JOURNAL OF COMPUTATIONAL PHYSICS **141,** 1–21 (1998)
ARTICLE NO. CP985900

# A Conserving Discretization for the Free Boundary in a Two-Dimensional Stefan Problem

Guus Segal,* Kees Vuik,* and Fred Vermolen†

**Faculty of Technical Mathematics and Informatics, Delft University of Technology, P.O. Box 5031, NL 2600 GA Delft, The Netherlands; †Laboratory of Materials Science, Delft University of Technology, P.O. Box 5045, NL 2600 GA Delft, The Netherlands*
E-mail: c.vuik@math.tudelft.nl

Received February 18, 1997; revised September 25, 1997

The dissolution of a disk-like $Al_2Cu$ particle is considered. A characteristic property is that initially the particle has a nonsmooth boundary. The mathematical model of this dissolution process contains a description of the particle interface, of which the position varies in time. Such a model is called a Stefan problem. It is impossible to obtain an analytical solution for a general two-dimensional Stefan problem, so we use the finite element method to solve this problem numerically. First, we apply a classical moving mesh method. Computations show that after some time steps the predicted particle interface becomes very unrealistic. Therefore, we derive a new method for the displacement of the free boundary based on the balance of atoms. This method leads to good results, also, for nonsmooth boundaries. Some numerical experiments are given for the dissolution of an $Al_2Cu$ particle in an $Al-Cu$ alloy. © 1998 Academic Press

*Key Words:* Stefan problem; moving finite elements; conserving discretization; particle dissolution; alloy homogenization.

## 1. INTRODUCTION

Heat treatment of metals is often necessary to optimise their mechanical properties both for further processing and for final use. During the heat treatment, the metallurgical state of the material changes. This change can either involve the phases being present or the morphology of the various phases. Whereas the equilibrium phases often can be predicted quite accurately from thermodynamic models, there are no general models for microstructural changes nor general models for the kinetics of these changes. In the latter cases both the initial morphology and the transformation mechanisms have to be specified explicitly. One of these processes, which is both of large industrial and scientific interest and amenable to modelling, is the dissolution of second phase particles in a matrix with a uniform initial composition.

To describe this particle dissolution in rigid media several physical models have been developed, incorporating the effects of long-distance diffusion [3, 21, 28] and nonequilibrium conditions at the interface [1, 13]. The long-distance diffusion models imply that the processes at the interface between particle and matrix proceed infinitely fast. Therefore, these models provide an upper bound for the dissolution rate.

Whelan [28] considered particles dissolving in an infinite medium using the stationary interface approximation. He derived an analytical solution of the diffusion equation in an infinite medium for spherical co-ordinates by the use of the Laplace transformation in time. The accuracy of the model increases with increasing interparticle distance.

All references discussed below transform the cell in which the particle dissolves into a cell which is equally shaped as the particle, requiring the transformed cell volume to be equal to the original cell volume. This allows a one-dimensional treatment of the moving boundary problem which can be solved easily using a finite difference discretization method. This method can be used for planar, cylindrical, and spherical geometry.

Baty, Tanzilli, and Heckel [3] were the first authors to apply a numerical method using a finite difference method to evaluate the interface position as a function of dissolution time. Their model is also applicable to situations in which the interparticle distance is small, i.e. when soft impingement occurs. The model they proposed is based on the assumption of local equilibrium at all stages of the dissolution process. They applied their numerical analysis to dissolving $Al_2Cu$ particles in aluminum. In their models the $Al_2Cu$ particles were assumed to be spherical. The poor fit of their calculations with the experimental data is probably due to the interface reactions or to the nonspherical shape of the regular particles which were not incorporated into their numerical model.

Tundal and Ryum [21] considered the effects of a finite cell size for spherical particles as well. They too applied a numerical method using a finite difference method to predict the dissolution kinetics. Their model is based on the assumption of local equilibrium during the entire dissolution process. They introduced a log-normal distribution for both the particle and cell size (i.e., the logarithm of the size is distributed according to a normal distribution). They showed that macroscopic dissolution rates depend strongly on the shape of the particle size distribution curve and possible interactions between the neighbouring cells.

Nolfi's model [13] did not include the interface migration, but as far as we know, it is the first model which incorporated nonequilibrium conditions at the interface. In the Nolfi model nonequilibrium conditions at the interface were incorporated by the introduction of a Robbins condition which relates the concentration gradient with the concentration at the interface. This semi-analytical solution consists of an infinite series solution for the concentration profile. Their method, however, is only accurate in the early stages of the dissolution process.

Aaron and Kotler [1] incorporated the nonequilibrium conditions at the interface too. However, their approach is only applicable for those situations in which the interparticle distance is sufficiently large, i.e. the diffusion fields do not impinge. They transformed the Robbins problem of Nolfi into a Dirichlet problem, in which the concentration is fixed at all stages of the dissolution process. Combining Whelan's [28] analytical approach for the interface velocity as a function of the annealing time, with a relation between the interface concentration and the interface position, they evaluated the interface position using a Picard-type iteration method. Aaron and Kotler also incorporated the effects of the particle curvature into their model using the Gibbs–Thomson equation [1]. In their model both the interface position and the interface concentration were taken momentarily stationary during the evaluation of the interface position as a function of time.

The effects of interfacial reactions on the rate of the dissolution of spherical particles in both infinite and finite media was examined by Vermolen and Van der Zwaag numerically [23]. Using a finite difference method it was shown that interfacial reactions can have a significant effect on the dissolution rate and, hence, on the concentration profiles in the matrix during particle dissolution.

Due to the difference in the crystal structures of the particle and its surroundings the total surface energy may be minimal for a disk-like geometry. Hence, an algorithm capable of solving two-dimensional moving boundary problems is desired. In the present work we introduce a general algorithm that can be used to solve two-dimensional problems. It is based on the application of the finite element method on two-dimensional moving boundary problems. A reason to use finite elements is that it allows the use of unstructured grids. Hence the finite element method is more flexible than other discretization methods using structured grids only. Especially in moving boundary problems as the ones that will be demonstrated in the remainder of this paper, unstructured grids are essential.

Before we derive our numerical algorithm, we present the mathematical model in Section 2. Actually, it turns out that this is a standard Stefan problem [4, 19, 25, 26]. We repeat the derivation of the Stefan condition at the free boundary, since this derivation is needed to improve the numerical method. In Section 3 we specify a numerical scheme to solve the Stefan problem. It will be shown that straightforward discretization of the equations and boundary conditions may lead to unrealistic interfaces. These problems are investigated and a new algorithm (Section 4) which produces nice results is presented. This improved algorithm is based on the derivation of the Stefan boundary condition. Section 5 contains some details of our remeshing techniques. In Section 6 results of the new algorithm applied to some practical problems will be demonstrated. The conclusions are formulated in Section 7.

## 2. A MATHEMATICAL MODEL FOR A DISSOLUTION PROCESS

Consider an $Al_2Cu$ particle in an $Al{-}Cu$ alloy at a given temperature. The initial concentration of $Al_2Cu$ in the aluminum phase is equal to $c_0$ (mol/m$^3$), whereas $c_{\text{part}}$ denotes the concentration of $Al_2Cu$ in the particle. When the temperature is increased, dissolution of the $Al_2Cu$ particle sets in. At the interface the $Al_2Cu$ concentration is $c_{\text{sol}}$ ($c_{\text{part}} > c_{\text{sol}} > c_0$).

