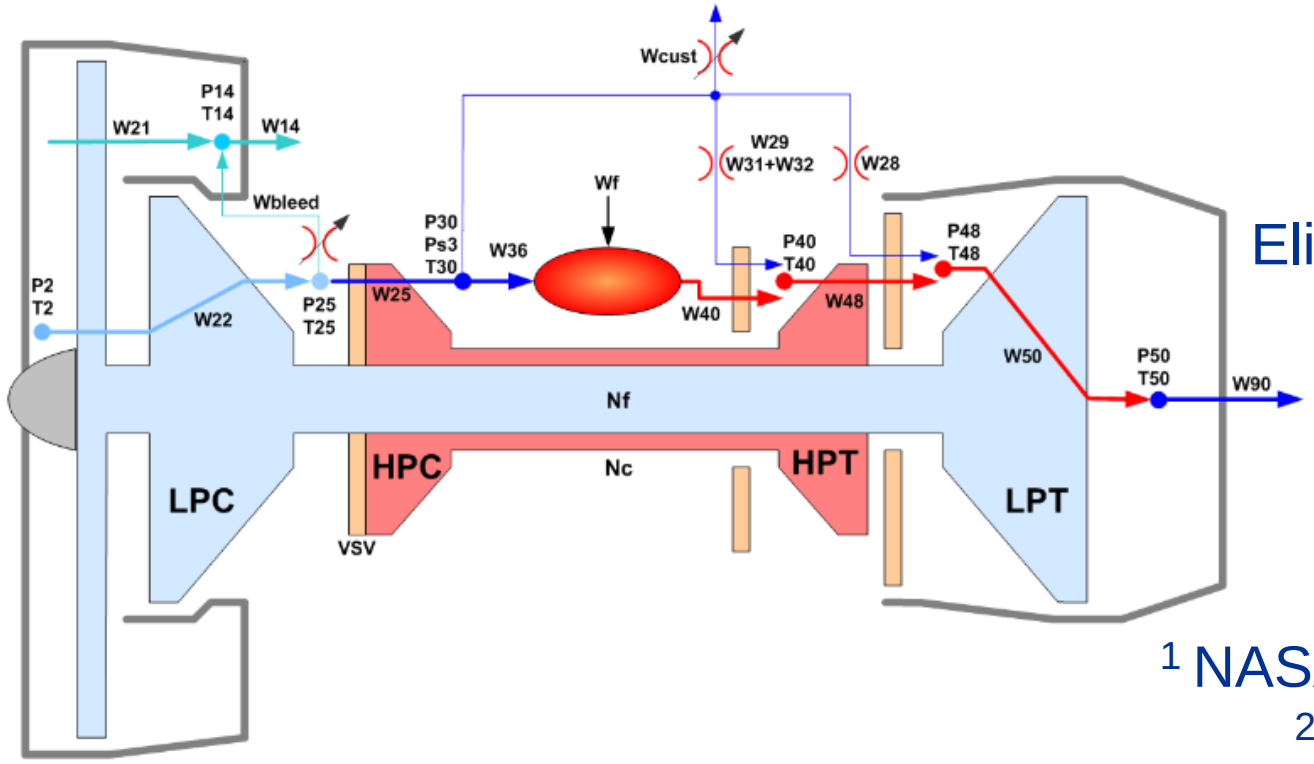
Benchmarking model variants in development of a hardware-in-the-loop simulation system



Eliot Aretskin-Hariton[1], Alicia Zinnecker[2], Jonathan Kratz[1], Dennis Culley[1], George Thomas[2]

SciTech 2016

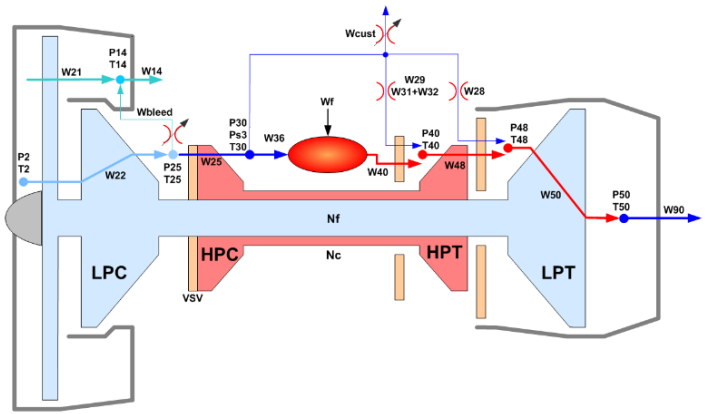
[1] NASA Glenn Research Center, Cleveland, OH
[2] N & R Engineering, Parma Hts., OH

Overview

- Motivation and goals
- Control system model enhancements for hardware-in-the-loop simulations
- Benchmarking results
- Conclusions

Motivation and Goals

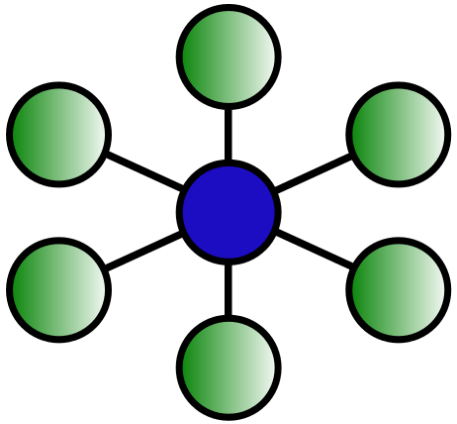
- Develop C-MAPSS40k engine model to better represent a HIL system
- Distributed Engine Control Working Group (DECWG) formed to build standards and explore the capabilities of distributed engine control
- One of NASAs contributions: conversion of C-MAPSS40k to a distributed model to enable exploration of distributed engine control
- Allows for the exploration of advanced engine control systems
  - Controls to compensate for information loss
  - Model based controls
  - Requirements exploration



