



# TauFactor: An open-source application for calculating tortuosity factors from tomographic data

S.J. Cooper<sup>a,\*</sup>, A. Bertei<sup>a</sup>, P.R. Shearing<sup>b</sup>, J.A. Kilner<sup>c</sup>, N.P. Brandon<sup>a</sup>

<sup>a</sup> *Electrochemical Science and Engineering, Earth Science and Engineering Department, Imperial College London, South Kensington, London, SW7 2AZ, UK*

<sup>b</sup> *Electrochemical Innovation Lab, Department of Chemical Engineering, UCL, London, WC1E 7JE, UK*

<sup>c</sup> *Electroceramics, Materials Department, Imperial College London, South Kensington, London, SW7 2AZ, UK*

Received 5 July 2016; received in revised form 15 September 2016; accepted 19 September 2016

## Abstract

*TauFactor* is a MatLab application for efficiently calculating the tortuosity factor, as well as volume fractions, surface areas and triple phase boundary densities, from image based microstructural data. The tortuosity factor quantifies the apparent decrease in diffusive transport resulting from convolutions of the flow paths through porous media. *TauFactor* was originally developed to improve the understanding of electrode microstructures for batteries and fuel cells; however, the tortuosity factor has been of interest to a wide range of disciplines for over a century, including geoscience, biology and optics. It is still common practice to use correlations, such as that developed by Bruggeman, to approximate the tortuosity factor, but in recent years the increasing availability of 3D imaging techniques has spurred interest in calculating this quantity more directly. This tool provides a fast and accurate computational platform applicable to the big datasets ( $>10^8$  voxels) typical of modern tomography, without requiring high computational power.

© 2016 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (<http://creativecommons.org/licenses/by/4.0/>).

**Keywords:** Tortuosity factor; Diffusion; Microstructure; Tomography

## Code metadata

Current code version	v1.10
Permanent link to code/repository used for this code version	<a href="https://github.com/ElsevierSoftwareX/SOFTX-D-16-00054">https://github.com/ElsevierSoftwareX/SOFTX-D-16-00054</a>
Legal Code License	FreeBSD License
Code versioning system used	git
Software code languages, tools, and services used	MatLab 2015+ (guide)
Compilation requirements, operating environments & dependencies	MatLab 2015+ with 'Image Processing Toolbox'
If available Link to developer documentation/manual	<a href="http://www.sourceforge.net/projects/taufactor/files">www.sourceforge.net/projects/taufactor/files</a>
Support email for questions	<a href="mailto:sjc08@ic.ac.uk">sjc08@ic.ac.uk</a>

## Software metadata

Current software version	v1.10
Permanent link to executables of this version	<a href="http://www.sourceforge.net/projects/taufactor">www.sourceforge.net/projects/taufactor</a>
Legal Software License	FreeBSD License
Computing platforms/Operating Systems	MatLab 2015+
Installation requirements & dependencies	MatLab 2015+ with 'Image Processing Toolbox'
If available, link to user manual — if formally published include a reference to the publication in the reference list	<a href="http://www.sourceforge.net/projects/taufactor/files">www.sourceforge.net/projects/taufactor/files</a>
Support email for questions	<a href="mailto:sjc08@ic.ac.uk">sjc08@ic.ac.uk</a>

\* Corresponding author.

E-mail address: [sjc08@ic.ac.uk](mailto:sjc08@ic.ac.uk) (S.J. Cooper).

## 1. Introduction

### 1.1. Theory and motivation

The effect of geometry on transport in heterogeneous media has been the focus of a great deal of academic research for well over a century, across a wide range of academic disciplines; from the flow of water through porous rocks [1], to the movement of cells within bone scaffolds [2] and from the transfer of heat through catalyst beds [3], to the passage of electrical current across battery electrodes [4]. Although for certain very simple geometries analytical solutions have been found [5], even fairly minor increases in geometric complexity preclude this approach. The recent, rapid development of 3D imaging has shown that many common materials have highly complex microstructure, which have proven difficult to analyse. In these cases, the systems can then only be described with either simulations or correlations based on other more accessible parameters, such as the volume fraction.

*TauFactor* is a MatLab application for simply and efficiently calculating the reduction in diffusive transport caused by convolution in the geometry of heterogeneous media, based on microstructural image data. This effect is captured in a term called the tortuosity factor,  $\tau$ , which is defined in

$$D^{\text{eff}} = D \frac{\varepsilon}{\tau} \quad (1)$$

where  $\varepsilon$  is the volume fraction of the conductive phase;  $D$  is the intrinsic diffusivity of the conductive phase; and  $D^{\text{eff}}$  is the effective diffusivity through a porous volume where the second phase is insulating. In this general form, the diffusive flow could represent electrical current, heat transfer or mass transport interchangeably. This formulation yields two features of the tortuosity factor: Firstly, when  $\tau = 1$ , the flow paths must be direct and, in Cartesian systems, prismatic; secondly, for all systems  $\tau \geq 1$ , otherwise the geometry would somehow be augmenting flow [6].

