Dynamic Mesh Adaptation for Three-Dimensional Electromigration Simulation

W. Wessner, H. Ceric, J. Cervenka, and S. Selberherr
Institute for Microelectronics, TU Vienna, Gußhausstraße 27–29/E360, 1040 Wien, Austria
Phone: +43-1-58801/36031, Fax: +43-1-58801/36099, E-mail: wessner@iue.tuwien.ac.at

***Abstract*— It is well known that the simulation of time-to-failure for copper (Cu) metal lines requires modeling of vacancy electromigration as well as void nucleation, growth, and movement. Because of the complexity of this problem, different approximate approaches to the physical formulation and solution appear in the literature. Based on our work for two-dimensional electromigration induced void migration and our experience on mesh adaptation techniques we present a computational method for three-dimensional tetrahedral mesh refinement and hierarchical coarsement according to the demands of advanced electromigration simulation.**

I. Introduction

Implementation of copper and low-K materials as major components of interconnect structures has resulted in the necessity to create new current design rules to ensure chip immunity to electromigration induced failures. This practical demand causes an enormous interest in understanding the fundamental reliability properties of interconnect copper metalization.

Modeling the micro-mechanics of electromigration caused void evolution is a long-standing scientific problem. It began with sharp interface models requiring an explicit finite element tracking of void surfaces during the course of evolution. Later, prompted by the complexity of void surfaces, diffuse interface models were introduced [1]. Diffuse interface models circumvent computationally costly surface tracking by application of a smooth order parameter field for representation of the void structures. We solve the diffuse interface model governing equation with a finite element scheme coupled with a powerful mesh adaptation algorithm. The robustness of the developed finite element approach with respect to the underlying mesh structure makes it possible to efficiently simulate the damage induced by electromigration in complex interconnect geometries.

II. Electromigration

Electromigration is the transport of material caused by the gradual movement of the ions in a conductor due to the momentum transfer between conducting electrons and diffusing metal atoms. All work in this field was pioneered by James R. Black [2], who set the basis for research in this area and after whom the Black semi-empirical equation

$$MTF = \frac{A}{j^n} \exp\left(\frac{E_a}{k_B T}\right) \tag{1}$$

is named, where MTF is the median time-to-failure, A is a pre-exponential constant. E_a is the activation energy, k_B is Boltzmann's constant, T is the temperature, j is the current density, and n is the so called current density exponent. The problem with this formula is, that the activation energy E_a and the current density exponent n have to be determined experimentally and therefore its validity is limited to specific test configurations. To understand the phenomenon of electromigration a more physical approach, based on the main diffusion scenes within interconnect structures must be applied.

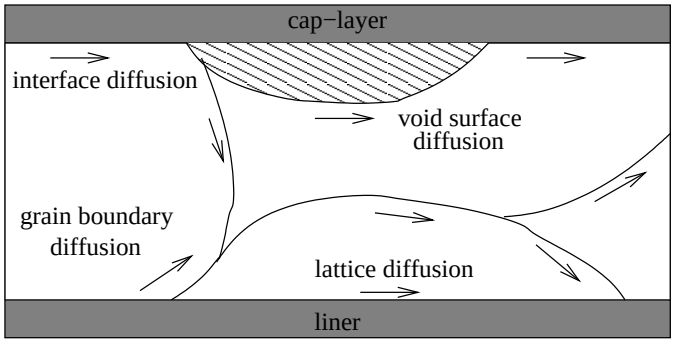


Figure 1: Mass transport of metal atoms along different diffusion paths in a typical Cu interconnect line.

A. Fast-diffusion-paths

The mechanism of electromigration can be explained by the interaction of two counteracting forces:

- activated, positively charged metal ions suffer a force to the cathode direction,
- the electrons, while moving to the anode, transmit an impulse to the metal ions.

Because of a shielding effect of the conduction electrons on metal ions, the first force generally is small. This means that in a metal lattice a mass flux of ions exists, which is directed parallel to the so called *electron wind*. The mass flow takes place in the form of diffusion along interfaces, as grain boundaries and surfaces, and by volume diffusion. In Cu interconnects, grain boundary and interface diffusion are the dominating transport mechanisms at normal temperature operating conditions [3]. A schematic overview of different *diffusion-fast-paths* is shown in Figure 1. Local mass flux divergences cause the formation, growth, and movement of voids and hillocks. In this work we mostly focus on modeling of void movement within an arbitrary interconnect structure and the development of a powerful mesh adaptation technique, which allows to use a diffuse interface model for the description of the metal void interface.

