MATLAB, Simulink, and Cantera for Aerospace Thermodynamic System Modeling

Joshua E. Freeh, Ananda Himansu, Christopher J. Steffen, Jr.,
Robert T. Tornabene, and Xiao-yen J. Wang
NASA Glenn Research Center
Cleveland, OH

Presented to the MathWorks Aerospace and Defense Conference 2006
June 15, 2006

MATLAB, Simulink, and Cantera

MATLAB

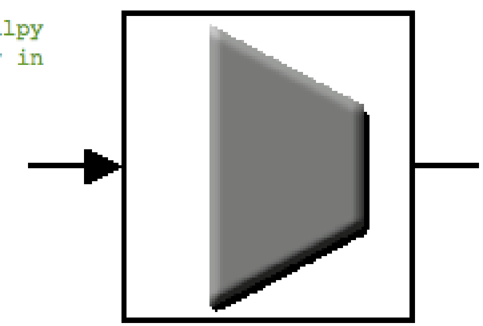
File Edit Text Go Cell Tools Debug

```
function u_out = compressor(u_in)
global nsp data

T_in = u_in(1);
P_in = u_in(2);
W_in = u_in(3);
X_in = u_in(4:nsp+3);

```

Cantera

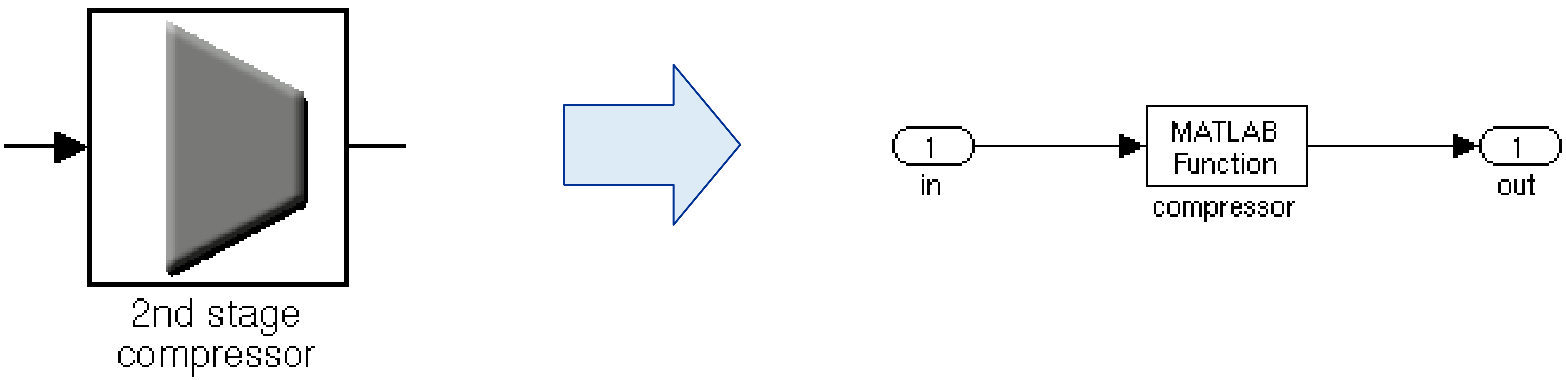
```
set(data.compressor.in.gas,'T',T_in,'P',P_in,'X',X_in);
data.compressor.in.W = W_in;

set(data.compressor.out.gas,'T',T_in,'P',P_in,'X',X_in);
data.compressor.out.W = data.compressor.in.W;

Pt_out = data.compressor.PR * pressure(data.compressor.i
 
Ht_in = enthalpy_mass(data.compressor.in.gas); %enthalpy
S_in = entropy_mass(data.compressor.in.gas); %entropy in
```

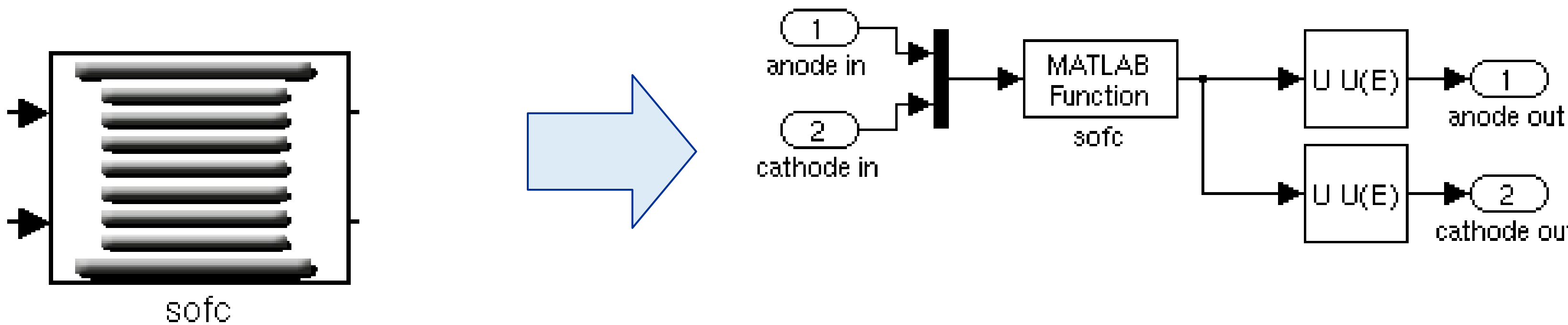
Simulink



2nd stage compressor

Simulink


2nd stage
compressor
1
in
MATLAB
Function
compressor
1
out


sofc
1
anode in
2
cathode in
MATLAB
Function
sofc
U U(E)
U U(E)
1
anode out
2
cathode out

