ELSEVIER

Materials Science and Engineering A327 (2002) 29–33

MATERIALS SCIENCE & ENGINEERING A

www.elsevier.com/locate/msea

Application software for data analysis for three-dimensional atom probe microscopy

Olof Hellman [a], Justin Vandenbroucke [b], John Blatz du Rivage [a], David N. Seidman [a,*]

[a] *Department of Materials Science and Engineering, Northwestern University, 2225 N. Campus Drive, Evanston, IL 60208-3108, USA*
[b] *Department of Physics, Stanford University, Stanford, CA 94305-4060, USA*

Abstract

We present a custom MacOS-based application, ADAM, for analysis of data collected by a three-dimensional atom probe (3DAP). The application is designed to carry out a common set of analysis tasks, to be customizable, to provide easy export and import of data, and to be simple enough for novice users to understand quickly. The integration of both graphical (GUI) and scripting (SUI) user interfaces and the functionality accessible from both is discussed. Different types of visualization can be used for best presentation of 3DAP data. © 2002 Elsevier Science B.V. All rights reserved.

Keywords: Three-dimensional atom probe; Atom probe-data analysis; Software for atom-probe analysis

1. Introduction

The three-dimensional atom probe (3DAP) method [1,2] is an analysis technique combining field ion microscopy and time-of-flight mass spectrometry. Specimens consist of very sharply polished metal wires. The result is a set of time-of-flight data positioned in three dimensions representing the original position of each ion in the sample, covering an analysis volume of the order of 20 nm × 20 nm × 100 nm, i.e. 10^6 atoms. Extraction of useful information from this large dataset requires intelligent data analysis and visualization.

Our approach to this problem has been to develop a custom application, ADAM, which incorporates some key analysis and visualization functionality, is flexible enough to leverage the processing capabilities of other applications and is easy to use. This paper presents ADAM's basic functionality and user interface.

2. Software

2.1. Functionality

ADAM uses the data acquired in a 3DAP experiment to produce many different kinds of data, which can be represented in a number of ways. ADAM produces concentration profiles, concentration maps, concentration spaces, isosurfaces, proxigrams, concentration distributions, etc. ADAM produces these data based on mass ranges and geometrical objects that are defined in relation to the dataset. For a concentration profile, atom types are defined by the mass ranges, and a region of interest is defined by a cylinder.

ADAM provides 3D visualization of many of the objects, and real-time display of concentration profiles and mass spectra. Data, both in graphical and numeric form, can be copied from ADAM for placement in other applications. ADAM also supports arbitrary data access, object manipulation and user customization through a scripting interface.

2.2. Design

ADAM is a document-centered application. That is, data from the 3DAP can be used as the source of data for an ADAM document. Parts of the analysis, such as definition of mass ranges for assigning species types and

* Corresponding author. Tel.: +1-847-491-4391; fax: +1-847-467-2269.
E-mail address: d-seidman@northwestern.edu (D.N. Seidman).

PII: S0921-5093(01)01887-1

definition of regions of interest in the reconstructed data are persistent: when the document is saved and reopened, these features are saved and reloaded along with the rest of the data.

This is in contrast to a data-flow model for computation. A data-flow model for analysis performs some sort of transformation on some input data, producing an explicit result. Data-flow based applications can be powerful and simple because they match the code-flow structure of common programming languages, but they are often difficult to match with a sensible user interface [3]. Document models have become pervasive in personal computing applications because they make more sense to users.

Document-centered applications do not produce explicit results as do data-flow analyses. Instead, different kinds of data are offered to a user who is manipulating a document. The user chooses to extract some data from the current document and place it elsewhere. This action produces a result analogous to the data-flow model, but is initiated by the user.

The power of the document model lies in the ability of the user to manipulate the document in a variety of ways and to extract different kinds of data without needing to reprogram the logic of the data flow. This is commonly achieved by allowing a user to make a selection and perform some operation on the selection, like copying its data or moving it to a new location.

For a document model application to work, the various objects and properties of a document need to be persistent, and organized in a logical way. The *object model* is the term used to describe this organization [4]. ADAM is based on the AppleEvent object model [5], in which each application object has a set of properties and a set of elements. Each element also has a set of properties and can have elements inside it. Documents are a class of objects in the AppleEvent object model, and the ADAM application contains documents as its elements.

