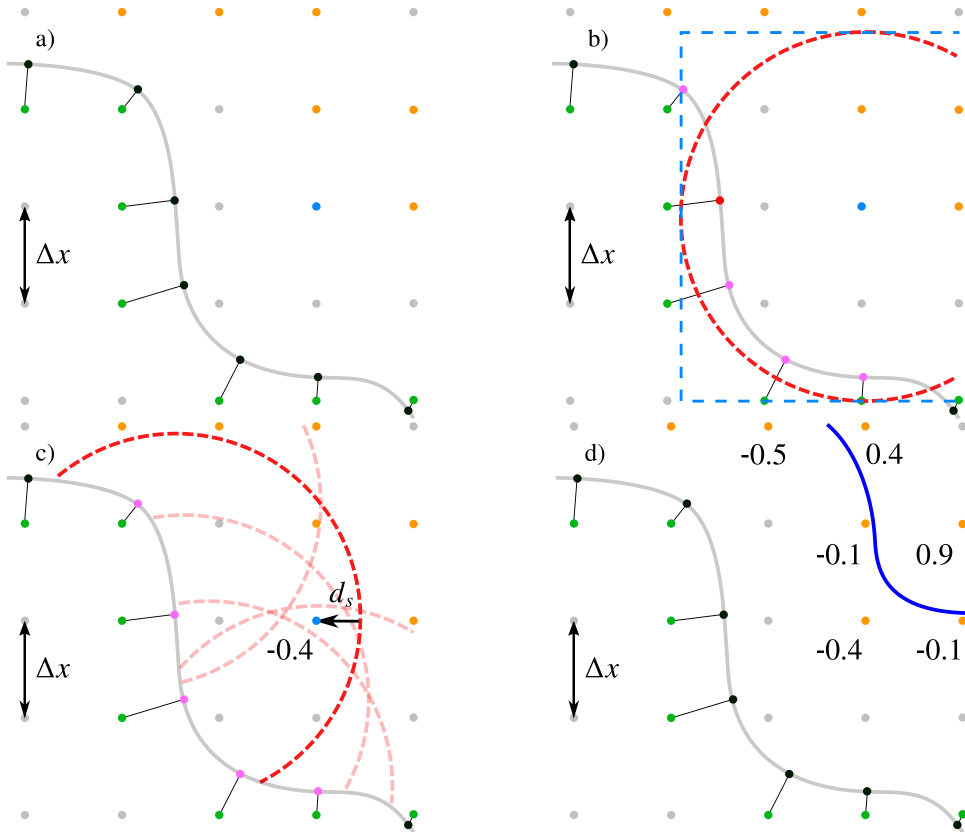

Fig. 4. Visualization of how the algorithm generates a complete LS from the old surface points: (a) Generate surface points (black) using active grid points (green). Identify candidate points (orange) and pick one for which the signed distance has not yet been calculated (blue). (b) Identify points which could contribute (pink) to the current candidate point (blue) using a bounding box (dashed blue). (c) Calculate d_s (see Eq. (7)) for all contribute points (pink) and choose the lowest value for the candidate point (blue). (d) After d_s has been calculated for all candidate points (orange), the new LS surface is complete. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

such processes accordingly. The presented algorithm does not have to be adjusted to produce correct results with masks. An additional LS which describes the mask material is simply passed to the algorithm. If this mask covers a certain area of the initial LS, the contribute points under the mask are simply ignored and never used for the signed distance calculation. Since some initial surface points are ignored during the algorithm, gaps left by discarded contribute points may appear in the final surface. These gaps are closed by including all grid points describing the mask material as already accepted candidate points in the final surface.

In order to guarantee numerical stability, the mask LS must be wrapped by the initial level set, meaning that the latter must in fact be a Boolean union of the initial surface and the mask material. The LS representation would otherwise produce unphysical voids or numerical discrepancies, especially at points where three different materials meet [37]. The initial surface passed to the algorithm must thus satisfy:

$$M_{\text{initial}} = M_{\text{initial}} \cup M_{\text{mask}} \tag{10}$$

Internally, the geometric advection algorithm will then extract the correct contribute points of the initial surface and perform the geometric advection. Since the mask must be included in the final surface as described above, the final LS also satisfies the layer wrapping approach leading to a robust multi-material description.

3.2.3. Limitations

The only limitation of this algorithm stems from the logic of choosing the correct d_s as the final value for a candidate point. Since the criterion for choosing the correct value is different for etching and deposition, only one

can be performed at a time. Therefore, processes which etch a material, but also deposit material in other areas, such as certain types of ion-enhanced plasma etching processes [38], cannot be modeled in one step. However, it is possible to split the emulation of those processes into two geometric advection steps, where the first is used for etching and the second for deposition, leading to the expected result.

3.2.4. Future improvements

In the described implementation, the entire simulation domain is considered for the identification of candidate points. Therefore, the entire dense grid must be checked to find all candidate points. This is a generic approach which is applicable to all processes and geometries. However, since the geometric relationship between the initial and final surface is known, a more efficient approach could be envisaged. If a single point of the final surface can be identified, it can be used to march along the final surface since the entire zero LS must be connected via grid points. Starting from one final surface candidate point, the d_s values of its neighboring grid points are calculated and the neighbors discarded, if they satisfy $|d_s| > \Delta x$. Non-discarded neighbor points are then used to find the next neighbors and the process is repeated. Once no neighbor grid points remain, all final surface points will have been visited and their final LS values will have been set.

4. Deep reactive ion etching

Since the introduction of the cyclic Bosch-Process in 1994 [39], DRIE has steadily gained importance and has now become crucial in the fields of 3D integration [40,41], high bandwidth memory [42,43], and even on-chip wireless communication [44], in addition to remaining a backbone of its original field of development, microelectromechanical systems (MEMS) [45,46]. DRIE is predominantly performed using cyclic processes, where sidewall passivation and etching are separated and applied many times to achieve higher aspect ratios than attainable by conventional plasma etching approaches [47]. Due to the isotropic nature of the etch step, these cyclic processes result in scallops on the sidewall, one for each etch cycle, which heavily influence the final geometry.

Next, the general modeling approach for this type of process will be discussed. Thereafter, several models for the description of modern DRIE processes are presented.

4.1. Modeling approach

In order to model this type of cyclic process using the geometric advection algorithm, the entire process including all cycles is split into two steps: First, a smooth profile of the final geometry is created, and in a second geometric advection step, the scallops are introduced. In process emulation, there is no need to account for the entire process including the deposition step over time. Therefore, these two steps suffice to generate the final structure. If there is a need to also represent the leftover polymer still present from the deposition steps, a subsequent deposition step can be added to generate this additional material.

4.1.1. Generation of the smooth profile

In order to generate an outline of the final profile, a directional geometric distribution is used, which will mimic a perfectly anisotropic etch. This can be achieved by using a rectangular distribution with the vertical edge length L equal to the final depth. The horizontal edge lengths must at least be equal to the grid spacing, since there might otherwise be some grid points of the final surface, which are not part of any distribution and might thus lead to unphysical voids in the final geometry. For minimal lateral etching, the horizontal edge lengths of the directional distribution are thus set equal to the grid spacing. Therefore, unmasked areas of the initial surface will simply be moved downward by the specified vertical edge length L, which is given by the product of the etch depth per cycle d_c and the number of cycles N_c for an ideal process:

$$L = N_c d_c \tag{11}$$

However, the final profile may also include tapering, due to a decreasing etch rate. Here, tapering is modeled by reducing L depending on the lateral proximity to the mask. Therefore, $L(\vec{x})$ depends on the coordinate of the contribute point around which it is centered. The specific dependence varies for different processes. We focus on positive tapering, as it is one of the predominant undesirable effects in DRIE. However, negative tapering can also be modeled [48]. Positive tapering means that the sidewalls are etched less and less with etch depth and $L(\vec{x})$ is smaller closer to the masked regions, i.e. closer to the sidewalls.

