SoftwareX 7 (2018) 81–87

Contents lists available at ScienceDirect

SoftwareX

journal homepage: www.elsevier.com/locate/softx

Original software publication

PuMA: the Porous Microstructure Analysis software

Joseph C. Ferguson [a,*], Francesco Panerai [b], Arnaud Borner [a], Nagi N. Mansour [c]

[a] Science and Technology Corp., NASA Ames Research Center, Moffett Field, CA 94035, United States
[b] AMA Inc., NASA Ames Research Center, Moffett Field, CA 94035, United States
[c] NASA Ames Research Center, Moffett Field, CA 94035, United States

ARTICLE INFO

Article history:
Received 3 November 2017
Received in revised form 15 February 2018
Accepted 5 March 2018

Keywords:
Microtomography
Oxidation
Conductivity
Tortuosity factor
Porous media

ABSTRACT

The Porous Microstructure Analysis (PuMA) software has been developed in order to compute effective material properties and perform material response simulations on digitized microstructures of porous media. PuMA is able to import digital three-dimensional images obtained from X-ray microtomography or to generate artificial microstructures. PuMA also provides a module for interactive 3D visualizations. Version 2.1 includes modules to compute porosity, volume fractions, and surface area. Two finite difference Laplace solvers have been implemented to compute the continuum tortuosity factor, effective thermal conductivity, and effective electrical conductivity. A random method has been developed to compute tortuosity factors from the continuum to rarefied regimes. Representative elementary volume analysis can be performed on each property. The software also includes a time-dependent, particle-based model for the oxidation of fibrous materials. PuMA was developed for Linux operating systems and is available as a NASA software under a US & Foreign release.

Code metadata

Current code version	2.1
Permanent link to code/repository used for this code version	https://github.com/ElsevierSoftwareX/SOFTX-D-17-00084
Legal Code License	NASA US & Foreign Release
Code versioning system used	git
Software code languages, tools, and services used	C++, QT, OpenMP, FFTW, LibTiff, OpenGL.
Compilation requirements, operating environments & dependencies	Linux OS, GCC 4.4.7 or above, QT 4.6 or above
If available Link to developer documentation/manual	https://software.nasa.gov/software/ARC-17920-1
Support email for questions	joseph.c.ferguson@nasa.gov

1. Motivation and significance

X-ray microtomography (micro-CT) has become an established tool in material science [1,2], allowing for the non-destructive characterization of microstructures at scales from hundreds of nanometers to centimeters. The resulting digital representations of the three-dimensional material microstructures constitute a high fidelity framework for calculating effective material properties [3,4] and simulating material response [5].

The development of the Porous Microstructure Analysis (PuMA) software is motivated by the possibility to characterize, from tomographic images, microscale properties and response of ablative heat shield materials used on NASA spacecraft. These microscale based computations are used to quantify material properties used in macroscale material response models. [6]

A common challenge when dealing with 3D tomography images is the size of the datasets, often on the order of several billion voxels; a dedicated computational infrastructure is needed to analyze and run simulations based on these images. Several commercial software packages are available for microstructure analysis, including GeoDict [7] (Math2Market, Kaiserslautern, Germany), Avizo XLab [8] (FEI, Hillsboro, Oregon, USA) and Simpleware [9] (Synopsys, Mountain View, California, USA). Avizo XLab and Simplewere are popular image processing and visualization packages that have recently included the option of computing thermal, mechanical and morphological properties of material structures based on X-ray tomography. GeoDict is an extensive material design platform that enables the calculation of properties from either tomography or artificially generated geometries.

* Corresponding author.
E-mail address: joseph.c.ferguson@nasa.gov (J.C. Ferguson).

https://doi.org/10.1016/j.softx.2018.03.001

Open-source and academic software is also witnessing a rapid development. The TauFactor software by Cooper et al. [10] has the capability of computing tortuosity factors for continuum flows in porous media, and to perform representative elementary volume (REV) analysis. Another software, iMorph [11], can compute continuum tortuosity factors and morphological properties based on tomographic images. For strain analysis of in situ X-ray microtomography experiments, another open source software, TomoWarp2 [12], can determine displacement fields through 3D digital image correlation.

The listed commercial and academic software primarily focus on low temperature applications, outside the realm of simulation capabilities needed for entry-systems modeling, where high enthalpies and chemistry play a fundamental role. Additionally, commercial software lacks the flexibility and access to add custom solvers, perform advanced statistics (e.g. robust sensitivity analysis and uncertainty quantification), or couple with other simulation tools.

With this goal in mind, the Porous Microstructure Analysis (PuMA) software was developed at the NASA Ames Research Center in order to calculate material properties and response on large digital microstructures.

The latest version of PuMA is available from NASA under a US & Foreign release and includes the ability to import X-ray microtomography datasets; computationally generate a number of ideal geometries; visualize the materials 3D microstructure; determine effective material properties such as porosity, surface area, thermal conductivity and tortuosity factors; and perform representative elementary volume (REV) analysis for each property. A particle-based microscale oxidation model [5,13] was also implemented and parallelized [14] to simulate high-temperature oxidation of carbon fiber materials.

This paper presents an overview of the PuMA (Version 2.1) software. Section 2 covers the four overarching modules of PuMA's architecture, namely the domain generation module (Section 2.2), the visualization module (Section 2.3), the material properties module (Section 2.4) and the material response module (Section 2.6).

2. Software description

2.1. Software architecture

The functional structure of PuMA is shown in Fig. 1a. It consists of four major modules: domain generation, visualization, material properties, and material response. These are described in detail in Sections 2.2–2.6.