…and MATLAB again

- `fmincon, fminbnd, fsolve`
  - Overall optimizers and nonlinear equation solvers
  - Minimize system mass or ensure conservation equations
- `gatool, patternsearch`
  - Non-gradient optimizers, though not necessary for current UAV model

Also…

- structures
  - Minimize variable handling and clarifies code
  - for example, `compressor.in.W` (or `.gas`, `.PR`, `.pwr`)

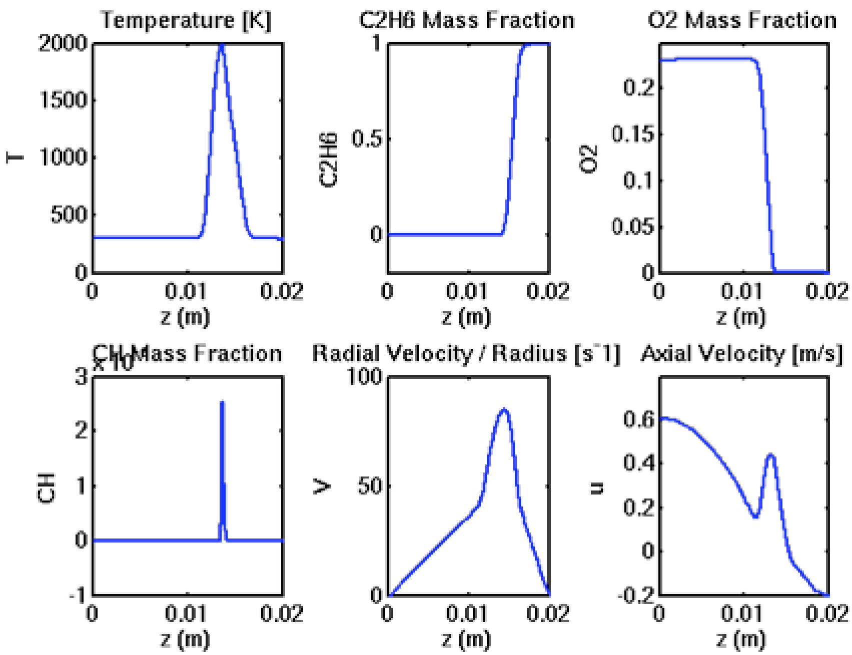
Cantera
(www.cantera.org)

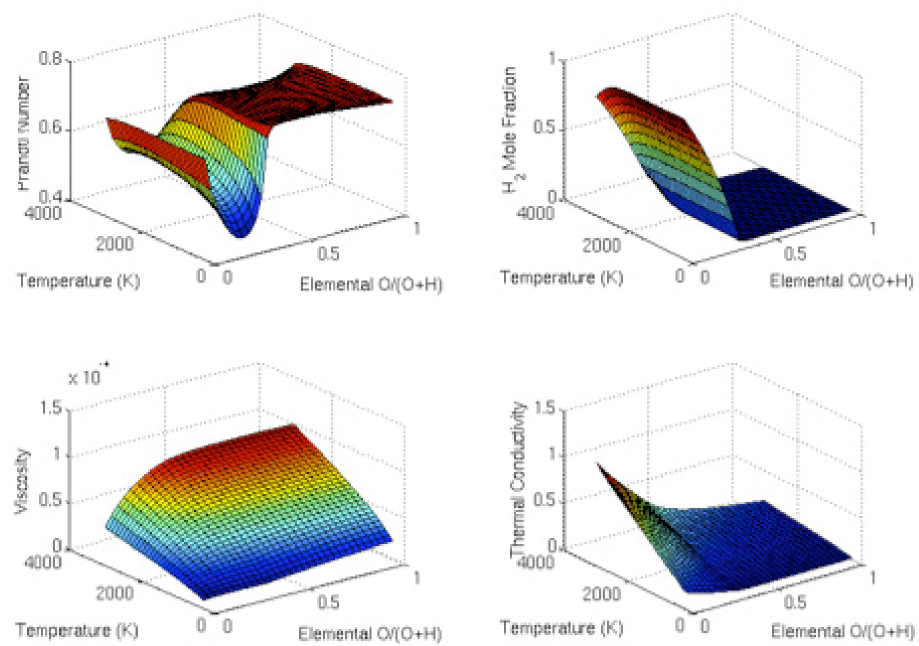
- Developed by David G. Goodwin, Professor of Mechanical Engineering and Applied Physics at California Institute of Technology

- Open source chemical reaction code
  - Written in C++
    - Compatible with MATLAB, Python, Fortran

- Capabilities
  - Thermodynamic and transport properties
  - Chemical equilibrium
  - Chemical kinetics
  - Electrochemistry
  - Pure substance equations of state
  - Others…

Cantera


Temperature [K]
T
z (m)
C2H6 Mass Fraction
C2H6
z (m)
O2 Mass Fraction
O2
z (m)
CH Mass Fraction
CH
z (m)
Radial Velocity / Radius [s^-1]
V
z (m)
Axial Velocity [m/s]
u
z (m)


Prandtl Number
Temperature (K)
Elemental O/(O+H)
H2 Mole Fraction
Temperature (K)
Elemental O/(O+H)
Viscosity
Temperature (K)
Elemental O/(O+H)
Thermal Conductivity
Temperature (K)
Elemental O/(O+H)

Cantera

MATLAB

```
gas = GRI30;
set(gas,'T',300.0,'P',OneAtm,'X','CH4:1,O2:2,N2:7.52');
equilibrate(gas,'HP');
disp(gas)
```