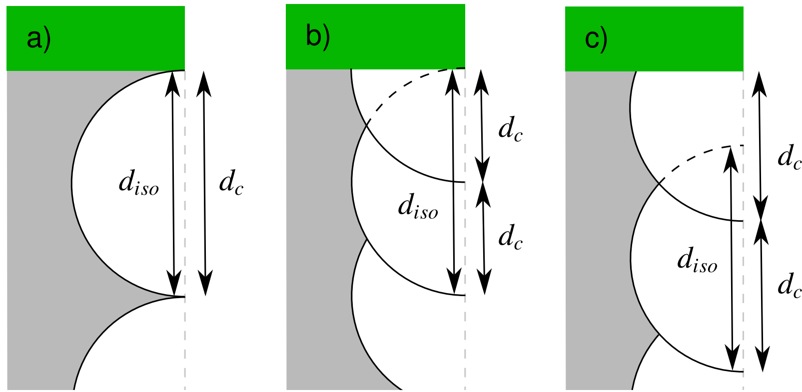

Fig. 5. Scallops for different values of f_t: (a) $f_t = 1$ corresponding to perfect free molecular transport, (b) $f_t = 2$ corresponding to perfect diffusion-limited etching, and (c) $f_t = 1.5$ corresponding to a combination of both transport phenomena.

4.1.2. Generation of scallops

The smooth profile obtained from the described procedure is used as the starting geometry to generate the final profile. The scallops characteristic for a Bosch Process are created by spherical geometric distributions, like the one shown in Fig. 2. To form such scallops, the radii of all distributions are set to zero, except those which are at a height, where a scallop is formed by the process. Therefore, the radii of the geometric distributions will vary with height, but are constant in all other directions. If there is no tapering, the vertical coordinate at which the nth scallop should be formed is therefore simply at:

$$z_n = d_c(n + 1/2) \quad for \quad n \in \mathbb{Z} \quad . \tag{12}$$

The diameter of a circular geometric distribution centered at any of the points z_n will be set to the isotropic etch depth of one etch cycle d_{iso}. d_{iso} can be measured by examining the depth and shape of the under etch of one scallop and is closely related to d_c. For simplicity, d_{iso} may be expressed as a fraction of d_c as

$$d_{iso} = f_t d_c \quad , \tag{13}$$

where $1 \leq f_t \leq 2$.

As shown in Fig. 5, a value of f_t of 1 corresponds to perfect free molecular transport, meaning that etchant molecules react with the surface instantly on impact without any movement on the surface. A value of 2 corresponds to perfect diffusion-limited transport to the etching site, meaning that particles first adsorb onto the surface and move along it freely to later react potentially far away of the initial point of impact. In reality, a combination of both phenomena is active, where molecules can move on the surface before reacting, but only for a short period of time. Therefore, $f_t \approx 1.5$ is a common value used for modern plasma etching processes [49,50].

In the given description, the different methods of particle transport to the etching site are discussed, but it has been assumed that the etching is indeed perfectly isotropic. Therefore, a spherical advection distribution could be used to describe each scallop. However, in modern SF_6 plasma chemistries lateral etching is likely slower than vertical etching. Again, it is useful to express this property as a fraction of the two etch rates

$$f_l = \frac{R_l}{R_v} \quad , \tag{14}$$

where R_l is the lateral etch rate and R_v is the vertical etch rate. In order to include slower lateral than vertical etching in the model, the spherical geometric advection distribution used for isotropic etching must be altered to a spherical cap shape resembling a convex lens, as shown in Fig. 6. This is performed by adjusting Eq. (9) so that the center section of the isotropic distribution is removed. This can be achieved most robustly by translating the vector $\vec{v}$ by the width of the removed center part of the sphere, as indicated in Fig. 6

$$\phi_{lens}(\vec{v}, f_l) = \frac{|\vec{v} + (1 - f_l) r_{lens} \overrightarrow{\mathrm{sgn}}(\vec{v}_i)| - r_{lens}}{\Delta x} \quad , \tag{15}$$

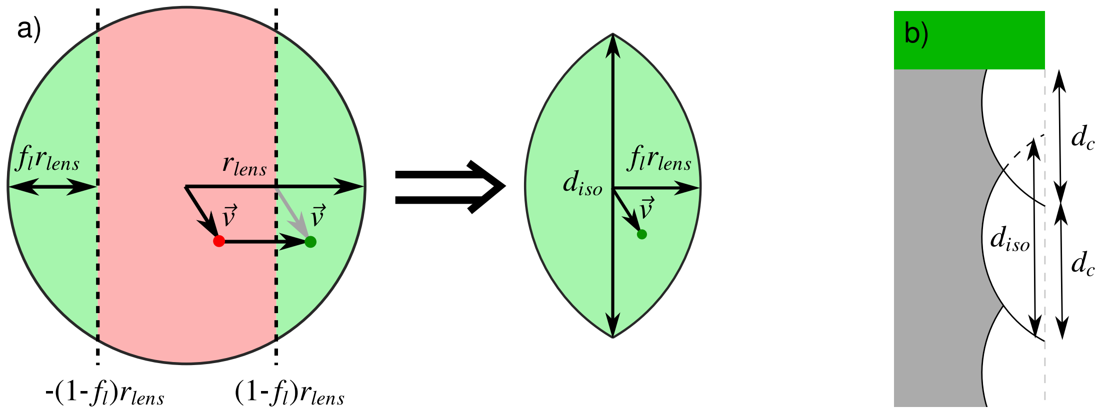

Fig. 6. (a) Starting from a spherical distribution, a lens distribution can be created by removing the center part. The vector $\vec{v}$ is translated to the remaining parts. (b) Using this lens distribution, a difference in vertical and lateral etch rate can be modeled for $f_t = 1.5$ and $f_l = 0.5$.

where i is either 2 or 3 for a two- or three-dimensional description, respectively, with the Cartesian vector $\vec{v}_i$ being either $\vec{v}_2 = (v_x, 0)$, or $\vec{v}_3 = (v_x, v_y, 0)$. $\overrightarrow{\text{sgn}}(\vec{x})$ is the element-wise sign function of $\vec{x}$ which returns a vector where each element is given by the signum function of the corresponding element in $\vec{x}$.

r_{lens} is the adjusted radius of the distribution, which normalizes the width of the spherical cap, so that the ratio of lateral to vertical extent of the final distribution is equal to f_l. The value of r_{lens} is given by:

$$r_{lens} = \frac{d_{iso}}{2} \begin{cases} 1 & if \quad f_l \geq 1 \\ \left(1 - f_l^2\right)^{-1/2} & if \quad 0 \leq f_l < 1 \\ undefined & if \quad f_l < 0 \end{cases} \tag{16}$$

Graphically, this adjustment can be visualized as two spherical caps, which are obtained by simply removing the center section of a unit sphere in both x- and y-direction between $-(1 - f_l)$ and $(1 - f_l)$, moving the slices back to the origin, and scaling by r_{lens}, as shown in Fig. 6.

4.2. DEM sequence

The original Bosch Process employed this sequence, which consists of one deposition step and one plasma etch step and is therefore referred to as the **D**eposit-**E**tch-**M**ultiple-Times (DEM) sequence. This results in scalloped sidewalls, while good sidewall protection and very straight profiles can be achieved, reducing the tapering significantly compared to plasma etching processes. However, for aspect ratios above 50, the simple 2-step process suffers from reactive ion etching (RIE) lag [51,52], which means that the etch rate decreases with increasing etch depth, until no more etching is feasible at the trench bottom. At the final feature depth, the etch step perfectly balances the deposition step and therefore no additional material is removed thereafter. This effect can be balanced by increasing etch times for later cycles, but due to prolonged etch times and comparably low mask selectivity, the mask can be worn away quickly [53].

4.2.1. Smooth profile

As described in [54], the DEM process leads to decreasing etch rates for large aspect ratios, due the depletion of etchant at the feature bottom [55]. Therefore, the sidewall will be slanted outwards. For simplicity, we assume that the decrease in etch rate is linear down the feature. In order to generate the smooth profile, one must consider the number of cycles affected by the reduced etch rate. The number of cycles influenced by the reduced etch rate N_t is defined as

$$N_t = N_c - N_s = N_c - \frac{L_t}{d_c} \quad , \tag{17}$$

where N_c is the total number of cycles, N_s is the number of scallops in the straight profile and L_t is the depth at which the tapering starts.

