U N I V E R S I T Y O F R E A D I N G

Interfacing of an oxidation model with the two-dimensional Moving Finite Element Model of dopant diffusion

by

Neill S. Cooper

Numerical Analysis Report No. 6/88

July 1988

University of Reading
Mathematics Department
P O Box 220
Reading

M A T H E M A T I C S D E P A R T M E N T

ABSTRACT

The mathematical formulation of the extension of the Reading two-dimensional Moving Finite Element model of dopant diffusion in silicon to include oxidation is described. Non-dimensionalisation of the equations shows that the oxidation and silicon diffusion problems can be seperated. Thus the only information about the oxidation required by the diffusion model is the location of the oxide boundary. This moving boundary effects both the location of the interfacial nodes and also the dopant concentrations there, and can lead to element folding. Results produced by the model when it is extended to include such a moving boundary are presented, illustrating both its potential and its limitations.

1. Introduction

The dimensions of silicon integrated circuits are continually being reduced, with critical features now as small as one micron. This places stringent requirements on silicon processing techniques, such as ion implantation, oxidation, annealling, lithograthy and etching, so that transistor junctions are well defined and correctly located. Thus the task of developing a manufacturable, small-geometry silicon integrated circuit process is extremely complex and expensive. It is important, therefore, to reduce the number of experimental tests to a minimum by using predictive models of the main physical processes (Godfrey, 1986).

To this end models have been developed in the Mathematics Department at the University of Reading over a number of years, using the Moving Finite Element (MFE) method devised by Miller and Miller (1981). This technique has the advantage over conventional fixed finite difference and finite element techniques of being able to concentrate nodes where they are required at any time, thus saving on computational time.

In some regions of silicon semiconductor structures 1-dimensional modelling gives an accurate simulation of the dopant diffusion during thermal annealling (Ho et al., 1983). Such models have been developed using the MFE method (Baines et al., 1986) with subsequent adaption to also include the effect of a moving oxide boundary (Moody and Please, 1987). This model is at present being extended to simulate multiple dopant diffusion (S. Chynoweth, private communication).

The applicability of 1-dimensional models is, however, limited to planar regions and, as device dimensions are reduced, the need for 2-dimensional models increases. A 2-dimensional model of dopant diffusion has been developed (M. Baines, private communication), to be implimented within the Alvey 066 software. However the extant code only models ion implantation (using Gaussian and error functions) and dopant diffusion. The important process of thermal oxidation, with the resultant moving boundary and dopant segregation at the boundary, was not included. It is to the interfacing of an oxidation model with the diffusion model that this report is addressed. As with other numerical integration techniques, the extension of the 1-D MFE method to 2-D is not as simple as it would first appear. For as well as greatly increasing the number of elements and thus causing a reduction in resolution, there are other difficulties related to the method and also to the problem:

(1) While in 1-D the local MFE method is equivalent to the global method, in 2-D this is not the case (Sweby, 1987).

(2) While in 1-D element folding is a simple process (Moody and Please, 1987), in 2-D it can take many forms and it is difficult to include all posibilities in the model (see section 3.2).

(3) The growth of the oxide is not purely in the vertical direction but also has a horizontal component.

In section 2 the models used to simulate diffusion, oxidation and segregation are described. These are then non-dimensionalised which highlights negligible terms, resulting in a decoupling of the oxide

diffusion problem from the silicon diffusion problem. Thus the silicon model only needs to know how the location of the oxide interface varies with time. In section 3 the technique to impliment this knowledge is described, with the MFE model needing to calculate both the new nodal position and, more difficult, the new interfacial dopant concentration. The moving upper boundary also results in elements folding and the methods used to cope with this are also described in this section.

Results of using this method are given in section 4. These results also highlight some shortcomings in the present model and suggestions for improving it are given in section 5. Details of the alteration to the numerical code are given in the Appendix, section 8.

2. The Mathematical Models

2.1 Diffusion in Silicon.

The model used here of 2-D dopant diffusion in silicon is that developed at Reading University (M. Baines, personal communication). It is based on the dopant diffusion equation

$$\frac{\partial c_s}{\partial t} = \nabla\{D_s(s_s)\nabla c_s\} \qquad (2.1)$$

where $c_s(x,y)$ is the total dopant concentration at the point (x,y) in the silicon and $D_s(c_s)$ is the diffusion coefficient at concentration c_s. The diffusion coefficient also depends upon dopant species (arsenic, boron, phosphorus or antimony), temperature, clustering effects and whether oxidation is taking place or not (Godfrey, 1984). In the case of the diffusion of a single dopant species the effect of clustering (precipitation) can be simulated without considering the clustered dopant as a seperate species. The technique for so doing is given by Baines et al. (1986).

The existing 2-D MFE code simulated the implantation of dopant followed by its diffusion, obeying equation (2.1) with the boundary condition of zero gradient in the dopant concentration at the edges of the domain. This model has been used to successfully reproduce the first benchmark problem, posed by GEC within the ALVEY 066 program, and has been implimented within the ALVEY 066 software package 'TAPDANCE'. However the model has difficulty in simulating the second benchmark problem, in which the implantation has realistic range, standard deviation and straggle and therefore sharper concentration gradients and smaller radii of curvature.

2.2 Oxidation.

Simulation of the oxide growth, and the diffusion of dopant within it, is a very complex problem (Chin et al., 1983) and will not be tackled here. Instead it is assumed that information about the oxide comes from one of two sources: An external oxide model (in the case of the ALVEY 066 software this will be provided by University College, Swansea) or else an analytical model.

2.2.1 The Analytical Model.

The analytical model is that of Penumalli (1981), which is used by the TITAN process simulator (Girodolle and Martin, 1987). Penumalli's model is a two-dimensional extension of the classic Deal and Grove (1965) one-dimensional model used by Moody and Please (1987), with the lateral variation in oxide thickness being described by an error function. Figure 1 illustrates the model together with an explanation of the notation used here.

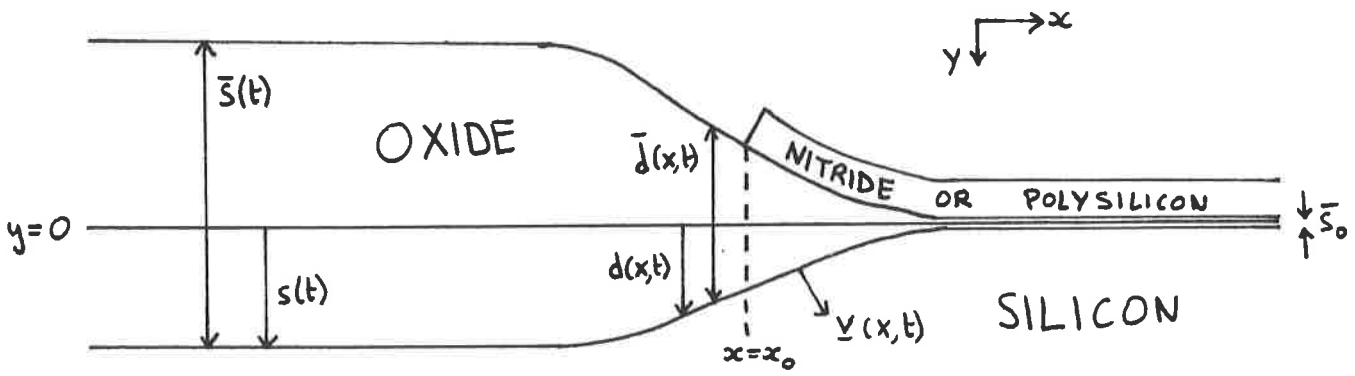


Figure 1

If the initial oxide thickness is s_o, then the oxide thickness at a large distance from the nitride (or polysilicon) mask at time t is given by the Deal and Grove (1965) formula:

$$\bar{s}(t) = \tfrac{1}{2}\left\{\sqrt{A^2 + 4[\bar{s}_o^2 + A\bar{s}_o + Bt]} - A\right\} \qquad (2.2)$$

where B and B/A are the parabolic and linear temperature-dependant rate constants respectively. The general oxide thickness $d(x,t)$, at any location x is then defined by

$$\bar{d}(x,t) = \left\{\frac{\bar{s}(t) - \bar{s}_o}{2}\right\}\left[1 + \mathrm{erf}\left\{\frac{2(x_o - x)}{\alpha(\bar{s}(t) - \bar{s}_o)}\right\}\right] + \bar{s}_o \qquad (2.3)$$

where α is the lateral extension of the oxide (0.6 for a nitride mask, 2.0 for polysilicon).

The location of the silicon-oxide boundary is defined by

$$d(x,t) = \bar{d}(x,t)/\gamma$$

where γ is the ratio of the silicon dioxide to silicon specific volumes ($\gamma = 2.27$). (Note that the y-axis is oriented downwards to be consistant with the 2D-MFE code).

To include oxidation within the MFE environment we need to know not just the oxide interface at a given time but also its velocity, which thus defines the velocity of the boundary nodes. For the analytical formula of Penumalli (1981), this velocity is not uniquely defined. Thus we need to make an assumption about the direction of oxide encroachment into the silicon, and we will assume that it is normal to the silicon-oxide interface. Thus the velocity of the silicon-oxide

interface, $v(x,t)$, is defined as

$$\underline{v}(x,t) = \frac{\partial d}{\partial t}\left\{\underline{j} + \left[\frac{\partial d}{\partial x}\right]\underline{i}\right\} / \left\{1 + \left[\frac{\partial d}{\partial x}\right]^2\right\} \qquad (2.4)$$

where

$$\frac{\partial d}{\partial x} = -\frac{2}{\alpha\sqrt{\pi}} \exp(-\zeta^2) \qquad (2.5)$$

$$\frac{\partial d}{\partial t} = \frac{\dot{s}}{2}[1 + \text{erf}\ \zeta] - \frac{\dot{s}\zeta}{\sqrt{\pi}} \exp(-\zeta^2) \qquad (2.6)$$

$\zeta = 2(x_o - x)/\{\alpha(s(t)\bar{s}_o)\}$ and $s = \bar{s}/\gamma$ (as with d), with $\underline{i}$ and $\underline{j}$ being the unit vectors in the x and y directions respectively.

To model dopant diffusion within the oxide we assume that the motion due to oxide growth is purely vertical. With this assumption then c_o, the dopant concentration within the oxide, obeys

$$\frac{\partial c_o}{\partial t} = D_o \nabla^2 c_o + (\gamma - 1)\frac{\partial d(x,t)}{\partial t}\frac{\partial c_o}{\partial y} \qquad (2.7)$$

where D_o is the diffusion coefficient within the oxide (assumed independant of concentration), and y is depth into the silicon, with the origin being the level prior to oxidation. In keeping with previous work, clustering effects in the oxide can be ignored as the dopant concentrations are low.

2.2.2 The Numerical Model.

In this alternative to the Penumalli model, data for the moving silicon-oxide interface was supplied in numerical form from the Swansea Oxidation Model (J. Waddell et al., 1987). The data comprise time, node number, location. The interfacial velocities were calculated by numerical differentiation of this data, while the dopant concentration is not modelled (see section 2.4).

2.3 Boundary Conditions.

Consistant with the previous silicon modelling work at Reading University using the MFE code, we assume no flux leakage at external boundaries, and thus $\frac{\partial c}{\partial n} = 0$, with c being either c_s or c_o and n being the normal to the boundary. Note that the domain used needs to be large enough for the oxide interface to be horizontal at the boundaries, otherwise the above boundary condition is not valid.

The internal boundary between the silicon and oxide is assumed to obey a segregation law, and thus the concentrations in the silicon and oxide obey the equations:

$$\left[D_s(c_s)\underline{\nabla}c_s + \underline{v}(x,t)c_s\right]\cdot\hat{\underline{v}} = -h\left[c_o - \frac{c_s}{m}\right] \qquad (2.8)$$

$$\left[D_o\underline{\nabla}c_o + \gamma\underline{v}(x,t)c_o\right]\cdot\hat{\underline{v}} = -h\left[c_o - \frac{c_s}{m}\right] \qquad (2.9)$$

where h and m are the boundary transport rate and the equilibrium segregation ratio respectively, while $\hat{\underline{v}}$ is the unit vector in the $\underline{v}$ direction.

The initial dopant distribution in the silicon is be assumed to be Gaussian with depth while having an error function dependance in the lateral direction. The values used here are:

Range	=	0.25 μm
Standard Deviation	=	0.05 μm
Straggle	=	0.05 μm

While these are not physically realistic, they are the values which have consistently been used in developing the 2-D MFE model, and thus the behaviour of the base model is known and understood.

2.4 Non-dimensionalisation.

In their study of 1-dimensional oxidation, Moody and Please (1987) found that by non-dimensionalising the model equations they could show that the diffusion of dopant in the oxide was negligible. The same method should lead to similar conclusions in 2-dimensions. However, the existing 2D MFE code does not use the same scalings, instead it uses the following normalisations:

Concentration	ni	(a function of temperature)
Length	$L = 1\mu m$	
Time	$t = L^2/D_i$	(D_i is a function of temperature).

Using these values equations (2.1), (2.7) - (2.9) can be rewritten in normalised co-ordinates, in which the following transformations are made:

$$(c_s, c_o) = n_i(C_s, C_o)$$
$$(x, y, d) = L(X, Y, D)$$
$$(t) = \tau(T)$$
$$(\underline{v}) = (L/\tau)(\underline{V})$$
$$\left[D_s(c_s), D_o\right] = \left[D_i D_S(C_S), D_i D_o\right] .$$

The result of such substitutions is to leave the equation for diffusion in the silicon (2.1) unchanged, while the remaining equations become:

$$\frac{\partial C_0}{\partial T} = \delta_1 \nabla^2 C_0 + (\gamma - 1) \frac{\partial D}{\partial T} \frac{\partial C_0}{\partial Y} \qquad (2.10)$$

$$\left[D_S(C_S)\underline{\nabla} C_S + \underline{V} C_S\right] \cdot \hat{\underline{V}} = -\delta_3 \left[C_0 - C_S/m\right] \qquad (2.11)$$

$$\left[\delta_1 \underline{\nabla} C_0 + \gamma \underline{V} C_0\right] \cdot \hat{\underline{V}} = -\delta_3 \left[C_0 - C_S/m\right] \qquad (2.12)$$

where

$$\delta_1 = \frac{D_0}{D_i} \qquad \text{and} \qquad \delta_3 = - Lh/D_i \ .$$

Using typical values for D_o, D_i and h leads to the conclusion that some of the terms in these equations are negligible, because:

$$\delta_1 << 1 \qquad \text{and} \qquad \delta_3 >> 1 \ .$$

In view of the smallness of δ_1 (due to diffusion in the oxide being much slower than diffusion in the bulk silicon), the diffusion term in equation (2.10) can be neglected, leading to

$$\frac{\partial C_0}{\partial T} = (\tau - 1) \frac{\partial D}{\partial \tau} \frac{\partial C_0}{\partial Y} \ . \tag{2.13}$$

This has reduced the order of the differential equation in the oxide layer from a second order diffusion equation to a first order advection equation. Thus the surface and edge boundary conditions in the oxide are no longer required.

Similarly, due the diffusion term in equation (2.12) being negligible, this equation can be combined with (2.11), resulting in the elimination of the boundary transport rate:

$$\left[D_S \underline{\nabla} C_S + \underline{V} C_S\right] \cdot \hat{\underline{V}} = \tau \underline{V} C_0 \cdot \hat{\underline{V}} \ . \tag{2.14}$$

In view of the large size of δ_3 and the lack of any term on the LHS of equation (2.12) to balance the RHS then the coefficient of δ_3 must be very small, leading to the equilibrium segregation condition

$$c_o = c_s/m \qquad \text{on} \qquad y = d(x,t) \tag{2.15}$$

Thus the interface boundary condition on the dopant concentration in the

bulk silicon (equation 2.14) can be further simplified so that it does not depend on the dopant concentration in the oxide:

$$\left\{D_S \underline{\nabla} C_S + \underline{V}\left[1 - \frac{\gamma}{m}\right] C_S\right\} \cdot \hat{\underline{V}} = 0 \qquad (2.16)$$

which is identical to the form used by TITAN (Girodolle and Martin, 1987).

As in the one-dimensional case of Moody and Please (1987), these simplifications have created a decoupling between the silicon and the oxide dopant diffusion problems. Thus, in the first order solution, the effect of the oxide boundary on the silicon diffusion problem is independant of the dopant concentration in the oxide, which in turn is directly proportional to the concentration at the silicon boundary. The dopant concentration within the oxide can be determined for any lateral section by direct calculation using the growth rate of the oxide and the temporal evolution of dopant concentration in the silicon at the interface.

As well as using this normalisation, the model is also expressed in transformed co-ordinates in which the dependant variable is no longer concentration but $\phi = c + \ln c$, a velocity potential associated with $D(c)$, (Please and Sweby, 1986). For large c ϕ behaves like c itself, while for small c ϕ behaves like $\ln c$ thus allowing better resolution of the wide range of concentrations simultaneously observed. This transformation leads to altered equations using

$$\delta\phi = \delta c * (c + 1) / c \; . \qquad (2.17)$$

3. Implimentation.

The method used here is to prescribe the interfacial nodal velocity from the oxidation model and then to find the new interfacial nodal concentration using equation (2.16), as in the TITAN process model (Gerodolle and Martin, 1987). This method could lead to slight errors in the oxide depth simulation as the velocity is continually changing and thus a correction to the interfacial node location should be made once the timestep has been calculated. However, with the small time-steps observed in the oxidation-diffusion model described here this error is negligible.

