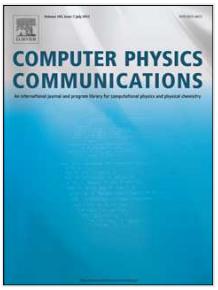
Open-source computational model of a solid oxide fuel cell

Beale, Steven B.; Choi, Hae-Won; Pharoah, Jon G.; Roth, Helmut K.; Jasak, Hrvoje; Jeon, Dong Hyup

Publisher's version / Version de l'éditeur:
https://doi.org/10.1016/j.cpc.2015.10.007
Computer Physics Communications, 2015

Accepted Manuscript



Open-source computational model of a solid oxide fuel cell

Steven B. Beale, Hae-Won Choi, Jon G. Pharoah, Helmut K. Roth, Hrvoje Jasak, Dong Hyup Jeon

PII: S0010-4655(15)00387-2
DOI: http://dx.doi.org/10.1016/j.cpc.2015.10.007
Reference: COMPHY 5777

To appear in: *Computer Physics Communications*

Received date: 2 February 2015
Revised date: 21 September 2015
Accepted date: 13 October 2015

Please cite this article as: S.B. Beale, H.-W. Choi, J.G. Pharoah, H.K. Roth, H. Jasak, D.H. Jeon, Open-source computational model of a solid oxide fuel cell, *Computer Physics Communications* (2015), http://dx.doi.org/10.1016/j.cpc.2015.10.007

Open-source Computational Model of a Solid Oxide Fuel Cell

Steven B. Beale[a,b,c,d], Hae-Won Choi[b], Jon G. Pharoah[b,c], Helmut K. Roth[d], Hrvoje Jasak[e,f], Dong Hyup Jeon[d,1]

[a] *Institute of Energy and Climate Research, IEK-3, Forschungszentrum Jülich GmbH, Jülich 52425, Germany*
[b] *Queen's-RMC Fuel Cell Research Centre, 945 Princess St., 2nd floor, Kingston ON K7L 5L9, Canada*
[c] *Mechanical and Materials Engineering, Queen's University, Kingston ON K7L 3N6, Canada*
[d] *National Research Council, 1200 Montreal Road, Ottawa ON K1A 0R6, Canada*
[e] *Mechanical Engineering and Naval Architecture, University of Zagreb, Ivana Lucica 5, 10000 Zagreb, Croatia*
[f] *Wikki Ltd., Unit 459, Southbank House, Black Prince Road, London SE1 7SJ, United Kingdom*

Abstract

The solid oxide fuel cell is an electro-chemical device which converts chemical energy into electricity and heat. To compete in today's market, design improvements, in terms of performance and life cycle, are required. Numerical prototypes can accelerate design and development progress. In this programme of research, a three-dimensional solid oxide fuel cell prototype, `openFuelCell`, based on open-source computational fluid dynamics software was developed and applied to a single cell. Transport phenomena, combined with the solution to the local Nernst equation for the open-circuit potential, as well as the Kirchhoff-Ohm relationship for the local current density, allow local electro-chemistry, fluid flow, multi-component species transport, and multi-region thermal analysis to be considered. The detailed physicochemical hydrodynamics included porous electrodes and electro-chemical effects. The `openFuelCell` program is developed in an object-oriented open-source C^{++} library. The code is available at `http://openfuelcell.sourceforge.net/`. The paper also describes domain decomposition techniques considered in the context of highly efficient parallel programming.

Keywords: solid oxide fuel cell, computational fluid dynamics, physicochemical hydrodynamics, electro-chemistry, heat and mass transfer.

1. Introduction

A solid oxide fuel cell (SOFC) is an electro-chemical device used for the conversion of the chemical energy of hydrogen-rich or hydrocarbon fuels into electricity, with heat as a byproduct. The SOFC exploits the oxygen-ion conducting property of a solid ceramic, Yttria-stabilised Zirconia

*Corresponding author. Tel: +49-2461 61-8856; Fax: +49-2461 61-6695.
Email addresses: `s.beale@fz-juelich.de` (Steven B. Beale), `haewon1972@gmail.com` (Hae-Won Choi), `pharoah@me.queensu.ca` (Jon G. Pharoah), `hrvoje.jasak@fsb.hr` (Hrvoje Jasak)
[1]Present address: Mechanical System Engineering, Dongguk University, Gyeongju 780-714, Republic of Korea

or similar, as an electrolyte. It has gained attention as a candidate for efficient, clean, distributed power generation [1]. SOFCs offer a number of advantages over existing power plants; high-power density, electrical efficiency, no moving parts, low emission, and fuel flexibility. However, the cell typically operates at $600-1000^{\circ}$C, and the high operating temperature imposes rigorous constraints on the material properties of the cell components to obviate premature mechanical degradation and failure. The development of suitable low-cost, durable materials is key to the adoption of the technology. Because of the very high temperatures involved, and in spite of various efforts, there are virtually no reliable experimental data available on e.g., local current density in operating SOFCs. Under the circumstances, a good model is therefore invaluable.

The operation of a SOFC involves the simultaneous transport of mass, momentum, species, and charge, coupled with electro-chemical reactions. The performance of a SOFC is directly related to these coupled transport phenomena. To comprehend these processes and conduct parametric studies on cell performance, numerical simulations based on computational fluid dynamics (CFD) models [2–9] have been considered. CFD models reduce, but do not eliminate, the amount of experimental data needed for cell development [3, 10].

Various commercial software programs, for example, PHOENICS [2, 11], Fluent [2, 8, 12–14], Star-CD [15, 16] and COMSOL Multiphysics [17] have been adapted to simulate SOFC applications. More recently, the development of open-source software such as `OpenFOAM` (Open Field Operation and Manipulation) [18] has stimulated activities in SOFC application/development by the present group [19–25]. and others [26–30]. However until now, there are no open source code repositories in existence, capable of performing comprehensive performance calculations for SOFCs. Flexible, well-written, open-source code can form a framework for research and development, which is not possible with commercial 'black-box' software with associated license-fees, limited programmability and extensibility by the scientist end-user.

The present writers are developing a multi-physics, multi-scale code suite for prototyping fuel cell designs. At the heart of the activity is the cell-level SOFC model. In addition, micro-scale and large-scale stack models are under development using similar program design techniques. The cell model is multi-regional, encompassing all salient components of a SOFC cell: interconnects, gas channels, actively-reacting and passively-supporting components of the porous anode and cathode layers and also the solid electrolyte. The CFD model considers fluid flow, multi-component species

transport, and conjugate heat transfer. The transport model is coupled with the solution of a Nernst Equation for the ideal or maximum potential minus electrical losses computed by the application of Kirchoff's second law to ensure a conservative electric-field combined with Ohm's law [20] based on a user-selectable area specific resistance (ASR) [21] to obtain the local current density. The detailed three-dimensional (3-D) CFD model is conservative, computationally efficient, fully parallelised, and within the optimised open-source library, `OpenFOAM` [18]. Detailed computational physics, thermo-chemical, and electro-chemical transport modules account for porous electrode effects and electro-chemical reactions.

From the point of view of implementation and software organisation, SOFC modelling poses several challenges. The root of the problem is the presence of multiple regions within the fuel cell; each region being governed by a different set of physics equations. In the fuel channels, the Navier-Stokes equations are solved with a set of species equations; the Navier-Stokes system in the air channel will carry a similar but different species set. In the solid interconnect and electrolyte components only the equation of energy requires to be solved, but this needs to be coupled with the energy solutions in the corresponding air and fuel channels. An interesting problem arises in the porous zones in the electrodes. For both heat transfer and electric charge transport, these regions need to be handled as intermingling solid and fluid regions. They also act as a resistance (drag) on fluid momentum. Depending on the level of approximation, the SOFC geometry is decomposed into a number of component regions: fuel/air channels, electrolyte, interconnect, porous layers, etc. Different levels of model complexity will decide whether the regions are abutting or overlapping. The initial `openFuelCell` development was based on a simplified model prototype for which calculations using two different commercial codes PHOENICS and Fluent [2, 11] had already been obtained. Once satisfactory agreement was obtained with the previous results; the present code was substantially extended in terms of capabilities, as described further below. A detailed fuel cell or stack model will cover a significant volume in space and require resolution of the flow and species concentration gradients, necessarily increasing the total computational-mesh size to millions of mesh cells. For efficient simulations, use of high-performance parallel computers is essential, further complicating the issues of software design. Two different implementation strategies which were considered in the development are described below. Keuler and co-workers reported on successfully running the code on 960 cores on the Jülich Aachen Research Alliance (JARA)

computing facility [31] with near linear performance.

Validation and verification of mathematical models are important subjects. The International Energy Agency SOFC benchmark [32] was developed by teams of researchers from 5 European countries as a reference for performance calculations. Le at el. [33] undertook a comprehensive comparison of results obtained with `openFuelCell`, with the original IEA benchmark, as well as with similar more recent performance calculations [34–37]. Beale et al. [21] contains a comparison between `openFuelCell` calculations and experimental work for the Jülich Mark F geometry, for which extensive experimental data [38–40] have been gathered, and is one of the only SOFC stacks which have operated for more than 60,000 hours. Figure 1 illustrates the plates, manifold and channel assembly of this geometry. The primary focus of this paper is on computational methods and techniques and the underlying logic of the `openFuelCell` code suite, rather than detailed presentation of the results of such calculations for any specific SOFC design, which the reader will find in the above references.

Figure 2 is a schematic illustration of the 'core' region of a planar SOFC. For convenience the manifolds have been removed. The unit is composed of a positive-electrolyte-negative (PEN) electrode assembly, interconnects, and gas channels. The PEN assembly consists of five distinct layers: (1) anode substrate layer (ASL), (2) anode functional layer (AFL), (3) electrolyte, (4) cathode functional layer (CFL), and (5) cathode current collector layer (CCCL). The gas channels provide fuel and oxidant, necessary for the reaction, and are embedded within the rectangular cross-section of the metallic interconnects. The fuel and air channels are connected to inlet and exhaust manifolds (not shown). It can be seen that the domain has been tessellated with a rectilinear computational grid which is uniform in the plane of the PEN.