Python

```
from Cantera import *
gas = GRI30()
gas.set(T = 300.0, P = OneAtm, X = 'CH4:1,O2:2,N2:7.52')
gas.equilibrate('HP')
print gas
```

C++

```
#include "Cantera.h"
#include "GRI30.h"
#include "equilibrium.h"

main() {
    GRI30 gas;
    gas.setState_TPX(300.0, OneAtm, "CH4:1,O2:2,N2:7.52");
    equilibrate(gas, "HP");
    cout << gas;
}
```

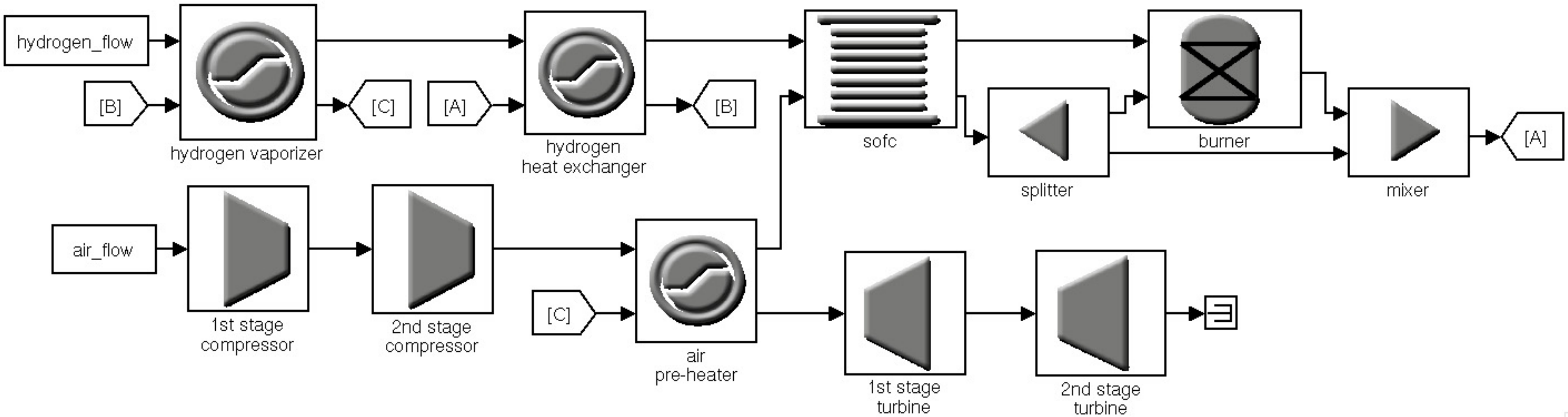
Objective

- Design a representative hybrid solid oxide fuel cell power system for high-altitude, long-endurance flight

- Understand the effect of major variables on the system
  - system level variables: altitude, power level, duration
  - component level variables: efficiencies, fuel cell resistivity