While the calculation of the interfacial node velocity is elimentary, this movement has three more complex effects. Firstly, the rate of change of the dopant concentration at the node is altered due to advection. Secondly, the rate of change of dopant concentration was calculated assuming its gradient was zero (normal to a boundary) or unconstrained. This choice needs to be extended to include the effect of a non-zero gradient at the moving boundary. Thirdly, and most problematic, the movement of nodes forced elements to fold. The first two of these points are dealt with in section 3.1 while the latter is considered in section 3.2.

The effect of segregation on the interfacial dopant concentration is given by equation (2.16). There are two ways of including this in the model:

(1) As a boundary condition, thus specifying concentration gradient at the interface (this was the method adopted by Moody and Please, 1987)

or

(2) As a flux condition thus using conservation of total dopant amount, the method used by the process simulators SUPREM III (Ho et al., 1983) and TITAN (Girodolle and Martin, 1987).

Each technique has been considered and they are described in more detail in section 3.3.

3.1 Interfacial nodal position.

The velocities of interfacial nodes can be specified in the model by overwriting the values which would be predicted by the code. At the heart of the 2D-MFE method the following matrix equation is solved:

$$\begin{bmatrix} \sum A_j & -\sum A_j m_j & -\sum A_j n_j \\ -\sum A_j m_j & \sum A_j m_j^2 & \sum A_j m_j n_j \\ -\sum A_j n_j & \sum A_j m_j n_j & \sum A_j n_j^2 \end{bmatrix} \begin{bmatrix} \dot{a}_i \\ \dot{x}_i \\ \dot{y}_i \end{bmatrix} = \begin{bmatrix} \sum A_j w_j \\ -\sum A_j m_j w_j \\ -\sum A_j n_j w_j \end{bmatrix} \qquad (3.1)$$

where a_i is the concentration at node i, which is located at (x_i, y_i), m_j and n_j are the horizontal and vertical concentration gradients in element j, A_j is the area of element j, the summation is over elements j surrounding node i and w_j are more complex values (see Sweby, 1987, for more details). For interfacial nodes this matrix equation can be reduced to

$$\begin{bmatrix} \sum A_j & -\sum A_j m_j & -\sum A_j n_j \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \dot{a}_i \\ \dot{x}_i \\ \dot{y}_i \end{bmatrix} = \begin{bmatrix} \sum A_j w_j \\ u \\ v \end{bmatrix} \qquad (3.2)$$

where (u,v) is the interfacial velocity (n.b. at the left and right edges of the interface $u = 0$) . This formulation not only specifies the movement of node i to be that of the interface at that location but also takes advection into account in calculating the rate of change of a_i .

As well as these effects, the moving boundary also alters recovery, a technique which allows the second derivatives to be calculated from the nodal concentrations by fitting a pseudo-cubic to the concentration gradients (details are again given in Sweby, 1987). In the 2D-MFE code seven points are used in the recovery calculation for each element: the three nodes, the three mid-edge points and the centroid. In general the gradients at the nodes and mid-edge points are weighted means of the appropriate surrounding elements while at the centroid the value is based on the gradients in the element with added components from the other 6 points. (Experience with various weightings showed that the most robust was to use the length of the element transversal.) This method is adapted at the edges of the domain, for there the gradients are specified by the zero flux condition. Thus in the existing 2D-MFE code the values of the gradients are set to be zero normal to the boundary.

In the case of oxidation, however, the zero flux boundary condition is replaced by equation (2.16). Thus the gradients calculated for the

surface nodes and mid-points needed to be altered to take this into account while leaving the tangential gradients unchanged. When using the transformed variable ϕ, equation 2.16 can be rewritten, using (2.17), as:

$$\frac{\partial \phi}{\partial n} = - |\underline{v}| \left[1 - \frac{\gamma}{m}\right](1 + C)/D(C) \qquad (3.3)$$

Letting w_x and w_y represent the gradients predicted by the recovery technique described above for a given surface node which has velocity (u,v), then the adjusted gradients are

$$w'_x = \frac{(w_x v_y - w_y v_x)v_y}{|v|^2} + \frac{\partial \phi}{\partial n} \frac{V_x}{|v|} \qquad (3.4)$$

$$w'_y = - \frac{(w_x v_y - w_y v_x)v_x}{|v|^2} + \frac{\partial \phi}{\partial n} \frac{V_y}{|v|} \qquad (3.5)$$

except at the left and right sides where $w'_x = 0$.

Integration of the model gives the expected interface motion, as shown in figures 3 to 5. The upper surface of the domain is gradually forced down by the oxidation, with the depth being greater in this case on the left-hand side (where there is no nitride cap). However, the timestep in many cases tends to zero, halving on each successive time-step. This was traced to element folding, and comparison of figures 4a, 4b, 5a and 5b shows that an interfacial element has vanished, having been flattened by the oxidation progression of the interfacial nodes. In other runs where less, and therefore larger, elements were used (the parameter DG being set to 2), this problem does not occur. This is because the gradient in dopant concentration forces the sub-surface nodes deeper into the bulk silicon faster than the oxide intrudes into it.

The logical method to overcome element folding is to allow elements to disappear. Thus the model was adapted to locate very small elements, eliminate them and alter the cross referencing in the node table.

3.2 Treatment of folding elements.

The inclusion of a moving boundary greatly increases the probability of element folding, particularly at the surface. This has been shown to occur in most integrations of the combined oxidation-diffusion model. Elements can fold in many different ways, as is illustrated by figure 2, but normally resulting in the three nodes becoming co-linear.

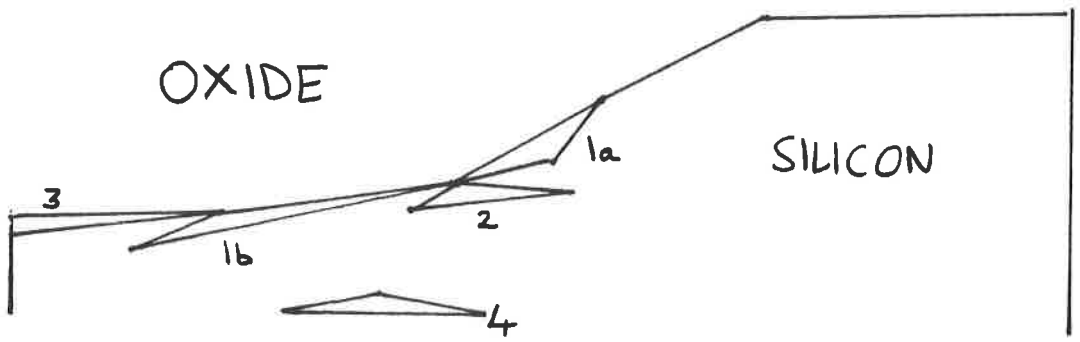


Figure 2 Examples of the various types of element folding.

In type 1a and 1b folding elements the descending surface nodes overtake the internal node, while type 2 is subtly, but significantly, different in that there is only one surface node. Type 3 folding occurs in a surface edge element when the two edge nodes become coincident, and thus as well as eliminating the element a node needs to be removed aswell. Type 4 is the generalized internal element folding. While only types 1 and 2 have so far been observed, the code has been altered to be able to handle all but the last situations, using the techniques now described.

3.2.1a Type 1a Element Folding.

In this case the folding element needs to be removed from the cross-referencing tables and the previously internal node becomes a surface node. Thus in the subsequent integration this node moves at the interface velocity. If this type of element folding occurred repeatedly then the number of surface nodes would become very large.

3.2.1b Type 1b Element Folding.

When the internal node is not between the interface nodes it does not asymptotically approach the surface. Thus in this case the element is not removed and no new interace node is created. Instead the method here is to find the neighbouring non-surface element and to swap the diagonal on the quadrilateral so produced, leaving the number of elements unchanged. This is equivalent to the method of Wathen (1984).

3.2.2 Type 2 Element Folding.

This case is like type 1b, except there is only one surface node. A similar method to that just described can be followed, again using the neighbouring non-surface element and swapping the diagonal of the quadrilateral.

3.2.3 Type 3 Element Folding.

In this case the two edge nodes have merged. The way to deal with this is to delete the upper node, along with the element, while the lower node is relabelled as a surface edge node, and thus subsequently moves with the interfacial vertical velocity (but is constrained not to move laterally).

3.2.4 Type 4 Element Folding.

Type 4 element folding is similar to types 1b and 2, but no surface node is involved. Thus a similar method can be followed by determining the neighbouring element on the longest side and swaping the diagonal on the quadrilateral so produced. This technique will obviously fail if this element is also folding, but as this is both unlikely and complex it will not be considered here.

After any of the calculations described above have been performed a complete recalculation of the element and node cross-referencing tables needs to be performed.

The treatment described here is not exhaustive, for instance an internal element could fold in a similar way to type 3. However such events have not been observed, but if they did then a simple extension of the above process could be used to overcome the problem.

3.3 Interfacial dopant concentration.

Due to the surface boundary condition not being one of zero flux, the dopant concentration at interface nodes needs to be altered from the value predicted by the 2D-MFE model. This change can be made using one of two methods:

(a) Solving the boundary condition directly

or

(b) considering the boundary condition as a flux condition and thus calculating the flux of dopant into the surface elements.

The first method is that used by Moody and Please (1987) on a 1-D MFE model while the latter is that used with a finite difference model by SUPREM III (Ho et al, 1983) and with a 'fixed' finite element scheme by TITAN (Girodolle and Martin, 1987).

3.3.1 Gradient Method.

In this technique equation 3.3 is solved for the surface dopant concentration at each node and used to overwrite the predicted value from the 2D-MFE model. As the diffusion coefficient depends upon concentration, equation 3.3 is implicit and thus needs to be solved iteratively by a technique such as Newton-Raphson.

The method of solution is first to locate which element is to be used. The required element is that which includes the normal to the surface at point I. It is located by going a short distance in the direction of $\underline{v}$ and then using the INSIDE function to locate which of the elements around node I this point is in. Next the gradients of ϕ with respect to x and y are calculated, using the GRAD function (already in the code to calculate m_j and n_j).

Following the method of Moody and Please (1987), equation (3.3) can be rewritten in the form

$$F(\phi_I) \equiv D(c)\hat{\underline{v}}\cdot\underline{\nabla}\phi + |\underline{v}|(1 - \frac{\gamma}{m})c = 0 . \qquad (3.6)$$

As c depends on ϕ_I this equation is implicit in ϕ_I, the dopant concentration at the surface. It is solved iteratively using the Newton Raphson method until

$$|F(\phi_I)| < \epsilon$$

where ϵ is a small number.

However, while this method successfully converges, the resulting surface concentrations are much too high. The reason for this becomes clear when physical situation is considered in conjunction with the mesh. The effect of segregation only has a direct effect very near the surface, and there produces significant gradients of the size predicted by equation 3.6. However the mesh used is optimised to have high resolution where the initial second derivatives of the concentration were high. Resolution near the interface is therefore poor and using the gradient from equation 3.6 over the whole element results in gross distortions. For this reason this gradient technique was abandoned and the alternative flux method used instead.

3.3.2 Flux Method.

Following TITAN (Girodolle and Martin, 1987), and many others, the diffusion equation can be split into a flux form:

$$\frac{\partial c}{\partial t} = -\nabla\underline{J} , \quad \underline{J} = -D\underline{\nabla}C \qquad (3.7)$$

In this notation, equation 3.3 can be rewritten as

$$J_n \equiv - D \frac{\partial c}{\partial n} = - |v|(1 - \frac{\gamma}{m})c \qquad (3.8)$$

where J_n is the flux normal to the interface. This flux is used to calculate the increase in concentration at the surface due to segregation:

$$Cnew = C + FLUX * (LENGTH / AREA) * DT \qquad (3.9)$$

where LENGTH is half the sum of the lengths of the interface edges of the two surface nodes including the node under consideration, while AREA is a third the sum of the areas of all the elements about the node. The size of this increase in C was found to be normally less than that due to the advection of the node.

4. Results.

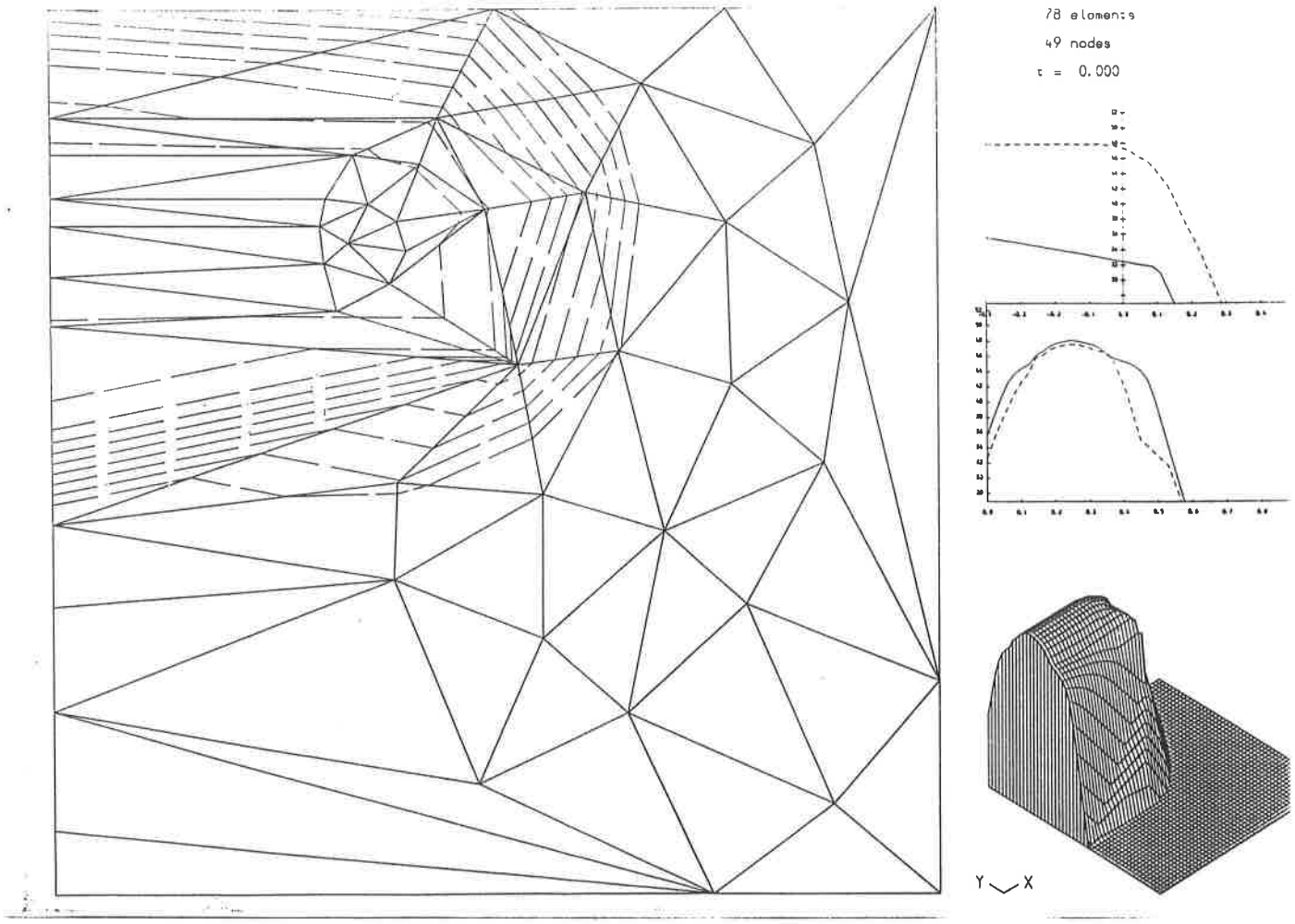
The combined oxidation-diffusion model, including element deletion and the surface flux calculation, has been used to simulate furnace anneals at 950°C. Three grids have been used with different resolutions, having initially 408,166 and 78 elements (by setting DG = 0.9,1.4 and 2.0 respectively). Results for integrations are shown in the accompanying figures, with the equivalent integration with no oxidation given for comparison. In figure 3 results for DG = 2 are shown for a 2 hours anneal, while figure 4 shows the same simulation but with the incresed resolution when DG = 1.4. Figure 5 shows the evolution of the model with high resolution (DG = 0.9) when simulating a 20 minute anneal.

In the low resolution case it can be seen that the amount of diffusion is overestimated (compare figure 3c with figure 4c). This is not due to the large grid size introducing extra numerical diffusion, for as figure 3d shows this does not occur when oxidation is not included. However poor resolution of the silicon-oxide interface leads to excess silicon conversion to silicon dioxide and thus to a greater injection of dopant, by segregation, and thus increased diffusion. Note that in this case no elements are removed during the simulation because the subsurface nodes move away from the interface before being overtaken.

When medium resolution is used (figure 4), a more realistic dopant distribution results, with the peak concentration and vertical diffusion being slightly greater than in the no oxidation case. In the horizontal the diffusion is noticably enhanced, particularly near the interface, due to segreagation increasing the amount of dopant present

in that area. During the integration a surface element has been eliminated due to its folding. This created an extra surface node and thus further improved the resolution of the interface profile. The elements in the region of the initial high concentration region have clustered together, due to the lowering interface, resulting in the timesteps remaining small and thus increasing the total number of timesteps, and the CPU requirement, by about a factor of 5 more than the no oxidation case.

When the resolution is increased yet further, the oxidation-diffusion model has severe difficulties. For while at 1200 secs the simulation is both realistic and stable (see figure 5c) at 2000 secs it goes unstable and rapidly produces infinite dopant concentrations. The cause of this is unclear and requires further investigation. One possible cause is that the nodes near the left-hand edge have been observed to oscillate with a period of two timesteps, and this could lead to instabilities and also timesplitting, as is observed when the leapfrog scheme is used with finite differences.