The positive tapering can then be modeled by reducing the vertical edge length $L(\vec{x})$ for contribute points closer to the mask, according to

$$L(\vec{x}) = \min\left(L_t + \frac{|\vec{x} - \vec{m}|}{w_t}(L_0 - L_t), L_b\right) \quad . \tag{18}$$

where w_t is the tapering width at the bottom of the feature and $\vec{m}$ is the point on the mask surface, which is nearest to $\vec{x}$. If the process is carried out for a long time, the depth at which etching would stop due to the etch rate going to zero is given by L_0. It is therefore a property of the etch process, because this specific depth depends on the exact materials and etch chemistry used. L_b is the depth of the trench after the final etch cycle is carried out in the specific experiment to be simulated and is calculated using

$$L_b = L_t + z_{N_t}(d_c, D) \quad , \tag{19}$$

where z_{N_t} is the distance of the final scallop from the start of tapering which only depends on d_c and D which is the distance along which the etch depth per cycle decreases from its initial value to zero. The value of D is computed numerically (see SI Appendix) and depends only on the number of tapered cycles N_t and r_e, which is the ratio of the etch depth of the last cycle d_{N_c} and the etch depth of the first cycle d_1.

$$r_e = \frac{d_{N_c}}{d_1} = 1 - \frac{w_t}{w_{tot}} \tag{20}$$

The tapering width at the bottom of trench w_t is effectively the lateral distance between the tapered sidewall and the straight sidewall at the bottom of the structure. w_{tot} is the final tapering width which would result from applying infinitely many cycles, i.e. the tapering width, when the etch rate reaches zero. The value of w_{tot} is a property of the specific process and must be tuned to the etch chemistry being modeled [56].

In order to include aspect ratio dependent etching (ARDE) [57], w_{tot} and L_t can be modified in accordance with the size of each etched feature. However, in the present model all features were the same size, meaning w_{tot} could just be treated as a constant.

Therefore, only the total number of cycles N_c, the etch rate per cycle d_c, and the tapering width w_t are required as inputs to the model which will create the smooth etch profile for a linearly decreasing etch rate. The tapering width, when the etch rate reaches zero (w_{tot}), and the depth at which tapering starts (L_t) are used to calibrate the model to a desired chemistry.

4.2.2. Generation of scallops

If tapering is included in the emulation, the coordinates z_n are equally spaced, until tapering starts. From this point onward, the etch depth per cycle d_c is assumed to decrease linearly, meaning the distance between the centers of two neighboring scallops decreases according to a linear progression, as detailed in the SI Appendix. The values of z_n are calculated using

$$z_n = \begin{cases} d_c n & if \quad n \leq N_s \\ d_c N_s + a\frac{1-b^{n-N_s}}{1-b} & if \quad n > N_s, \end{cases} \tag{21}$$

where a and b are defined as:

$$a = \frac{d_c}{1 + \frac{d_c}{2D}} \tag{22}$$

$$b = \frac{1 - \frac{d_c}{2D}}{1 + \frac{d_c}{2D}} \tag{23}$$

The radii of the geometric distributions should also be adjusted accordingly, meaning that d_{iso} will also linearly decrease. The diameter of the nth geometric distribution is therefore given by:

$$d_n = d_{iso} \begin{cases} 1 & if \quad n \leq N_s \\ 1 - (z_n - L_t)/D & if \quad n > N_s \end{cases} \tag{24}$$

4.3. DREM sequence

An improvement of the DEM sequence, which separates the plasma etching step into two independent processes, was proposed in [58]. First, an etching step at low pressure and high bias is carried out to remove the protective passivation at the bottom of the feature [47]. The second step subsequently etches the unprotected substrate at the bottom isotropically. Therefore, this sequence is referred to as **D**eposit-**R**emove-**E**tch-**M**ultiple-Times (DREM). This separation leads to shorter etch times at high bias and therefore higher mask selectivity [59]. The problem of RIE lag still remains, but due to the higher mask selectivity, it can be solved by increasing only the isotropic etch times in later cycles. This so-called time ramping can therefore produce uniform scallops all the way down the etched feature, when tuned correctly [60].

For high aspect ratios, one problem still remains from the deposition of the protective layer. Due to the pileup of the polymer at the top of the feature, the top opening starts to close off, reducing the number of high energy ions which are able to enter the feature and remove the protective film at the bottom [59]. Therefore, the opening at the bottom of the feature also decreases in size, while the etch depth of the isotropic etch remains the same due to time ramping. This results in positively tapered sidewalls and scallops which are constant in size.

4.3.1. Smooth profile

The tapering in this process is modeled similarly to the tapering in the DEM sequence. The vertical edge length of the box distribution used to form the smooth profile is the same as for the DEM sequence and given by Eq. (18). However, the definition of the parameters L_0 and L_b changes due to the different mechanisms leading to tapering. Since the scallops themselves do not change in size, the bottom of the trench L_b is expressed as

$$L_b = L_t + d_c N_t = d_c N_c \quad , \tag{25}$$

while L_0 is given by

$$L_0 = L_t + D = L_t + \frac{d_c N_t}{w_t / w_{tot}} \quad . \tag{26}$$

w_t therefore determines how strongly the closing of the top of the feature affects the removal of protective polymer at the bottom of the trench. If $w_t = 0$, there is no effect at all, and if $w_t \geq w_{tot}$ the feature is pinched off entirely.

4.3.2. Generation of scallops

Since the scallops are constant in size down the feature, they all have the same diameter d_{iso}. However, if the time ramping of the isotropic etch is not tuned properly, the DEM model must be used to include decreasing scallop sizes.

4.4. DREAM sequence

Although the pileup of polymer at the top of the feature in the DREM sequence can be avoided by carefully tuning the deposition and remove steps, it is time consuming and therefore impractical. To overcome this problem, an additional ashing step is introduced, which selectively removes the protective polymer and thus prevents the closing of the feature top opening [61], leading to the **D**eposit-**R**emove-**E**tch-**A**sh-**M**ultiple-Times (DREAM) sequence. Employing this sequence, tapering can be avoided entirely and good scallop uniformity can be achieved. The tapering width therefore only depends on the ash step [49]. If this step is not effective, for example by being too short, the feature is not cleaned properly. In this case, the time ramping of the isotropic etch step is not tuned properly to the closing top of the structure any more. Therefore, tapering occurs in a manner similar to the DEM sequence. In order to build a model describing this behavior, the ratio of final to initial etch depth per cycle r_e can be related directly to the duration of the ash step.

From the experimental data in [49] and [61], a simple model describing the change of r_e with ash time was developed. Assuming N_c and L_t remain constant with changing ash time, r_e can be related to ashing time by

$$r_e(t_a) = p_0 - \frac{p_1}{p_2 + t_a} \quad , \tag{27}$$

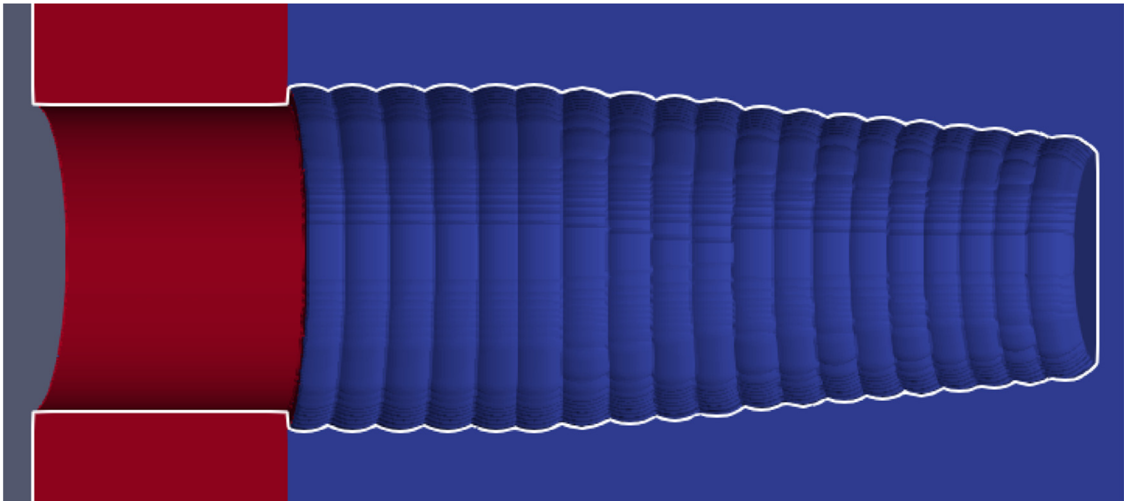

Fig. 7. Three-dimensional geometric DEM Process with $N_c = 19$ and other parameters listed in Table 1. The process was emulated using 32 threads and the simulation time was 44.6 s. Volume meshes for the mask and substrate are shown in red and blue, respectively. The white line is a slice of the final surface extracted from the LS. The final sparse-field LS surface is represented by 277,684 LS values. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

where t_a is the ashing time in seconds. Fitting this model to the experimental data, the model parameters were found to be $p_0 = 1.17$, $p_1 = 0.59$, $p_2 = -0.44$. Since r_e is bound by [0, 1], negative values are treated as 0 and values larger than unity are treated as 1. Therefore, ashing only has an effect if the ash time is longer than $t_0 = p_1/p_0 - p_2$. Similarly, if the ash time exceeds the maximum value of $t_m = p_1/(p_0 - 1) - p_2$, no additional benefit is achieved.