2.3. Parallel user interfaces

ADAM provides both a graphical user interface (GUI) and a scripting user interface (SUI). Users will find that some tasks are more suited to one than the other. The GUI consists of user actions, such as mouse clicks on interface elements such as menus and buttons, and of the graphical feedback produced in visualization. The SUI consists of the ability of users to write scripts in text form that direct the application on how to manipulate its elements or what data should be extracted, and of the presentation of the results of such commands in text form.

The SUI is based on the interpreted language AppleScript [6]. As an example, one might want to define a cylinder of analysis for a concentration profile. To specify a name, radius and direction for the cylinder using the SUI, one uses the following script:

```
make new cylinder with properties {name: 'Monica',
radius: 5, axis direction: {0,0,1}}
```

Subsequently, one can use this new cylinder as a spatial reference. For example

```
count the number of atoms of cylinder 'Monica'
```

returns the number of atoms in that cylinder and

```
get the data of concentration profile of cylinder
'Monica'
```

returns the numerical data for the graph of the profile corresponding to that cylinder. Objects can be specified in very flexible terms, for example

```
select every atom whose mass is less than 30 and
mass is greater than 28
```

The scripting interface is particularly well suited for tasks requiring precision and reproducibility, especially those where the precision or flexibility of a GUI is not sufficient. For example, the placement and orientation of a cylinder can be controlled with the GUI, but not with the precision and reproducibility possible through a text-based scripting interface.

The SUI is self-documenting. That is, the application provides a dictionary consisting of the terminology that is appropriate for the application, including the names of all the appropriate classes of objects and the names of any property that might be useful.

The GUI and SUI interfaces can be mixed together. That is, if a scripting command refers to a graphical user interface element or a property, which affects the display, ADAM will respond with the updated display. Furthermore, many GUI actions will result in an equivalent command being seen in the scripting interface, which allows the recording of user interface actions in script form.

Because of the wide variety of material systems subject to atom probe analysis and the variety of physical phenomena represented in the microstructures viewed thereby, there will always be data analysis tasks which are not anticipated by an application such as ADAM. The scripting interface allows export of any data from ADAM into another application better suited to perform that calculation or export into a file in whatever format is required. This means that ADAM can be used in conjunction with other applications easily, allowing the experimenter to use the right software tool for each job.

2.4. Data objects

In this section we discuss the various objects created by the data manipulation tools provided in ADAM. There are other objects in the application not specific to ADAM such as *window*, *menu*, and *button* not discussed here.

2.5. Ions and atoms

The fundamental objects of the original data are ions and atoms. There is a subtle difference between them in that a single ion might represent zero, one or multiple atoms. Ions have properties associated with the original data collection and positioning performed by the instrument: they have a 3D position and a mass-to-charge state ratio. If provided by the Atom Probe file, they may also have an evaporation voltage, a pulse number or other information associated with the data collection.

2.6. Mass ranges

The point of a mass spectrometry technique is to associate ions with specific elements based on the measured mass-to-charge state ratios. This is done by defining a *mass range* object ('mass window' is not used to avoid confusion with the other meaning of 'window' in the user interface). A mass range has lower and upper limits of mass-to-charge state ratio, and is associated with a specific atom type or list of atomtypes, in the case of a molecular ion. Ions that are not in a mass range are considered to correspond to no atom at all. If mass ranges overlap, ions are considered to be in the first mass range for which they qualify, and not in any of the others.

2.7. Mass spectrum

A mass spectrum can be generated from any subset of the ions; it is a histogram of the mass-to-charge state ratios of that subset. It has properties of binsize, minimum, maximum, and other parameters relating to its display on screen.

2.8. Concentration space

A concentration space is a regular 3D grid of concentration values, i.e. a scalar field array. The concentration values at each point of the grid represent the local composition at that point in the space, as calculated from a sampling of the ions in the vicinity of that point. How the sampling is performed is based on a transfer function and a characteristic length. The sampling can also produce an estimation of the error in the calculation of the local concentration values, which is important when the concentration space is used for a subsequent calculation or visualization.

The concentration space itself has a spatial dimension expressed as a number of grid points in each dimension, an origin and a grid spacing. Because the grid is regular in all dimensions, and the original data is irregular, there is no necessarily good matching between this grid and the box enclosing the original data. Concentration spaces are drawn using the VOLPACK library [7], and have properties corresponding to how this rendering should be carried out.

2.9. Isoconcentration surface

An isoconcentration surface is the isosurface corresponding to a particular concentration threshold. It is generated from the concentration space using the marching cubes algorithm [8]. Although the algorithm is a core method in computer graphics, the result of the algorithm is not just a graphics object to be drawn for visualization. It can also be treated as a real feature in the analyzed sample and used as a reference point for further calculations, as in the proxigram.