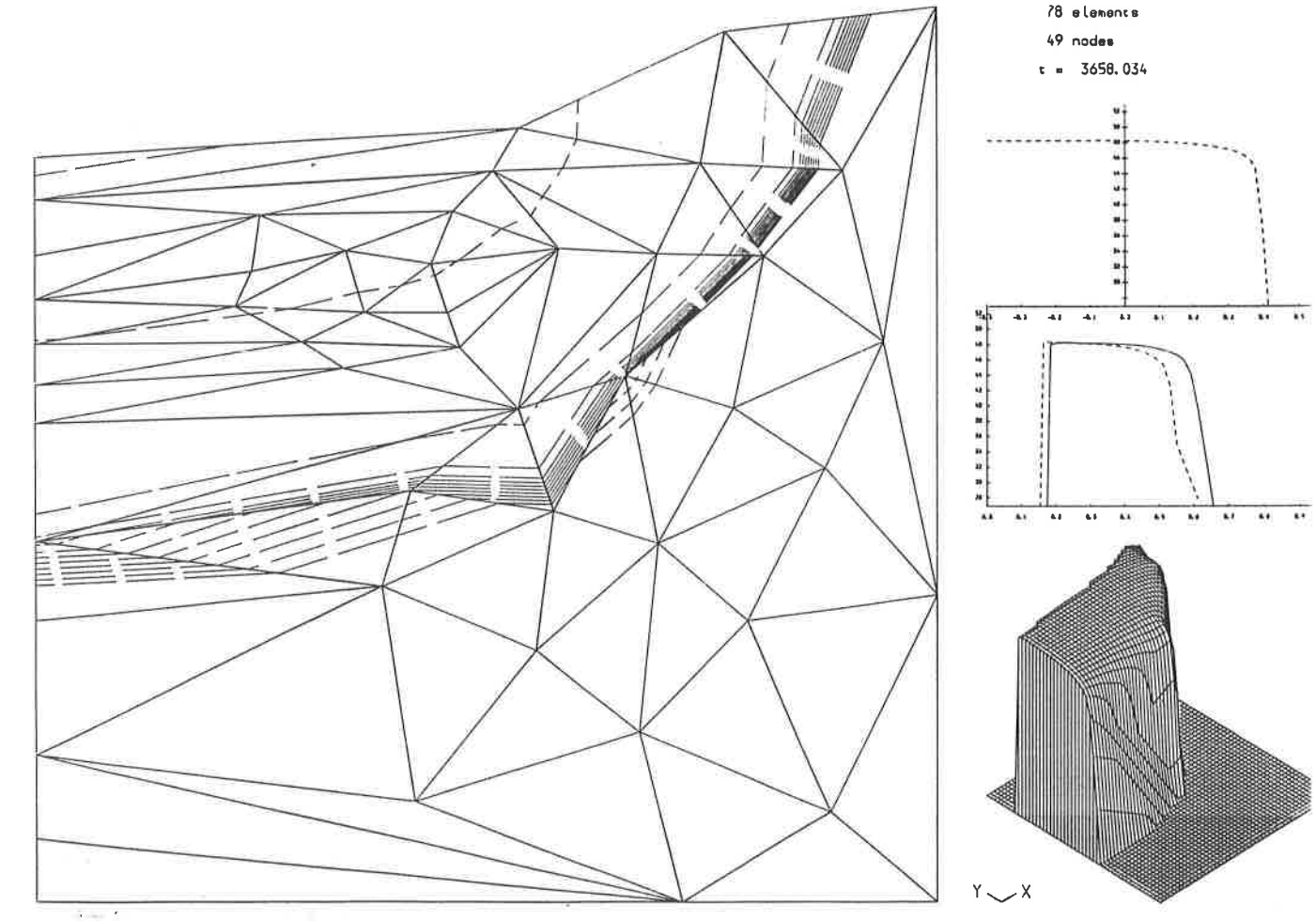
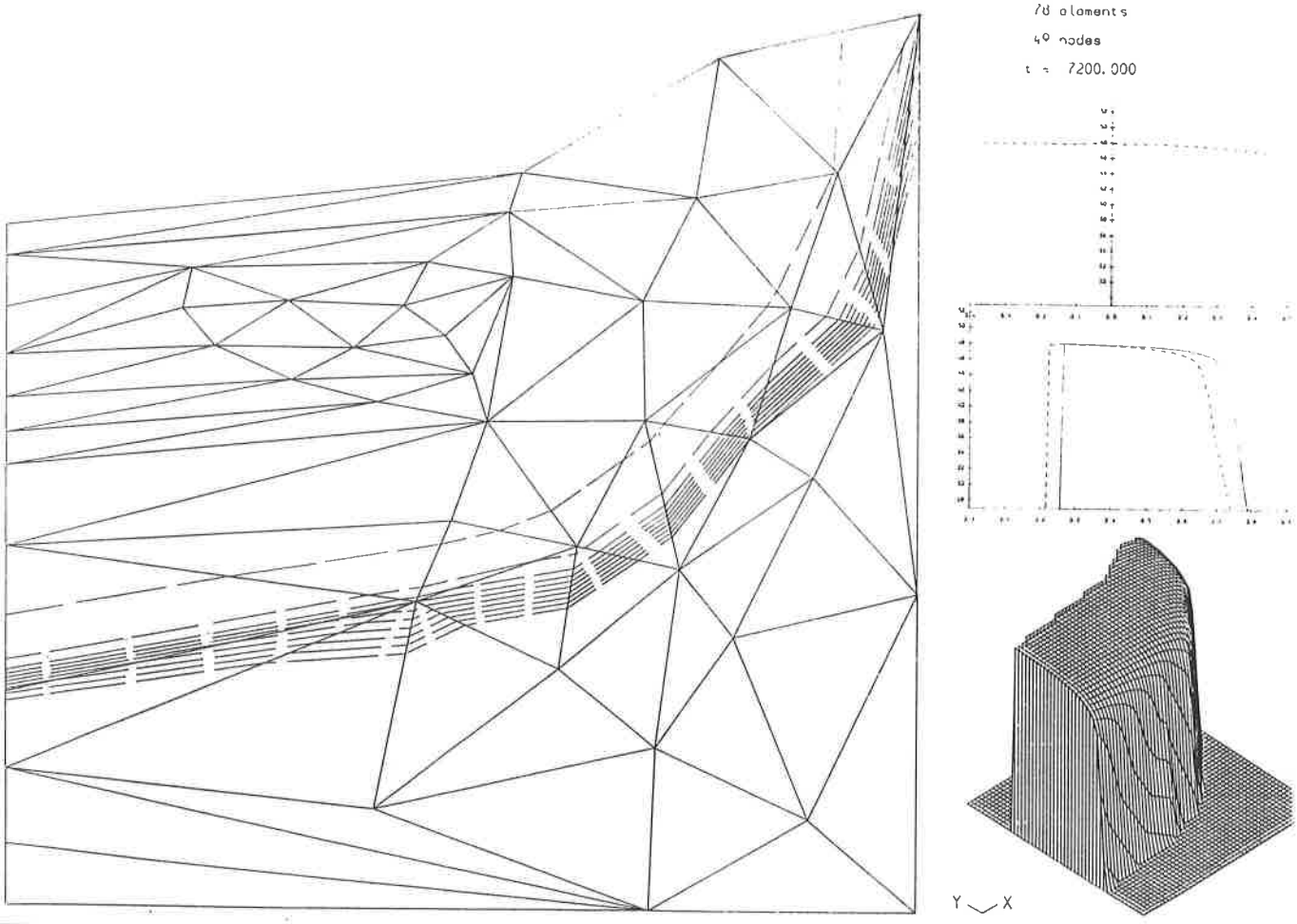
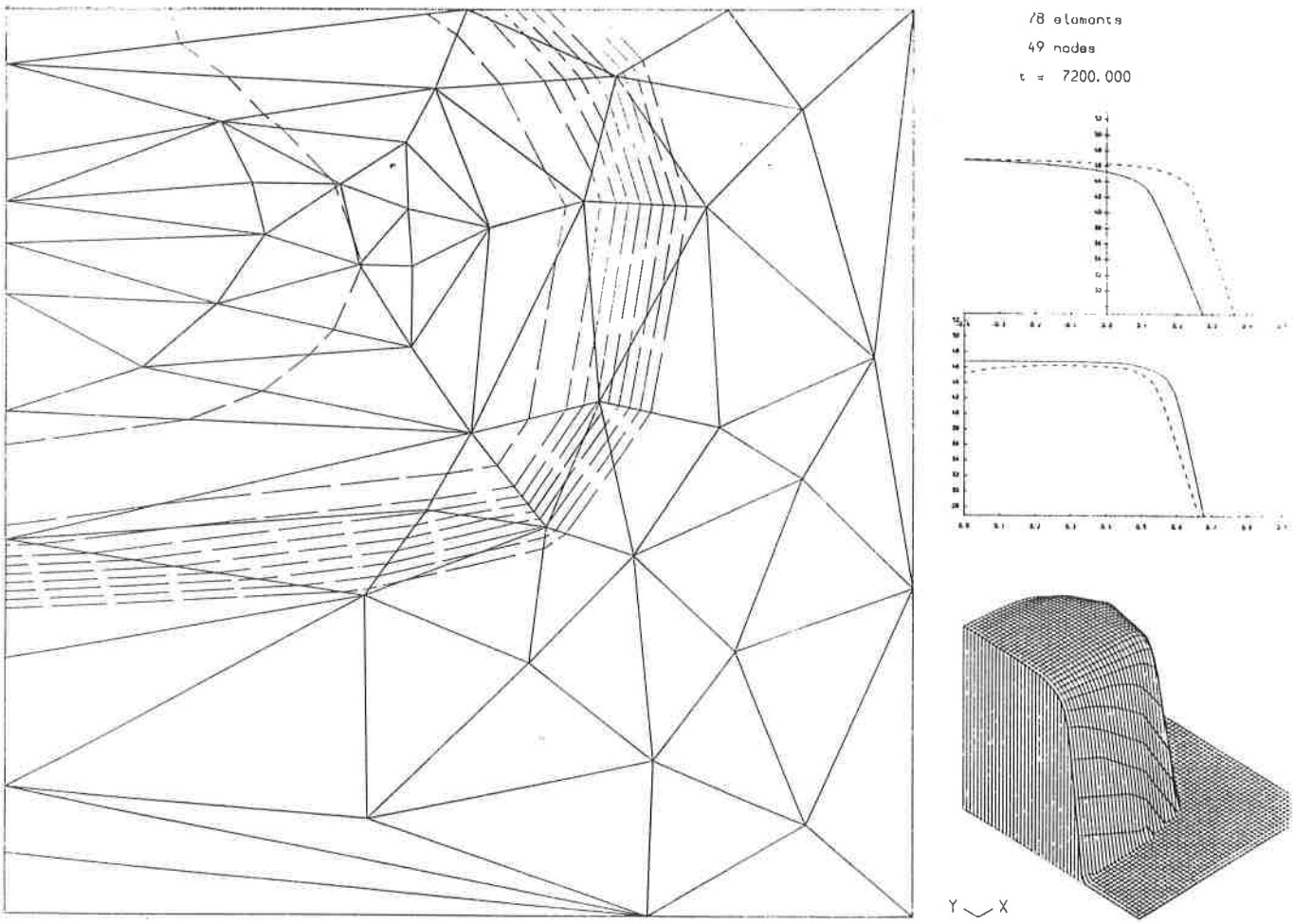
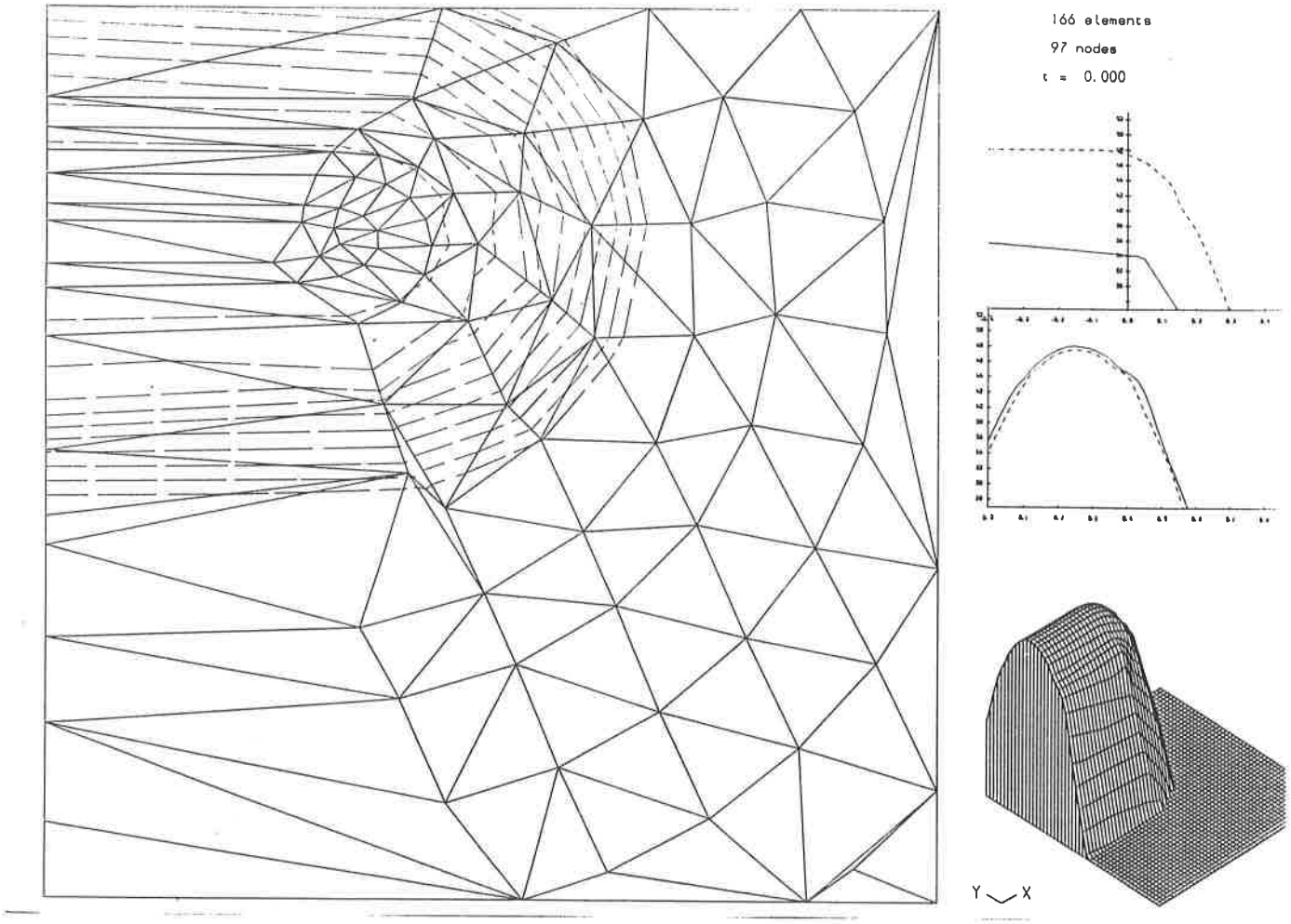


Figure 3. (a) Initial conditions for DG=2. Solid lines - elements, dashed lines - contours. Plots on rhs: top: lateral sections at y=0 (dashed) and y=.25 (solid), middle: vertical sections at x=-0.4 (solid) and x=0 (dashed), bottom 3-D visualisation.
(b) After annealling for about 1 hour at 950 C.





(c) After annealling for 2 hours.
(d) As (c) but with no oxidation.