PuMA is written in C++ with a graphical user interface (GUI) built on Qt. A snapshot of the GUI is presented in Fig. 1b, showing the tomography import window. The visualization toolkit is based on OpenGL [15]. The current version of the software (Version 2.1) uses two third party libraries: LibTiff [16] is used to import tiff stacks from microtomography and the LibFFTW [17] is used for Fast Fourier Transform (FFT) operations. Every computationally intensive element in the code has been parallelized with OpenMP [18] for shared memory architectures.

The software is developed for Linux systems, optimized for multi-processor workstations.

2.2. Domain generation

Computational domains in PuMA can be constructed artificially or imported as a 3D TIFF (stack) from microtomography. PuMA does not currently include image filtering capabilities, though this will likely be incorporated into future versions. In our typical tomography pipeline, datasets that need filtering for artifact removal or edge detection enhancement are processed with the Fiji software [19] after tomographic reconstruction. Processed TIFF stacks can be imported in PuMA and a user defined subset can be selected for visualization and analysis. A simple histogram-based thresholding segmentation method is provided, where a single material phase can be isolated using a grayscale thresholding method. Fig. 1b provides an example in which a carbon-fiber microtomography sample is imported for analysis and a typical bimodal grayscale distribution is shown. In many cases, threshold-based segmentation techniques may not be appropriate. For such cases, segmentation can be performed in dedicated software that allow for more advanced methods such as random walk or machine learning algorithms. Examples of software with advanced segmentation features are Fiji [19], scikit-image [20], and Biomedisa [21].

The artificial material generator allows for the creation of random 3D fiber structures, packed sphere beds, periodic foam structures, as well as a series of ideal geometries, such as spheres, cylinders or slabs, used for testing purposes. Examples of artificial domains generated in PuMA are shown in Fig. 2.

2.3. Visualization

The visualization toolkit in PuMA provides an interactive 3D view of the computational domain. The visualization is based on the marching cubes algorithm [22,23], in which an approximate iso-surface is constructed as a collection of triangles. From this set of triangles, a surface rendering is generated using OpenGL, with triangle normals used for shading.

Fig. 3 shows subsets of two synchrotron X-ray tomography images of a carbon fiber material [24] collected at the Advanced Light Source of the Lawrence Berkeley National Laboratory, taken at different resolutions. In Fig. 3, the domain is visualized using a surface rendering in PuMA V2.1, with approximately 450 million triangles in the largest of the visualizations. The sample is shown at multiple scales, demonstrating the level of detail that the tomography is able to capture.

2.4. Material properties

In this section we describe a number of morphological, thermal, and transport properties that PuMA is able to compute from a given material microstructure (either tomographic or artificially generated).

2.4.1. Porosity and volume fraction

Based on the thresholding segmentation method, the volume fraction ϕ_i of a given phase i can be determined as a ratio of the number of voxels, N_i, in the phase to the total number of voxels, N_{tot}, in the domain: $\phi_i = N_i/N_{tot}$. The porosity, ϵ, is defined as the volume fraction of the void phase.

2.4.2. Specific surface area

The specific surface area (S) is defined as the ratio between the total surface area, and the volume, V. To numerically estimate the surface area, two methods [25] were included in the PuMA software: a voxel-based approach, where the total surface area is computed as the sum of exposed solid face areas of each voxel, and an iso-surface based approximation, where the total surface area is determined as a sum of the individual triangle areas in the computed triangulation.

$$S = \frac{\frac{1}{2} l_v^2 \sum_i | \vec{u}_i \times \vec{v}_i |}{V} \tag{1}$$

where $\vec{u}_i$ and $\vec{v}_i$ are the vectors defining triangle i, and l_v is the voxel length. The iso-surface approximation is generated with a marching cubes [22] or marching tetrahedra [26] algorithm. The

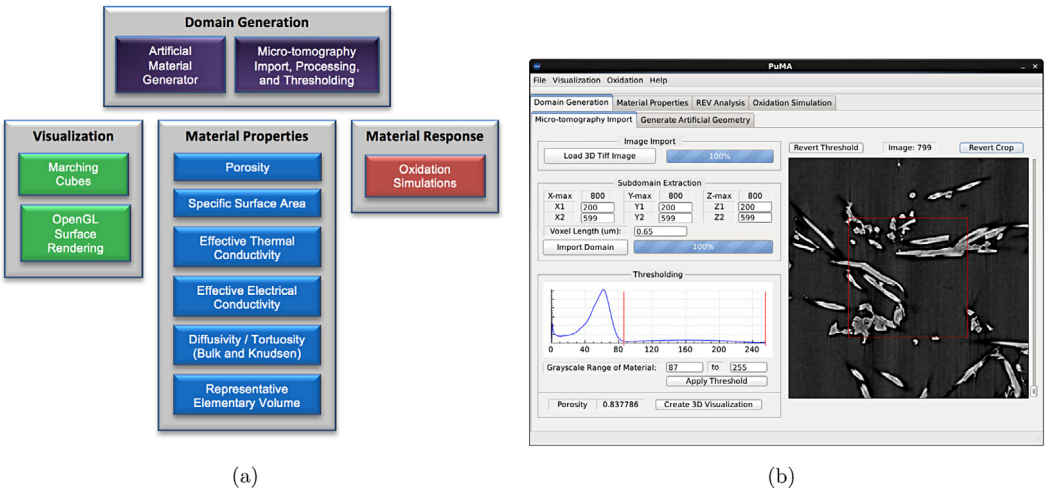

Fig. 1. (a) PuMA V2.1 organizational flow chart (b) Graphical User Interface of PuMA V2.1.

