EXPEDITED REAL TIME PROCESSING FOR THE NETL HYPER CYBER-PHYSICAL SYSTEM (FE0030600)

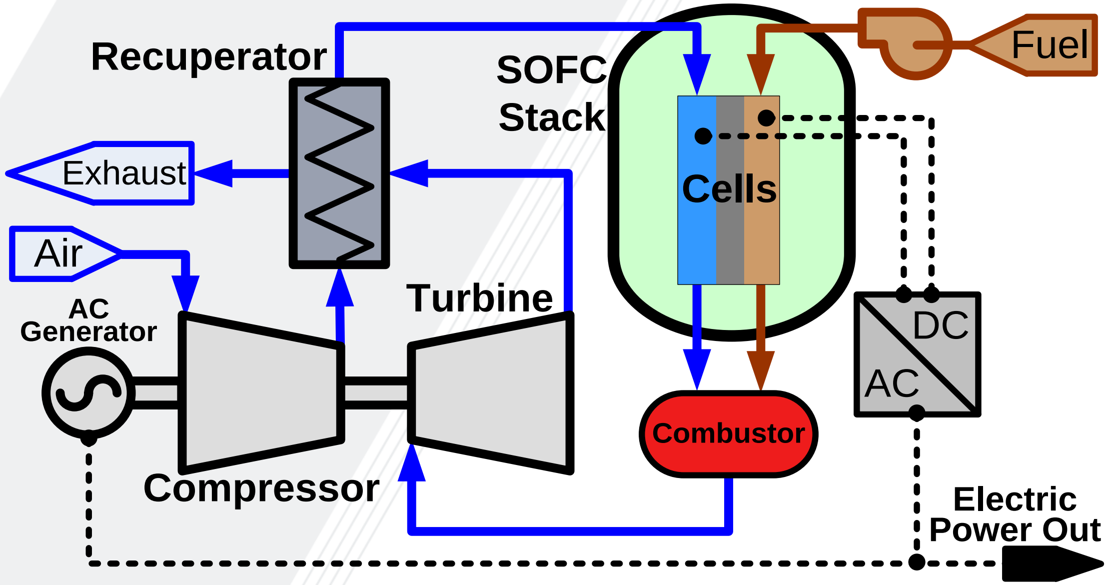
FY22 FECM Spring R&D Project Review Meeting

May 10th, 2022

Comas Haynes
Georgia Institute of Technology
Georgia Tech Research Corporation
Comas.Haynes@gtri.gatech.edu

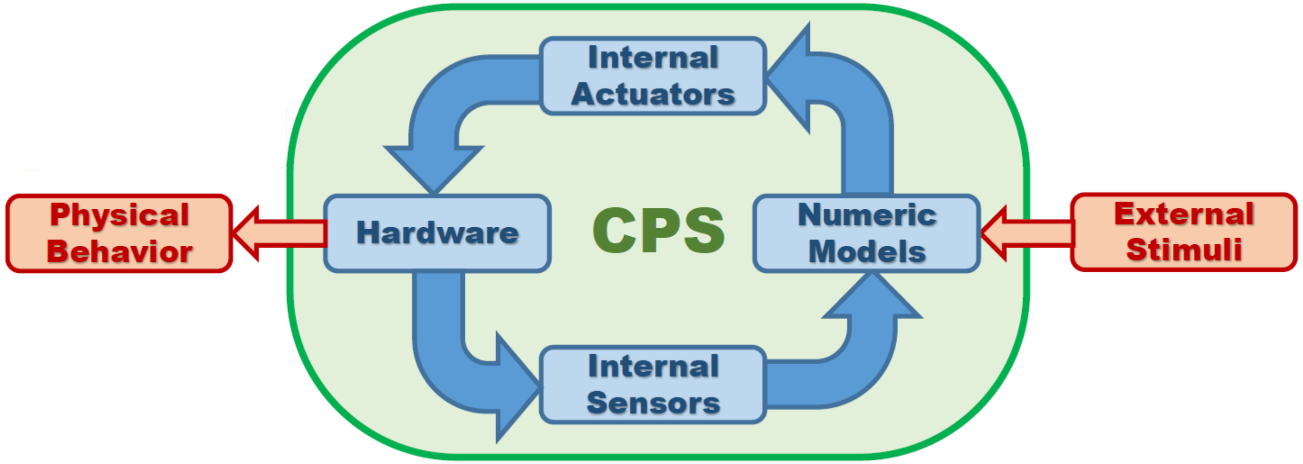
Solid Oxide Fuel Cell/Gas Turbine (SOFC/GT) Hybrids

- Feature high electrical efficiency for power generation
- Compressor provides pressurized airflow to SOFC stack
- Pressurized SOFC provides thermal effluent to drive turbine
- Can use natural gas or coal syngas