Before 3D geometric information was commonly available, a wide variety of porosity-tortuosity correlations were published based either on empirical observations of flow or, in some cases, on fundamental diffusion theory. The detailed review by Shen and Chen [7] summarises the origins of many of these and underlines the necessity for a more detailed approach. Perhaps the most commonly used porosity-tortuosity correlation is that derived by Bruggeman [8], which is based closely on

earlier work by Lorentz [9] and Lorenz [10] in the field of optics. An updated discussion of Bruggeman's derivation is presented in a paper by Tjaden et al. [11], which highlights some of the inevitable limitations of this and other correlation based approaches, such as the assumed isotropic nature of the pore networks. Epstein further explains why path length type analysis alone, such as in [12,13], can only be related to the tortuosity factor in simple capillary geometries and so cannot be reliably used to quantify transport in complex pore networks [14].

Over the past decade, high resolution 3D tomographic imaging has become widely available, which has created the opportunity for the tortuosity factor to be quantified directly from microstructure using simulation [15–19], although the approaches vary significantly in terms of their fidelity and computational efficiency.

### 1.2. Microstructural data

Tomography is a term that encompasses a variety of 3D imaging techniques, including X-ray computed tomography (XCT) and focused ion beam scanning electron microscopy imaging (FIB-SEM), both of which are commonly used to capture microscopic structures [20]. Tomographic data is typically processed and stored as cuboid voxels (volume pixels), initially containing grey-scale image data, but can be converted into discrete regions using segmentation algorithms, which are increasingly based around machine learning techniques [21].

This segmented data can then either be employed directly in a simulation or used to generate surface and volume meshes. Fig. 1 shows the workflow from an initially noisy greyscale tomography dataset, through segmentation to a binary volume and finally the result of a diffusion simulation performed in *TauFactor*. Although careful meshing can result in significant efficiency gains, it can also cause smoothing and other geometrical distortions. Our previous work investigated this effect for a range of methods presented in the literature and found significant discrepancies between nominally identical approaches [22]. To avoid this, *TauFactor* simulates diffusion using the voxels directly as mesh elements, as discussed in more detail in the following section.

### 1.3. Simulation

The tortuosity factor can be obtained from simulation by comparison of the steady-state diffusive flow through a pore

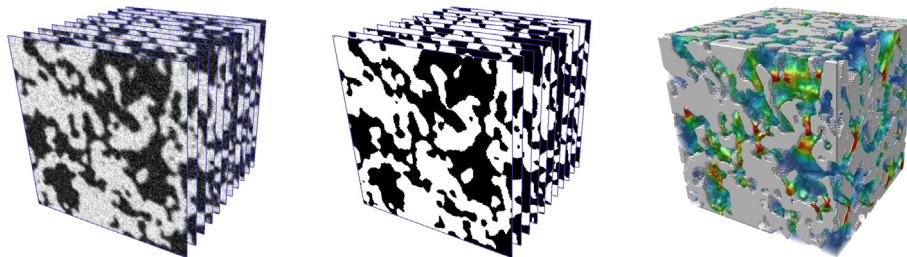


Fig. 1. Illustration of the workflow (l–r) from tomography derived greyscale data, through to a segmented 2-phase volume and finally the result of a diffusion simulation performed using *TauFactor*, highlighting regions of high flux density in red. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

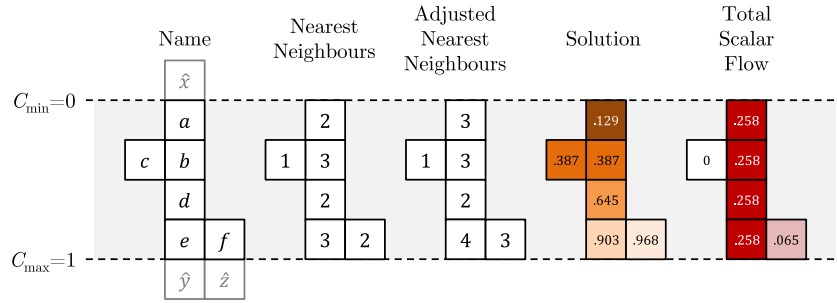


Fig. 2. Example 2D geometry to illustrate application of ghost node concept.

network,  $F_p$ , to that through a fully dense control volume of the same size, diffusivity and potential difference,  $F_{cv}$ .

$$F_p = -A_{cv} D \frac{\epsilon}{\tau} \frac{\Delta C}{L_{cv}} \quad (2)$$

$$F_{cv} = -A_{cv} D \frac{\Delta C}{L_{cv}} \quad (3)$$

where  $D$  is the diffusivity of the conductive phase;  $C$  is the local concentration of the diffusing species; and  $A_{cv}$  and  $L_{cv}$  are the cross-sectional area and length of the control volume respectively. Taking the ratio of these two expressions and rearranging yields a definition equivalent to that in Eq. (1).