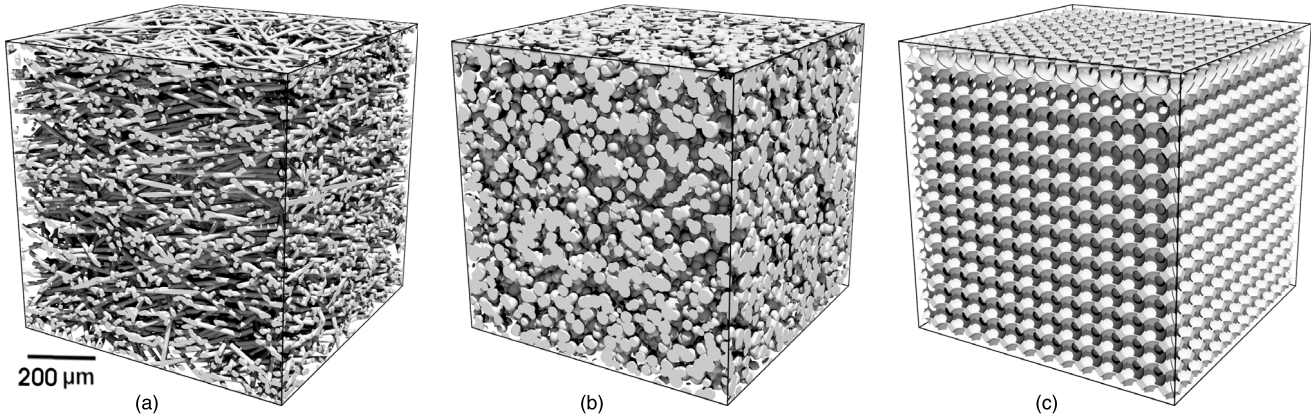

Fig. 2. Example microstructures computationally generated within PuMA V2.1: (a) Random transverse isotropic fiber preform, (b) Packed sphere bed with varying sphere diameters, and (c) Periodic foam structure with uniform void diameter.

marching tetrahedra algorithm avoids the ambiguous cases that exist in the marching cubes algorithm, but produces significantly more triangles.

For ideal geometries with no atomic-level roughness, using the iso-surface to approximate the surface area provides significant improvements over using a cuberille grid. For example, when determining the surface area of an ideal sphere using a cuberille grid, the solution converges to the sum of the 2D projection areas of the sphere onto each of the six faces of the enclosing cube, $SA = 6\pi r^2$, an error of 50%. However, an iso-surface approximation on a continuous grid will converge to the analytical solution, $SA = 4\pi r^2$, as the radius (in voxels) increases [27].

Computing the specific surface area of a real material with surface roughness becomes more ambiguous. If the material is not smooth at the atomic scale, the solution will always be a function of the resolution. It is therefore useful to conceptualize the specific surface area as a function of length scale, or resolution, $S(l)$, where the appropriate choice of length scale and computational method may depend on the specific application.

2.4.3. Effective thermal/electrical conductivity

The effective thermal or electrical conductivity of a material is determined by the material microstructure and the conductivity of its constituents. The steady state diffusion equation, $\nabla\cdot(D\nabla\phi) = 0$, can be expressed for heat conduction as $\nabla \cdot (k\nabla T) = 0$ and electrical conduction as $\nabla \cdot (\sigma\nabla V) = 0$, where D is the local diffusion coefficient, ϕ is the local concentration, k is the local thermal conductivity, T is the local temperature, σ is the local electrical conductivity, and V is the local voltage potential. The equation can be solved three ways in PuMA: using a simple finite difference method, the Explicit-Jump finite difference method by Wiegmann et al. [28,29] or using a random walk method.

The random walk method solves the steady state heat equation using Brownian motion [30] to simulate the diffusion of thermal particles within the domain. The local mean particle velocity, $\bar{v}$, is computed from the local thermal diffusivity, $\alpha_i = (k/\rho c_p)_i$, where k_i is the local thermal conductivity. A reflection probability law is used at the interface between two materials in order to satisfy the temperature continuity condition, and symmetric boundary conditions are used for particles that exit the domain. A mean square displacement method is used to compute the effective thermal diffusion in the material, from which the effective thermal conductivity can be determined.

The two finite difference methods compute the effective conductivity of a composite material by imposing a temperature gradient and solving for the steady state temperature field in the material. From the steady state temperature field, the steady state heat flux can be computed and used to solve for the effective conductivity of the composite material. The simple finite difference method uses Dirichlet (constant value) boundary conditions in the simulation direction, and periodic or reflective boundary conditions in the side directions. The system of equations is solved using the conjugate gradient method [31].