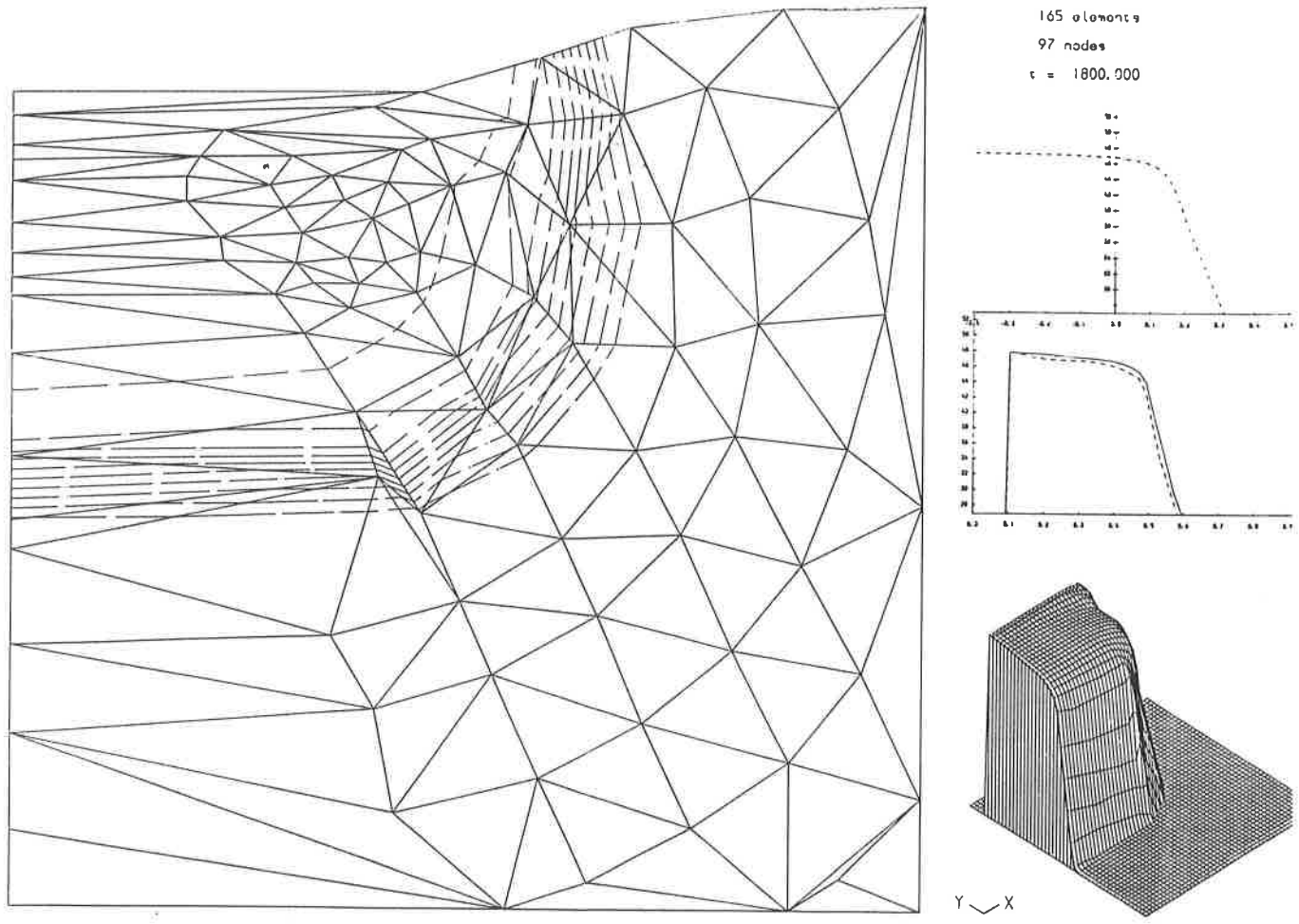
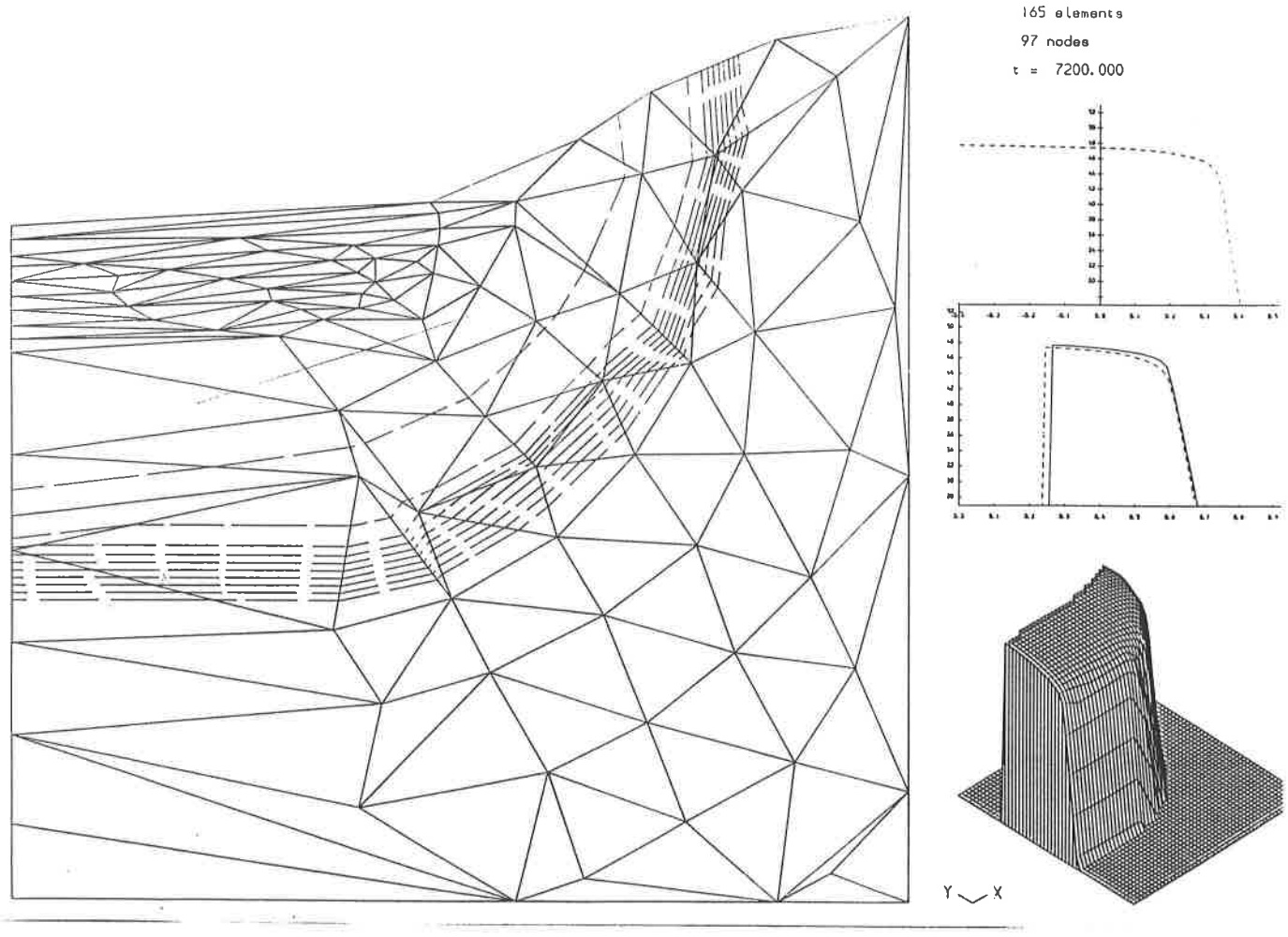
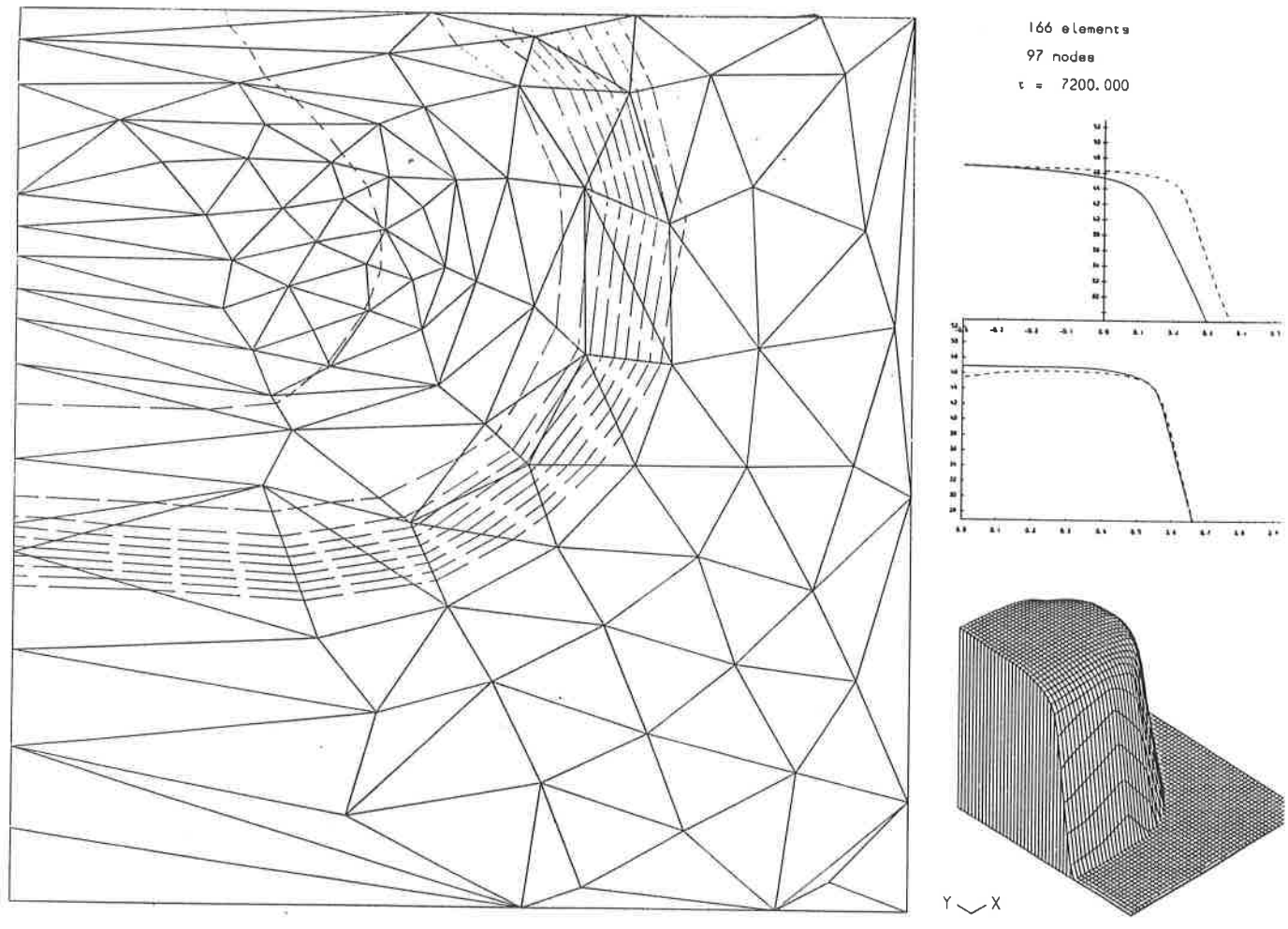
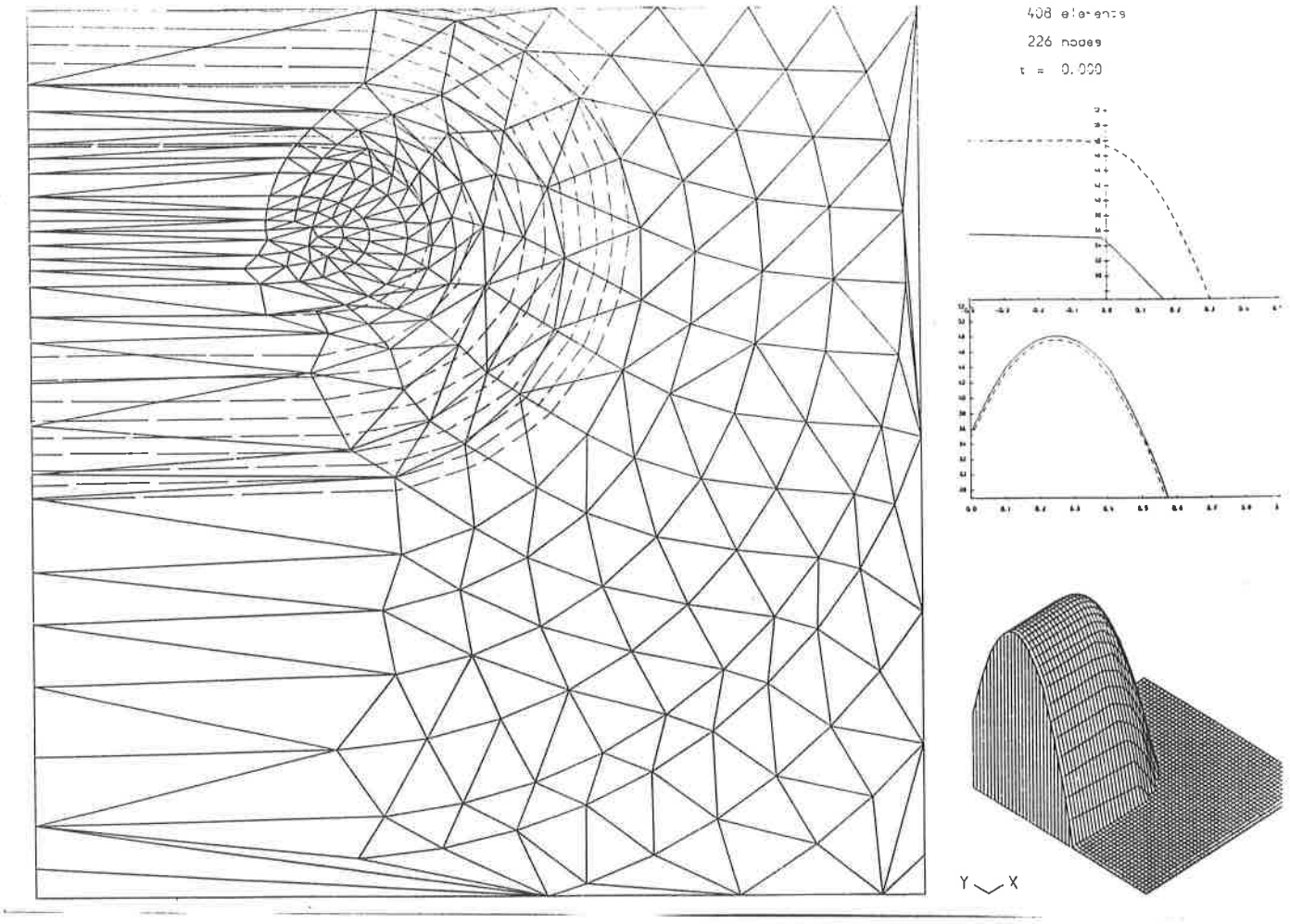


Figure 4. (a) As 3(a) but with DG=1.4.
(b) After 30mins anneal at 950 C.





(c) After 2 hours anneal.
(d) As (c) but with no oxidation.