B. Diffuse interface model

In diffuse interface models void and metal area are presented through an order parameter ϕ which takes values $+1$ in the metal area, -1 in the void area, and $-1 < \phi < +1$ in the void-metal interface area. The model equations for the void evolving in an interconnect line are the balance equations for the order parameter ϕ

$$\frac{\partial \phi}{\partial t} = \frac{2D_{\mathrm{s}}}{\epsilon\pi}\nabla \cdot (\nabla\mu - |e|Z^*\nabla V), \tag{2}$$

$$\mu = \frac{4\Omega\gamma_{\mathrm{s}}}{\epsilon\pi}(f'(\phi) - \epsilon^2\Delta\phi), \quad \text{and} \tag{3}$$

$$\nabla \cdot (\sigma(\phi)\nabla V) = 0 \tag{4}$$

for the electrical field, where μ is the chemical potential, $f(\phi)$ is the double obstacle potential, Z^* is the effective valence, e is the charge of an electron and ϵ is a parameter controlling the void-metal interface width which is approximately $\pi/2$. γ_s is the surface energy, Ω marks the volume of an atom, and D_{s} is given by an Arrhenius law

$$D_{\mathrm{s}} = \frac{D_0\delta_s}{k_BT}\exp\left(\frac{-Q_s}{k_BT}\right), \tag{5}$$

where, δ_s is the thickness of the diffusion layer, k_BT has the conventual meaning, Q_s is the activation energy for the surface diffusion, and D_0 is the pre-exponential for mass diffusion.

In order to handle properly thin interfaces one needs a very fine local mesh. Equations (2)-(4) are solved by means of the finite element method [4] on the sequence of the meshes $\Lambda_h(t_0 = 0), \Lambda_h(t_1), \cdots, \Lambda_h(t_N)$ each one adapted to the position of the void-metal interface from the previous time step. The initial mesh $\Lambda_h(0)$ is produced by refinement of a basic mesh according to the initial profile of the order parameter ϕ.

III. Mesh adaptivity methods

In general there are three mesh adaptation methods, namely the *r*-, *h*-, and *p*-method [5].
Using the *r*-method, the mesh connectivity is unchanged. Instead, node relocation is used to move the mesh nodes either by means of weighted barycentric smoothing based on the location and the weight of the nodes in some neighborhood, or by means of element distortion. The criteria (weights) governing these operations are obtained by analyzing the actual solution.
h-method adaptation is defined in terms of local or global mesh enrichment by means of refining (by partitioning) or coarsening selected elements or all the elements in a mesh.
The *p*-method approach is based on an invariant mesh (in terms of points (nodes, vertexes) and elements) and adjusts the degree (in terms of the interpolation functions) of the finite elements constructed on the mesh elements as a function of the current solution analysis.
To achieve a local partitioning which is required for the *h*-method three configurations of inserting a new vertex (node) on the three-dimensional simplex are possible. It is likely to define a new vertex along an edge, on a face, or inside the tetrahedron (see Figure 2).
In our implementation we use a special kind of an *h*-method adaptation where only mesh refinement by inserting a vertex on the longest edge of the refinement tetrahedron is allowed [6], [7].

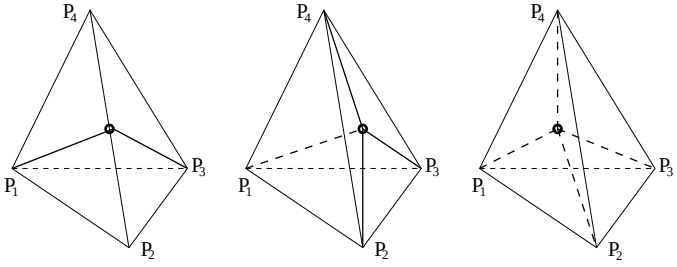


Figure 2: Local three-dimensional simplex partitioning.