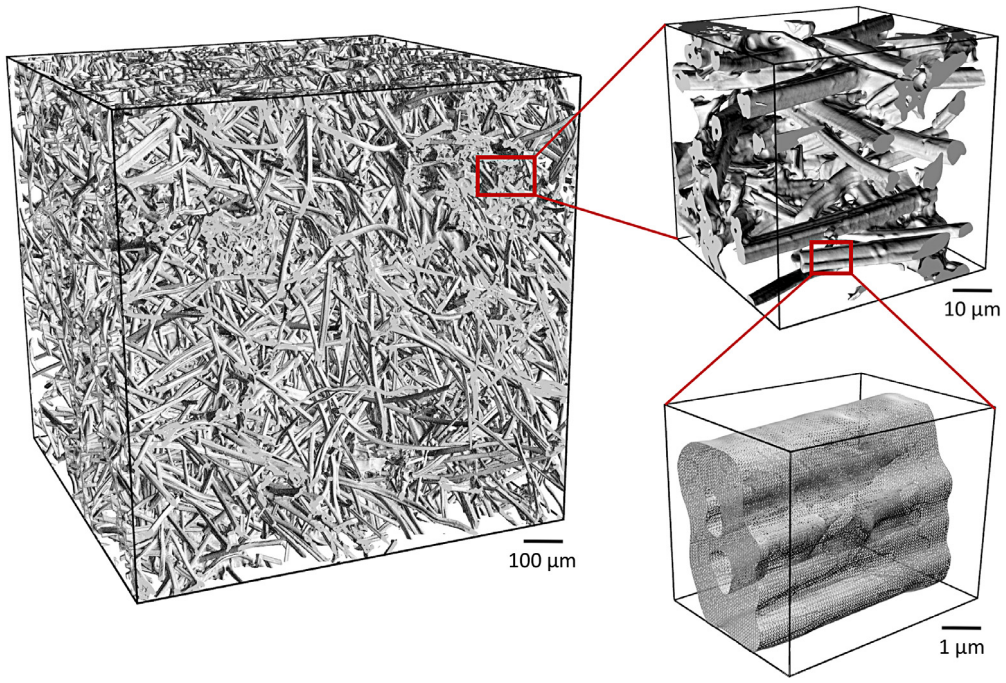

Fig. 3. Surface renderings of FiberForm, a carbon fiber insulator, visualized at three resolutions in PuMA V2.1.

The Explicit Jump method [28,29] is considerably faster than the simple finite difference method. Our implementation uses periodic boundary conditions in the simulation and side directions, which may not be appropriate for some simulations. The linear system of equations is solved using fast Fourier transforms (FFTs) [17] and the bi-conjugate gradient stabilized method (BiCGSTAB) [32].

Each conductivity solver was parallelized for shared memory systems, and verified against a number of simple and complex analytical solutions [33], as detailed in the user manual of PuMA included in the distribution. Two example steady-state temperature fields, from simulations on a fibrous material, are shown in Fig. 4 based on simulations using both of the finite difference solvers.

2.4.4. Tortuosity factor

Physical model. The tortuosity factors, η, of a porous media quantifies the effects of the material microstructure on diffusion. The tortuosity factor is often defined for each direction, to account for material anisotropy.

$$\eta = \epsilon \frac{D_{\text{ref}}}{D_{\text{eff}}} \tag{2}$$

where ϵ is the porosity, D_{eff} is the effective diffusion coefficient in the porous material, and D_{ref} is the reference diffusion coefficient. Here, D_{ref} is defined as the diffusion coefficient through a capillary of diameter l_{D}. It can be expressed using the Bosanquet approximation as [34,35]:

$$D_{\text{ref}} = \frac{1}{3}\bar{v}\left(\frac{\bar{\lambda} l_{\text{D}}}{\bar{\lambda} + l_{\text{D}}}\right) \tag{3}$$

where $\bar{v}$ and $\bar{\lambda}$ are the mean thermal velocity and the mean free path of the gas, respectively. In the continuum, where the Knudsen number $\text{Kn} = \bar{\lambda}/l_{\text{D}} \ll 1$, the Knudsen effects can be ignored and $D_{\text{ref}} = D_{\text{bulk}} = \bar{v}\bar{\lambda}/3$. Using this definition, the bulk tortuosity factor can be determined by computing D_{eff} in the porous material at low Knudsen number and solving for η using Eq. (2).

In our formulation, the tortuosity factor is a purely geometric property [36], rather than a function of the Knudsen number, as is the case in other formulations [35,37]. For this to be true, the characteristic length, l_{D}, in each direction must be determined by running a simulation in the high Knudsen number regime, $\text{Kn} \gg 1$, and computing the length scale as

$$\frac{1}{l_{\text{D}}} = \frac{\bar{v}}{3\eta D_{\text{eff}}} - \frac{1}{\bar{\lambda}} \tag{4}$$

where η is the bulk tortuosity factor computed using continuum assumptions, and D_{eff} is the computed effective diffusion coefficient of the high Knudsen number simulation, defined by the mean thermal velocity, $\bar{v}$, and the mean free path, $\bar{\lambda}$. This relation is based on the Bosanquet approximation, which has shown good agreement for 3D fiber geometries [35], and assumes that a single value for the characteristic length can be used at all Knudsen numbers [36].

Numerical methods. In PuMA, the bulk tortuosity factor can be determined using either of the finite difference methods described in Section 2.4.3, or using a random walk method [35] at a low Knudsen number. Since the heat equation and the diffusion equation take the same form, the method can be easily adapted to compute $D_{\text{eff}}/D_{\text{ref}}$.