To describe the mathematical model we use the geometry as given in Fig. 1. The domain filled with aluminum is denoted by $\Omega(t)$. The boundary of this domain consists of the interface $S(t)$ and the outer boundaries $\Gamma_i$, $i \in \{1, 2, 3, 4\}$. The outer boundaries are fixed in time, except the intersections of $\Gamma_1$ and $\Gamma_4$ with $S(t)$. In the aluminum-rich phase $\Omega(t)$, the $Cu$ concentration $c(x, y, t)$ satisfies the diffusion equation

$$\frac{\partial c}{\partial t} = \mathbb{D}\Delta c, \quad (x, y) \in \Omega(t),\ t \in (0, T]. \tag{1}$$

The diffusion coefficient $\mathbb{D}$ (m$^2$/s) is supposed to be independent of concentration. As the initial condition we use

$$c(x, y, 0) = c_0(x, y), \quad (x, y) \in \Omega(0), \tag{2}$$

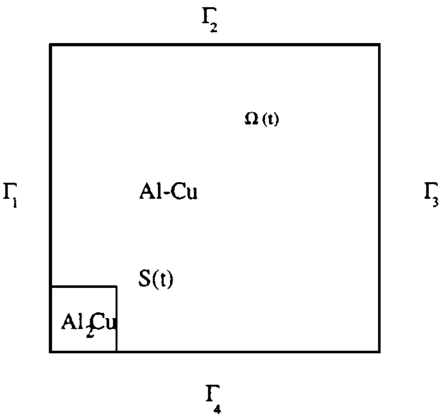

**FIG. 1.** Geometry of an $Al_2Cu$ particle in aluminum.

where $\Omega(0)$ is prescribed. We assume no flux of *Cu* through the outer boundaries, so

$$\frac{\partial c}{\partial \mathbf{n}}(x, y, t) = 0, \quad (x, y) \in \Gamma_i, i \in \{1, 2, 3, 4\}, t \in [0, T]. \tag{3}$$

At the interface the concentration satisfies the equation

$$c(x, y, t) = c_{\text{sol}}, \quad (x, y) \in S(t),\ t \in (0, T]. \tag{4}$$

To determine the position of the interface one extra condition is necessary. To derive this boundary condition for a spatially three-dimensional problem, we consider a small part of the interface. Suppose that the interface is smooth, which means that it can locally be described by differentiable functions. For a small time step $\Delta t$ the interface moves in the direction perpendicular to the interface. The $x$-axis is chosen along the normal. With this choice the position of the interface is locally described by the relation $x = S(t)$. We consider a control volume of width $\Delta y$ and $\Delta z$. The intersection of the control volume with the surface $y = 0, z = 0$ is shown in Fig. 2. The balance of *Cu* atoms leads to the following equation:

$$(S(t + \Delta t) - S(t))\Delta y \Delta z c_{\text{sol}} = (S(t + \Delta t) - S(t))\Delta y \Delta z c_{\text{part}} - \mathbb{D}\frac{\partial c}{\partial x}\Delta y \Delta z \Delta t. \tag{5}$$

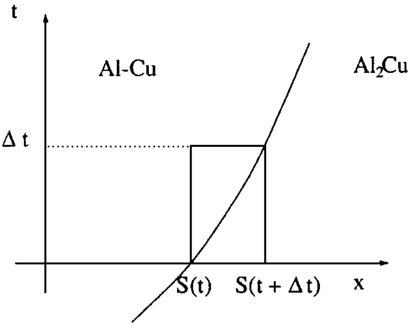

**FIG. 2.** The control volume.

Dividing (5) by $\Delta y \Delta z \Delta t$ and taking the limit $\Delta t \to 0$, one obtains

$$(c_{\text{part}} - c_{\text{sol}}) \frac{dS}{dt} = \mathbb{D} \frac{\partial c}{\partial x}.$$

So the extra boundary condition at the interface is

$$(c_{\text{part}} - c_{\text{sol}}) v_n(x, y, t) = \mathbb{D} \frac{\partial c}{\partial \mathbf{n}}(x, y, t), \quad (x, y) \in S(t),\ t \in (0, T], \tag{6}$$

where $\mathbf{n}$ is the unit normal vector on the interface pointing outward with respect to $\Omega(t)$ and $v_n$ is the normal velocity of the interface. The moving boundary problem given by (1), (2), (3), (4), and (6) is known as a Stefan problem [4].

The normalised concentration $\hat{c} = (c - c_0)/(c_{\text{sol}} - c_0)$, together with a characteristic length scale $L$ and time scale $L^2/\mathbb{D}$, is used to make the problem dimensionless. After replacing $\hat{c}$ by $c$ we obtain the following Stefan problem: find a concentration $c$ and an interface $S$ such that

$$\frac{\partial c}{\partial t} = \Delta c, \quad (x, y) \in \Omega(t), \quad t \in (0, T], \tag{7}$$

$$c(x, y, 0) = 0, \quad (x, y) \in \Omega(0), \tag{8}$$

$$\frac{\partial c}{\partial \mathbf{n}}(x, y, t) = 0, \quad (x, y) \in \Gamma_i, i \in \{1, 2, 3, 4\}, t \in [0, T], \tag{9}$$

$$c(x, y, t) = 1, \quad (x, y) \in S(t), t \in (0, T], \tag{10}$$

$$v_n(x, y, t) = \lambda \frac{\partial c}{\partial \mathbf{n}}(x, y, t), \quad (x, y) \in S(t), t \in (0, T], \tag{11}$$

where the dimensionless number $\lambda$ is given by

$$\lambda = \frac{c_{\text{part}} - c_0}{c_{\text{part}} - c_{\text{sol}}}. \tag{12}$$

An extensive review of literature on Stefan problems is given in [20]. This bibliography contains 2500 titles on moving boundary problems. In [16] a recent overview is given of numerical methods to simulate convection/diffusion phase change problems.

## 3. SOLUTION OF THE STEFAN PROBLEM

Various numerical methods are known to solve Stefan problems. In Crank [4] the following types of method are distinguished: front-tracking, front-fixing, and fixed-domain methods. In a front-fixing method a transformation to body-fitted curvilinear coordinates is used (a special case is the isotherm migration method (IMM)). A drawback is that such a transformation can only be used for a relatively simple geometry. Fixed-domain methods are the enthalpy method (EM) and the variational inequality method (VI). Various methods are compared in [5, 27]. The latter methods (IMM, EM, VI) are only applicable when the interface is an equi-concentration line. In the near future we want to simulate also dissolution processes with a varying concentration on the interface (e.g., first-order reaction at the interface [23], or dissolution in ternary Al-alloys [22]). Therefore, we choose a front-tracking method to solve our problem numerically, although the current problem can also be solved by other methods (IMM, EM, VI). The front-tracking method we apply is a two-dimensional extension of the method of Murray and Landis [12] as described in [10, 11, 17, 27]. A detailed description of the adaptation of the interface in time is lacking in most of the literature on front-tracking methods, except [10]. We present various methods for this adaptation in this section and Section 4.