The system of equations in sys. 4 captures this steady-state diffusion problem, including the fixed value (Dirichlet) conditions imposed at two parallel boundaries, where  $Q = (0, L_x) \times (0, L_y) \times (0, L_z)$  is a cuboid in  $\mathbb{R}^3$  and  $\Omega \subset Q$  is the conductive region of a porous medium inside  $Q$ .  $T$ ,  $I$  and  $B$  are two-dimensional subsets of  $Q$  (i.e. Top, Interfacial and Bottom), such that  $\partial\Omega = T \cup I \cup B$  and  $\partial\Omega|_{z=L_z} = B$ ,  $\partial\Omega|_{z=0} = T$ ,  $\partial\Omega|_{0 < z < L_z} = I$ . The flow through the porous medium  $\Omega$  is then modelled by the solution to

$$\begin{cases} \nabla^2 C = 0, & \text{in } \Omega, \\ C = 0, & \text{on } T, \\ \nabla C \cdot n = 0, & \text{on } I, \\ C = 1, & \text{on } B, \end{cases} \quad (4)$$

where  $n$  is the outward pointing unit normal to  $\Omega$ .

By remapping sys. (4) onto a set of discrete cuboid volume elements, it can be solved as a system of linear equations, which describe the concentration in each element only in terms of the concentrations of its face-adjacent neighbours. This can either be thought of as a finite-difference type description or equivalently as a directly applied mass conservation for the flux at each face. In either case, the solution for cubic elements is simply that the value at a point must be equal to the mean values of its conducting neighbours, further discussion of which can be found in [23].

It is important to note that for diffusive scenarios, a statistical continuum, as well as a physical continuum, can be modelled. This means that the governing expression would hold even for a single diffusing particle. This can be best understood from the ‘‘random walker’’ derivation of the diffusion equation, where the progress of individual particles is followed, each independent of the others, as detailed in [17].

Inconsistency in the evaluation of tortuosity factors significantly hinders the acceptance of 3D imaging as a robust and reliable material quantification technique, as results can appear arbitrary and inconsistent between studies. *TauFactor* overcomes this barrier as it solves the problem in a way that is in-line with the fundamental definition, whilst requiring no additional intervention from the user, such as remeshing or resampling (which in themselves are time consuming).

## 2. Software description

### 2.1. Software architecture

In order to efficiently enforce the Dirichlet boundaries at the surface of the boundary voxels, as specified in sys. 4, the ghost node concept is employed. For example, to find the concentration at node  $e$  of the simple 2D geometry shown in Fig. 2, we first set the node’s value to be the mean of its conductive face-adjacent neighbours.

$$e = \frac{d + f + \hat{y}}{3} \quad (5)$$

Then, in order to impose the boundary at the interface between  $e$  and  $\hat{y}$ , the following must also be true.

$$\frac{e + \hat{y}}{2} = C_{\max} \quad (6)$$

Substituting Eq. (6) into Eq. (5) and rearranging yields,

$$e = \frac{d + f + 2C_{\max}}{4} \quad (7)$$

Crucially, the ghost node,  $\hat{y}$ , does not appear in this final expression, but its presence is implied by increasing the nearest neighbour count at the boundary voxels by 1 and doubling the applied surface concentration outside the boundary. The same approach is used at the opposite boundary where  $C_{\min}$  is enforced. Fig. 2 also contains the solution to this system and the resulting total scalar flow passing through each voxel. Notice that the presence of node  $f$  has increased the total flow through the system, whereas node  $c$  has had no impact.

Now that the boundaries have been suitably defined, it is possible to solve this steady-state problem directly using matrix inversion by simply excluding the non-conductive nodes. However, *TauFactor* employs an iterative method with Over Relaxation (OR), which significantly accelerates convergence

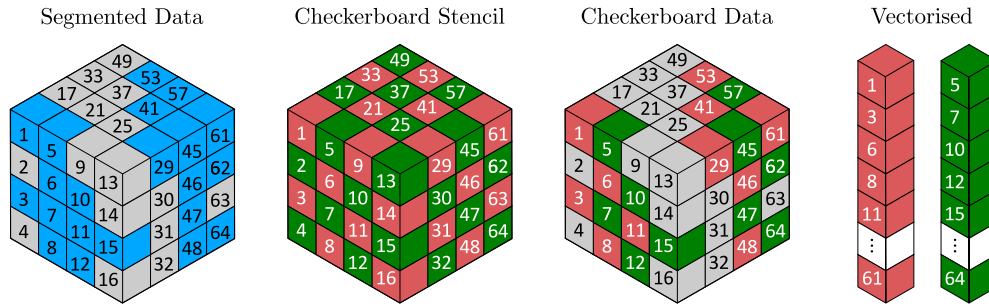


Fig. 3. Schematic representation of converting 3D geometric data (where the blue voxels represent the conducting phase) into checkerboard adjusted vectors. These two optimisations in combination increased the speed of the code by at least a factor of 4. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

(around 3 orders of magnitude fewer iterations required), whilst using little additional memory. This approach is well established in the computational literature and uses a simple linear extrapolation based on both the current value at a node and that found from the conventional Jacobi method.