The random walk method [35,37] implemented into PuMA is able to solve for D_{eff} in the porous material at any Knudsen number, based on an imposed mean thermal velocity, $\bar{v}$, mean free path, $\bar{\lambda}$, and random number generator seed. Individual particles are given a random velocity vector and free path based on an exponential distribution. Symmetric boundary conditions are used for particles that exit the domain. Surface collisions can be determined based on a cuberille grid or a marching cubes iso-surface approximation [22], with reflections treated as fully diffuse. At low Knudsen numbers, the two collision methods have reasonable convergence; however, at high Knudsen numbers there is up to 20% difference between the two methods for complex 3D geometries. We believe that the iso-surface collision method is more appropriate as it provides a higher fidelity representation of the surface; however, this assumption needs experimental validation. The mean-square displacement method [30,35] is used to determine the effective diffusion coefficient in each direction:

$$D_{\text{eff}_{\text{l}}} = \frac{\langle \xi^2 \rangle_l}{2t} \tag{5}$$

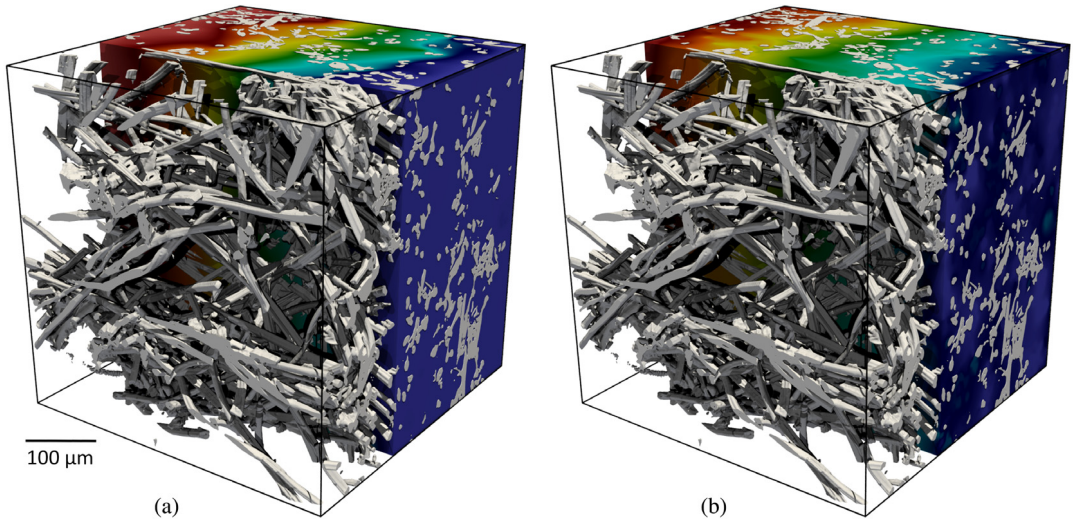

Fig. 4. Steady state temperature fields from simulations on FiberForm, a carbon fiber preform used in NASA's thermal protection systems. (a) Results from the finite difference method, with Dirichlet boundary conditions in the simulation direction and reflective boundary conditions in the side directions. (b) Results from the Explicit Jump solver with periodic boundary conditions in all directions.

where $\langle \xi^2 \rangle$ is the mean square displacement of the particles in the direction, l, and t is the simulation time.

Application. In a typical work flow to determine the tortuosity factors of a material, a low Knudsen number simulation, using either the finite difference or the random walk method, is run in order to determine the bulk tortuosity factor. A high Knudsen number simulation is then run using the random walk method and the computed value of D_{eff} is used in Eq. (4) with the bulk tortuosity factor to determine the diffusion length scale, l_{D}.

Once the two geometrical properties have been determined for each direction (tortuosity factor and length scale), the effective diffusion coefficient in each direction of the porous material for any gas with known mean thermal velocity, $\bar{v}$, and mean free path, $\bar{\lambda}$, can be determined based on the Bosanquet approximation as

$$D_{\text{eff}} = \frac{\epsilon}{3\eta} \bar{v} \left(\frac{\bar{\lambda} l_{\text{D}}}{\bar{\lambda} + l_{\text{D}}} \right) \tag{6}$$

2.4.5. Comparison with other software packages

In this section, material property computations in PuMA are verified against results from one academic software package — TauFactor [10], and one commercial software package — GeoDict [7]. A periodic packed sphere bed was generated in PuMA with a domain size of 400 voxel^3 and porosity $\phi = 0.5$, built from intersecting spheres of radius 10 voxels. A voxel length of 1μm was assigned to the material. Specific surface area, S, estimations are shown in Table 1 for both the cuberille and iso-surface approximation methods. Thermal conductivity computations are tested with a solid conductivity of 10 W/(m·K) and a void conductivity of air at standard temperature and pressure, 0.024 W/(m·K). For both thermal conductivity and tortuosity factor computations, the subscripts 1 and 2 in Table 1 indicate the boundary conditions used: (1) periodic boundary conditions in all directions and (2) Dirichlet boundary conditions in the simulation direction and reflective boundary conditions in the side directions.

A comparison of the Laplace solvers' runtime is shown in Table 2, for continuum tortuosity factor simulations in each of the three softwares tested. Simulations were performed on a 44-core workstation with two Intel Xeon E5-2699 v4 processors (55M Cache, 2.20 GHz). The results shown are based on single-processor simulations, though both PuMA and GeoDict can be run in parallel.

Table 1
Software comparison of material properties computations on a 400 voxel^3 intersecting sphere bed, with 10 voxels radius spheres and a porosity of 0.5.

	PuMA	GeoDict	TauFactor
Porosity, –	0.5	0.5	0.5
S_{cube}, 1/m	1.514e5	1.514e5	1.52e5
S_{iso}, 1/m	1.020e5	1.004e5	–
$k_{\text{eff},1}$, W/(m·K)	1.323	1.323	–
$k_{\text{eff},2}$, W/(m·K)	1.303	1.303	–
η_1, –	1.596	1.596	–
η_2, –	1.602	1.602	1.60

Table 2
Single processor simulation times for steady state diffusion in a 400 voxel^3 intersecting sphere bed with 10 voxels radius spheres and a porosity of 0.5. Simulation times are shown for PuMA, GeoDict [7], and TauFactor [10].