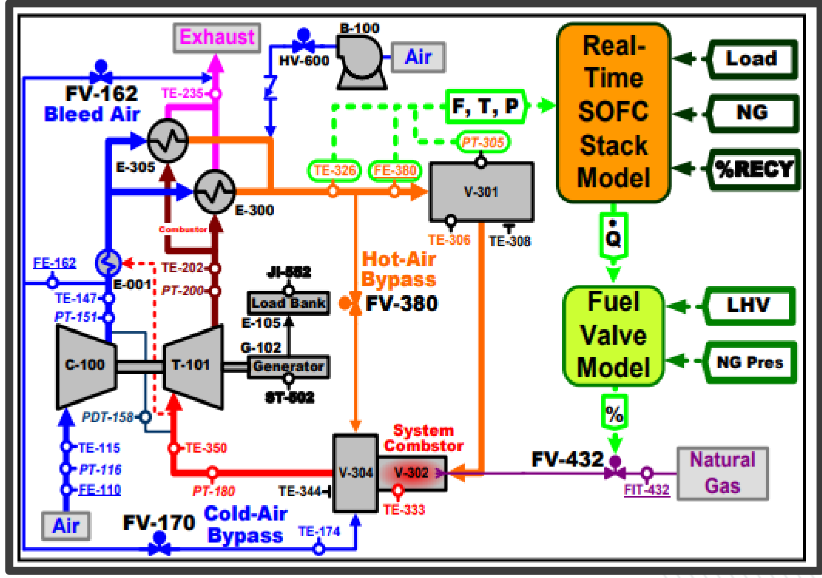


Cyber-Physical Systems

- Cyber-physical systems combine physical and virtual simulation using coupled sensor and transfer systems
- Cost effective means to accurately test dynamic behavior
- CPS can be applied to SOFC/GT hybrids
  - SOFC thermal phenomena readily modeled by numerical simulation
  - Complex turbomachinery fluid dynamics are recreated using actual physical hardware



Expedited Real Time Processing for HyPer



- HyPer is a cyber-physical SOFC/GT system located at NETL
- The physical components include the compressor, turbine, and recuperator
- The simulated components include the numerical SOFC stack model and the fuel valve model
- The expedited real time processing tasks include:
  1. Decreasing the calculation time of the SOFC model to below 5 milliseconds (the limiting response time for turbomachinery control mechanisms) by accelerating the convergence of its electrochemical algorithm through various rootfinding methods
  2. Decreasing the computational burden and enhancing the accuracy by implementing real-time adaptive meshing for the SOFC model
  3. Extra: Re-purposing the SOFC algorithms in a SOFC replica to develop and test new SOFC stack components

Electrochemical Algorithm

Determine Cell Voltage

- Guess overall cell voltage
- Iterate to determine correct voltage for cell load

Determine Local Current Density

- Converge upon current density for discretized cell slice
- Iterate throughout cell to determine overall cell current for set voltage

- Flowchart of iterative process for determining Voltage-Current relationship for fuel cell.
- Both use rootfinding recipes to determine values and subsequently must be solved simultaneously for a given timestep.

$$\mathrm{I(V)} - \mathrm{I_{load}} = 0 \qquad (1)$$

$$\mathrm{V(I)} - \mathrm{V_{guess}} = 0 \qquad (2)$$

- The current produced by the cell is determined by the current generated by each node at a certain voltage

$$I_{Total}(\mathrm{V}) = I_1(\mathrm{V}) + I_2(\mathrm{V}) + \cdots + I_n\,(\mathrm{V})$$