Further efficiency gains are derived from decomposing the volume array into checkerboarded, vectorised subdomains, as illustrated in Fig. 3. The vectorisation is crucial for efficiency in the MatLab environment and checkerboarding effectively halves the number of calculations required for an iteration [24].

The following pseudocode shows the structure of the program that is called when the *Tortuosity* button is pressed in the GUI.

```

%% Begin Function
Check phases and directions specified by the user.
if Voxels are specified as anisotropic by user
    Calculate directional weighting factors
end
Calculate number of "Nearest Neighbours" at each
    conducting voxel
Apply checkerboarding to find addresses of RED and
    GREEN voxels
Build address vectors of conducting neighbours
while Convergence criteria not met
    Over-relaxed iteration on RED vector
    Over-relaxed iteration on GREEN vector
    if Batch of iterations complete
        Calculate tortuosity at top and base
        Assess convergence using tortuosities
    end
end
Generate report figure
%% End Function

```

Work by Rhazaoui et al. employed a direct solution approach to solving this problem through matrix inversion in the MatLab environment. Using an Intel Xeon® octo-core processor @2.4 GHz with 24 GB of RAM, the effective conductivity of a  $10^6$  voxel volume took over 2 h to solve [25]. The iterative approach deployed in *TauFactor* is capable of solving the same size volume in less than 30 s using a single Intel i7® @2.9 GHz core and *c.* 100 MB of RAM.

## 2.2. Software functionalities

Microstructural data must be loaded as a single \*.tif file (3D or 2D), in which the relevant phases (up to 3) have already

been segmented. This data is then converted into a 8-bit integer array and presented in the application window, as shown in Fig. 4, along with the labelling scheme for the phases (black, white, green) and orientation (1, 2, 3) system. If the user selects multiple \*.tif files in the loading prompt, these will be treated as a batch. This means that whatever calculations are called will be applied to each microstructure sequentially, with the results allocated to separate structured variables in the main MatLab workspace. It is also possible to call the functions in *TauFactor* directly from within another script, without opening the GUI. The form of the required input variables is detailed in the manual that accompanies this paper.

For each value of the tortuosity factor calculated, *TauFactor* generates a report figure, shown in Fig. 6, that illustrates the distribution of concentration and flux in a slice of the volume. The report also contains a graph of the tortuosity factor, as calculated at the top and bottom faces, against iteration number. Convergence of the system was established by analysis of both the rate of change of these two tortuosity factor values, as well as comparison of their absolute values. This hybrid approach allows the program to account for both slow convergence and symmetrical systems.

In addition to the tortuosity factor, the report also provides several other relevant microstructural quantities including percolation fraction and a graph of volume fraction variation in the direction considered. The scalar concentration values can be used to calculate a map of the total flux through each voxel, as well as the separate contributions from the through-plane and in-plane directions, and can be exported as normalised \*.tif stacks. These flux maps can be used for visualising bottlenecks (see Fig. 1), which, in the case of current flow, for example, may be useful for understanding localised degradation mechanisms through over heating.

Four types of representative volume analysis (RVA) can also be implemented by selecting a method from the drop down box. Three of these methods are shown in Fig. 5 and in each case increments of approximately 5% are used to generate a plot suitable for investigating the extent to which the volume considered can be expected to represent the material in general. The fourth method is similar to the third, except that the study starts from the base of the volume rather than the top.

The report shown in Fig. 6(b) contains three further RVAs (generated when the *Metrics* button is pressed) for the volume

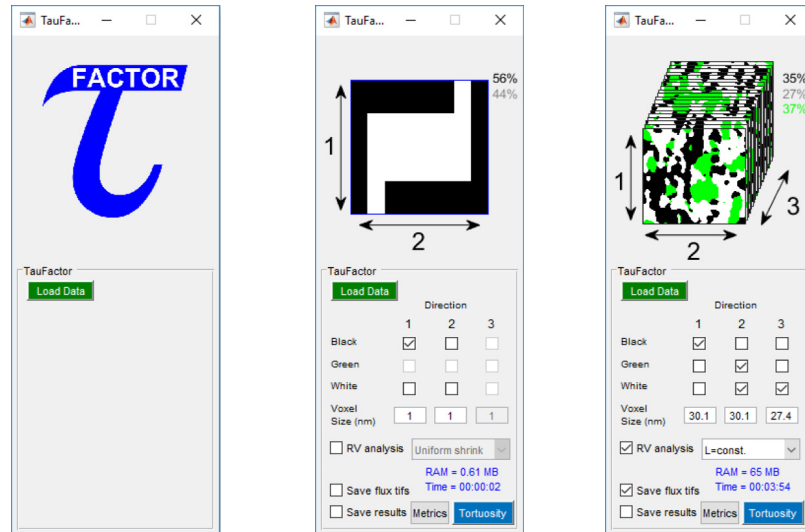


Fig. 4. The window view when the *TauFactor* application is (l-r) first opened, after a simple 2D dataset with 2 phases has been loaded and after a 3D dataset with 3 phases has been loaded.