- Optimize system design
  - Current figure of merit: minimize mass

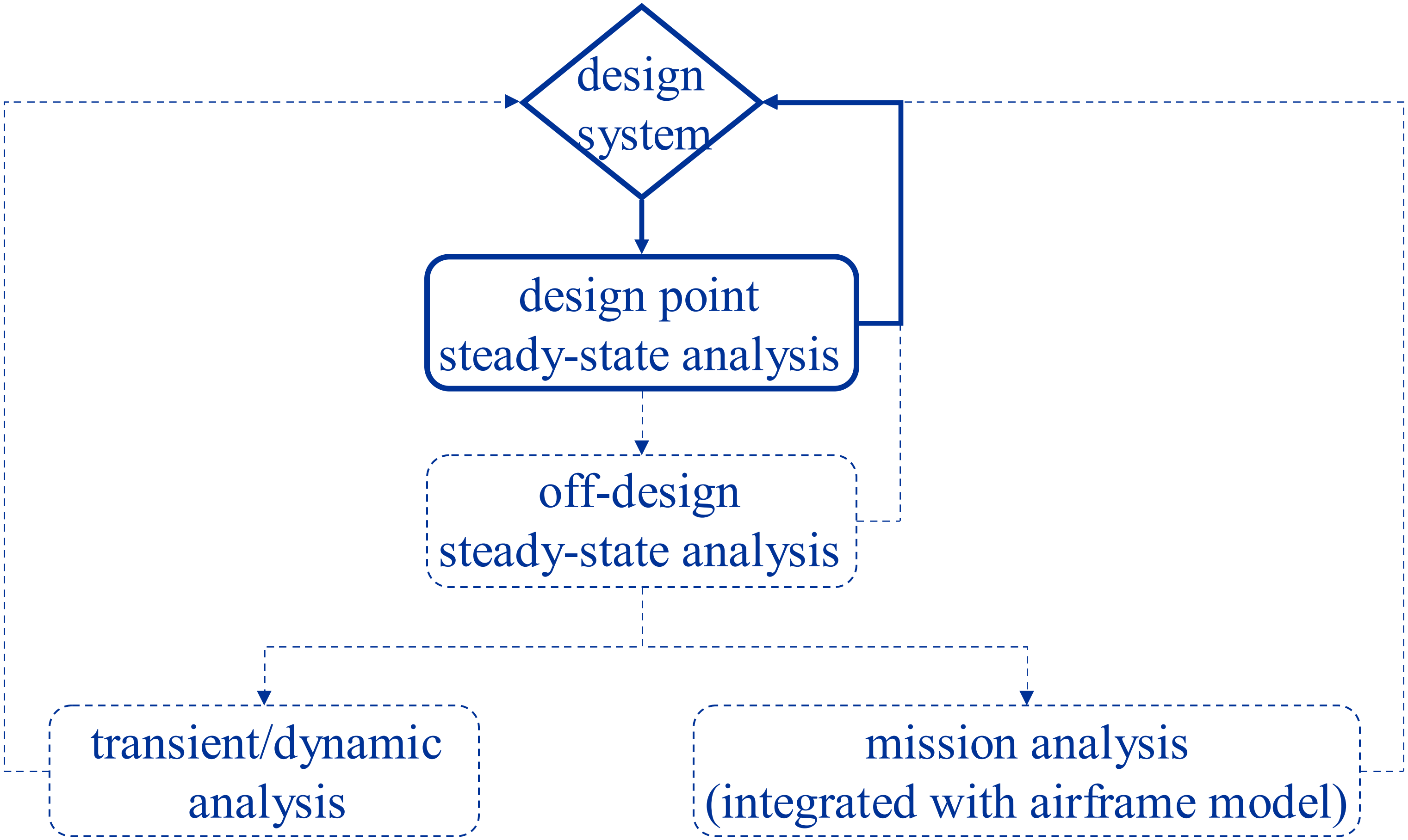
UAV power system



UAV power system specifications

- Baseline system
  - 21 km altitude (68,897 ft)
  - 50 kW net electrical power
  - η_{ad} = 0.8 for compressor stage, 0.85 for turbine stage
  - 2 stage radial turbomachinery designs
  - Planar solid-oxide fuel cell
    - 700°C inlet temperature, 1 atm inlet pressure
    - 100°C temperature increase across stack (air-cooled)
    - Maximum of 25 cells per stack, 25 cm^2 single cell active area
- Variations:
  - Altitude: 16 km
  - Power: 20 kW
  - Sensitivities on compressor and turbine η_{ad}, fuel cell ASR and stack design

Assumptions and Limitations



Past studies

Previous work (most undocumented)