5. Results and discussions

All simulations were performed using the minimal resolution necessary for the LS to correctly represent all features, because there is no resolution limitation of the geometric advection algorithm itself. Therefore, comparably high resolutions (finer grids) might be necessary to resolve smaller and smaller scallops, while lower resolutions (coarser grids) were used for processes with almost uniform scallop sizes.

5.1. DEM sequence

In order to demonstrate the capabilities of the DEM model and the geometric advection in general, a Bosch Process was emulated, including sidewall tapering as well as decreasing scallop sizes. As can be seen in Fig. 7, all features are generated as expected. Although the resolution of the LS itself is just enough to represent the smaller scallops, there are no numerical issues introduced by the algorithm, as opposed to commonly encountered problems with iterative schemes discussed earlier [24].

The applicability of the DEM model to state of the art processes is demonstrated by reproducing a structure from [59], including the observed reduction in scallop size. The experimental observation and the emulated structure are shown in Fig. 8. Although a simple linear decrease in scallop size was assumed, a good match to the observed more complex depletion behavior has been achieved. The fitting parameters and their values of the DEM model are listed in Table 1. The model produces robust and predictable structures for any initial surface and any number of cycles. Therefore, once the model has been fitted to a process, the process specific effects on different geometries can be straightforwardly investigated. Additionally, the effect of changing the number of etch cycles on different structures, such as additional tapering, decrease in etch rate, etc. can be investigated providing that the etch chemistry remains unchanged.

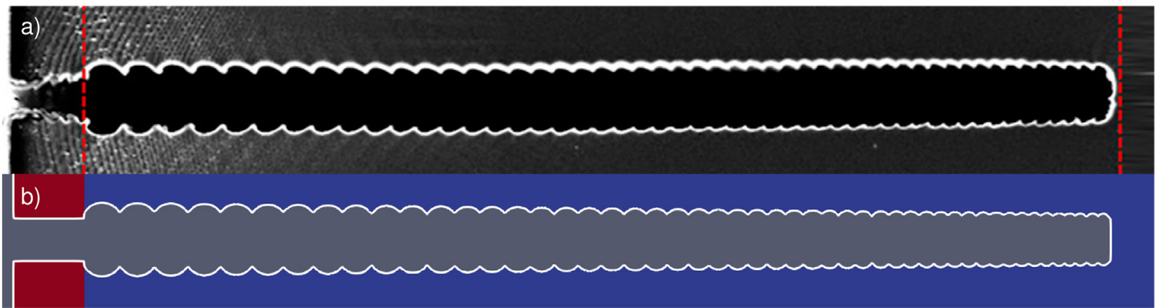

Fig. 8. (a) DEM process without tapering (reprinted from [59]), but a decrease in scallop sizes down the feature. (b) Emulation of the same process with $N_c = 50$.

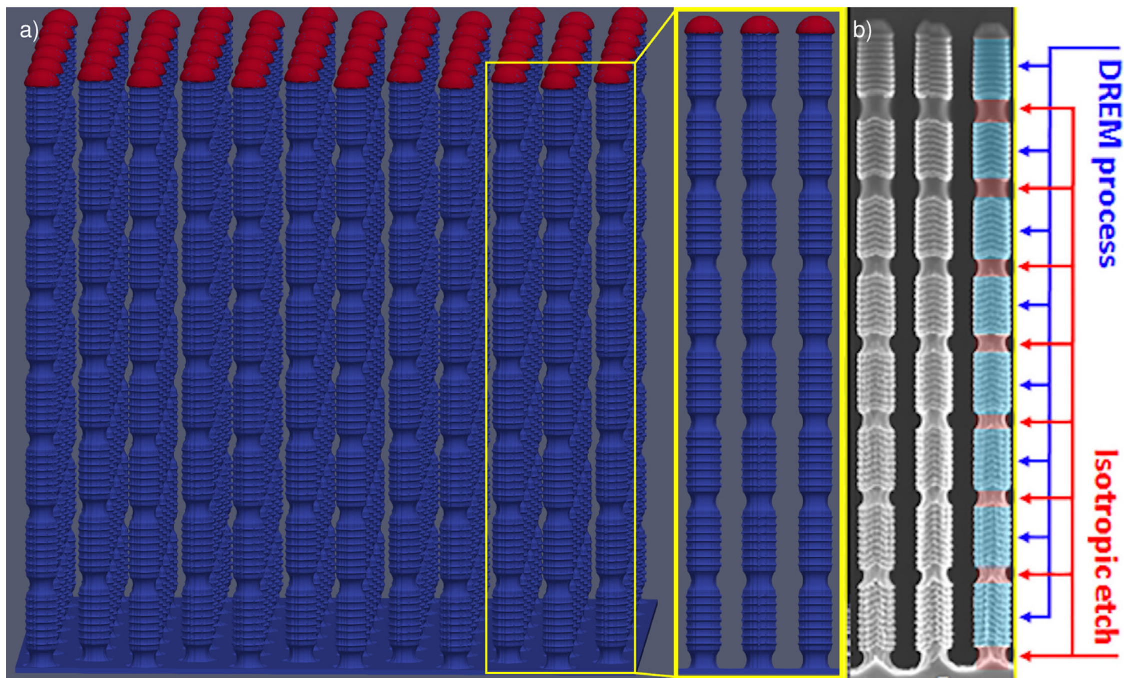

Fig. 9. (a) Emulation of sausage-chain like structures using several DREM sequences, as well as longer intermittent isotropic etch steps. (b) SEM image of the sausage-chain pillars generated by the DREM process, reprinted from [59]. The emulation was performed using 32 threads and took 14 min and 52 s. 80 cycles of the DREM model were performed without tapering using the fitting parameters given in Table 1. In order to generate the isotropic etch parts, every 10th scallop's isotropic etch depth was increased to $d_{iso} = -1.0$.

5.2. DREM sequence

As an example of the ability to emulate the fabrication of large structures efficiently, the sausage-chain-like pillars from [59] were emulated. This structure is generated by performing 8 cycles of a DREM sequence followed by an isotropic etch step. In the emulation, these steps were repeated 8 times to generate pillars with a sausage-chain-like appearance. The final three-dimensional structure shown in Fig. 9 consists of 72 pillars described by 3,827,322 LS points. The ability to generate such process-aware structures enables TCAD to fully model process-induced properties and their influence on final device performance. Furthermore, the modeling of such complex processes for large structures opens the possibility for the process-aware simulation of entire MEMS actuators [62] or photonic crystals [63], and therefore provides a strong basis for progress in several scientific fields.