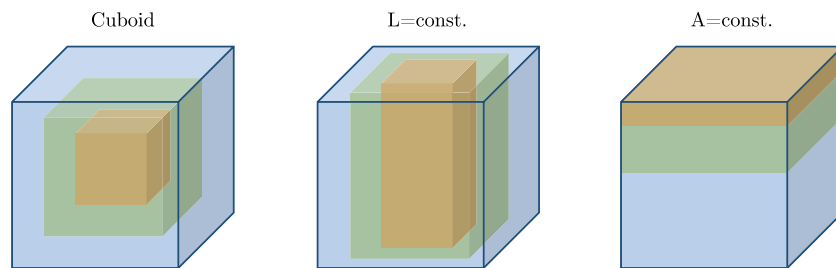


Fig. 5. Illustration of the three methods of representative volume analysis. In each case, three volume steps are shown, whereas in reality 20 steps (for the *Metrics* study) and 10 steps (for the *Tortuosity* study) are taken of constant volume increment.

fractions, surface areas and triple phase boundary (TPB) densities. In a cuboid lattice, each edge is in contact with four voxels; TPBs are edges where three of these four voxels contain different phases. Similarly, each cuboid face is in contact with two voxels and is labelled as an interface if these two are not of the same phase. For the surface area and TPB studies, the areas and lengths of the cuboid faces and edges were used directly, rather than applying any smoothing correction (such as “Marching Cubes” [26]). Also, for both of these metrics, the outermost faces and edges of the dataset cannot be assigned definitive labels, as we do not have any information about the next voxel layer. To overcome this problem, the first layer is excluded from the volume density calculations and its information is only used for labelling of the second layer. In addition, to ensure that the number of edges and surfaces scale linearly with volume (i.e. that  $3n^3$  are used in the density calculation), those edges/faces occurring at the outer surface of the ROI are in effect weighted by 0.5 in each step of the RVA.

It is worth noting that using the faces and edges of the cuboid voxels to approximate the true surface area and TPB density of a real microstructure is likely to be a source of significant error. However, whether this approach will under- or overestimate the truth will depend on the nature of the sample being imaged and the resolution of the imaging technique. Fig. 7 shows three noteworthy scenarios in which spheres are represented

by cubic voxels, under the assumption of “perfect tomography” (i.e. voxels containing  $\geq 0.5$  of a phase are modelled as a voxel entirely made from that phase).

The first scenario shows the limiting case where a single spherical particle is represented by a single voxel (assumed to be surrounded by empty voxels), causing an over estimation of the surface area by a factor of  $c. 2$ . The second case shows a sphere represented by many voxels, which can again be shown analytically to cause a minimum overestimation factor of 1.5 [27]. Finally, it is clear from the third example that if a single voxel contains a highly complex structure (i.e. the smallest features are far smaller than the imaging resolution), this will result in an arbitrarily large underestimation of the true surface area and TPB densities. It must therefore be left to the user to make a judgement on a case-by-case basis about what correction factors might need to be applied and whether the imaging resolution was appropriate to the sample.

The final key feature of *TauFactor* is that it allows for the user to specify the three dimensions of the cuboid voxels. For the tortuosity factor calculation, although the absolute voxel size is not relevant (as it is a dimensionless parameter), anisotropy does effect the diffusion and must be accounted for with an additional correction factor.