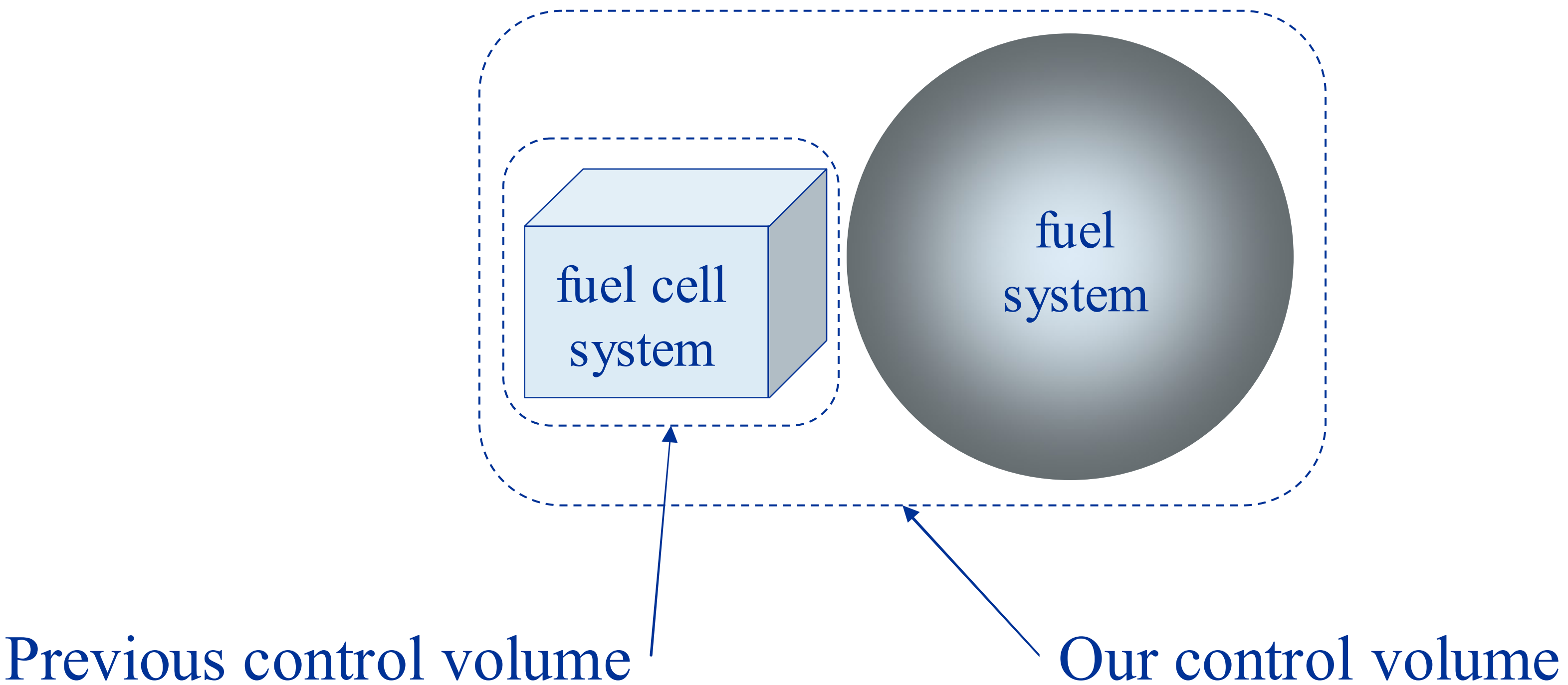
Minimize

fuel cell system mass

Our work

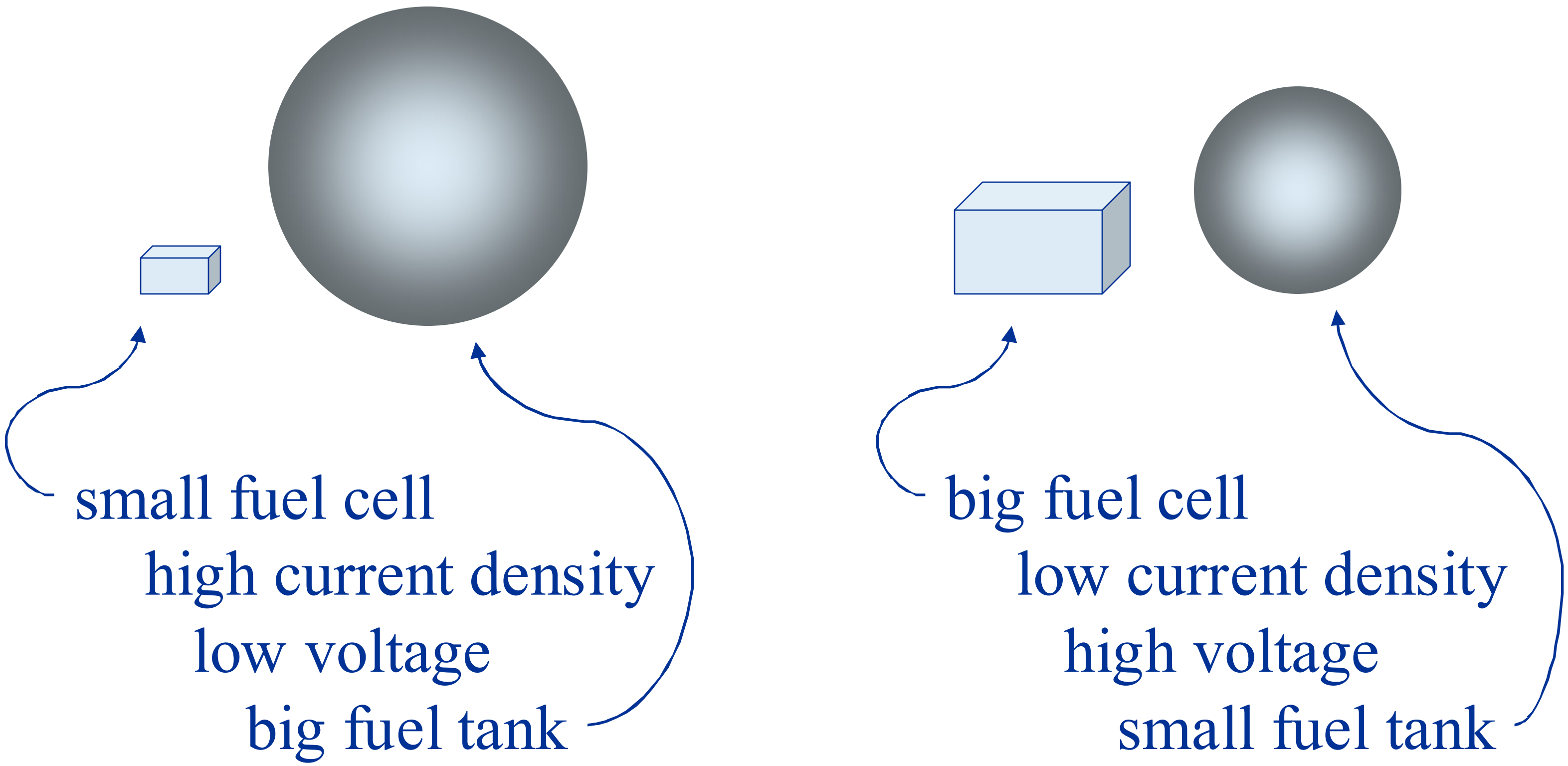
Minimize

(fuel cell system + fuel system) mass



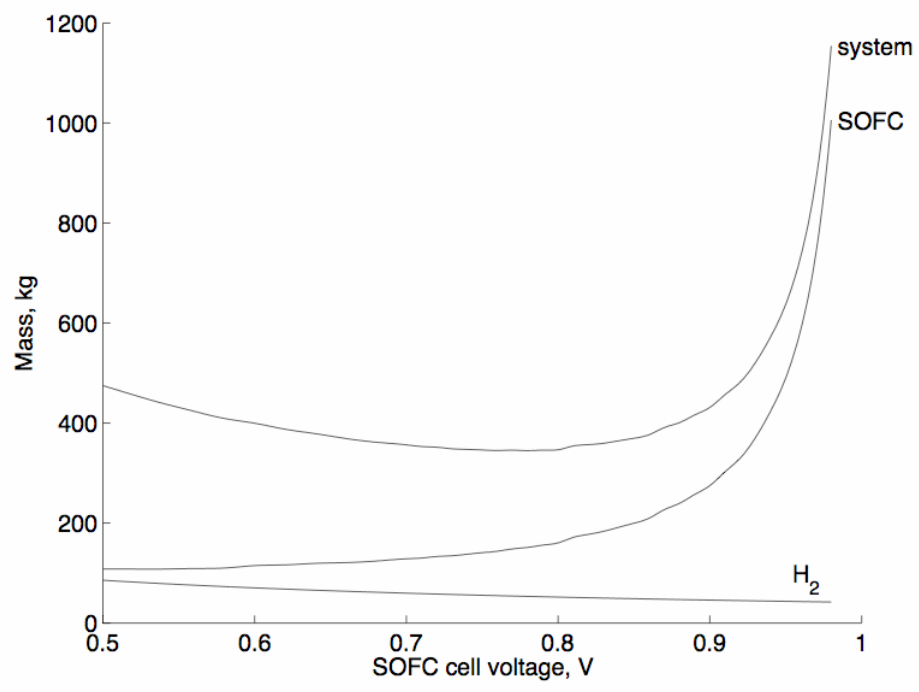
Qualitative (pre)analysis

for constant fuel cell power,

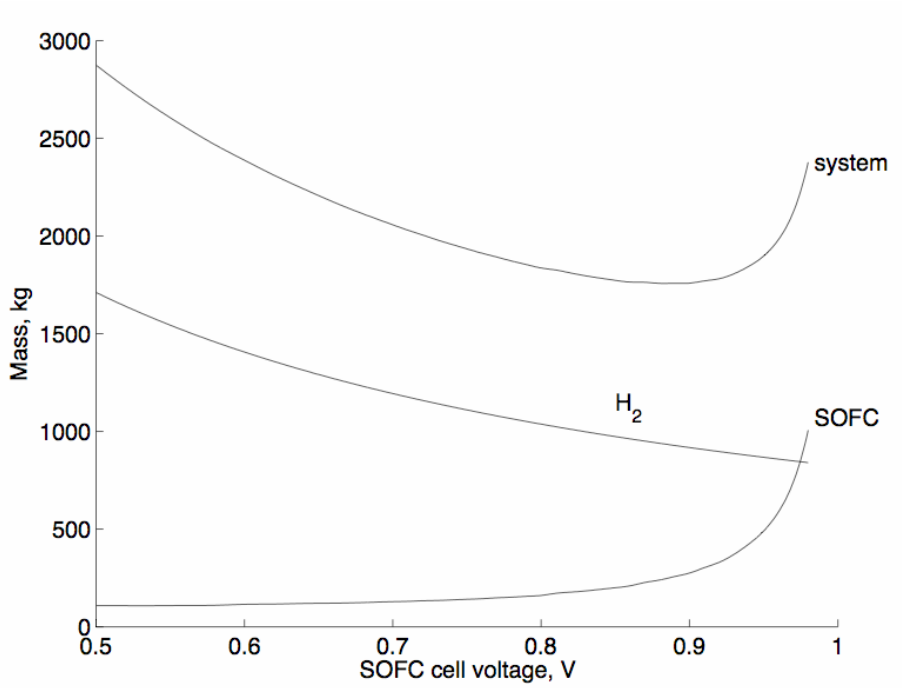


∴ there is a optimum system that minimizes mass/volume

Effect of mission duration



