Modeling control elements with an outlook toward hardware-in-the-loop (HIL) operation

2013 Propulsion Control and Diagnostics Workshop
Cleveland, OH

Alicia Zinnecker, Ph.D.
N&R Engineering

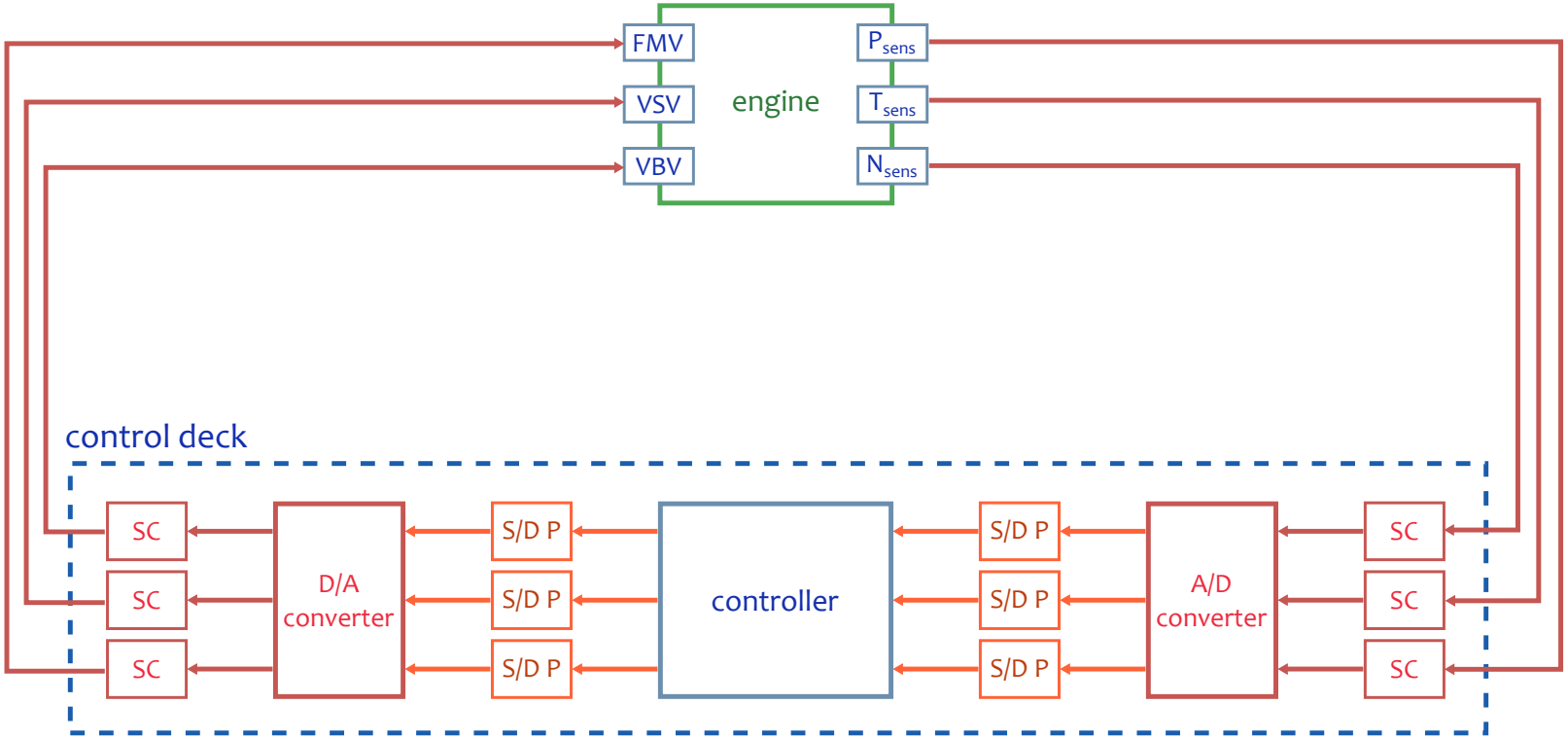
Controls and Dynamics Branch
NASA Glenn Research Center