Centralized vs. Distributed Control

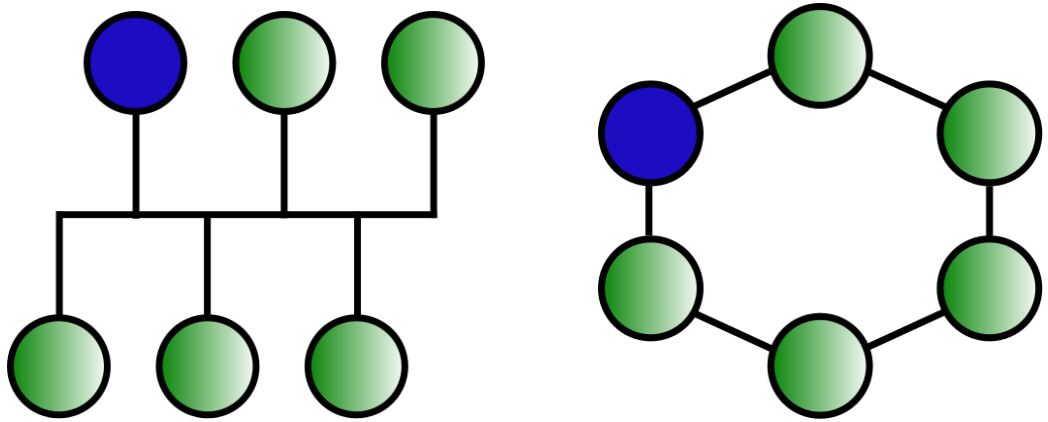
Centralized & Analog

- Simultaneous data availability
- Dedicated cable for each node
- A/D handled by central node
- Analog sensor/actuator interfaces prevent easy replacement & alternative sourcing of components



Distributed & Digital

- Sequential data availability
- A/D imbedded in the smart node
- Drop-in component compatibility
- Packet delay
- Packet loss
- Packet corruption

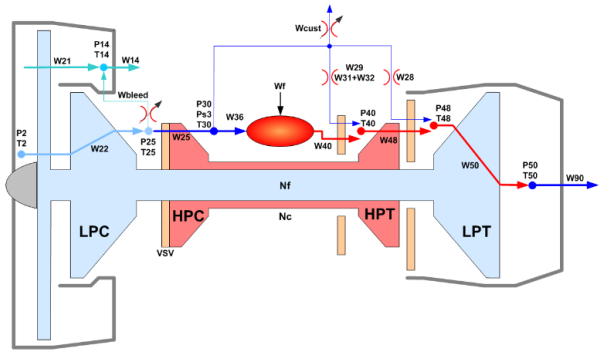


Controller Models

- Baseline: C-MAPSS40k out of the box, 1 computer

Baseline C-MAPSS40k

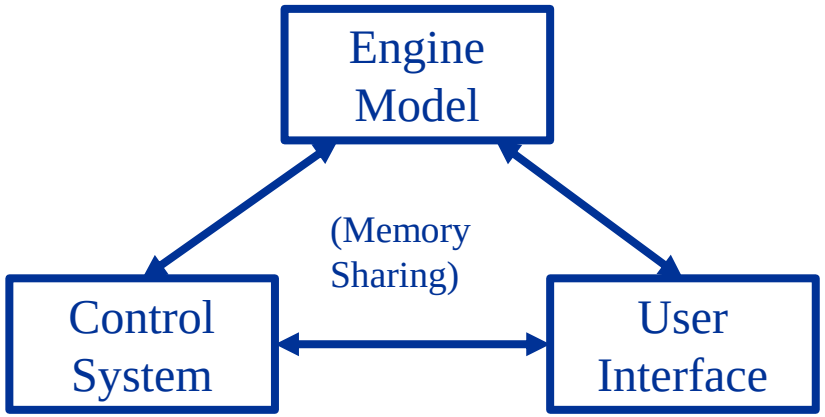
Engine Model
Control System
User Interface



Several new configurations were compared to the original C-MAPSS40k:

- Unstructured
- Distributed
- Networked
- Processor-in-the-Loop

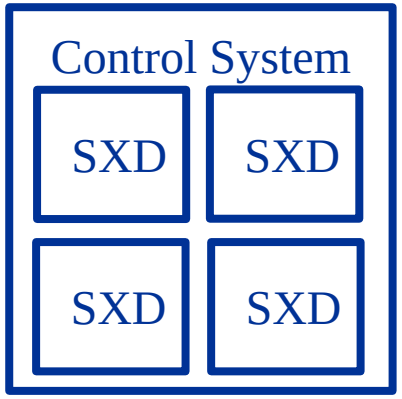
Controller Models - Unstructured

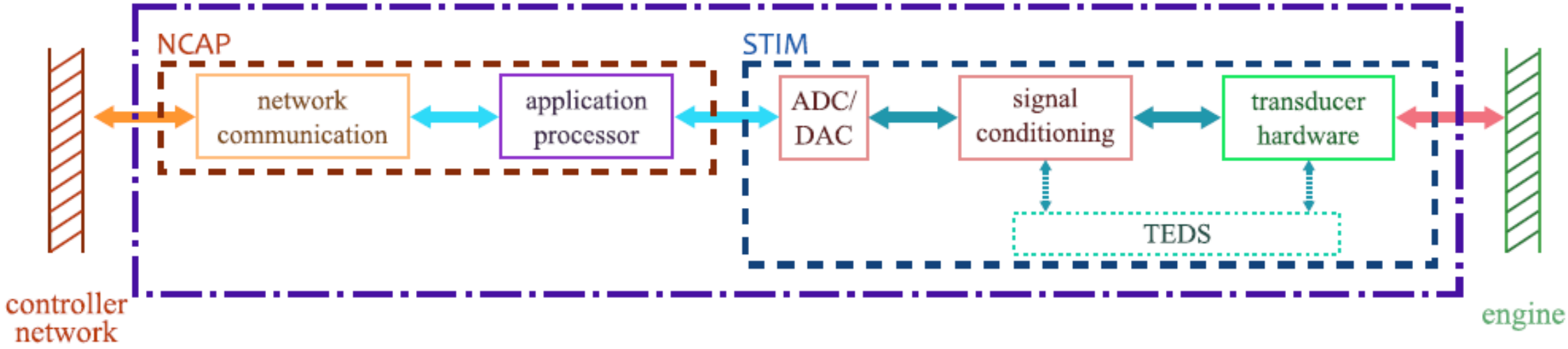


- Baseline C-MAPSS40k with independent models for control and engine plant
- UDP Ethernet network used to transparently share information between models
- Modular engine model
- Causes minimum of one time step lag between models

Smart Transducer Model (SXD)

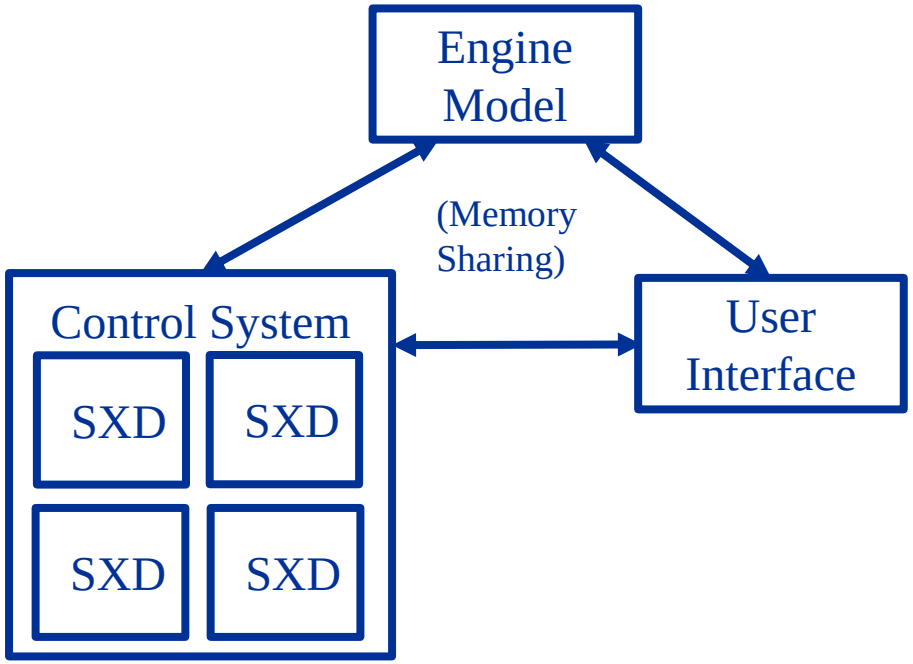
- Network Capable Application Processor (NCAP)
  - Network communication
  - Application interface to STIM
- Smart Transducer Interface Module (STIM)
  - Analog to Digital and Digital to Analog converter
  - Signal conditioning
  - Interface to transducer





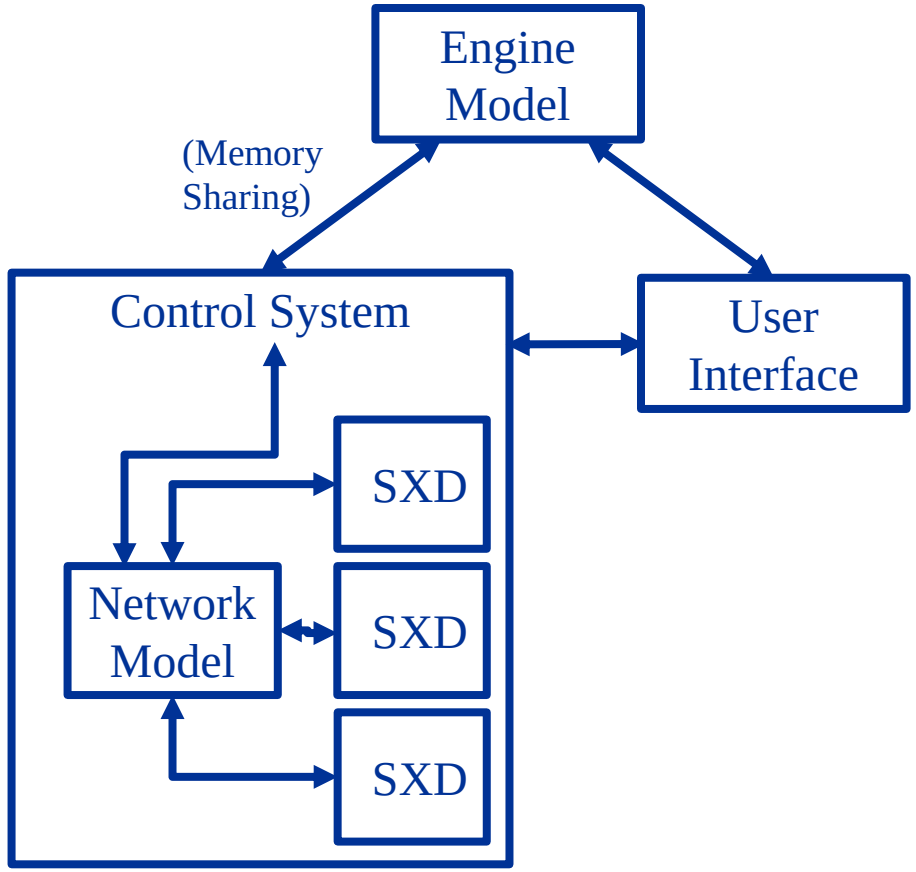
Controller Models - Distributed

- Enhanced fidelity of control elements by inclusion of smart transducer models (SXD)
- SXD models include quantization effects of 14 bit A/D conversion
- Increased computational complexity and increased errors due to quantization



