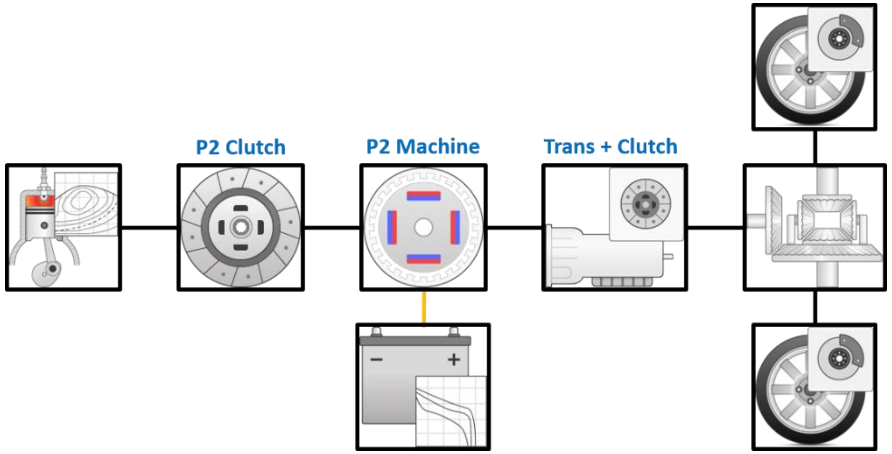
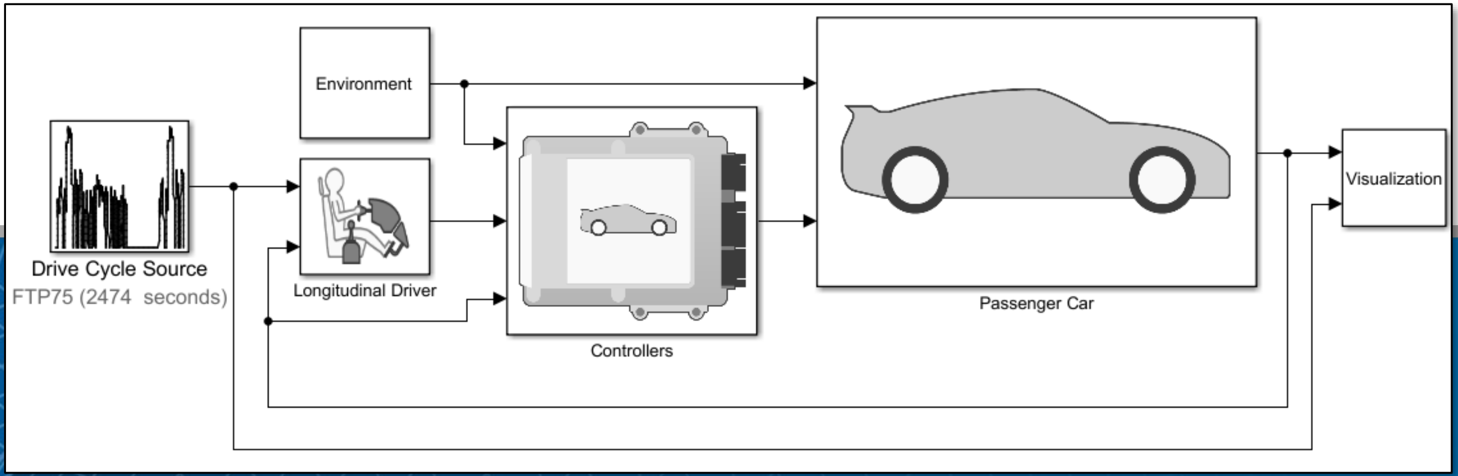
Full Vehicle Simulation for Electrified Powertrain Selection



