Automatic Optimization Algorithm for a Direct 2D and 3D Mesh Generation from the Layout Information

E. Gnani, F. Ghidoni, M. Rudan

E. De Castro Advanced Research Center on Electronic Systems (ARCES), and
Department of Electronics, Computer Science and Systems (DEIS)
University of Bologna, Viale Risorgimento 2
40136 Bologna, Italy
Tel. +39-051-20-93773, Fax. -93779
E-MAIL egnani@deis.unibo.it

Abstract

A new algorithm has been designed and implemented, that allows for the automatic generation of an optimum 2D and 3D grid for device simulation from the information available at the circuit-layout level. The tool fills a gap in the top-down design chain of integrated circuits, easing the task of circuit designers who are often unaware of specific features of device-modeling tools.

1 Introduction

Among the different phases of the integrated-circuit design, an important step is the verification that the devices designed at the circuit level are capable of the expected performances. For this, it is necessary to carry out extensive simulations at the device level. This requires in turn the extraction, from the circuit layout, of a number of data to be used as input of the device simulator after suitable manipulation. As far as the problem of the grid generation is concerned, such extraction and manipulation are by no means an easy task. This is true not only because the competence required for it are not necessarily possessed by the circuit designers, but also because the complexity of the typical integrated circuits gives rise to a huge amount of data, whose manipulation on a case-by-case basis is not affordable.

Goal of this work is to automatically define the input parameters needed by the commercial grid generators for device analysis that are routinely used in industrial and research environments. Such definition is based directly onto the information available at the circuit-layout level, and provides a set of parameters that make the grid generator able to produce an optimal grid. The implementation has produced a tool adapted to the typical CAD environment for layout design, which is familiar to the circuit designer. The input to the whole procedure is the information about the available technological process and layout description. For each layer of the chip, the algorithm computes an optimal relation between the geometric dimensions of the individual structures within the layer and the minimum and maximum grid element's size needed by the grid generator. The specific input data are the chip's mask geometry and doping profiles. Such data, along with the refinement specifications given to the grid generator, determine the shape and density of the generated grid.
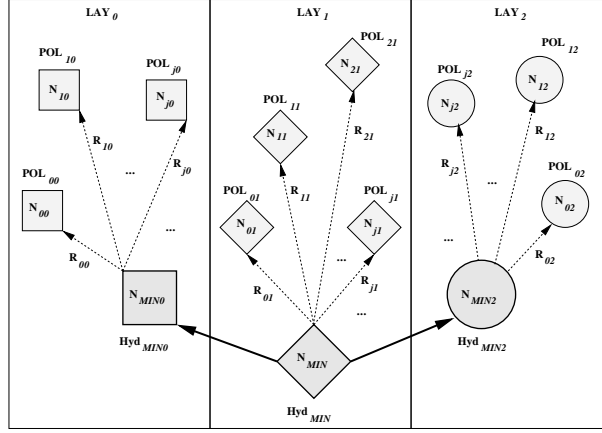
Figure 1: Two-level tree structure used by the algorithm to calculate the minimum and maximum mesh-element size of each polyhedron. Different geometric shapes have been adopted to distinguish the layers. The branch relations of the first level are drawn in solid lines, those of the second level in dashed lines.

2 Layout analysis and chip description

A layout is formally described by an array of layers (chip masks), each characterized by a doping profile (i.e., nwell, pwell, buried layer, etc.) and/or material type (i.e., silicon, oxide, poly, etc.). For a given technology, the set of process-related geometrical variables is fixed for the vertical z axis (normal to the chip), while the coordinates in the x, y plane (parallel to the chip) are used to define the dimension and position of each structure within a layer.

Here, a structure will be referred to as "polyhedron", because any diffusion or implantation region is described by a prismatic polyhedron with a height equal to the layer's depth and a base equal to the implantation/diffusion window augmented by the lateral diffusion. The choice of the refinement level of a polyhedron is based upon information such as the electrical and physical behaviour of the polyhedron's layer, and the geometrical dimension and shape of the polyhedron. From the geometrical standpoint, parameters able to carry the dimension and shape information are needed. The base area and perimeter of the jth polyhedron belonging to the kth layer are denoted by A_{jk} and P_{jk}, respectively. Both are important because the area in itself is not sufficient for the definition of the mesh-element size, since a finer mesh is needed if an irregular edge is present, while the perimeter in itself does not give information about the size of the occupied area. A geometric parameter easy to compute and able to provide the correct geometric information is the *hydraulic diameter* of the polyhedron's base, defined by $H_{jk} = 4A_{jk}/P_{jk}$. The hydraulic diameter is extensively used in fluid-dynamics investigation because it provides a useful characteristic length for the section of a flow pipe. It is related to the average diameter of a plane section and allows one to compare the features of polygons of different width and shape.

Table 1: Time used by the algorithm for generating the input files for the grid generator on a PentiumIV, 1-GB RAM PC, using two different technologies (3D case). The target total element number E_T has always been fixed at 40.000.

