INSTITUTE OF PHYSICS PUBLISHING JOURNAL OF MICROMECHANICS AND MICROENGINEERING

J. Micromech. Microeng. **13** (2003) 927–941 PII: S0960-1317(03)61658-X

Systematic mask synthesis for surface micromachined microelectromechanical systems

Venkataraman Ananthakrishnan[1], **Radha Sarma**[2] **and G K Ananthasuresh**[3]

[1] Pathway Technologies, Inc, Blue Bell, PA 19422, USA
[2] Department of Mechanical Engineering, The University of Michigan, Ann Arbor, MI 48105, USA
[3] Department of Mechanical Engineering and Applied Mechanics, University of Pennsylvania, Philadelphia PA 19104, USA

Received 1 April 2003, in final form 23 June 2003
Published 14 August 2003
Online at stacks.iop.org/JMM/13/927

Abstract
In the context of designing surface-micromachined microelectromechanical systems (MEMS), there does not appear to be systematic means, with the exception of parametrized layout models, to generate the mask data after the geometric model of a MEMS device is refined through behavioral simulations. This paper focuses on automatically generating masks, given a geometric model of the MEMS device and the process sequence (referred to here as the inverse problem). This necessitates a systematic solution of the forward problem, which involves automatically generating a geometric model of the MEMS device given the masks. A systematic and implementation-independent framework for the geometric modeling of MEMS is presented in order to solve the forward and inverse problems for general surface-micromachined devices. In particular, the geometric problem of mask synthesis is reduced to a system of linear equations.

(Some figures in this article are in colour only in the electronic version)

1. Introduction

Microelectromechanical systems (MEMS) are manufactured by adapting existing VLSI (very large scale integration) microelectronic fabrication technology that is used to fabricate integrated circuits (ICs). While IC fabrication uses relatively simple Manhattan geometries, the functionality of a MEMS device is highly dependent on its geometric sophistication. Currently, some common MEMS fabrication processes, which can achieve complex geometries, include bulk micromachining, surface micromachining and LIGA (Madou 1997, Fatikow and Rembold 1997). Bulk micromachining is a popular MEMS fabrication method, but the variety of shapes obtainable is restricted by the crystal structure of the substrate. LIGA is a relatively expensive process. Thus, of the three processes, surface micromachining has emerged as a versatile and affordable fabrication option (Sniegowski 1996) and is the process of choice in this paper. Henceforth, for brevity, ‘surface micromachining’ is referred to simply as ‘process’.

In surface micromachining, as is well known, MEMS devices are built up by the deposition and etching of layers of different materials. Each deposited layer can be shaped by etching selected regions of the layer via a photolithographic process. Photolithographic masks determine the regions to be etched. The material properties of each deposited layer can also be modified (wholly or partially through masks) by a process called doping. Any specific surface micromachining process (e.g., the MUMPs process or the Sandia SUMMiT process) comprises a fixed sequence of process steps made up of specific instances of the three generic sub-processes, namely deposits, etches and dopings (Madou 1997). Thus the degree of complexity of a MEMS device manufactured by a specific surface micromachining process depends on

(a) the number and variety of layers offered and (b) the different instances of deposits, etches and dopings supported by that process.

One of the primary differences between traditional macrofabrication processes and MEMS microfabrication processes (described above) is that the sequence of process steps in microfabrication processes are fixed. This is unlike macrofabrication processes where a multitude of intermediate process steps can be selected in almost any desired order. The main reason for the fixed sequence of steps in microfabrication processes is that it is very time consuming and expensive to fine-tune each process step to yield accurate, reliable and repeatable results. For the same reason, the thicknesses of deposits and the depths of etches and dopings are fixed for an instance of a process step. Hence, for a given surface micromachining process, a designer can only control the geometry and topology of the masks in order to tailor the geometry and topology of the MEMS device. Thus, given a surface micromachining process, the sophistication of MEMS devices can be largely attributed to the lithographic mask design.

1.1. Art-to-part problem

Traditionally, MEMS designers begin a new design by creating the masks (artwork) that would lead to a geometric model of the MEMS device, a practice carried over from VLSI design where there is a direct correlation between the device and mask geometry. This is referred to as the *art-to-part* design approach, which involves the solution of the forward problem, i.e. generation of a geometric model from the masks and a given process. At the macrolevel, this is analogous to generating a geometric model from the tool paths. Designing of masks before the geometry of a part is finalized through behavioral simulations is not straightforward (except in the most simple cases) because the designer needs to take into account the relative arrangement and dependencies of the layers leading to an intertwining of the device design with mask generation processes. Very often a single mask affects more than one layer thereby adding considerably to the design effort. In addition, the designer will not be able to directly visualize the MEMS device without virtually or physically constructing the device.

1.2. Part-to-art problem

In contrast to the traditional MEMS design process, designers of macrodevices have the advantage of starting with a geometric model and being able to directly visualize or manipulate their designs. The geometric model is then queried to generate the process specific data. Even if a geometric model of a MEMS device has been created using the art-to-part paradigm, there is no systematic means to generate the mask data after the geometric model of a MEMS device has been refined or optimized through behavioral simulations. *A part-to-art* paradigm, when applied to the design of MEMS devices, will thus substantially reduce the design effort involved. However, for this approach to be adopted for MEMS, it is necessary to have a procedure for automatically synthesizing mask data given the geometric model. This involves the solution of the *inverse problem*, i.e. the generation of masks from a geometric model of a MEMS device.

1.3. Scope of the paper

This paper focuses on the systematic and implementation-independent modeling of the inverse problem and the forward problem (to the extent it enables the systematic solution of the inverse problem) in three dimensions. The rest of this paper is structured as follows. Section 2 describes the literature relevant to the forward and inverse problems. In section 3 several process descriptors are outlined, followed by section 4 in which a process specification language is introduced. This process specification language is used to describe the forward problem in section 5 laying the foundation for the inverse problem. In section 6, a solution to the inverse problem is described. Concluding remarks are made in section 7.

2. Related work

The OYSTER project at IBM (Koppelman and Wesley 1983, Koppelman 1989) and the MEMCAD project at MIT (Senturia *et al* 1992) first recognized the need for computer-aided design (CAD) tools in the micromechanical realm. Emphasis was placed on creating process simulations and three-dimensional CAD models that could be used to predict the physical behavior of MEMS devices. In these integrated CAD environments, three-dimensional models were created from mask and process data (Senturia 1998).

Image processing and morphological operations were also used on three-dimensional cellular representations to simulate etches and deposits based on etch and deposition rates (Strasser and Selberherr 1995). MEMShapes, another three-dimensional simulator, uses solid modeling techniques to build models of MEMS devices (Dixit *et al* 1997). A process algebra was developed that captured all the geometric and material transformations occurring to the MEMS device as it is being fabricated.

While the above projects focused on general processes, other works focused on MEMS devices that were fabricated using a specific process. Given the masks, three-dimensional models of MEMS devices fabricated using a bulk micromachining process were generated using a crystal plane offset approach (Hubbard and Antonsson 1994) and a cellular automata approach (Hubbard and Antonsson 1997). The problem of finding the masks for a bulk-etched single layer using multiple etchants was attempted using genetic and evolutionary algorithms (Li and Antonsson 1998, Lee and Antonsson 2000, Ma and Antonsson 2000).

Several projects related to process modeling are also relevant and include the SUPREM project at Stanford (Ho *et al* 1983) and the MiSTIC project at Michigan (Hasanuzzaman and Mastrangelo 1996, Zaman *et al* 1999a, 1999b). The latter focuses on a systematic method to synthesize the process flow (i.e. sequence of deposition, etching, lithography, ion implantation, diffusion and reactive growth needed to build a device) for a two-dimensional cross-section of a device.

Finally, several MEMS–CAD companies[1] have modules for creating geometric models of MEMS devices. Some of these rely on existing solid modeling software, and some have developed their own solid modeling capabilities to solve the forward problem with varying degrees of flexibility. For example, some do not allow multiple etches to the same layer or a single etch affecting previous layers. Some do not store the material information along with the geometric information. None of these are currently capable of solving the inverse problem.

3. Process descriptors

A surface micromachining process is characterized by (a) the number and variety of layers offered and (b) the types of deposits, etches and dopings provided. At the highest level, a surface micromachining process is described in this paper by the following identifiers: the number of layers, the number of process steps, a list of deposit descriptors, a list of etch descriptors and a list of doping descriptors. As can be observed from the following description, these descriptors are defined in a general way so that future extensions can be made beyond surface micromachining.

3.1. Number of layers

A surface micromachining process is usually described as an n-layered process if it allows the deposition of n structural layers. A structural layer typically is a polycrystalline silicon layer that forms the load bearing bulk of the device. The MUMPS process is an example of a process with three structural layers, and Sandia's SUMMiT process is an example with four structural layers. The latter was also extended to five layers (Sniegowski and Rodgers 1998). Such an increase in the number of layers enables the design of more complex devices. In this paper, the number of layers refers to the total number of physical layers present in a fabricated MEMS device, not to the number of structural layers.

3.2. Number of process steps

A MEMS device is built up in layers by thin film deposits of a variety of materials using one of several methods (e.g., physical vapor deposition, chemical vapor deposition, oxide growth, epitaxial growth, electrochemical deposition, etc). Each layer that has been deposited (not just the structural layer) could be etched or modified by doping. Thus the entire process consists of an ordered series of deposits, etches and dopings. The total number of etches, deposits and dopings is referred to as the number of process steps.

3.3. Deposit descriptor

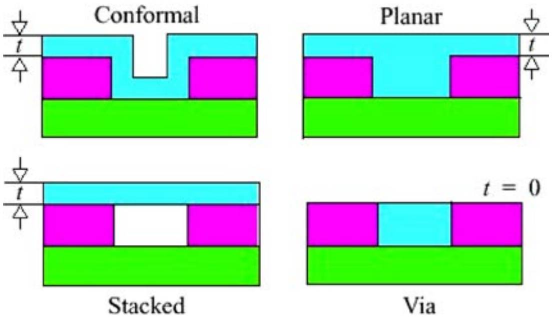


Figure 1. Types of deposits.

A deposit descriptor contains information about the type of deposit, material of deposit and deposit parameters. Figure 1 shows four deposit types, namely conformal (e.g., chemical vapor deposition), planar (e.g., electroplating and chemical mechanical polishing), stacked (e.g., wafer bonding) and via (e.g., filling a paste and polishing to the top), used in both microelectronic and micromechanical fabrication (Osterberg and Senturia 1995). Some commonly used materials in the surface micromachining processes include silicon nitride, polysilicon, silicon dioxide and some metals. The number of materials used depends on the specific process under consideration. The thickness t is a parameter for a deposit. If desired, the surface quality and residual stress may also be added as additional parameters.