2. Governing Equations

2.1. Fluid flow and mass transfer in gas channels and porous transport layers

The physicochemical hydrodynamic phenomena are presumed to be governed by the following set of coupled equations. The fuel cell was considered to be in steady-state, though transience can be added to the model equations with ease. In the gas channels and porous electrodes, the so-called

primitive or divergence form of the transport equations are solved within volumetric region Ω;

$$\operatorname{div}(\rho\mathbf{u}) = 0, \tag{1}$$

$$\operatorname{div}(\rho\mathbf{u}\mathbf{u}) = -\operatorname{grad} p + \operatorname{div}(\mu \operatorname{grad}\mathbf{u}) - \frac{\mu\mathbf{u}}{\kappa_D}, \tag{2}$$

where $\mathbf{u}$ denotes local velocity (m/s) (superficial or filter velocity in the porous transport layers), p is pressure (Pa), and μ the mixture viscosity (Pa·s) is computed according to Wilke [41] from the individual species viscosities [42]. κ_D is permeability (m^2), and ρ is mixture density ($\mathrm{kg\,m}^3$),

$$\rho = \frac{p}{R_g T}\sum x_i M_i = \frac{p}{R_g T}/\sum \frac{y_j}{M_j}, \tag{3}$$

The molecular weight of species i is given as M_i (kg/kmol), the molar and mass fractions of species i are denoted by x_i and y_i, respectively, and the universal gas constant, $R_g = 8.3124\,\mathrm{J/K\cdot mol}$. The set of scalar species equations

$$\operatorname{div}(\rho\mathbf{u}y_i) = \operatorname{div}\left(\rho D_i^{\mathrm{eff}} \operatorname{grad} y_i\right), \tag{4}$$

is solved for $i = 1, 2, .., n-1$, the n^{th} species, being considered a 'shadow' species, is obtained algebraically, $y_n = 1 - \sum_{i=1}^{n-1} y_i$. The effective diffusivity, D_i^{eff} (m^2/s), may be computed according to a number of run-time selectable options, as discussed further below.

At the boundaries, $\partial\Omega$, of the electrodes, which are treated as surfaces, the normal mass flux, $\dot{m}'' = \rho\mathbf{u}$ is just the sum of the individual reactant/product species fluxes,

$$\dot{m}'' = \sum_i \dot{m}_i''. \tag{5}$$

These are computed by Faraday's law,

$$\dot{m}_i'' = \pm\frac{jM_i}{n_i F}, \tag{6}$$

where $F = 96\,485\,\mathrm{C/mol}$ is Faraday's constant, n_i is the charge number, and by convention the sign of Eq.(6) is positive for products and negative for reactants. A no slip velocity condition is

applied at the air and fuel channel walls for the tangential velocities.

The boundary fluxes for the individual species diffusion equations on $\partial\Omega$ are obtained from the identity

$$\dot{m}'' - \frac{\partial}{\partial n}\left(\rho D_i^{\text{eff}} y_i\right) = \dot{m}_i'', \tag{7}$$

which is easily re-arranged to obtain an expression for the normal gradient $\partial y_i/\partial n$ or a transferred substance state boundary value, $y_i = \dot{m}_i''/\dot{m}''$ on $\partial\Omega$.

2.1.1. Diffusion coefficients

Depending on the desired level of complexity and the zone of interest, various closure schemes are possible for mass transfer. The binary gas diffusion coefficient (D_{ij}) is calculated by means of the correlation of Fuller, Schettler and Giddings [42–45]:

$$D_{ij} = \frac{10^{-7} T^{1.75} \left(1/M_i + 1/M_j\right)^{1/2}}{p\left(V_i^{1/3} + V_j^{1/3}\right)^2}, \tag{8}$$

where T is absolute temperature (K), p is pressure (atm), and V_i and V_j are diffusion volumes (cm^3) for species i and j, respectively.

In porous transport layers, diffusion is reduced relative to an open channel by the ratio of the porosity, ϵ, and tortuosity factors, τ [46], with the binary and Knudsen diffusion coefficients combined harmonically [47, 48],

$$D_{ij}^{\text{eff}} = \frac{\epsilon}{\tau}\left(\frac{1}{D_{ij}} + \frac{1}{D_{i\text{Kn}}}\right)^{-1}, \tag{9}$$

The Knudsen diffusion coefficient $D_{i\text{Kn}}$ is defined by

$$D_{i\text{Kn}} = \frac{1}{3} d_{\text{cl}} \left\langle \nu_T \right\rangle_i, \tag{10}$$

$$\left\langle \nu_T \right\rangle_i = \sqrt{\frac{8 R_g T}{\pi M_i}}, \tag{11}$$

where d_{cl} denotes the characteristic length of the pore domain (m), $\left\langle \nu_T \right\rangle_i$ represents the mean thermal molecular velocity of gas species (m/s).

The determination of d_{cl} for porous media is a matter of importance when the Knudsen number, $\text{Kn} > 0.1$. For example, d_{cl} may be associated with a pore diameter, d_{pore} [49, 50], which in turn may be expressed in terms of particle diameter. Similarly a constant tortuosity factor may be selected [51, 52] or this may be considered to vary as $\tau = \epsilon^{-1/2}$ from the Bruggeman relation [53], namely, $D^{\text{eff}}/D^{\text{ref}} = \epsilon/\tau = \epsilon^{1.5}$.

For multi-component diffusion, the mixture diffusion coefficient may be estimated from the individual binary coefficient or the effective binary coefficients by means of, say, Wilke's approach [41]

$$D_i = (1 - x_i) \left(\sum_{j \neq i} \frac{x_j}{D_{ij}^{\text{eff}}} \right)^{-1}, \tag{12}$$

in the open channels, with the effective diffusion coefficent, D_{ij}^{eff}, supplanting the binary coefficient, D_{ij}, in porous media in Eq. (12). The various diffusivity models are defined as instances of an abstract base class selected from a virtual base class heirachy (family) at run-time and parameterised with a `dictionary` of tagged values in the user's `case` directory, which allows the user to select appropriate diffusion coefficient models, according to region and level-of-complexity.

2.2. Electro-chemical reactions

The PEN structure is electro-chemically active and includes the thickness of the electrolyte and the two adjacent active layers. Although the reactions are distributed throughout the active layers of the electrodes, these are tacitly assumed to take place at a single interface between the given electrode and electrolyte. The local open circuit, or ideal potential, is presumed to be given by the Nernst equation [54]. For the reaction

$$\sum_i a_i R_i \rightarrow \sum_j b_j P_j, \tag{13}$$

having reactants R_i with stoichiometric coefficients a_i and products P_j with stoichiometric coefficients b_j, the ideal or Nernst potential, E (V), is calculated as,

$$E = E^0 + \frac{R_g T}{zF} \ln Q, \tag{14}$$

where E_0 (V) is a standard electrode potential and,

$$Q = \frac{\prod_j [P_j]^{b_j}}{\prod_i [R_i]^{a_i}}, \tag{15}$$

is the equilibrium constant, and $[P_j]$, $[R_i]$ denotes the molar concentration of P_j, R_i. The form of Eq. (14) is quite general. The air mixture of oxygen and nitrogen is illustrated in a `speciesList dictionary` shown in Table 1, with reactants, products, and inert substances 'tagged' as -1, 1, 0, with thermodynamic properties as shown in the figure.

The cell voltage, V, is equal to the theoretical open circuit voltage, reduced by the anode and cathode overpotentials and the product of the ohmic resistance and local current density in the electrolyte, as follows:

$$V = E - \eta_{\mathrm{a}} - \eta_{\mathrm{c}} - jR. \tag{16}$$

Equation (16) is obtained by simply combining Kirchhoff's second law for the circuit (including the load), with Ohm's law for the electrolyte, and is henceforth referred to here as the 'Kirchhoff-Ohm relationship'. Equation (16) provides an explicit expression for the local current density, j (A/m^2), as a function of the prescribed external voltage, V, and local Nernst potential, E. In a high-temperature SOFC, activation losses are typically quite small, and these may simply be lumped as an overall area specific resistance, see for example, [11]. The user may easily code implementations for the resistance as a function of temperature [55] via a run-time selection library. If it is desired to explicitly compute the local activation terms, Butler-Volmer type equations [54] may be solved iteratively to obtain η_{a} and η_{c} as a function of j, based for example on the experimental data of Leonide et al. [56] using a numerical root-solver.

The heat of reaction, which is not delivered as energy to the load, generates a volumetric heat source within the PEN assembly. This is computed as:

$$\dot{q}''' = \left(-\frac{1}{2F} \Delta H(T) - V \right) \frac{j}{\delta h_E}, \tag{17}$$

where the enthalpy of formation, ΔH (J/mol), is by convention positive for an exothermic reaction. This is obtained from expressions for the heat capacity [42, 57]. Outside the electrolyte, heat sources

are presumed zero.

2.3. Heat transfer

The energy equation is solved for temperature, T, in all solid and fluid regions, in the convective form,

$$\rho c_p \mathbf{u} \cdot \operatorname{grad} T = \operatorname{div} (k \operatorname{grad} T) + \dot{q}''', \tag{18}$$

where k is thermal conductivity (W/m·K) computed according to [58], and c_p specific heat (J/kg·K). The term $\dot{q}'''$ is due to electro-chemical and Joule heating. The continuity boundary condition for the two electrode walls is obtained from the mass flux divided by the mixture density, the former being computed according to Eq. (5). Adiabatic conditions are applied at the outer walls of the entire cell. Radiation heat transfer [59] is not considered in the present study.

3. Numerical Implementation

The open source software suite `OpenFOAM` version 1.6-ext was selected as the development platform for the multi-physics and multi-scale calculation. The open source implementation of the SOFC model is in a form which allows non-expert programmers to inspect, modify and extend the code. This is achieved by re-using mesh handling, numerical discretisation and high performance computing support of the underlying `OpenFOAM` library. The SOFC model implementation itself consists of approximately 10,000 lines of code, as opposed to the 1.3 million lines of the base libraries in `OpenFOAM`. `OpenFOAM` provides run time selection of boundary conditions, operator discretization schemes, and linear system solvers. Users are free to modify or extend the code, as desired. Developers are not hampered by inaccessible proprietary source code. The set of governing equations is solved by a finite-volume method written in the object-oriented C++ programming language. The implementation is modular; individual components of the cell model (eg. electro-chemistry or thermodynamics) can be replaced while preserving overall structure of the solver. The code provides for modern automatic meshing techniques, often necessary in order to properly capture solution features in real SOFC design. This is achieved via unstructured polyhedral mesh support of underlying `OpenFOAM` discretisation components. A useful feature of `OpenFOAM` is the provision of a full set of implicit finite volume discretisation operators and associated linear system solver

classes, allowing transparent representation of partial differential equations in the code. An overriding principle in `OpenFOAM` design is to group families of objects performing the same function (in different ways) under a common interface. In this manner, the consuming class does not need to know about specific options on performing the function. In C++, this is achieved via pure virtual functions which define the interface and concrete derived classes which implement the functionality, via virtual functions. This design pattern is used in all places where multiple choice of models and user-defined extensions exist, in place of if-then-else model selection of procedural programming. Examples of such families are numerous; from boundary conditions, gradient discretisation schemes, linear equations solvers, to turbulence models or thermophysical property libraries. Diffusivity models in the `openFuelCell` are a good example of such a family and a class hierarchy encoding it. The selection of linear solvers and their parameters are also chosen at run time. For the calculations reported here, symmetric linear systems were solved using conjugate gradient with incomplete Cholesky pre-conditioning (ICCG) [60] and asymmetric systems using bi-conjugate gradient solver Bi-CGStab [61]. The `openFuelCell` code was first validated by comparison with results previously-obtained with two commercial codes, PHOENICS and Fluent, using the same geometry and operating conditions as in [2, 62], for which a number of engineering assumptions; single cell, constant properties, negligible activation overpotential, idealised boundary conditions (i.e., no manifolds), were presumed corresponding to the original work. Once satisfactory agreement was obtained with the earlier results, the code was subsequently refined so as to include more advanced physics, multiple fuel cells (stack geometry with manifolds), variable properties, kinetics. The user is not required to write any code (as with commercial codes) to activate these features, but is free to do so, if desired.