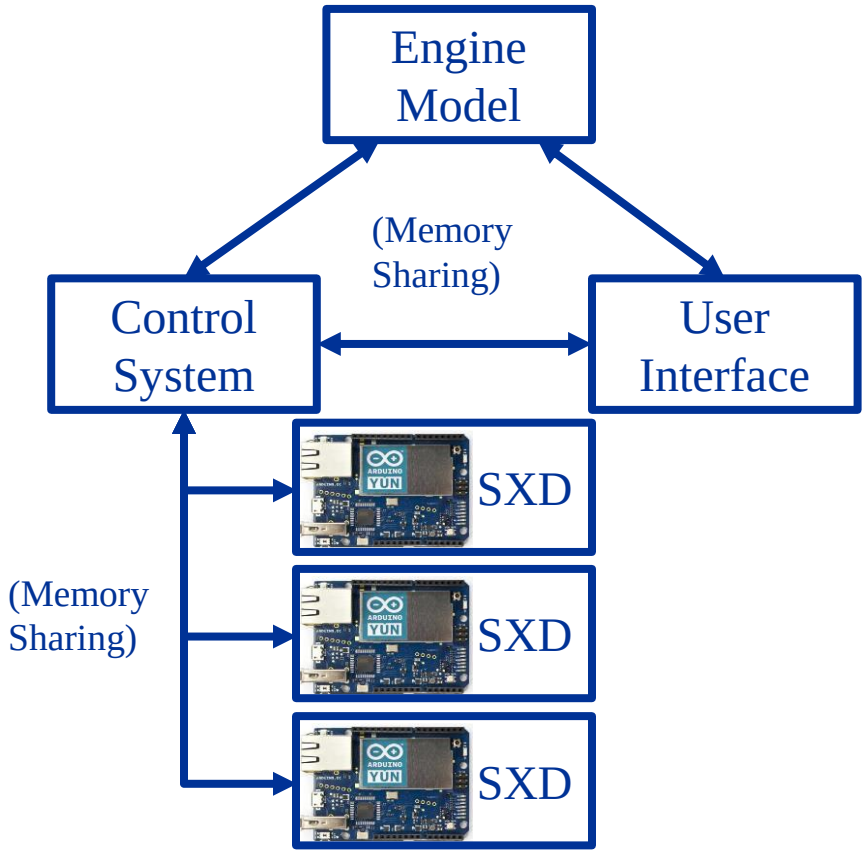
Controller Models - Networked

- Bulk network model limits communication between control system and SXD models
- Model includes packet loss and packet delay effects as settable percentages
- Increases simulation time
- Not as detailed as a packet level network model

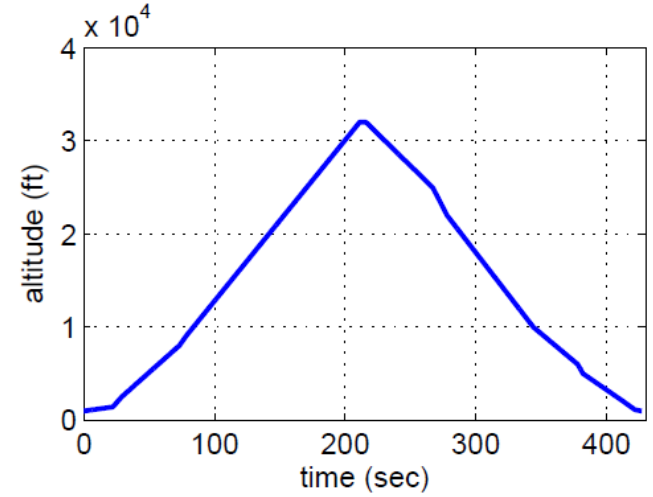


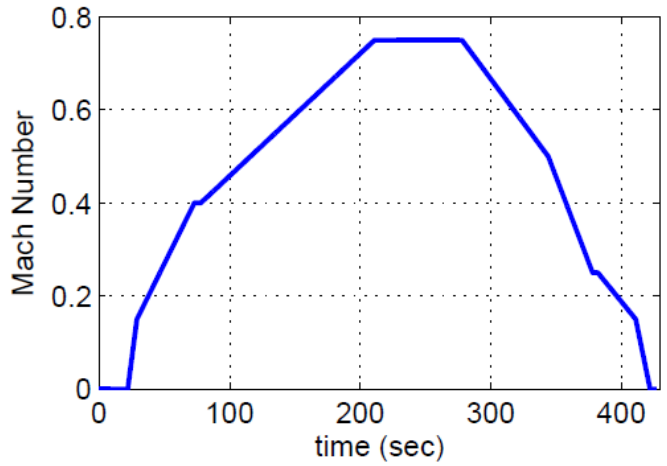
Controller Models – Processor-in-the-Loop

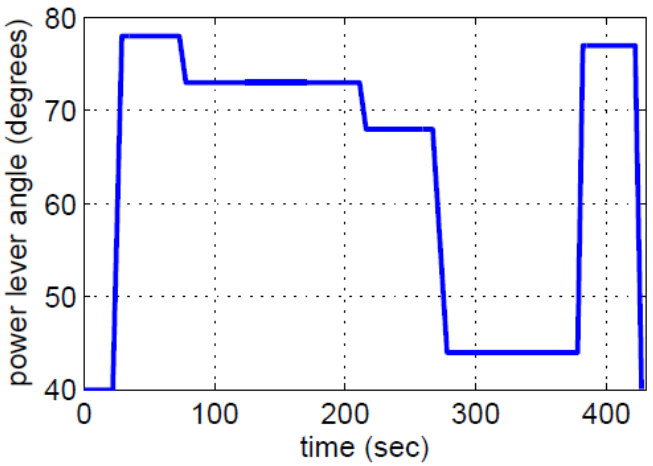
- Integration of SXD models into microcontrollers
- Ethernet UDP network used to communicate between control system and SXD models
- Better simulates processor limitations in SXD models
- Increase command / response latency due to message buffering issues in each system

