Domain Transformation Problems in 2-D Oxidation

Pavlin B. Entchev†, Dimitris C. Lagoudas†
and
Oleg P. Iliev‡,

†Center for Mechanics of Composites
Department of Aerospace Engineering
Texas A&M University
College Station, TX 77843-3141

‡Institute of Mathematics and Informatics
Bulgarian Academy of Sciences
Sofia 1113, Bulgaria

Abstract

The oxidation process in metals governed by oxygen diffusion is modeled in this paper for 2-D domains with sharp corners. The emphasis of the paper is on discussing efficient numerical algorithms for simulating two dimensional oxide layer formation at an internal wedge tip. Two numerical algorithms for computing oxide layer growth are presented. The first algorithm is based on transforming the computational domain into a rectangular one, and then applying a finite difference scheme for the transformed equations. The second algorithm is based on a finite element discretization of the original domain with locally refined grid near the wedge surface. Results from the numerical simulations are presented and discussed.

1 Introduction

Light metallic alloys are commonly used to fabricate many structural components for aerospace applications. However, at elevated temperatures, oxidation significantly reduces their mechanical properties. Therefore, a number of researchers have investigated the oxidation kinetics of various metallic alloy systems (see [1] for a complete summary for titanium (Ti) oxidation) in order to both improve their mechanical properties, as well as increase their oxidation resistance. Since eliminating oxidation does not appear to be a viable solution, being able to accurately model the effect of oxidation on the structural integrity of metallic components is an important task. Therefore, the present work focuses on the modeling of oxidation in metals and in particular in titanium.
Understanding the oxidation behavior of metals requires a detailed knowledge of oxidation rates and their temperature dependence. Researchers in references [2] and [3] have specifically investigated the reaction rate of titanium oxidation and its dependence on temperature. Reference [3] classifies the most common oxidation rates to be of three different types: logarithmic, parabolic and linear. The logarithmic oxidation rate for titanium is generally observable at lower temperatures (below 400°C) and corresponds to an initially rapid reaction which then drops off to very small values. At higher temperature (600°C-800°C) titanium has been observed to follow a parabolic oxidation rate, with a transition region between 400°C and 600°C [4, 5]. This is followed by a shift to a linear oxidation rate in the temperature range of 800°C-1000°C, with the parabolic-to-linear transition being a function both of temperature, as well as oxidation time.

Since most high temperature aerospace structural applications are envisioned for use in temperatures between 600°C and 800°C, the work considered herein will be restricted to modeling oxidation using the parabolic law.

In addition to the oxidation rate, one must also consider the kinetics involved during parabolic oxidation. Researchers in references [6], [7] and [8] have studied the oxidation mechanisms of titanium and its alloys when undergoing parabolic scale growth and have concluded that the parabolic oxidation rate is controlled by diffusional processes. These processes include either an inward diffusion of oxygen or simultaneous inward diffusion of oxygen and outward diffusion of titanium. The works [6] and [7] indicate that at initial stages of oxidation it is possible for titanium to diffuse through the oxide layer. However, the physical nature of the oxide layers formed on pure Ti in an oxygen atmosphere has been examined in [8] and it has been concluded that the oxidation process in Ti depends on the reaction between oxygen and titanium occurring at the interface; i.e., it is oxygen and not titanium which diffuses through the layer. Thus, although the oxidation process of Ti involves simultaneous diffusion of both Ti and oxygen, the oxygen diffusion dominates the process.

Based on the above discussion, a diffusion controlled oxidation model [5] has been considered in this paper. This model accounts for inward diffusion of oxygen, as well as for parabolic oxidation rate and it has been widely used by researchers (see, for example, [9, 10, 11]) to solve problems in 1D and simple 2D geometries. On the other hand, its application to complex multi-dimensional cases, such as oxidation of metal matrix composites, edges and cracks, is computationally complicated. The complication occurs because the solution involves a moving metal-oxide interface, whose position and its evolution with time depend on the solution itself. As a consequence, special numerical techniques must be considered. The present work aims at presenting an algorithm for numerical simulation of high-temperature oxidation in complex 2D domains, and, in particular, of internal wedges (see Fig. 1). The selection of internal wedges as the preferred geometrical shape is due to their importance in representing a wide range of material defects including cracks as the limiting case.

