

ODIN v1.01 – A finite-difference model to predict solute depletion in alloys due to high temperature oxidation

by Dr Bill Pragnell, University of Birmingham, 2005

Including:

- The Java runtime environment (JRE) from Sun. This software is the 'virtual CPU' of Java, and must be installed before any Java applications can be run. It is also available free for download from <http://java.sun.com>.
- The file '*Odin.jar*'. This is an executable file and can be run just like an .exe, by double-clicking or invoking from a command prompt. If you have any software installed that recognizes / opens .jar files (e.g. WinRar), you may need to de-associate it from the .jar extension before it can be run.
- This manual.

System requirements

This JRE is for Windows 95/98/2000/XP. However, the ODIN program is platform independent and may be run using any JRE, for example, that built into MacOS X. Recommended machine specifications depend completely upon the complexity of the simulation you wish to run and how patient you are. As a rough guide, the program was developed on a 900MHz Win98 system with 128MB RAM, without problems. However, for large grids, as much memory and speed as you can lay your hands on are recommended. You may find that making a grid larger has a disproportionate effect upon running time; this is usually because Windows has suddenly decided to start using virtual memory (i.e., the hard drive) instead of the main RAM, which is much faster. The JRE install occupies some 15Mb of disk space; by contrast, '*Odin.jar*' is about 31K.

Installation

Simply run the file '*jre1.3.1.exe*' and follow the instructions to install the JRE. '*Odin.jar*' may be kept and run from wherever is most convenient, or referenced via a shortcut.

Overview

Model assumptions

- a) Only one solute forms oxide
- b) The oxide is formed externally in a dense adherent layer
- c) The oxide does not spall or crack during exposure
- d) The oxide/metal interface remains at the location of the original metal surface, i.e. interface recession does not occur
- e) The alloy is a homogenous single-phase material with no composition discontinuities, i.e. multiphase systems, coatings or diffusion couples are not considered
- f) Diffusion coefficients are independent of local composition
- g) The sample geometry is assumed to be infinite in the third dimension

A typical usage cycle of the model might be as follows:

- 1) Select alloy system and composition
- 2) Specify diffusion coefficients
- 3) Specify oxidation kinetics
- 4) Specify geometry required
- 5) Select output options
- 6) Run model

7) Start over

Main window

The main window is the top level of the application. It contains the most fundamental input fields, defining the composition of the material to be modeled, and four buttons which provide access to the diffusion, kinetics, geometry and output dialogs respectively. Below these is situated the messages window which is used for displaying any errors that may arise in the input parameters, and the simulation progress. Below the messages window are a further four buttons. The left-hand two buttons allow the current state of the input fields to be saved or loaded to/from a data file. This data file is also a readable text file, and may be examined using any standard text editor or document processor. The right-hand buttons control the simulation. Once started, the model may be paused/resumed by using the same button as prompted. When paused, the current elapsed time of the simulation is displayed in the messages window. The other button is a reset button, and terminates the current simulation.

C₁ and C₂ concentration

These fields are used for specifying the initial concentration of the oxidising solute (*C₁*) and the secondary solute (*C₂*). The units can be either atomic percent (at%) or mass percent (wt%), selected via the 'units' pull-down menu below.

Alloy system

This pull-down menu defines the alloy system under examination. For FeCrAl, NiCrAl and CoNiCrAl, Al is taken as *C₁*, Cr as *C₂*, and the matrix as the balance (Co and Ni are split in the latter case). For the FeCrNi system, Cr is taken as *C₁* and Ni as *C₂*.

Oxide method

This menu is included for completeness and permits the choice of two different oxide assumptions, specifically, whether to use a value of the Pilling-Bedworth ratio calculated from the molar masses of the alloy components, or a value corresponding to oxide formation on a pure substrate. It is recommended that the former option is used; it has been found to give good agreement with experimental results, whereas the latter does not.