Smart-power technology	
# polyhedrons	Time
66 polyhedrons 6 contacts	425 ms
6 polyhedrons 6 contacts	93 ms
CMOS digital technology	
# polyhedrons	Time
208 polyhedrons 386 contacts	1380 ms
50 polyhedrons 20 contacts	342 ms
10 polyhedrons 3 contacts	88 ms

3 The algorithm for an optimal grid-parameter calculation

The algorithm presented here defines a two logical-level tree structure, whose nodal elements are parameters related to the minimum and maximum element size of each polyhedron. The branch connecting two nodes defines the relations between the nodal values. A schematic representation of the tree is shown in Fig. 1. The root node corresponds to the polyhedron H^{m} with the minimum hydraulic diameter, while $N_{\mathrm{M}}^{\mathrm{m}}$ is the maximum number of nodes that can be accepted for this polyhedron. This particular value is taken as a reference for the next calculations. Once all relations among the nodes are defined, the minimum and maximum grid element of each polyhedron are functions of the root node value $N_{\mathrm{M}}^{\mathrm{m}}$. For a given layout, once the refinement parameters and electrical properties of the polyhedron under investigation have been collected, the resulting mesh density is estimated from the properties of the grid generator at hand. In fact, indicating with E_T the target total element number of the mesh, a relation $E_T = f(N_{\mathrm{M}}^{\mathrm{m}})$ holds embodying the branch relations of the tree structure and the meshing rules of the grid generator. Although the functional dependence of f on $N_{\mathrm{M}}^{\mathrm{m}}$ is prescribed by the grid generator's meshing rules, f is in any case a monotonically increasing function of $N_{\mathrm{M}}^{\mathrm{m}}$. In conclusion, its inversion provides $N_{\mathrm{M}}^{\mathrm{m}}$ in terms of a given target E_T. From this, the maximum-minimum mesh element range for each polyhedron is calculated according to the branch relations. It is also importante to note that the algorithm presented here allows for the definition of a target total element number, which is typically missing in the standard software for mesh generation.
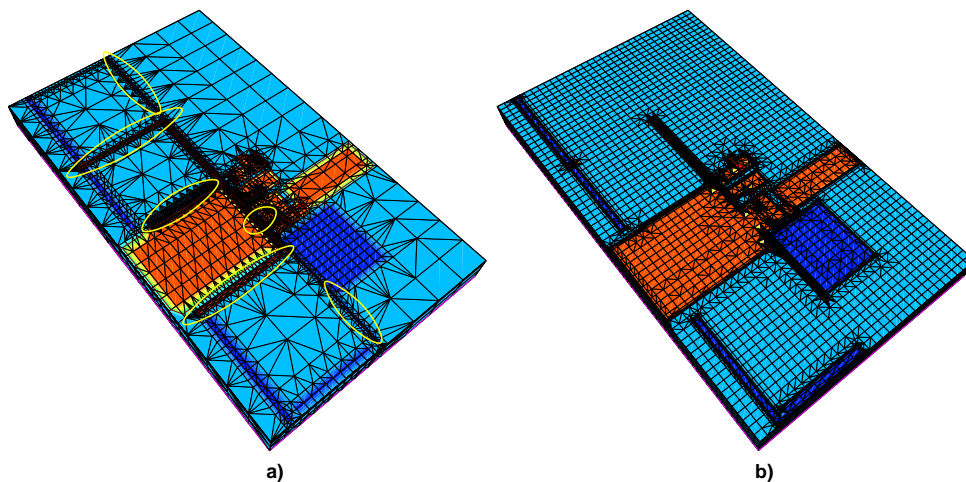
Figure 2: Meshes obtained for a 3×5 mm^2 test chip with (a) and without (b) the optimization algorithm. The ellipses in (a) mark the regions that are correctly meshed thanks to the algorithm. Such regions are completely disregarded by a standard meshing using the same target E_T (b). The grids have been generated with the ISE TCAD mesh software©.

4   Results and conclusions

The algorithm for the automatic generation of the input parameters for a grid generator has been tested with different technologies. The time needed for the algorithm to generate the input files in the case of a 3D mesh are shown in Table 1 for different chip sizes and polyhedrons' numbers. Two different technologies (Smart power and Digital CMOS) have been used. The meshes obtained for a 3×5 mm^2 Smart-power chip with (a) and without (b) the present optimization algorithm are compared in Fig. 2, where the ellipses indicate small well regions. It is seen that the well regions are correctly meshed in (a), whereas they are fully disregarded in (b). Obviously, both cases have the same target E_T.

This work has presented the implementation of a CAD tool able to automatically generate, from the information available at the circuit-layout level, an optimum grid for device simulation. Thanks to this feature, the tool fills a gap in the top-down design chain of integrated circuits, easing the task of circuit designers who are often unaware of specific features of device-modeling tools. The interface with the CAD environment for layout design on one end, and that with the grid generator on the other end, are typical of the standard design tools of integrated circuits. This makes the new tool suitable for application in industrial and research environments. The implementation is based on a reliable algorithm that since long has successfully been used in the field of fluid dynamics. Examples of application to realistic cases are given, showing the efficiency of the tool.