3.1. Global domain decomposition

Volumetric equations in all regions are solved using a cell-centred second order accurate Finite Volume Method (FVM) with polyhedral cell support. `OpenFOAM` introduces the concept of a `mesh`, as a set of consecutively numbered cells covering a region of space, bounded by `patches`. The mesh supports scalar and vector fields and, via discretisation operators, allows a single set of equations to be solved over that region. Boundary `patches` may be either *regular*, representing standard FVM boundary conditions, or *coupled*, where regions of space are implicitly coupled to other regions. Both types are encoded via the concept of a `patchField`. Examples of regular `patchFields` are

fixed value (Dirichlet), fixed gradient (Neumann), combined (Robin), symmetry or wedge patch fields (all decoupled), while coupled patch fields include concepts such as periodic, jump, and conjugate coupled boundaries. The `mesh` may be further composed of a number of `subregions`, for example the *air* mesh contains 3 regions: *channel*, *CCL* and *CFL* regions. The reader will note that in `openFuelCell` the mesh can contain regions or parts which need not be connected; e.g., the the *interconnect* mesh includes (two) unconnected solid regions, see Figure 2.

Parallel execution in `OpenFOAM` is facilitated by splitting a given mesh into *processor meshes*, each of which is assigned to a separate processor and which communicate with each other using a coupled `processorPatchField` field concept. This can be further extended to multi-region coupling, where a mesh can be implicitly coupled to another mesh via a region coupling patch field. For cases where a SOFC is handled as a set of non-overlapping regions, the SOFC model can be implemented in several ways, for example:

1. A **single global mesh**, where a single, physically continuous, mesh is tessellated through all the different component regions, and the full set of finite-volume equations are solved throughout, with terms being disabled as necessary within appropriate regions. This approach is simple, aids in parallelisation, and is consistent with earlier CFD code designs. It is however wasteful, in terms of memory and speed, and was not therefore considered in this programme of work.
2. A set of **coupled meshes**, where individual parts are tessellated with their own individual meshes, and these in turn support different solution sets. Boundaries between different components may be coupled, if necessary. Thus when, say, the air-channel species equations are being solved, the boundaries of the air mesh will remain decoupled; but when heat transfer is considered, the boundaries between fuel, electrolyte, air and inter-connect will, of necessity, be coupled, to yield a global solution for temperature. This method is efficient for decoupled components but not when coupling dominates, nor does it readily facilitate parallelisation.
3. An **integrated cell concept**, where a single global mesh is created and then split up into individual component 'child' meshes for air and fuel channels, electrolyte, inter-connect etc. The global mesh provides a framework for the solution of the heat transfer problem, while the individual meshes support solution for additional fields based on the governing transport equations. Fields and properties may be passed back and forth between global and compo-

nent meshes by mapping functions. Boundaries between regions are incorporated as internal surfaces during mesh generation, such that the integrated mesh may be split into components.

At first sight, the coupled mesh approach might seem most suitable, as it allows the user to individually create non-matching component meshes, and connect them as required. However, this is only practicable when the mesh is decomposed in a non-overlapping manner. For consistency in code design at multiple scales the integrated cell model was ultimately adopted in preference to the coupled sub-component design after both were considered in detail: In massively parallel simulations often required for CFD problems, it is essential to achieve good load balancing in order to preserve efficiency of execution. For cases where hundreds or thousands of processors are employed, it is optimistic to expect the user to achieve load balancing by individually decomposing each disparate component. Also, the coupled model needs to deal with a complex data transfer involving multiple processors in various combinations on either side of the coupled patch interfaces. While this is efficient and reliable on a small parallel decomposition, e.g. up to 64 processors, a thousand or more processor decompositions will not scale. The integrated cell concept, on the other hand requires the entire domain (parent mesh) to be decomposed into processor chunks; this being followed by identifying SOFC components locally on each processor. In this framework it is easier to achieve good load balance, and the problem of global data matching on coupled surfaces for thousands of processors is avoided. Combining better parallel efficiency and allowing support for multiple overlapping regions gave the present authors sufficient reason to choose the integrated cell concept as our primary model of implementation. Both the coupled and integrated cell concepts are fully implemented and tested.

3.2. Integrated Cell Algorithm

The model is solved on two sets of computational meshes: a single global or parent mesh for the entire SOFC, and on a set of child meshes for the air, fuel, electrolyte, and interconnect regions. Figure 3 illustrates the concept schematically. Each mesh supports its own particular field of state-variables, though the field set associated with the interconnect mesh is empty. Continuity, momentum, and species mass fractions, Eqs. (1)-(4) are solved-for on the two fluid (fuel and air) meshes, and electro-chemical heating is calculated within the electrolyte. The energy equation, Eq. (18), is solved exclusively on the global mesh, using heat source terms, Eq. (17), mapped from the

electrolyte and velocity values mapped from the fuel and air meshes. Cell mappings between child mesh cells and parent mesh cells are established during mesh generation (pre-processing) phase. The porous anode and cathode zones are implemented within the fuel and air meshes, respectively, by introducing suitable values for the permeability, κ_D, in the Darcy-modified Navier Stokes equation, Eq. (4). Electro-chemical reactions are assumed to occur at the electrode-electrolyte interface, as a surface phenomena on $\partial\Omega$. The choice of the `patch` location is somewhat arbitrary, and surface values of j, E etc. are subsequently 'smeared' into the volume, Ω, of the electrolyte. The resulting mass fluxes are used to derive Dirichlet-velocity and Neumann-species boundary conditions on the fuel and air boundaries adjacent to the electrolyte. Either the desired mean current density, j_0 (galvanostatic problem), or the electric potential, V (potentiostatic problem), may be prescribed.

The solution algorithm proceeds as follows:

i. Initialise meshes, constants, and other parameters. Specify initial fields and boundary conditions, physical properties, cell voltage or prescribed average current density.

ii. Map temperature from 'parent' mesh to fluid 'child' meshes.

iii. Compute air and fuel density and viscosity and solve Eqs. (1) and (2) for pressure and momentum according to the SIMPLE algorithm [63].

iv. Calculate diffusion coefficients and solve Eq. (4) for species mass fractions in fluid (fuel and air) 'child' meshes.

v. Calculate Nernst potential, Eq. (14), lumped internal resistance, current density, Eq. (16), on surface `patch` $\partial\Omega$.

vi. Calculate heat source term within the volume, Ω, of the electrolyte according to Eq. (17).

vii. Calculate electro-chemical mass fluxes, hence fluid region boundary condition for momentum and species at boundaries adjacent to electrolyte.

viii. Map 'child' mesh fields to global mesh.

ix. Solve Eq. (18) for global temperature.

x. Repeat from step ii until convergence is obtained.. For the potentiostatic problem the voltage is fixed, whereas for the galvanostatic formlation, the cell voltage must be adjusted

until the computed mean current density, $\bar{j}$, is identical to the set point, j_0,

$$V+ = \hat{R}\,(\bar{j} - j_0) \tag{19}$$

where $\hat{R}$ is just a relaxation constant.

4. Library, solvers and tutorial cases

The reader may obtain the code by issuing the command `git clone git://git.code.sf.net/p/openfuelcell/git openfuelcell` and subseqently following the instructions given at `http://openfuelcell.sourceforge.net/`. This will build an `openFuelCell` directory locally, containing all the files needed. Figure 4 illustrates salient features of the file structure which is adapted from the typical `OpenFOAM` directory hierarchy. Together with a `src` directory composed of compilable files `apprsrc` and class libraries `libsrc`, the code delivery includes a `run` directory with 5 demonstration cases, which allow the user to further refine, not only the run time inputs, but also the source code to his/her own future research or application needs. At present, these are for a planar anode-supported geometry, operating under conditions of `coFlow`, `counterFlow`, and `crossFlow`. In view of the processor requirements for these cases a simplified `quickTest` case which consists of only 3 channels, Figure 3, of a cell operating under co-flow is also provided. This will be used to further illustrate the file structure, below. In addition, another case, `quickTestStack`, which consists of a stack of 3 cells, is provided to demonstrate the application of the model, under galvanostatic conditions, to a stack of cells connected electrically in series, but hydraulically in parallel. In order to minimise computer resources each cell is composed of (only) 6 air and fuel channels: in reality fuel cell stacks are typically much larger. Additional `run` cases will be provided in the future.

Initial and boundary conditions are prescribed in `quickTest/0`; for example the boundary conditions to the Navier-Stokes and individual species in `quickTest/0/fuel/U` and `quickTest/0/fuel/YH2` etc. with similar files in a `quickTest/0/air` directory. Overall properties such as current density or voltage are set in `quickTest/constant/cellProperties`. Similarly, region-specific properties such as diffusivity, are specified in `quickTest/constant/fuel/fuelProperties`, `quickTest/constant/air/airProperties`, `quickTest/constant/electrolyte/electrolyteProperties`, `quickTest/constant/interconnect/interConnectProperties` and so forth. Consistent with

OpenFOAM conventions, system parameters related to the overall solution scheme are located in `quickTest/system/controlDict` whereas parameters related to the solutions on the child meshes may be found in files located in `quickTest/system/fuel`, `quickTest/system/air` etc. Files which perform the decomposition of the parent to the child meshes, `patches` definitions etc. are located in `quickTest/config`.