Initial Dt and Max Dt

These fields are for fine-tuning the time increment Δt during the simulation, and for most scenarios the default values should be adequate. Although the stability of the finite-difference equations is controlled automatically by the program, the interfacial solute flux at very short times can be extremely high. In order to ensure accuracy, the time step is ramped from a very small value (specified in Initial Dt) to the maximum permitted. This is either decided by the internal stability criterion or the Max Dt field, whichever is smaller. The default Max Dt value of 120 seconds is fairly arbitrary – for some materials (e.g. ferritic chromia-formers) diffusion speeds can be small enough that values of Δt up to several hours may be permitted, so increasing this can minimize the simulation time.

Diffusion coefficients

This dialog allows the selection of diffusion coefficients, in units of cm^2s^{-1} : either a single value for the oxidising solute or ternary cross-terms which control both solute species. In most cases a single value is sufficient, and in any case is usually easier to come by than ternary cross-terms, which are scarce. It should be noted that the model does not account individually for grain boundary or lattice diffusion mechanisms; if the coefficients specified here are obtained experimentally (as is usually the case) then they will be effective coefficients comprising contributions from both mechanisms. The effective coefficient can also be calculated using

$$D_{\text{eff}} = \delta D_{\text{gb}} + (1-\delta)D_{\text{lattice}} \quad (1)$$

where D_{gb} and D_{lattice} are the grain boundary and lattice diffusion coefficients and δ is the volume fraction occupied by grain boundaries.

Oxidation kinetics

The oxidation kinetics dialog allows the specification of the oxidation growth rate law. All the simulation need consider is the solute removal at the metal/oxide interface as a function of time, so the oxidation rate is modelled using simple empirical equations. There are two options: firstly, a power law of the form

$$\Delta m/A = k \cdot t^n \quad (2)$$

where $\Delta m/A$ is the specific mass gain in $\text{mg cm}^{-2} \text{hr}^{-1}$, k is the rate constant in $\text{mg}^2 \text{cm}^{-4} \text{hr}^{-n}$ and n is the rate exponent. For the case of parabolic oxidation, $n = 0.5$ and $k \equiv k_p$. In such cases, care should be taken when entering a rate constant, since equation 2 is often expressed in the form

$$(\Delta m/A)^2 = k_p \cdot t \quad (3)$$

in which case $k = k_p^{0.5}$ in equation 2.

The second option is a hyperbolic equation of the form

$$\frac{\Delta m}{A} = \frac{at}{b+t} + \frac{ct}{d+t} \quad (4)$$

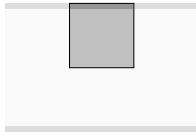
where a , b , c and d are empirical fit constants. This equation is useful for modelling transitional oxidation rates at short times which cannot be described by a power law. It should be remembered that the constants a - d should be fit to values of mass gain against time in seconds, not hours; the rate constant in equation 2 is converted to s^{-1} whereas these are not.

Grid geometries

All grids employed by the model are contained within a rectangular area extending from the origin (0, 0) in the top-left corner to $(x_{\text{max}}, -y_{\text{max}})$ in the bottom-right corner. All grid shapes are rectilinear in nature, composed of a rectangular array of nodes bounded by either oxidising surfaces or zero flux planes. Briefly, a zero flux plane is defined as a plane across which there is no net solute flux; i.e., matter flows away from it equally in both directions. This will occur at any plane of symmetry within a sample (e.g., the mid-plane of a foil), making a zero flux plane a diffusional 'mirror' and providing a method for minimizing the number of nodes in a grid.

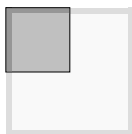
Since construction of arbitrary grid geometries is both complicated to implement for the programmer and potentially arduous for the user, a selection of predefined shapes have been supplied which can be given the dimensions required by the user and constructed automatically by the program. These geometries are listed below and illustrated by simple diagrams: the region within the black rectangle represents the model grid, and the effective geometry thus simulated is shown in the paler shades.