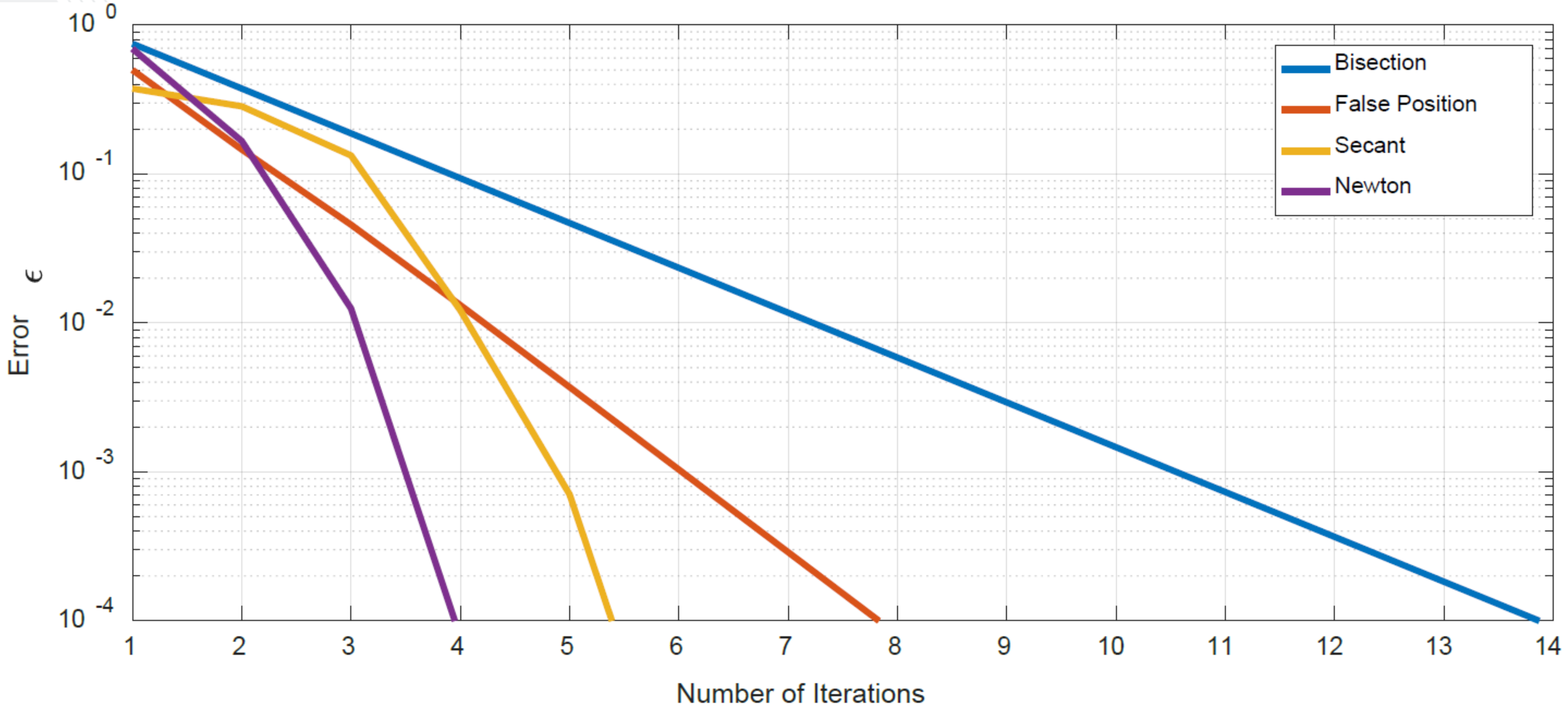
- Used to iteratively solve for cell voltage. (Eqn. 1)

- The voltage of a fuel cell V_{cell} is determined from several factors and is presented as:

$$\mathrm{V}_{cell} = \mathrm{V}(i) = \mathrm{V}_{Nernst} - \eta_{dif} - \eta_{act} - \eta_{ohm}$$

- Used to iteratively solve for local current density. (Eqn. 2)