The directory `src/appSrc` contains compilable applications code including the main program `sofc.C` which is written in a modular fashion, and made up of a series of `include` files which perform the construction, implementation, solution and output of the hydrodynamic, electro-chemical and thermal equations sets on the multiple meshes, as described in this paper. For example `src/appSrc/solveElectrochemistry.H` contains the C++ code to solve the electro-chemical models described above, whereas `src/appSrc/SolveFuel.H` solves the Navier-Stokes equations on the region tesselated with the `fuel` mesh. `src/appSrc/solveElectrochemistry.H` includes more files, such as `src/appSrc/NernstEqn.H` which contains the C++ code to enumerate the Nernst equation, Eq. (14), for a `scalarField` of values for the `List` of nodes corresponding to the electrode/electrolyte boundary mesh $\partial\Omega$. `src/libSrc` contains run-time selectable models such as the various models of effective diffusivity (constant value, Fuller-Schettler-Giddings etc.)

In the cases provided, a mixture of hydrogen and water is prescribed as fuel with default inlet fractions of hydrogen (H_2) and water (H_2O) of 78.2% vol. and 21.8% vol, respectively, given as mass fractions in `dictionary` files `run/case/0/fuel/YH2` and `run/case/0/fuel/YH2O`. Similarly the oxidiser is prescribed as atmospheric air composed of 23.3% vol. oxygen (O_2) and 76.7% vol. nitrogen (N_2), respectively. Inlet velocities are in files and `case/0/air/U`. The user is free to prescribe his or her own material composition, initial and boundary values, as described above. In electro-chemistry, it is common practice to fix the fuel and air utilisations, defined as the ratios of fuel and air consumed to that provided, and a galvanostatic problem formulation is appropriate. Physicists may prefer to prescribe a potential, V. Galvanostatic/potentiostatic boundary conditions, and values for reference voltage/mean current are set by the user in `run/case/constant/cellProperties`.

The effects of the various flow configurations on current density and temperature uniformity have been considered by the present authors [33] and others in the electro-chemical literature, and will therefore be discussed only briefly here. A high temperature is expected at the outlet for co-flow, at the cell centre for counter-flow, and at the outlet corner for cross-flow. Generally speaking

performance is best for counter-flow and poorest for co-flow; the counter-flow configuration generally results in the highest average cell temperature, and the cross-flow configuration the lowest.

An item of interest to electro-chemists is the voltage-current density or polarisation curve. Reference [21] compares results obtained from this code with experimental data gathered for the Jülich Mark-F design, in terms of the $V - j$ curve. In computational fluid dynamics, convergence history and grid independence studies are undertaken, and such studies were made within the present context [64]. In this regard, the polarisation curve is perhaps not a particularly good measure of grid independence, due to the dominance of the area specific resistance on the slope, but rather local spot values of temperature, species mass fraction, and current density values and their extrema, should be scrutinised.

In planar SOFCs local Reynolds/P*é*clet numbers are often small, particularly on the fuel side, and utilisations based simply on integrating the convective flux, $\rho \mathbf{u} y_i$, may be distorted by diffusive transport, $\rho D_i^{\text{eff}} \operatorname{grad} y_i$ at the inlet and outlet boundaries. Under these circumstances there is a disparity in utilisation calculated from the outlet and inlet values compared to reaction rate and inlet values. At high current densities and utilisations, the mass fraction of fuel and oxidant will deplete, and both actual fuel cells and the model prototype here will fail, due to starvation, as $y_i \to 0$ at the electrode, the latter due to the Nernst formulation. This typically occurs at a point midway between gas channels corresponding to the solid rib of the interconnect and is unavoidable, due to the distribution/difference of mass fraction in the cross-plane to the main flow being proportional to the magnitude of the current density; as the mean current density is increased the global minima proportionally decreases up to the point of starvation.

One alternative to the present Nernst-equation based formulation, is to solve for the local electronic and ionic electric field potentials in the electrodes, electrolyte and interconnects, using an additional Laplace-Poisson system of equations, whole-field. An advantage of the integrated cell approach is that it will readily admit to a second 'family' of 'child' meshes corresponding to solid electronic (electrodes+interconnects) and ionic (electrodes+electrolyte) conducting regions to be synthesised in addition to the existing children.

An advantage of the present Nernst formulation is that the code may readily be employed to perform calculations for stacks of fuel cells. Because of the flexible nature of the code, Keuler et al. [31, 65] modified the `openFuelCell` code to perform calculations on a high temperature polymer

electrolyte fuel cell, operating at $160 - 180°C$, where hydrogen protons are transported across the membrane to the cathode (as opposed to oxygen ions going to the anode), demonstrating the flexibility of the code suite. Choi et al. [66] applied the code to the problem of a solid oxide electrolysis cell, where hydrogen and oxygen are produced from water, electricity, and heat. In an earlier work Choi et al. [23] conducted cell-level calculations as described here, with effective diffusivity including Knudsen diffusion, tortuosity factor, and permeability computed using volume-averaging from micro-scale calculations. The authors maintain that heuristic methods such as the Bruggeman relation [53] may readily be supplanted by numerical reconstruction techniques [23, 67, 68] based on random walk or Monte-Carlo simulations [22] to obtain effective micro-structural properties such as the tortuosity factor, τ, and characteristic length, d_{cl}. Various approaches have been adopted for evaluating d_{cl} [22, 47, 69–72]. The present authors estimate heuristic methodologies overestimate d_{cl} by 10-38 % and underestimate the tortuosity factor, τ, by up to 57 % compared to numerical evaluation. Because of this, the effective diffusion coefficient can be out (too large) by up to 5-63 %. Micro-scale and nano-scale calculations can and should be used, in preference to traditional semi-empirical methods, to parameterise continuum-scale models.

It is believed that the present cell code is the first fully comprehensive SOFC code. Although there are other open source fuel cell codes [73], these are primarily for polymer electrolyte membrane fuel cells. The SOFC code referred to in [30] appears to be primarily a mass-transport library; moreover constant temperature is presumed. The local temperature strongly impacts the solution, e.g., through the electrolyte resistance. The present code architecture is capable of performance calculations for stacks as well as single cells, and is fully-parallelised. This is among the very few efficient implementations of coupled overlapping multi-region flow/heat/mass transfer infrastructure which support high performance computing without a significant loss of efficiency;

Figures 5 and 6 illustrate the results of performance calculations conducted for two of the `run` library cases. Figure 5 shows the hydrogen mass fraction together with fuel velocity profiles along the channels for the `coFlow` case. It can be seen that the hydrogen is consumed by the reaction as indicated by the reduction in mass fraction, and that the velocity profiles correspond to those associated with laminar fully-developed flow. Figure 6 shows local values of the Nernst potential, i.e., the theoretical maximum thermodynamic potential that can be obtained, and it can be seen that performance degrades from the fuel/air inlet to the fuel/air outlet. The current

density under the ribs is lower, due to reduced mass transfer [49], and this results in a non-uniform temperature distribution (due to Joule and other heating effects being reduced here). Observed values of temperatures underneath the ribs are therefore lower than those beneath the channels. The reader may readily download the cases from `http://openfuelcell.sourceforge.net/`. Following execution, figures similar to Figs. 5 and 6 may readily be generated using the open source package ParaView [74] or equivalent post-processing and visualisation tools. An illustrative example of code application, this time for the Jülich Mark F geometry is displayed in Fig. 7. Typical outputs include (a) air-side pressure, (b) plate temperature, (c) air-side stream-lines, (d) hydrogen mass fraction, (e) local current density, and (f) Nernst potential. It can be seen that the reaction rate (local current density) drops off towards the sides of the cell where there is no active surface. The Nernst potential, Eq. (14), is seen to decrease going from inlet to outlet, as the fuel is depleted due to the combined effects of hydrogen consumption and water production decreasing the equilibium constant, Q, Eq. (15). This results in a decrease in the observed local current density, j. A reduced current density means less heating in the PEN. The temperature is an important parameter as it affects a number of other parameters, especially the ohmic resistance, R, which decreases with increasing temperature. An important goal in SOFC design is structural integrity: Failure can occur at the glass-metal seals when temperature gradients become excessive. Metallic interconnects will distribute heat by conduction more uniformly than ceramic ones. Flow maldistributions also can lead to 'hotspots', and while this is not the case here, pressure non-uniformities due, e.g. to manifold design, are to be avoided for similar reasons. This figure illustrates how the design and development of practical fuel cells may be enhanced as a result of virtual prototypes, which themselves are based on sound physics and chemistry, and software best practices.

5. Conclusion and future work

A computer code was developed to obtain performance calculations for planar-type SOFCs using the object-oriented continuum mechanics platform, `OpenFOAM`. The code was initially validated by conducting performance calculations for a simplified model for which previous calculations had been obtained using two independent commercial CFD packages. In all cases near identical results were obtained. The present code accounts for all significant electro-chemical and transport phenomena at the cell level, being based on the solution of the Nernst and Kirchhoff-Ohm equations coupled with

the equations for concentration, heat, and momentum. Activation overpotential may be accounted-for, by default, with the solution of one or two Butler-Volmer type equations, or else with other user-prescribed kinetics. The `openFuelCell` code was designed to be general purpose, to allow for future developments/application to other situations such as non-planar (tubular) architectures; different fuel cell types, for example low and high temperature polymer electrolyte membrane fuel cells; and other devices such as electrolysers; and also employ other fuels other than hydrogen, for example methane. Many of these developments are ongoing.

The model implements a new framework of multi-region modelling, to answer to the specific challenges of the fuel cell problem. Here, multiple equation sets for fluid flow, species concentration, heat and mass transfer and chemical reactions need to interact within a general framework, supported by the OpenFOAM structure. Of particular interest is multi-domain coupling and data transfer which is capable of supporting mutiple overlapping and partially overlapping regions with efficient region-to-region coupling (no search algorithms are needed). This is achieved without sacrificing the fully implicit solution of cell-wise equations such as heat transfer, which is needed for efficiency and stability. The code is suitable for execution on large-scale parallel computers and has been run on 960 cores and more. The authors' aim is to freely share the code interface, source code and underlying physics, without necessarily giving out details specific to any particular fuel cell design, as this may be commercially sensitive. Thus, by sharing the interface but not the application, it is hoped the state-of-the-art of fuel cells technology in the market will be accelerated.

The Open Source SOFC model is envisaged as an open code base where multiple researchers are in position to introduce improvements to individual components of the SOFC model, while re-using parts of the code without changes. In this setup, contributions of individual alternative formulations of the physics model can be individually judged on a fair basis, as they are executed within a unified environment, eliminating issues such as code consistency, order of accuracy or mesh-related discretisation errors, present in multi-code validation exercises.