Plane surface



This geometry is a rectangular region bounded by a single oxidising surface and three zero flux planes. It defines the cross-section of a planar sheet or foil with infinite in-plane extent, oxidising on both sides.

Corner



This geometry is a rectangular region bounded by two oxidising surfaces and two zero flux planes. It defines a quarter cross-section of a rectangular rod of infinite length.

Double corner



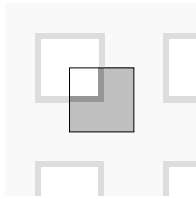
This geometry is a rectangular region bounded by three oxidising surfaces and one zero flux plane. It defines half a cross-section of a rectangular rod of infinite length.

Rectangular section



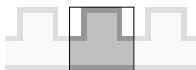
This geometry is a rectangular region bounded by four oxidising surfaces. It defines the complete cross-section of a rectangular rod of infinite length. (The last three grids actually describe the same scenario, but it can be useful to look at a complete grid in some cases).

Interior corner



This geometry is an L-shaped region bounded by two oxidising surfaces on the concave sides and four zero flux planes. It defines an array of rectangular holes repeating in both directions, like an air-brick.

Rib



This geometry is a T-shaped region bounded by six oxidising surfaces on the upright, the underside and topside of the cross-piece, and two zero flux planes, at each end of the cross-piece. It defines an infinite planar sheet with parallel rows of rectangular ribs on one side.

Note: these geometries were created to model suitable sample geometries during testing. The list is neither exhaustive nor final, and may be added to with relative ease. If any geometries are required which cannot be treated using the above grids, please contact the author (see below).

Width and depth

These fields define the width and depth of the rectangular grid region in microns.

X and Y nodes

These fields define the number of nodes in the x and y directions. The node spacings Δx and Δy in each direction are then calculated using $\Delta x = \text{width}/(\text{nodes}_x - 1)$.

Feature width and depth

These fields are used to provide the additional dimensional information (in microns) needed for the interior corner and rib shapes (and have no effect on the others). For the interior corner, they define the size of the empty space in the top-left corner. For the rib, they define the size of the rectangular rib protuberance. It

should be noted that a feature will not line up with the grid exactly if the calculated node spacing is not also a factor of the feature size. In such cases, the program will truncate the feature to the nearest gridlines.

Output options

The program writes out four separate types of data file. The files are given the extension .dat because they are data files, but all values are written as text so they can also be opened using any text editor. Columns of values are white-space separated, and can easily be imported by spreadsheets or graphing software such as Microsoft Excel or SigmaPlot. The files are written in the same directory as the file 'Odin.jar'; if the program is quit and run again, previous output files with the same filenames will be overwritten.

First of all, when the simulation starts it generates the file 'Log.dat', which contains a record of the input parameters used for the simulation. Secondly, the file 'Continuous.dat' is written at regular intervals throughout the course of the simulation. 'Continuous.dat' contains a list of time values in hours, along with corresponding values for the time increment Δt in seconds, the net mass gain in mg cm^{-2} , the solute flux in $\text{mg cm}^{-2} \text{s}^{-1}$, the interface concentration, and finally the interpolated concentration at a user-defined location within the grid. For the latter two values, the concentration units (at% or wt%) can also be specified by the user. The interface concentration will vary across the surface depending on the geometry under consideration, so the value actually written is that of the top-left-most interface node. For the plane surface geometry this is the same as any other interface node; for the corner, double corner and closed rectangular geometries this is the corner node; for the interior corner geometry this will be the interface node furthest from the cusp (at the top of the 'L'); for the rib geometry this will be the top-left corner of the rectangular rib.