December 11, 2013

Outline

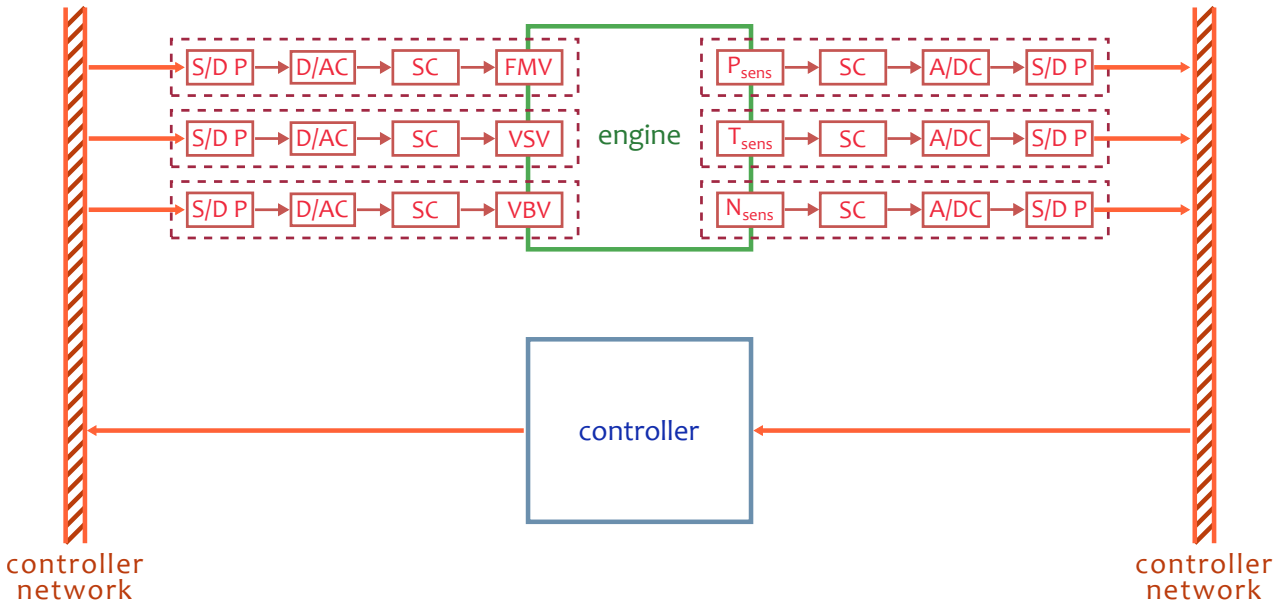
1. Migration from centralized to distributed control
   - Designing toward a hardware-in-the-loop (HIL) system
2. Models of control elements
   - Transducer models
   - Benefits
3. Modeling a distributed control scheme
   - Challenges
   - Alternative approach: Parallelization
4. Conclusions and outlook

Centralized control



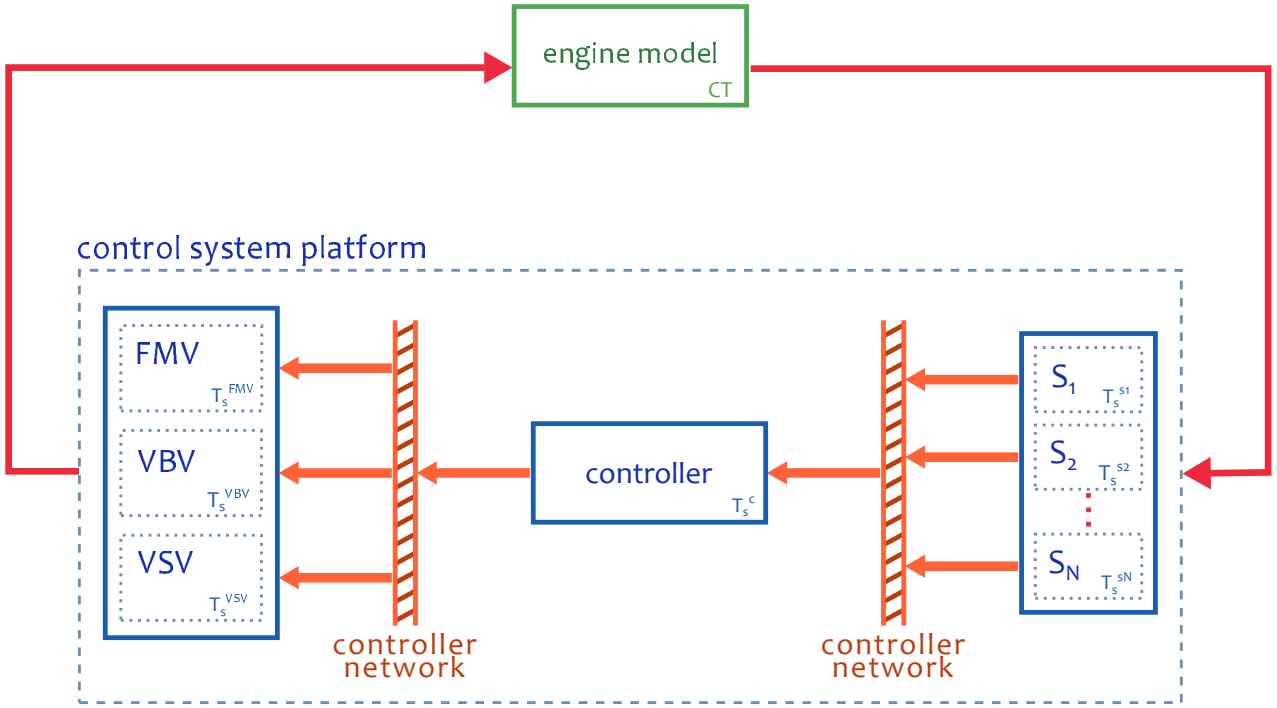
- **Signal processing** and **control calculations** handled by **centralized** control deck (FADEC)
- Sensor measurements and actuator commands are **analog signals** traveling between hardware and controller