Our method to solve the Stefan problem runs as follows. In the first time-step the diffusion equation (7), together with the initial (8) and boundary conditions (9) and (10), is solved. On the free boundary $S$ only the Dirichlet condition (10) is used. In the next time steps first the boundary is moved using the boundary condition (11). This means that the co-ordinates of the free surface at time $t + \Delta t$ are approximated by

$$\mathbf{x}(t + \Delta t) = \mathbf{x}(\mathbf{t}) + v_n \Delta t \, \mathbf{n} = \mathbf{x}(\mathbf{t}) + \lambda \frac{\partial c}{\partial \mathbf{n}} \Delta t \, \mathbf{n}. \tag{13}$$

Once the boundary is moved, the concentration $c$ can be computed in the new region using Eq. (7). However, the computation of the concentration implies that we have to compute $(c(t+\Delta t)-c(t))/\Delta t$. We do not know $c(t)$ in the nodal points since, due to the displacement of the boundary also, all nodes have been moved [2]. So either we have to interpolate the concentration to the new nodes, or we have to make a correction for the displacement. Interpolation is of course possible, but relatively expensive. The correction is much more simple. If we compute the time-derivative based on the old and the new points, then it is clear that we actually use a material derivative,

$$\frac{Dc}{Dt} = \frac{\partial c}{\partial t} + \mathbf{u}_{\text{mesh}} \cdot \nabla c, \tag{14}$$

where the so-called mesh velocity is defined by

$$\mathbf{u}_{\text{mesh}} = \frac{\mathbf{x}(t + \Delta t) - \mathbf{x}(t)}{\Delta t}, \tag{15}$$

with $\mathbf{x}$ the co-ordinate vector in a point. Substituting Eq. (14) into Eq. (7) leads to

$$\frac{Dc}{Dt} - \Delta c - \mathbf{u}_{\text{mesh}} \cdot \nabla c = 0. \tag{16}$$

This approach is common practice in fluid mechanics, where it is known as the arbitrary Lagrangian Eulerian approach (ALE) (see, for example, [6, 7, 8, 9, 15]).

The discretization of Eq. (16) is performed by a standard Galerkin finite element method using linear triangles. The time-discretization is performed by an implicit Euler method.

ALGORITHM 1. A straightforward method to compute the new position of the free boundary: Compute the gradient of the concentration in the elements connected to the free boundary. Compute the gradient of the concentration and the normal in the vertices at the free boundary by averaging over neighbouring elements. Compute $\partial c/\partial \mathbf{n}$ in the vertices by the inner product of the gradient and the normal vector. Apply formula (13) to compute the new positions of the vertices.

Initially the normal of the interface is not defined in the middle point of $S$ (see Fig. 1), where the boundary has a corner of 90°. Note that in fact the normal is not defined in vertices at the boundary unless the two surrounding line elements have the same direction. For that

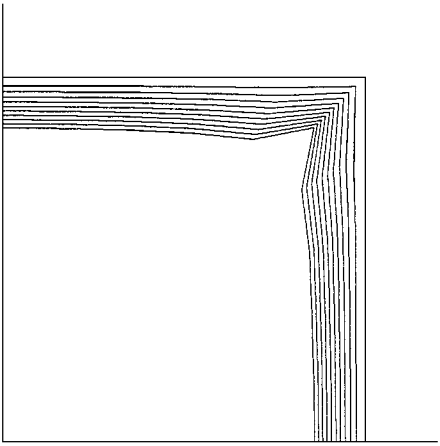

**FIG. 3.** Free boundary in the first 10 time steps.

reason the outward normal in a vertex point is defined as the normalized mean value of the normals at the two line elements adjacent to that point. Using this definition of the normal vector, the displacement of the vertices of the boundary can immediately be deducted from Eq. (13).

A different way of computing the normal in the vertices is presented by Lynch [10]. He proposes to take a weighted average of the surrounding normals, where the weights are defined by the length of the adjacent boundary elements. A clear disadvantage of this approach is that the normal depends on the local grid size. For example, if at a 90° corner the mesh size at one side is larger than at the other one, the normal will not be defined under an angle of 45° as in our method.

Figure 3 shows the computed free boundary during the first 10 time steps. The corresponding mesh is shown in Fig. 4. It is clear that in the middle point of the free boundary, where we start with the 90° corner, the free boundary moves more slowly than in the other points. However, from the physical point of view one would expect this point to move faster than the surrounding points. To investigate this strange behaviour we consider the artificial case where all the equi-concentration lines are parallel to the free boundary. Figure 5 shows

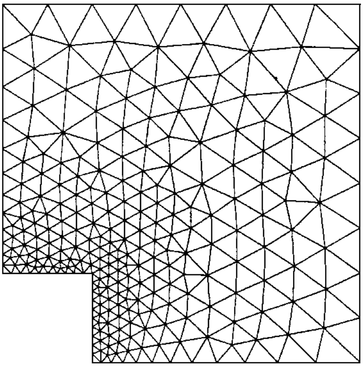

**FIG. 4.** The mesh used.

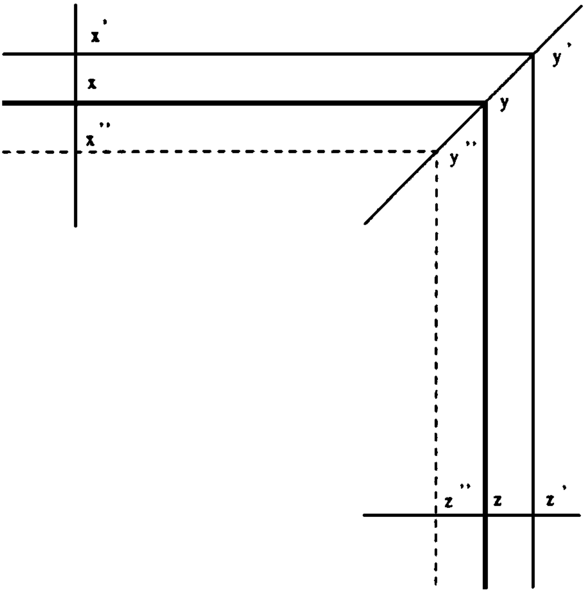

**FIG. 5.** Assumed position of free boundary in case of parallel concentration lines.

the boundary (solid line) and one such a concentration line. The dashed line denotes the new position of the free boundary, one would intuitively expect, after one time step $\Delta t = 1$ with the parameter $\lambda$ equal to 1. Let the difference in concentration between free boundary and the plotted equi-concentration line be equal to $\Delta c$. Let the distance between the points $\mathbf{x}$ and $\mathbf{x}'$ be equal to $\Delta x$. Then the value of $\partial c/\partial \mathbf{n}$ is approximately $\Delta c/\Delta x$. However, the distance between the points $\mathbf{y}$ and $\mathbf{y}'$ is equal to $\sqrt{2}\Delta x$. Hence, along the line $\mathbf{x}''$, $\mathbf{x}'$ the value of $\partial c/\partial \mathbf{n}$ is approximately $\Delta c/\sqrt{2}\Delta x$. Since the average normal is equal to $\mathbf{n} = (\sqrt{2}/2)\binom{1}{1}$ this implies that the free boundary in the point $\mathbf{y}$ does not move to $\mathbf{y}''$, but only to the point precisely in the middle of $\mathbf{y}$ and $\mathbf{y}''$. The reason for this inconsistency is, in fact, that the Stefan boundary condition is derived under the assumption of a smooth boundary. In the case of a nondifferentiable boundary, the limit does not exist and only the integral form (5) of the Stefan boundary condition makes sense.

A natural solution of the problem mentioned above is not to use the gradient and the normal in the vertices, but only those in the mid-side points of the elements. Since the gradient of the concentration is constant per triangle, the value of both the normal and the normal derivative in these mid-side points is unique. So one can compute the displacement of the mid-side points by Eq. (13). The only remaining problem is to define how the vertices must be moved. An obvious choice is to choose the new positions of the vertices right in the middle of the new positions of the mid-side points. This method is denoted as **Algorithm 2**.