Virtual prototypes enhance but do not replace experimental work. The authors plan to conduct a detailed progamme of validation and verification in the future. The International Energy Agency Advanced Fuel Cells Implementing agreement (IEA AFCIA) [75] recently created a modelling annex which is dedicated to open-source computational modelling, which includes the benchmarking of codes, and also development of experimental protocols for validation and verification. With this

ongoing effort, it is anticipated that significantly more technical development and capabilities will be generated in the years ahead in collaboration with participating IEA AFCIA member countries. The cell-level model is the first of several multi-scale efforts being developed by the present group. Fuel cells are normally operated in stacks, and the requirements of a stack-scale model are significantly different from those for a single-cell model. A SOFC stack code, based on volume-averaging, with similar architecture to the present code suite is currently under development. Moreover, as discussed above, a number of effective properties and model equation coefficients may be amenable to micro-scale models of both transport and chemical kinetics. These multiple scale solutions to the physicochemical hydrodynamics associated with electro-chemical processes and devices such as fuel cells and electrolysers, will be published both in the scientific literature and online as code repositories, in the future.

Acknowledgments

Financial support for this work was provided through the Solid Oxide Fuel Cell Canada Strategic Research Network from the Natural Science and Engineering Research Council, Forschungszentrum Jülich GmbH, the National Research Council of Canada, Defence Research and Development Canada, and the Program of Energy Research and Development of Natural Resources Canada. The contribution of Ron Jerome to the maintainance of the `openFuelCell` code and web site is acknowledged with gratitude. The authors thank Dieter Froning who read the manuscript and contributed some suggestions. Robert Nishida wrote the `quickTestStack` case which appears in the run library.

References

[1] S. Singhal, K. Kendall (Eds.), High Temperature Solid Oxide Fuel Cells: Fundamentals, Desigan and Applications, Elsevier, 2003.

[2] S. B. Beale, Y. Lin, S. V. Zhubrin, W. Dong, Computer methods for performance prediction in fuel cells, Journal of Power Sources 118 (1-2) (2003) 79–85.

[3] M. Andersson, J. Yuan, B. Sundén, Review on modeling development for multiscale chemical

reactions coupled transport phenomena in solid oxide fuel cells, Applied Energy 87 (2010) 1461–1476.

[4] Z. Qu, P. Aravind, N. Dekker, A. Janssen, N. Woudstra, A. Verkooijen, Three-dimensional thermo-fluid and electrochemical modeling of anode-supported planar solid oxide fuel cell, Journal of Power Sources 195 (2010) 7787–7795.

[5] M. Chen, L. A. Rosendahl, T. Condra, A three-dimensional numerical model of thermoelectric generators in fluid power systems, International Journal of Hydrogen Energy 54 (2011) 345–355.

[6] S. A. Hajimolana, M. A. Hussain, W. A. W. Daud, M. Soroush, A. Shamiri, Mathematical modeling of solid oxide fuel cells: A review, Renewable & Sustainable Energy Reviews 15 (2011) 1893–1917.

[7] A. Mauro, F. Arpino, N. Massarotti, Three-dimensional simulation of heat and mass transport phenomena in planar SOFCs, International Journal of Hydrogen Energy 36 (2011) 10288–10301.

[8] Z. Qu, P. Aravind, S. Boksteen, N. Dekker, A. Janssen, N. Woudstra, A. Verkooijen, Three-dimensional computational fluid dynamics modeling of anode-supported planar SOFC, International Journal of Hydrogen Energy 36 (2011) 10209–10220.

[9] M. Andersson, H. Paradis, J. Yuan, B. Sundén, Three dimensional modeling of an solid oxide fuel cell coupling charge transfer phenomena with transport processes and heat generation, Electrochimica Acta 109 (2013) 881–893.

[10] M. Peksen, R. Peters, L. Blum, D. Stolten, Numerical modelling and experimental validation of a planar type pre-reformer in SOFC technology, International Journal of Hydrogen Energy 34 (15) (2009) 6425–6436.

[11] W. Dong, S. B. Beale, R. J. Boersma, Computational modelling of solid oxide fuel cells, in: Proceedings of the 9th Conference of the CFD Society of Canada - CFD 2001, 2001, pp. 382–387.

[12] V. A. Danilov, M. O. Tade, A CFD-based model of a planar SOFC for anode flow field design, International Journal of Hydrogen Energy 34 (21) (2009) 8998–9006.

[13] Q. Wang, L. Li, C. Wang, Numerical study of thermoelectric characteristics of a planar solid oxide fuel cell with direct internal reforming of methane, Journal of Power Sources 186 (2) (2009) 399–407.

[14] D. Froning, L. Blum, A. Gubner, L. de Haart, M. Spiller, D. Stolten, Experiences with a CFD based two stage SOFC stack modeling concept and its application, ECS Transactions 7 (1) (2007) 1831–1840.

[15] T. X. Ho, P. Kosinski, A. C. Hoffmann, A. Vik, Modeling of transport, chemical and electrochemical phenomena in a cathode-supported SOFC, Chemical Engineering Science 64 (12) (2009) 3000–3009.

[16] T. X. Ho, P. Kosinski, A. C. Hoffmann, A. Vik, Numerical analysis of a planar anode-supported SOFC with composite electrodes, International Journal of Hydrogen Energy 34 (8) (2009) 3488–3499.

[17] N. Akhtar, S. Decent, K. Kendall, Numerical modelling of methane-powered micro-tubular, single-chamber solid oxide fuel cell, Journal of Power Sources 195 (23) (2010) 7796–7807.

[18] H. G. Weller, G. Tabor, H. Jasak, C. Fureby, A tensorial approach to computational continuum mechanics using object-oriented techniques, Computers in Physics 12 (6) (1998) 620–631.

[19] H.-W. Choi, A. Berson, B. Kenney, J. G. Pharoah, S. B. Beale, K. Karan, Effective transport coefficients for porous microstructures in solid oxide fuel cells, ECS Transactions 25 (2009) 1341–1350.

[20] D. H. Jeon, S. B. Beale, H.-W. Choi, J. G. Pharoah, H. Roth, Computational study of heat transfer issues in solid oxide fuel cells, in: 21st International Symposium on Transport Phenomena, Kaohsiung City, Taiwan, 2010.

[21] S. B. Beale, A. D. Le, H. K. Roth, J. G. Pharoah, H.-W. Choi, L. G. J. de Haart, D. Froning, Numerical and experimental analysis of a solid oxide fuel cell stack, ECS Transactions 35 (1) (2011) 935–943.

[22] A. Berson, H.-W. Choi, J. G. Pharoah, Determination of the effective gas diffusivity of a porous composite medium from the three-dimensional reconstruction of its microstructure, Physical Review E. 83 (2) (2011) 026310.

[23] H.-W. Choi, A. Berson, J. G. Pharoah, S. B. Beale, Effective properties of the porous electrodes in solid oxide fuel cells, Proceedings of the Institution of Mechanical Engineers, Part A: Journal of Power and Energy 225 (2) (2011) 183–197, special issue on fuel cells for stationary applications.

[24] H.-W. Choi, A. Berson, J. G. Pharoah, S. B. Beale, Effect of porous micro-structural properties on the results of a cell-level model in solid oxide fuel cells, in: 219th ECS Meeting: Solid Oxide Fuel Cells, Twelfth International Symposium (SOFC-XII), Montreal, QC, Canada, 2011, pp. 1107–114.

[25] H.-W. Choi, D. Gawel, A. Berson, J. G. Pharoah, K. Karan, Comparison between FIB-SEM experimental 3-D reconstructions of SOFC electrodes and random particle-based numerical models, ECS Transactions 35 (1) (2011) 997–1005.

[26] M. García-Camprubí, A. Sánchez-Insa, N. Fueyo, Multimodal mass transfer in solid-oxide fuel-cells, Chemical Engineering Science 65 (2010) 1668–1677.

[27] M. García-Camprubí, N. Fueyo, Mass transfer in hydrogen-fed anode-supported SOFCs, International Journal of Hydrogen Energy 35 (2010) 11551–11560.

[28] M. García-Camprubí, H. Jasak, F. N., CFD analysis of cooling effects in H2-fed solid oxide fuel cells, Journal of Power Sources 196 (17) (2011) 7290–7301.

[29] N. S. K. Gunda, H.-W. Choi, A. Berson, B. Kenney, K. Karan, J. G. Pharoah, S. K. Mitra, Focused ion beam-scanning electron microscopy on solid-oxide fuel-cell electrode: Image analysis and computing effective transport properties, Journal of Power Sources 196 (7) (2011) 3592–3603.

[30] V. Novaresio, M. García-Camprubí, S. Izquierdo, P. Asinari, N. Fueyo, An open-source library for the numerical modeling of mass-transfer in solid oxide fuel cells, Computer Physics Communications 183 (1) (2012) 125–146.

[31] S. Keuler, Q. Cao, S. Lee, D. Froning, U. Reimer, S. B. Beale, W. Lehnert, Generic OpenFOAMfuel cell model applied to HT-PEFC, in: ModVAL11 11th Symposium on Fuel Cell and Battery Modeling and Experimental Validation, Winterthur, Switzerland, 2014.

[32] E. Achenbach, IEA programme on R, D & D on advanced fuel cells Annex II: Modeling and evaluation of advanced solid oxide fuel cells, SOFC stack modeling, Tech. Rep. IEA Programme on Advanced Fuel Cells Annex II, International Energy Agency (1996).

[33] A. D. Le, S. B. Beale, J. G. Pharoah, Validation of a solid oxide fuel cell model on the international energy agency benchmark case with hydrogen fuel, Fuel Cells 15 (1) (2015) 27–41.

[34] J. Ferguson, J. Fiard, R. Herbin, Three-dimensional numerical simulation for various geometries of solid oxide fuel cells, Journal of Power Sources 58 (1996) 109–122.

[35] R. Braun, Optimal design and operation of solid oxide fuel cell systems for small-scale stationary applications, Phd (2002).

[36] C. Colpan, I. Dincer, F. Hamdullahpur, A review on macro-level modeling of planar solid oxide fuel cells, International Journal of Energy Research 32 (2008) 336–355.

[37] M. Li, J. Powers, J. Brouwer, A finite volume SOFC model for coal-based integrated gasification fuel cell systems analysis, Journal of Fuel Cell Science and Technology 7 (2010) 041017.