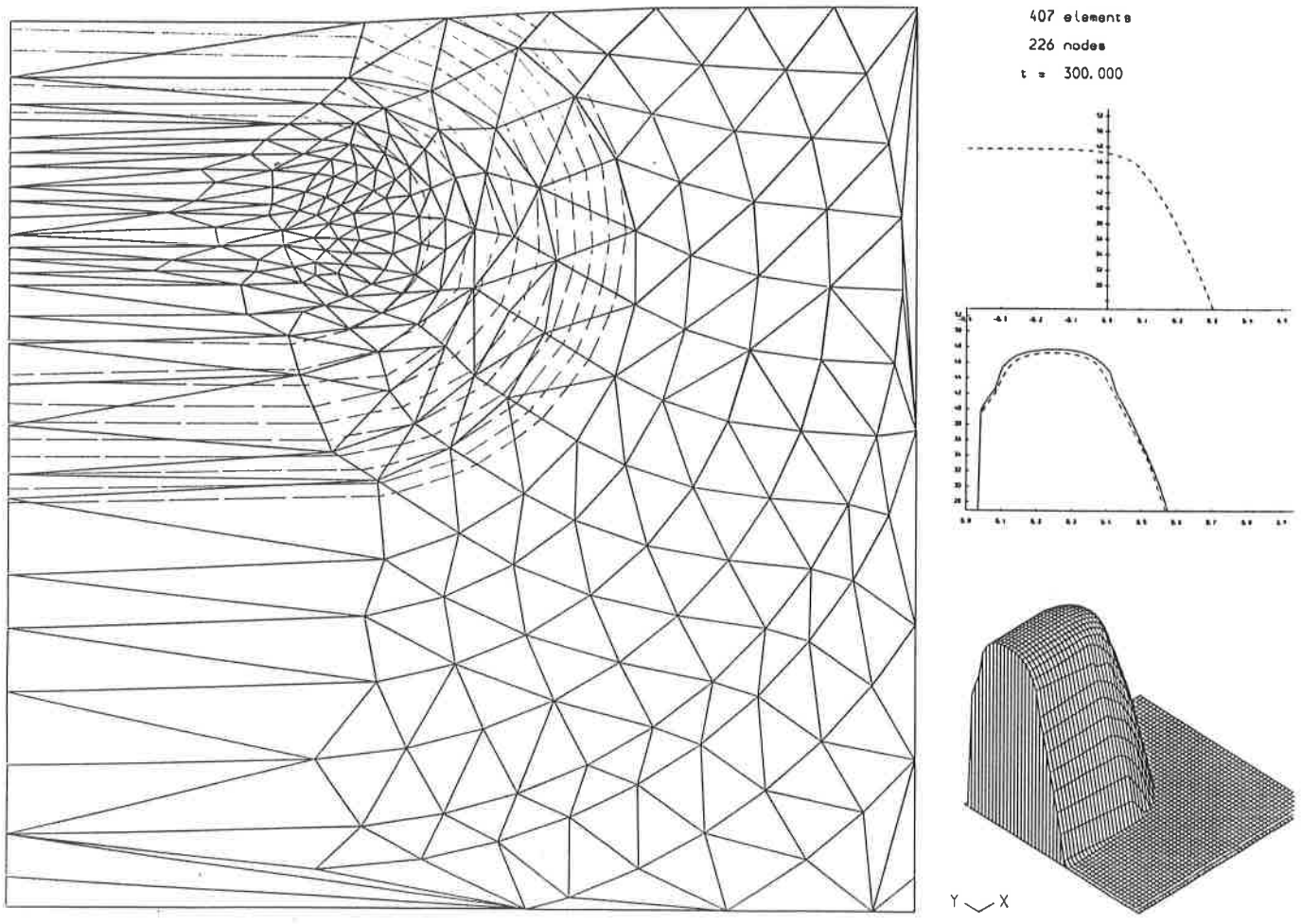
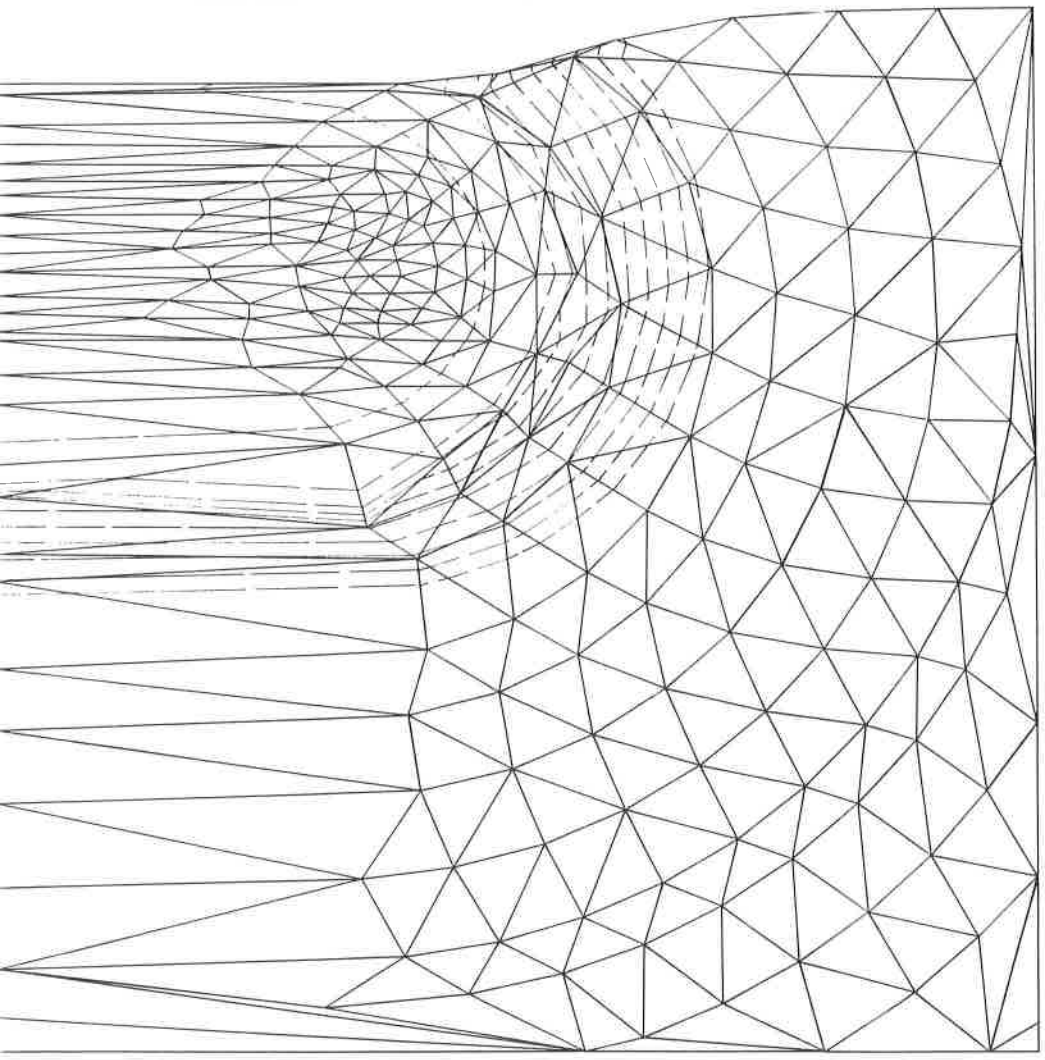
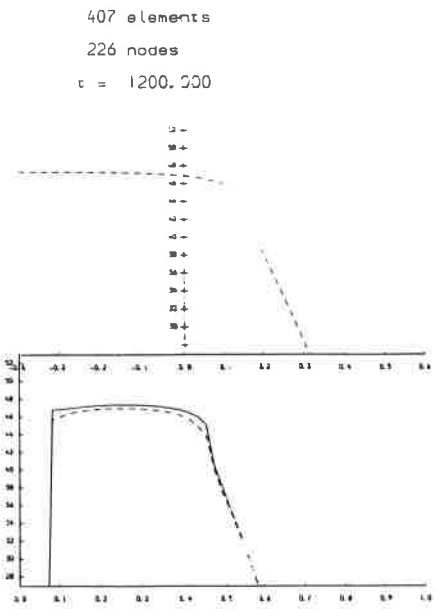
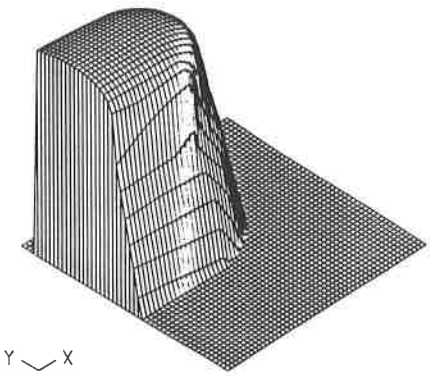
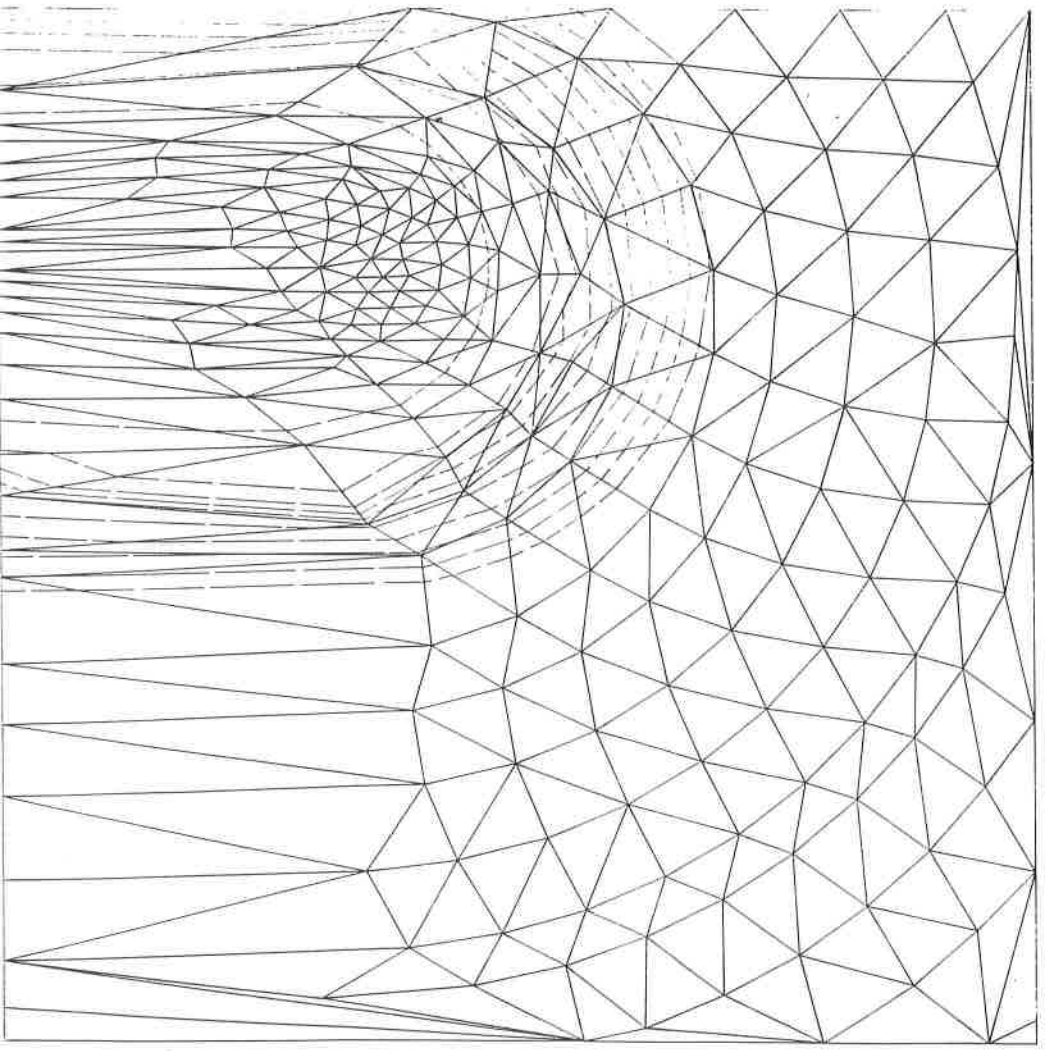


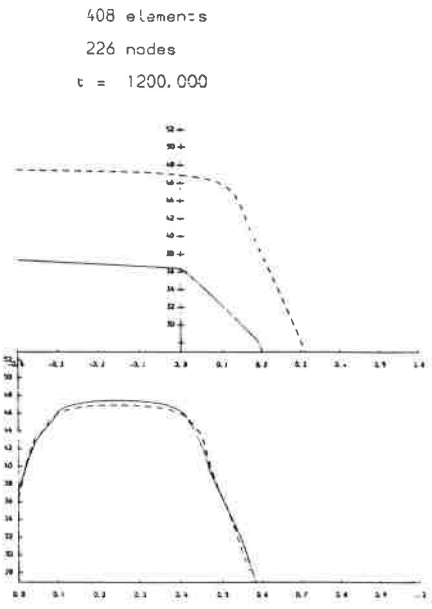
Figure 5. (a) As 3(a) but with DG=0.9
(b) After 5 mins anneal at 950 C.

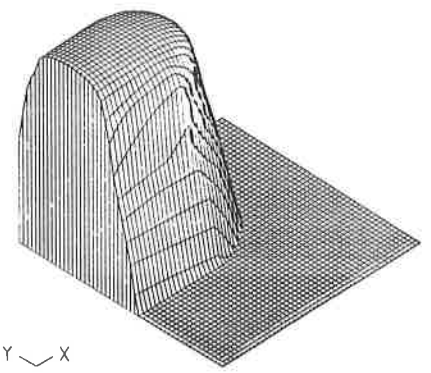












(c) After 20 mins anneal.
(d) As (c) but with no oxidation.

5. Conclusions.

The 2D MFE model of the diffusion of dopant in silicon has been extended to include the effect of oxidation. The code for this at present needs external data to describe the interface movement, but an analytical model has been presented which could alternatively be used. In either case, a scaling analysis of the relevant equations shows that dopant diffusion in the oxide is negligible compared to that in the bulk silicon, and this leads to the oxidation and diffusion problems being separable. Thus the diffusion model only needs to know the location of the oxide interface.

The main alterations implimented relate to the calculation of the position and dopant concentration of the interface nodes. Prescribing the motion of surface nodes alters not just their location but also, by advection, their dopant concentration. The recovery functions used to calculate the rate of change of concentration at a given node needs to take into account the new interface boundary condition (equation 2.16). The effect of segregation of dopant from the oxide into the silicon needs to be included, and this is achieved by writing the interface condition in flux form (equations 3.7 and 3.8).

Due to the moving oxide interface, the upper nodes are forced down, which tends to cause element folding. In all cases this can be overcome by applying the appropriate geometrical transformation to the element definition.

The new model can provide reasonable simulation of the oxidation-diffusion process, as is shown by figure 4. However, the model still has various problems which need to be tackled. Primary amoung these is its failure to simulate anneals longer than 30 minutes

with the high resolution mesh. Another serious deficiency is the amount of computer time required for integrations. The small timesteps forced on the model are due to the excess number of nodes in the region below the fastest oxidation, creating small elements in a region of almost constant dopant concentration. There is a need for the model to automatically reduce the number of nodes in this area, thus allowing larger timesteps.

The model is of suitable ability to now be include in the ALVEY 066 Software ('TAPDANCE'). This will require the appropriate input/output code written to interact with the overall package and also some cosmetic alterations to the code.

6. Acknowledgements.

I wish to thank the members of the Numerical Analysis Group in the Department of Mathematics at the University of Reading. In particular, Mike Baines who arranged this project and guided me, and Simon Chynoweth, Neil Hall, Rob Moody and Pete Sweby who gave me much useful advice. This work was supported by the Alvey Directorate through the SERC.

7. References.

BAINES M J, PLEASE C P and SWEBY P K, 1986, "Numerical Solution of Dopant Diffusion Equations" in "Simulation of Semiconductor Devices and Processes" vol 2, (Ed Board K and Owen D R J), Pineridge Press, Swansea pp271-286, 1986.

CHIN D, OH S-Y and DUTTON R W, 1983, "A General Solution Method for Two-Dimensional Nonplanar Oxidation", IEEE Trans Electron Devices ED-30: 993-998.

DEAL B E and GROVE A S, 1965, "General Relationship for the Thermal Oxidation of Silicon", J Appl Phys 36: 3770-3778.

GERODOLLE A and MARTIN S, 1987, "TITAN 4, 2D Process Simulation, User Guide", CNET, Grenoble. GODFREY D J, 1984, "Improvements to the Suprem 2.5 Silicon Process Model", GEC J Res 2: 232-239.

GODFREY D J, 1986, "Modelling Physical Processes in Silicon Integrated Circuit Fabrication", Phys. Technol. 17: 260-264.

HO C P, PLUMMER J D, HANSEN S E and DUTTON R W, 1983, "VLSI Process Modelling - SUPREM III", IEE Trans Elect Devices, ED-30: 1438-1453.

MILLER K and MILLER R N, 1981, "Moving Finite Elements, Part 1", SIAM J Num Anal 18: 1019-1032.

MOODY R O and PLEASE C P, 1987, "A Numerical Simulation of One-Dimensional Dopant Diffusion in Oxidising Silicon", Numerical Analysis Report 3/87, Dept. of Mathematics, University of Reading.

PENUMALLI B R, 1981, "Lateral Oxidation and redistribution of dopants" in Proc NASECODE Conf (Ed. J H H Miller).

PLEASE C P and SWEBY P K, 1986, "A Transformation to Assist Numerical Solution of Diffusion Equations", Numerical Analysis Report 5/86, Dept. of Mathematics, University of Reading.

SWEBY P K, 1987, "Some observations on the Moving Finite Element Method and its Implemnetation", Numerical Analysis report 13/87, Dept. of Mathematics, University of Reading.

WATHEN A J, 1984, "Moving Finite Elements and Oil Reservior Modelling", Reading Univeristy Ph. D. Thesis.

Appendix - The Computer Code.

A.1 Interfacial Node calculations.

In the present implimentation the oxide growth is simulated using data supplied by J. Waddell of University College, Swansea from their oxidation model, rather than using the Penumalli (1981) model given in equations (2.2) and (2.3). This data is read in by the new subroutine OXDATA (called by DATA), and stores oxidation boundary node data in OXX and OXY for times OXT, and velocities OXU and OXV are calculated. Units used are μm and seconds. The option of left side or right side oxidation are available, corresponding to a mask on the right or left respectively.

The velocity of oxide growth at each interfacial node is calculated in subroutine TSTEP and stored in array BDC. Conversion to normalized units is performed simultaneously with the interpolation, by dividing by DI.

Specification of interfacial boundary condition affects the calculation of the adjacent nodal dopant concentration as the constraints on gradient affects the recovery function. Thus the elements of the array NBC can now take one of 5 values, rather than the 3 previously allowed by MFE2D. These values are:

0	Totally free to move	(Internal point)
1	Constrained to move on a boundary	(Edge point)
2	Fixed, zero velocity	(Corner point)
3	Constrained to move with oxide	(Interfacial point)
4	As 3, but with no lateral motion	(Interfacial corner)

The new values of 3 and 4 are defined in XINDEX by adding 2 to any point with y co-ordinate less than 1.E-5. The difference between NBC = 3 and NBC = 4 is only used once, in the calculation of MTDW(2), which is 0 for NBC = 4 but given by the oxidation value if NBC = 3 .

The other main code changes are in loop 60 of TSTEP. Indeed this already had two existing errors:

(a) The case of NBC(N1).EQ.1.AND.NBC(N2).EQ.2 was not considered.

(b) The case of N1 and N2 being on different edges (corner element) was not considered and thus could result in nodes being fixed to move in the wrong direction. Thus an extra test is now included to check that (ABS(BDC(N1,1)-BDC(N2,1).LT.1.E-6) ie that N1 and N2 are on the same edge.

Loop 60 was then extended to consider all the possible permutations of NBC(N1) and NBC(N2) and to then calculate the appropriate constraint on the gradient. (Note that this does not prescribe the nodal motion, but is only used to calculate the node value and position velocities. The motion for points with NBC > 0 is overwritten later in the subroutine by specifying MTDW and WTDW.)

The subroutine OXCAL2 calculates the flux into surface node I due to segregation at the interface. The method of calculation is described in section 3.3.

A.2 Node elimination.

The subroutine GETDT has been altered so that the element which restricts the timestep is recorded. When element folding occurs it is

the folding element which is thus recorded. If the area of the element drops below 1.E-6 then it is removed or rearranged, depending upon location, as explained in section 3.2.

To determine which class of element (see section 3.2) is under consideration then the sum of NBC for its 3 edges is calculated. For type 1, NSUM = 6, for type 2 NSUM = 3 (or perhaps 4), for type 3 NSUM = 8 while for type 4 NSUM = 0 (but can be up to 4 if in a bottom corner).

To distinguish between types 1a and 1b folding elements w need to determine whether the non-surface element is between the other two, or at one end. If it is between then the element is type 1a and the element is to removed, thus the element reference number needs to be reduced by 1 for all elements whose reference number was higher than the removed elements. Similarly NE is reduced by 1. However, if the non-surface element is not in the middle then the element is not to be removed. Instead the element neighbouring the longest edge of the folding element needs to be located, then the diagonal of the quadrilateral so produced can be swapped resulting in both elements having at least one surface node.

Before assuming a type 2 element is being considered, a check that there is a surface node is made, for otherwise a type 4 mode has folded. Next the neighbouring element not including the surface node is found. This then allows the quadrilateral formed by these two elements to have its diagonal swapped, by changing the element nodes.

For type 3 a node is removed as well as an element, thus N needs to be reduced by 1 as well as NE. All nodes and elements with reference numbers higher than those removed need to be reduced by 1.