There is a variety of methods for solving problems with moving interfaces available in the literature (see [12] for a complete review). Of all the various methods, three are considered herein for possible implementation. These include: interface tracking, domain transformation and interface smearing methods. An overview of the Interface Tracking Method is given in the reference [13]. In addition, Glimm and his co-workers [14, 15] have performed extensive research on this method. The second widely used method is the Domain Transformation
The essence of this method is mapping of each subdomain into a fixed domain. Several variations of this method have appeared in the literature: transformation of spatial variables only has been used in [16], while both transformation of spatial variables, as well as transformation of the time variable has been used in [11]. Even though the domain transformation methods provide acceptable results for 1D problems (for example, researchers in reference [11] have applied this technique for solving the oxidation problem in both plane and axisymmetric cylindrical geometries, as well as in a domain with spherical geometry), their implementation in 2D for more complex domains is rather complicated and in some cases impossible.

The method chosen for implementation in this work is The Interface Smearing Method [17]. The method is based on reformulating the diffusion equations for both the oxide and the metal, resulting in a single non-linear equation for the whole domain. The numerical solution of the resulting equation involves the application of either the Finite Differences Method (FDM) or the Finite Element Method (FEM). However, the application of FDM in complex domains is computationally difficult. In order to reduce complexities associated with internal wedge domains, the authors have chosen to map the non-regular computational domain wedge into a reference rectangular domain and apply the FDM in the reference domain. In addition, direct application of the FEM is implemented. The purpose of the paper is not to compare properties of the two approaches, but, rather, to demonstrate different procedures for performing numerical simulations of oxide layer formation on complex geometries.

The remainder of the paper is organized as follows. The mathematical model as well as
the used smearing technique are briefly described in the second section. The third section is devoted to the numerical solution of the oxidation problem. A subsection devoted to the transformation of the original domain is also included in the third section. Results from numerical experiments are presented and discussed in the fourth section. Finally, some conclusions are presented.

2 Two-Dimensional Diffusion Model of Oxidation

Let \( \Omega \subset \mathbb{R}^2 \) be a fixed domain with \( \partial \Omega \) as its boundary. \( \Omega \) is partitioned by the interface curve \( \tau (t) \) into two subdomains \( \Omega_1 \) and \( \Omega_2 \), respectively, such that \( \Omega = \Omega_1 \cup \Omega_2 \), \( \tau = \Omega_1 \cap \Omega_2 \), and \( \partial \Omega = \partial \Omega_1 \cup \partial \Omega_2 \) (see Fig. 2). The subdomain \( \Omega_1 \) corresponds to the oxide and \( \Omega_2 \) to the unoxidized metal.

![Two-Dimensional domain with interface \( \tau(t) \) between the oxide region \( \Omega_1 \) and the unoxidized metal region \( \Omega_2 \).](image)

The mass concentration of oxygen in \( \Omega_1 \) and \( \Omega_2 \) is assumed to satisfy the following diffusion equations [5]:

\[
\frac{\partial C_1 (x, y, t)}{\partial t} = \text{div}(D_1 \text{grad}C_1), \quad (x, y) \in \Omega_1, \tag{1}
\]

\[
\frac{\partial C_2 (x, y, t)}{\partial t} = \text{div}(D_2 \text{grad}C_2), \quad (x, y) \in \Omega_2. \tag{2}
\]

Here \( C_1 (x, y, t) \) and \( C_2 (x, y, t) \) denote mass concentrations of oxygen in \( \Omega_1 \) and \( \Omega_2 \), respectively and \( D_1 \) and \( D_2 \) are the diffusion coefficients of two phases that occupy \( \Omega_1 \) and \( \Omega_2 \), respectively. It is assumed that at the beginning of the process \( \Omega_1 = \emptyset \), i.e., the whole domain is occupied by the metal phase.
The interface \( \tau (t) \) that partitions \( \Omega \) into two regions is the phase boundary, separating the oxidized part from the metallic part, where oxidation has not taken place. Assuming that the chemical reaction occurs in a short time scale compared with the diffusion process, oxidation instantaneously takes place as soon as the concentration of oxygen reaches a critical value, and the interface moves as a function of time, always satisfying the critical concentration requirement. The appropriate interface conditions, expressing conservation of mass across the interface and initiation of the oxidation process whenever a critical oxygen concentration is reached, are given by

\[
- D_1 (\nabla C_1) \cdot \mathbf{N} = - D_2 (\nabla C_2) \cdot \mathbf{N} + [C] V,
\]