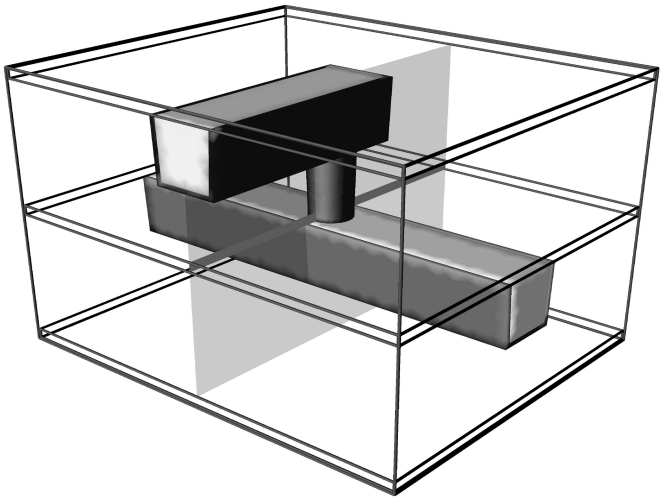
A. Recursive approach

When bisecting a tetrahedron, a particular edge – called the *refinement edge* – is selected and split into two edges by a new vertex, cf. Figure 2 (left). This mesh refinement strategy was enhanced by a recursive tetrahedral bisection approach which produces quite regular refined elements under consideration of the geometric element quality, which allows a very smooth transition from coarse to fine mesh elements [8].
To guarantee a conforming mesh during the refinement procedure all tetrahedrons that share a common refinement edge are divided. A tetrahedron is said to be *compatibly divisible* if its refinement edge is either the refinement edge of all other tetrahedrons sharing the refinement edge or its part of the boundary of the domain. If a tetrahedron is compatibly divisible, we divide the tetrahedron and all other refinement edge sharing tetrahedrons simultaneously. If a tetrahedron is not compatibly divisible, we ignore it temporarily and divide a neighbor tetrahedra by the same process first. This leads to the recursive algorithm.

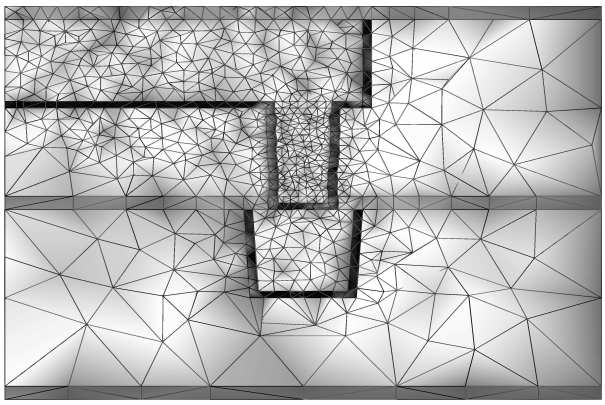
The choice of the longest edge as the refinement edge appears to be good from a numerical point of view, but on the other hand, the recursion can become very large. The refinement of a particular element can enforce the bisection of remote elements. For the purpose of a local refinement it appears to be more convenient to spend some preprocessing time and find the refinement edges where almost all tetrahedrons are compatibly divisible, which implies that recursion will be as small as possible.

B. Hierarchical mesh coarsement

According to the needs of void movement simulation during electromigration also a hierarchical *mesh coarsement* part was introduced. The basic idea behind this coarsement strategy is that a previous refinement step is reversed. This

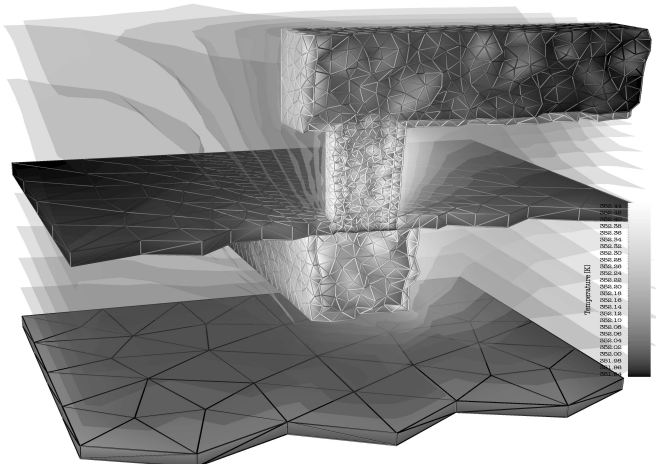


(a) Simulation domain, with cross cutting plane.