After any of these changes the cross reference table NND and NEIGHE need to be recalculated, and this is done using code copied from XINDEX.

2.2 Oxidation.

Simulation of the oxide growth, and the diffusion of dopant within it, is a very complex problem (Chin et al., 1983) and will not be tackled here. Instead it is assumed that information about the oxide comes from one of two sources: An external oxide model (in the case of the ALVEY 066 software this will be provided by University College, Swansea) or else an analytical model.

2.2.1 The Analytical Model.

The analytical model is that of Penumalli (1981), which is used by the TITAN process simulator (Girodolle and Martin, 1987). Penumalli's model is a two-dimensional extension of the classic Deal and Grove (1965) one-dimensional model used by Moody and Please (1987), with the lateral variation in oxide thickness being described by an error function. Figure 1 illustrates the model together with an explanation of the notation used here.

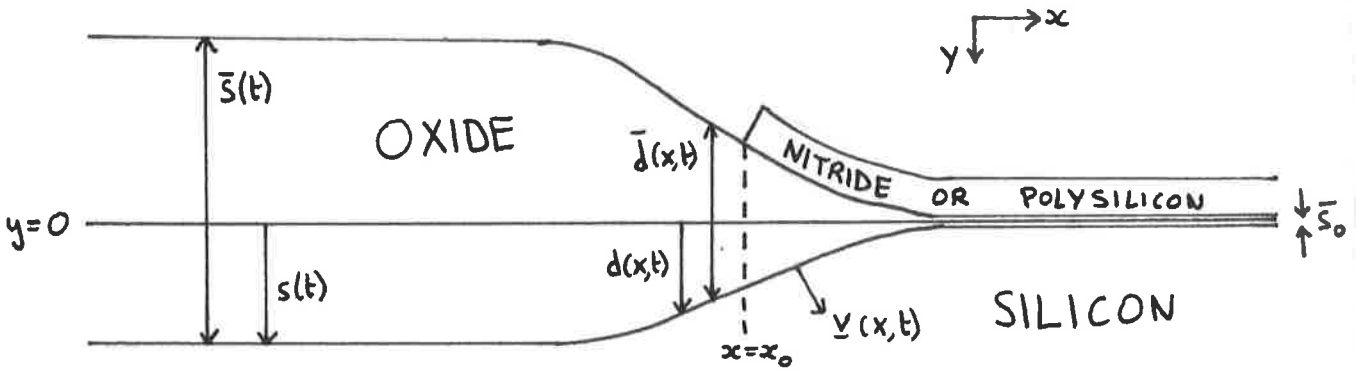


Figure 1

The logical method to overcome element folding is to allow elements to disappear. Thus the model was adapted to locate very small elements, eliminate them and alter the cross referencing in the node table.

3.2 Treatment of folding elements.

The inclusion of a moving boundary greatly increases the probability of element folding, particularly at the surface. This has been shown to occur in most integrations of the combined oxidation-diffusion model. Elements can fold in many different ways, as is illustrated by figure 2, but normally resulting in the three nodes becoming co-linear.

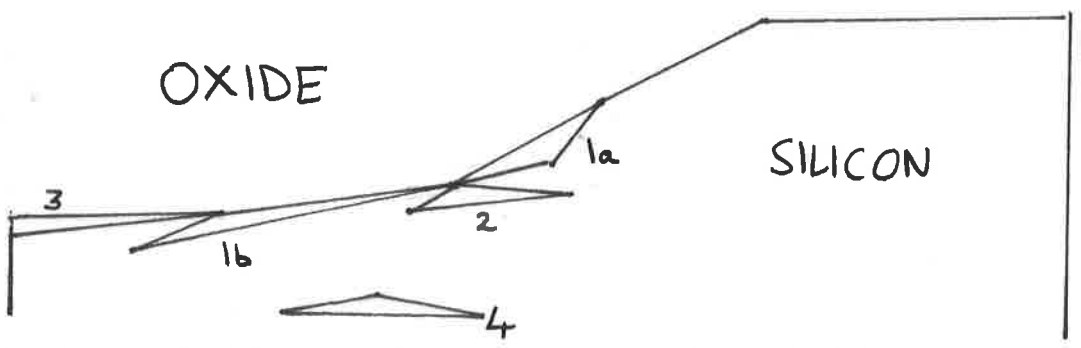
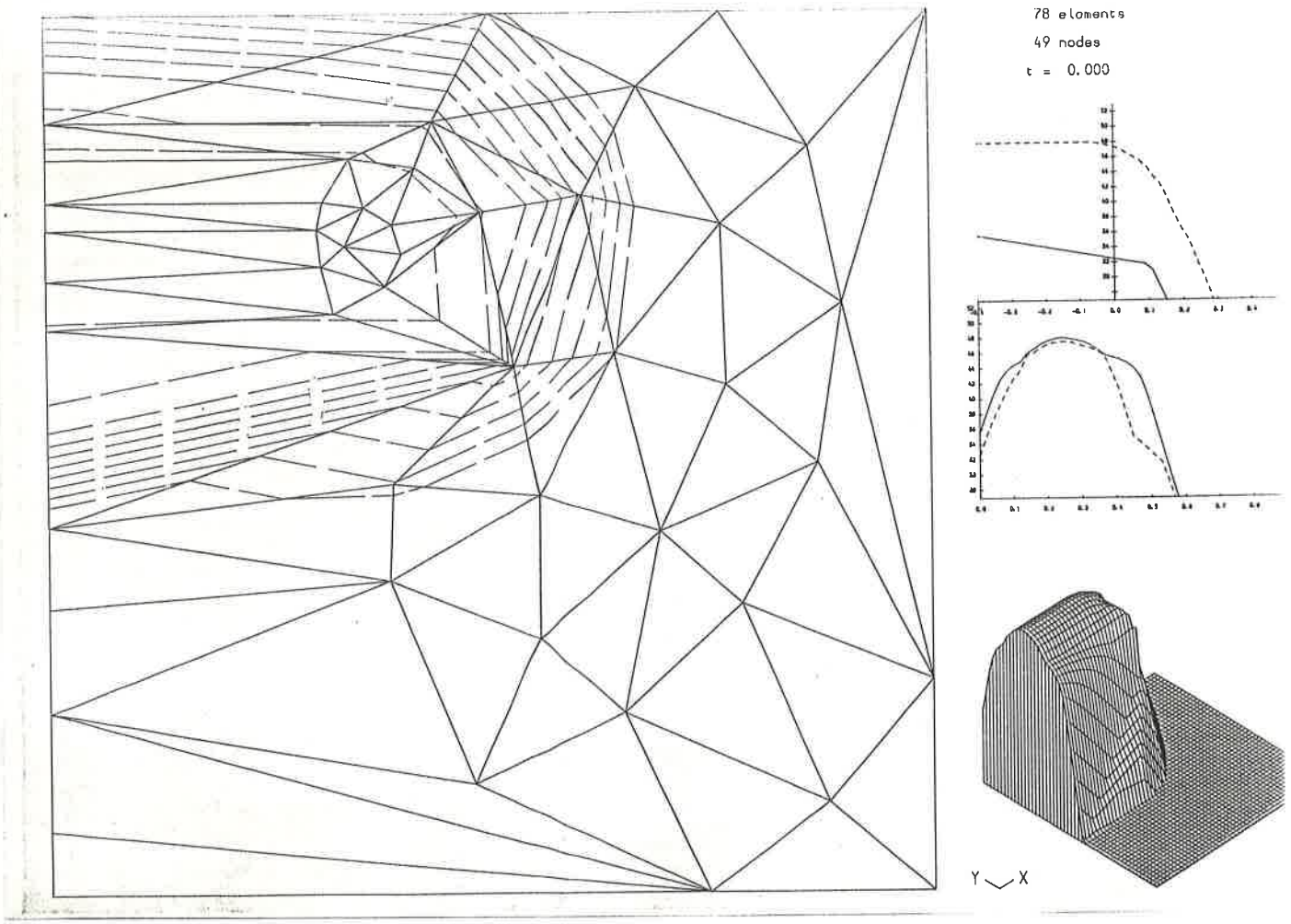


Figure 2 Examples of the various types of element folding.



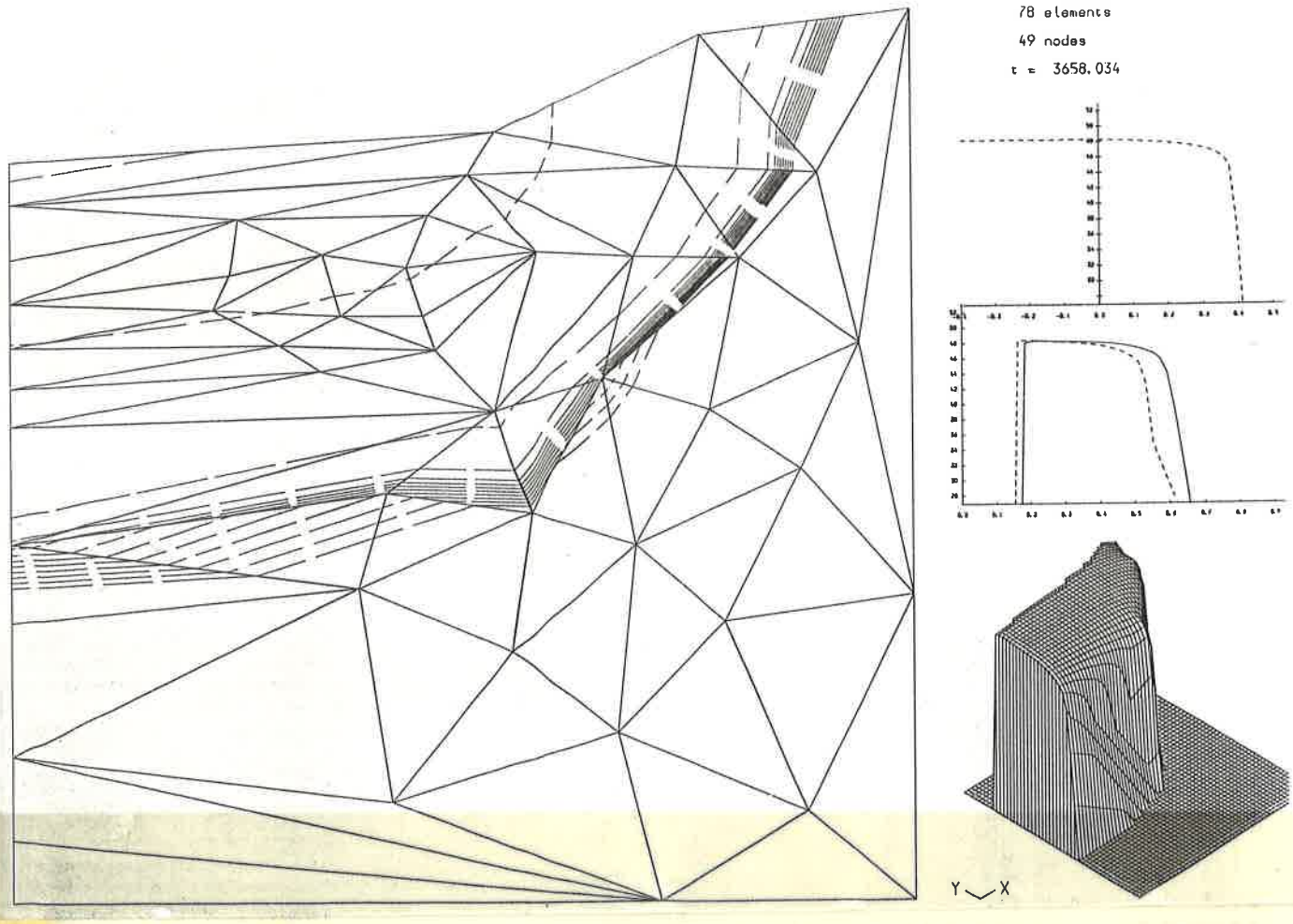
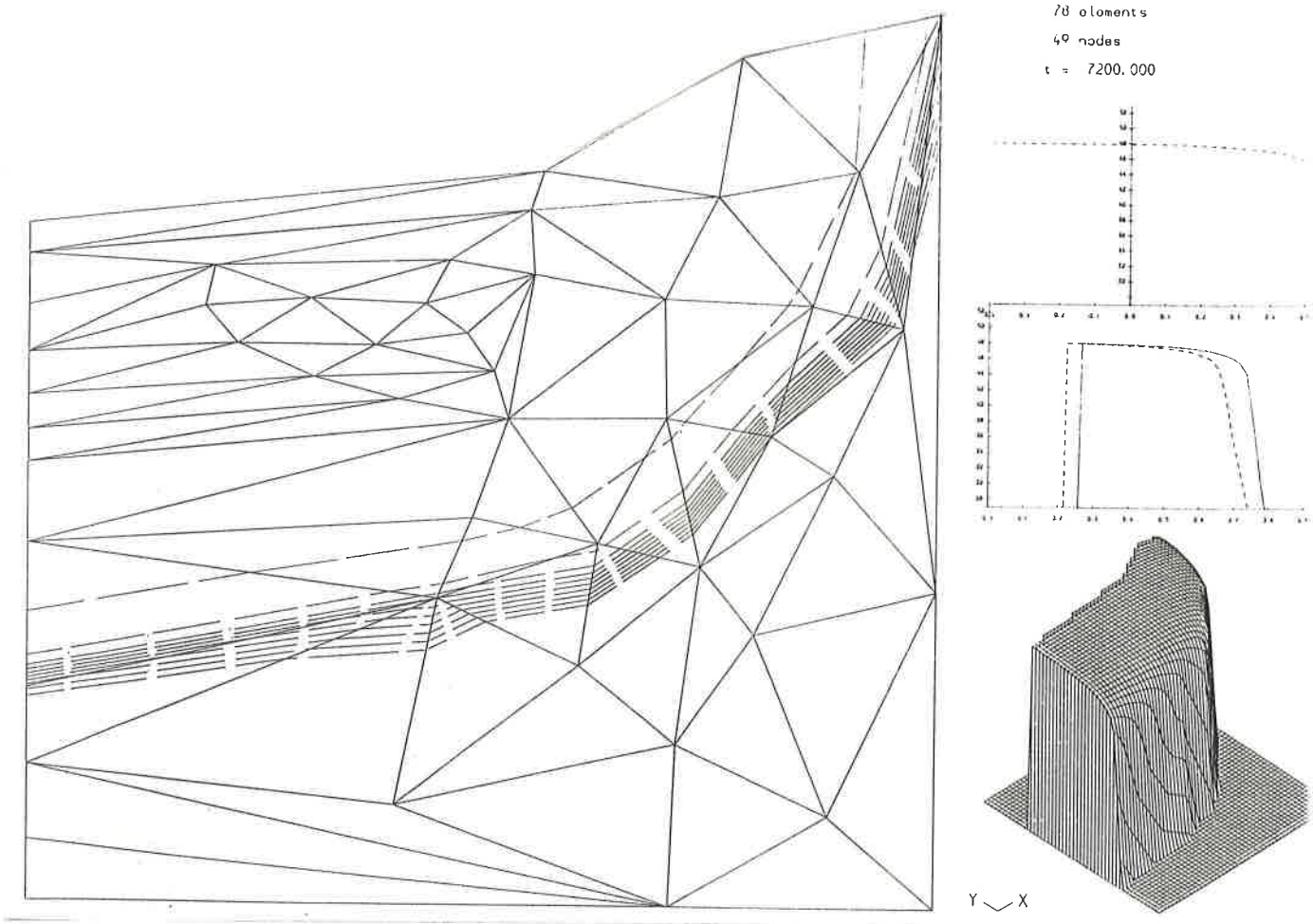
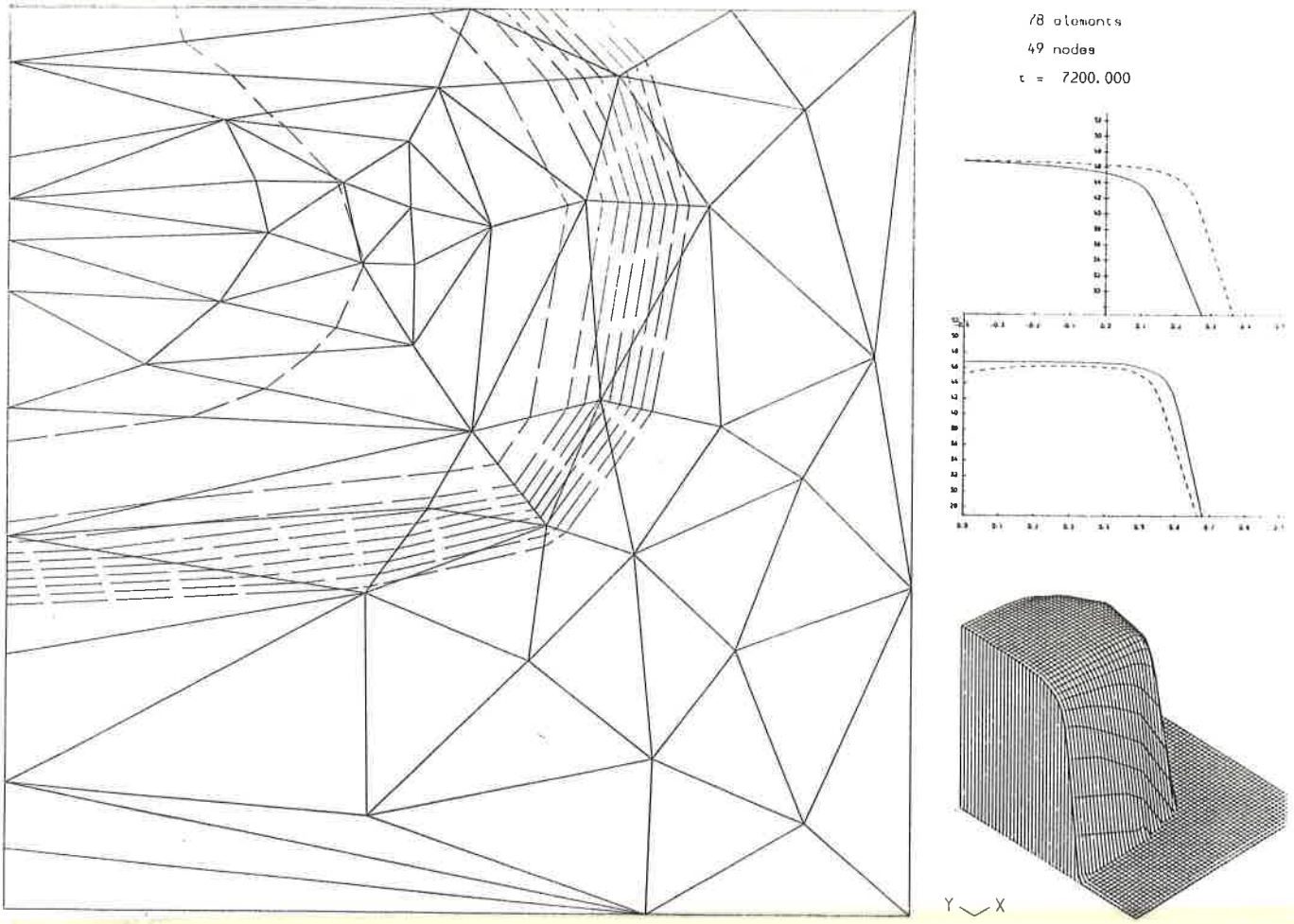
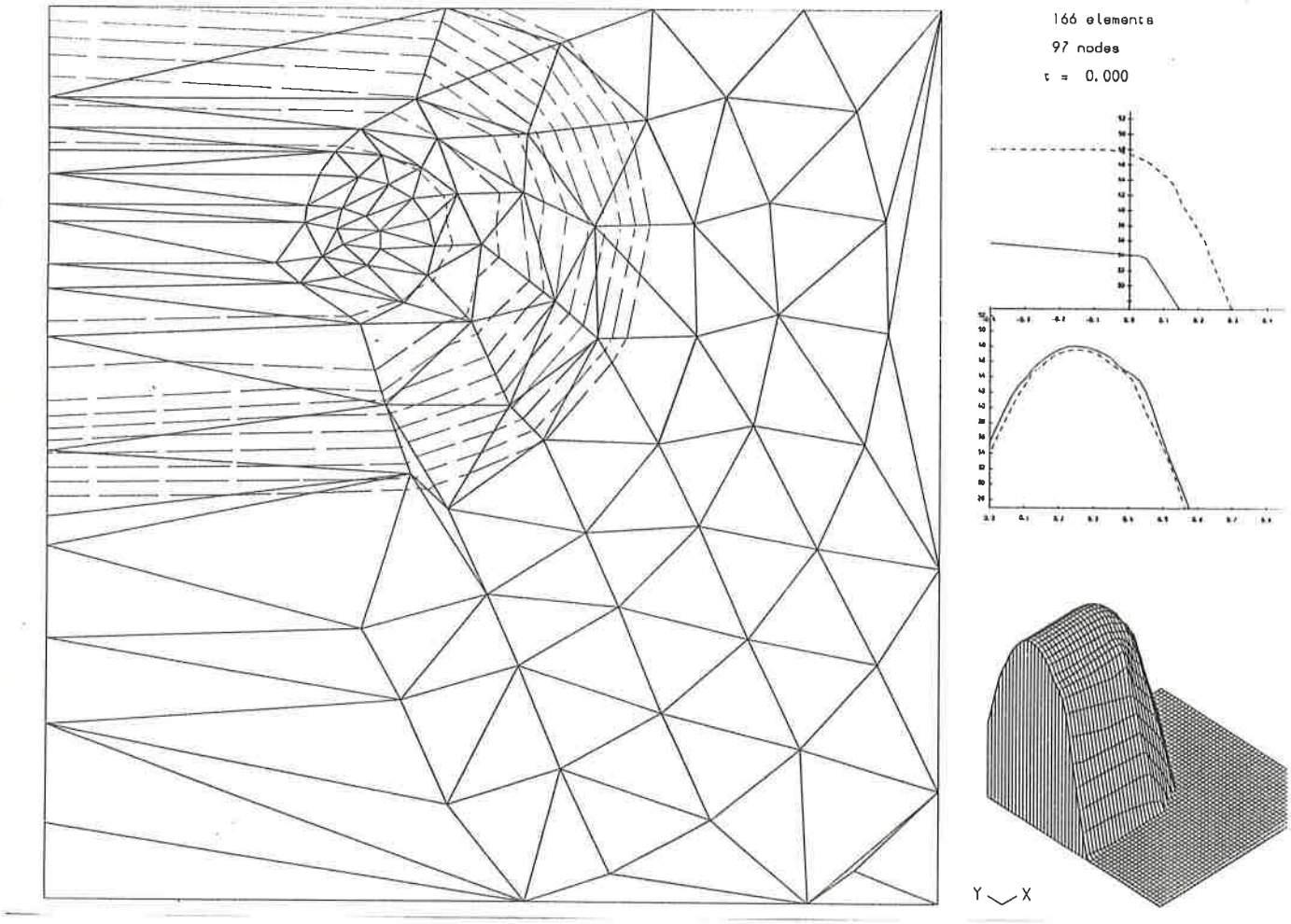