3.4. Etch descriptor

An etch descriptor is described by the following identifiers: type of etch, ordered list of layers affected by the etch and depth of etch in the last affected layer. Etches are classified into three categories: regular, undercut and release. Regular and undercut etches are those that are used in conjunction with masks to remove material in selected regions. Four types of commonly used regular etches are illustrated in figure 2 for a hypothetical process. These include etch-to-layer (i.e. etching through one or more layers until a specified layer, e.g., layer 1, is reached), etch-to-material (i.e. etching through one or more layers until a specified material, e.g., poly, is reached), etch-to-depth-in-layer (i.e. etching through one or more layers to a specified depth in a given layer, e.g., layer 1) and etch-to-depth-in-material (i.e. etching through one or more layers to a specified depth in a particular material, e.g., poly, is reached). An undercut etch is a regular etch in which one of the affected layers is undercut as shown in figure 2. A release etch is the final etch, which frees the MEMS device by dissolving the silicon dioxide (sacrificial material). This etch does not usually need any masks.

Consider an etch that is initiated after a layer, e.g., layer m, has been deposited. The etch could potentially affect several layers that have already been deposited. An ordered list of layers affected by the etch is maintained, e.g., layer m, layer $m - 2$ and layer $m - 3$. The ordered list of affected layers must be specified upfront for all etches to preserve the intent of each instance of an etch. For example, the chemical or physical reaction intended for a given etch is usually fine tuned to produce an accurate and reliable outcome only for the intended sequence of layers/materials.

An etch need not affect the entire thickness of the last affected layer. Hence, the fractional depth (a fraction between

[1] MEMS-CAD companies (incomplete list): CFD-GEOM, CFD Research Corporation (webpage: www.cfdrc.com); CoventorWare, Coventor (webpage: www.memcad.com); IntelliSuite, Corning Intellisense Inc (webpage: www.intellisense.com); MEMSCAP, Memscap (webpage: www.memscap.e-sip.com); MEMS-Pro, Tanner Research (webpage: www.tanner.com).

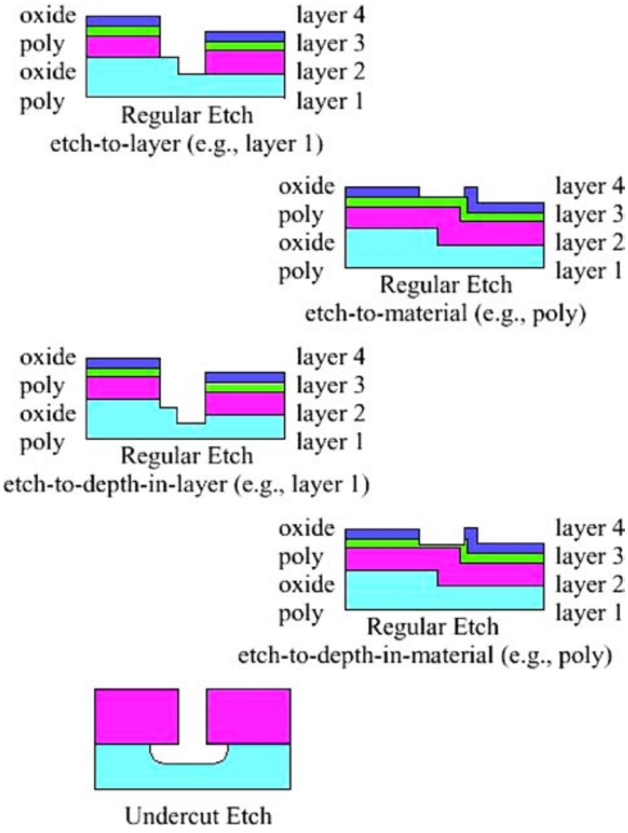


Figure 2. Types of etches.

0 and 1) of the etch or undercut in the last affected layer needs to be specified. If the fractional depth of the etch in the last affected layer is zero, it is assumed to be a regular etch of the types etch-to-layer or etch-to-material.

3.5. Doping descriptor

A doping descriptor is described by the type of doping, material of doping, ordered list of layers affected by the doping and depth of doping in the last affected layer. This is similar to etching in all respects except that the material type will be changed in the affected portions rather than removing them.

4. Process builder

The process builder is a set of classes (Deposit, Etch, Dope, and ProcessBuild) that assists in modeling the process sequence. A basic outline of the main classes is illustrated below. Since various attributes (e.g., thickness of deposits, depths of etches, etc) are fixed for a given process, the designer only needs to instantiate the classes based on knowledge of a specific process. As an example consider a hypothetical one-layer surface micromachining process (involving a total of three deposited layers) shown in figure 3. The variables f_{e1}, f_{e2} and f_{o1} indicate the fractional depths to which the last affected layer is etched/doped; t_1, t_2 and t_3 indicate the thicknesses of the deposited layers; and m_1, m_2 and m_3 indicate the material tags associated with a layer. This process sequence is used as an example later in the paper to illustrate the forward and inverse problems.

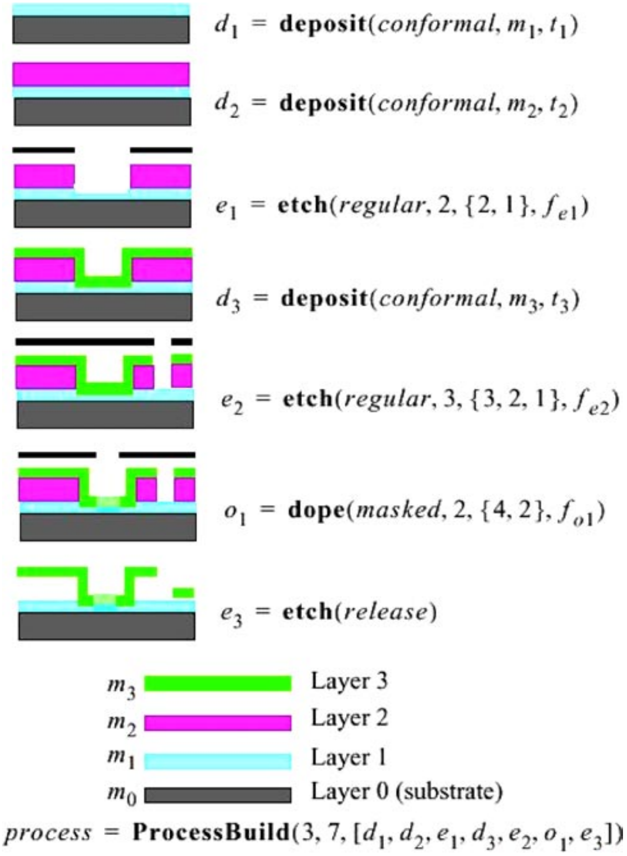

d_1 = deposit(conformal, m_1, t_1)
d_2 = deposit(conformal, m_2, t_2)
e_1 = etch(regular, 2, {2, 1}, f_{e1})
d_3 = deposit(conformal, m_3, t_3)
e_2 = etch(regular, 3, {3, 2, 1}, f_{e2})
o_1 = dope(masked, 2, {4, 2}, f_{o1})
e_3 = etch(release)
m_3 Layer 3
m_2 Layer 2
m_1 Layer 1
m_0 Layer 0 (substrate)
process = ProcessBuild(3, 7, [d_1, d_2, e_1, d_3, e_2, o_1, e_3])

Figure 3. A hypothetical one-layer process.

```
class Deposit (Type, Material, Thickness) {
    char *Type;
    char *Material;
    double Thickness;
    Deposit (char ty, char ma, double th) {
    Type = ty;
    Material = ma;
    Thickness = th;
    }
}
class Etch (Type, nAffectedLayers, AffectedLayerList,
FractionalDepth) {
    char *Type;
    int nAffectedLayers;
    int *AffectedLayerList;
    double FractionalDepth;
    Etch (char ty, int nl, int *ll, double fd) {
      Type = ty;
      nAffectedLayers = nl;
      AffectedLayerList = new int(nl);
      for (int i = 1; i < nl; i++)
         AffectedLayerList[i] = ll[i];
    FractionalDepth = fd;
    }
}
class Dope (Type, nAffectedLayers, AffectedLayerList,
FractionalDepth) {
    char *Type;
    int nAffectedLayers;
    int *AffectedLayerList;
    double FractionalDepth;
    Dope (char ty, int nl, int *ll, double fd) {
        Type = ty;
        nAffectedLayers = nl;
        AffectedLayerList = new int(nl);
        for (int i = 1; i < nl; i++)
            AffectedLayerList[i] = ll[i];
            FractionalDepth = fd;
    }
}
class ProcessBuild (nLayers, nSteps, StepsList) {
    int nLayers;
    int nSteps;
    struct_of_pointers *StepsList;
    //struct_of_pointers is a structure that can contain a
    //list of pointers to instances of the classes etch,
    //deposit and dope
    ProcessBuild(int nl, int ns, struct_of_pointers *sl) {
        nLayers = nl;
        nSteps = ns;
        for (int i = 1; i < ns; i++) StepsList[i] = sl[i];
    }
}
```

5. The forward problem

The forward problem is a formal framework of definitions and operations with which one can construct and represent the geometric model. Without this formalism it would be difficult to attempt the inverse problem, which will involve querying the geometric model. A two-step approach is used here wherein first a mathematical model (i.e. a modeling space, state-change operators and query operators) and then a computer or geometric model (Mantyla 1985) of the MEMS device are generated. The mathematical model helps to define the forward problem (and later the inverse problem) in an implementation-independent manner so that other potential computer representations and implementation techniques could be investigated in the future. Each step in the surface micromachining process is modeled in terms of set operations (e.g., unions, differences and the Minkowski additions) on heterogeneous or multi-material point sets. For this paper, the mathematical model of heterogeneous point sets is represented in the computer and implemented by means of a cellular or voxel representation (Chandru and Manohar 1997), although any other suitable implementation may be adopted as well.

5.1. The mathematical model

Consider a MEMS device of J layers that is fabricated in I process steps (sum of the total number of deposits J, etches K and dopings L, i.e. $I = J + K + L$). The indices i, j, k and l are henceforth used count to process steps, deposit steps, etch steps and doping steps, respectively. The model M of a MEMS device is a layered, *multi-material point set* that can be interpreted as a union of all J layers, i.e. $M = \cup L_j$, which are themselves multi-material point sets. Thus the modeling space comprises a point set $\{P\}$, a corresponding set of unique material tags $\{m\}$ at each point and a corresponding unique layer tag LT at each point, i.e. $M = \{P, \{m\}, LT\}$. This modeling space allows for the representation of both doped and undoped regions, for example, in an undoped region a point will be associated with a single primary material tag, e.g., (p, m_p, lt), whereas in a doped region a point may be associated with additional secondary material tags corresponding to the doped material as in $(p, (m_p, m_{s1}, m_{s2}, \ldots), lt)$.