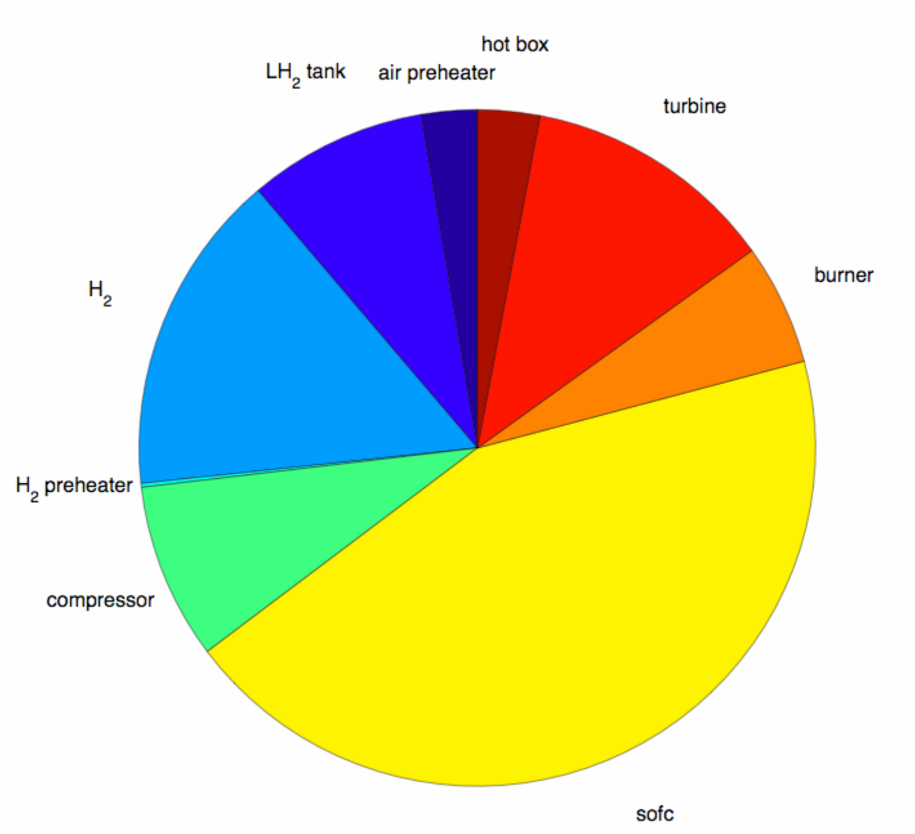
1 day mission



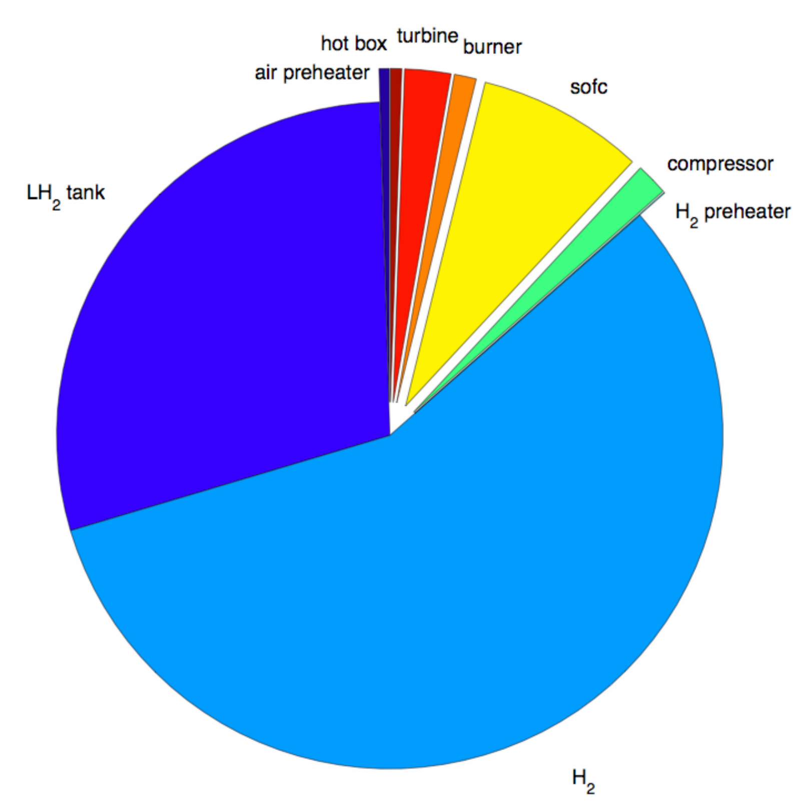
20 day mission

Another view of mission duration

(using optimum solutions from previous plots)

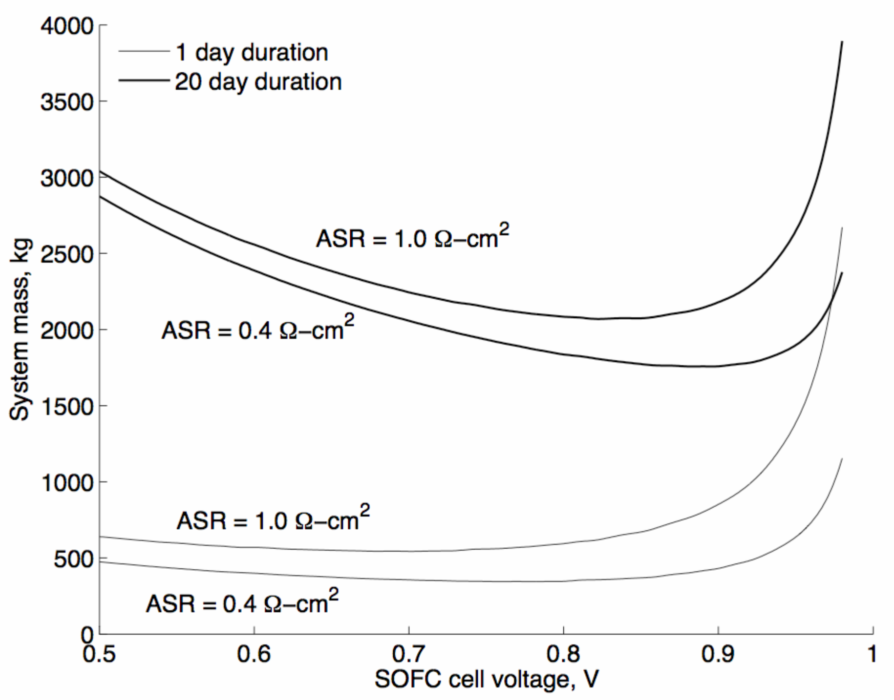


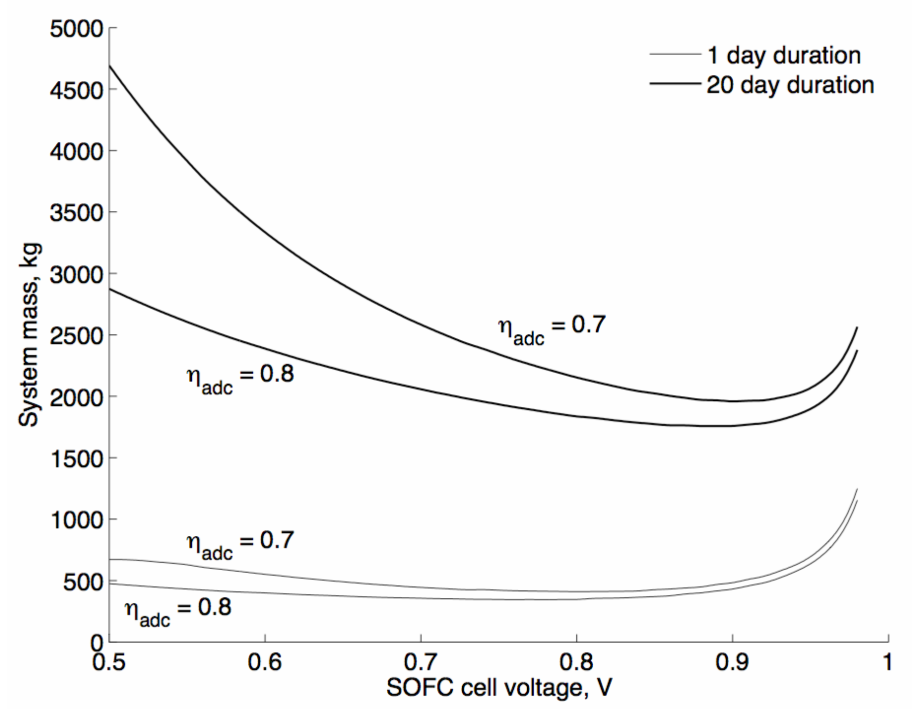
1 day mission



20 day mission

Sensitivity studies





Conclusions

- Combination of MATLAB, Simulink, and Cantera
  - Flexible and customizable interface
    - Currently using similar approach for lunar oxygen production system analysis
  - Scalable analysis capability
    - Cantera offers much more than currently being used here
    - MATLAB/Simulink as well
      - other toolboxes
        - » Statistics, SimPowerSystems
      - Also full capability of Simulink
        - » Dynamic and transient modeling