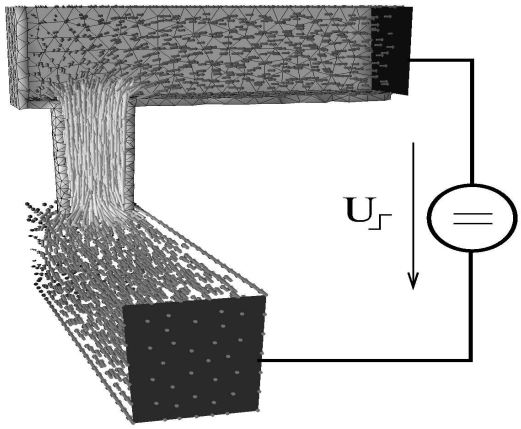


(b) Cross cut with mesh elements.

Figure 3: Three-dimensional interconnect electromigration simulation domain with trapezoid shaped tantalum (Ta) covered copper (Cu) lines, round conical via, and horizontal silicon carbide (SiC) etch stop layers embedded in silicon dioxide (SiO_2).



(a) Temperature distribution.



(b) Electrical field distribution.

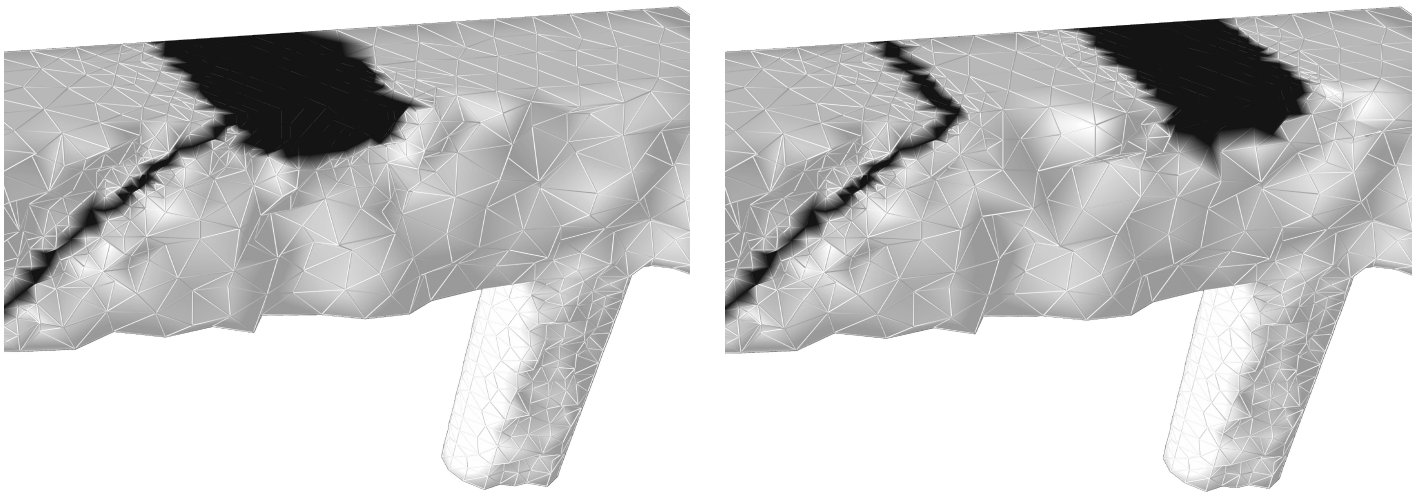
Figure 4: Temperature distribution and electrical field calculated on the simulation domain by applying appropriate electrical and thermal boundary conditions. Both quantities influence void nucleation.

procedure can be handled easily by introducing a hierarchical element structure as shown in Figure 6.

During transient simulation the position of the interface is detected after every timestamp and the mesh resolution is controlled. Too coarse elements are refined by recursive tetrahedral bisection. Regions which have been refined in a previous time step and which are not covered by the void-metal interface area are loaded into the coarsement module. Due to the properties of our hierarchical element data structure, the initial (before the refinement) mesh constellation can be recovered easily. It's in the nature of this approach that the initial mesh is always part of the current mesh and no coarser mesh than the initial one can be reached. This seems to be a handicap but on the other hand the most coarse mesh is defined by the initial one and therefore the lowest spatial resolution is well known which helps to bound the numerical error introduced by the mesh [9].