As isoconcentration surfaces are based on data provided by a concentration space, changing the properties of the concentration space, e.g. its binsize or its error tolerance, will result in an update of the isoconcentration surface. The isoconcentration surface thus has a 'concentration space' property, which is a concentration space object.

2.10. Proxigram

Essentially, the proxigram is a plot of concentration versus distance, analogous to a linear concentration profile. However, the distance axis of the proxigram is not measured in a unique direction. Instead, the distance is measured as the proximity of each point of space to a reference surface. The surface can be of any shape of arbitrary complexity, as long as it is closed and continuous. Such surfaces can be generated by calculating the isosurface of a scalar 3D field, as when an isoconcentration surface is defined from a concentration space. Positive and negative proximities correspond to the point of space being on different sides of the surface.

Because the reference surface can be any shape, the proxigram is much more flexible and much more representative of the actual concentration profiles than a simple linear profile or a radial profile could be at many interfaces, which exist in nature. Details of the generation of the proxigram are given by Hellman et al. [9]. The proxigram is appropriate for quantitative measurements of interfacial excess quantities, especially those for which determining the location of the interface is difficult or ambiguous.

2.11. Density space and isodensity surface

A density space is similar to a concentration space, but with local densities: thus it is a scalar field array. Visualization of the density space is done with VOLPACK [7]. Variations in local density in 3DAP data is a function of both variations in the sample density and

the perturbation in ion optics caused by both the crystal lattice of the sample and inhomogeneities in the shape and dielectric constant of the sample tip [10].

The isodensity surface is the surface corresponding to a particular value of the density of one or all the atomtypes. The concentration space and the density space both yield similar information. However, the two approaches can reveal different effects. Which one is better will depend on the sample and the kind of information that is desired.

2.12. Cylinder and plane

Cylinder and *plane* are the two geometric objects used to define regions of interest for a subsequent analysis. Cylinders are the basis for one-dimensional (1D) composition profiles and integral profiles. The axis direction specifies the direction over which to calculate the profile, while the profile is calculated based on the atoms that appear within the cylinder. Planes are the basis for composition maps. In a composition map, the local concentrations or densities are calculated at each point in the plane, considering only the atoms close to the plane near that point.

2.13. Scheme

The scheme object is the object that controls how a particular atom type is drawn during a visualization. Schemes have properties of size, color and visibility. Because different windows might be displaying the same dataset, schemes are assigned on a per window basis: i.e. some atoms might be drawn in one window but not in the other.

3. Examples

A common task in 3DAP analysis is the generation of an isoconcentration surface. One of the problems with existing tools is that there is minimal compensation for the statistical error at boundaries on the analysis volume, which produces artifacts in the visualization. ADAM accounts properly for this error. Fig. 1 shows a script which performs this task. The first line instructs the application to generate a concentration space, i.e. a 3D grid of concentration values, which will be used as input for the isoconcentration surface calculation. The dimensions property refers to the number of grid points in the x, y and z directions. The cell length refers to the spacing between the grid points. The offset is the $\{x,y,z\}$ coordinates of the zeroth grid point. As such, the concentration space generated by the script will span 100 Å in both x and y directions, and be centered on the line defined by $x = 0$, $y = 0$. The second command generates the isoconcentration surface, using the new concentration space as its input. This must be explicitly specified, because the document might contain any number of concentration spaces that were previously defined. The threshold value and the atom type are also specified.

An example of a concentration space visualization rendered by the VOLPACK library is shown in Fig. 2. In this case, the sample is an Fe–Mo–V alloy [11]. There is a phase separation to Mo-rich and Fe-rich regions. For comparison, an isoconcentration surface for 30 at.% Mo is shown in Fig. 3.

4. Conclusions

3DAP microscopy is an emerging analytical technique for which data analysis promises to be an important part of the understanding of its results. It is crucial for good data analysis tools to be available to researchers for this purpose. It is our intention to make ADAM a best-of-class application for this purpose.

Acknowledgements

This research is supported by the National Science Foundation, Division of Materials Research, Grant DMR-972896, Bruce MacDonald, Grant Officer. Dr Dieter Isheim provided the Fe–Mo–V dataset shown in the visualizations. J. Vandenbroucke and J. Blatz du Rivage were supported by an NSF Research Experiences for Undergraduates grant.

```
make new concentration space with properties {dimensions:{11,11,31}, cell length: 10,
    offset:{-50,-50,0},  name: "my cspace", confidence sigma: 1}

make new isosurface with properties {concentration space:concentration space "my
    cspace", type:"Mg" , threshold:0.025}
```

Fig. 1. Make new concentration space with properties {dimensions, {11,11,31}; cell length, 10; offset, $\{-50,-50,0\}$; name, 'my cspace'; confidence sigma, 1}. Make new isosurface with properties {concentration space, concentration space 'my cspace'; type, 'Mg'; threshold, 0.025}.