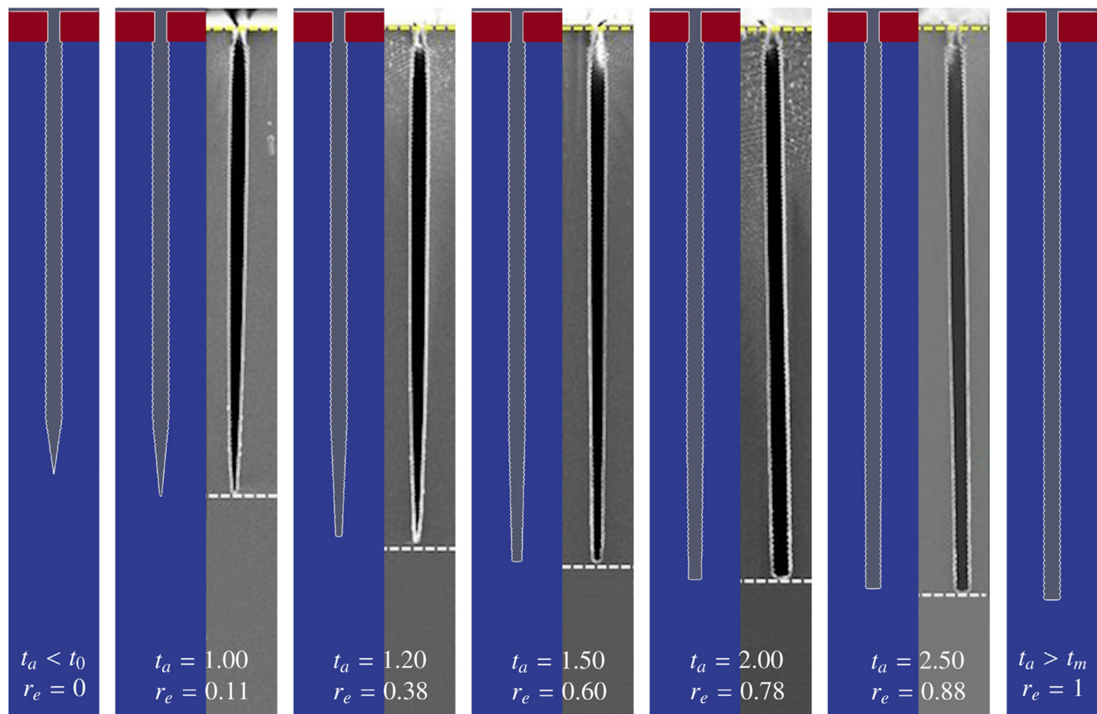

Fig. 10. DREAM Sequence simulations for $N_c = 100$ with different ash step times compared to SEM data, reprinted from [49]. Only the ash time was varied as input for the DREAM model, all other parameters were fixed with values shown in Table 1.

Table 1
Fitting parameters of all presented process emulations.

	L_t [μm]	d_c [μm]	f_t	f_l	r_e
DEM 3D	−10.00	−1.86	1.15	0.50	0.70
DEM 2D	0.00	−0.98	1.20	0.75	0.30
DREM	–	−0.25	1.20	0.50	1.00
DREAM	−24.96	−0.37	1.20	0.50	$r_e(t_a)$ (see Eq. (27))

5.3. DREAM sequence

When the ash time in the model of the DREAM process approaches zero, the DEM model is recovered. As the ash time reaches its maximum value, the DREAM model approaches the results of the DREM model. Therefore, model parameters can be related directly to the ash time to generate a full description of the DREAM sequence. In order to confirm the robustness of the model, a sequence of structures was emulated with ash times ranging from zero to the maximum ash time, shown in Fig. 10. Even though the model is quite simple, it produces accurate structures matching the experimental data well. This model highlights how powerful the geometric advection algorithm is for the quick development of compact models which can be used for subsequent device simulations. Additionally, the algorithm is as accurate as the LS itself and does not introduce additional numerical errors common to iterative processes. It can therefore be executed at lower resolutions, saving time, and computational effort, compared to iterative advection schemes, while avoiding undesirable effects, such as stepped surfaces, compared to voxel-based approaches. Furthermore, due to the small resolution requirements, it is possible to simulate large structures and even entire integrated circuit cells quickly without the loss of process specific features, such as changing scallop sizes and sidewall tapering.

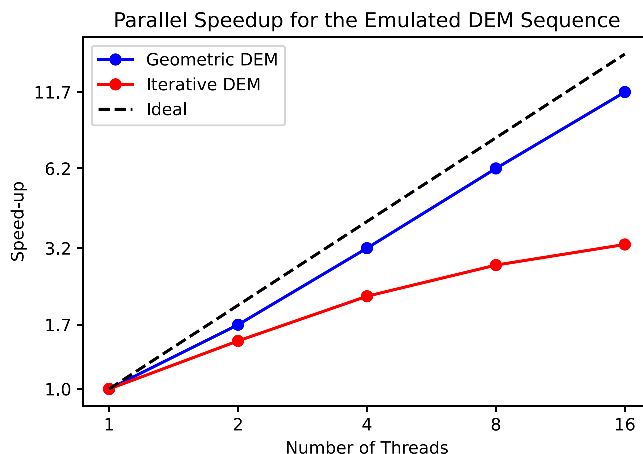

Fig. 11. Speed-up of the parallelized geometric advection algorithm emulating the DEM process shown in Fig. 7 compared to speed-up of the same emulation carried out using an iterative advection scheme.

5.4. Parallel performance

As the geometric advection algorithm offers new possibilities for the simulation of large structures, the scaling behavior was investigated and compared to the commonly used iterative Engquist–Osher scheme [17]. This was carried out by comparing the run time speed-up of the DEM sequence emulation shown in Fig. 7 for the different schemes. Since every point of the advected surface can be treated completely independently of all others, the geometric advection algorithm scales very well, especially compared to the iterative scheme, as shown in Fig. 11. The loss in performance of iterative schemes stems from the fact that the independently advected parts of the LS must be joined again. This involves additional computational effort which cannot be parallelized. The geometric advection, however, is complete once all points have been computed and no additional joining step is necessary, leading to almost ideal scaling behavior.

6. Conclusion

In order to develop devices and integrated circuits with ever smaller dimensions and higher overall performance, designers rely heavily on simulations. However, the inclusion of process specific properties of devices has remained a challenge due to the involved computational effort required by TCAD. Process emulations are one promising way to reduce computing times and still provide insights into the effects of process specific geometries which affect electric properties. Currently available methods for emulations are predominantly based on explicit surface representations less robust than implicit methods. Our algorithm is a new take on process emulation and combines the computational efficiency of explicit surface representations and the robustness of implicit surfaces, thereby avoiding stepped interfaces as encountered with the commonly used voxel-based emulation approach. It therefore provides an entirely new platform upon which to build more efficient and more accurate process emulation models. Additionally, as the algorithm is executed entirely in the LS, it can be integrated easily into common process simulation tools which predominantly rely on LS surface descriptions. Hence, complex process-aware initial structures for intricate physical simulations can be generated quickly. Therefore, the presented algorithm enables the investigation of fundamental physical principles of fabrication processes carried out on complex process-aware initial structures.

The applicability of the geometric advection algorithm to modern fabrication processes was demonstrated by providing several DRIE process emulation models. Emulating recent DRIE sequences using only simple models, already shows good agreement with experimental data, providing accurate descriptions of the process' effects with considerably less computational effort than traditional methods. Due to the algorithms' performance, it is also possible to emulate large structures at high accuracies with reasonable runtimes. This was shown by creating a sausage-chain like pillar structure using a DREM sequence with intermittent isotropic etch steps. Due to the large size of this particular structure, it was shown that even entire devices can be emulated within reasonable time frames,

providing a better description of their structure and thus their electrical and physical properties. The presented algorithm therefore provides a strong basis for the future of process emulations, helping device engineers to create improved integrated circuits by incorporating process-aware variations and parasitics when simulating and designing devices.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work was supported in part by the Austrian Research Promotion Agency FFG (Bridge Young Scientists) under Project 878662 "Process-Aware Structure Emulation for Device-Technology Co-Optimization".

Appendix. Derivations behind the equations used in the DRIE models

Derivation of scallop circle centers and etch depth

As discussed in [51,52], the etch depth per cycle d_c decreases with trench or via depth from a certain distance down the feature, which we define as L_t. For simplicity, we assume that this decrease is linear. Scallops are modeled as spheres spaced out along the feature sidewalls. These spheres must touch each other at the extremes and, since their radii are linearly decreasing, the spacing between the centers of two circles must also decrease linearly. The depth at which the circle radii are zero is L_0. Therefore, the distance along which the circle radii change from the initial value d_c to zero is:

$$D = L_0 - L_t \tag{A.1}$$

For simplicity, the origin of z is defined at L_t, so the first tapered circle is centered at $z_0 = 0$. Therefore, the distance between the nth and (n+1)th circle center is given by:

$$z_{n+1} - z_n = \left(1 - \frac{z_{n+1}}{D}\right)\frac{d_c}{2} + \left(1 - \frac{z_n}{D}\right)\frac{d_c}{2} \tag{A.2}$$