Computations show a very smooth progress of the free boundary. There is one serious drawback in this method. Suppose that none of the mid-side points are moved, for example because $\lambda = 0$. In that case one would expect that the vertices also will be kept at their old positions. However, after the first time step the vertex corresponding to the 90° corner has been moved inwardly. In the next step an extra smoothing is applied and if this process is repeated we end up with a straight line. For an initial corner larger than 180°, the boundary even moves into the wrong direction. So we are looking for a method that keeps the vertices

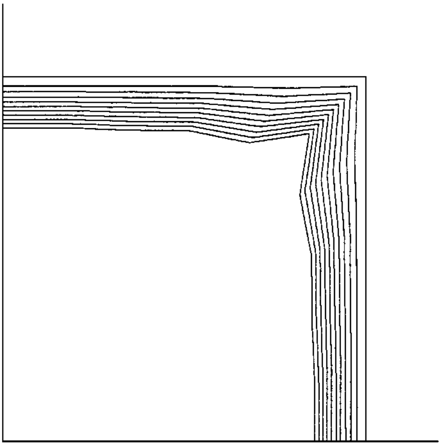

**FIG. 6.** Free boundary in the first 10 time steps computed by the "weak approach."

fixed as long as the mid-side points do not move, but also lacks the behaviour shown in Fig. 3. In the next section we will introduce such a method.

Before considering our new method we consider a variant of a method proposed by Lynch [10]. Lynch suggests using a weighted average method to compute the position of the new boundary. However, for an equidistant mesh size along the boundaries his method is identical to our Algorithm 1 and for nonequidistant mesh sizes it is even worse. Nevertheless, his derivation, based on a weak formulation of the Stefan condition (11), inspired us to use a Galerkin approximation of Eq. (11). The idea is that this is more an integral approach than a differential approach. The Galerkin formulation of Eq. (11) is given by

$$\int_S v_n \phi \, d\Gamma = \int_S \lambda \frac{\partial c}{\partial n} \phi \, d\Gamma, \tag{17}$$

with $\phi$ an arbitrary test function. Exact integration of Eq. (17) leads to a tridiagonal system of equations, where the matrix to be inverted is the standard mass matrix along the free boundary. Lumping of this matrix reduced this method to Algorithm 1. Figure 6 shows the free boundary during the first 10 time steps, using the consistent mass matrix. Indeed the boundary is better than the one shown in Fig. 3, but still the corner problem is visible.

## 4. A NEW METHOD TO COMPUTE THE FREE BOUNDARY POSITION

Before deriving our new approach for the computation of the free boundary, we first formulate two demands a good numerical method should satisfy:

- Let $\Delta x$ be a measure for the displacement of the boundary. It is necessary that $\Delta x \to 0$ when $\Delta t \to 0$.
- Suppose that the corner between two adjacent elements at the moving boundary, measured with respect to the dissolving material, is equal to $\alpha$. If $\alpha < 180^\circ$ then one expects that this corner increases as soon as the material dissolves. The reason for such a behaviour is that in the case of a straight boundary, diffusion can take place in one direction only, whereas in a corner, diffusion is possible in various directions. So a numerical method should satisfy this property too.

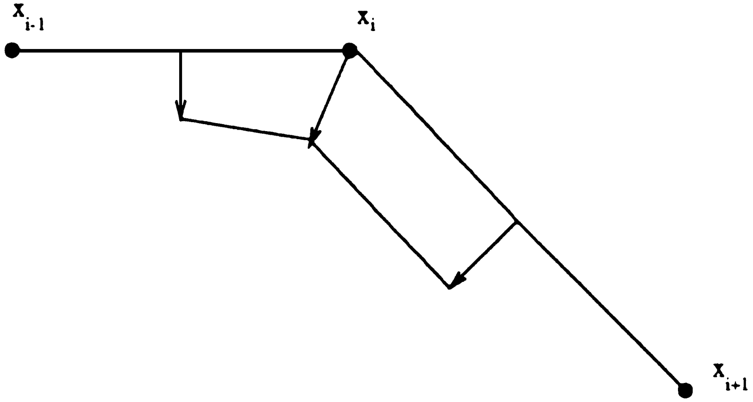

**FIG. 7.** Area occupied by the region defined by the displacement of the vertex.

It is clear that the methods mentioned in Section 3 do not fulfil these two demands at the same time. Algorithm 1 does not satisfy the second demand, whereas Algorithm 2 does not fulfil the first demand. We propose the following method.

ALGORITHM 3. Improved scheme for the computation of the free boundary. Compute the gradient of the concentration in the elements connected to the free boundary. Compute $\partial c/\partial \mathbf{n}$ in the mid-side points of the elements at the free boundary, by the inner product of the gradient and the normal vector. Apply formula (13) to compute the new positions of the mid-side points. Define the new positions of the vertices along the normal, in such a way that, the area occupied by the region defined by the displacement of the vertex and its surrounding mid-side points (Fig. 7) is equal to the mean value of the areas defined by the displacement of both mid-side points (Fig. 8).

This method combines the smoothing properties of Algorithm 2 with the nondisplacement of the vertex in case the mid-side points remain unchanged. The motivation for this approach is the following. From the derivation of the Stefan boundary condition it follows that the area of the particle that has been dissolved is equal to the amount of diffused material. The flux through the element $(x_{i-1}, x_i)$ is approximately equal to

$$\mathbb{D}\frac{\partial c}{\partial \mathbf{n}}(x_{i-1/2})l_i\,\Delta t, \tag{18}$$

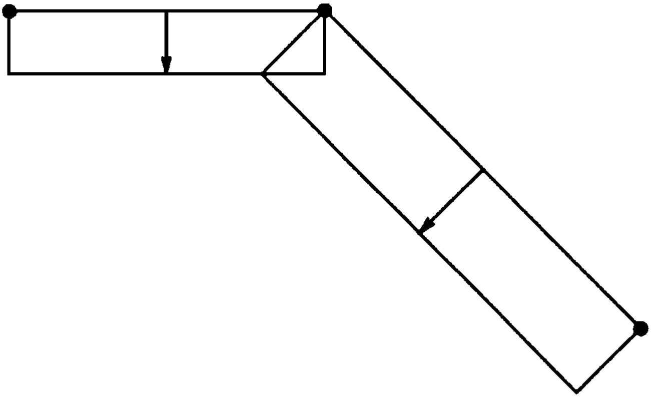

**FIG. 8.** Area occupied by the region defined by the displacement of both mid-side points.

with $l_i$ the length of the line element $(x_{i-1}, x_i)$. Hence, the amount of diffused material through the boundary $(x_{i-1/2}, x_{i+1/2})$ is equal to

$$\frac{\Delta t}{2}\left(\mathbb{D}\frac{\partial c}{\partial \mathbf{n}}(x_{i-1/2})l_i + \mathbb{D}\frac{\partial c}{\partial \mathbf{n}}(x_{i+1/2})l_{i+1}\right). \tag{19}$$

The amount $M$ of material dissolved, is approximately equal to $(c_{\text{part}} - c_{\text{sol}})O$, where $O$ is the area defined in Fig. 7. Due to the balance of atoms $M$ must be equal to the amount of diffused material given in Eq. (19). Making this equation dimensionless this is equal to (see (12))

$$O = \frac{\lambda \Delta t}{2}\left(\frac{\partial c}{\partial \mathbf{n}}(x_{i-1/2})l_i + \frac{\partial c}{\partial \mathbf{n}}(x_{i+1/2})l_{i+1}\right). \tag{20}$$

The right-hand side of (20) is the mean value of the areas defined in Fig. 8.