Distributed control



- **Signal processing** functionality moved onto **smart transducers**
- **Digital data** replaces analog signals carrying data between hardware and controller
  - Signal susceptibility to noise **reduced** with digital data
  - Replacing hardware accompanied by **minimal effects** on overall functionality
  - **Network** connecting sensors, actuators, and controller becomes important component

Developing a hardware-in-the-loop (HIL) system



- *Computational* engine model captures **dynamics of engine**
- Control system implements **algorithm meeting design goal** of closed-loop system
- Controller network *must be modeled* to ensure **accurate simulation** when hardware prototypes are not connected in the loop

Developing a hardware-in-the-loop (HIL) system

- At the *simulation* level, added fidelity to control components (sensors, actuators) is **necessary**
  - Functionality of models reflects **distributed nature** of control scheme
  - **Numerical precision** of data used by the controller should match that of real hardware, controller network
  - High accuracy allows for comparison between computation results and those collected with hardware connected in simulation loop
- Simple control hardware replaced by smart transducers
  - Off-loads some processing from control deck
  - **Digital** data transfer replaces **analog** signal transmission
  - Simulink® library (under development) contains blocks that may be combined, configured to construct **high-fidelity hardware models**

Developing a hardware-in-the-loop (HIL) system

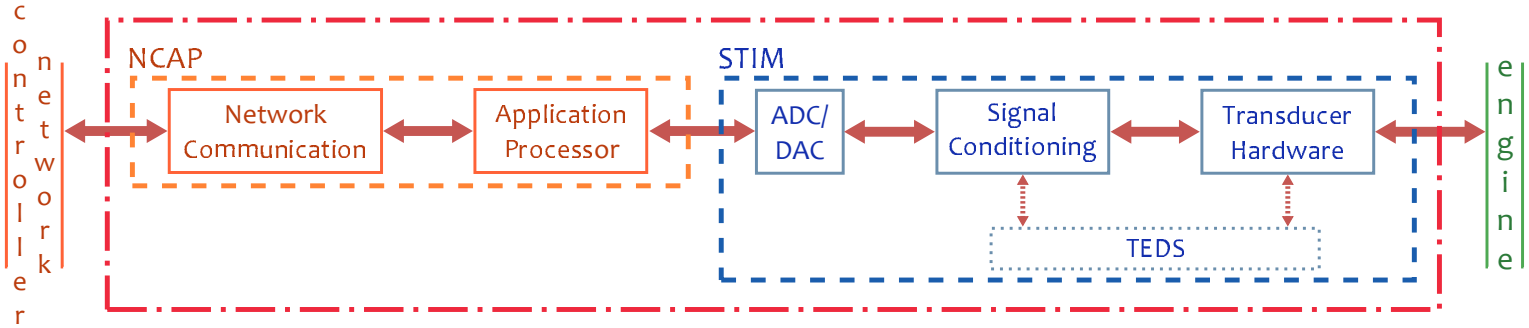
- **Additional challenges** are present in relating controller model to actual implementation of closed-loop system
  - Hardware components operate **asynchronously** and at **different sampling frequencies**
  - Simulation with hardware in the loop requires that the model supports **real-time simulation**
  - Reconfiguration, through parallelization of processing, may be considered to reflect how data and processing occurs in HIL system

