Logistics

1. Start time: 8:30AM / end time: 11:30AM
2. Laura / Wensi will be at the site at 7:30AM.
3. Can couple more people be there by 8AM?
4. For those who will show the Argonne National Lab model
   - Mention that the model was developed by ANL as part of Autonomie.
   - Include text on the top level of the model that says the same
   - Direct any questions about the ANL model and Autonomie to ANL. Larry Michaels from ANL will be in the audience.
5. Which laptops will we present from?
   - Pre-seminar loop / Laura / Wensi / Pete / Tom Egel using Tom’s laptop
   - John using John’s laptop (assumption)
   - Kerry / Tom P. using Kerry’s laptop
6. Other details:
   - Screen resolution
   - Web cam
7. Print workshop info and eval form (Wensi will do that)
Model-Based Design for Hybrid Electric Vehicle Development

February 18, 2010

Novi, Michigan
The MathWorks at a Glance

- **Headquarters:**
  Natick, Massachusetts US

- **US:**
  California, Michigan, Texas, Washington DC

- **Europe:**
  France, Germany, Italy, Spain, the Netherlands, Sweden, Switzerland, UK

- **Asia-Pacific:**
  Australia, China, India, Korea

- Worldwide training and consulting

- Distributors in 25 countries

Earth’s topography on an equidistant cylindrical projection, created with MATLAB and Mapping Toolbox.
Key Capabilities Drive The MathWorks Business

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>MATLAB</td>
<td>Data Analysis and Algorithm Development</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Control design  • Signal processing</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Optimization  • Statistics</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Image processing  • Financial modeling and analysis</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Application deployment  • Student version  • Distributed and parallel computing</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Simulink</td>
<td>System Modeling and Simulation</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• DSP designs  • Communication systems</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• State charts  • Physical modeling</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Discrete-event modeling  • Video processing</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Rapid prototyping and HIL  • Embedded software  • DSP software  • VHDL/Verilog</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Test, Verification, Validation</td>
<td>Automatic Code Generation</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Test and measurement  • Model checking  • Code verification</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Founded in 1984
The MathWorks Today

- Revenues ~$500M in 2008
- Privately held
- More than 2,000 employees worldwide
- Worldwide revenue balance:
  45% North America, 55% international
- More than 1,000,000 users in 175+ countries
Key Industries

- Aerospace and Defense
- Automotive
- Biotech and Pharmaceutical
- Communications
- Education
- Electronics
- Energy and Power Production
- Financial Services
- Industrial Automation and Machinery
- Semiconductor
Core MathWorks Products

MATLAB®

The leading environment for technical computing

- The *de facto* industry-standard, high-level programming language for algorithm development
- Numeric computation
- Data analysis and visualization
- Toolboxes for signal and image processing, statistics, optimization, symbolic math, and other areas
- Foundation of MathWorks products
Core MathWorks Products

**SIMULINK**

The leading environment for modeling, simulating, and implementing dynamic and embedded systems

- Linear, nonlinear, discrete-time, continuous-time, hybrid, and multirate systems
- Foundation for Model-Based Design, including physical-domain modeling, automatic code generation, and verification and validation
- Open architecture for integrating models from other tools
- Applications in controls, signal processing, communications, and other system engineering areas
Model-Based Design for Hybrid Electric Vehicle Development

February 18, 2010

Novi, Michigan
SAIC to develop 'green' cars

By Joyce Pan | 2009-7-4 | NEWSPAPER EDITION

SAIC MOTOR Corp yesterday said it plans to invest 12 billion yuan (US$1.76 billion) in the next three to five years to develop new-energy sedans and auto parts.

The nation’s largest car maker signed an agreement with the Shanghai government, which has pledged to offer favorable policies, product research and testing, human resources and government procurement to help SAIC accelerate the marketing of new-energy vehicles.

The investment will divvied up between 41 projects under its blueprint for environmentally friendly vehicles and SAIC plans to pour 2.1 billion yuan into commercializing these cars this year, it said.
Engineering Challenges

- Multitude of design choices but lacking past designs as references
  - System architecture, component sizing, ...
- Squeezing out efficiency improvements across the system with brand new technologies
  - Batteries, power electronics, traction motors …
  - More than an embedded software challenge
- Fast design iterations
  - Embedded SW developed in parallel with HW
- Getting verification done
  - Complex interaction among subsystems
  - Large amount of embedded software developed in a short time

“Creating well over a million lines of code on a compressed schedule, while ensuring that the control system meets GM’s high quality standards and complying with all legislative regulations and safety standards, was a tremendous challenge,”

- Greg Hubbard, Senior Manager, Hybrid and Electric Drive Controls, GM.
Model-Based Design