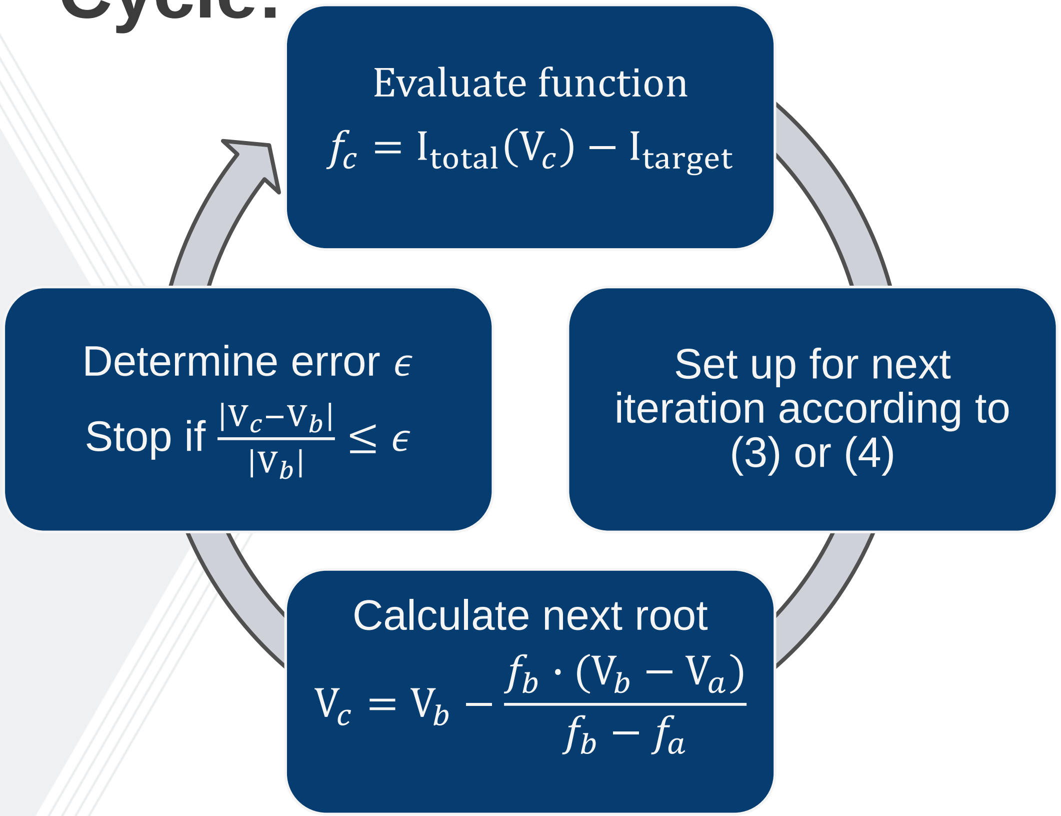
Electrochemical Algorithm Cont. 1



- Relative Error per Number of Iterations for different rootfinding methods.
- Illustrates the potential for optimization by using higher order rootfinding schemes.
  - Bisection requires 14 iterations vs. Secant requiring 6 iterations.
- Study performed on representative problem 0 = ln(x).

Electrochemical Algorithm Cont. 2

Cycle:



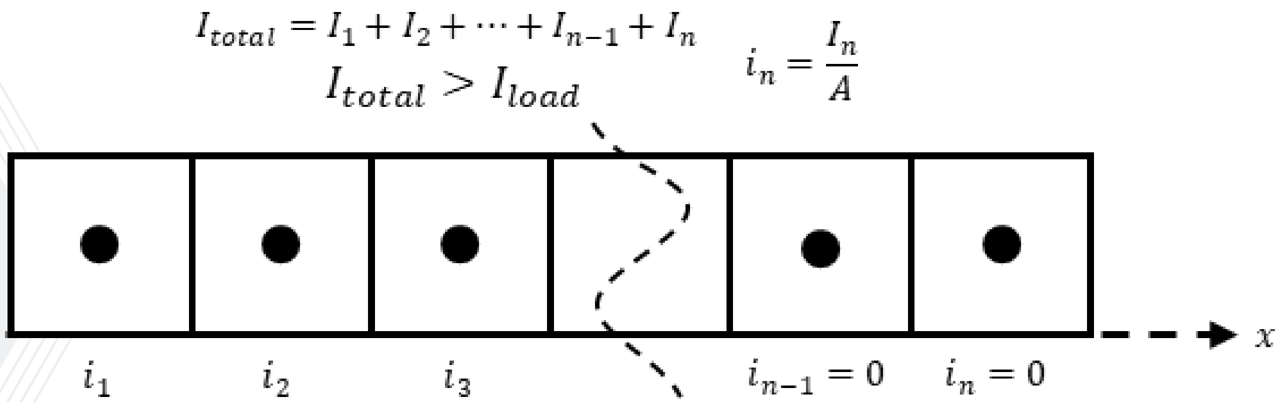
False Position Method:	
If $f_a * f_c < 0$ **Then** $\mathrm{V}_b = \mathrm{V}_c$, **otherwise** $\mathrm{V}_a = \mathrm{V}_c$	(3)
Secant Method:	
$f_a = f_b, \mathrm{V}_a = \mathrm{V}_b$ $f_b = f_c, \mathrm{V}_b = \mathrm{V}_c$	(4)

- Diagram of rootfinding algorithm as implemented in voltage finding scheme.
- Implements current density estimation scheme and uses the results to allow for the implementation of higher order rootfinders. Cycle continues until a converged cell voltage is reached.

Electrochemical Algorithm Cont. 3

Early Termination: (Previous)

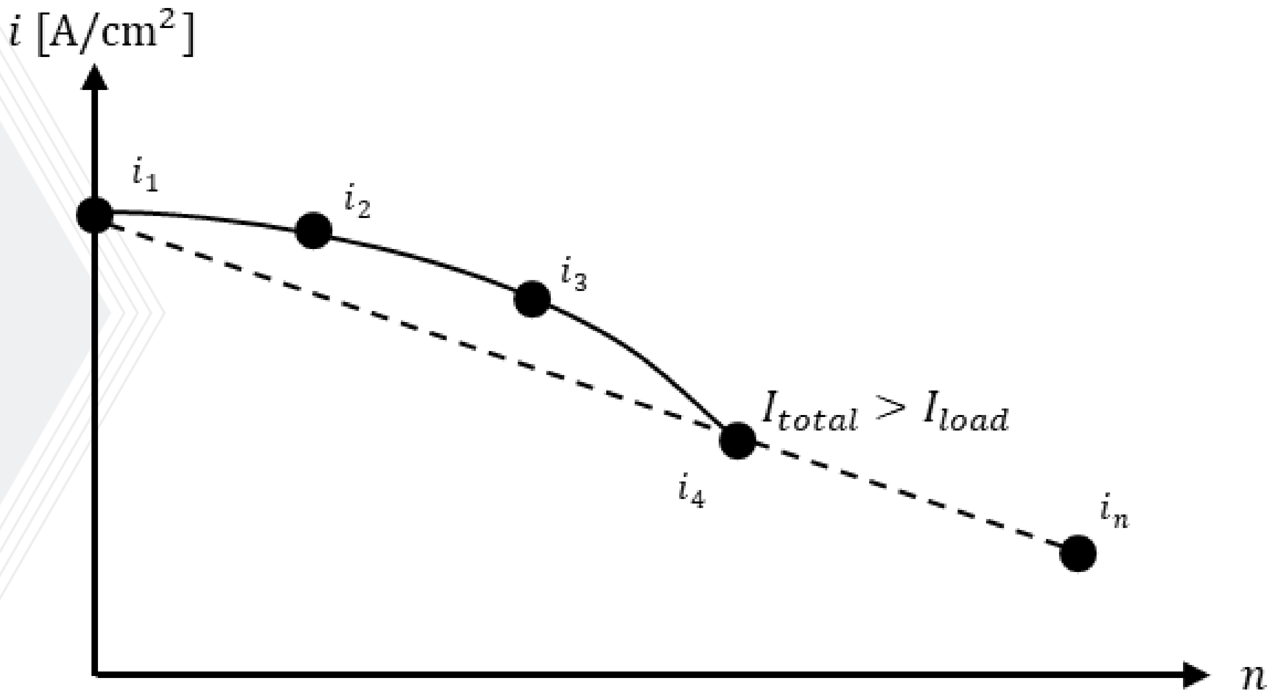
- Schematic of current density algorithm sweeping throughout the computational fuel cell nodes.