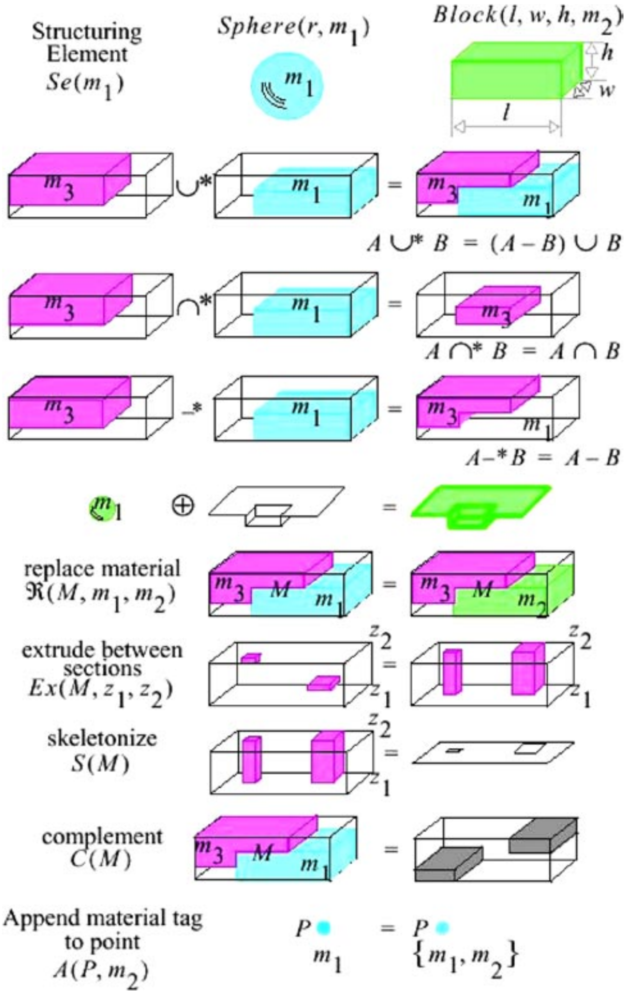


Figure 4. State change operators.

The exposed boundary of a model M is represented by the symbol $\partial_e M$. Each ith process step modifies the existing model M_i to create the updated model M_{i+1}. The modification of models in the modeling space is done by means of state change operators (figure 4) and queries (figure 5). Note that each operation is pictorially shown in addition to its mathematical representation. For instance, the multi-material union operator is defined such that the first material is overridden (consequently the operation is non-commutative). Note also that the union, intersection and subtraction operators are also non-commutative. The interested reader may find more detailed mathematical descriptions of the general

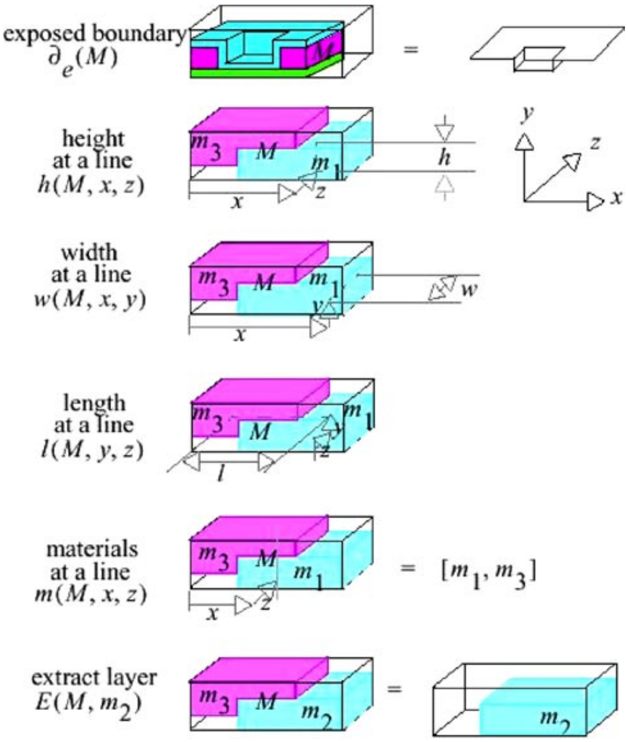


Figure 5. Queries on the mathematical and geometric models.

multi-material modeling paradigm in the paper by Kumar and Dutta (1998).

The operators and queries described above are used to model (as concisely and intuitively as possible) the deposits, etches and dopings, which are the building blocks that enable the construction of a MEMS device given the process sequence and the masks. Note that the building blocks are not unique because there could be several ways of modeling a deposit, etch or doping. Furthermore, the mathematical model is valid for 2D and 3D. A detailed account of equations and illustrations in both 2D and 3D are discussed in Ananthakrishnan (2000).

5.1.1. Deposits. A given deposit step d_j (instantiated as illustrated in figure 3) of the ith process step acts on the current model of the device M_{i-1} as represented $M_i = Deposit(M_{i-1}, d_j)$. The function *Deposit* may be considered as a generic function that knows the individual steps involved in constructing M_{i+1} from M_i for each type of deposit (i.e. conformal, stacked, planar and via). For example, in a given step i and layer j, a conformal deposit of material m_j can be modeled in 2D (see figure 6) by considering (a) a circular structuring element S (a point set with the appropriate material tag) whose radius r is equal to that of the thickness t of the deposit, (b) the exposed boundary $\partial_e M_i$ of the current model and (c) the current model M_i as follows. Similarly, the three equations below represent planar, via and stacked deposits respectively.

$$\begin{aligned} S &= \text{circle}(r, m_j) \\ M_{i+1} &= (S \oplus \partial_e M_i) \cup M_i \end{aligned} \tag{1}$$

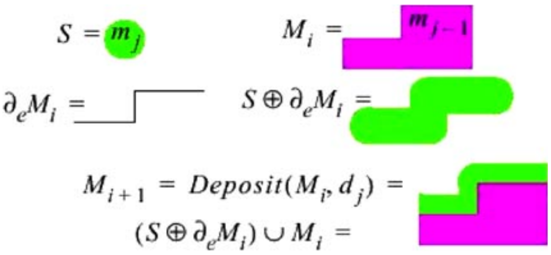

$S =$
m_j
$M_i =$
m_{j-1}
$\partial_e M_i =$
$S \oplus \partial_e M_i =$
$M_{i+1} = Deposit(M_i, d_j) =$
$(S \oplus \partial_e M_i) \cup M_i =$

Figure 6. A conformal deposit in 2D.

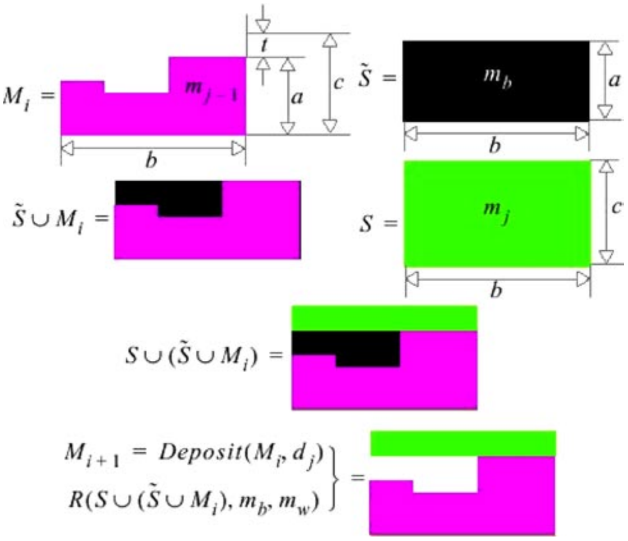

$M_i =$
m_{j-1}
t
a
c
b
$\tilde{S} =$
m_b
a
b
$\tilde{S} \cup M_i =$
$S =$
m_j
c
b
$S \cup (\tilde{S} \cup M_i) =$
$M_{i+1} = Deposit(M_i, d_j)$
$R(S \cup (\tilde{S} \cup M_i), m_b, m_w)$
=

Figure 7. Modeling a stack deposit (e.g., wafer bonding).

planar deposit:

$$\begin{aligned} a &= \max\{h(M_i, x)\} + t \\ b &= \max\{w(M_{i-1}, y)\} \\ S &= \text{rectangle}(a, b, m_j) \\ M_{i+1} &= S \cup M_i \end{aligned} \tag{2}$$

via filling:

$$\begin{aligned} a &= \max\{h(M_i, x)\} \\ b &= \max\{w(M_{i-1}, y)\} \\ S &= \text{rectangle}(a, b, m_j) \\ M_{i+1} &= S \cup M_i \end{aligned} \tag{3}$$

stack deposit:

$$\begin{aligned} a &= \max\{h(M_i, x)\} \\ b &= \max\{w(M_{i-1}, y)\} \\ \tilde{S} &= \text{rectangle}(a, b, m_{\text{black}}) \\ c &= \max\{h(M_i, x)\} + t \\ S &= \text{rectangle}(c, b, m_j) \\ M_{i+1} &= R(S \cup (\tilde{S} \cup M_i), m_{\text{black}}, m_{\text{white}}) \end{aligned} \tag{4}$$

The most complex of the above, i.e. the stack deposit, is shown pictorially as well in figure 7.

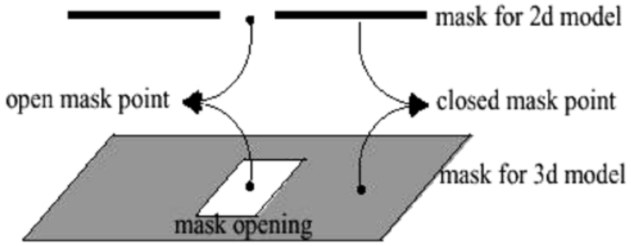


Figure 8. Masks in 2D and 3D.

5.1.2. Masks. Masks, which may be used in etches and dopings, are of one dimension less than the dimension of the geometric model. That is, they are lines in 2D modeling and planes in 3D. All points on a mask are classified as open or closed as illustrated in figure 8. An open point exposes the substrate directly underneath and allows the etchant/dopant to act on the exposed portions of the part. Conversely, a closed point protects the surface directly underneath. Any contiguous set of open points on the mask is referred to as a mask opening. A single mask may have multiple openings.

5.1.3. Etches and dopings. Each etch step is associated with a unique mask (except the release etch, which does not need a mask). A kth etch step modifies the model as per the type of etch and the mask $\bar{m}_i$. That is

$$M_{i+1} = \text{Etch}(M_i, e_k, \bar{m}_i). \tag{5}$$

Figures 9 and 10 illustrate how a regular straight-wall etch and an undercut isotropic etch are modeled. At first sight, even though these operations may look involved, they are algorithmically convenient and are unambiguous in the way they ought to be implemented. Finally, as release etch is modeled by simply replacing the specified sacrificial material with 'white' material using the replace operator. Doping operation is similar to etching except that the replacing material in the final step is a modified material rather than 'white' material. A doping step may or may not have a mask.

6. The inverse problem

The inverse problem involves determining the set of masks given a complete geometric model M of the MEMS device and a process sequence chosen by the designer. The technique presented here to solve the inverse problem uses an intermediate geometric model that is reconstructed from the beginning (in stages) using the known process sequence. The procedure starts with the initialization of an intermediate geometric model $\tilde{M}_0$ that represents the substrate. The list of steps in the process sequence is then sequentially queried, and the corresponding intermediate geometric model $\tilde{M}_i$ $(i = 1, \ldots, I)$ is built. For each step i in the process sequence, a series of actions are undertaken depending on whether an etch, deposit, or doping is encountered. The detailed actions are described below, and a pictorial sequence of the actions is shown in figure 11. Note that the description below is general enough to cover three-dimensional MEMS devices; only the illustration is two dimensional.