Modeling elements in the control platform

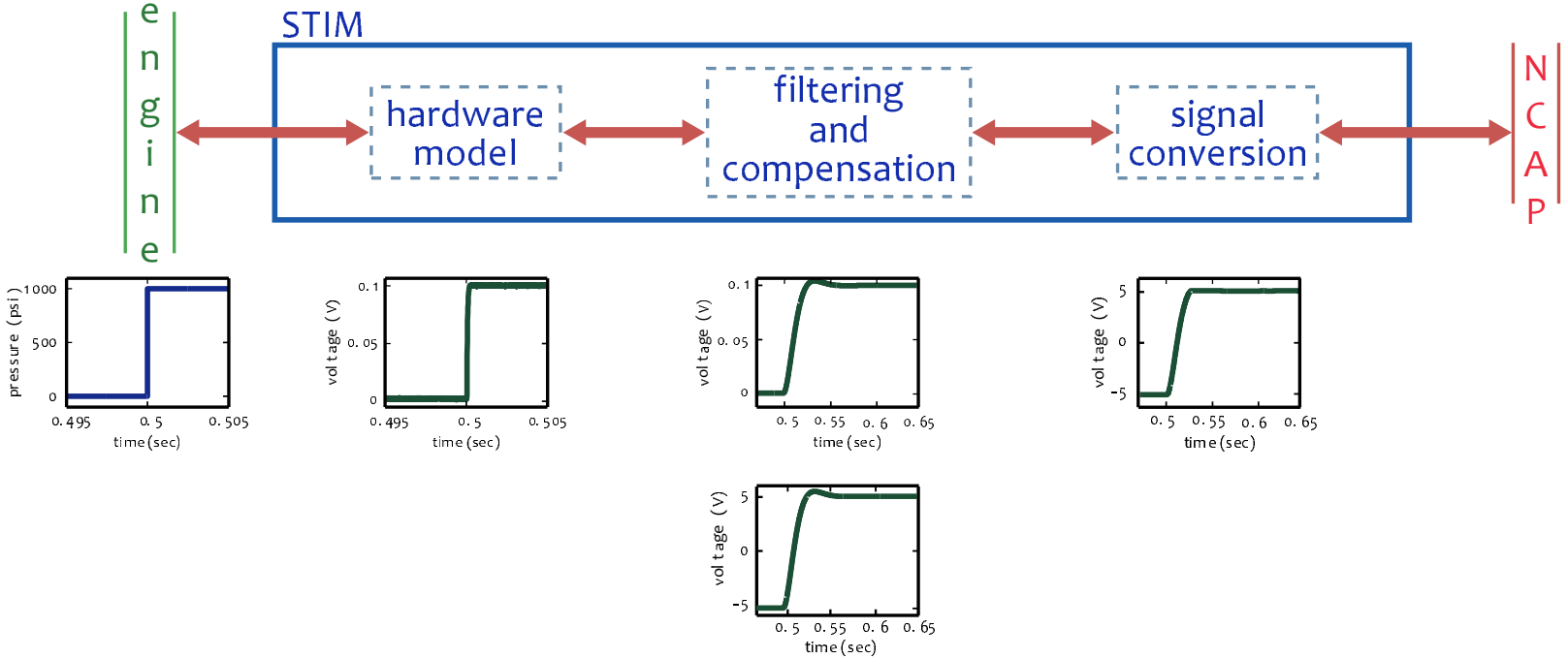
- The **control platform** contains the following components
  - Sensor nodes (take measurements from the engine/engine model)
  - Actuator nodes (affect engine operation by taking action commanded by controller)
  - Controller (implements prescribed control algorithm)
  - Controller network (on which sensors, actuators, and controller exchange data) – not discussed here
- Currently, the **C-MAPSS40k engine model** is used, where first-order transfer functions model sensor and actuator hardware

Modeling IEEE 1451 smart modules

- Replace hardware with **smart transducers**, which contain the following components
  - Transducer hardware
  - Signal conditioning, conversion, processing
  - Network connection interface
- IEEE 1451 standards: guidelines for '**plug-and-play**' smart modules
  - **S**mart **t**ransducer **i**nterface **m**odule (STIM) contains components interfacing via **analog** signals
  - **N**etwork **c**apable **a**pplication **p**rocessor (NCAP) contains components communicating with **digital** signals
  - **T**randsucer **e**lectronic **d**ata**s**heet (TEDS) contains identification and calibration information for hardware