MathWorks Automotive Conference

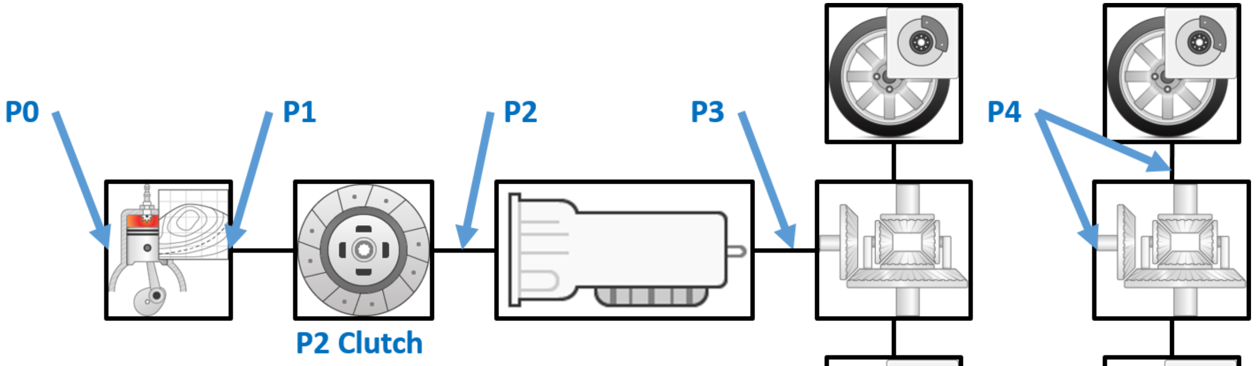
April 11, 2019

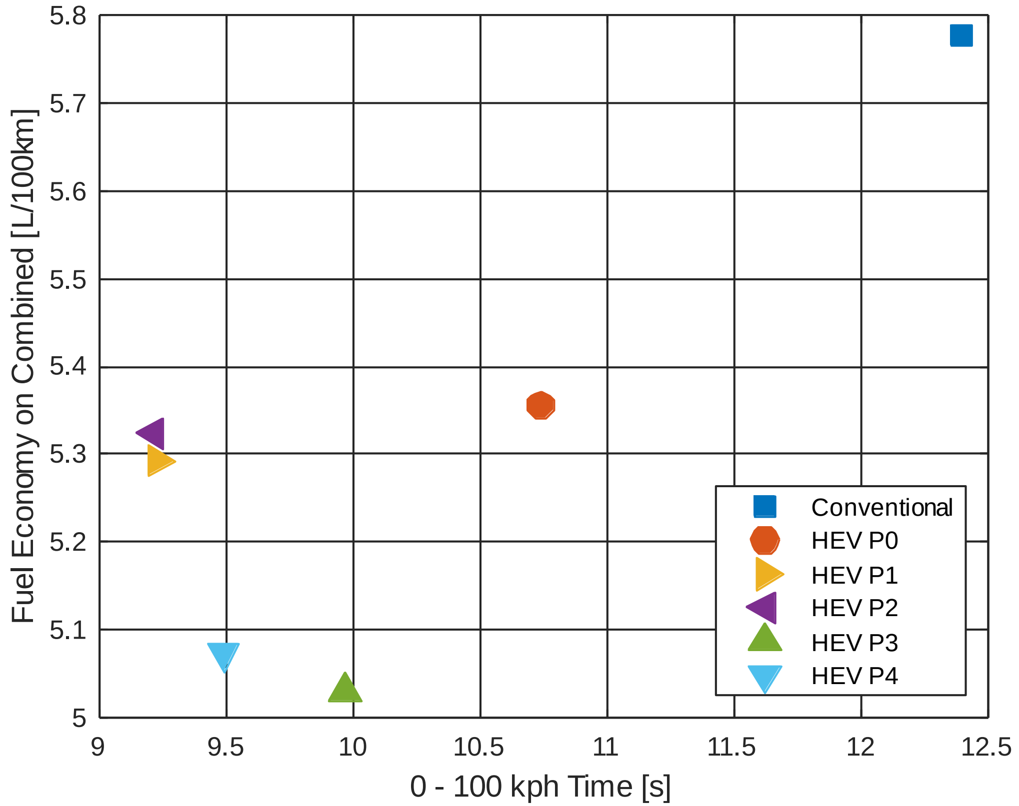


Eva Pelster, Application Engineer

Key Points

- Customize pre-built vehicle models to assess electrified powertrain variants
- Apply optimal control techniques to make fair comparisons
- Quantify tradeoffs between fuel economy and performance





Agenda

- Context
- Case study description
- Tools used
- Plant model and controls
- Results
- Next steps

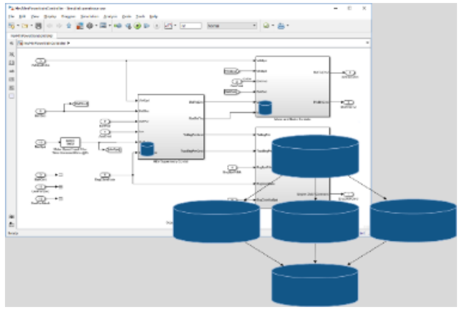
What Is Meant By “Full Vehicle Simulation”?