Hence, the centers of all circles are defined by the recursive relation

$$z_{n+1} = \frac{d_c}{1 + \frac{d_c}{2D}} + z_n \frac{1 - \frac{d_c}{2D}}{1 + \frac{d_c}{2D}} = a + b z_n \quad , \tag{A.3}$$

with the initial condition

$$z_0 = 0 \quad . \tag{A.4}$$

Since each z_{n+1} is defined by a geometric progression, z_n can also be expressed in terms of a, b and n only:

$$z_n = a\frac{1 - b^n}{1 - b} \tag{A.5}$$

If the last scallop should have an etch rate of zero, the depth L_0 could just be specified and the radii of the circles used as geometric distributions would just decrease until they reach zero at L_0. However, it is possible that not enough etch cycles are performed to actually reach an etch rate of zero, since each cycle achieves less and less depth. Therefore, it is useful to only specify the number of cycles and the etch depth of the first and last cycle. We define the ratio of final etch depth to starting etch depth as:

$$r_e = \frac{d_f}{d_c} = 1 - \frac{w_t}{w_{tot}} \quad , \tag{A.6}$$

where d_f is the final etch depth per cycle and d_c the initial etch depth per cycle. If the data stems from the physical understanding of a process or a physics-based model, r_e is the ratio of the etch rates at the top and bottom of the feature. If data is acquired from experimental measurements, this ratio can also be defined using the ratio of

measured tapering width w_t and the tapering width created if an infinite number of cycles were performed w_{tot}. In the case of a trench or via, w_{tot} can go up to half of the trench width or the radius of the via. If the number of tapered etch cycles N_t is known, r_e can be used to calculate D, defined by the expression:

$$1 - r_e = \left(1 - \left(\frac{1 - \frac{d_c}{2D}}{1 + \frac{d_c}{2D}}\right)^{N_t}\right)\left(1 + \frac{d_c}{2D}\right) \tag{A.7}$$

A solution for D is found numerically using a root-finding algorithm. The presented implementation uses the bisection method as it is very robust and a solution has to be found only once, so performance was not a major concern.

The value of D is then used in Eq. (A.3) to find a and b, which are then used in Eq. (A.5) to find the circle centers. The deepest point of the final feature L_b can therefore be calculated using:

$$L_b = L_t + z_{N_t} = L_t + a\frac{1 - b^{N_t}}{1 - b} \tag{A.8}$$

DREAM sequence model

The ashing step of the DREAM Sequence avoids the closing of the cavity at the top of the feature during the deposition cycles. The closing would lead to the inaccessibility of etchant to the bottom of the feature, thereby halting the etch process. As the diameter of the top opening decreases, the etchant concentration at the bottom of the trench decreases drastically, leading to retarded etch rates. As described in [64], the ratio of etchant concentration at the bottom to the top of a deep via is given by:

$$n_r = \frac{2e^{-h_T}}{1 + e^{-2h_T}} \quad , \tag{A.9}$$

where h_T is the Thiele modulus which describes how conformal a process is. Small values of h_T mean that a process is conformal, while large values mean that the etchant concentration varies down the feature. For a cylindrical via, h_T depends only on the via diameter d, the sticking probability of the etchant β and the current etch depth z:

$$h_T = \sqrt{3\beta}\frac{z}{d} \quad . \tag{A.10}$$

At around $h_T = 1$, the process transitions away from being conformal to decreasing etchant concentrations down the feature, which is exactly what happens when tapering starts. Approximating Eq. (A.9) around this value leads to:

$$n_r \approx sech(1)\,(1 - tanh(1)(h_T - 1)) \tag{A.11}$$

Since the etch rate at the bottom is directly proportional to the etchant concentration, the etch ratio r_e of a DREAM sequence with changing top opening diameter can be related directly to d:

$$r_e = n_r \propto -\frac{1}{d} \tag{A.12}$$

The diameter of the top opening decreases during the process, if the ash step is not effective and too much passivating material is left after each cycle. As the pileup on the top opening is the same each cycle, d decreases linearly with the number of cycles. Assuming the removal of passivation through ashing proceeds with a constant rate, the diameter of the top opening d is directly proportional to the ash time t_a. Therefore, the final model for the etch ratio at the bottom of the feature based on the ash time is given by:

$$r_e(t_a) = p_0 - \frac{p_1}{p_2 + t_a} \quad , \tag{A.13}$$

where p_0, p_1, and p_2 are fitting parameters. They encompass all unknowns of the systems, such as the sticking probability of the etchant β, the constant of proportionality between the passivation removal and the ash time, or the initial time it takes for the ash process to actually start removing passivating material.

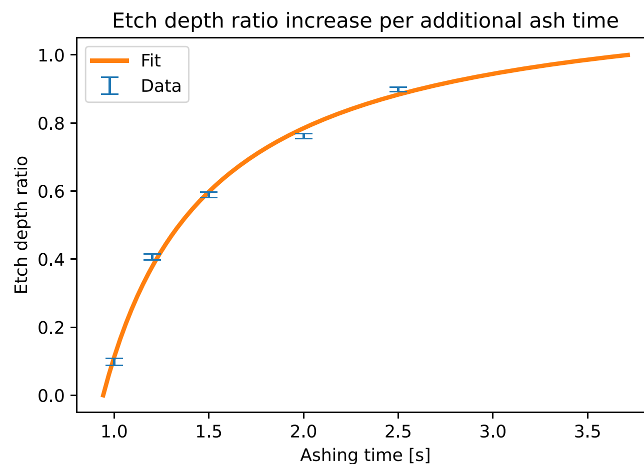

Fig. A.12. Fit of the DREAM Sequence model to experimental values of the etch depth ratio generate from trench depths for different ash times of the DREAM sequence presented in [61]. Experimental values are shown as blue error bars and the fitted model is shown as an orange line going from $r_e = 0$ to $r_e = 1$.

DREAM sequence model fitting

The model presented above was fit to experimental data for 100 cycles of the DREAM process in [61]. The first step in fitting the model was to determine after how many cycles the tapering starts, in order to identify L_t and thus N_t. L_t is found by considering where r_e starts decreasing, i.e. for which h_T the approximation in Eq. (A.11) returns values smaller than 1. The actual description in Eq. (A.9) is evaluated at this h_T and yields 0.96. Therefore, the tapering in the simple model in Eq. (A.11) should start at the depth where $r_e = 0.96$ in order to yield the best fit. This is the depth at which the via diameter is only 0.96 times the original width. From the data in [61] this depth was found to be $L_t = 24.96$ µm from the top of the first scallop, meaning that the first $L_t/d_c = 67$ cycles are not influenced by the tapering and the following $N_t = 33$ are. Subsequently, measurements of the etch depths were taken to find experimental values of D, which can be used to numerically solve Eq. (A.7) for the experimental values of r_e for each of the shown ash times t_a. These experimental values are shown in Fig. A.12 with the error bars indicating the error from the resolution of the images the measurements were taken from. The least squares fit is shown in orange and presents the final DREAM sequence model used for the presented emulations. The extracted model parameters were $p_0 = 1.17$, $p_1 = 0.59$, and $p_2 = -0.44$.

References

[1] International Roadmap for Devices and Systems (IRDS), 2020, [Online] Available: http://irds.ieee.org/.

[2] X. Klemenschits, S. Selberherr, L. Filipovic, Modeling of gate stack patterning for advanced technology nodes: A review, Micromachines 9 (12) (2018) 631–1–631–31, http://dx.doi.org/10.3390/mi9120631.

[3] N.P. van Dijk, K. Maute, M. Langelaar, F. van Keulen, Level-set methods for structural topology optimization: A review, Struct. Multidiscip. Optim. 48 (3) (2013) 437–472, http://dx.doi.org/10.1007/s00158-013-0912-y.

[4] B. Houston, M.B. Nielson, C. Batty, O. Nilsson, K. Museth, Hierarchical RLE level set : A compact and versatile deformable surface representation, ACM Trans. Graph. 25 (1) (2006) 151–175, http://doi.acm.org/10.1145/1122501.1122508.

[5] D. Dunn, J.R. Sporre, V. Deshpande, M. Oulmane, R. Gull, P. Ventzek, A. Ranjan, Guiding gate-etch process development using 3D surface reaction modeling for 7 nm and beyond, in: Proc. SPIE 10149, Advanced Etch Technology for Nanopatterning VI, 2017, 101490Q–1–101490Q–12, http://dx.doi.org/10.1117/12.2271389.

[6] A. Toifl, F. Rodrigues, L.F. Aguinsky, A. Hössinger, J. Weinbub, Continuum level-set model for anisotropic wet etching of patterned sapphire substrates, Semicond. Sci. Technol. 36 (4) (2021) 045016–1–045016–12, http://dx.doi.org/10.1088/1361-6641/abe49b.