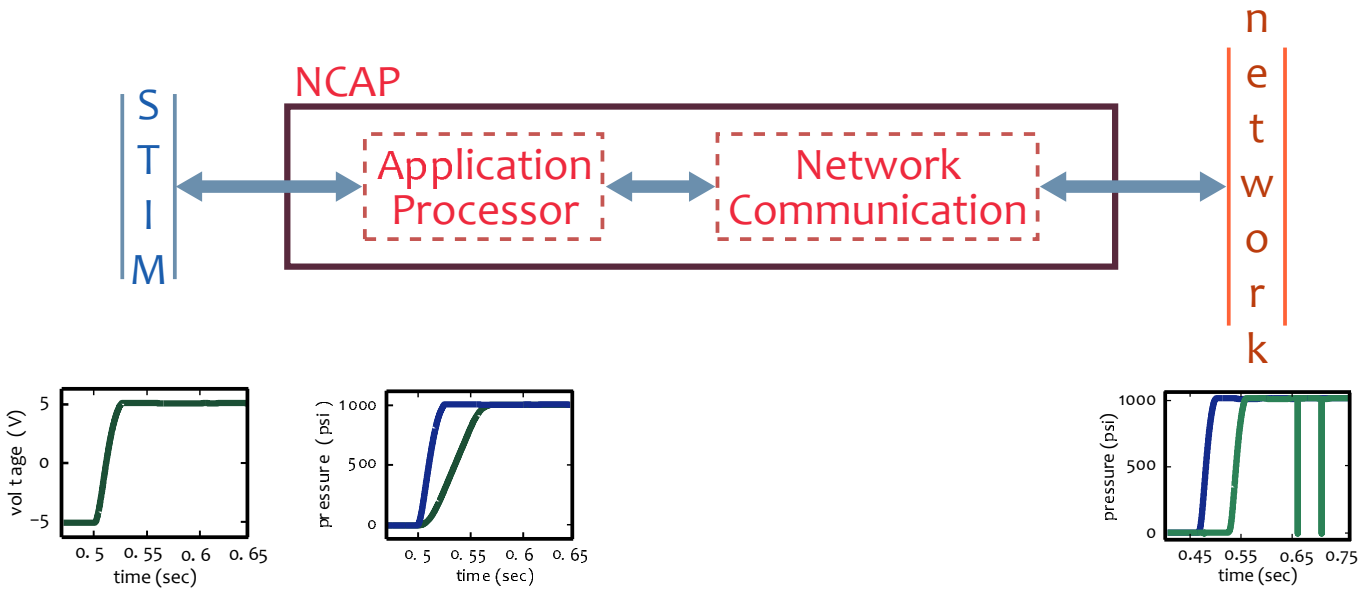


Modeling IEEE 1451 smart modules



- STIM components **process** sensor data to place on controller network, or actuator data received from controller network
- Fidelity will be improved as specifications are made available

Modeling IEEE 1451 smart modules



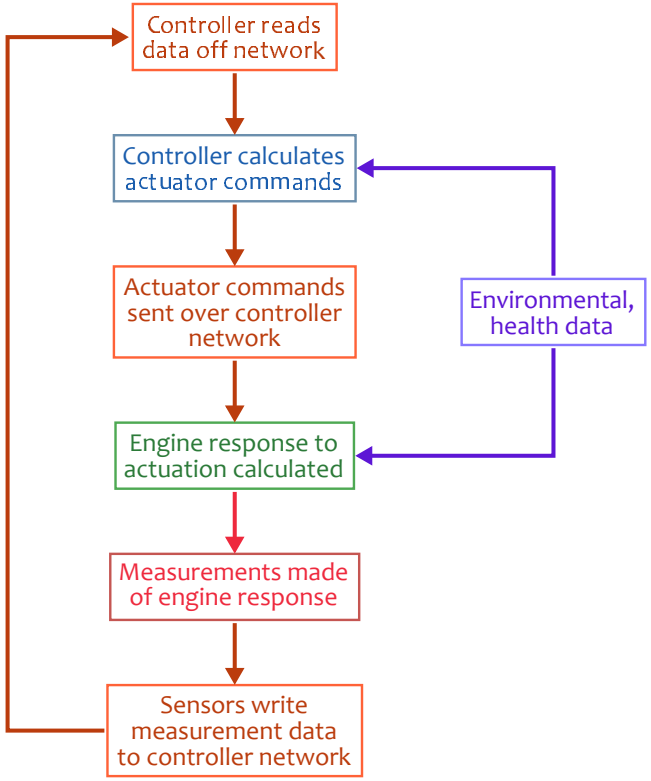
- NCAP provides **interface** between hardware (STIM) and control network
- *Application-specific models* of components are likely to be needed

Benefits of developing smart-sensor model library

- Basic functionality may be used and adapted to **model any hardware**, if specifications are available
- Allows for implementation of control algorithm (in Simulink®) **independent of hardware models**
- Establishes **interface between hardware and controller**, allowing for component model replacement without additional modification
  - Transducer model replaced by 'dragging-and-dropping' new block and 'connecting lines' (or changing model reference file)
  - Precursor for hardware-in-the-loop capability: replace computational model with connection to hardware on physical controller network
- Presence of network hardware model **improves simulation fidelity**
  - Better analysis of network effects on closed-loop stability and performance without setting up full controller network
  - Ensures simulation results of a *computational* model **predict results** from *hardware-in-the-loop* simulation

Modeling considerations for real-time operation

Execution order at each time-step:



- Hardware sampling rates not currently considered in controller model
  - Transducers operate at much **faster rate** than controller
  - Sensors may operate **asynchronously**, requiring data management scheme
- Simulations run 'as fast as possible' with no hardware in loop
  - Real-time simulation **necessary** with hardware in loop
  - **Timing constraints** on computation, communication imposed by real-time operation