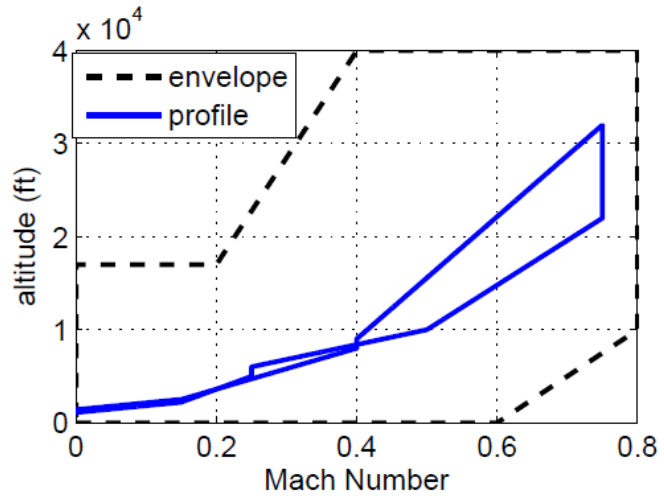


Test Profile

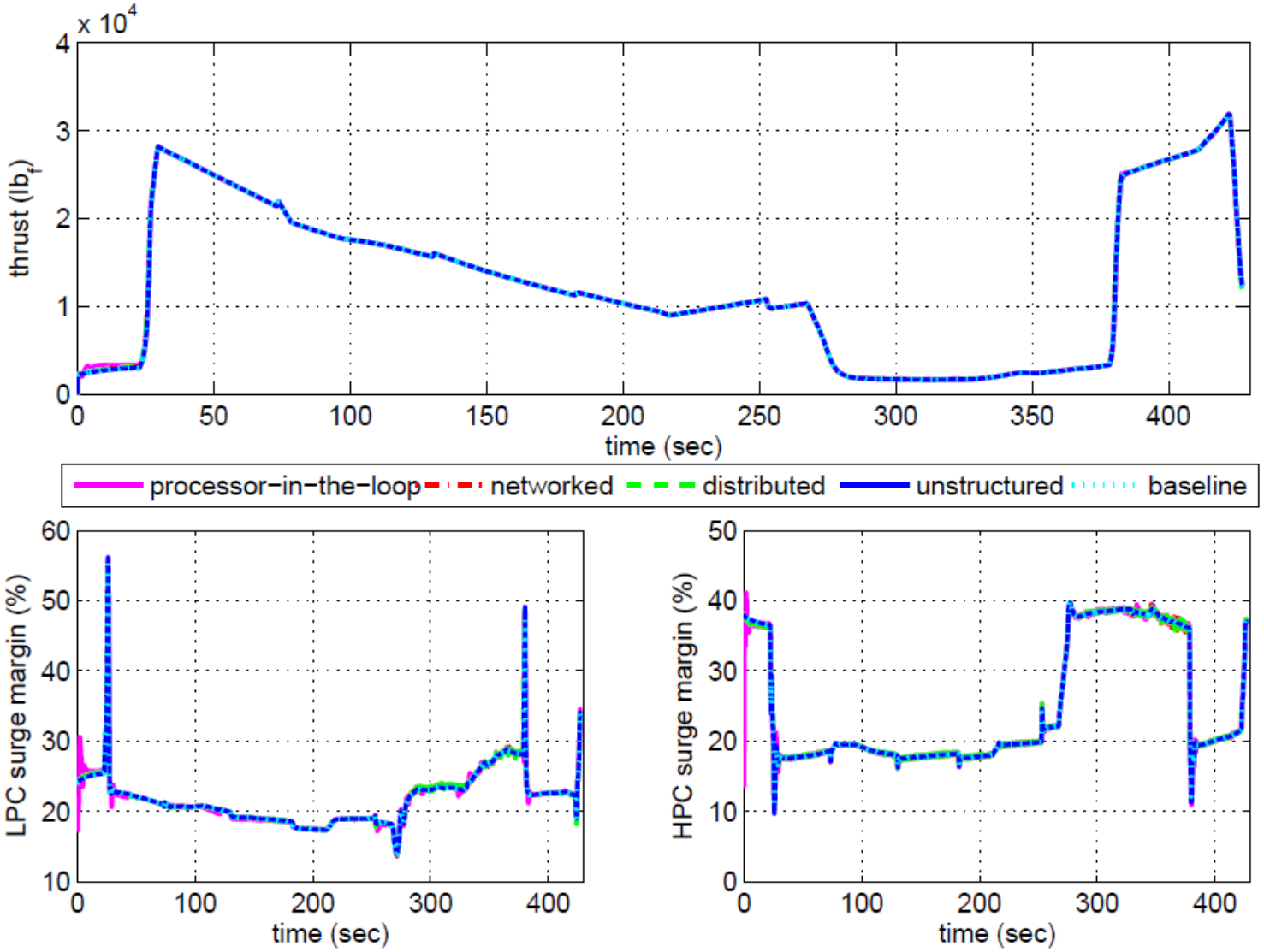




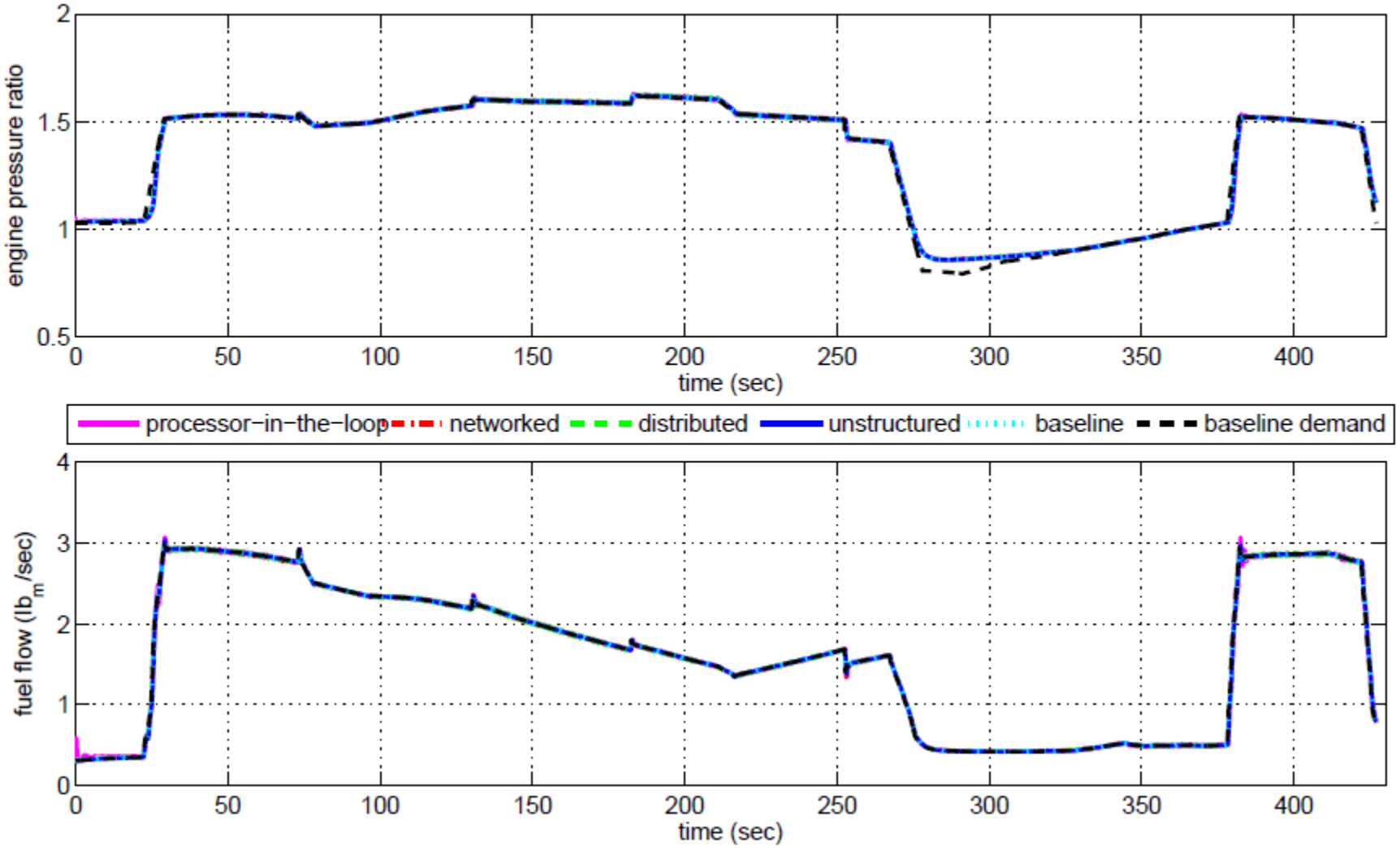




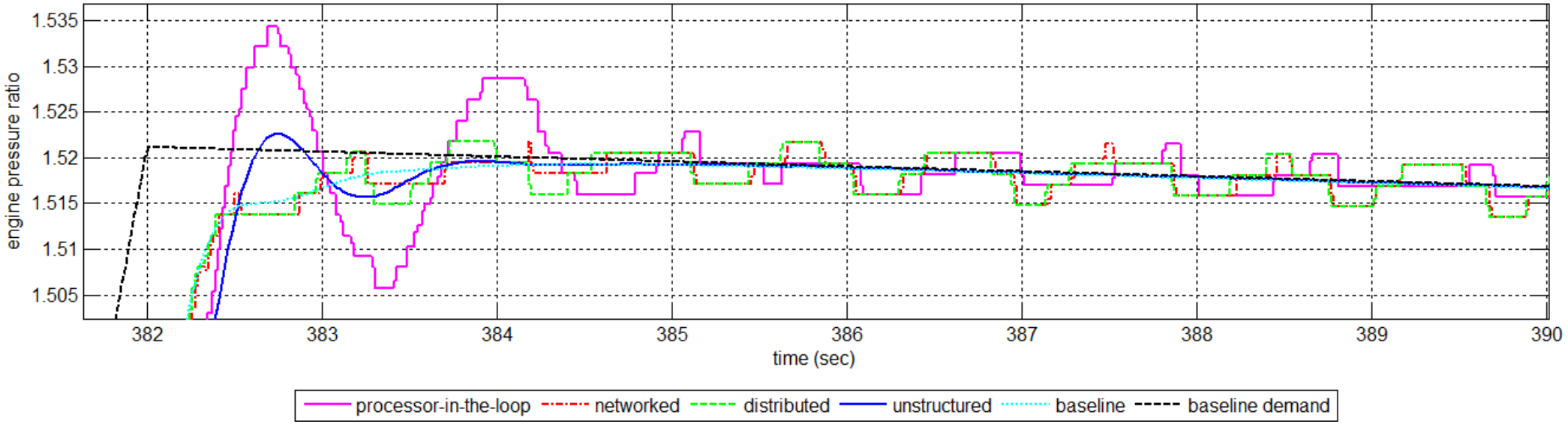
Benchmarking Results - Thrust



Results – EPR demand and Fuel Flow

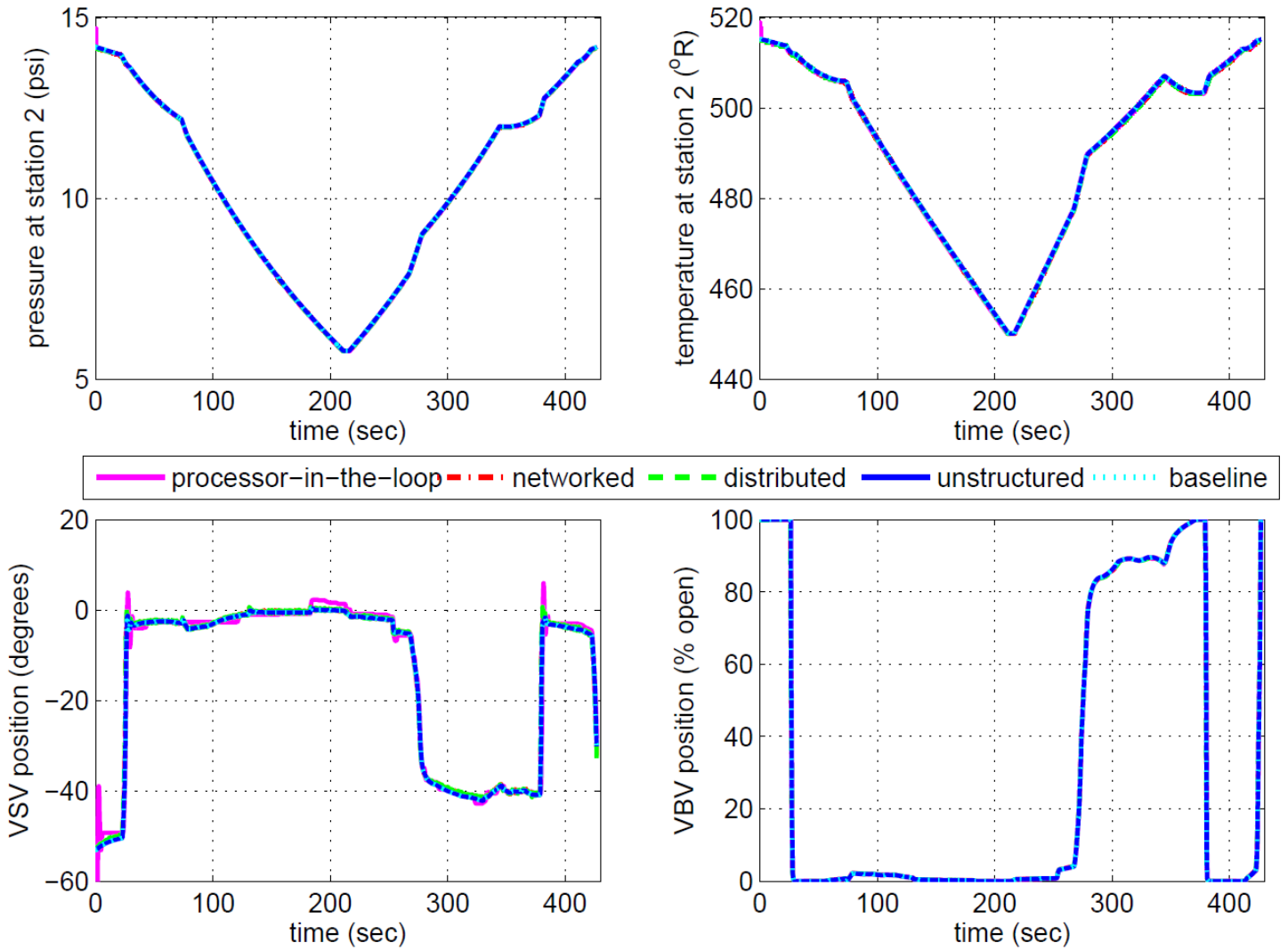


Results – EPR demand



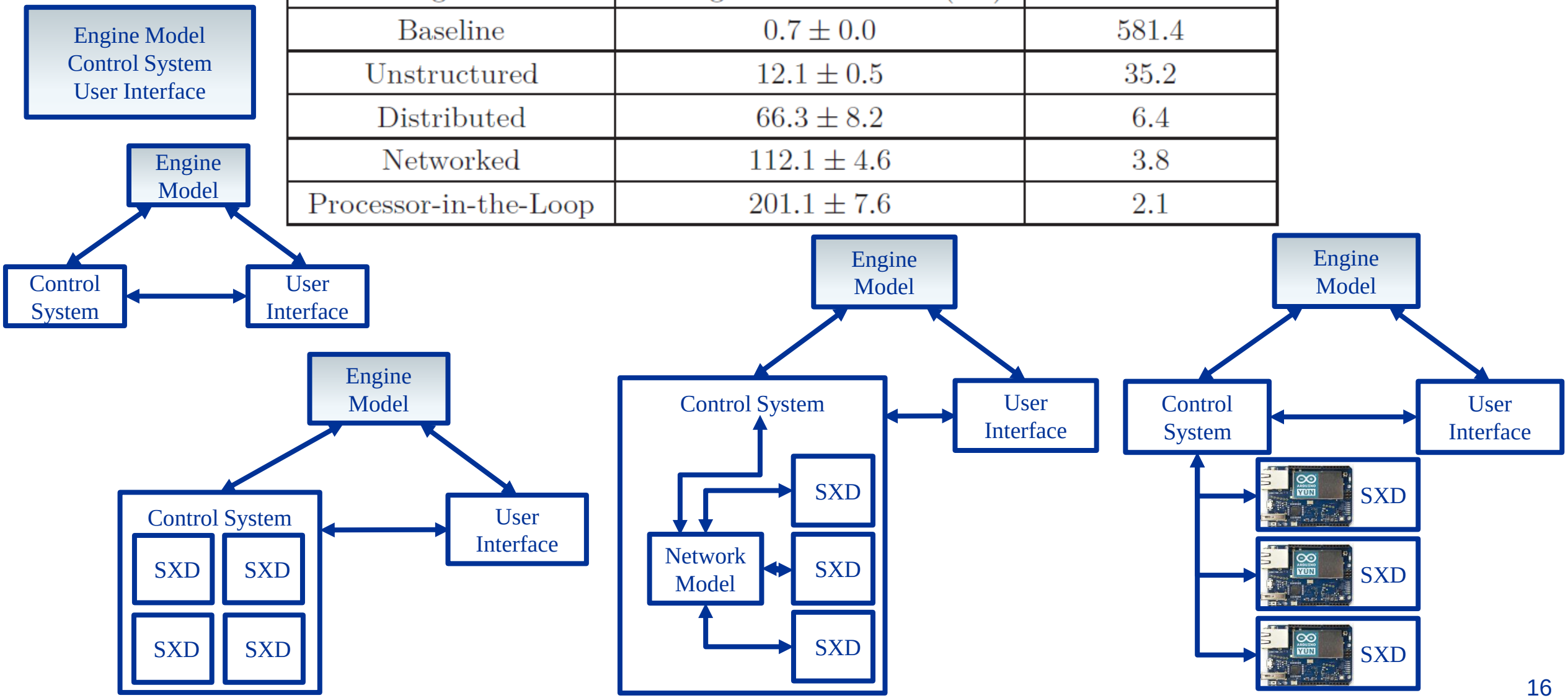
- Simulations with quantization clearly visible
- Quantization effects cascade from the sensors into the control system and then into the actuator commands