	Periodic	Dirichlet & reflective
PuMA time, s	177.0	890.6
GeoDict time, s	384.6	4039.3
TauFactor time, s	–	3280.0

2.5. Representative elementary volume (REV) analysis

The representative elementary volume (REV) of a material constitutes the size at which a sample of the microstructure is representative of the macroscale structure. In PuMA, this is defined as the sample size for which the standard deviation of a given property drops to below a specified threshold, often taken to be 1–2% of the mean property value. It is important to note that the REV changes depending on the material property of interest.

In PuMA, for a given geometry, the user can specify a sequence of subdomain sizes and select properties of interest. The standard deviation for each size and property is determined, and a power law is used to approximate the REV size for each property. The approximation becomes less accurate the further extrapolation is needed. Fig. 5 shows an example REV analysis performed on a 2000 voxel^3 sample of an artificially generated carbon fiber material. For the tortuosity factor computations shown in Fig. 5, simulations were performed using the random walk method. The normalized standard deviation, defined as the ratio of the standard deviation to the mean value, for each property is shown at the 5 tested domain sizes.

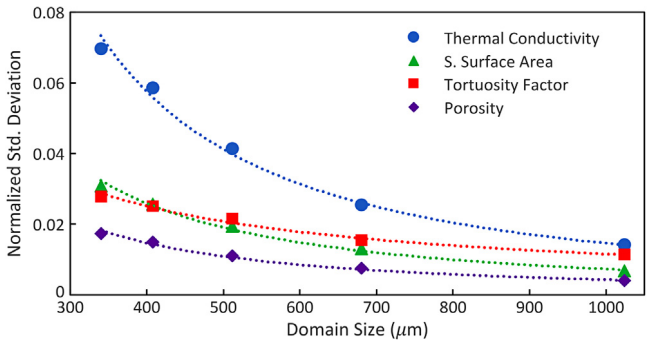

Fig. 5. Representative elementary volume (REV) analysis on a 2000 voxel^3 sample of artificially generated isotropic random fibers. The dashed lines show the power law curve-fits associated with each analyzed property.

For deterministic methods, the standard deviation should converge to zero as the domain size approaches infinity. Conversely, for stochastic methods, such as the random walk tortuosity factor calculation shown in Fig. 5, there also exists uncertainty in the method itself, based on the number of particles used and the total diffusion time. Therefore, as the domain size approaches infinity, the standard deviation of the results will converge to the standard deviation of the method itself; this value would approach zero as the number of particles and total diffusion time approach infinity.

2.6. Material response: Microscale oxidation simulations

PuMA includes an implementation of a particle-based oxidation model first proposed by Lachaud et al. for carbon/carbon materials [13] and later extended in our recent work to highly porous fibrous insulators [5]. The algorithm was parallelized for shared memory architectures, to allow for simulations on large datasets [24]. The model simulates the high temperature decomposition and recession of carbon materials due to oxidation. It is described in Refs. [5,14], with only a broad overview presented here. Reactants are represented as particles, initially contained in a reservoir above the computational domain, called the "buffer zone", which satisfies a Dirichlet boundary condition. Reactants are diffused into the material using a Brownian motion technique. Collisions of reactants with the surface are determined using a linear interpolation method [5] which determines the intersection point between particle trajectories and the approximate material iso-surface. Reactions are simulated with a sticking probability law [13,38]. The oxidation model was parallelized for shared memory systems using OpenMP [18,39].

The implementation has been verified against analytical solutions [5,13] and applied to tomography of real carbon fiber TPS materials [5,24]. Fig. 6 shows an oxidation simulation on a tomography image of FiberForm, a carbon fiber preform used in ablative thermal protection systems [40]. Using the oxidation model, the evolution of the material properties described in Section 2.4 can be studied during the material decomposition.

PuMA also allows for the addition of a second material phase, meant to represent a phenolic matrix used in ablative materials, implemented as either a coating around the fibers or fully dispersed in the voids. By imposing a different density and reactivity to the matrix phase, the method allows for the study of the oxidation of resin infused ablative materials [14].

3. Impact

X-ray tomography is a fast growing technique within the material science community, allowing for non-destructive 3D imaging of materials at multiple scales. The capability is actively applied in many research and technology fields, including aerospace and defense (composites, textiles, thermal insulators, high temperature ceramics), geophysics (rocks, magmas, meteorites), energy storage (battery, fuel cells), and biology (tissues, bones). Heat shield ablation, the focus of our research, is a multiphysics process, that involves mass transport, heat transfer, chemistry and solid mechanics. Since many of these physical processes are similar across diverse disciplines involving porous media, the properties computed in PuMA from digital microstructures can be broadly applied to very different materials. For example, gas diffusion at various Knudsen regimes is of importance in the design of fuel cells, but also applies to the simulation of species transport within an ablative thermal protection system. Thermal conductivity is a fundamental parameter not only for heat shield materials but for any structure operating under thermal gradients. Because of this, PuMA has application potential across very different domains. PuMA has been designed as an open-source and user-friendly environment that provides efficient access to material properties from microstructures, in a context where state-of-the-art commercial tools are not always available to the research community.

At NASA, the software was designed for the modeling of ablative thermal protection systems, but has also been applied to the study of parachute broadcloths and characterization of meteorite materials.