- Algorithm uses rootfinder to solve $V(i_n) - V_{guess} = 0$ to find local current density which is converted to the amount of current generated in the slice.
  - The slice currents are then added up to determine the total amount of current generated by the cell.
- If the calculated current becomes higher than the prescribed load the algorithm terminates early.

Early Termination: (New Current Estimator)

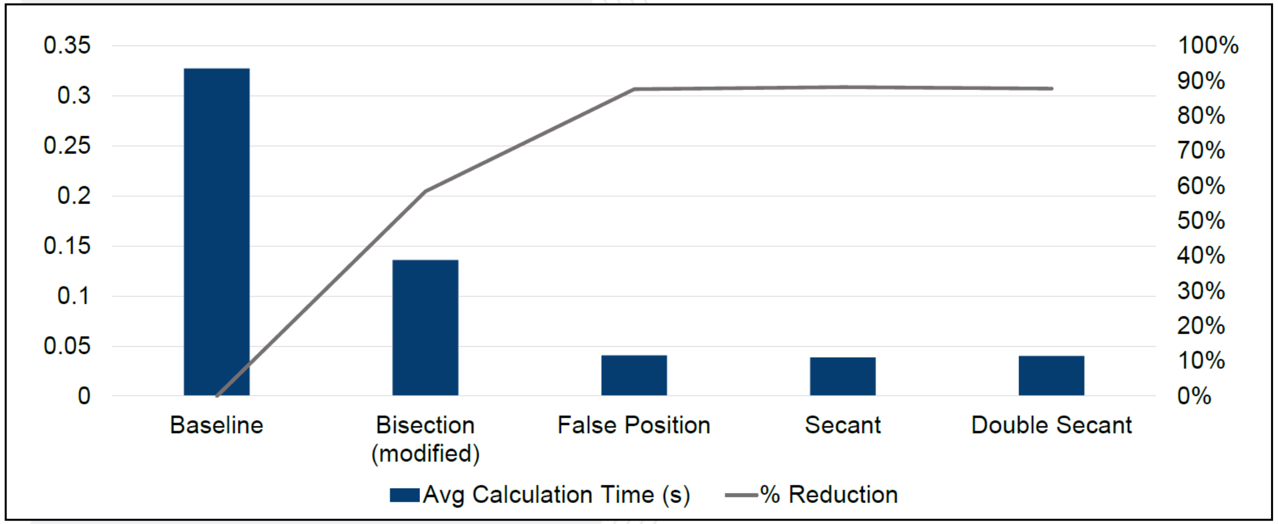
- Schematic of current density algorithm along with the current density extrapolation scheme.



- The current density at the end of the cell is extrapolated using the current density at the first node and the current density when the total cell current surpasses the load current.
- Extrapolation disappears upon full convergence.