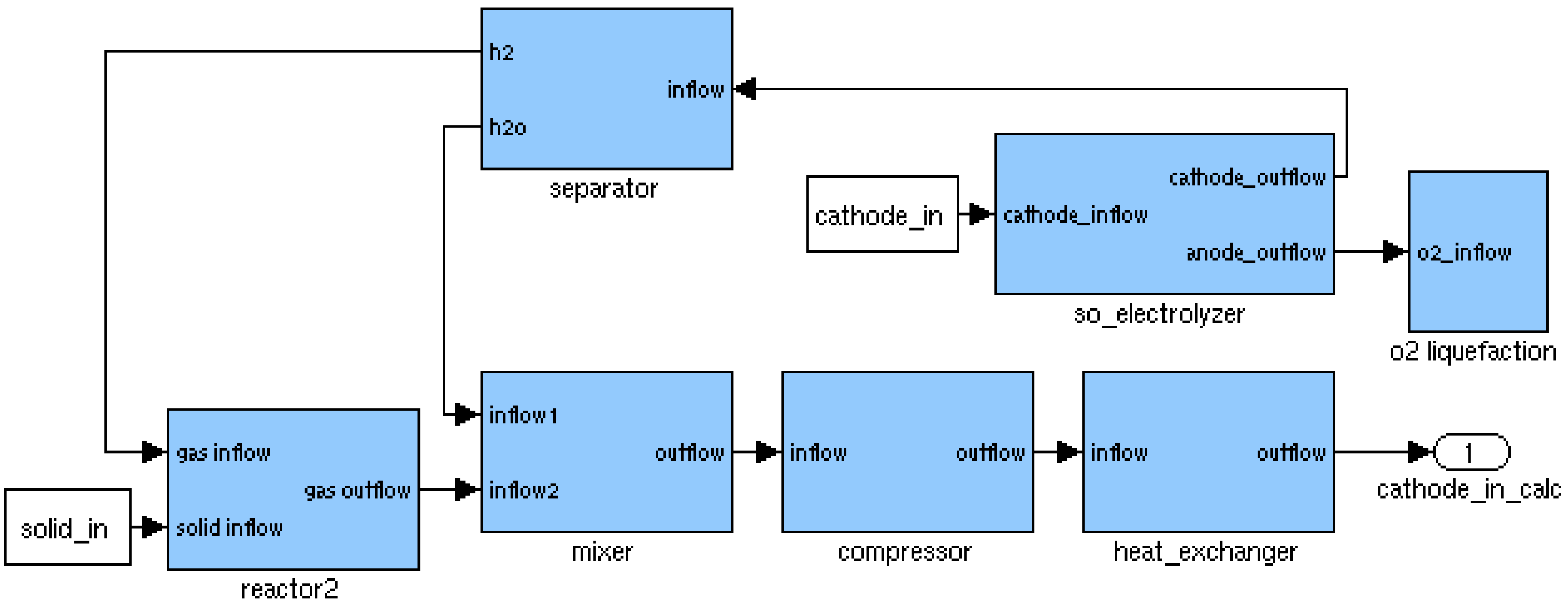
- UAV fuel cell system is possible when analyzed at the correct system level
  - especially for this application (high altitude, long endurance)

More information on this work

- Results will be presented in more detail on June 19
  - ASME Fuel Cell Science, Engineering, and Technology Conference in Irvine, CA
    - Paper # FUELCELL2006-97095

- Also available as a NASA Technical Memorandum
  - NASA TM-2006-214328
  - (will be online soon at http://ntrs.nasa.gov)

Lunar O_2 production model



Recommendations

- There is a better way to integrate these codes *(though we haven't found it yet)*
- Simulink can only pass numerical signals
  - Structures (with different data types) would help
  - Cantera objects (pointers?) would be even better
- More focus on Simulink as a steady-state tool
  - e.g. more flexibility and power with algebraic loop solver
    - Currently using MATLAB's `fsolve` to balance system .mdl
- Most of the current analysis time is interacting between three codes
  - Simulink-native (S-functions?) Cantera would help
  - Better S-function documentation or examples would help move this along

Thanks!

Contact information

Josh Freeh

NASA Glenn Research Center

(216) 433-5014

joshua.e.freeh@nasa.gov

Recommendations

- Cantera/mex interface does not always build correctly
  - May be a Cantera issue
  - and/or mex issue
  - and/or Macintosh issue (different versions of gcc, Tiger, etc.)

- Macintosh version can be much better

  (although some of these issues go beyond the scope of The MathWorks)
  - Can be much more than just a Unix/X11 port
  - Native Aqua, Quartz and other Mac-specific features
  - Use of AltiVec capabilities
  - Better Excel interface
  - Automatic local parallelization (for multi-processor machines)
  - XGrid