In the field of entry system modeling, PuMA is being coupled to the Porous material Analysis Toolbox based on OpenFOAM (PATO) [6], an open source NASA software for reactive multiphase material response, as well as with the Dakota platform [41] by Sandia National Laboratories for uncertainty quantification, sensitivity analysis, and optimization.

Determining material properties from microstructures (PuMA) and feeding those properties to a macroscale simulation tool (PATO) opens the possibility of computationally designed materials, with optimized microstructures and constituents to a given set of performance requirements.

4. Conclusions

The Porous Microstructure Analysis (PuMA) software is a computational framework for the determination of material properties and material response based on microtomography. Using PuMA, the porosity, specific surface area, thermal conductivity, electrical conductivity, tortuosity factor, and representative elementary volume of a material can be determined based on its X-ray microtomography data. A model for the microscale oxidation is included in order to study the material decomposition at high temperature.

PuMA is released as a NASA software under a US & Foreign release, and is available to the research community as a framework for X-ray microtomography analysis.

The software is actively under development. Modules for computing permeability in the continuum regime and solving the coupled conduction and radiative heat transport with particle methods are currently being implemented. Future releases will include the capability of computing permeability and thermo-mechanical properties of materials.

Acknowledgments

This work was supported by the Entry System Modeling project (M.J. Wright project manager, M.D. Barnhardt principal investigator) as part of the NASA Game Changing Development program. The authors would like to acknowledge T. Sandstrom, C. Henze, D. Ellsworth, and B. Nelson (NASA Ames Research Center) for useful discussions on the PuMA architecture and implementation and J. Lachaud for his contributions to the development of the oxidation model in PuMA. J. Meurisse and J. Thornton (Science and Technology Corp. at NASA Ames Research Center) are acknowledged for

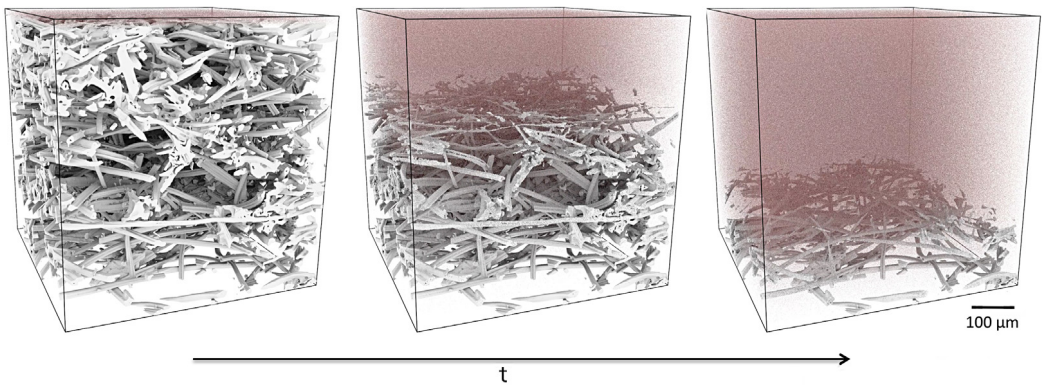

Fig. 6. Microscale oxidation simulation at moderate temperature on FiberForm, a carbon fiber preform material.

their review of the manuscript and useful discussions. The NASA Ames microtomography effort is performed in collaboration with the Lawrence Berkeley National Laboratory Advanced Light Source. D.Y. Parkinson, H.S. Barnard and A.A. MacDowell are acknowledged for their assistance with tomography measurements.