There is a drawback of Algorithm 3, which can be explained as follows. Once the displacement of the vertices has been computed by means of Algorithm 3, the mid-side points are moved to new positions, since they are always in the middle of two vertices. As a consequence the amount of dissolved material is no longer equal to $M$. If we want to have a displacement which gives an amount of dissolved material that is exactly equal to $M$, the situation is more complex. Consider two adjacent line elements $(x_{i-1}, x_i)$, and $(x_i, x_{i+1})$, with length $l_i$ and $l_{i+1}$, respectively (Fig. 7). The mid-side points of these elements are denoted by $x_{i-1/2}$ and $x_{i+1/2}$. Let the displacement in the mid-side points computed from formula (13) be equal to $\delta x_{i-1/2}$ and $\delta x_{i+1/2}$. The new position of vertex $x_i$ is denoted by $\hat{\mathbf{x}}_i$. The length of the displacement is given as $\Delta x_i = \|\hat{\mathbf{x}}_i - \mathbf{x}_i\|$. Once the displacement in the vertices is computed, also the displacements in the mid-side points change. In order to get both a local and global equilibrium in the amount of dissolved material, it is necessary that the new area be equal to

$$\frac{1}{2}l_i\delta x_{i-1/2} + \frac{1}{2}l_{i+1}\delta x_{i+1/2}. \tag{21}$$

The area $O$ depends on $\Delta x_i$, $\Delta x_{i-1/2}$ and $\Delta x_{i+1/2}$, where $\Delta x_{i-1/2}$ is the adapted length of the displacement in $x_{i-1/2}$. Since $\Delta x_{i-1/2}$ and $\Delta x_{i+1/2}$ depend on $\Delta x_{i-1}$, $\Delta x_i$, and $\Delta x_{i+1}$ the relation is nonlinear. In order to solve this system of nonlinear equations we propose the following algorithm.

ALGORITHM 4. "Exact" satisfaction of balance between dissolution and diffusion.

```
for all i do
    Compute $\Delta x_i$ from $\delta x_{i-1/2}$ and $\delta x_{i+1/2}$ according to Algorithm 3.
    $\hat{\mathbf{x}}_i = \mathbf{x}_i + \Delta x_i \mathbf{n}_i$.
end for
while not converged do
    for all i do
        $\hat{\mathbf{x}}_{i-1/2} = \frac{1}{2}(\hat{\mathbf{x}}_{i-1} + \hat{\mathbf{x}}_i)$.
        Compute $\Delta x_i$ from Algorithm 3 using the areas occupied by the known
            quadrilaterals $(\mathbf{x}_{i-1/2}, \mathbf{x}_i, \hat{\mathbf{x}}_i, \hat{\mathbf{x}}_{i-1/2})$ and $(\mathbf{x}_{i+1/2}, \mathbf{x}_i, \hat{\mathbf{x}}_i, \hat{\mathbf{x}}_{i+1/2})$.
        $\hat{\mathbf{x}}_i = \hat{\mathbf{x}}_i + \omega(\mathbf{x}_i + \Delta \mathbf{x}_i \mathbf{n}_i - \hat{\mathbf{x}}_i)$.
    end for
end while
```

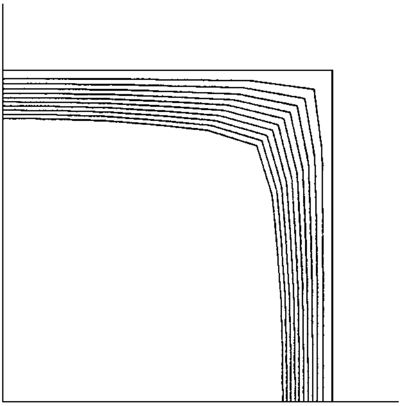

**FIG. 9.** Free boundary during the first 10 time-steps using Algorithm 3.

Numerical experiments showed that, in the case of a relaxation parameter $\omega = 1$, Algorithm 4 did not converge. The free surface switched between two states for successive iterations. This behaviour is caused by an overestimation of the correction in each iteration. To solve this problem an underrelaxation factor $\omega < 1$ has been tried. Practical computations show that the iteration process is rather insensitive for the value of the $\omega$. For $\omega = 0.5$ we usually got convergence within five iterations. The process is stopped as soon as the difference between the area due to the mid-side displacements, as defined in Eq. (19) and the final area is less than 1%.

From Figs. 9 and 10 it appears that, especially during the first time-steps, the results from Algorithm 3 and Algorithm 4 are quite different. In order to compare the accuracy of both algorithms we halved the space-step (Figs. 11 and 12). It is clear from these figures, that the boundary of Algorithm 3 is unacceptable, whereas that of Algorithm 4 seems reasonably good. The reason for this strange behaviour is that the new positions of the two mid-side points around the sharp corner are very close to the line $x = y$ (Fig. 13). As a consequence the first estimate is very inaccurate. Iteration as used in Algorithm 4, however, is able to solve this problem nicely. If we enlarge the time-step, the displacement of the mid-side

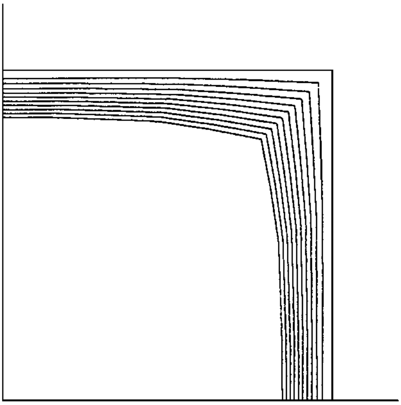

**FIG. 10.** Free boundary during the first 10 time-steps using Algorithm 4.

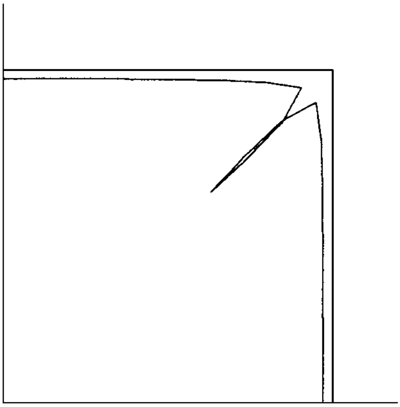

**FIG. 11.** Free boundary during the first time-step using Algorithm 3, mesh-size has been halved.

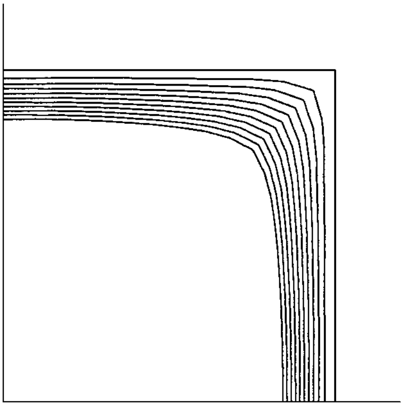

**FIG. 12.** Free boundary during the first 10 time-steps using Algorithm 4, mesh-size has been halved.

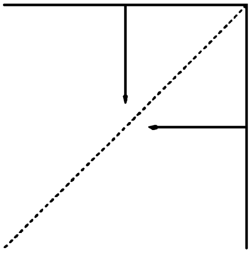

**FIG. 13.** Displacement of the mid-side points is too close to the line $x = y$.

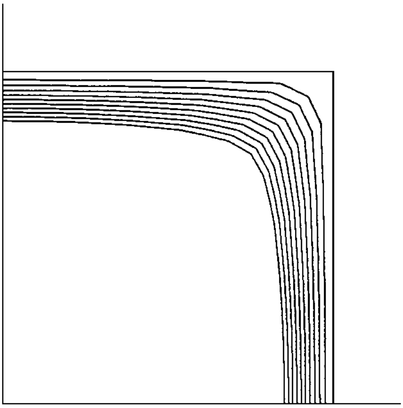

**FIG. 14.** Free boundary during the first 10 even time-steps using Algorithm 3; mesh-size and time-step have been halved.

points may even cross the line $x = y$. In that case both algorithms fail, since they produce a negative displacement for the vertex at $y = x$.