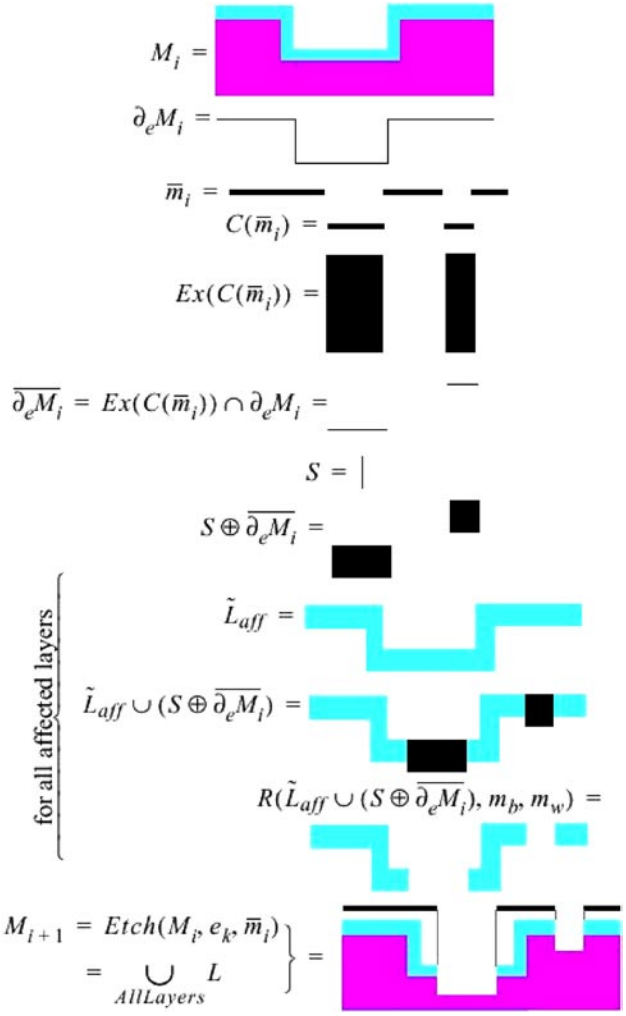

M_i =
$\partial_e M_i$ =
$\overline{m}_i$ =
$C(\overline{m}_i)$ =
$Ex(C(\overline{m}_i))$ =
$\overline{\partial_e M_i} = Ex(C(\overline{m}_i)) \cap \partial_e M_i$ =
S =
$S \oplus \overline{\partial_e M_i}$ =
$\tilde{L}_{aff}$ =
$\tilde{L}_{aff} \cup (S \oplus \overline{\partial_e M_i})$ =
$R(\tilde{L}_{aff} \cup (S \oplus \overline{\partial_e M_i}), m_b, m_w)$ =
for all affected layers
$M_{i+1} = Etch(M_i, e_k, \overline{m}_i) = \bigcup_{AllLayers} L$ =

Figure 9. Modeling a straight-wall regular etch in 2D.

Deposits. If a deposition step d_j is encountered, the intermediate geometric model $\tilde{M}_i$ corresponding to the ith step in the process sequence is generated using the forward problem as shown in steps d_j $(j = 1, 2, 3)$ in figure 11. Note that it is possible to build the intermediate geometric model since deposits do not use masks, and all other information is known from the process sequence.

Etches. Three actions are undertaken for each etching step encountered. First, the intermediate geometric model $\tilde{M}_i$ is set equal to $\tilde{M}_{i-1}$. Second, the layer L of the complete geometric model (i.e. the part) corresponding to the first affected layer of etch e_k is extracted. Finally, the layer $\tilde{L}$ of the intermediate geometric model corresponding to the first affected layer is extracted. The mathematical representation of these actions is shown below.

$$\begin{aligned} \tilde{M}_i &= \tilde{M}_{i-1} \\ f &= e_k \cdot \text{AffectedLayerList}[1] \\ L &= E(M, m_f) \\ \tilde{L} &= E(\tilde{M}, m_f). \end{aligned} \tag{6}$$

The extracted layers L and $\tilde{L}$ are then compared to generate a potential set of mask openings as explained in section 6.1.

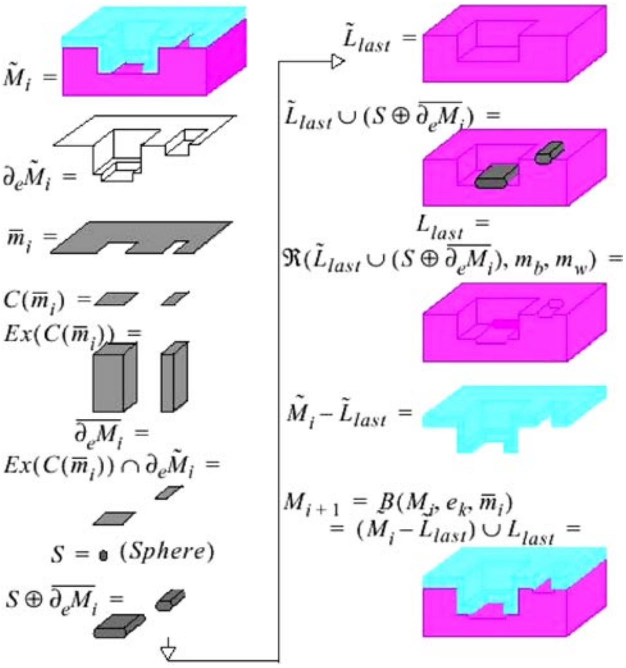

$\tilde{M}_i =$
$\partial_e \tilde{M}_i =$
$\bar{m}_i =$
$C(\bar{m}_i) =$
$Ex(C(\bar{m}_i)) =$
$\overline{\partial_e M_i} =$
$Ex(C(\bar{m}_i)) \cap \partial_e \tilde{M}_i =$
$S =$ ● $(Sphere)$
$S \oplus \overline{\partial_e M_i} =$
$\tilde{L}_{last} =$
$\tilde{L}_{last} \cup (S \oplus \overline{\partial_e M_i}) =$
$L_{last} =$
$\Re(\tilde{L}_{last} \cup (S \oplus \overline{\partial_e M_i}), m_b, m_w) =$
$\tilde{M}_i - \tilde{L}_{last} =$
$M_{i+1} = B(M_j, e_k, \bar{m}_i)$
$= (M_i - \tilde{L}_{last}) \cup L_{last} =$

Figure 10. Modeling an undercut isotropic etch in 3D.

These steps are summarized in the first column of step e_1 in figure 11. The mask openings obtained in this manner may not be correct as several etches could affect any given layer. Hence a procedure to detect and eliminate incorrect mask openings is undertaken as explained in section 6.2, resulting in the correct set of masks $\bar{m}_i$. The intermediate geometric model is then updated appropriately. The procedure described is shown in steps e_1 and e_2 in figure 11.

Doping. A doping step is handled in the same way as an etching step with the difference that in doping the material change is considered, as it remains unaltered geometrically. The step o_1 in figure 11 illustrates this. All the steps in the inverse problem are best summarized algorithmically as follows:

```
InverseProblem (Model M, ProcessBuild P)
$\tilde{M}_0 = substrate$;
i = 0//A counter to indicate the current step in process sequence
for (i = 1; i <= P.nSteps; i++) {
    j = 0; //counter to indicate the current deposit step
    k = 0; //counter to indicate the current etch step
    l = 0; //counter to indicate the current doping step
    if P.StepsList[i] is a deposit {
        j = j + 1;
        d_j = P.StepsList[i]; //building the intermediate geometric
        model from the process sequence
$\tilde{M}_i = Deposit(\tilde{M}_{i-1}, d_j)$;
} else if P.StepsList[i] is an etch {
k = k + 1;
    e_k = P.StepsList[i];
$\tilde{M}_i = \tilde{M}_{i-1}$; //instantiate the current model
$f = e_k \cdot AffectedLayerList[1]$; //find the first affected layer of the etch
$L = E(M_i, m_f)$; //extract a layer from complete geometric model
$\tilde{L} = E(\tilde{M}_i, m_f)$//extract a layer from intermediate geometric model
$\bar{m}_i = GenerateMasks()$; //generate a potential mask set
$\bar{m}_i = SubdivideMasks(\bar{m}_i)$; //subdivide each opening of the potential mask set based on
    //1. change in thickness of last affected layer
    //2. layer in contact with the upper surface of the last
    affected layer.
$\bar{m}_i = CorrectMaskOpenings(\bar{m}_i)$; //resolve mask conflicts and generate correct mask
$\tilde{M}_i = Etch(\tilde{M}_i, e_k, \bar{m}_i)$; //update the intermediate geometric model with the correct mask
}
} else if P.StepsList[i] is a doping {
```

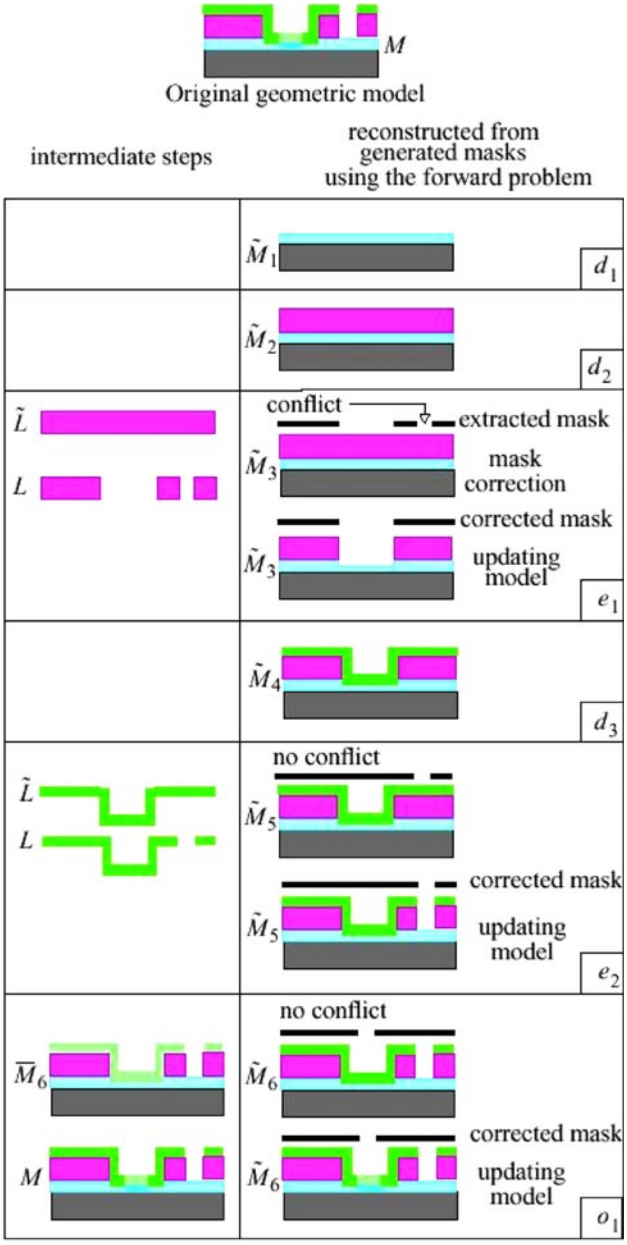


Figure 11. Illustrated procedure for the inverse problem.