(3)

\[
C_1 (x, y, t) = C_{cr}, \quad C_2 (x, y, t) = C_{cr} - [C], \quad (x, y) \in \tau (t)
\]

(4)

where \( \mathbf{N} \) is the unit normal vector on \( \tau (t) \) pointed outward \( \Omega_2 \); \( V \) is the normal velocity of the phase change interface; \( C_{cr} \) is the critical mass concentration for the oxidation; and, \([C]\) is the concentration jump accounting for the immobilized oxygen.

Accounting for the fact that initially the whole domain is occupied by the metal phase, the initial concentration of oxygen is set to zero. The boundary conditions, specified on \( \partial \Omega \), depend on the specific problem and will be described in a sequel.

Using the Interface Smearing Method approach presented in Ref.[18], the following governing equation is obtained from equations (1) and (2):

\[
\tilde{G}(C) \frac{\partial C}{\partial t} = \text{div}(\tilde{D}(C) \text{grad} C),
\]

(5)

where

\[
\tilde{G}(C) = 1 + [C] \tilde{\delta} (C - C_{cr}, \Delta),
\]

(6)

\[
C = \begin{cases} 
C_1, & \text{if } (x, y) \in \Omega_1, \\
C_2 + [C], & \text{if } (x, y) \in \Omega_2,
\end{cases}
\]

(7)

\( \tilde{\delta} (C - C_{cr}, \Delta) \) is the smoothed Dirac \( \delta \)-function (\( \Delta \) is the extent of a semi-interval for concentration \( C \) in which \( \tilde{\delta} (C - C_{cr}, \Delta) \neq 0 \)); and \( \tilde{D} \) is the smoothed diffusivity coefficient, i.e.,

\[
\tilde{D}(C) = \begin{cases} 
\frac{D_1 - D_2}{2\Delta} C + \frac{D_1}{2\Delta}, & \text{if } C > C_{22}, \\
\frac{D_2 C_{22} - D_1 C_{11}}{2\Delta}, & \text{if } C_{11} \leq C \leq C_{22}, \\
D_2, & \text{if } C < C_{11},
\end{cases}
\]

(8)

where

\[
C_{11} = C_{cr} - \Delta, \quad C_{22} = C_{cr} + \Delta.
\]

(9)

Thus equation (5) with appropriate initial and boundary conditions fully determines the solution of the oxidation problem in the domain \( \Omega = \Omega_1 \cup \Omega_2 \).

3 Numerical Solution of the Oxidation Problem

3.1 Finite Difference Approach

Finite Differences Method is known to be very powerful for numerical solution of PDEs; but, usually its performance is superior only in rectangular domains. Therefore, special
techniques for transforming the internal wedge computational domain into a rectangular one should be considered. One of the popular methods for such a transformation – fictitious regions method – is presented in [18]. This method, while easy to implement, is limited to specifying only Dirichlet boundary condition on the part of the boundary to be transformed. In this work, another method – Landau transformation – is used for transforming the non-regular domain. The presented method allows specification of both Dirichlet and Neumann boundary conditions on the boundary.