This implies that when the space-step is reduced, one also has to decrease the time-step. Figures 14 and 15 show the free boundaries for the finer mesh with the halved time-step. The boundaries are plotted at the same time-levels as the ones in Figs. 9 and 10. Further halving of the space- and time-steps leads to nice pictures for Algorithm 4. However, Algorithm 3 again produces in the first time-step a free boundary like the one depicted in Fig. 11.

## 5. REMARKS CONCERNING THE MESH GENERATION DURING TIME-STEPPING

At the start of the computation a mesh is generated using the initial boundary. In each time-step the boundary is updated according to one of the given algorithms. The free boundary itself consists of a number of curves (in the previous example 2 curves) and each curve at the free boundary is approximated by a spline. The number of nodes is not changed. The nodes along the splines are distributed in such a way that local refinement in the initial free boundary is kept. Hence, the relative distribution of nodes along the initial boundary is

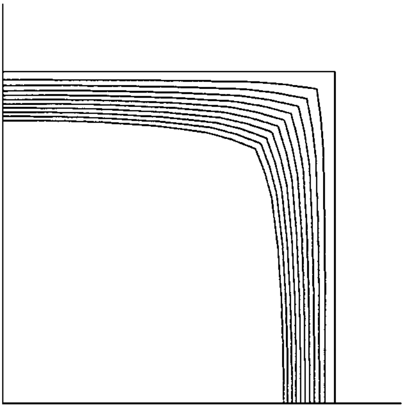

**FIG. 15.** Free boundary during the first 10 even time-steps using Algorithm 4; mesh-size and time-step have been halved.

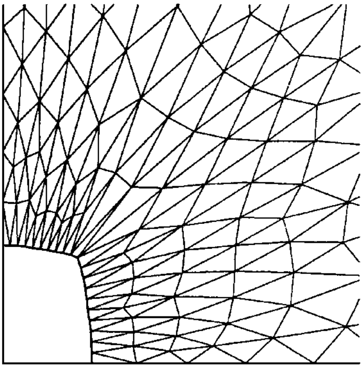

**FIG. 16.** Free boundary without re-meshing.

maintained during the whole process. Once the boundary is changed, the mesh is updated. In the first instance all points in the mesh are repositioned by taking the mean value of the coordinates of neighbouring nodes. This averaging process is performed in a number of steps in a Gauss–Seidel-like procedure.

When the mesh is created, the quality of the mesh and the distances at the free boundary are checked. If the angles of the triangles in the new mesh are too large, or if the distances between nodes at the free boundary differ too much from the original distances, the mesh is completely regenerated. In that case also the nodes at the free boundary are recomputed and the number of nodes at this boundary may be changed. After that, the just computed solution is interpolated to the new mesh.

If we do not re-mesh the elements in the neighbourhood of the free boundary become very distorted. Figure 16 shows a part of the mesh at $t = 60$ s, when no re-meshing is applied. Due to the large distortions, the derivatives of the concentration along the free boundary are very inaccurate and the free boundary has an unnatural shape. For that reason re-meshing has been applied. Figure 17 shows the same part of the mesh at $t = 60$ s as before, and it is clear that the boundary is much more realistic.

In Fig. 18 the position and velocity of the free boundary versus time is given. The points considered are the singular point and the intersection of the free boundary with a fixed

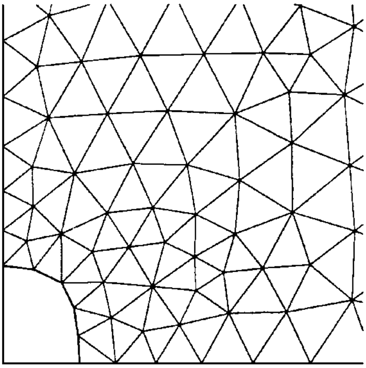

**FIG. 17.** Free boundary with re-meshing.

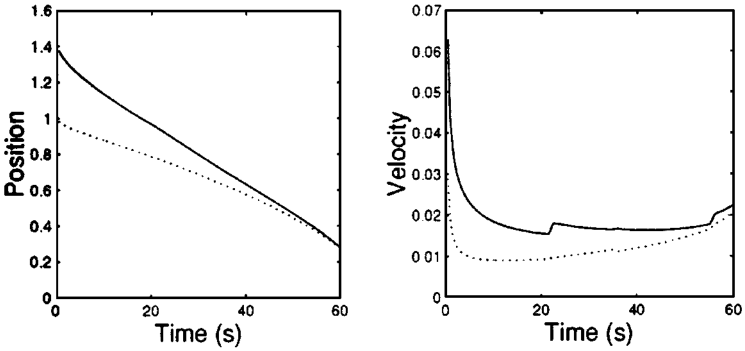

**FIG. 18.** Position and velocity of the singular point (———) and the intersection of the free boundary with a fixed boundary (. . .).

boundary. Re-meshing occurs at 22, 35.5, and 55.5 s. Note that at these times small bumps appear in the velocity graphs.

The interpolation used is not conservative and mass may be created or lost. Indeed the mass before and after interpolation (only used when re-meshing is applied) differs somewhat but in our computations it is less than 0.2%. A more accurate approach would be, for example, an $L^2$-projection. However, even in 1D such a projection is quite complex due to the fact that basis functions defined over elements in the new mesh are discontinuous in the old mesh. See, for example, [14] for a discussion on this subject.

Although our interpolation technique is not the most sophisticated one, still the previous results show that re-meshing is a necessary option once the mesh is largely deformed.

## 6. NUMERICAL EXPERIMENTS

An algorithm has been developed, suitable for use to investigate the dissolution kinetics for a two-dimensional case. This algorithm has been implemented in our finite element code SEPRAN [18]. One of the goals of the algorithm is to determine the dissolution kinetics of disk-like particles. Before two-dimensional problems are considered, we compare the outcomes of our algorithm to a one-dimensional axi-symmetric problem. Numerical experiments show that the differences between the results obtained from the one-dimensional finite difference method [23] and the two-dimensional finite element method are very small. Disk-like particles are treated using axi-symmetry as well. In all our examples we have chosen the following parameters:

| | |
|---|---|
| diffusion coefficient | $\mathbb{D} = 0.04858,$ |
| concentration in the particle | $c_{\text{part}} = 54,$ |
| concentration at the interface | $c_{\text{sol}} = 3.88,$ |
| initial concentration | $c_0 = 0.0011,$ |
| Stefan factor | $\frac{\mathbb{D}}{c_{\text{part}} - c_{\text{sol}}} = 0.000969.$ |

### 6.1. *Disk-like Problems*

The dissolution of disk-like particles is, in fact, a three-dimensional problem. Due to the axi-symmetry it can be solved as a two-dimensional problem. We suppose that a disk with radius 1 and disk length $D$ dissolves into a cell with radius 4 and cell length $L$.