Requirements and Specifications

- Environment models
- System Behavior models
- Components models

Design

- Environment models
- Components models
- Algorithms

Implementation

- Algorithms
- Generate
- Generate

Test and Verification

- Early Verification
  - Analyze subsystem interactions
  - Fix design errors before implementation
  - Parallel SW & HW development
  - Reduce # of iterations and rework

- Automatic code generation
  - Speed up design iterations
  - Eliminate translation errors

System Level Simulation
- Establish sys. architecture
- Optimize performance
- Determine key control parameters

- System Level Simulation
- Early Verification
- Implementation
- Test and Verification

Languages:
- C, C++
- VHDL, Verilog
- MCU, DSP, FPGA, ASIC
Successful Deployment of Model-Based Design for Hybrid Vehicle Development

Tesla Roadster: 1st electric sports car
- Reduced project duration by years
- Completed within limited budget
- Decreased reliance of physical prototypes

Dongfeng EV: battery management system:
- Battery model key to early verification
- Finished early with limited resources
- Generated 100% of application code automatically

GM: Two-Mode hybrid powertrain
- Met aggressive delivery date
- Enabled worldwide collaboration
- Reused design across product lines
Seminar Agenda

- HEV design optimization
- HEV system level simulation
  - motor, generator and battery examples
- Break
- Model verification:
  - energy management controller example
- Controller implementation:
  - motor controller example
Questions before we dive into the details?
HEV System-Level Optimization

Argonne National Labs PHEV System-Level Simulation Model

Numerical Optimizer

Fuel Consumed On 32 Real-World Drive-Cycles

Financial Calculations

Net Consumer 15 yr Fuel Savings ($)

IC Engine-On Threshold

IC Engine-Off Threshold

Battery Storage Capacity

Battery Max Discharge Power
Motivation for HEV System Optimization

- Fuel Economy and Emission Requirements Are Stringent
- HEV’s Provide Energy Source Flexibility to Meet Requirements
- HEV Flexibility Increases Powertrain Complexity Dramatically
- Until Recently HEV Focus Has Been Focused Mostly on Integration
- System-Level Optimization Is Now Required In Addition to Integration
- System-Level Optimization Requires Fast Models via Parallel Computing
- System-Level Optimization Requires Robust Global Optimizers

*MathWorks Has System-Level Optimization Capability*
Maximize Customer Fuel Cost Savings By Optimizing
IC Engine Control Parameters and Battery Hardware Parameters
HEV Optimization Problem-Solving Approach

- Use Rapid Accelerator for Execution Speed on 32 Drive-Cycles
- Use Parallel Computing To Make Execution Speed Scalable and Controllable
- Use Direct Search Optimization for Robustness to Local Minima

Search Systematically Steps Through The Search Parameter Space
HEV Optimization Problem-Solving Approach

1. Initialize Optimization Parameters
2. Generate N Parameter Variations With Pattern Search
3. Run 32 Real-World Drive-Cycle Simulations Per Parallel Computing Worker For Each of The N Variations
4. Size Pattern Search Mesh Smaller Than Tolerance? (No → Go Back, Yes → Report Results)

Typical Run Results: 4hrs on Quad-Core PC, ~1000 Simulations
HEV Optimization Vehicle Assumptions

- 120 kW motor
  - steady state efficiency map with speed & torque axes
- 110 kW gasoline engine
  - steady state fuel rate map with speed & torque axes
- 50 kW generator
  - steady state efficiency map with speed & torque axes
- 16 kWh Li battery with PNGV/ANL/Saft cell data (nominal)
  - Internal resistance, OCV as a function of soc
- Vehicle
  - Cd 0.31
  - Frontal area 2.06 m^2
  - Test mass 1350kg
HEV Optimization Financial Assumptions

- 300 days a year, 15 yrs of vehicle life (240,000 km)
- Battery life 2000 cycles
  - Steady performance for first 6 yrs
    (4% degradation every year after that)
- Battery cost estimate DOE goals
  - Battery replacement after 2000 cycles is optional

- Conventional vehicle 0.0789 litre/km
- PHEV actual fuel consumption obtained from simulation over real world drive cycles

- Rate of returns is assumed to be 7% for NPV calculation
- Electricity cost 0.1 $/kWh
- Gasoline cost 0.79 $/litre = $3.24/gal
HEV Optimization Results

- Optimal Parameters Achieved in 2.5hrs on Parallel 8-core multi-threaded PC
- Optimization Increased Consumer Savings from $3450 at Nominal Design to $5234

Consumer Savings At the Optimal Design Settings Below:

**Optimal Hardware**

- Battery spec range
- (18, 40)

**Optimal Controls**

- ON threshold = 80 kW
- OFF threshold = 10 kW
HEV Optimization Conclusions

- Fast-Running Large-Scale Models Required for HEV System-Level Optimization

- A Combination of Optimization, Plant Modelling, and Parallel Computing Tools Make Large-Scale HEV Optimization Practical for Everyday Engineers