[38] Q. Fang, L. Blum, P. Batfalsky, N. H. Menzler, U. Packbier, D. Stolten, Durability test and degradation behavior of a 2.5 kw SOFC stack with internal reforming of LNG, International Journal of Hydrogen Energy 38 (36) (2013) 16344–16353.

[39] L. Blum, U. Packbier, I. Vinke, L. de Haart, Long-term testing of SOFC stacks at Forschungszentrum Jülich, Fuel Cells 13 (4) (2013) 646–653.

[40] L. Blum, P. Batfalsky, Q. Fang, L. de Haart, J. Malzbender, N. Margaritis, N. H. Menzler, R. Peters, Solid oxide fuel cell, stack and system development status at forschungszentrum jlich, ECS Transactions 68 (1) (2015) 157–169.

[41] C. R. Wilke, Diffusional properties of multicomponent gases, Chemical Engineering Progress 46 (2) (1950) 95–104.

[42] B. Todd, J. B. Young, Thermodynamic and transport properties of gases for use in solid oxide fuel cell modelling, Journal of Power Sources 110 (1) (2002) 186–200.

[43] E. N. Fuller, P. D. Schettler, J. C. Giddings, New method for prediction of binary gas-phase diffusion coefficients, Industrial and Engineering Chemistry 58 (5) (1966) 18–27.

[44] B. Poling, J. Prausnitz, J. Connell, The Properties of Gases and Liquids, McGraw-Hill, New York, 2001.

[45] R. Eslamloueyan, M. H. Khademi, A neural network-based method for estimation of binary gas diffusivity, Chemometrics and Intelligent Laboratory Systems 104 (2010) 195–204.

[46] N. Epstein, On tortuosity and the tortuosity factor in flow and diffusion through porous media, Chemical Engineering Science 44 (1989) 777–789.

[47] J. M. Zalc, S. C. Reyes, E. Iglesia, The effects of diffusion mechanism and void structure on transport rates and tortuosity factors in complex porous structures, Chem. Engrg. Sci. 59 (2004) 2947–2960.

[48] R. Krishna, J. A. Wesselingh, The Maxwell-Stefan approach to mass transfer, Chemical Engineering Science 52 (6) (1997) 861–911.

[49] D. H. Jeon, J. H. Nam, C.-J. Kim, Microstructural optimization of anode-supported solid oxide fuel cells by a comprehensive microscale model, Journal of The Electrochemical Society 153 (2) (2006) A406–A417.

[50] D. H. Jeon, A comprehensive CFD model of anode-supported solid oxide fuel cells, Electrochimica Acta 54 (10) (2009) 2727–2736.

[51] R. E. Williford, L. A. Chick, G. D. Maupin, S. P. Simner, J. W. Stevenson, Diffusion limitations in the porous anodes of SOFCs, Journal of The Electrochemical Society 150 (8) (2003) A1067–A1072.

[52] V. H. Schmidt, C.-L. Tsai, Anode-pore tortuosity in solid oxide fuel cells found from gas and current flow rates, Journal of Power Sources 180 (1) (2008) 253–264.

[53] D. A. G. Bruggeman, Calculation of various physics constants in heterogenous substances i dielectricity constants and conductivity of mixed bodies from isotropic substances, Annalen der Physik 24 (7) (1935) 636–664.

[54] A. J. Bard, L. R. Faulkner, Electrochemical methods: fundamentals and applications, 2nd Edition, Wiley New York, 2001.

[55] S. Nagata, A. Momma, T. Kato, Y. Kasuga, Numerical analysis of output characteristics of tubular SOFC with internal reformer, Journal of Power Sources 101 (1) (2001) 60–71.

[56] A. Leonide, Y. Apel, E. Ivers-Tiffee, SOFC modeling and parameter identification by means of impedance spectroscopy, ECS Transactions 19 (20) (2009) 81–109.

[57] E. Hernández-Pacheco, M. D. Mann, The rational approximation method in the prediction of thermodynamic properties for SOFCs, Journal of Power Sources 128 (1) (2004) 25–33.

[58] E. Mason, S. Saxena, Approximate formula for the thermal conductivity of gas mixtures, Physics of Fluids (1958-1988) 1 (5) (1958) 361–369.

[59] K. J. Daun, S. B. Beale, F. Liu, G. J. Smallwood, Radiation heat transfer in planar SOFC electrolytes, Journal of Power Sources 157 (1) (2006) 302–310.

[60] D. S. Kershaw, The incomplete Cholesky-Conjugate gradient method for the iterative solution of systems of linear equations, Journal of Computational Physics 26 (1) (1978) 43–65.

[61] H. A. van der Vorst, Bi-CGSTAB: A fast and smoothly converging variant of Bi-CG for the solution of nonsymmetric linear systems, SIAM Journal of Sci. Computing 13 (2) (1992) 631–644.

[62] S. Beale, S. Zhubrin, A distributed resistance analogy for solid oxide fuel cells, Numerical Heat Transfer, Part B 47 (2005) 573–591.

[63] S. V. Patankar, D. B. Spalding, Calculation procedure for heat, mass and momentum-transfer in 3-dimensional parabolic flows, International Journal of Heat Mass Transfer 15 (10) (1972) 1787–1806.

[64] D. H. Jeon, CFD modeling of solid oxide fuel cells, Tech. Rep. TR PET-1596-105, National Research Council of Canada (2010).

[65] S. Keuler, Generisches, openfoam-basiertes brennstoffzellenmodell angewandt auf die hochtemperatur-polymerelektrolyt-brennstoffzelle, Msc (2013).

[66] H.-W. Choi, J. G. Pharoah, D. Ryland, A. Kettner, N. Gnanapragasam, Computational fluid dynamics modeling of solid oxide electrolysis cell, ECS Transactions 57 (1) (2013) 3161–3170.

[67] B. Kenney, M. Valdmanis, C. Baker, J. G. Pharoah, K. Karan, Computation of TPB length, surface area and pore size from numerical reconstruction of composite solid oxide fuel cell electrodes, Journal of Power Sources 189 (2009) 1051–1059.

[68] A. Bertei, H.-W. Choi, J. G. Pharoah, C. Nicolella, Percolating behavior of sintered random packings of spheres, Powder Technology 231 (2012) 44–53.

[69] B. Lu, S. Torquato, Chord-length and free-path distribution functions for many-body systems, Journal of Chem. Phys. 98 (1993) 6472–6482.

[70] B. Derjaguin, Measurement of the specific surface of porous and disperse bodies by their resistance to the flow of rarified gases, Proceedings of the USSR Academy of Sciences 53 (1946) 623–626.

[71] P. Levitz, Knudsen diffusion and excitation transfer in random porous media, Journal of Phys. Chem. 97 (1993) 3813–3818.

[72] P. Levitz, Off-lattice reconstruction of porous media: critical evaluation, geometrical confinement and molecular transport, Adv. Colloid Interface Sci. 76-77 (1998) 71–106.

[73] A. Z. Weber, R. L. Borup, R. M. Darling, P. K. Das, T. J. Dursch, W. Gu, D. Harvey, A. Kusoglu, S. Litster, M. M. Mench, et al., A critical review of modeling transport phenomena in polymer-electrolyte fuel cells, Journal of The Electrochemical Society 161 (12) (2014) F1254–F1299.

[74] ParaView: Open-source, multi-platform data analysis and visualization application available from http://www.paraview.org/.

[75] International Energy Agency Advanced Fuel Cells Implementing Agreement web site http://www.ieafuelcell.com/.

Tables

```
speciesList

(

O2

N2

);

O2          O2    31.9988   4   -1   0   205.152;

N2          N2    28.0134   0   0    0   191.609;

//          |     |         |   |    |   |

//          |     |         |   |    |   StandardEntropy [J/(mol K)]

//          |     |         |   |    enthalpy of Formation [J/mol]

//          |     |         |   produced=1 | inert=0 | consumed=-1

//          |     |         molecular charge for FaradaysLaw

//          |     molecularWeight [kg/kmol]

//          name
```

Table 1: Prescription of air species in a dictionary

Figure Captions

Figure 1. Schematic illustrating geometry of the Jülich Mark F solid oxide fuel cell case showing details of the gas manifolds and channels.

Figure 2. Computational geometry and meshes for a typical planar anode supported SOFC. The parts are tessellated with rectilinear meshes.

Figure 3. Schematic showing decomposition of parts using 'parent' and 'child' meshes for the `quickTest` case.

Figure 4. Directory and file structure for the `openFuelCell` code.

Figure 5. Velocity profile and hydrogen mass fractions in the anode of the `coFlow` case.

Figure 6. Nernst potential for `crossFlow` case.

Figure 7. Results of performance calculations for the Jülich Mark F solid oxide fuel cell showing (a) air-side pressure, (b) plate temperature, (c) stream-lines, (d) hydrogen mass fraction, (e) local current density, (f) Nernst potential.

Figures

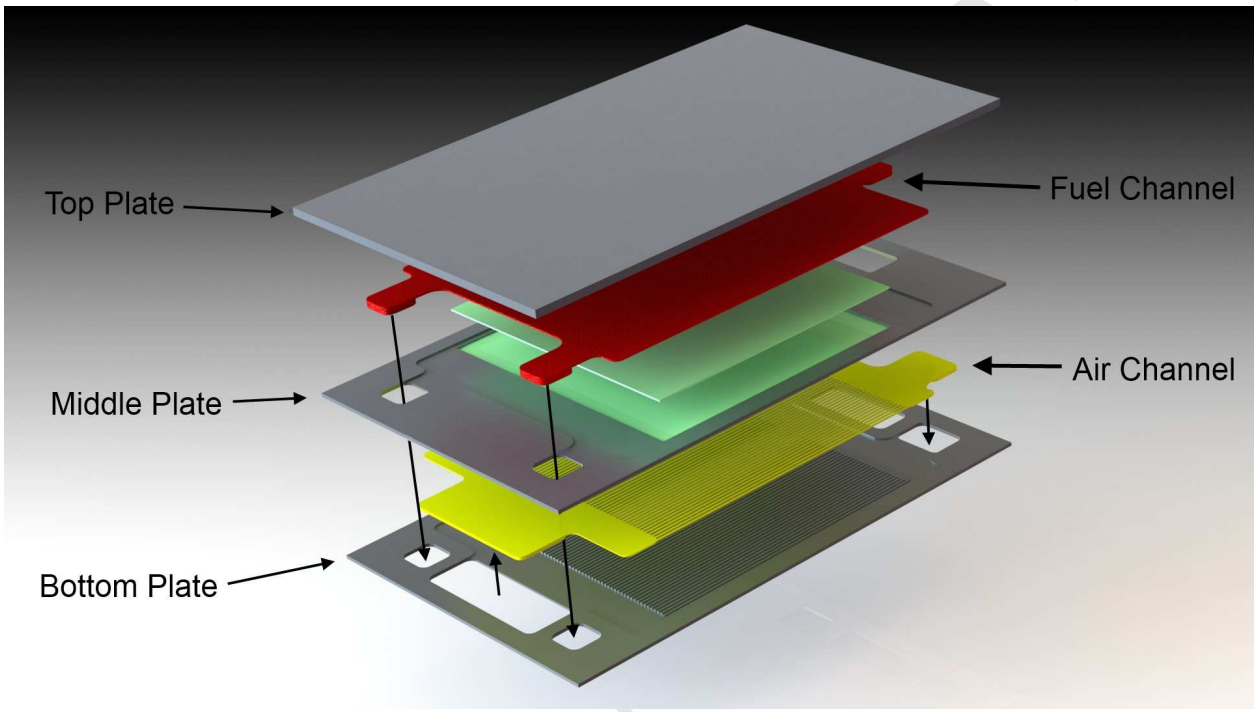


Figure 1: Schematic illustrating geometry of the Jülich Mark F solid oxide fuel cell case showing details of the gas manifolds and channels.