Figure 3. (a) Initial conditions for DG=2. Solid lines - elements, dashed lines - contours. Plots on rhs: top: lateral sections at y=0 (dashed) and y=.25 (solid), middle: vertical sections at x=-0.4 (solid) and x=0 (dashed), bottom 3-D visualisation.
(b) After annealling for about 1 hour at 950 C.





(c) After annealling for 2 hours.
(d) As (c) but with no oxidation.



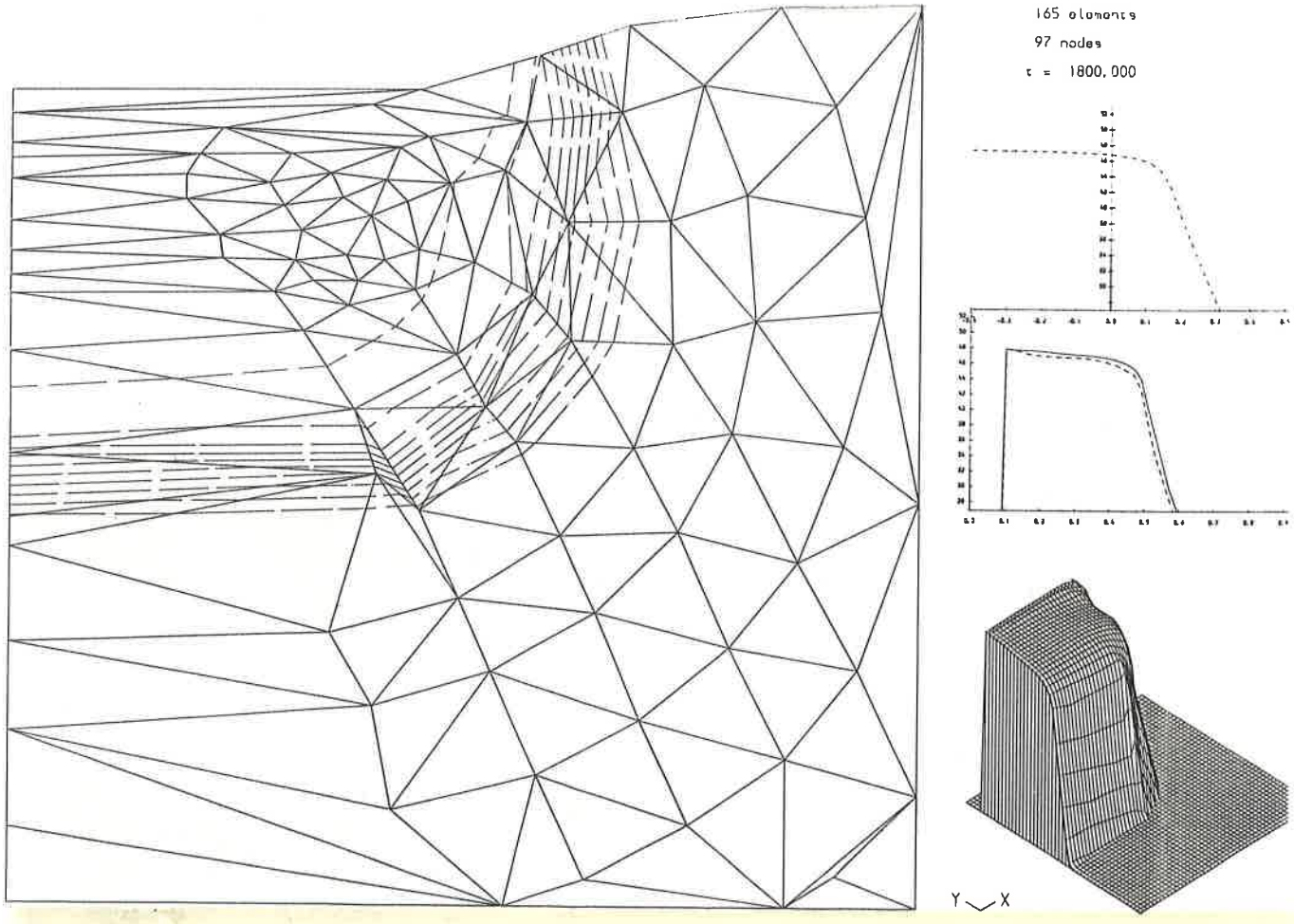
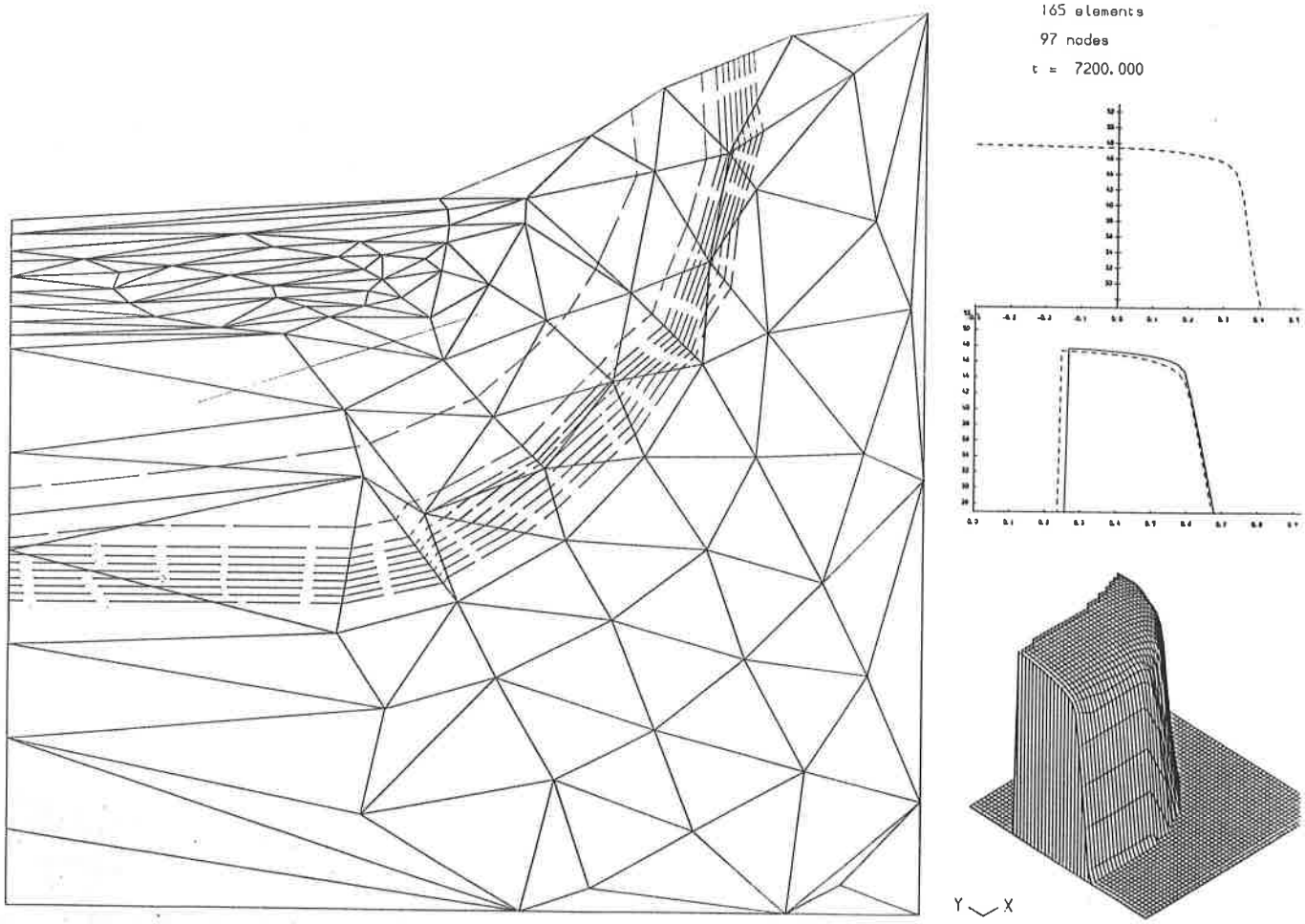
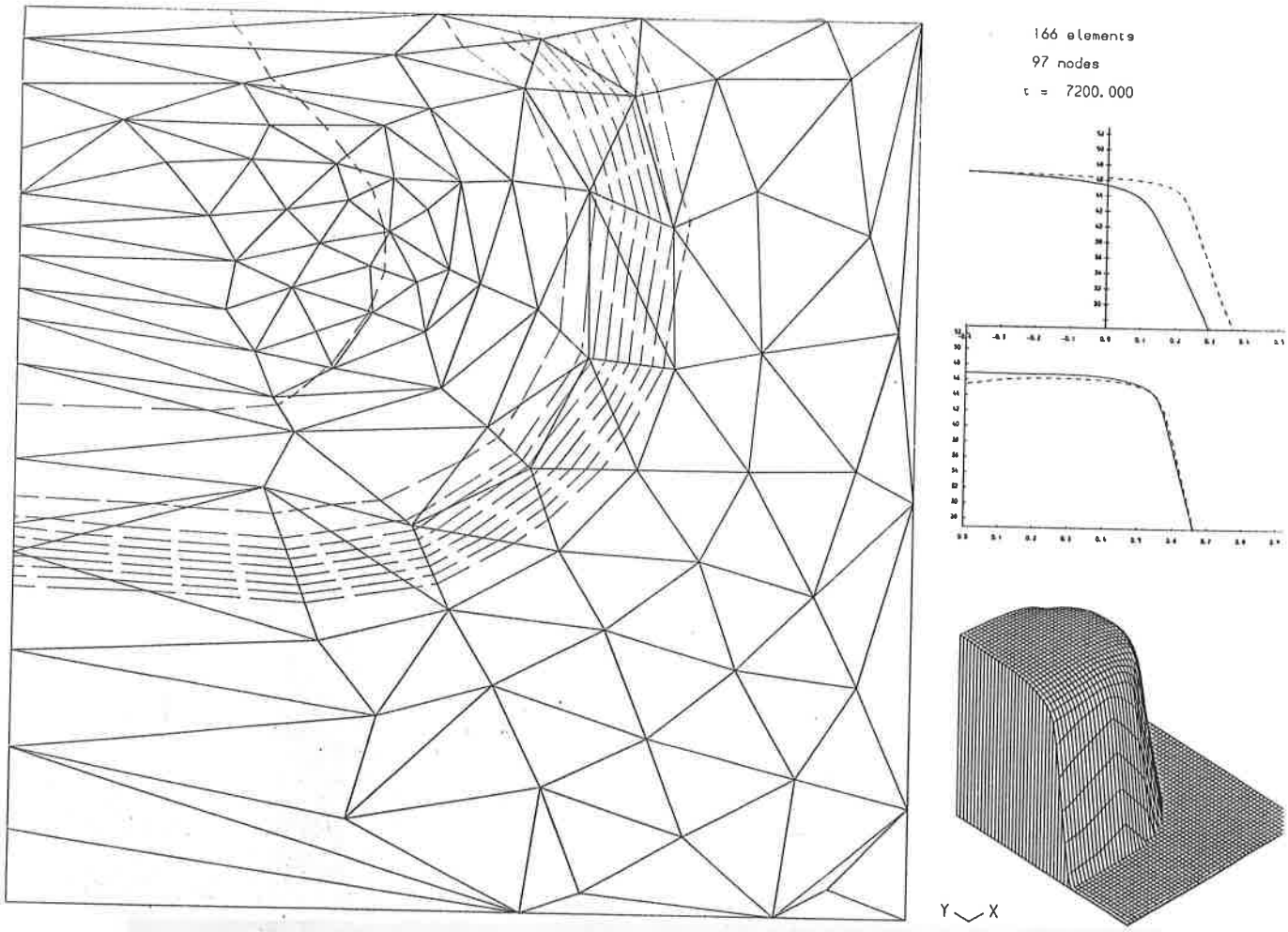
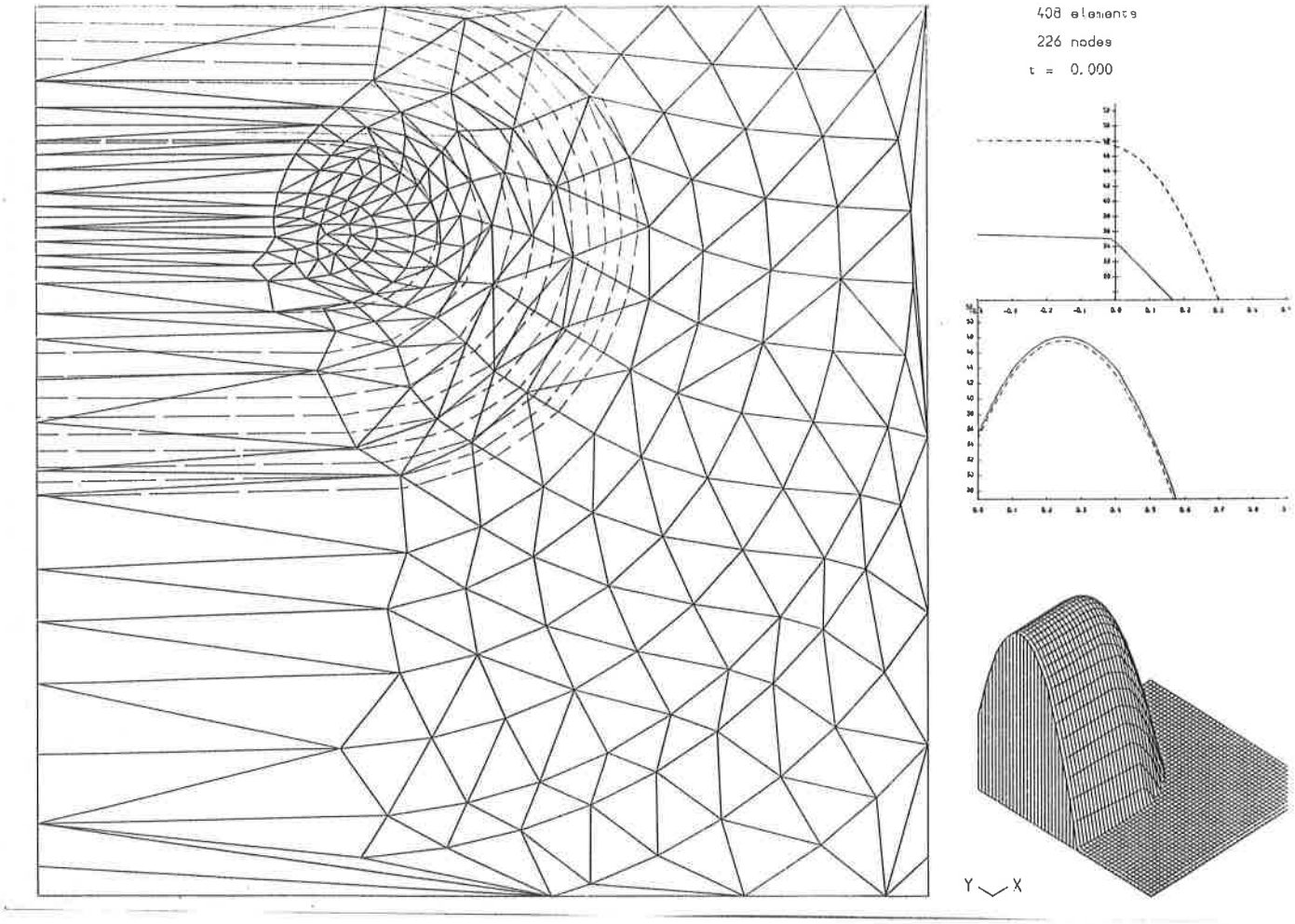