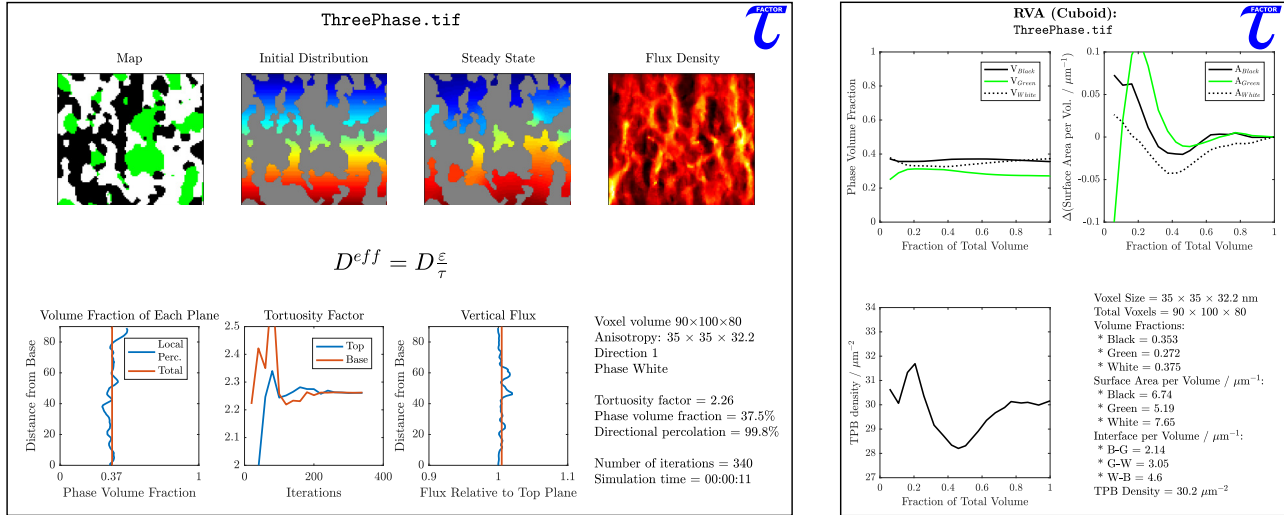


Fig. 6. Reports generated by *TauFactor* for (left) a tortuosity factor calculation and (right) a representative volume study of various other metrics including volume fractions, surface area and triple phase boundary densities on some example three phase microstructural data.

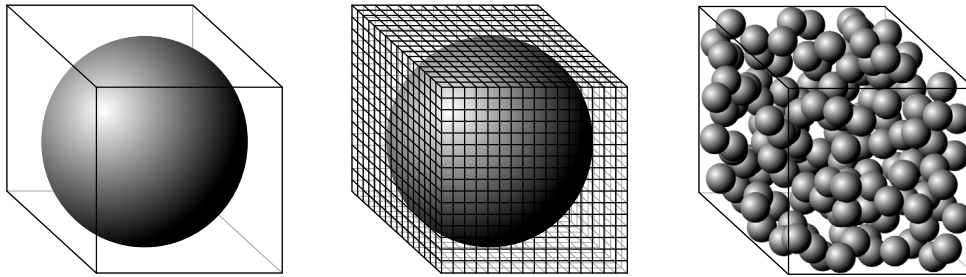


Fig. 7. Illustration of the potential estimation errors caused by cuboid voxel representations. In each case, “perfect tomography” is assumed, such that voxels containing  $\geq 0.5$  of a phase are modelled as a voxel entirely made from that phase.

If the dimensions of a cuboid voxel are  $l_1$ ,  $l_2$  and  $l_3$ , then a weighting factor,  $\phi$ , needs to be calculated for each direction.

$$\phi_1 = \frac{1}{2} \left( \frac{l_1 l_2}{l_3} + \frac{l_3 l_1}{l_2} + \frac{l_2 l_3}{l_1} \right)^{-1} \frac{l_2 l_3}{l_1}. \quad (8)$$

The map of geometry data,  $M$ , is a binary array where the conductive voxels are labelled as 1s. It is used to calculate a map of nearest neighbours, corrected to account for the changes in separation distance between adjacent nodes,  $B_{\text{ani}}$ ,

$$B_{\text{ani}} = [\phi_1 (M_{\text{north}} + M_{\text{south}}) + \phi_2 (M_{\text{east}} + M_{\text{west}}) + \phi_3 (M_{\text{up}} + M_{\text{down}})] \quad (9)$$

where the subscripts of  $M$  represent the location on the map relative to the voxel considered. With these two adjustments, the anisotropic 3D Jacobi solution becomes

$$C^{i+1} = [\phi_1 (C_{\text{north}}^i + C_{\text{south}}^i) + \phi_2 (C_{\text{east}}^i + C_{\text{west}}^i) + \phi_3 (C_{\text{up}}^i + C_{\text{down}}^i)] / B_{\text{ani}} \quad (10)$$

where  $C^i$  is the concentration after iteration  $i$  and once again subscripts represent the location of a node, relative to that being evaluated.

For the surface area and TPB density calculations, both the absolute dimensions and the anisotropy of the voxels are

important, but are simple factors to apply as the contributions to these metrics in each of the three directions are calculated separately.

### 3. Simulation capabilities

#### 3.1. Illustrative example

A previous study by the author compared tortuosity factor values calculated using a variety of techniques cited in the literature [22]. The microstructural data used in this work were taken from [28] and consisted of two X-ray nano-tomography datasets of a small solid oxide fuel cell cathode sample. The study found that calculated tortuosity factors varied not just between different theoretical frameworks (such as distance mapping as a proxy for diffusion), but also between nominally identical diffusive simulations.

Fig. 8 was constructed using the same data as in Fig. 1 of [22], but was also extended to include the results obtained using *TauFactor*. For further details about the alternative simulation tools used, see [22].

As is clear for Fig. 8, although there is significant variation between methods, *TauFactor* reproduces the results generated by the *Avizo XLab* simulation. This is to be expected as this simulation also used the segmented voxel data directly as the grid.

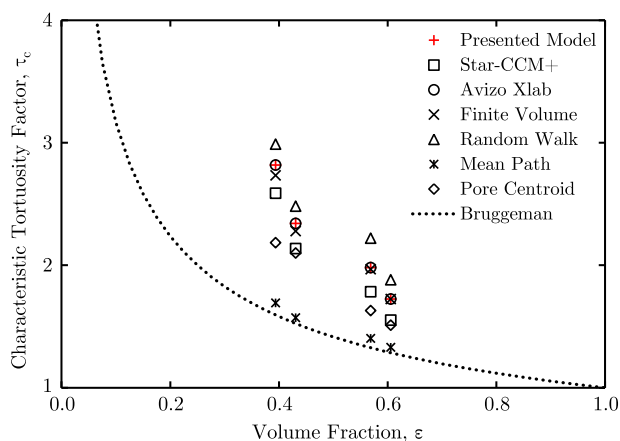


Fig. 8. Comparison of characteristic tortuosity factors calculated using 8 methods for the solid oxide cathode sample reported in [22].