### 3.1.1 Domain Transformation

Let us consider a non-rectangular domain \( \hat{\Omega} \) (see Fig.3) and the transformation \( \mathcal{X} \) [19] which transforms \( \hat{\Omega} \) into a rectangular domain \( \Omega^r \) with the boundary \( \Gamma^r \), where

\[
\Omega^r \cup \Gamma^r = \{ (\zeta, \eta) : 0 \leq \zeta \leq a, 0 \leq \eta \leq b \}. \tag{10}
\]

![Figure 3: Schematic representation of the non-rectangular domain \( \hat{\Omega} \): (a) suitable for transformation; (b) unsuitable for transformation.](image)

The transformation \( \mathcal{X} \) itself expands the domain \( \hat{\Omega} \) to a rectangular one and has the following form:

\[
\mathcal{X} : \begin{cases} 
\zeta = x \\
\eta = \psi(x, y) 
\end{cases}, \tag{11}
\]

where

\[
\psi(x, y) = b \frac{y - \varphi(x)}{b - \varphi(x)}, \tag{12}
\]

and function \( \varphi(x) \) describes the lower boundary of the domain \( \hat{\Omega} \), as shown in Fig.3. To be possible to perform the above described transformation the function \( \varphi(x) \) should be a single valued function (see Fig.3a). In the case that \( \varphi(x) \) is not a single valued function (see Fig.3b) it is not possible to apply a transformation described by equations (11) and (12).
In this case one can subsequently perform several transformations and still transform the domain $\hat{\Omega}$ into $\Omega^r$. However, this case is beyond the scope of the present work.

To rewrite the governing equation in the new variables, we will consequently substitute in (5) the new variables from (12). The derivative of the function $\psi$ with respect to $x$ is:

$$\frac{\partial \psi}{\partial x} = \varphi'(\zeta) \frac{\eta - b}{b - \varphi(\zeta)}. \tag{13}$$

It is necessary to find the partial derivatives of $C$ in terms of $\zeta$ and $\eta$. Using (11) and (13) we have:

$$\begin{align*}
\frac{\partial C}{\partial x} &= \frac{\partial C}{\partial \zeta} \frac{\partial \zeta}{\partial x} + \frac{\partial C}{\partial \eta} \frac{\partial \eta}{\partial x} = \frac{\partial C}{\partial \zeta} + \frac{\varphi'(\zeta)}{b - \varphi(\zeta)} \frac{\eta - b}{\partial \zeta} \frac{\partial C}{\partial \eta}, \\
\frac{\partial C}{\partial y} &= \frac{\partial C}{\partial \zeta} \frac{\partial \zeta}{\partial y} + \frac{\partial C}{\partial \eta} \frac{\partial \eta}{\partial y} = \frac{b}{b - \varphi(\zeta)} \frac{\partial C}{\partial \eta}. \tag{14}
\end{align*}$$

In a similar manner we obtain the second-order partial derivatives of $C$. Using (14) we have:

$$\begin{align*}
\frac{\partial}{\partial x} \left( \tilde{D} \frac{\partial C}{\partial x} \right) &= \frac{\partial}{\partial \zeta} \left( \tilde{D} \frac{\partial C}{\partial \zeta} \right) \frac{\partial \zeta}{\partial x} + \frac{\partial}{\partial \eta} \left( \tilde{D} \frac{\partial C}{\partial \zeta} \right) \frac{\partial \eta}{\partial x} \\
&= \frac{\varphi'(\zeta)}{b - \varphi(\zeta)} \frac{\eta - b}{\partial \zeta} \frac{\partial C}{\partial \eta} + \left( \frac{\varphi'(\zeta)}{b - \varphi(\zeta)} \right)^2 \frac{\partial}{\partial \eta} \left( \tilde{D} \frac{\partial C}{\partial \eta} \right) \\
&+ 2 \varphi'(\zeta) \frac{\eta - b}{(b - \varphi(\zeta))^2} \tilde{D} \frac{\partial C}{\partial \zeta} + \frac{\partial}{\partial \zeta} \left( \tilde{D} \frac{\partial C}{\partial \eta} \right) \\
&+ \varphi'(\zeta) \frac{\eta - b}{b - \varphi(\zeta)} \frac{\partial}{\partial \zeta} \left( \tilde{D} \frac{\partial C}{\partial \eta} \right) + \frac{\varphi''(\zeta)}{b - \varphi(\zeta)} \frac{\partial \eta}{\partial \zeta} \frac{\partial C}{\partial \eta}. \tag{16}
\end{align*}$$

Similarly, using (15) we have:

$$\frac{\partial}{\partial y} \left( \tilde{D} \frac{\partial C}{\partial y} \right) = \left( \frac{b}{b - \varphi(\zeta)} \right)^2 \frac{\partial}{\partial \eta} \left( \tilde{D} \frac{\partial C}{\partial \eta} \right). \tag{17}$$