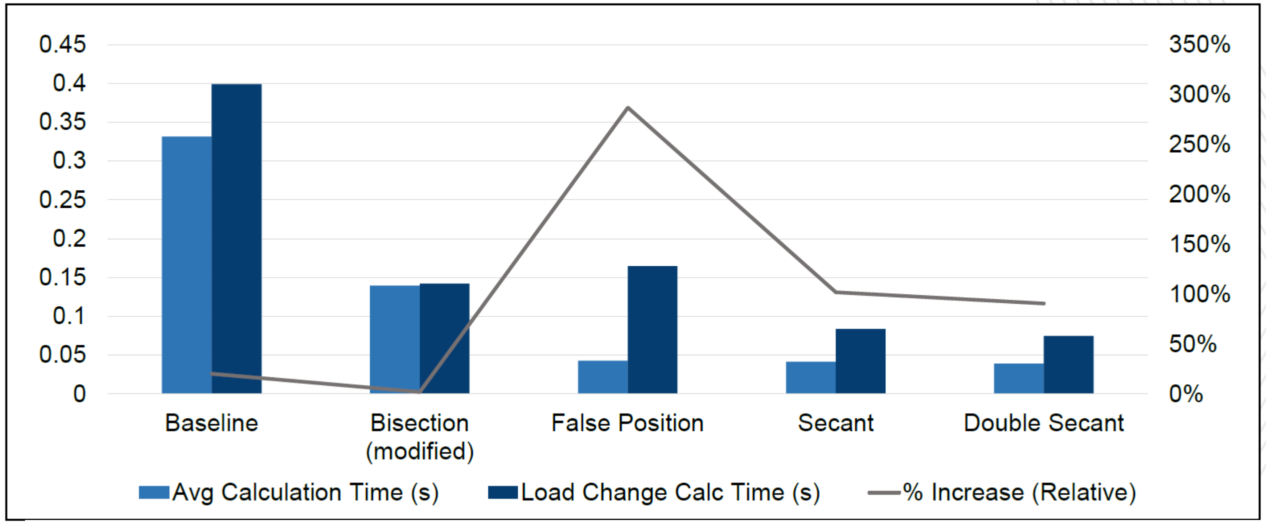
Results of Accelerated Electrochemical Algorithm

Plot of average calculation time in seconds for different rootfinding methods along with the percent reduction in calculation time from baseline.



- Illustrates the drastic reduction in calculation time between baseline code and higher order methods.
- Model parameters are for a load of 250 A, 80% fuel utilization, time steps Δt of 40 ms, and input temperature of 1000 K.

Plot of average calculation time and calculation time during a load step change event, both in seconds, and the percent relative increase in calculation time during the two events for each rootfinding scheme.



- Illustrates the drastic reduction in calculation time between baseline code and higher order methods.
- Higher order methods are more susceptible to rapid changes in inputs.
- Model parameters are for a load of 250 A at a 50% fuel utilization that is then increased to 95% fuel utilization resulting in a final load of 450 A.

Adaptive Meshing/Rediscretization Algorithm

- The SOFC model simulates an SOFC as a collection of nodes distributed in the direction of the flow within the cell
- Previously, the distance between adjacent nodes was held constant and the distance between any two adjacent nodes was the same throughout the entire model
- However, actively changing the distance between nodes allows for a more strategic allocation of limited computational resources by redistributing nodes towards regions with larger gradients and away from regions with smaller gradients

Adaptive Meshing/Rediscretization Algorithm Cont.

- Based on de Boor's algorithm error equidistribution:

$$\int_{x_j^n}^{x_{j+1}^n} M\left(f(x,t^n)\right)dx = \frac{1}{m}\int_{x_1^n}^{x_{m+1}^n} M\left(f(x,t^n)\right)dx$$

- M(f(x,t)) – monitor function, used as a representation of the error
  - Algorithm works by allocating a greater density of nodes where the value of M is higher and vice versa
  - In practice, because f will not be a continuous function but a collection of distinct values at each node, interpolation and numerical integration is required
- Selection of appropriate monitor function M is key
- One approximation, in absence of rigorous error calculation, is scaled arc length, based on local gradient:

$$M(f(x,t)) = \sqrt{1+\alpha^2\left(\frac{\partial f}{\partial x}\right)^2}$$

- α scales how closely solution is based on gradient, to ensure sufficient remaining nodes in areas of flat gradients
- SOFC model is not a single variable f – it is a combination of many tracked variable such as solid temperature, gas temperature, current, voltage, fuel stream composition, etc.