### 3.2. Expected impact

The key impact that *TauFactor* is expected to have on the microstructural analysis community is an increase in consistency within and comparability between studies. As the example above illustrates, the many techniques currently available in the literature cannot be depended on to generate results that would allow for direct comparison across several studies. Furthermore, these techniques tend to fall into one of two categories not conducive to high quality science: Firstly, many are part of expensive propriety software packages, which prevents certain institutions from utilising them. Secondly, either due to commercial sensitivity or academic exclusivity, many codes referred to in the literature are essentially ‘black boxes’, which makes their comparison to nominally similar tools difficult.

In addition, by including a representative volume analysis function, this should encourage a greater degree of rigour in microstructural studies, which may be overlooked in many cases due to the significant increase in work load for the operator. *TauFactor* is at the heart of two studies in the field of energy materials that have already been submitted for publication and is currently being used by energy materials groups at Imperial College London [29] and University College London, as well as by an industrial partner in the same field. Although all users are currently focused on battery and fuel cell materials, a wider audience will be approached following publication.

## 4. Conclusions

The application, *TauFactor*, presented in this article allows fast and simple calculation of the tortuosity factor directly from voxelized microstructural data. Volumes larger than  $10^8$  voxels have been successfully analysed on a single core in just a few hours, which makes the application suitable for characterising data derived from modern tomographic imaging techniques. In addition, volume fractions, surface areas and triple phase boundary densities can all be calculated and representative volume analyses of all metrics can be performed automatically. *TauFactor* also allows the user to specify the dimensions of the voxels and accounts for the anisotropy in all calculations.

It is hoped that by making this application freely available to the community, a greater degree of consistency and rigor will emerge in the analysis of microstructural data.

## Acknowledgements

The authors would like to acknowledge the financial support of the EPSRC Supergen Fuel Cells programme EP/J016454/1 as well as EP/M014045/1. SJC would like to thank Dr. Fred Marquis, Dr. Na Ni and Franca Hoffmann of ICL, Donal Finegan of UCL and Dr. Martin Ebner of ETH Zurich for their contributions to the discussion surrounding this work. AB acknowledges funding from the European Union’s Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 654915. PRS also acknowledges the financial support of the RAEng. Thanks are also due to the many MathWorks file-exchange contributors, especially John Iversen for the `freezeColors` tool.

## Appendix A. Supplementary data

Supplementary material related to this article can be found online at <http://dx.doi.org/10.1016/j.softx.2016.09.002>.

## References

- [1] Sahimi M. Flow and transport in porous media and fractured rock: from classical methods to modern approaches, Vol. 4606. second ed. Weinheim, Germany: John Wiley & Sons; 2011.
- [2] Simpson RL, Wiria FE, Amis AA, Chua CK, Leong KF, Hansen UN, Chandrasekaran M, Lee MW. Development of a 95/5 poly (l-lactide-co-glycolide)/hydroxylapatite and  $\beta$ -tricalcium phosphate scaffold as bone replacement material via selective laser sintering. *J Biomed Mater Res Part B: Appl Biomater* 2008;84(1):17–25.
- [3] Wakao N, Kagei S. Heat and mass transfer in packed beds, Vol. 1. Taylor & Francis; 1982.
- [4] Torquato S, Haslach HW. Random heterogeneous materials: microstructure and macroscopic properties, Vol. 55. Springer; 2002.
- [5] Crank J. The mathematics of diffusion, Vol. 2. Oxford: Clarendon press; 1975.
- [6] Wiener O. Die Theorie Des Mischkörpers Fur Das Feld Der stationären. Erste Abhandlung. Die Mittelwertsätze Fur Kraft, Polarisation Und Energie, Vol. 23. BG Teubner; 1912.
- [7] Shen L, Chen Z. Critical review of the impact of tortuosity on diffusion. *Chem Engrg Sci* 2007;62(14):3748–55.
- [8] Bruggeman DAG. Berechnung verschiedener physikalischer Konstanten von heterogenen Substanzen. I. Dielektrizitätskonstanten und Leitfähigkeiten der Mischkörper aus isotropen Substanzen. *Ann Phys* 1935;416(7): 969–664.
- [9] Lorentz HA. Ueber die Beziehung zwischen der Fortpflanzungsgeschwindigkeit des Lichtes und der Körperdichte. *Wiedemanns Ann* 1880;9(4):641–65.
- [10] Lorenz L. 1880 Ueber die Refraktionsconstante. *Ann Phys* 2005;247(9): 1–35.
- [11] Tjaden B, Cooper SJ, Brett DJL, Kramer D. On the origin and application of the Bruggeman correlation for analysing transport phenomena in electrochemical systems. *Curr Opin Chem Engrg* 2015;1–19.
- [12] Jørgensen PS, Hansen KV, Larsen R, Bowen JR. Geometrical characterization of interconnected phase networks in three dimensions. *J Microscopy* 2011;244(1):45–58.