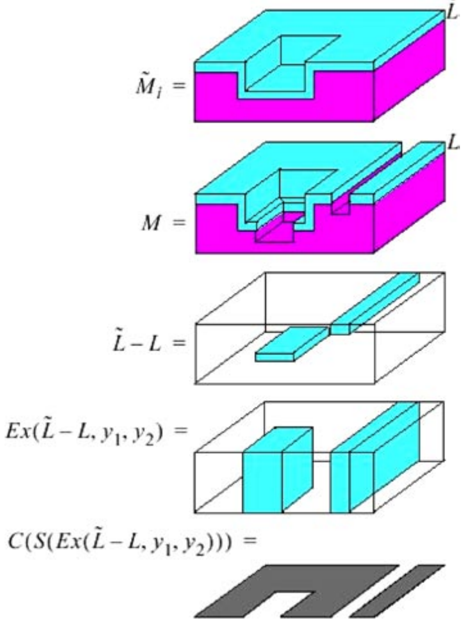

$\tilde{L}$
$\tilde{M}_i =$
L
$M =$
$\tilde{L} - L =$
$Ex(\tilde{L} - L, y_1, y_2) =$
$C(S(Ex(\tilde{L} - L, y_1, y_2))) =$

Figure 12. Generation of potential masks.

```
        $l = l + 1$;
        $o_1$ = P.StepsList[i];
    $\tilde{M}_i = \tilde{M}_{i-1}$; //instantiate the current model
    $\tilde{M}_i = Dope(\tilde{M}_i, o_l, unmasked)$; //Perform unmasked
        doping on intermediate geometric model
    $\bar{m}_i = GenerateMasks()$; //generate a potential mask set
    ... //steps similar to etching in obtaining the correct
masks
    $\tilde{M}_i = Dope(\tilde{M}_i, o_l, \bar{m}_i)$; //update the intermediate
        geometric model with the correct mask
        }
}
```

6.1. Generating potential mask openings

Comparing the openings on corresponding layers of geometric models (complete and intermediate), gives an indication of the mask openings of the mask for a particular etch or doping step. However, not all the predicted mask openings need to belong to the mask associated with the step under consideration. This is because multiple etches/dopings can affect one layer. Hence they are termed as potential mask openings. This section outlines how these potential mask openings are generated. Sections 6.2 and 6.3 describe how to sort out whether a mask opening can be correctly associated with a particular etch step.

Etch masks are generated by extracting the first affected layer of the kth etch. These layers are subtracted from each other to detect regions of difference in layers. These differences indicate the presence of mask openings. The mask openings are generated by extruding and skeletonizing the intersections in layers as illustrated in figure 12 and in the following equations:

$$\bar{m}_i = C(S(Ex(\tilde{L} - L), y_1, y_2))$$

where

$$L = E(M, m_f) \qquad \text{and} \qquad \tilde{L} = E(\tilde{M}_i, m_f). \tag{7}$$

The mask generation procedure for dopings is given below

$$\begin{aligned} \tilde{M}_i &= \text{Dope}(\tilde{M}_i, o_l, \text{nullmask}) \\ \bar{m}_i &= C\big(S\big(Ex\big(E\big(M, m_{o_l}\big) \cap E\big(\tilde{M}_i, m_{o_l}\big), y_1, y_2\big)\big)\big). \end{aligned} \tag{8}$$

Since the procedure for doping steps is almost similar to the etches, henceforth, only etches are discussed.

6.2. Subdivision of mask openings

A single opening in a mask (as predicted by the procedure in section 6.1) may be the result of more than one etch acting on the layer under consideration. Thus each mask opening needs to be unambiguously subdivided into distinct mask openings, which reflect a distinct series of etches acting on them. The cases where this occurs are enumerated below and illustrated in figure 13. Completely overlapping masks of two distinct etches acting on the same layer will yield a single potential mask opening. Partially overlapping masks of two distinct etches acting on the same layer will yield a single potential mask opening. Touching masks of two distinct etches acting on the same layer will yield a single potential mask opening. Redundant etches specified in the process sequence, e.g., specifying an etch of the same parameters two times in sequence, is disallowed. In all the cases mentioned above, the potential mask opening needs to be subdivided into more than one distinct mask openings. Two following criteria are used for the subdivision.

(i) *The subdivision of a mask opening based on the thickness of the last affected layer of the etch*: This criterion is illustrated by the 'process flow 1' shown in figure 13, where the potential mask for etch e_1 is subdivided based on the thickness of the last affected layer. If the thickness of the last affected layer changes, it is an indication that more than one etch has occurred. Hence, there are as many subdivisions in the mask as the number of different thicknesses in the layer.

(ii) *The subdivision of a mask opening based on the layer which the last affected layer is immediately in contact with*: This criterion is illustrated by 'process flow 3' shown in figure 13. In this case the openings of two different masks are adjacent to each other and have the same depth in the last affected layer making it impossible to subdivide based on thickness. Hence the layers immediately in contact with the top surface of the last affected layer are checked. The subdivision is based on regions of the last affected layer in contact with a distinct upper layer. The case shown in 'process flow 2' is a combination of 'process flow 1' and 'process flow 3'.

6.3. Validating mask openings

This step follows after the distinct potential mask openings have been generated using the procedures outlined in the previous sections. Each distinct mask opening has to be checked to detect whether the current etch has affected the fth layer (first affected layer of the etch under consideration). To check whether a given mask opening affects the fth layer, it is sufficient to check the complete geometric model along any given line (referred to as a *check line*) passing anywhere through the mask opening, e.g., line AA′ or BB′ in figure 14. Note that if the subdivision of masks was

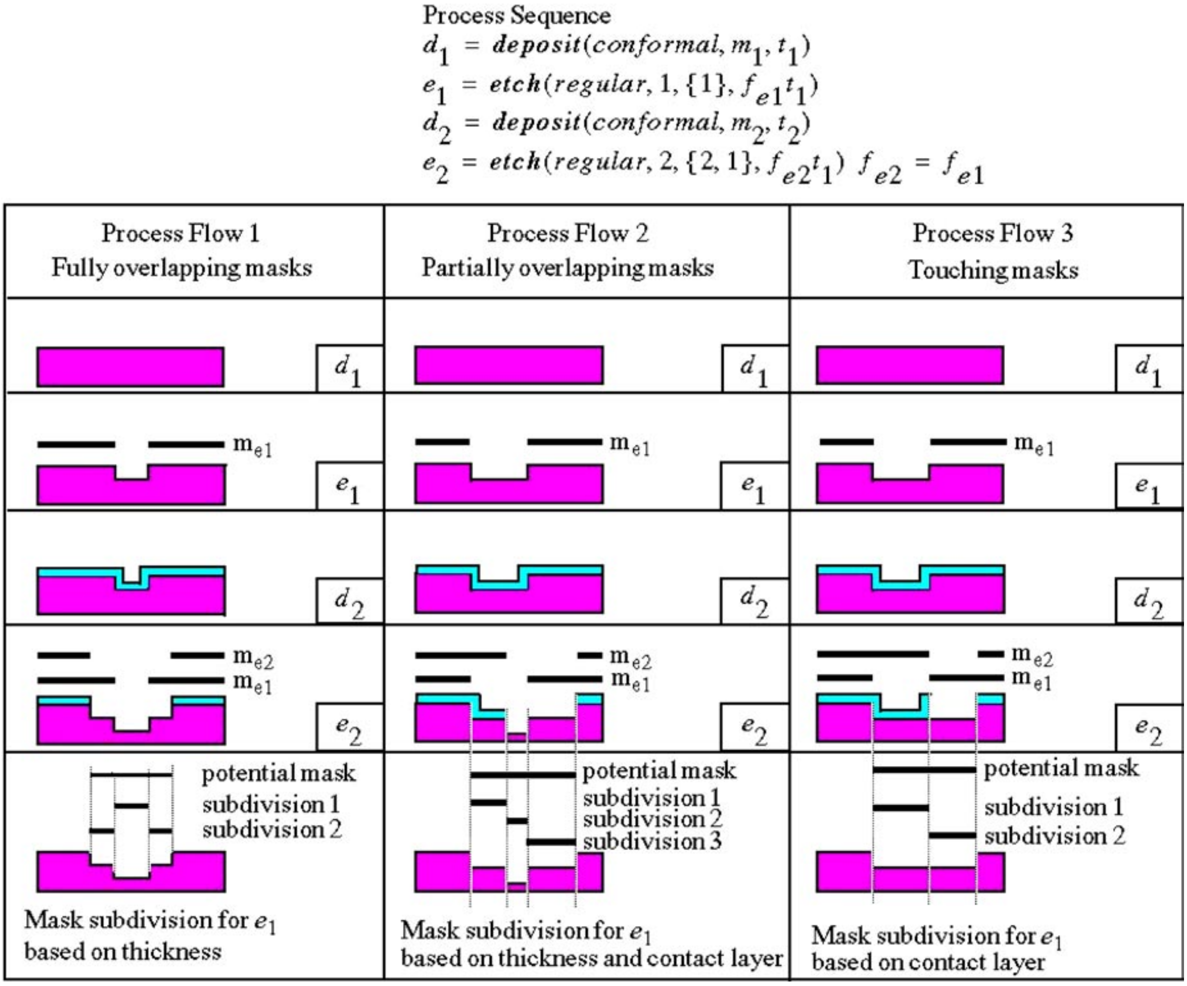

Process Sequence
d_1 = deposit(conformal, m_1, t_1)
e_1 = etch(regular, 1, {1}, $f_{e1}t_1$)
d_2 = deposit(conformal, m_2, t_2)
e_2 = etch(regular, 2, {2, 1}, $f_{e2}t_1$) f_{e2} = f_{e1}
Process Flow 1
Fully overlapping masks
Process Flow 2
Partially overlapping masks
Process Flow 3
Touching masks
d_1
m_{e1}
e_1
d_2
m_{e2}
m_{e1}
e_2
potential mask
subdivision 1
subdivision 2
subdivision 3
Mask subdivision for e_1 based on thickness
Mask subdivision for e_1 based on thickness and contact layer
Mask subdivision for e_1 based on contact layer

Figure 13. Subdivision of mask openings to associate an etch to a particular step.