The data files 'Profile.dat' and 'Map.dat', are the most important, and contain detailed snapshots of the sample composition at the user-specified times. Each data set is written to its own file, so if three output times are specified there will be three each of 'Profile.dat' and 'Map.dat' with the output time (in hours) appended to the filenames. For example, if output is specified after 1.5 hours and 5 hours, files 'Profile00001_50.dat' and 'Profile00005.dat' are output (and similar for 'Map.dat'). 'Profile.dat' is an interpolated 1D line profile such as might be obtained experimentally from a cross-sectioned specimen using EDX analysis, and consists of 200 points spaced equally along a line between two user-defined locations within the grid. At each point, the distance x (in microns) along the line is written along with the concentrations of all three alloy components in either at% or wt% (as specified). Also output, before the profile values, are the current elapsed time, the current time increment Δt and the number of iterations performed. The final file, 'Map.dat', is a 3D surface describing the composition across the entire grid. The surface takes the form of two coordinate columns in x and y (in microns), and the concentrations of all three alloy components in either at% or wt% (as specified). As with the line profiles, the time, time step and iteration number are written just before the map data. It should be noted that 'Map.dat' can end up extremely large if using a small grid spacing and without the 'restrict area' option (see below); several Mb or more is quite common. It is recommended that the Odin program be exited before the output files are opened in other applications to prevent access conflicts.

Mesh

If this is ticked, 'Map.dat' files will be created. The units of concentration can be selected using the drop-down menu.

Separate rows

If this is ticked, blank lines will be inserted in the 'Map.dat' files between each row of mesh points. This data format is required by certain plotting programs, e.g. Gnuplot, but is unsuitable for, e.g., SigmaPlot, which requires continuous data.

Restrict area

This section can be used to restrict the area output in the 'Map.dat' files. If the grid is necessarily large but only a small area is of interest, a rectangular region can be specified by top-left and bottom-right coordinates, in microns. It should be remembered that the grid extends from the origin in the positive x -direction but the negative y -direction, with (0, 0) in the top-left corner.

Line profile

If this is ticked, 'Profile.dat' files will be created. The units of concentration can be selected using the drop-down menu. The two fields below this specify the start and end coordinates of the line in microns.

Output times

This field is for specifying a list of output times for map and profile data. Times should be given in hours, separated with commas and without spaces. At each output time, a message will appear in the messages box to inform the user. Once the final time has been reached, the simulation will terminate. If neither Mesh nor Profile is ticked, no output files will be written although the times will still be reported.

Continuous output options

Here the output units (at% or wt%) can be chosen for the interface and user-defined point concentrations written to Continuous.dat. In addition, the coordinates of the point are specified in microns.

Note: while the program will attempt to trap errors arising from invalid coordinate inputs, problems may still occur in certain circumstances. To ensure smooth operation, all coordinates should be kept within the rectangle defined by the width and depth fields, i.e. within the range (0, 0) to (width, -depth).

Important: fatal errors

The program has been tested fairly thoroughly, and will trap and report simple errors like invalid input or files. However, it is possible that for certain geometry configurations, the grid may be built incorrectly (i.e. missing node links). If this happens, the simulation will crash and terminate. Unfortunately, the main window will remain open and responsive – *the error will be invisible to the user*. Although the JRE will attempt to report the error, without a console open the report will not reach the screen.

There is a way to detect such errors. If the simulation is paused (see above), the current simulation time will be printed in the messages window. Resuming and then pausing the simulation again will of course produce a slightly later simulation time. However, if the second simulation time is identical to the first, this means that the simulation was not resumed and has therefore crashed.

It is therefore important to observe the simulation as it runs. Always provide at least one short-term output time (i.e., 1 hour) to make sure the simulation is running properly – most errors of this type will occur immediately when the simulation is started. If you suspect a fatal crash, try pausing and resuming the simulation to see how far it is getting. If you detect a crash, please contact the author with the exact geometry parameters so that the error may be investigated and corrected.

Contact details

For support, bug reports or geometry requests, the author can be contacted thus:

Dr Bill Pragnell
Department of Metallurgy and Materials
University of Birmingham
Edgbaston
Birmingham
B15 2TT
UK
+44 (0)121 414 3268
Email: w.m.pragnell@bham.ac.uk