Figure 4. (a) As 3(a) but with DG=1.4.
(b) After 30mins anneal at 950 C.





(c) After 2 hours anneal.
(d) As (c) but with no oxidation.



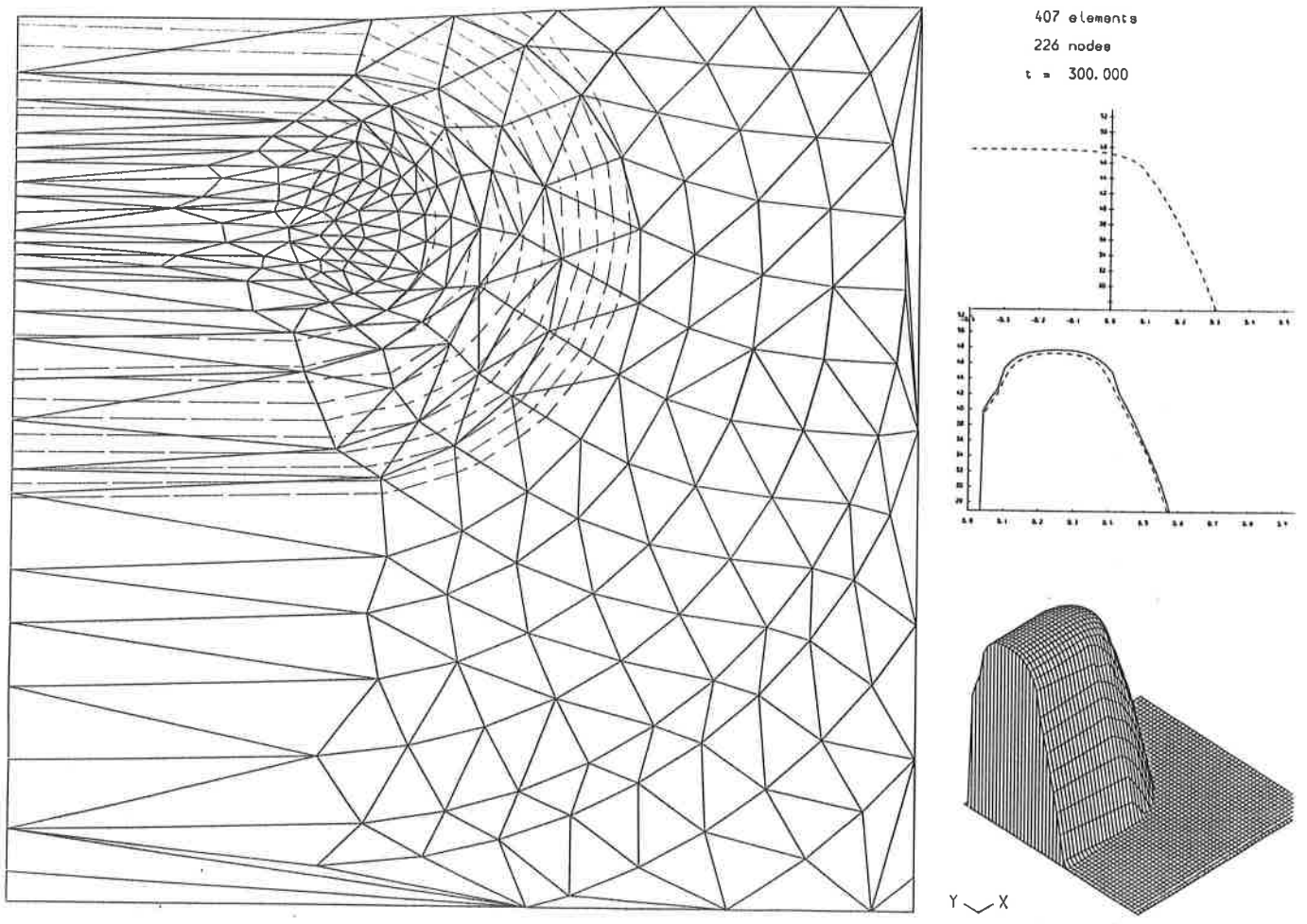
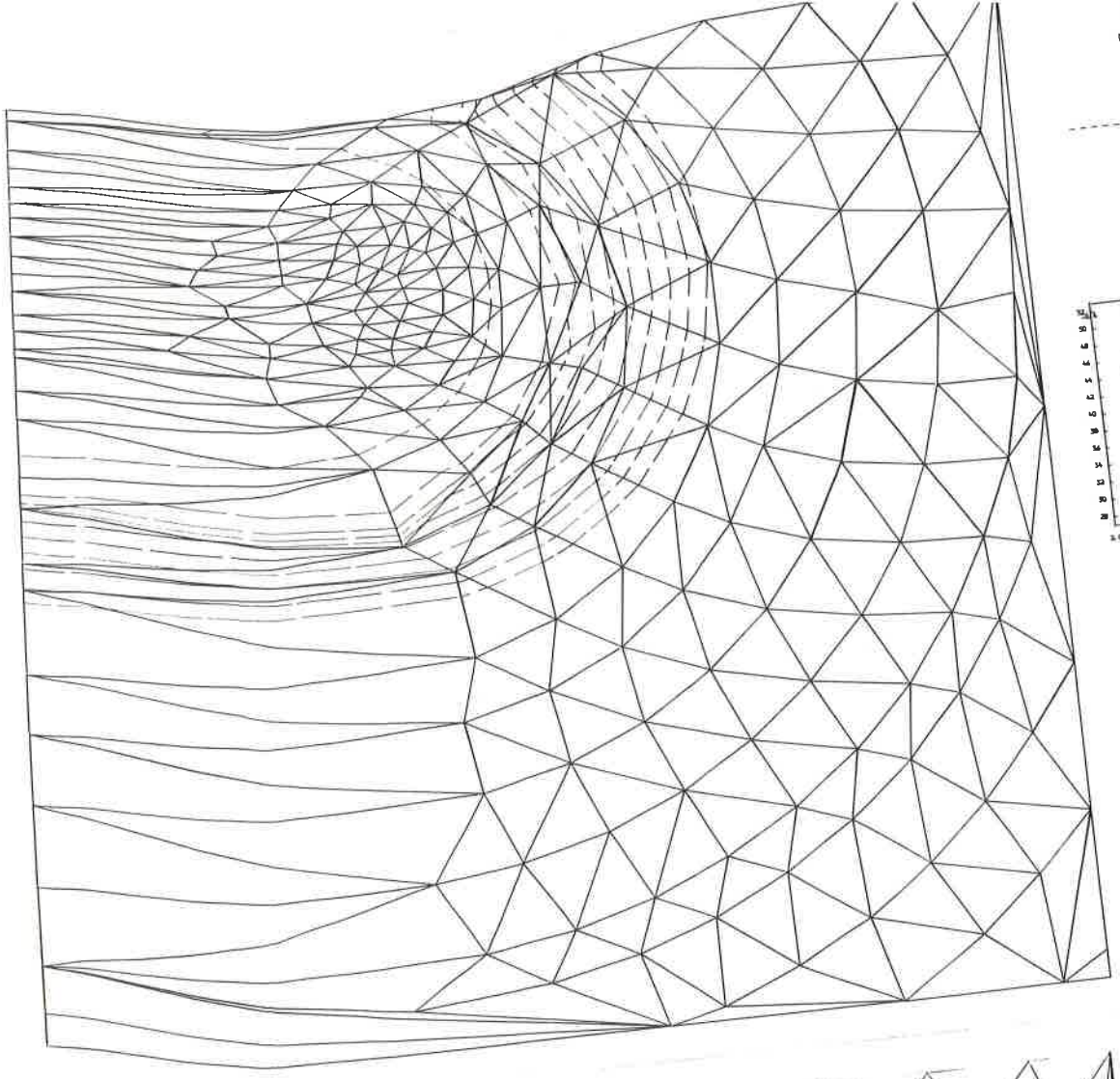
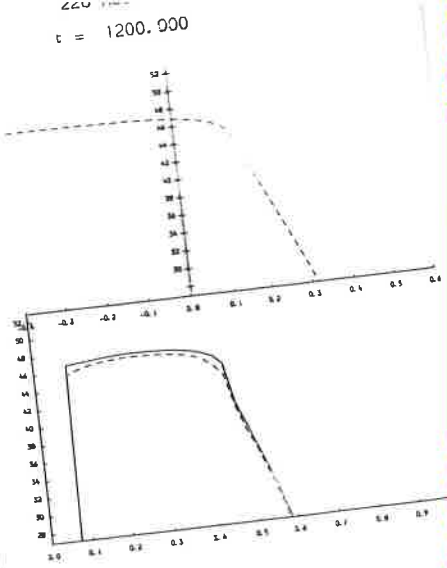
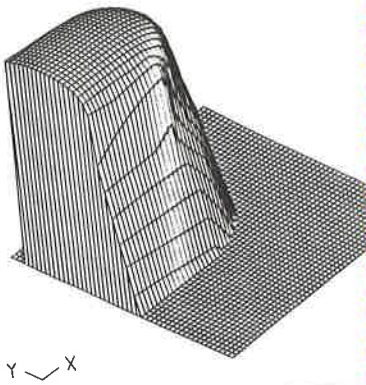
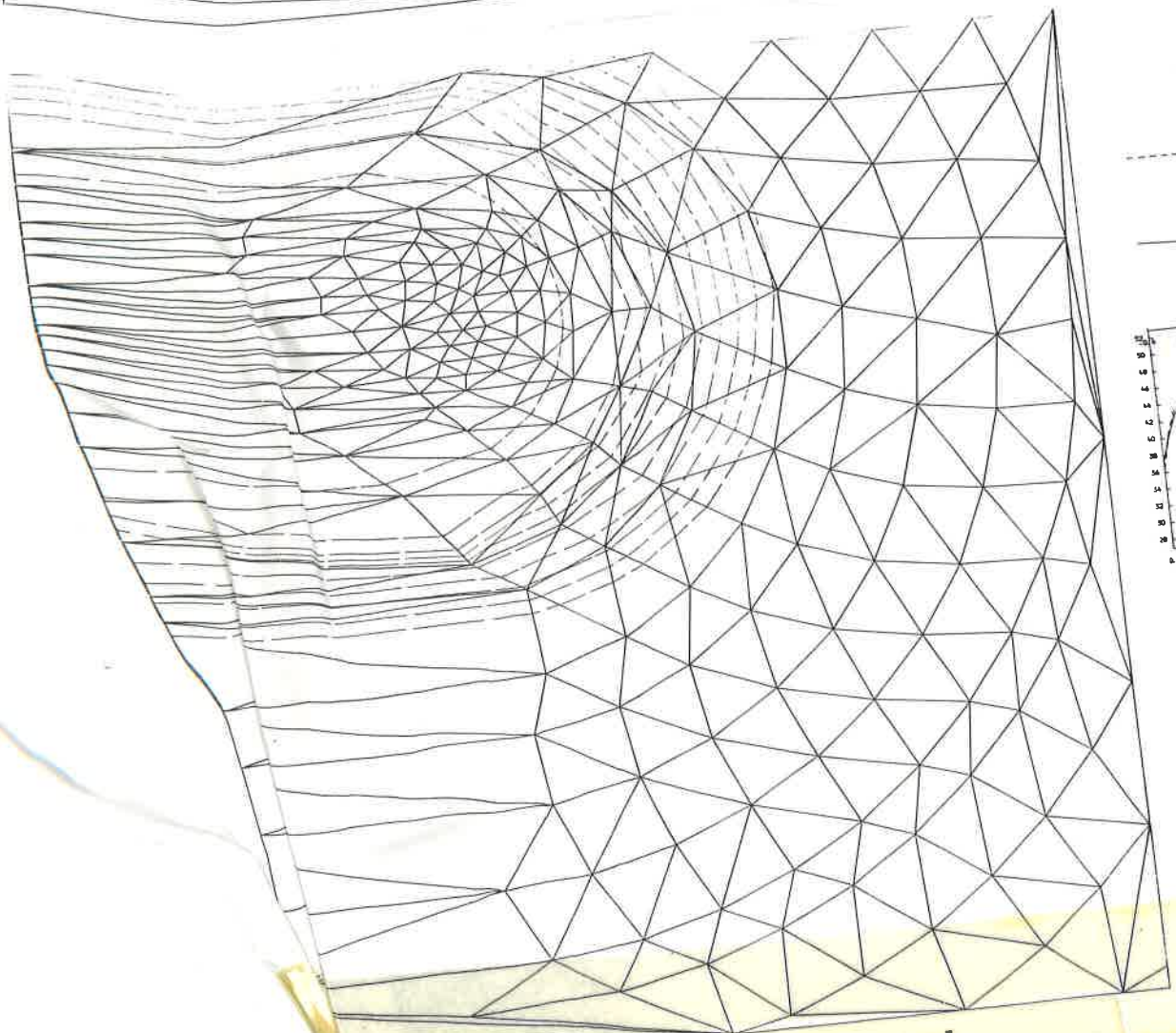


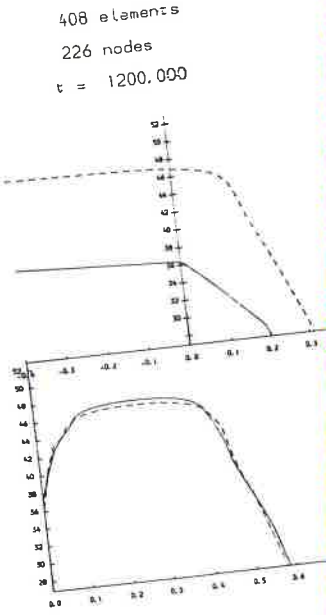
Figure 5. (a) As 3(a) but with DG=0.9
(b) After 5 mins anneal at 950 C.

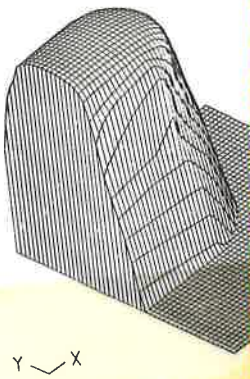












(c) After 20 mins anneal.
(d) As (c) but with no oxidation.