References

[1] Maire E. X-ray tomography applied to the characterization of highly porous materials. Ann Rev Mater Res 2012;42:163–78.
[2] Hsieh J. Computed tomography: principles, design, artifacts, and recent advances. 4th edition. SPIE Press; 2009.
[3] Arns CH, Knackstedt MA, Pinczewski MV, Lindquist W. Accurate estimation of transport properties from microtomographic images. Geophys Res Lett 2001;28(17):3361–4.
[4] Maire E, Buffiere J-Y, Salvo L, Blandin JJ, Ludwig W, Letang J. On the application of X-ray microtomography in the field of materials science. Adv Energy Mater 2001;3(8):539–46.
[5] Ferguson JC, Panerai F, Lachaud J, Martin A, Bailey SC, Mansour NN. Modeling the oxidation of low-density carbon fiber material based on micro-tomography. Carbon 2016;96:57–65.
[6] Lachaud J, Mansour NN. Porous material analysis toolbox based on OpenFoam and applications. J Thermophys Heat Transfer 2014;28(2):191–202.
[7] GeoDict. math2Market. Kaiserslautern, Germany; 2017. URL http://www.geodict.com.
[8] Avizo x-lab. fei. Oregon, USA; 2017. URL http://www.fei.com/software/avizo-xlab-ds.pdf .
[9] Simpleware. Synopsys. California, USA; 2017. URL http://www.simpleware.com.
[10] Cooper S, Bertei A, Shearing P, Kilner J, Brandon N. TauFactor: An open-source application for calculating tortuosity factors from tomographic data. SoftwareX 2016;5:203–10.
[11] Brun E, Vicente J, Topin F, Occelli R. IMorph: A 3D morphological tool to fully analyse all kind of cellular materials. Cell Metals Struct Funct Appl 2008.
[12] Tudisco E, Andó E, Cailletaud R, Hall SA. TomoWarp2: A local digital volume correlation code. SoftwareX 2017;6(Suppl. C):267–70.
[13] Lachaud J, Vignoles G. A brownian motion technique to simulate gasification and its application to C/C composite ablation. Comput Mater Sci 2009;44(6):1034–41.
[14] Ferguson JC, Panerai F, Lachaud J, Mansour NN. Theoretical study on the micro-scale oxidation of resin-infused carbon ablators. Carbon 2017;121:552–62.
[15] Woo M, Neider J, Davis T, Shreiner D. OpenGL programming guide: the official guide to learning OpenGL, version 1.2. Addison-Wesley Longman Publishing Co., Inc.; 1999.
[16] Leffler S. LibTIFF–TIFF Library and Utilities, URL http://www.libtiff.org/v3.6.1.html; 2003.
[17] Frigo M, Johnson SG. FFTW: An adaptive software architecture for the FFT. In: Acoustics, Speech and Signal Processing, 1998. Proceedings of the 1998 IEEE International Conference on, vol. 3. IEEE; 1998. p. 1381–4.
[18] OpenMP Architecture Review Board. OpenMP Application Program Interface Version 3.0; 2008. URL http://www.openmp.org/mp-documents/spec30.pdf.
[19] Schindelin J, Arganda-Carreras I, Frise E, Kaynig V, Longair M, Pietzsch T, et al. Fiji: an open-source platform for biological-image analysis. Nat Methods 2012;9(7):676–82.
[20] Van der Walt S, Schönberger JL, Nunez-Iglesias J, Boulogne F, Warner JD, Yager N, et al. scikit-image: image processing in Python. PeerJ 2014;2:e453.
[21] Lösel P, Heuveline V. Enhancing a diffusion algorithm for 4D image segmentation using local information. In: Medical imaging 2016: Image processing, vol. 9784. International Society for Optics and Photonics; 2016. p. 97842L.
[22] Lorensen WE, Cline HE. Marching cubes: A high resolution 3D surface construction algorithm. SIGGRAPH Comput Graph 1987;21(4):163–9.
[23] Chernyaev EV. Marching cubes 33: Construction of topologically correct isosurfaces, Institute for High Energy Physics, Moscow, Russia, Report CN/95-17, 42; 1995.
[24] Panerai F, Ferguson JC, Lachaud J, Martin A, Gasch MJ, Mansour NN. Micro-tomography based analysis of thermal conductivity, diffusivity and oxidation behavior of rigid and flexible fibrous insulators. Int J Heat Mass Transfer 2017;108:801–11.
[25] Kenmochi Y, Klette R. Surface area estimation for digitized regular solids. In: Vision geometry IX, vol. 4117. International Society for Optics and Photonics; 2000. p. 100–12.
[26] Treece GM, Prager RW, Gee AH. Regularised marching tetrahedra: improved iso-surface extraction. Comput Graph 1999;23(4):583–98.
[27] Vignoles GL, Donias M, Mulat C, Germain C, Delesse J-F. Simplified marching cubes: An efficient discretization scheme for simulations of deposition/ablation in complex media. Comput Mater Sci 2011;50(3):893–902.
[28] Wiegmann A, Zemitis A. EJ-HEAT: A fast explicit jump harmonic averaging solver for the effective heat conductivity of composite materials; 2006.
[29] Wiegmann A, Bube KP. The explicit-jump immersed interface method: finite difference methods for PDEs with piecewise smooth solutions. SIAM J Numer Anal 2000;37(3):827–62.
[30] Einstein A. Zur theorie der brownschen bewegung. Ann Phys 1906;324:371–81.
[31] Hestenes MR, Stiefel E. Methods of conjugate gradients for solving linear systems. J. Res. Natl Bureau of Standards 1952;49(6).
[32] Van der Vorst HA. Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems. SIAM J Sci Stat Comput 1992;13(2):631–44.
[33] Progelhof R, Throne J, Ruetsch R. Methods for predicting the thermal conductivity of composite systems: a review. Polym Eng Sci 1976;16(9):615–25.
[34] Pollard W, Present RD. On gaseous self-diffusion in long capillary tubes. Phys Rev 1948;73(7):762.
[35] Tomadakis MM, Sotirchos SV. Ordinary and transition regime diffusion in random fiber structures. AIChE J 1993;39(3):397–412.
[36] Zalc JM, Reyes SC, Iglesia E. The effects of diffusion mechanism and void structure on transport rates and tortuosity factors in complex porous structures. Chem Eng Sci 2004;59(14):2947–60.
[37] Tomadakis MM, Sotirchos SV. Knudsen diffusivities and properties of structures of unidirectional fibers. AIChE J 1991;37(8):1175–86.
[38] Salles J, Thovert JF, Adler PM. Deposition in porous media and clogging. Chem Eng Sci 1993;48:2839–58.
[39] Dagum L, Menon R. OpenMP: an industry standard API for shared-memory programming. IEEE Comput Sci Eng 1998;5(1):46–55.
[40] Mansour NN, Panerai F, Martin A, Parkinson DY, MacDowell AA, Fast T, et al., A new approach to light-weight ablators analysis: from micro-tomography measurements to statistical analysis and modeling, In: Proc. 44th AIAA thermophysics conference, American Institute of Aeronautics and Astronautics, 2013–2768, AIAA.
[41] Adams BM, Bohnhoff W, Dalbey K, Eddy J, Eldred M, Gay D et al. DAKOTA, a multilevel parallel object-oriented framework for design optimization, parameter estimation, uncertainty quantification, and sensitivity analysis: Version 5.0 user's manual. Sandia National Laboratories, Tech. Rep. SAND2010-2183; 2009.