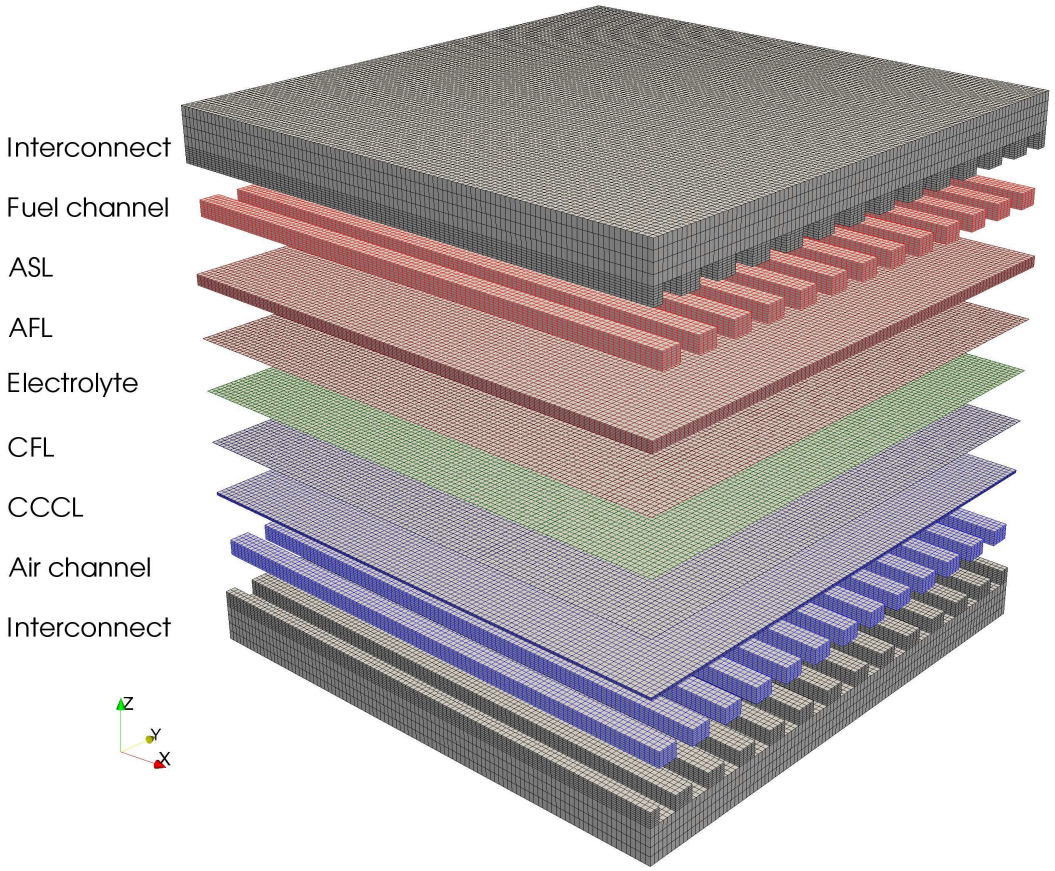


Figure 2: Computational geometry and meshes for a typical planar anode supported SOFC. The parts are tessellated with rectilinear meshes.

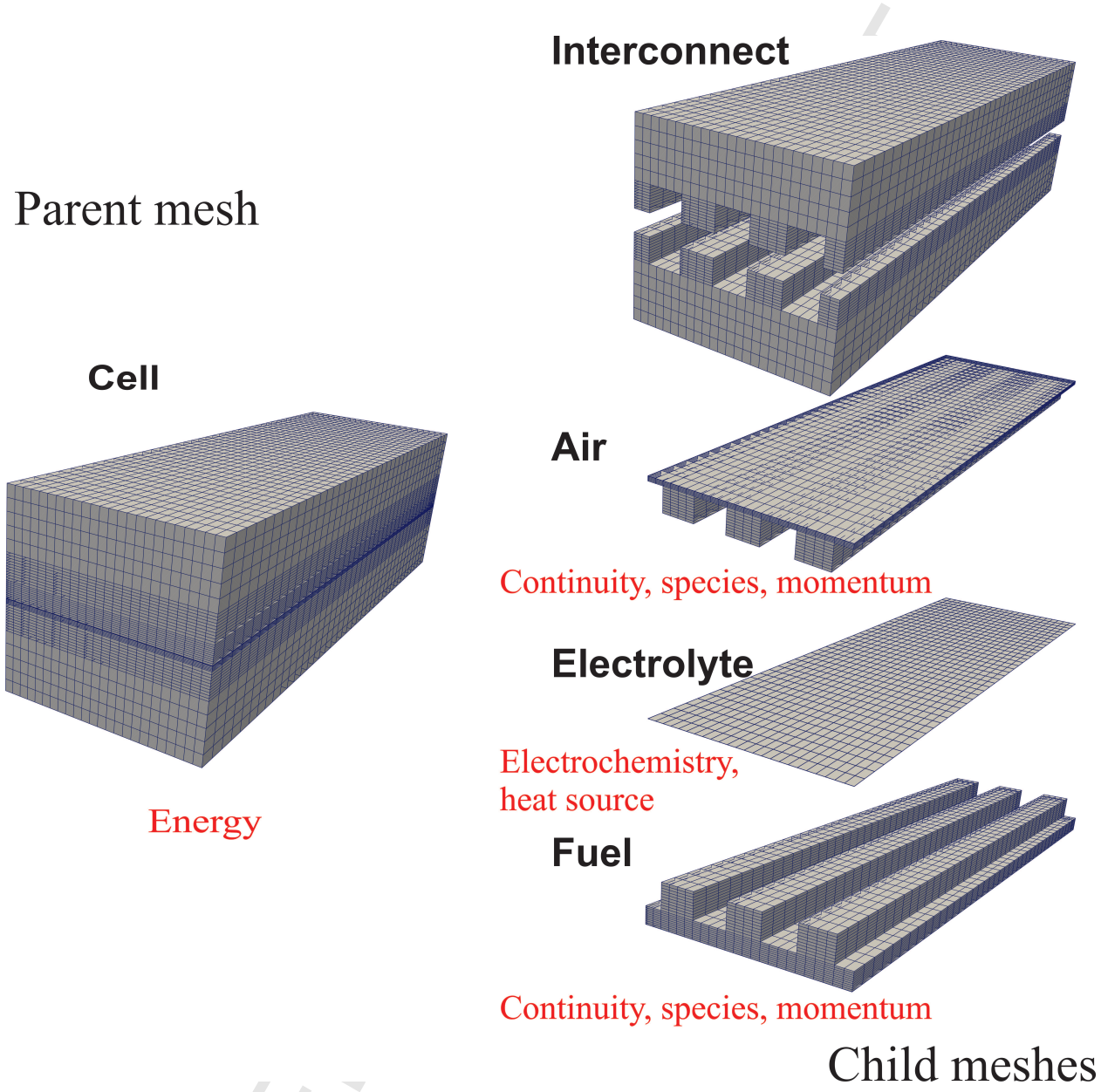


Figure 3: Schematic showing decomposition of parts using 'parent' and 'child' meshes for the `quickTest` case.

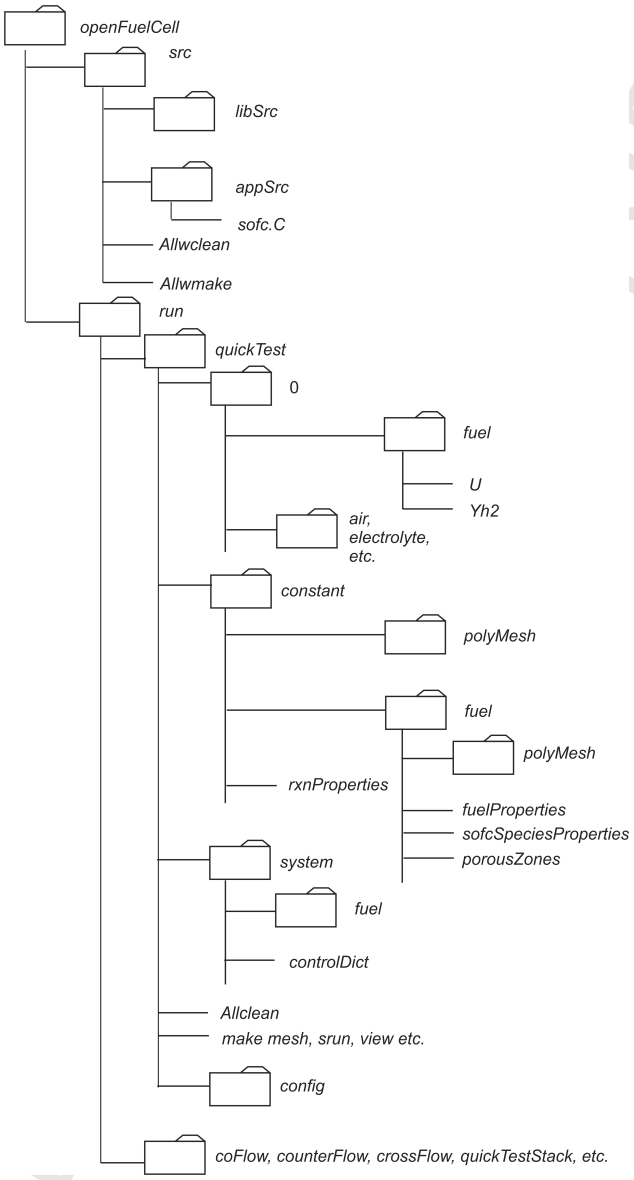


Figure 4: Directory and file structure for the `openFuelCell` code.