IV. Examples

Our mesh adaptation techniques are demonstrated on a typical three-dimensional interconnect structure (see Figure 3) with trapezoid tantalum covered copper lines with etch stop layers embedded in silicon dioxide. For the generation of the initial unstructured mesh, we used Gmsh which is a finite element mesh generator (primarily Delaunay) with built-in pre- and post-processing facilities [10]. This mesh generator allows to control the initial spatial resolution of the simulation domain quite well and therefore a low error bound can be guaranteed (cf. section III-B). Temperature and electrical field distribution



(a) Void is formed near Cu grain boundary.

(b) Void moved towards the interconnect via.

Figure 5: During void formation and movement a dynamic mesh adaptation scheme is used to guarantee a good spatial resolution on the void copper interface. Also copper grain boundaries are taken into account and an appropriately fine mesh was computed.

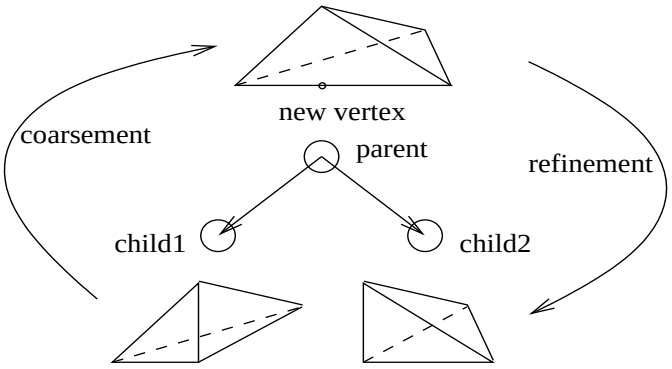


Figure 6: Data structure of refinement-coarsement scheme.

for the test circuit are given in Figure 4. As mesh adaptation example a typical void movement is chosen where the void was formed from an Cu - cap-layer interface near a Cu grain boundary. Figure 5(a) shows an initial void formation with a fine mesh near the void-metal interface. During the transient simulation the void moves towards the interconnect via and the mesh is adapted dynamically as depicted in Figure 5(b). The recursive refinement procedure works quite well and shows in practice a quite local and robust behavior being important for the number of overall mesh elements which have a direct impact on computational costs like memory usage and simulation time.

V. Conclusion

We presented a dynamic mesh refinement and coarsement strategy which fulfills the demands of electromigration simulation based on a finite element diffuse interface approach. As refinement procedure a recursive tetrahedral bisection method was used, which gives in practice satisfactorily local behavior, and is therefore a good tradeoff between element quality and computational effort. For the coarsement step a hierarchical *refinement step-back* procedure was chosen, which keeps the data structure complexity low and makes the adaptation fast and robust. Based on this dynamic mesh adaptation techniques a typical void movement within an interconnect line has been shown.

References

[1] M. Mahadevan and R. Bradley, Phys. Rev. B **59**, 11037 (1999).

[2] J. Black, IEEE Transactions On Electron Devices **ED-16**, 338 (1969).

[3] A. Buerke, H. Wendrock, and K. Wetzig, Cryst. Res. Technol. **6-7**, 721 (2000).

[4] O. Zienkiewicz and R. Taylor, *The Finite Element Method*, fourth ed. (McGRAW-HILL Book Company Europe, Shoppenhangers Road, Maidenhead, Berkshire, England, 1989), Vol. 1.

[5] P.-L. George and H. Borouchaki, *Delaunay Triangulation and Meshing*, 1 ed. (Editions HERMES, 8, quai du Marché-Neuf, 75004 Paris, France, 1998).

[6] M.-C. Rivara and G. Iribarren, Mathematics of Computation **65**, 1485 (1996).

[7] D. N. Arnold, A. Mukherjee, and L. Pouly, SIAM J. Sci. Comput. **22**, 431 (2000).

[8] I. Kossaczký, ELSEVIER Journal of Computational and Applied Mathematics 275 (1994).

[9] C. Johnson, *Numerical solution of partial differential equations by the finite element method* (Cambridge University Press, 32 East 57th Street, New York, NY 10022, USA, 1987).

[10] C. Geuzaine and J.-F. Remacle, Gmsh: a three-dimensional finite element mesh generator , `http://www.geuz.org/gmsh/`.