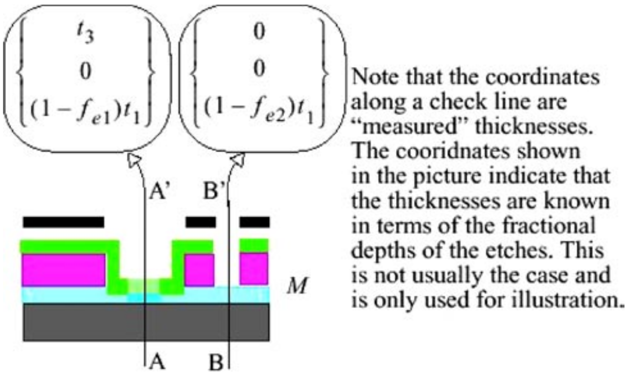

$\begin{Bmatrix} t_3 \\ 0 \\ (1-f_{e1})t_1 \end{Bmatrix}$
$\begin{Bmatrix} 0 \\ 0 \\ (1-f_{e2})t_1 \end{Bmatrix}$
Note that the coordinates along a check line are "measured" thicknesses. The cooridnates shown in the picture indicate that the thicknesses are known in terms of the fractional depths of the etches. This is not usually the case and is only used for illustration.
A'
B'
M
A
B

Figure 14. Check lines to validate mask openings.

not undertaken every point in the mask opening has to be checked using the procedure described below whereas after the subdivision only one point in the mask opening need be checked. Coordinates are then given to the complete geometric model along the check line. The coordinates are simply the 'measured' thicknesses of layers along the check line from top to bottom.

Figure 14 shows the coordinates along check lines AA$'$ and BB$'$. The coordinates exist in J-dimensional space $\Re^J$, where J is the maximum number of deposited layers. Since the example illustrated in figure 3 has three deposited layers, the coordinates in figure 14 lie in $\Re^3$. Each etch and deposit is also given a coordinate depending on the thickness of the deposit and the depth of the etch. The coordinate of a deposit is a vector with a single positive entry corresponding to the deposit thickness. The coordinate of an etch is a vector with negative entries corresponding to the etch depth in each affected layer. Coordinates of the deposits of figure 3 are listed in (27).

$$d_1 \to \begin{Bmatrix} 0 \\ 0 \\ t_1 \end{Bmatrix} \qquad d_2 \to \begin{Bmatrix} 0 \\ t_2 \\ 0 \end{Bmatrix} \qquad d_3 \to \begin{Bmatrix} t_3 \\ 0 \\ 0 \end{Bmatrix}$$
$$e_1 \to \begin{Bmatrix} 0 \\ -t_2 \\ -f_{e1}t_1 \end{Bmatrix} \qquad e_2 \to \begin{Bmatrix} -t_3 \\ -t_2 \\ -f_{e2}t_1 \end{Bmatrix}. \tag{9}$$

Each coordinate of the check line is a result of a linear combination of a sequence of deposits, and etches as shown below for check lines AA$'$ and BB$'$, respectively:

$$\begin{Bmatrix} t_3 \\ 0 \\ (1-f_{e1})t_1 \end{Bmatrix} = c_1 \begin{Bmatrix} 0 \\ -t_2 \\ -f_{e1}t_1 \end{Bmatrix} + c_2 \begin{Bmatrix} -t_3 \\ -t_2 \\ -f_{e2}t_1 \end{Bmatrix}$$
$$+ c_3 \begin{Bmatrix} 0 \\ 0 \\ t_1 \end{Bmatrix} + c_4 \begin{Bmatrix} 0 \\ t_2 \\ 0 \end{Bmatrix} + c_5 \begin{Bmatrix} t_3 \\ 0 \\ 0 \end{Bmatrix} \tag{10a}$$

$$\begin{Bmatrix} 0 \\ 0 \\ (1-f_{e2})t_1 \end{Bmatrix} = c_1 \begin{Bmatrix} 0 \\ -t_2 \\ -f_{e1}t_1 \end{Bmatrix} + c_2 \begin{Bmatrix} -t_3 \\ -t_2 \\ -f_{e2}t_1 \end{Bmatrix} + c_3 \begin{Bmatrix} 0 \\ 0 \\ t_1 \end{Bmatrix} + c_4 \begin{Bmatrix} 0 \\ t_2 \\ 0 \end{Bmatrix} + c_5 \begin{Bmatrix} t_3 \\ 0 \\ 0 \end{Bmatrix}. \qquad (10b)$$

At a given check line, the unknown constants c_i $(i = 1, \ldots, 5)$ can only take values of 0 or 1 because an etch/deposit either is there or not. In particular, the constants related to deposits (i.e. c_i $(i = 3, \ldots, 5)$) are always 1 because deposits always exist. Thus to detect whether a particular etch occurred at a check line, one needs to find the values of the constants c_i $(i = 1, 2)$ related to the etches. For a general process with J deposits and K etches, this will yield J equations in K unknowns. The system of equations will be over-constrained, perfectly constrained or under-constrained depending on whether the number of deposits is greater than, equal to or lesser than the number of etches. In all the cases, there can be multiple solutions, one solution or no solutions depending on the rank of the above linear system of equations. Another notable point is that all the equations are independent of deposit thicknesses and only depend on the fractional depths of the etches. The equations above simplify to the form:

$$[A]\{c\} = \{b\}. \qquad (11)$$

For check lines AA′ and BB′, respectively, equation (11) can be obtained as

$$\begin{bmatrix} 0 & 1 \\ 1 & 1 \\ f_{e1} & f_{e2} \end{bmatrix} \begin{Bmatrix} c_1 \\ c_2 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 1 \\ f_{e1} \end{Bmatrix}$$

$$\text{and} \quad \begin{bmatrix} 0 & 1 \\ 1 & 1 \\ f_{e1} & f_{e2} \end{bmatrix} \begin{Bmatrix} c_1 \\ c_2 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 1 \\ f_{e2} \end{Bmatrix}. \qquad (12)$$

The general solution of the system of equations involves using a singular value decomposition (SVD) to find the basic and free variables (Strang 1988, Press *et al* 1993). If there are R free variables, there are 2^R potential solutions because the variables here can only take on the values 0 and 1. However, not all the 2^R solutions need be feasible because the basic variables may not be only 1s and 0s. Thus the system of equations can potentially yield zero, one or multiple solutions. The situation of no solutions implies that the given device cannot be fabricated with the specified process. Multiple solutions imply that the given mask opening can be fabricated by different sequences of etches. For the example shown above the solutions are unique ($(c_1, c_2) = (1, 0)$ for AA′ and $(c_1, c_2) = (0, 1)$ for BB′) and correspond to the original mask openings in figure 3.

The above solution procedure thus gives an answer as to whether the given etch e_k is present or absent for a given mask opening. If the etch is present, then the mask opening can be kept, if not, the mask opening needs to be closed. If there are multiple feasible solutions, user input is required to select a given solution or multiple branches of solutions could be maintained. Additionally, if the status of a prior etch has been determined (i.e. the corrected mask openings of a previous etch have already been calculated and determined), there will be a reduction in the number of variables. In summary, the steps involved in detecting and eliminating incorrect mask openings are

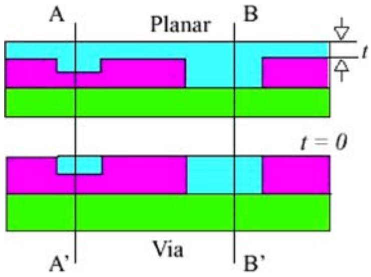


Figure 15. A difficulty of mask-dependent variable thickness in planar and via deposits.

1. Select any check line that passes through the given mask opening.
2. Find the coordinates of the geometric model along the check line.
3. Find the coordinates of all the etches and deposits.
4. Express the coordinates along the check line as a linear combination of etches and deposits to get a system of linear equations.
5. Solve the system of linear equations using SVD, while taking care to ensure that some variables may have been assigned values previously.
6. If there are multiple solutions, prompt the designer to decide on whether or not the mask opening will remain.
7. Finally, retain or eliminate (i.e. close up) the potential mask openings depending on the solution.

6.4. Remarks on the inverse problem: assumptions and restrictions

The main assumption made in the inverse problem is that the sacrificial layers are assumed to be present (i.e. the device is not released) in the complete geometric model of the device. This is not a serious limitation because the designer can easily add sacrificial layer to the part being designed. However, this constraint should be relaxed in future work where the design of the device is taken into account, as the sacrificial layers do not contribute to the structural behavior of the device. A restriction that the solution to the inverse problem presented in this paper imposes on the process is that planar and via deposited layers cannot be the last affected layer of any etch. This is so because, to etch the last affected layer, it is required that its thickness be known consistently. In the case of planar and via deposited layers, the thickness of a layer at any point is a variable, which depends on the surface topography of the device at the previous stage. For example in figure 15, it is not possible to know the depth to which the planar or via deposited layers can be etched without affecting the layer below it.

Three special cases have been identified in dealing with the inverse problem. Consider the problem of finding the coordinates of the checkline CC′ in figure 16(a). In this case, the thickness of the layer will be greater than the corresponding thickness of the deposit. Requiring that the coordinate (that is greater than the thickness of the corresponding deposit) to be equal to the thickness of the corresponding deposit yields correct solutions under the assumptions stated in the previous sections.

Figure 16(b) shows another special case where an undercut etch $e_1 = Etch(undercut, 2, \{2, 1, 0\}, 0\}$ in layer d_1

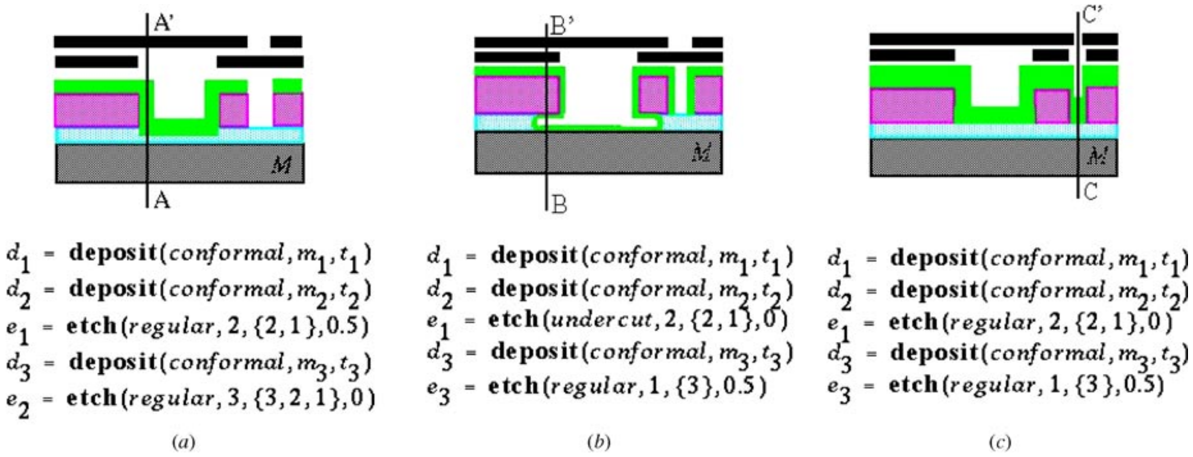

A'
A
M
d_1 = deposit(conformal, m_1, t_1)
d_2 = deposit(conformal, m_2, t_2)
e_1 = etch(regular, 2, {2, 1}, 0.5)
d_3 = deposit(conformal, m_3, t_3)
e_2 = etch(regular, 3, {3, 2, 1}, 0)
(a)
B'
B
M
d_1 = deposit(conformal, m_1, t_1)
d_2 = deposit(conformal, m_2, t_2)
e_1 = etch(undercut, 2, {2, 1}, 0)
d_3 = deposit(conformal, m_3, t_3)
e_3 = etch(regular, 1, {3}, 0.5)
(b)
C'
C
M
d_1 = deposit(conformal, m_1, t_1)
d_2 = deposit(conformal, m_2, t_2)
e_1 = etch(regular, 2, {2, 1}, 0)
d_3 = deposit(conformal, m_3, t_3)
e_3 = etch(regular, 1, {3}, 0.5)
(c)

Figure 16. Three special cases encountered in the inverse problem. All three are resolved.