Thus, after substitution from (16) and (17), the governing equation (5) can be written as:

$$\tilde{G} \frac{\partial C(\zeta, \eta, t)}{\partial t} = \frac{\partial}{\partial \zeta} \left( \tilde{D} \frac{\partial C}{\partial \zeta} \right) + \varphi'(\zeta) \frac{\eta - b}{b - \varphi(\zeta)} \frac{\partial}{\partial \zeta} \left( \tilde{D} \frac{\partial C}{\partial \eta} \right) + \frac{\varphi'(\zeta)}{b - \varphi(\zeta)} \frac{\partial}{\partial \zeta} \left( \tilde{D} \frac{\partial C}{\partial \eta} \right) + \left( \frac{\varphi'(\zeta)}{b - \varphi(\zeta)} \right)^2 \frac{\partial}{\partial \eta} \left( \tilde{D} \frac{\partial C}{\partial \eta} \right) + \left[ \left( \frac{\varphi'(\zeta)}{b - \varphi(\zeta)} \right)^2 + \left( \frac{b}{b - \varphi(\zeta)} \right)^2 \right] \frac{\partial}{\partial \zeta} \left( \tilde{D} \frac{\partial C}{\partial \eta} \right) + \frac{\partial}{\partial \zeta} \left( \tilde{D} \frac{\partial C}{\partial \eta} \right) + \frac{\varphi''(\zeta)}{b - \varphi(\zeta)} \frac{\partial \eta}{\partial \zeta} \frac{\partial C}{\partial \eta}. \tag{18}$$

The above derived equation (18) with the appropriate boundary conditions fully describes the oxidation problem in the transformed domain $\Omega^r$. It should be mentioned that the expressions for the boundary conditions containing derivatives should be also transformed in a similar manner.
3.1.2 Transformation of an Internal Wedge Domain

Now to apply the above transformation we consider a domain corresponding to an internal wedge (see Fig.4). Due to symmetry in the geometry with respect to the $x$ axis the domain under consideration is only one-half of the actual domain. To set the boundary conditions we consider the computational domain to be chosen as a small part of the actual domain, as shown in Fig.5. Then, accounting for the symmetry, we set:

$$\frac{\partial C}{\partial y} = 0 \quad \text{on} \quad \Gamma_1.$$  \hspace{1cm} (19)

On boundary $\Gamma_2$ we set the oxygen concentration to be equal to a given value, i.e.,

$$C = C_0 \quad \text{on} \quad \Gamma_2.$$  \hspace{1cm} (20)

Assuming that far from the wedge tip the oxidation is close to the 1D case, we set the following boundary condition on $\Gamma_3$:

$$(\nabla C) \cdot n_1 = 0 \quad \text{on} \quad \Gamma_3,$$  \hspace{1cm} (21)

where $n_1$ denotes the normal vector on a line perpendicular to $\Gamma_2$ (see Fig.4). In addition, it is assumed that the boundaries $\Gamma_4$ and $\Gamma_5$ are far from the exposed boundary and are not affected by the oxidation. Therefore, we set:

$$C = [C] \quad \text{on} \quad \Gamma_4.$$  \hspace{1cm} (22)
Having specified the boundary conditions, we now consider the application of the domain transformation, described earlier, to the specific internal wedge domain. For this particular case the function \( \varphi(x) \) (\( \varphi(x) = \Gamma_1 \cup \Gamma_2 \)) can be written in the following form:

\[
\varphi(x) = \alpha x + \beta,
\]

where

\[
\alpha = \begin{cases}
0 & , 0 \leq x \leq a_1 \\
\frac{b_1}{a - a_1} & , a_1 < x \leq a 
\end{cases}
\]

\[
\beta = \begin{cases}
0 & , 0 \leq x \leq a_1 \\
-\frac{a_1 b_1}{a - a_1} & , a_1 < x \leq a 
\end{cases}
\]