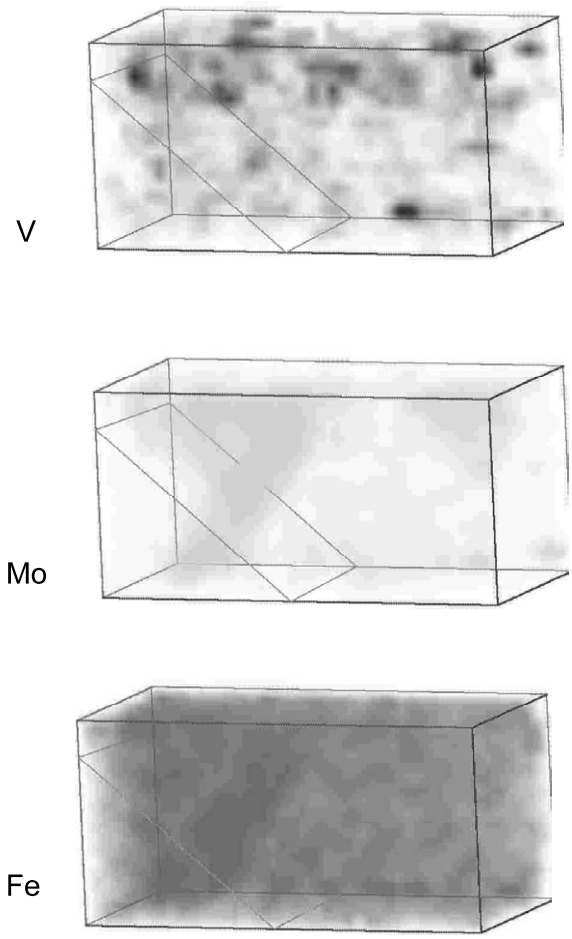

Fig. 2. Concentration space visualizations rendered with VOLPACK.

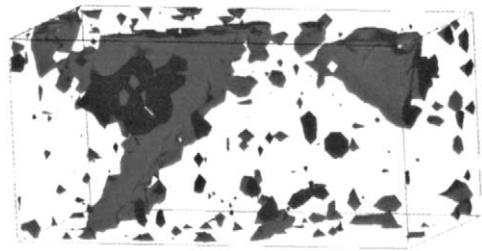

Fig. 3. 30 at.% Mo isooncentration surface for the same sample in Fig. 2. Visualization of the isoconcentration surface highlights shapes but is sensitive only to concentrations near the threshold. Additionally, statistical noise can distract from real features in the data.

References

[1] D. Blavette, B. Deconihout, A. Bostel, J.M. Sarrau, M. Bouet, A. Menand, Rev. Sci. Instrum. 64 (1993) 2911.
[2] A. Cerezo, T.J. Godfrey, S.J. Sijbrandi, G.D.W. Smith, P.J. Warren, Rev. Sci. Instrum. 69 (1998) 49.
[3] S.G. Parker, C.R. Johnson, Supercomputing'95, 1995.
[4] P.G. Smith, Develop 18 (1994) 26.
[5] R. Clark, Develop 10 (1992) 8.
[6] Apple Computer, AppleScript Language Guide: English Dialect, Apple Computer: Cupertino, CA, 1996.
[7] Philippe Lacroute, Marc Levoy, Proceedings of SIGGRAPH'94, Orlando, Florida, 24–29 July 1994, Computer Graphics Proceedings, Annual Conference Series, 1994, ACM SIGGRAPH, pp. 451–458.
[8] W.E. Lorensen, H.E. Cline, ACM SIGGRAPH Comput. Graphics 21 (1987) 163.
[9] O. Hellman, J. Vandenbroucke, Jörg Rüsing, Dieter Isheim, David N. Seidman, Microsc. Microanal. 6 (5) (2000) 437.
[10] F. Vurpillot, A. Bostel, D. Blavette, Appl. Phys. Lett. 76 (21) (2000) 3127.
[11] D. Isheim, O.C. Hellman, D.N. Seidman, F. Danoix, D. Blavette, Scripta Mater. 42 (2000) 645.