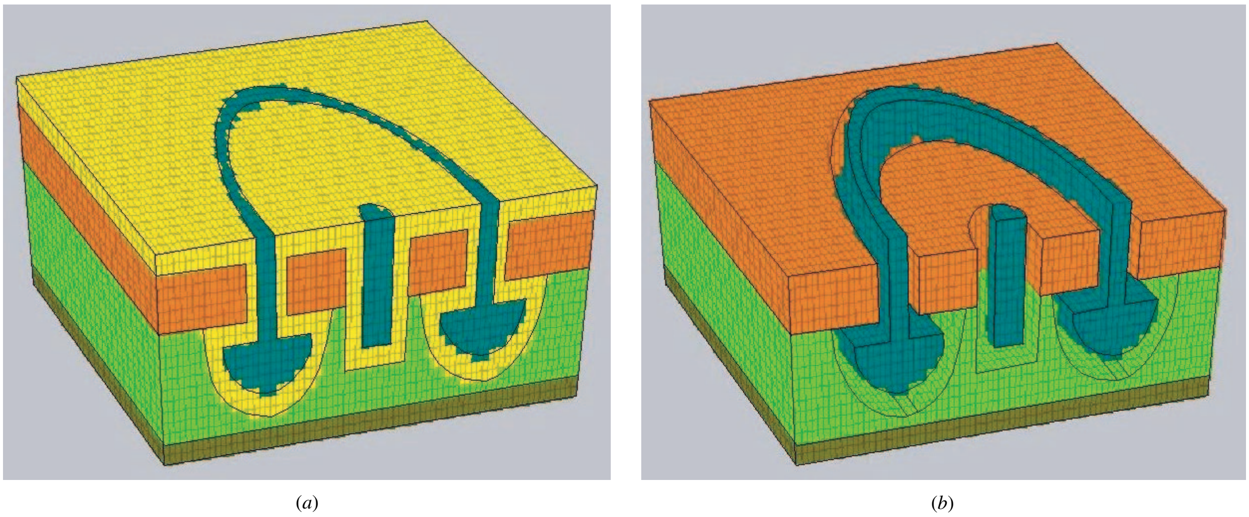


Figure 17. Creation of the undercut in the microhinge fabricated using the SUMMiT process (*a*) before release (*b*) after release.

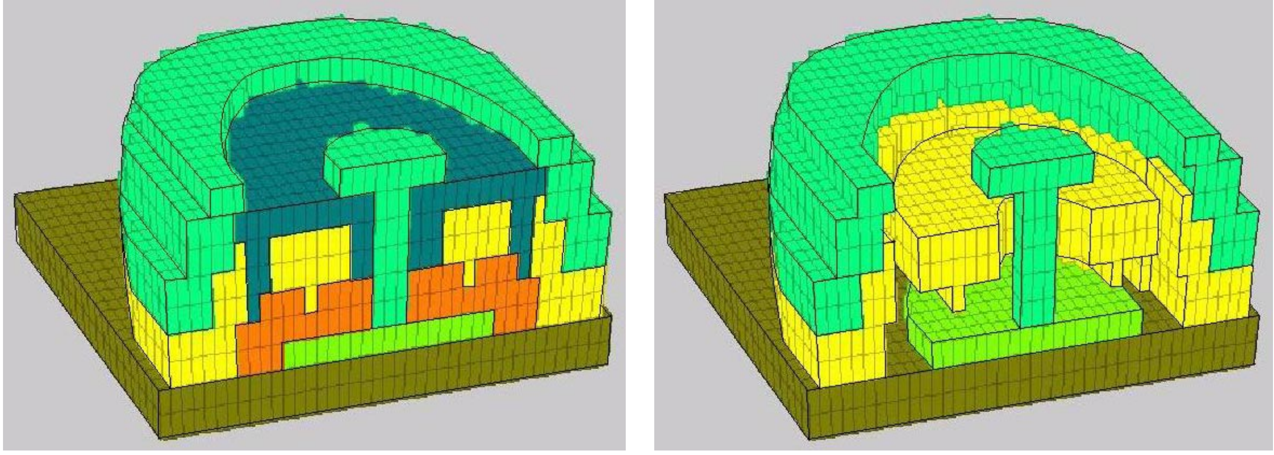
and a subsequent conformal deposit d_3. The initial mask opening is constructed in the same way as that of a regular etch. To verify that the mask opening belongs to the undercut etch and not to another regular etch having similar coordinates, i.e. $e_1 = Etch(regular, 2, \{2, 1, 0\}, 0)$, it is sufficient to verify the coordinates at the check line BB′ constructed as shown. If the entry corresponding to the last but one affected layer is non-zero, and the entry corresponding to the last affected layer is less than the original deposited thickness of the layer, then it is an undercut etch as a regular etch at the same point would have also removed the last but one affected layer.

The third special case arises due to filling up of narrow holes by conformal deposits. As shown in figure 16(*c*), the conformal deposit of layer d_3 on the two sidewalls of the cut in d_2 (along check line CC′) have merged to completely fill the cut. This leads to a problem that is similar to that of vertical walls of conformal deposits discussed previously and is dealt with in a similar manner.

7. Implementation and results

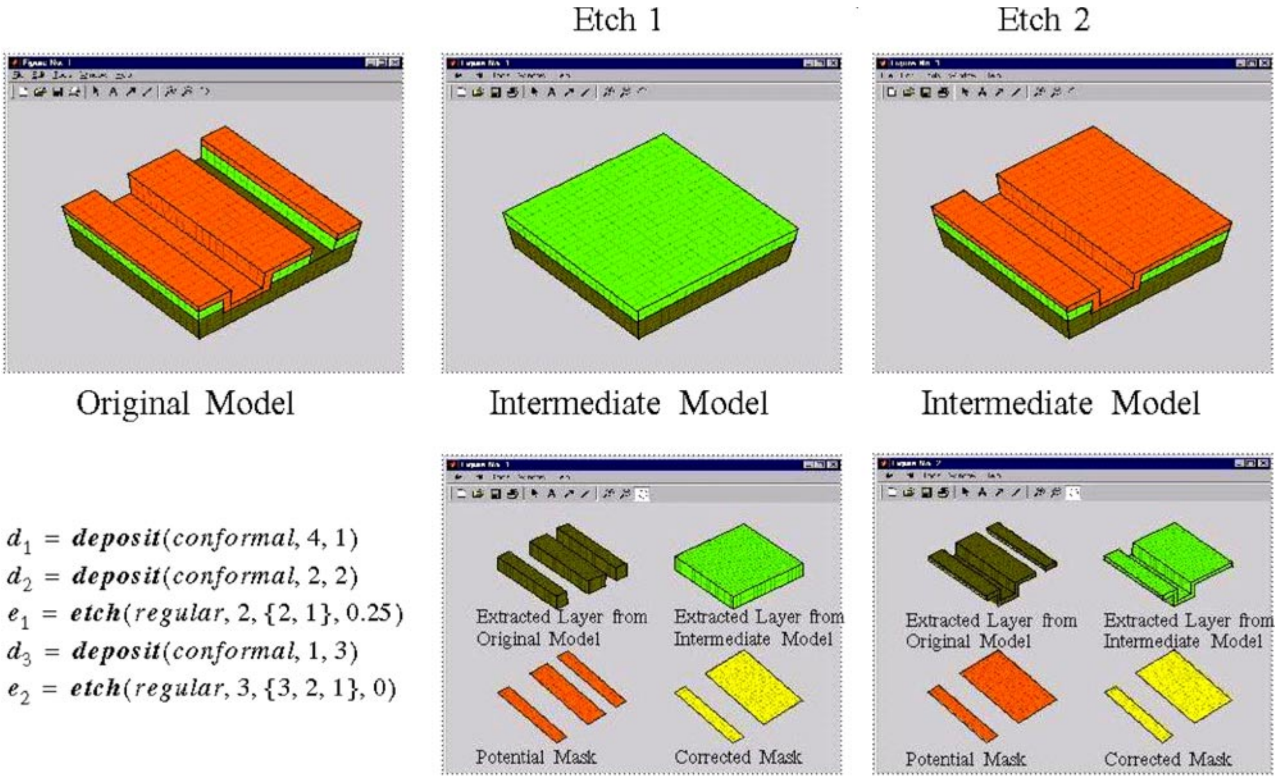
The mathematical model, operations and steps presented in the previous sections can be implemented in many different ways depending on how the geometric model is represented in 2D or 3D and manipulated in the computer. In this work, a volumetric model of MEMS devices was implemented using *voxels* (Chandru and Manohar 1997). A voxel is a representation of a volume element of a solid body just as a pixel is an area element of a 2D representation of a body. Although the size of the voxels within a model can be varied, uniform size was used in this work. Thus, the coordinates of its center and a material tag describe a voxel. A number of voxels are put together to form a volumetric model. This data structure is amenable for the Boolean and other operators shown in figures 4 and 5 that need to be implemented to create a computer model of the MEMS devices. The implementation of the voxel model was developed in C++, and the model was displayed by a program written in MATLAB®. A flood-filling algorithm (Lieberman 1978) was used for rendering purposes. A few generic operations are described below.

For the conformal deposit, in one of the implementations, the exposed boundary query operation $\partial_e M$ is done as follows. A closed bounding box is created around the model with some empty space above it. A seed cell of black color is started in the 3D space above the model and is recursively propagated



(*a*) (*b*)

Figure 18. Simulation of a micromotor fabricated using the MUMPs (*a*) before release (*b*) after release.