[7] R.J. Belen, S. Gomez, M. Kiehlbauch, E.S. Aydil, Feature scale model of si etching in SF_6/O_2/HBr plasma and comparison with experiments, J. Vac. Sci. Technol. A 24 (2) (2006) 350–361, http://dx.doi.org/10.1116/1.2173268.

[8] O. Luere, E. Pargon, L. Vallier, B. Pelissier, O. Joubert, Etch mechanisms of silicon gate structures patterned in SF_6/CH_2F_2/Ar inductively coupled plasmas, J. Vac. Sci. Technol. B 29 (1) (2011) 011028–1–011028–10, http://dx.doi.org/10.1116/1.3522656.

[9] E. Parker, T. Udeshi, MEMulator: A fast and accurate geometric modeling, visualization and mesh generation tool for 3D MEMS design and simulation, in: Proc. NANOTECH, 2003, pp. 480–483.
[10] C. Chong, A. Senthil Kumar, H. Lee, Automatic mesh-healing technique for model repair and finite element model generation, Finite Elem. Anal. Des. 43 (15) (2007) 1109–1119, http://dx.doi.org/10.1016/j.finel.2007.06.009.
[11] C.M. Huard, Y. Zhang, S. Sriraman, A. Paterson, K.J. Kanarik, M.J. Kushner, Atomic layer etching of 3D structures in silicon: Self-limiting and nonideal reactions, J. Vac. Sci. Technol. A 35 (3) (2017) 031306–1–031306–15, http://dx.doi.org/10.1116/1.4979661.
[12] X. Klemenschits, O. Ertl, P. Manstetten, J. Weinbub, L. Filipovic, ViennaLS, 2021, [Online]. Available: https://github.com/ViennaTools/ViennaLS.
[13] M. Pauly, M. Gross, L.P. Kobbelt, Efficient simplification of point-sampled surfaces, in: Proc. IEEE Visualization, 2002, pp. 163–170, http://dx.doi.org/10.1109/VISUAL.2002.1183771.
[14] D.L. Chopp, Computing minimal surfaces via level set curvature flow, J. Comput. Phys. 106 (1) (1993) 77–91, http://dx.doi.org/10.1006/jcph.1993.1092.
[15] R.T. Whitaker, A level-set approach to 3D reconstruction from range data, Int. J. Comput. Vis. 29 (3) (1998) 203–231, http://dx.doi.org/10.1023/A:1008036829907.
[16] R.P. Fedkiw, G. Sapiro, C.-W. Shu, Shock capturing, level sets, and PDE based methods in computer vision and image processing: A review of Osher's contributions, J. Comput. Phys. 185 (2) (2003) 309–341, http://dx.doi.org/10.1016/S0021-9991(02)00016-5.
[17] B. Engquist, S. Osher, One-sided difference approximations for nonlinear conservation laws, Math. Comp. 36 (154) (1981) 321, http://dx.doi.org/10.1090/S0025-5718-1981-0606500-X.
[18] J. Sethian, D. Adalsteinsson, An overview of level set methods for etching, deposition, and lithography development, IEEE Trans. Semicond. Manuf. 10 (1) (1997) 167–184, http://dx.doi.org/10.1109/66.554505.
[19] S. Osher, C.-W. Shu, High-order essentially nonoscillatory schemes for Hamilton–Jacobi equations, SIAM J. Numer. Anal. 28 (4) (1991) 907–922, http://dx.doi.org/10.1137/0728049.
[20] B. Radjenović, J.K. Lee, M. Radmilović-Radjenović, Sparse field level set method for non-convex Hamiltonians in 3D plasma etching profile simulations, Comput. Phys. Comm. 174 (2) (2006) 127–132, http://dx.doi.org/10.1016/j.cpc.2005.09.010.
[21] A. Toifl, M. Quell, X. Klemenschits, P. Manstetten, A. Hössinger, S. Selberherr, J. Weinbub, The level-set method for multi-material wet etching and non-planar selective epitaxy, IEEE Access 8 (2020) 115406–115422, http://dx.doi.org/10.1109/ACCESS.2020.3004136.
[22] R. Courant, K. Friedrichs, H. Lewy, Über die partiellen differenzengleichungen der mathematischen physik, Math. Ann. 100 (1) (1928) 32–74, http://dx.doi.org/10.1007/BF01448839.
[23] O. Ertl, Numerical Methods for Topography Simulation (Ph.D. thesis), TU Wien, 2010, http://dx.doi.org/10.34726/hss.2010.001.
[24] T.F. Dupont, Y. Liu, Back and forth error compensation and correction methods for removing errors induced by uneven gradients of the level set function, J. Comput. Phys. 190 (1) (2003) 311–324, http://dx.doi.org/10.1016/S0021-9991(03)00276-6.
[25] S. Osher, R.P. Fedkiw, Level Set Methods and Dynamic Implicit Surfaces, Vol. 1, Springer, New York, 2005.
[26] P. Pfäffli, H. Wong, X. Xu, L. Silvestri, X. Lin, T. Yang, R. Tiwari, S. Mahapatra, S. Motzny, V. Moroz, T. Ma, TCAD modeling for reliability, Microelectron. Reliab. 88–90 (2018) 1083–1089, http://dx.doi.org/10.1016/j.microrel.2018.06.109.
[27] M.F. Trujillo, L. Anumolu, D. Ryddner, The distortion of the level set gradient under advection, J. Comput. Phys. 334 (2017) 81–101, http://dx.doi.org/10.1016/j.jcp.2016.11.050.
[28] Z. Tan, M. Furmanczyk, M. Turowski, A.J. Przekwas, CFD-Micromesh: A fast geometric modeling and mesh generation tool for 3D microsystem simulations, in: Proc. SPIE 4019, Design, Test, Integration, and Packaging of MEMS/MOEMSS, 2000, pp. 193–199, http://dx.doi.org/10.1117/12.382289.
[29] C.M. Huard, Y. Zhang, S. Sriraman, A. Paterson, M.J. Kushner, Role of neutral transport in aspect ratio dependent plasma etching of three-dimensional features, J. Vac. Sci. Technol. A 35 (5) (2017) 05C301–1–05C301–18, http://dx.doi.org/10.1116/1.4973953.
[30] M. Fujinaga, N. Kotani, 3-D topography simulator (3-D MULSS) based on a physical description of material topography, IEEE Trans. Electron Devices 44 (2) (1997) 226–238, http://dx.doi.org/10.1109/16.557710.
[31] E. Strasser, S. Selberherr, Algorithms and models for cellular based topography simulation, IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst. 14 (9) (1995) 1104–1114, http://dx.doi.org/10.1109/43.406712.
[32] Y. Zhang, C. Huard, S. Sriraman, J. Belen, A. Paterson, M.J. Kushner, Investigation of feature orientation and consequences of ion tilting during plasma etching with a three-dimensional feature profile simulator, J. Vac. Sci. Technol. A 35 (2) (2017) 021303–1–021303–16, http://dx.doi.org/10.1116/1.4968392.
[33] X. Klemenschits, S. Selberherr, L. Filipovic, Geometric advection algorithm for process emulation, in: Proc. SISPAD, 2020, pp. 59–62, http://dx.doi.org/10.23919/SISPAD49475.2020.9241678.
[34] B. Wyvill, A. Guy, E. Galin, Extending the CSG tree. Warping, blending and boolean operations in an implicit surface modeling system, Comput. Graph. Forum 18 (2) (1999) 149–158, http://dx.doi.org/10.1111/1467-8659.00365.
[35] O. Ertl, S. Selberherr, Three-dimensional topography simulation using advanced level set and ray tracing methods, in: Proc. SISPAD, 2008, pp. 325–328, http://dx.doi.org/10.1109/SISPAD.2008.4648303.
[36] C.A. Mack, Fundamental Principles of Optical Lithography: The Science of Microfabrication, Wiley-Blackwell, Hoboken, NJ, 2007.
[37] O. Ertl, S. Selberherr, A fast level set framework for large three-dimensional topography simulations, Comput. Phys. Comm. 180 (8) (2009) 1242–1250, http://dx.doi.org/10.1016/j.cpc.2009.02.002.
[38] R. D'Agostino, D.L. Flamm, Plasma etching of Si and SiO_2 in SF_6–O_2 mixtures, J. Appl. Phys. 52 (1) (1981) 162–167, http://dx.doi.org/10.1063/1.328468.
[39] F. Lärmer, A. Schlip, German patent DE4241045c1, 1992, US Patent 5501893.
[40] M. Kawano, X.-Y. Wang, Q. Ren, Trench isolation technology for cost-effective wafer-level 3D integration with one-step TSV, in: Proc. ECTC, 2020, pp. 1161–1166, http://dx.doi.org/10.1109/ECTC32862.2020.00186.