- [13] Chen-Wiegart Yu-chen Karen, DeMike Ross, Erdonmez Can, Thornton Katsuyo, Barnett Scott A, Wang Jun. Tortuosity characterization of 3D microstructure at nano-scale for energy storage and conversion materials. *J Power Sources* 2014;249:349–56.
- [14] Epstein N. On tortuosity and the tortuosity factor in flow and diffusion through porous media. *Chem Engrg Sci* 1989;44(3):777–9.
- [15] Wilson JR, Kobsiriphat W, Mendoza R, Chen H-Y, Hiller JONM, Miller DJ, Thornton K, Voorhees PW, Adler SB, Barnett SA. Three-dimensional reconstruction of a solid-oxide fuel-cell anode. *Nature Mater* 2006;5(July):541–4.
- [16] Izzo JR, Joshi AS, Grew KN, Chiu WKS, Tkachuk A, Wang SH, Yun W. Nondestructive reconstruction and analysis of SOFC anodes using X-ray computed tomography at Sub-50 nm resolution. *J Electrochem Soc* 2008; 155(5):B504.
- [17] Kishimoto M, Iwai H, Saito M, Yoshida H. Quantitative evaluation of solid oxide fuel cell porous anode microstructure based on focused ion beam and scanning electron microscope technique and prediction of anode overpotentials. *J Power Sources* 2011;196(10):4555–63.
- [18] Joos J, Ender M, Carraro T, Weber A, Ivers-Tiffée E. Representative volume element size for accurate solid oxide fuel cell cathode reconstructions from focused ion beam tomography data. *Electrochimica Acta* 2012;82:268–76.
- [19] Cooper SJ, Eastwood DS, Gelb J, Damblanc G, Brett DJL, Bradley RS, Withers PJ, Lee PD, Marquis AJ, Brandon NP, Shearing PR. Image based modelling of microstructural heterogeneity in  $\text{LiFePO}_4$  electrodes for Li-ion batteries. *J Power Sources* 2014;247:1033–9.
- [20] Shearing PR, Eastwood DS, Bradley RS, Gelb J, Cooper SJ, Tariq F, Lee PD. Exploring electrochemical devices using x-ray microscopy: 3d micro-structure of batteries and fuel cells. *Microsc Anal* 2013;27(2): 19–22.
- [21] Shipman WJ, Nel AL, Chetty D, Miller JD, Lin C-L. The application of machine learning to the problem of classifying voxels in x-ray microtomographic scans of mineralogical samples. In: *Industrial Technology (ICIT), 2013 IEEE international Conference on*. IEEE; 2013. p. 1184–9.
- [22] Cooper SJ, Kishimoto M, Tariq F, Bradley RS, Marquis AJ, Brandon NP, Kilner JA, Shearing PR. Microstructural analysis of an LSCF cathode using in situ tomography and simulation. *ECS Trans* 2013;57(1):2671–8.
- [23] Cooper SJ. Quantifying the transport properties of solid oxide fuel cell electrodes. [Ph.D. thesis]; 2015.
- [24] Ames WF. Numerical methods for partial differential equations. Academic press; 2014.
- [25] Rhazaoui K, Cai Q, Adjiman CS, Brandon NP. Towards the 3D modeling of the effective conductivity of solid oxide fuel cell electrodes: I. Model development. *Chem Engrg Sci* 2013;99(0):161–70.
- [26] Lorensen WE, Cline HE. Marching cubes: A high resolution 3D surface construction algorithm. In: *Proceedings of the 14th annual conference on computer graphics and interactive techniques - SIGGRAPH'87*, vol. 21. ACM; 1987. p. 163–9.
- [27] Rajon DA, Patton PW, Shah AP, Watchman CJ, Bolch WE. Surface area overestimation within three-dimensional digital images and its consequence for skeletal dosimetry. *Med Phys* 2002;29(5):682–93.
- [28] Shearing PR, Bradley RS, Gelb J, Lee SN, Atkinson A, Withers PJ, Brandon NP. Using synchrotron X-ray Nano-CT to characterize SOFC electrode microstructures in three-dimensions at operating temperature. *Electrochem Solid-State Lett* 2011;14(10):B117–20.
- [29] Ni N, Cooper SJ, Williams REA, Kemen N, McComb DW, Skinner SJ. Degradation of  $(\text{La}_{0.6}\text{Sr}_{0.4})_{0.95}(\text{Co}_{0.2}\text{Fe}_{0.8})\text{O}_{3-\delta}$  solid oxide fuel cell cathodes at the nanometre scale and below. *ACS Appl Mater Interfaces* 2016.