Sequential versus parallel model simulation

- **Engine** and **control platforms** currently simulated on **separate** computer systems (calculations performed simultaneously)
  - 'Partial' parallelization of closed-loop model
  - Alleviates some time constraints on execution

Engine model

1. **Exchange** data with control platform model
2. **Calculate** engine state at next time-step

Control platform model

1. **Exchange** data with engine model
2. **Evaluate** actuator models
3. **Evaluate** sensor models
4. **Calculate** controller commands

- Fully-parallel *simulation* may be considered to reflect how 'actual' system operates
  - Controller, hardware elements each executed on **separate processors**
  - Requires **scheduling scheme** to ensure 'correct' data transfer
  - Necessary to procure compatible (simulation) hardware and overhaul implementation of simulation

Conclusions

- Modification of controller model for a hardware-in-the-loop system includes **adding fidelity** to models of control components
  - **Smart sensor library** (in development) may be utilized for creation of high-fidelity control (and network) component models
  - Development of **network model** will further improve fidelity of model with respect to controller network
  - Modeling control components separate from control algorithm produces **modular design** that provides bridge to connecting hardware
  - Challenges relating to **real-time operation** of the system (required when hardware connected in control loop) must be addressed

Outlook toward a hardware-in-the-loop system

- Computational simulation may be **useful precursor** to hardware-in-the-loop system if component model fidelity is very high
  - Models should be complex enough to meet fidelity requirements *without violating real-time timing constraints*
  - Dynamics of control components and network models should accurately represent dynamics of *real* hardware on a *real* network
- Presently, **smart transducer library** contains preliminary hardware models and limited additional functionality that should be expanded
  - **Hardware models** improved based on availability of specifications, or data, for actual components
  - **Signal conditioning, conversion** implemented to same accuracy expected from physical hardware
  - Specifications of **network component models** based off data collected from implemented controller network
  - **Network model** (separate from, but included in, controller model) should capture functionality, role of implementation

Questions/Discussion

References

1. Csank, J., May, R. D., Litt, J. S., and Guo, T.-H., 'Control Design for a Generic Commercial Aircraft Engine,' *Proceedings of the 46th AIAA/ASME/SAE/ASEE Joint Propulsion Conference*, No. AIAA-2010-6629, Nashville, TN, July 2010.
2. Culley, Dennis. 'Transition in gas turbine control system architecture: Modular, distributed, and embedded'. In *ASME Turbo Expo2010: Power for Land, Sea, and Air*, volume 3, pp. 287Ű297, Glasgow, Scotland, United Kingdom, June 2010.
3. Culley, Dennis E., Thomas, Randy, and Saus, Joseph. 'Concepts for distributed engine control'. In *Proceedings of the 43rd Joint Propulsion Conference and Exhibit*, No. AIAA-2007-5709, Cincinnati, OH, July 2007.
4. Culley, D., Thomas, R., and Saus, J., 'Integrated tools for future distributed engine control technologies,' *Proceedings of the ASME Turbo Expo 2013*, No. GT2013-95118, San Antonio, TX, USA, June 2013.
5. 'IEEE Standard for a Smart Transducer Interface for Sensors and Actuators - Common Functions, Communication Protocols, and Transducer Electronic Data Sheet (TEDS) Formats', *IEEE Std 1451.0*, September 2007.
6. 'IEEE Standard for a Smart Transducer Interface for Sensors and Actuators - Network Capable Application Processor (NCAP) Information Model', *IEEE Std 1451.1*, 2000.
7. 'IEEE Standard for a Smart Transducer Interface for Sensors and Actuators - Transducer to Microprocessor Communication Protocols and Transducer Electronic Data Sheet (TEDS) Formats', *IEEE Std 1451.2*, 1998.
8. 'IEEE Standard for a Smart Transducer Interface for Sensors and Actuators - Digital Communication and Transducer Electronic Data Sheet (TEDS) Formats for Distributed Multidrop Systems', *IEEE Std 1451.3*, April 2004.
9. Lee, Edward A.. 'Modeling concurrent real-time processes using discrete events,' *Annals of Software Engineering*, Vol. 7, No. 1-4, 1998, pp. 25-45.
10. Lee, Kang. 'IEEE 1451: A standard in support of smart transducer networking.' In *Proceedings of the 17th IEEE Instrumentation and Measurement Technology Conference*, volume 2, pp. 525-528, Baltimore, MD, May 2000.

References

11. Lee, Edward Ashford and Messerschmitt, David G. 'Static Scheduling of Synchronous Data Flow Programs for Digital Signal Processing,' *IEEE Transactions on Computers*, Vol. C-36, No. 1, January 1987, pp. 24-35.