Substituting this expressions into equation (18) and taking all the necessary derivatives of \( \varphi(x) \) we obtain:

\[
\frac{\partial \tilde{C}}{\partial t} = \frac{\partial}{\partial \zeta} \left( \tilde{D} \frac{\partial C}{\partial \zeta} \right) + \alpha \frac{\eta - b}{b - \varphi(\zeta)} \frac{\partial}{\partial \zeta} \left( \tilde{D} \frac{\partial C}{\partial \eta} \right) + \alpha \frac{\eta - b}{b - \varphi(\zeta)} \frac{\partial}{\partial \eta} \left( \tilde{D} \frac{\partial C}{\partial \zeta} \right) + 2 \alpha^2 \frac{\eta - b}{(b - \varphi(\zeta))^2} \tilde{D} \frac{\partial C}{\partial \eta} + \left( \alpha^2 \left( \frac{\eta - b}{b - \varphi(\zeta)} \right)^2 + \left( \frac{b}{b - \varphi(\zeta)} \right)^2 \right) \frac{\partial}{\partial \eta} \left( \tilde{D} \frac{\partial C}{\partial \eta} \right).
\]

Figure 5: Computational domain as a part of the actual domain.

\[
C = [C] \quad \text{on} \quad \Gamma_5.
\]

\[ (23) \]

\[ (24) \]

\[ (25) \]

\[ (26) \]
It should be mentioned that in this particular case the derivative of the function \( \varphi(x) \) is discontinuous at \( x = a_1 \).

### 3.1.3 Finite Difference Discretization

A uniform spatial grid is introduced in \( \Omega^r \): \( (\omega_h = (\zeta, \eta_j)|\zeta_i = i \cdot h_{\zeta}, \ i = 0, 1, \cdots, N_\zeta, \ \eta_j = j \cdot h_{\eta}, \ j = 0, 1, \cdots, N_\eta) \). The spatial grid is chosen so that the point \( x = a_1 \) is confined between two points of the grid. Constant time step \( \Delta t \) is used throughout the computations. In establishing a finite difference scheme for equation (27), the approach from [20] is used.

The approximation of derivatives is made using a finite volumes approach. In particular, the approximation of a mixed derivative reads:

\[
\frac{\partial}{\partial \zeta} \left( D \frac{\partial C}{\partial \eta} \right) \approx \frac{1}{2} \left( a_{i,j-1/2} C_{\eta}^h \right) \zeta + \frac{1}{2} \left( a_{i,j+1/2} C_{\eta}^h \right) \zeta + O \left( |h|^2 \right). \tag{28}
\]

In the above equation, a subscript indicates the usual forward differentiation, for example, \( C_{\eta} = \frac{C_{i,j+1} - C_{i,j}}{h_{\eta}} \) and a subscript with a bar denotes the backward differentiation, for example, \( C_{\eta}^b = \frac{C_{i,j} - C_{i,j-1}}{h_{\eta}} \). Special care is taken in determining the coefficients \( a_{i,j+1/2} \). They are chosen to be in the following form:

\[
a_{i,j+1/2} = \frac{\tilde{D}_{i,j} + \tilde{D}_{i,j+1}}{2}. \tag{29}
\]

Treating other terms in a similar way, we obtain the following finite difference scheme, approximating (27):

\[
\tilde{G}(C^{k+1}) \frac{C_{i,j}^{k+1} - C_{i,j}^k}{\Delta t} = (a_{i,j} C_{\zeta}^k) \zeta \\
+ \alpha \frac{\eta_j - b}{b - \varphi(\zeta_i)} \left( \frac{1}{2} \left( a_{i,j-1/2} C_{\eta}^h \right) \zeta + \frac{1}{2} \left( a_{i,j+1/2} C_{\eta}^h \right) \zeta \right) \\
+ \alpha \frac{\eta_j - b}{b - \varphi(\zeta_i)} \left( \frac{1}{2} \left( a_{i-1/2,j} C_{\zeta}^h \right) \eta + \frac{1}{2} \left( a_{i+1/2,j} C_{\zeta}^h \right) \eta \right) \\
+ 2\alpha^2 \frac{\eta_j - b}{(b - \varphi(\zeta_i))^2} a_{i,j} C_{\eta}^h \\
+ \left( \alpha^2 \left( \frac{\eta_j - b}{b - \varphi(\zeta_i)} \right)^2 + \left( \frac{b}{b - \varphi(\zeta_i)} \right)^2 \right) (a_{i,j} C_{\eta}^h) \eta, \tag{29}
\]

where \( C_{i,j}^k \) denotes \( C_{i,j}^k = C(\zeta_i, \eta_j, t^k) \). We can write the above equation (29) in matrix form as:

\[
AC^{k+1} = b. \tag{30}
\]