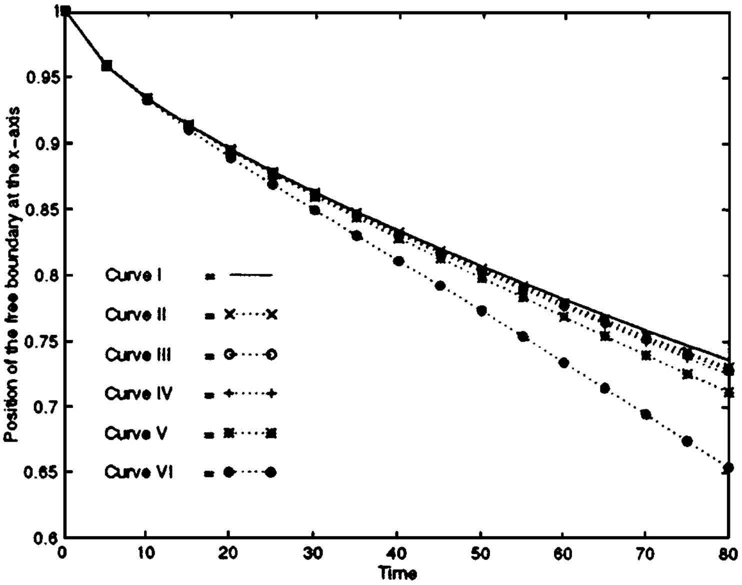

**FIG. 19.** Radius of cross section of the dissolving cylinder with the *OXY*-plane.

First, we consider the special case of $D = L$. The resulting problem is again one-dimensional. In Fig. 19 Curve I corresponds to the 1D axi-symmetric finite difference method (fdm). From a comparison of this method with an analytical method [24] the fdm method appears to be reliable. The results from the finite element method are shown as Curve II. Both results are nearly the same.

Second, we consider the position of the intersection of the free boundary and $\Gamma_4$ as a function of time for various disk-heights. Curves III, IV, V, and VI correspond to disk-lengths of $0.99L$, $0.95L$, $0.90L$, and $0.75L$, respectively. In the limit (Curves III and IV), the position approaches that of the one-dimensional case with axial symmetry. Note that for a small value of $D$ ($D = 0.75L$), the final position of the intersection is close to 0.65, whereas the final position in the one-dimensional problem is approximately equal to 0.75. So the behaviour of a disk-like particle is different from a cylindrical one, especially when the time increases.

Figure 20 gives the concentration $c$ at $r = 4$, $z = 0$, as a function of the disk length for different dissolution times. It can be seen in Fig. 20 that the influence of the disk length on the concentration at $r = 4$ increases with increasing dissolution time. This is in accordance with what one expects physically, because at early stages of the dissolution process the diffusion fields are small, so the shape of the particle hardly influences the concentration at the intersection of $\Gamma_3$ and $\Gamma_4$.

### 6.2. *Dissolution in a Bar*

In metallurgical literature only one-dimensional algorithms areused to investigate dissolution kinetics. This requires the use of an equally shaped cell in which the particle dissolves such that the cell volume equals the real volume around the particle. Since the two-dimensional algorithm has been developed, the error of the last mentioned approach may be analysed.

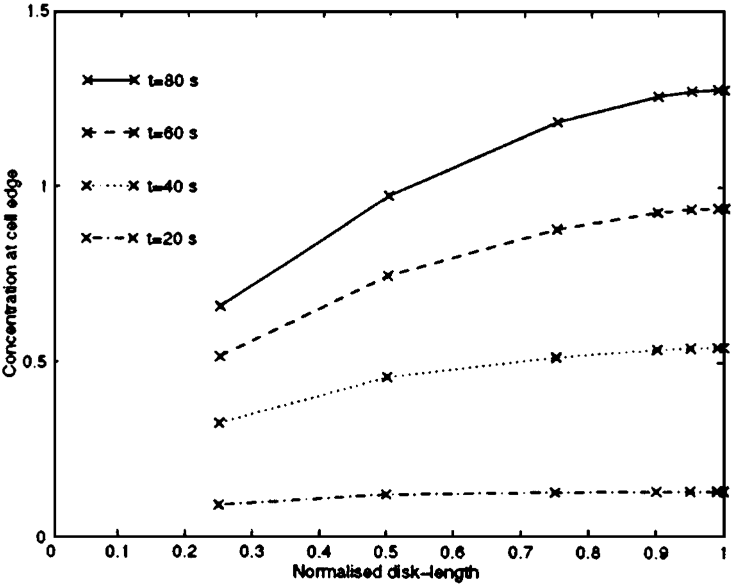

**FIG. 20.** Concentration at $r = 4$, $z = 0$ as function of the disk-length.

The dissolution kinetics of a cylinder, dissolving in a bar, has been compared to the dissolution of a cylinder dissolving in a cylinder with an equal volume. As can be expected for cases in which the cell size is large, the approach with an equally shaped cell is most accurate. The average distance between the free boundary and the origin does not differ much for both geometries of the cell. However, for the case of a bar-like cell, the movement of the free boundary near the intersections with the co-ordinate axes will be smaller than the movement of the free boundary near the intersection with the line y = x, causing a shape change during dissolution. These effects will be more pronounced as dissolution proceeds.

Reducing the cell size with respect to the particle size would reveal a larger difference between both approaches. However, such small cell sizes are not likely to occur in metallurgy, so the approach made in literature can be considered as reasonable.

The free boundary of a square dissolving in a square has been sketched at different stages of the dissolution process in Fig. 21. The edge of the particle has length 1; the edge of the cell has length 4. One sees that the shape of the free boundary becomes more rounded as dissolution proceeds. The shape of the free boundary even becomes almost circular at later

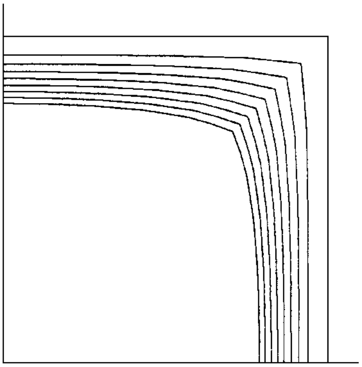

**FIG. 21.** Free boundary of a bar dissolving in a bar at various stages of the dissolution process.

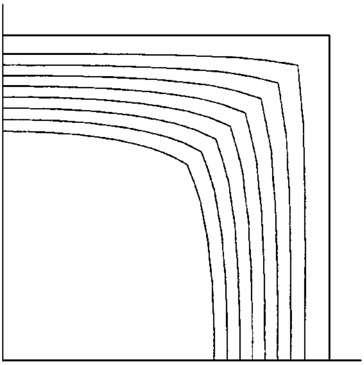

**FIG. 22.** Free boundary of a disk dissolving in a disk at various stages of the dissolution process.

stages of the dissolution process. The same has been done for the dissolution of a disk in a disk in Fig. 22. In this axi-symmetric case $D = 1, L = 4$, the particle radius is equal to 1 and the cell radius equal to 4. The same shape transition is observed. Moreover, it is observed that the displacement of the interface is larger at the $r$-axis (horizontal axis) than at the $z$-axis (vertical axis) (see Fig. 22).

### 6.3. *Two Particles Dissolving in One Cell*

In metallic systems the particle size is nonuniform. Most authors in literature incorporate a particle size distribution using the assumptions

- all particles dissolve in an equally shaped cell,
- the average concentration of alloying element is equal in each cell.

This implies a unique correspondence between the cell size and the particle size. Furthermore, most authors assume that there is no mass transfer between the cells. Tundal and Ryum [21] consider the approach that when the smallest particle is dissolved completely, the cells related to the residual particles are enlarged by them such that the volume of the cells equal the sum of the volumes of the original cells.

In our two-dimensional numerical method, these assumptions can be released. Therefore we compute the dissolution of two particles in one cell. In this problem the length of the edge of the square is equal to 5, whereas the radii of the particles are equal to 2 and 0.5, respectively. The movement of the free boundaries is visualized in Fig. 23.