12. Lian, Feng-Li, Moyne, James, and Tilbury, Dawn. 'Network design consideration for distributed control systems,' *IEEE Transactions on Control Systems Technology*, Vol. 10, No. 2, March 2002, pp. 297-307.

13. May, R. D., Csank, J., Lavelle, T. M., Litt, J. S., and Guo, T.-H., 'A High-Fidelity Simulation of a Generic Commercial Aircraft Engine and Controller,' *Proceedings of the 46th AIAA/ASME/SAE/ASEE Joint Propulsion Conference*, No. AIAA-2010-6630, Nashville, TN, July 2010.

14. Misra, Jayadev. 'Distributed discrete-event simulation,' *ACM Computing Surveys*, Vol. 18, No. 1, March 1986, pp. 39-65.

15. Song, Eugene Y. and Lee, Kang. 'Understanding IEEE 1451 - Networked smart transducer interface standard,' *IEEE Instrumentation & Measurement Magazine*, April 2008, pp. 11-17.

16. Torngren, Martin. 'Fundamentals of implementing real-time control applications in distributed computer systems,' *Real-Time Systems*, Vol. 14, No. 3, May 1998, pp. 219-250.

Appendix

Developing a hardware-in-the-loop (HIL) system

- Transition from **centralized** to **distributed** control is slowed due to the limited role control design has in engine design
- A **hardware-in-the-loop system** may provide additional motivation for this shift in control implementation
  - Control design and hardware testing can be performed **simultaneous to** engine design, without need for physical engine
  - Computational models of engine and controller **interface with** hardware elements (computational or prototypes)
  - Can create test conditions **unattainable in test cells**
  - **Minimize** risks, costs associated with testing physical system

Developing a hardware-in-the-loop (HIL) system

Smart sensor model

- (Analog) measurement altered by physical hardware dynamics, signal conditioning
- Signal converted to 'digital' for use by controller

Smart actuator model

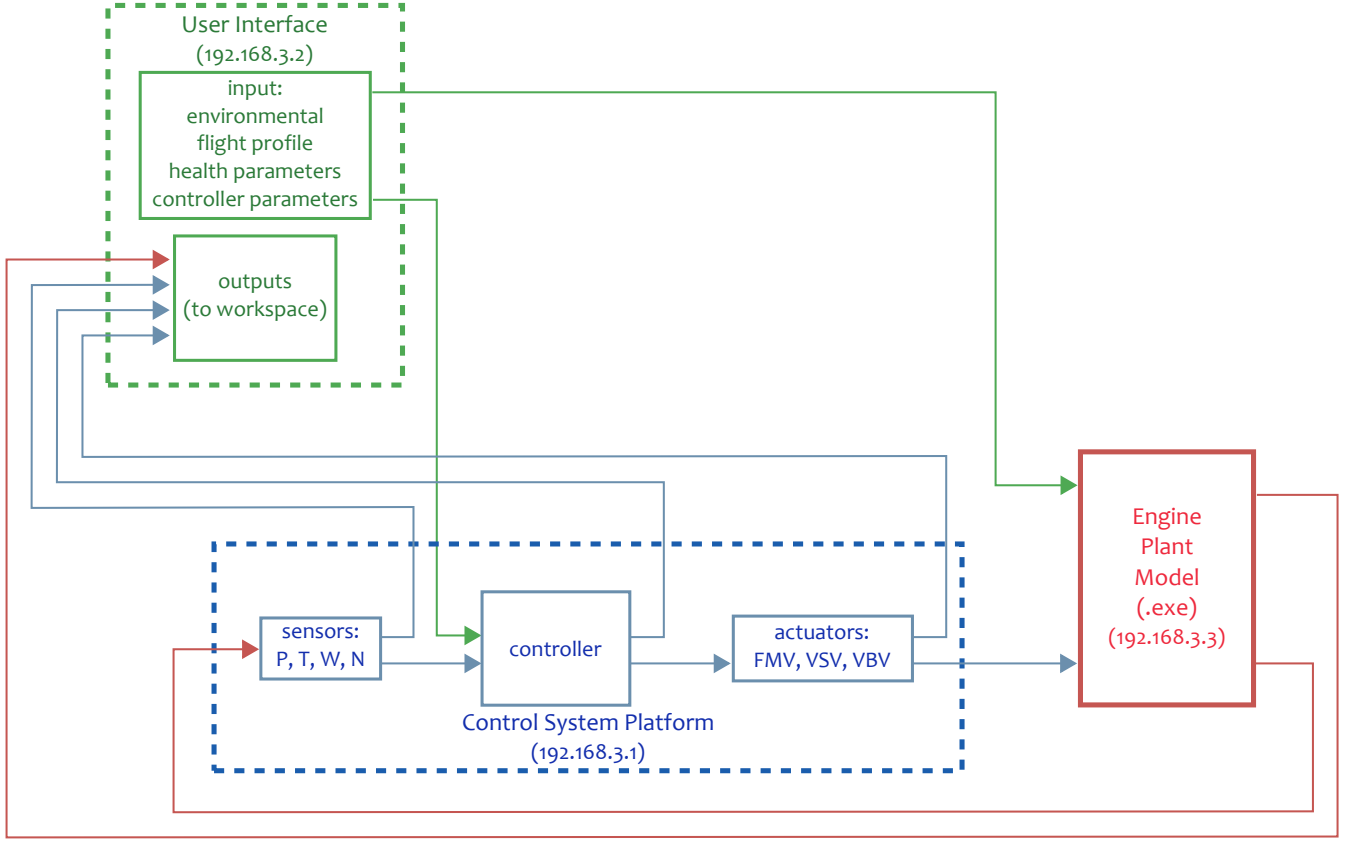
- (Digital) control command converted to 'analog' for use by hardware
- Signal conditioning, hardware dynamics further affect (analog) command