Because both the matrix \( A \) and the vector \( b \) depend on the solution, an iterative process is used in solving (30). For the iterative process the value of \( C^{k+1} \) from the previous iteration is used; i.e., \( (s)^{k+1} C \) is used for determining \( (s+1)^{k+1} C \), whereby \( s \) denotes the iteration number. For solving the non-symmetric set of linear algebraic equations at each iteration, ORTHOMIN, which is a variant of the Preconditioned Conjugate Gradient (PCG) method, is used [21].
3.2 Finite Element Approach

One of the strongest advantages of the Finite Element Method over Finite Differences Method is its ability to handle non-regular domains. Thus the formulation of FEM is presented in the original (untransformed) domain Ω. Multiplying equation (5) by test function ϕ and integrating over the domain Ω for a fixed time t we obtain:

\[ \int_{\Omega} \tilde{G}(C) \frac{\partial C}{\partial t} \phi dA = \int_{\Omega} \nabla \cdot (\tilde{D}(C) \nabla C) \phi dA. \] (31)

After applying the Green’s formula to the right hand side term, the above equation (31) becomes:

\[ \int_{\Omega} \tilde{G} \frac{\partial C}{\partial t} \phi dA + \int_{\Omega} \tilde{D} \nabla C \cdot \nabla \phi dA - \int_{\partial \Omega} \tilde{D} \frac{\partial C}{\partial n} \phi ds = 0. \] (32)

Using the standard FEM approach, the oxygen concentration is approximated as

\[ C(x, y, t) = \sum_{i=1}^{N} C_i(t) \phi_i(x, y), \] (33)

where \( C_i(t) \) are the nodal values of oxygen concentration and \( \phi_i(x, y) \) are the shape functions. Substituting equation (33) into (32) and enforcing equation (32) for all test functions \( \phi_j \), we obtain:

\[ \mathbf{M} \frac{dC}{dt} + \mathbf{K} C = 0, \] (34)

where \( \mathbf{K} \) and \( \mathbf{M} \) are corresponding “stiffness” and “mass” matrices with entries

\[ M_{ij} = \int_{\Omega} \tilde{G} \phi_j \phi_i dA, \] (35)

\[ K_{ij} = \int_{\Omega} \tilde{D} \nabla \phi_j \cdot \nabla \phi_i dA. \] (36)

Using the backward Euler method for the time derivative in the semi-discrete problem given by (34), we obtain

\[ \mathbf{M} \frac{C^k - C^{k-1}}{\Delta t} + \mathbf{K} C^k = 0, \] (37)

where \( C^k \) and \( C^{k-1} \) are the evaluations of \( C \) at the \( k^{th} \) and \( (k - 1)^{st} \) time steps and \( \Delta t \) is the time increment. For a given \( C^{k-1} \), equation (37) results in the following system of equations for the unknown \( C^k \):

\[ \left( \frac{\mathbf{M}}{\Delta t} + \mathbf{K} \right) C^k = \frac{\mathbf{M}}{\Delta t} C^{k-1}. \] (38)

As in the previous case (cf. equation (30)), matrices \( \mathbf{M} \) and \( \mathbf{K} \) include the non-linear coefficients \( \tilde{G}(C) \) and \( \tilde{D}(C) \). Therefore, the same iterative process as in Section 3.1.3 is used to solve (38).
4 Numerical Results and Discussion