- Simulations with 8 bit quantization was unable to converge

Benchmarking Results



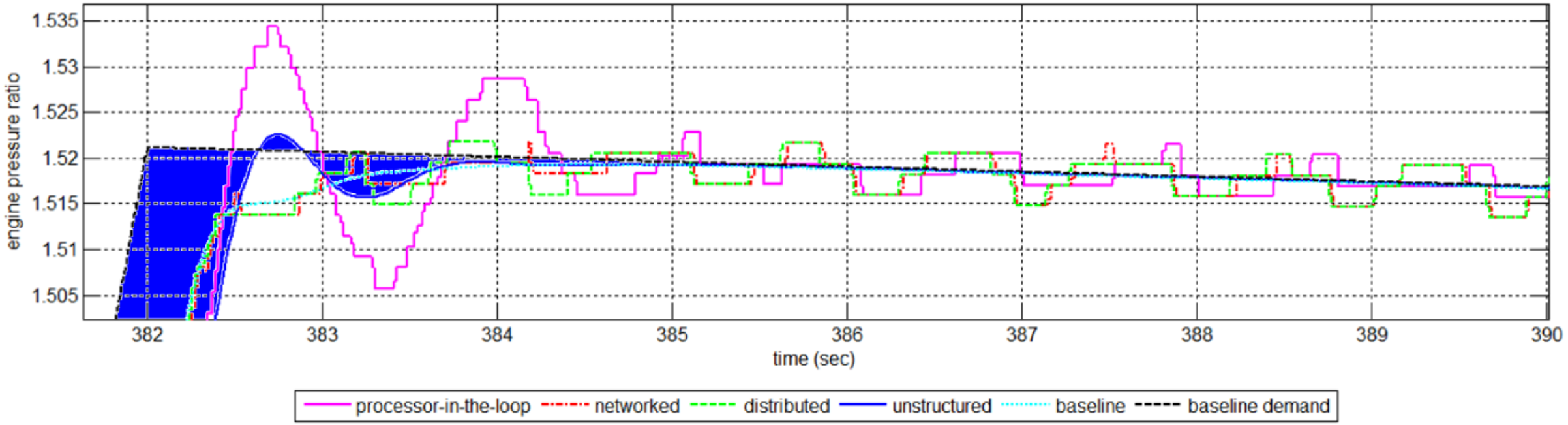
Average Benchmarking Run Times

Configuration	Average total run time (sec)	Real-time factor
Baseline	0.7 ± 0.0	581.4
Unstructured	12.1 ± 0.5	35.2
Distributed	66.3 ± 8.2	6.4
Networked	112.1 ± 4.6	3.8
Processor-in-the-Loop	201.1 ± 7.6	2.1



EPR Tracking Error

Configuration	Percentage Mean Absolute EPR Tracking Error
Baseline	1.047 ± 0.00
Unstructured	1.091 ± 0.001
Distributed	1.197 ± 0.009
Networked	1.198 ± 0.008
Processor-in-the-Loop	1.227 ± 0.004



Summary & Conclusions

- Five Configurations of C-MAPSS40k Engine Model Tested
- Each configuration added additional complexity to the simulation
- Small (>1%) differences between configurations
- This shows that the implementation is solid
- These new configurations will be used to study real-time and network model integration to help us answer important questions like:
  - How much bandwidth do I need to perform minimum control operations
  - How much processing capability do I need on my smart nodes to ensure safe operation (this may increase with decreasing bandwidth because you need more local limiters)

Questions?

Eliot Aretskin-Hariton

earetski@mail.nasa.gov

This work is being performed by the Intelligent Control and Autonomy branch at NASA Glenn Research Center and is supported by the Transformative Tools and Technologies (TTT) project which is part of the Aerospace Research Mission Directorate (ARMD)

Future Work – Network-in-the-Loop