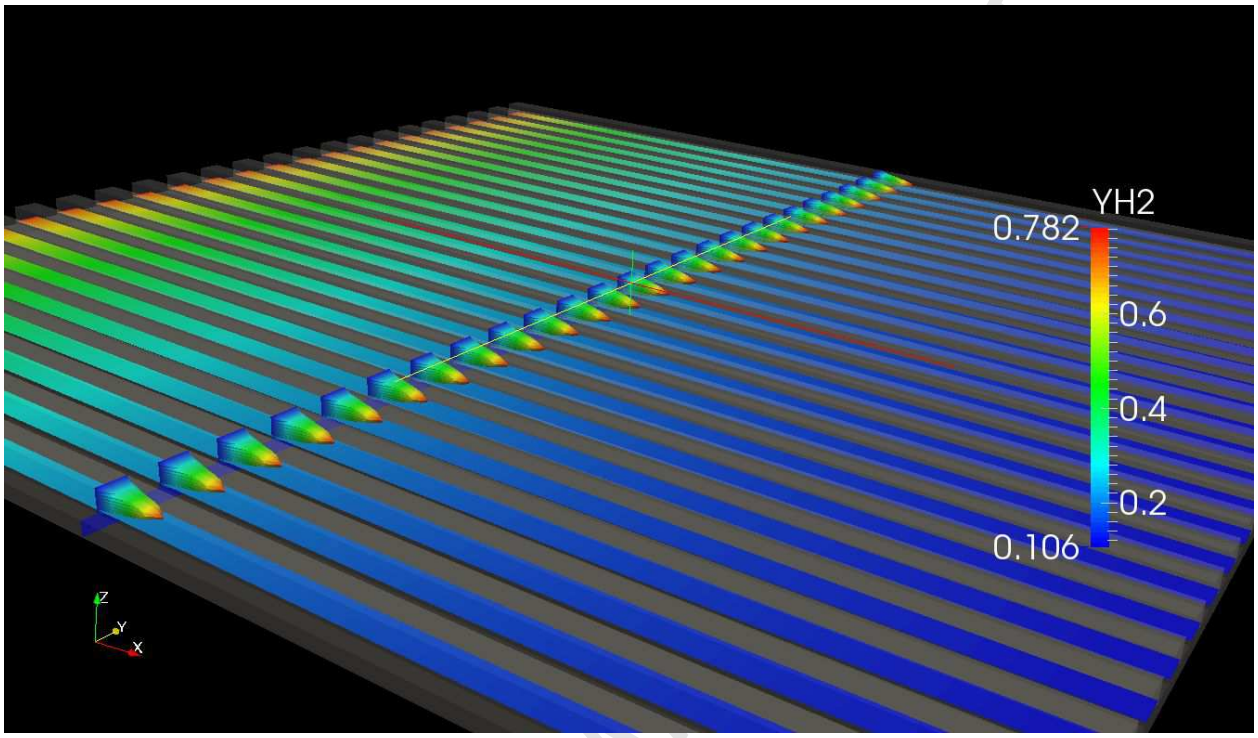


Figure 5: Velocity profile and hydrogen mass fractions in the anode of the `coFlow` case.

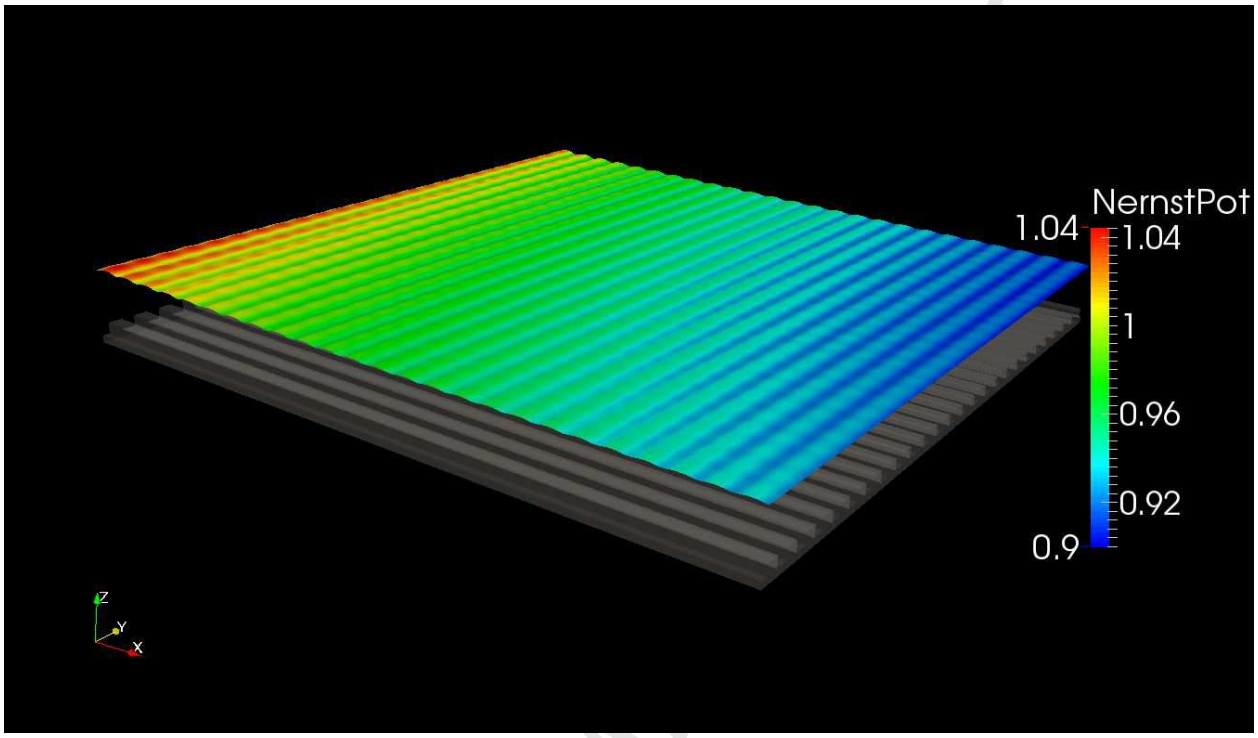


Figure 6: Nernst potential for `crossFlow` case.

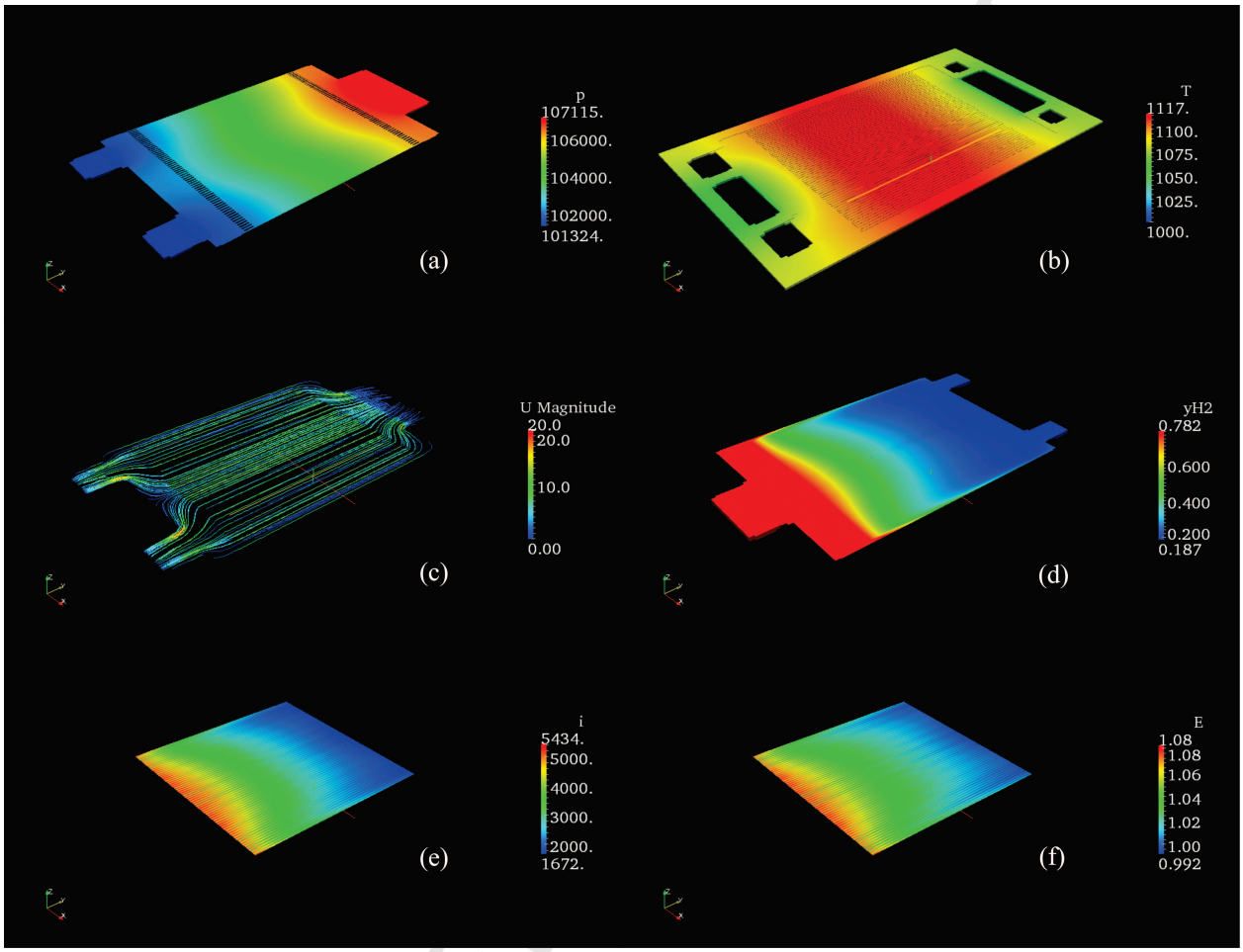


Figure 7: Results of performance calculations for the Jülich Mark F solid oxide fuel cell showing (a) air-side pressure, (b) plate temperature, (c) stream-lines, (d) hydrogen mass fraction, (e) local current density, (f) Nernst potential.