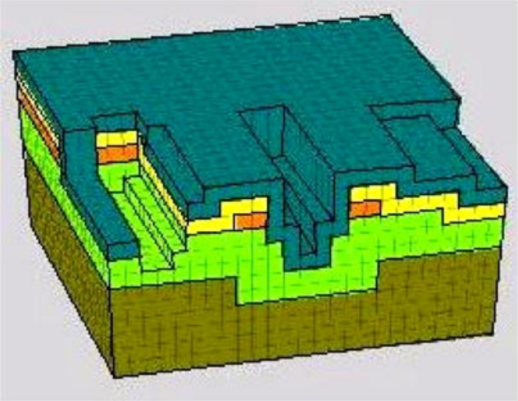

Etch 1
Etch 2
Original Model
Intermediate Model
Intermediate Model
Extracted Layer from Original Model
Extracted Layer from Intermediate Model
Potential Mask
Corrected Mask
Extracted Layer from Original Model
Extracted Layer from Intermediate Model
Potential Mask
Corrected Mask

$$d_1 = \boldsymbol{deposit}(conformal, 4, 1)$$
$$d_2 = \boldsymbol{deposit}(conformal, 2, 2)$$
$$e_1 = \boldsymbol{etch}(regular, 2, \{2, 1\}, 0.25)$$
$$d_3 = \boldsymbol{deposit}(conformal, 1, 3)$$
$$e_2 = \boldsymbol{etch}(regular, 3, \{3, 2, 1\}, 0)$$

Figure 19. A hypothetical surface-micromachined device to illustrate the potential versus corrected masks.

until no more empty cells are found. In the recursive process, whenever a colored cell is encountered, it is marked as an exposed boundary. The Minkowski operation (denoted by $\oplus$) is done by scanning a sphere-structuring element along the exposed boundary. A stack deposit is implemented by finding the top-most cell height and knowing the thickness of the deposited layer, a rectangular prism of voxels is created with appropriate material and added to the model. The via deposit operation uses the flood-fill algorithm. The isotropic etch is similar to the conformal deposit with the difference that the sphere of 'empty' material is swept across the exposed boundary. For an anisotropic vertical etch, from a given mask, a rectangular prism of voxels of suitable size is created and is subtracted from the model.

Some of the results of forward problem are shown in figures 17 and 18. Figures 17(*a*) and (*b*) show the voxel-model of the undercut used in the microhinge created with SUMMiT process. Different gray scale shades (colors) are used to show different layers. The darkly shaded (blue) layer at a distance away from the cylinder at the center is the moving part. As can be seen in figure 17(*b*), after removing the sacrificial layer (lightly shaded (yellow) layer in figure 17(*a*)), the moving part can only rotate with little clearance in the vertical and radial directions. Additional layers are used in the SUMMiT process



$$
\begin{aligned}
d_1 &= \textbf{\textit{deposit}}(conformal, 4, 1)\\
e_1 &= \textbf{\textit{etch}}(regular, 1, \{1\}, 0.5)\\
d_2 &= \textbf{\textit{deposit}}(conformal, 3, 2)\\
e_2 &= \textbf{\textit{etch}}(regular, 1, \{2\}, 0.5)\\
d_3 &= \textbf{\textit{deposit}}(conformal, 1, 3)\\
e_3 &= \textbf{\textit{etch}}(regular, 2, \{3, 2\}, 0)\\
d_4 &= \textbf{\textit{deposit}}(conformal, 1, 4)\\
e_4 &= \textbf{\textit{etch}}(regular, 2, \{4, 2\}, 0)\\
e_5 &= \textbf{\textit{etch}}(regular, 3, \{4, 3, 2\}, 0.25)\\
d_5 &= \textbf{\textit{deposit}}(conformal, 1, 5)\\
e_6 &= \textbf{\textit{etch}}(regular, 2, \{5, 2\}, 0.25)
\end{aligned}
$$

Figure 20. A five-layer part whose masks are to be synthesized for the process shown.

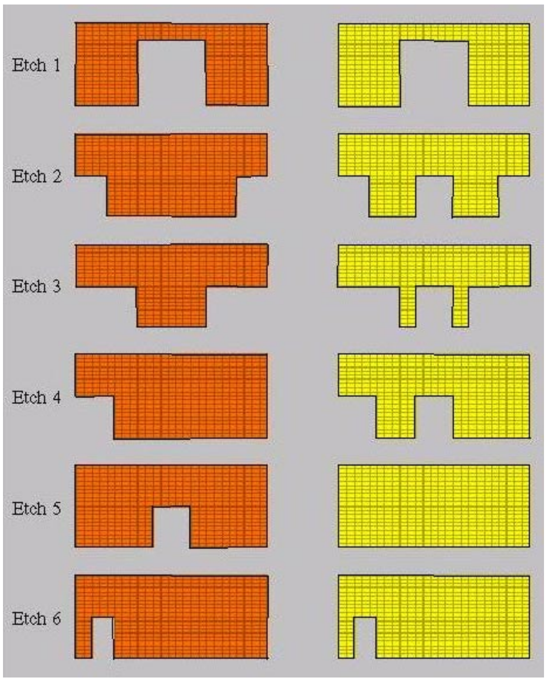


Figure 21. Two valid sets of masks synthesized for the part shown in figure 20.

to create gears with teeth at multiple levels, etc. Figure 18 shows the essential parts of the electrostatic micromotor using the MUMPs process. In the released image (figure 18(*b*)), the central hub (pin-shaped part), the rotor with feet and the stator around the rotor can be seen. Different shades (colors) here correspond to different layers (polysilicon and oxide).

Figures 19–21 show two examples of the inverse problem. Figure 19 illustrates how a potential mask is corrected and validated by following through with the intermediate model as described in sections 6.1 through 6.3. Figure 20 shows a hypothetical five-layer part along with a process flow used to create it. The application of the procedure described in section 6 results in two sets of masks (figure 21) that yield the same 3D model of the part. Thus, this example shows that multiple solutions can be identified when such a situation is encountered.

8. Closure

In this paper, the geometric mask synthesis problem for surface-micromachined devices was posed systematically and solved mathematically using singular value decomposition of a linear system of equations. The main contribution of this paper is the foundation for a systematic method to perform the forward and inverse problems that facilitate the modeling and process planning of MEMS devices. This enables a MEMS designer to automatically generate masks from a geometric model of the MEMS device. Furthermore, it relieves the MEMS designer the necessity to constantly check if a change in the design is compatible with a process when trying to improve the performance of a device. Future modeling work will focus on relaxing some of these assumptions so that the inverse problem can be undertaken for a larger domain of MEMS devices fabricated by processes other than surface micromachining. Finally, while the inverse problem is able to find all solutions for mask openings, it is not able to currently suggest a feasible process that can fabricate the device when no solutions are found. This is another practically useful extension of this work.

Acknowledgments

The grant support DMI-9970021 (RS) and DMI-9970059 (GKA) by National Science Foundation is gratefully acknowledged. The authors wish to acknowledge Andrew Perrin for suggesting the flood-fill algorithm and for implementing it as part of his NSF–REU work at the University of Pennsylvania. Partial support from the Iowa State University Carver Trust Grant is also gratefully acknowledged by RS.

References

Ananthakrishnan V 2000 Part to art: The basis for a systematic geometric design tool for surface micromachined MEMS *Masters thesis* University of Toledo Toledo OH USA

Chandru V and Manohar S 1997 Volume modeling for emerging manufacturing technologies *Sadhana* **22** 199–216

Dixit H, Kannapan S and Taylor D L 1997 3D geometric simulation of MEMS fabrication processes: A semantic approach *Proc. 4th ACM Symposium on Solid Modeling and Applications* pp 376–87

Fatikow S and Rembold U 1997 *Microsystem Technology and Microrobotics* (Berlin: Springer)

Hasanuzzaman M and Mastrangelo C H 1996 Process compilation of thin film microdevices *IEEE Trans. Comput.-Aided Des. Integra. Circuits Syst.* **15** 745–64

Ho C P, Plummer J D, Hansen S E and Dutton R W 1983 VLSI process modeling—SUPREM III *IEEE Trans. Electron Devices* **30** 1438–53

Hubbard T and Antonsson E K 1994 Emergent faces in crystal etching *J. Microelectromech. Syst.* **3** 19–28

Hubbard T and Antonsson E K 1997 Cellular automata in MEMS design *Sensors Mater.* **9** 437–48

Koppelman G M 1989 OYSTER: A three-dimensional structural simulator for microelectromechanical design *Sensors Actuators* **20** 179–85

Koppelman G M and Wesley M A 1983 OYSTER: A study of integrated circuits as three-dimensional structures *IBM J. Res. Dev.* **27** 149–63

Kumar V and Dutta D 1998 Approach to modeling and representation of heterogeneous objects *ASME J. Mech. Des.* **120** 659–67

Lee C-Y and Antonsson E K 2000 Self-adapting vertices for mask layout synthesis *Modeling and Simulation of Microsystems Conference (San Diego, March 27–29)*ed M Laudon and B Romanowicz pp 83–6

Li H and Antonsson E K 1998 Evolutionary techniques in MEMS synthesis *Proceedings of the ASME Design Engineering Technical Conferences (Atlanta, GA)*

Lieberman H 1998 How to color in a coloring book *SIGGRAPH 78 (Atlanta GA, August)* (New York: ACM) pp 111–6

Ma L and Antonsson E K 2000 Automated mask-layout and process synthesis for MEMS *Modeling and Simulation of Microsystems Conference (San Diego, March)* ed M Laudon and B Romanowicz pp 20–23

Madou M J 1997 *Fundamentals of Microfabrication* (Boca Raton FL: CRC Press)

Osterberg P M and Senturia S D 1995 MEMBUILDER: An automated solid model construction program for micromechanical structures *Proc. Inter. Conf. Solid State Sensors and Actuators and Eurosensors IX* vol 2 pp 21–4

Press W H, Teukolsky S A, Vetterling W T and Flannery B P 1993 *Numerical Recipes in C: The Art of Scientific Computing* (Cambridge: Cambridge University Press)

Senturia S D, Harris R M, Johnson B P, Kim S, Nabors K, Shulman M A and White J K 1992 A computer-aided design system for microelectromechanical systems *IEEE/ASME J. Microelectromech. Syst.* **1** 3–13

Senturia S D 1998 CAD challenges for microsensors, microactuators, and microsystems *Proc. IEEE* **86** 1611–26

Smith A R 1979 Tint fill *SIGGRAPH 79 (Chicago IL, August)* (New York: ACM) pp 276–83

Sniegowski J J 1996 Multi-level polysilicon surface micromachining technology: Applications and issues *Invited presentation at the 1996 ASME International Mechanical Engineering Congress and Exposition (Atlanta GA, Nov)*

Strang G 1988 *Linear Algebra and Its Applications (San Diego: Harcourt Brace Jovanovich)*

Strasser E and Selberherr S 1995 Algorithms for cellular based topography simulation *IEEE Trans. Comput.- Aided Des. Integr. Circuits Syst.* **14** 1104–14

Zaman M H, Carlen E T and Mastrangelo C H 1999a Automatic generation of thin film process flows. Part I: Basic algorithms *IEEE Trans. Semiconduct. Manuf.* **12** 116–28

Zaman M H, Carlen E T and Mastrangelo C H 1999b Automatic generation of thin film process flows. Part II: Recipe generation, flow evaluation, and system framework *IEEE Trans. Semicond. Manuf.* **12** 129–38