Numerical calculations have been performed using the following material parameters, corresponding to commercially pure Ti oxidized at 700°C [5]: $D_1 = 3.92 \times 10^{-4} \mu m^2/sec$, $D_2 = 1.431 \times 10^{-3} \mu m^2/sec$, $[C] = 0.5$ and $C_{cr} = 0.65$, where the values of the concentration have been normalized by the value of the concentration on the Dirichlet boundary. Two test cases have been considered to perform convergence study of the developed Finite Differences numerical algorithm. The first case corresponds to a 24hrs. oxidation from the surface of a 30° internal wedge ($\theta = 15^\circ$) and the second case corresponds to an oxidation from a crack ($\theta = 0^\circ$). The convergence study has been performed using the following sequence of meshes: $20 \times 20$, $40 \times 40$, $80 \times 80$ and $160 \times 160$. The results from the convergence study are presented in Fig.6 and Fig.7. It can be seen that the sequence of the results is convergent. A careful observation of the results in Fig.6 shows that there is a disturbance in the contour lines near the wedge tip. Examination of the numerical algorithm shows that this effect is caused by the non-smoothness of the transformation $X$ at the wedge tip due to the discontinuity of the derivative of function $\varphi(x)$ in equation (12). This distortion is not observed for the results in Fig.7, since for the second test case the function $\varphi(x)$ has continuous derivative.

Comparison of the results obtained with FEM and the developed FDM has been performed for the case of oxidation from the surface of a 30° internal wedge. The positions of the oxidation front after 6, 12, 18 and 24hrs. of oxidation obtained with both methods are shown in Fig.8. It can be seen that the results are in good agreement. It should be mentioned that the comparison of the computational times shows that the developed FDM algorithm is much faster that the FEM algorithm: for a mesh of 400 nodes the measured computational time on a SGI O2 workstation for FDM is 30sec. while the time for FEM is 71min.

The growth of the oxide layer from the surfaces of an internal wedge, shown in Fig.1 has been modeled using both FDM and FEM algorithms. The computational domain for the FDM has been approximated as a 72° internal wedge (shown with dashed line in Fig.9) while for the case of FEM a close approximation of the real domain has been used (shown with solid line in Fig.9). The comparison of the experimental results after 72hrs. of oxidation with the numerical results is shown in Fig.10 and Fig.11 for FDM and FEM, respectively. It can be seen that while the FDM captures the general shape of the oxide front and gives a good approximation far from the wedge tip, there is a significant discrepancy close to the wedge tip. This effect is caused both by the initial error in the approximation of the computational domain and by the fact that the real wedge is not exactly symmetric. The results shown in Fig.11 suggest that the FEM algorithm gives much better approximation of the oxidation front near the wedge tip while far from the tip the results are close to the ones obtained with FDM.
Figure 6: Convergence study for the case of a 30° internal wedge ($\theta = 15^\circ$).

Figure 7: Convergence study for the case of a crack ($\theta = 0^\circ$).

Figure 8: Comparison of the results obtained with FDM and FEM for the case of a 30° internal wedge ($\theta = 15^\circ$).
Figure 9: Approximation of the computational domain for FDM and FEM.
Figure 10: Location of the oxidation front after 72hrs. of oxidation at 700°C of a Ti wedge: comparison of the experimental results with results obtained with FDM.
Figure 11: Location of the oxidation front after 72hrs. of oxidation at 700°C of a Ti wedge: comparison of the experimental results with results obtained with FEM.
5 Conclusions

Oxide formation from the surfaces of internal metallic wedges has been considered in this work. A diffusion controlled model with moving boundary has been used to model the oxidation process. The moving phase change interface has been handled by a smearing interface approach. A numerical algorithm for dealing with the non-regular computational domain has been implemented. The algorithm has been formulated for a generalized non-rectangular domain and applied to the specific case of a domain corresponding to an internal wedge. After transforming the domain, the Finite Difference Method has been used to derive the discrete form of the governing equation. A variant of the Finite Element Method algorithm has also been implemented. The developed numerical procedures have been used to model the growth of the oxide layer from the surface of an internal wedge. Results from numerical simulation for different cases have been presented and discussed.

Acknowledgments — The authors wish to express appreciation for the support provided by AFOSR grant No.F49620-97-1-0338. The third author acknowledges the support provided by grant MM-811. The authors also acknowledge P.K. Imbrie for the SEM photographs of the oxidized specimens and for his valuable comments and discussions.

References