Results of Rediscretization Algorithm

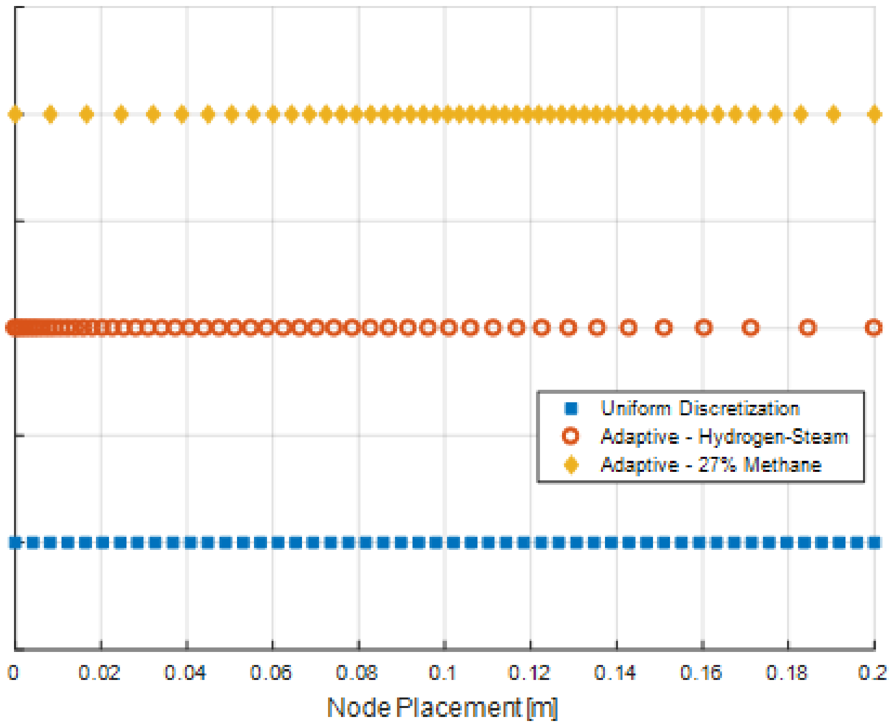


Fig. 1 - Demonstration of Rediscretization upon Fuel Stream Composition Change

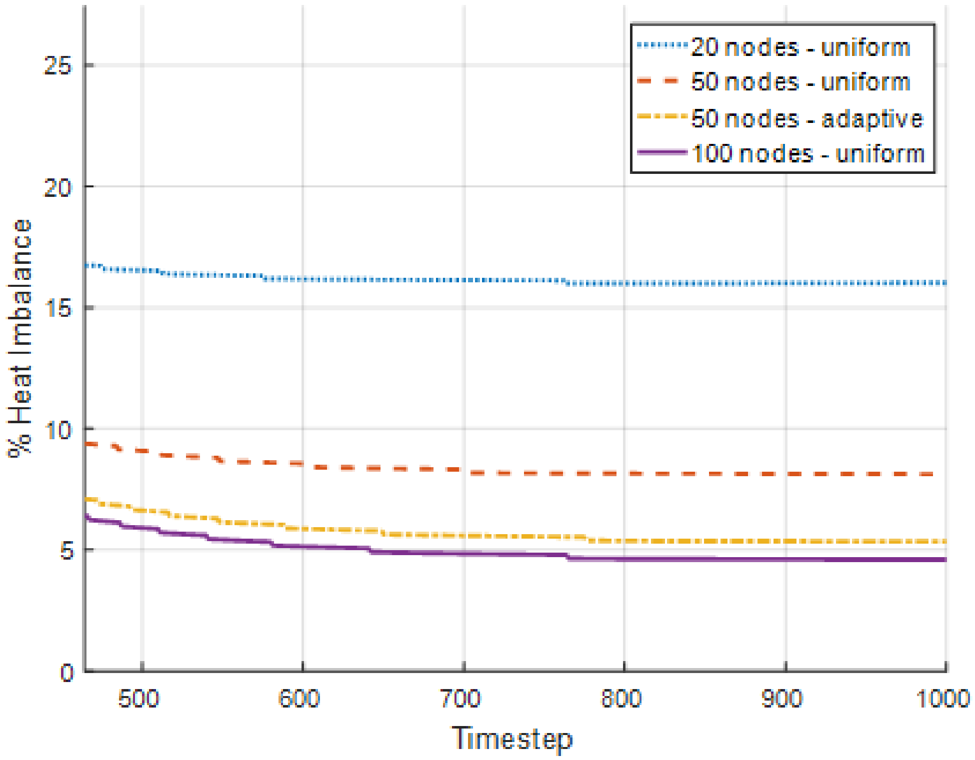
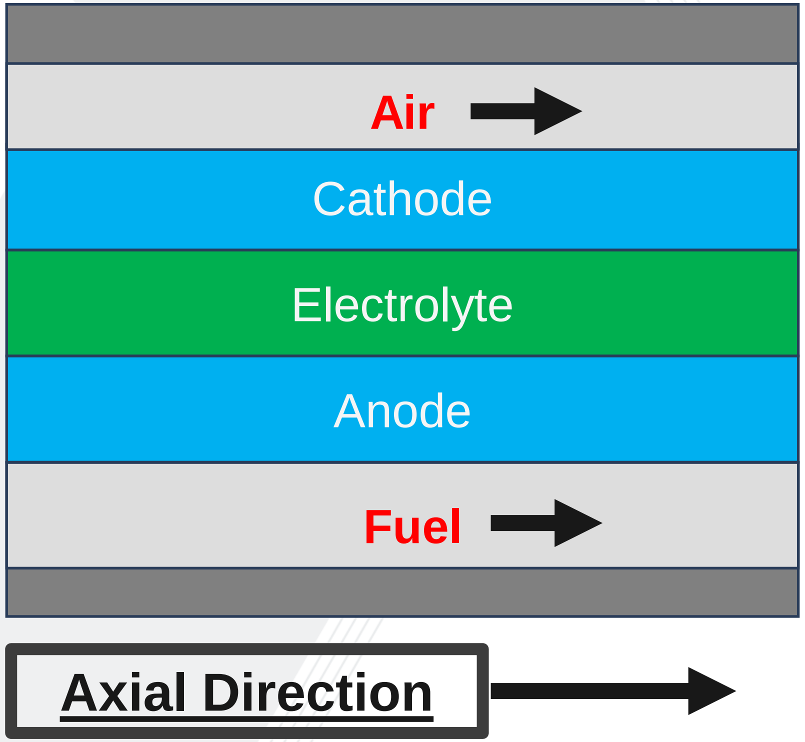
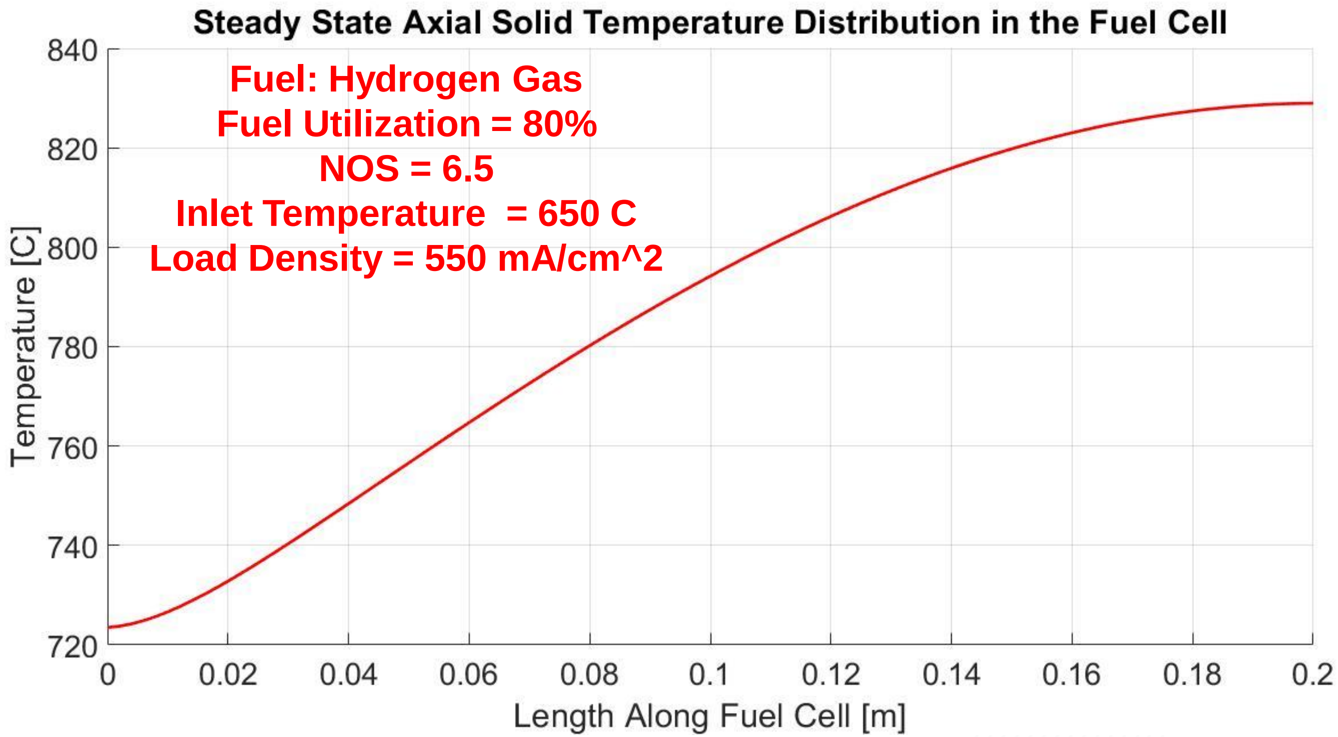


Fig. 2- Demonstration of Steady-State Heat Imbalance Reduction with Adaptive Meshing

Applying the SOFC Model to a Replica

- A large temperature gradient for a standard SOFC in co-flow is shown below
- Such large gradients can limit SOFC dynamic operability
- Current/future work involves using the SOFC numerical model and an SOFC replica to computationally design and test thermally robust SOFC interconnects by observing transient and steady state thermal behavior





Conclusion

- Electrochemical Algorithm Acceleration:
  - Calculation time after code optimization and updated electrochemical solver is below 5 milliseconds
  - With new hardware, calculation time is further decreased, nearing 1 millisecond
- Rediscretization Algorithm:
  - Increases resolution of solution while negligibly increasing the computation time
  - Most of the computation time is still related to the rootfinders in electrochemical algorithm

Georgia Tech Research Institute

Acknowledgement and Disclaimer

Acknowledgment: "This material is based upon work supported by the Department of Energy Award Number DE- FE0030600."

Georgia Tech Research Institute