[41] P. Kumar, I. Dutta, Z. Huang, P. Conway, Materials and processing of TSV, in: 3D Microelectronic Packaging, Springer, Singapore, 2021, pp. 47–70, http://dx.doi.org/10.1007/978-981-15-7090-2_3.
[42] K. Bae, J. Park, Efficient TSV fault detection scheme for high bandwidth memory using pattern analysis, in: Proc. ISOCC, 2020, pp. 19–20, http://dx.doi.org/10.1109/ISOCC50952.2020.9333115.
[43] K.C. Chun, Y.K. Kim, Y. Ryu, J. Park, C.S. Oh, Y.Y. Byun, S.Y. Kim, D.H. Shin, J.G. Lee, B.-K. Ho, M.-S. Park, S.-J. Cho, S. Woo, B.M. Moon, B. Kil, S. Ahn, J.H. Lee, S.Y. Kim, S.-K. Choi, J.-S. Jeong, S.-G. Ahn, J. Kim, J.J. Kong, K. Sohn, N.S. Kim, J.-B. Lee, A 16-GB 640-GB/s HBM2E DRAM with a data-bus window extension technique and a synergetic on-die ECC scheme, IEEE J. Solid-State Circuits 56 (1) (2021) 199–211, http://dx.doi.org/10.1109/JSSC.2020.3027360.
[44] V. Pano, I. Tekin, I. Yilmaz, Y. Liu, K.R. Dandekar, B. Taskin, TSV antennas for multi-band wireless communication, IEEE J. Emerg. Sel. Top. Circuits Syst. 10 (1) (2020) 100–113, http://dx.doi.org/10.1109/JETCAS.2020.2974236.
[45] U. Soysal, F. Marty, E. Géhin, C. Motzkus, E. Algré, Fabrication, electrical characterization and sub-ng mass resolution of sub-μm air-gap bulk mode MEMS mass sensors for the detection of airborne particles, Microelectron. Eng. 221 (2020) 111190–1–111190–9, http://dx.doi.org/10.1016/j.mee.2019.111190.
[46] M. Liu, Y. Zhu, C. Wang, Y. Chen, Y. Wu, H. Zhang, Y. Du, W. Wang, A novel low-g MEMS bistable inertial switch with self-locking and reverse-unlocking functions, J. Microelectromech. Syst. 29 (6) (2020) 1493–1503, http://dx.doi.org/10.1109/JMEMS.2020.3032586.
[47] H.V. Jansen, M.J. de Boer, S. Unnikrishnan, M.C. Louwerse, M.C. Elwenspoek, Black silicon method: X. A review on high speed and selective plasma etching of silicon with profile control: An in-depth comparison between bosch and cryostat DRIE processes as a roadmap to next generation equipment, J. Micromech. Microeng. 19 (3) (2009) 033001–1–033001–41, http://dx.doi.org/10.1088/0960-1317/19/3/033001.
[48] M.K. Hooda, M. Wadhwa, S. Verma, M. Nayak, P. George, A. Paul, A systematic study of DRIE process for high aspect ratio microstructuring, Vacuum 84 (9) (2010) 1142–1148, http://dx.doi.org/10.1016/j.vacuum.2010.01.052.
[49] V. Thi Hoang Nguyen, C. Silvestre, P. Shi, R. Cork, F. Jensen, J. Hubner, K. Ma, P. Leussink, M. de Boer, H. Jansen, The CORE sequence: A nanoscale fluorocarbon-free silicon plasma etch process based on SF_6/O_2 cycles with excellent 3D profile control at room temperature, ECS J. Solid State Sci. Technol. 9 (2) (2020) 024002–1–024002–13, http://dx.doi.org/10.1149/2162-8777/ab61ed.
[50] B. Wu, A. Kumar, S. Pamarthy, High aspect ratio silicon etch: A review, J. Appl. Phys. 108 (5) (2010) 051101–1–051101–20, http://dx.doi.org/10.1063/1.3474652.
[51] D. Chin, S.H. Dhong, G.J. Long, Structural effects on a submicron trench process, J. Electrochem. Soc. 132 (7) (1985) 1705–1707, http://dx.doi.org/10.1149/1.2114195.
[52] D. Keil, E. Anderson, Characterization of reactive ion etch lag scaling, J. Vac. Sci. Technol. B 19 (6) (2001) 2082–2088, http://dx.doi.org/10.1116/1.1414116.
[53] K.J. Owen, B. VanDerElzen, R.L. Peterson, K. Najafi, High aspect ratio deep silicon etching, in: Proc. MEMS, 2012, pp. 251–254, http://dx.doi.org/10.1109/MEMSYS.2012.6170138.
[54] M.A. Blauw, T. Zijlstra, R.A. Bakker, E. van der Drift, Kinetics and crystal orientation dependence in high aspect ratio silicon dry etching, J. Vac. Sci. Technol. B 18 (6) (2000) 3453–3461, http://dx.doi.org/10.1116/1.1313578.
[55] W. Jacobs, A. Kersch, P. Moll, W. Sabisch, G. Icking-Konert, A feature scale model for trench capacitor etch rate and profile, in: Digest. IEDM, 2003, pp. 891–894, http://dx.doi.org/10.1109/IEDM.2002.1175980.
[56] N. Roxhed, P. Griss, G. Stemme, A method for tapered deep reactive ion etching using a modified Bosch process, J. Micromech. Microeng. 17 (5) (2007) 1087–1092, http://dx.doi.org/10.1088/0960-1317/17/5/031.
[57] S.L. Lai, D. Johnson, R. Westerman, Aspect ratio dependent etching lag reduction in deep silicon etch processes, J. Vac. Sci. Technol. A 24 (4) (2006) 1283–1288, http://dx.doi.org/10.1116/1.2172944.
[58] M.A. Blauw, G. Craciun, W.G. Sloof, P.J. French, E. van der Drift, Advanced time-multiplexed plasma etching of high aspect ratio silicon structures, J. Vac. Sci. Technol. B 20 (6) (2002) 3106–3110, http://dx.doi.org/10.1116/1.1518018.
[59] B. Chang, P. Leussink, F. Jensen, J. Hübner, H. Jansen, DREM: Infinite etch selectivity and optimized scallop size distribution with conventional photoresists in an adapted multiplexed bosch DRIE process, Microelectron. Eng. 191 (2018) 77–83, http://dx.doi.org/10.1016/j.mee.2018.01.034.
[60] B. Chang, F. Jensen, J. Hübner, H. Jansen, DREM2: a facile fabrication strategy for freestanding three dimensional silicon micro- and nanostructures by a modified bosch etch process, J. Micromech. Microeng. 28 (10) (2018) 105012–1–105012–10, http://dx.doi.org/10.1088/1361-6439/aad0c4.
[61] B. Chang, Technology Development of 3D Silicon Plasma Etching Processes for Novel Devices and Applications (Ph.D. thesis), Technical University of Denmark, 2018, URL https://orbit.dtu.dk/en/publications/technology-development-of-3d-silicon-plasma-etching-processes-for.
[62] S. Bertini, M. Verotti, A. Bagolini, P. Bellutti, G. Ruta, N. Belfiore, Scalloping and stress concentration in DRIE-manufactured comb-drives, Actuators 7 (3) (2018) 57–1–57–22, http://dx.doi.org/10.3390/act7030057.
[63] B. Chang, C. Zhou, A.T. Tarekegne, Y. Yang, D. Zhao, F. Jensen, J. Hübner, H. Jansen, Large area three-dimensional photonic crystal membranes: Single-run fabrication and applications with embedded planar defects, Adv. Opt. Mater. 7 (2) (2019) 1801176–1–1801176–9, http://dx.doi.org/10.1002/adom.201801176.
[64] A. Yanguas-Gil, Growth and Transport in Nanostructured Materials: Reactive Transport in PVD, CVD, and ALD, Springer, 2016, pp. 74–78, http://dx.doi.org/10.1007/978-3-319-24672-7.