Network hardware model

- Account for packet delay and packet loss due to presence of network hardware
- Higher fidelity introduced by network system model (separate from controller model)

- Simulink® library **under development** for constructing **high-fidelity hardware models** (when specifications are known) from blocks with basic component functionality

Hardware-in-the-loop (HIL) system set-up

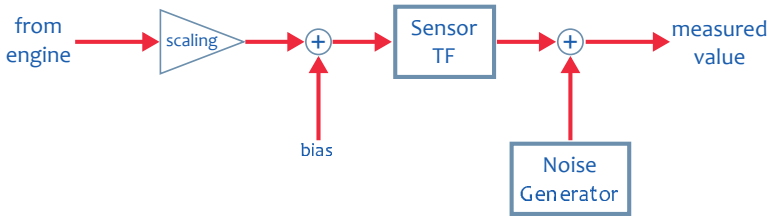


- **Current set-up**: three computers communicating over UDP
  - User interface and control system platform: Simulink® models
  - Engine plane model: executable (compiled using Simulink Coder®)
- Nearly all simulation setup and control from computer hosting user interface

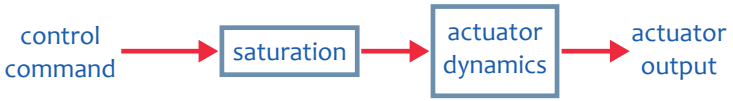
Modeling hardware in C-MAPSS40k

(Baseline) controller in the C-MAPSS40k engine model contains models of ten sensors and three actuators

- Four pressure sensors, four temperature sensors, two speed sensors
  - Dynamics modeled using a **first-order transfer function**
  - Measurement noise and bias/scaling can be modeled



- Fuel metering valve and variable geometry actuators are modeled
  - Fuel metering valve is modeled using a **first-order transfer function** and a delay
  - Variable geometry dynamics are modeled using **first-order transfer functions**



Modeling IEEE 1451 smart modules: Simulink® library

- Results on previous slides demonstrate **current functionality** of library
  - First-order lag dynamics used to model hardware
  - Signal conditioning functions limited to scaling, offset, and filtering
  - Analog-to-digital conversion quantizes input to a given resolution on a given (voltage or current) range
  - Digital-to-analog conversion uses filter to 'smooth' input
  - Processing functions include mapping between electrical and physical signals and averaging signals
- Hardware models may be improved with **availability of specifications** (or data) from actual components
- Additional conditioning and processing functions may be added as necessary for given hardware or application

Network component modeling

- Network hardware (and implementation) **not presently modeled**
  - Network hardware models represent **interface** between sensor, actuator models and control network
  - Packet delays and data loss may cause closed-loop system to **lose performance**, or **become unstable**
  - Modeling of network implementation done separately, not part of smart transducer library
- Network cable model introduces **packet delay** and **packet drop**
  - Probability distributions determine length of (random) packet delay and whether data is lost during transmission
  - Sensor and actuator hardware models may be connected to controller through cable model, adding fidelity to the model
  - Additional network effects will be included in network model

Challenges of modeling network effects

- Components of closed-loop engine system **not inherently synchronized**: operate without knowledge of other components' operation
- **Controller network** can be used to **synchronize** operation by providing timing/clock information in each data packet
  - Most-recent measurements used to calculate control command
  - Most-recently calculated command used by actuators
  - Implement method to **handle dropped/delayed data** packets while **maintaining stable closed-loop operation**
- **Accurate modeling** of network requires capturing packet drop and data corruption probabilities to improve reliability
  - Characteristics of probability distributions determined from measurements taken for specific network used to implement the HIL system