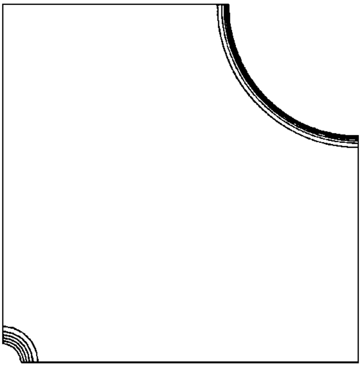

**FIG. 23.** The movement of two circular free boundaries in a cell in which two particles dissolve.

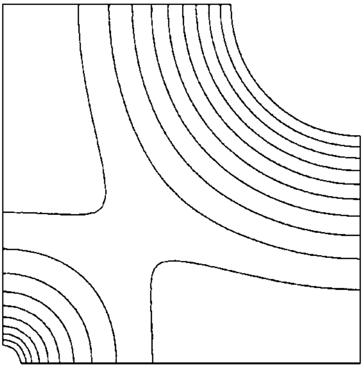

**FIG. 24.** Iso-concentration lines in a cell in which two particles dissolve.

As soft-impingement occurs, i.e. the interaction of the diffusion fields around the particles, movement of the free boundary is influenced by the presence of different particles. Iso-concentration lines for this case have been sketched in Fig. 24. It is a straightforward exercise to extend these calculations to a system with more cells or particles.

## 7. CONCLUSIONS

Particle dissolution in binary alloys is investigated by numerical techniques. The mathematical model used is that of a free boundary problem of Stefan type. This problem is solved by a two-dimensional finite element method. It has been shown that this approach leads to an accurate solution of the problem.

With respect to the adaptation of the free boundary during time-stepping it has been demonstrated that sharp corners require a special algorithm. Several algorithms have been developed. From these algorithms, the nonlinear approach, based on the discretization of the integral balance between dissolution and diffusion, has proven to be superior.

The finite element method applied is based upon a displacement of all nodes. The free boundary is approximated by a spline and the nodes are redistributed in order to maintain the original coarseness of the nodes. Re-meshing is applied if necessary.

It has been shown that for some types of particles two-dimensional effects cannot be neglected. For those cases it is not sufficient to run a one-dimensional code and our 2D finite element method gives a very attractive alternative.

## REFERENCES

1. H. B. Aaron and G. R. Kotler, Second phase dissolution, *Metall. Trans.* **2**, 393 (1971).
2. M. J. Baines, *Moving Finite Elements*, Monographs on numerical analysis (Clarendon, Oxford, 1994).
3. U. L. Baty, R. A. Tanzilli, and R. W. Heckel, Dissolution kinetics of $CuAl_2$ in an $Al_{-4}Cu$ alloy, *Metall. Trans.* **1**, 1651 (1970).
4. J. Crank, *Free and Moving Boundary Problems* (Clarendon Press, Oxford, 1984).
5. A. J. Dalhuijsen and A. Segal, Comparison of finite element techniques for solidification problems, *Int. J. Num. Engng.* **23**, 1807 (1986).
6. J. Donea, S. Giuliani, and J. P. Halleux, An arbitrary Lagrangian-Eulerian finite element method for transient dynamic fluid-structure interactions, *Comp. Meth. Appl. Mech. Engng.* **33**, 698 (1982).
7. C. W. Hirt, A. A. Amsden, and J. L. Cook, An arbitrary Lagrangian-Eulerian computing method for all flow speeds, *J. Comp. Phys.* **14**, 227 (1974).

8. A. Huerta and W. K. Liu, Viscous flow with large free surface motion, *Comp. Meth. Appl. Mech. Engng.* **69**, 277 (1988).

9. T. J. R. Hughes, W. K. Liu, and T. K. Zimmerman, Lagrangian-Eulerian finite element formulation for incompressible viscous flows, *Comp. Meth. Appl. Mech. Engng.* **29**, 329 (1981).

10. D. R. Lynch, Unified approach to simulation on deforming elements with application to phase change, *J. Comp. Phys.* **47**, 387 (1982).

11. D. R. Lynch and K. O'Neill, Continuously deforming finite elements for the solution of parabolic problems, with and without phase change, *Int. J. Num. Meth. Engng.* **17**, 81 (1981).

12. W. D. Murray and F. Landis, Numerical and machine solutions of transient heat-conduction problems involving melting or freezing, *Trans. ASME (C) J. Heat Transfer* **81**, 106 (1959).

13. F. V. Nolfi Jr., P. G. Shewmon, and J. S. Foster, The dissolution and growth kinetics of spherical precipitates, *Transactions of the Metallurgical Society of AIME* **245**, 1427 (1969).

14. A. Oliveira and A. M. Baptista, A comparison of integration and interpolation Eulerian-Lagrangian methods, *Int. J. Num. Meth. Fluids* **21**, 183 (1995).

15. B. Ramaswamy and M. Kawahara, Arbitrary Lagrangian-Eulerian finite element method for unsteady, convective, incompressible viscous free surface flow, *Int. J. Num. Meth. Fluids* **7**, 1053 (1987).

16. A. A. Samarskii, P. N. Vabishchevich, O. P. Iliev, and A. G. Churbanov, Numerical simulation of convection/diffusion phase change problems—a review, *Int. J. Heat Mass Transfer* **36**, 4095 (1993).

17. V. R. B. Santos, A direct method for solving two-dimensional one phase Stefan problems, *Comp. Meth. Appl. Mech. Engng.* **25**, 51 (1981).

18. G. Segal, *SEPRAN manuals* (Sepra Leidschendam, 1984).

19. J. Stefan, Über die Theorie der Eisbildung, insbesondere über die Eisbildung im Polarmeere, *Annalen der Physik und Chemie* **42**, 269 (1891).

20. D. A. Tarzia, "A bibliography on moving-free boundary problems for the heat-diffusion equation. The Stefan problem." Technical report, Istituto Matematico "Ulisse Dini," University of Florence, Florence, 1988.

21. U. H. Tundal and N. Ryum, Dissolution of particles in binary alloys: Part I. Computer simulations, *Metall. Trans.* **23A**, 433 (1992).

22. F. Vermolen, K. Vuik, and S. van der Zwaag, A numerical analysis for the dissolution of second phase particles in ternary alloys, in *Moving Boundaries IV, Computational Modelling of Free and Moving Boundary Problems, Proceedings of the Fourth International Conference on Moving Boundaries, Gent, August 1997*, edited by R. van Keer and C. A. Brebbia, Computational Mechanics Publications, Southampton, 1997, p. 153.

23. F. J. Vermolen and S. Van der Zwaag, A numerical model for the dissolution of spherical particles in binary alloys under mixed mode control, *Materials Science and Engineering A* **220**, 140 (1996).

24. F. J. Vermolen, P. van Mourik, and S. Van der Zwaag, Analytical approach to particle dissolution in a finite medium, *Materials Science and Technology* **13**, 308 (1997).

25. C. Vuik, *The Solution of a One-Dimensional Stefan Problem*, CWI-tract 90 (CWI, Amsterdam, 1993).

26. C. Vuik, Some historical notes about the Stefan problem, *Nieuw Archief voor Wiskunde, 4e serie* **11**, 157 (1993).

27. C. Vuik and C. Cuvelier, Numerical solution of an etching problem, *J. Comp. Phys.* **59**, 247 (1985).

28. M. J. Whelan, On the kinetics of particle dissolution, *Metal Science Journal* **3**, 95 (1969).