- Plant model + closed-loop control algorithms
  – Production code out of scope for today’s presentation (OBD, timing, etc.)

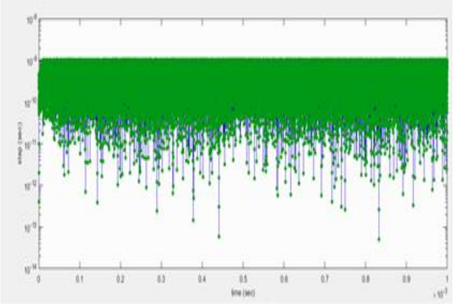
- Right balance of accuracy / speed
  – Sufficient detail for attribute analysis (fuel economy, performance, drivability, …)
  – Fast enough for design optimization (much faster than real-time)

- Heterogeneous modeling environment
  – Support for inclusion of 3rd party simulation tools (S-function, FMU, …)

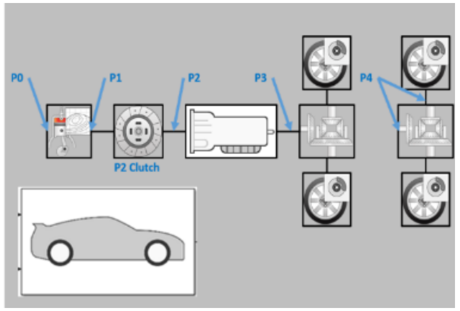
Simulink as a Simulation Integration Platform



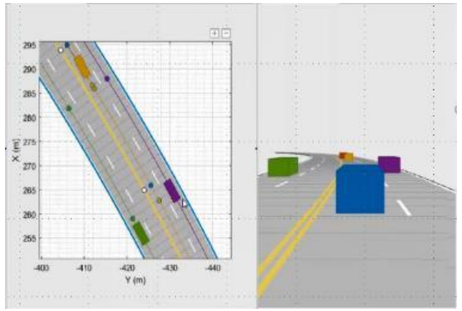
Data Management



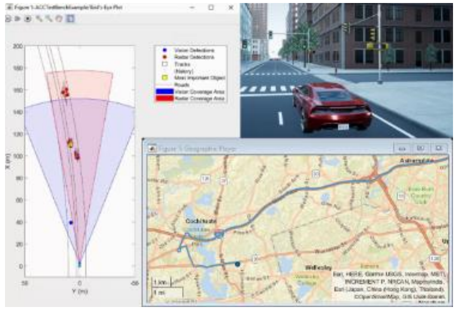
Solver Technology



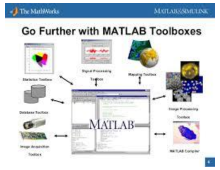
Vehicle Configuration

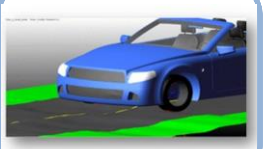


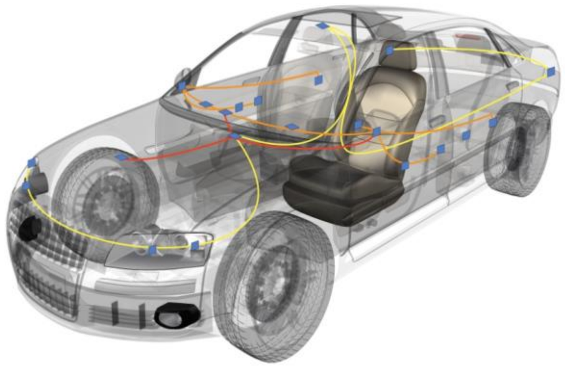
Multi-actor Scenarios



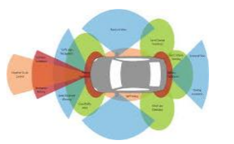
Visualization







Simulink

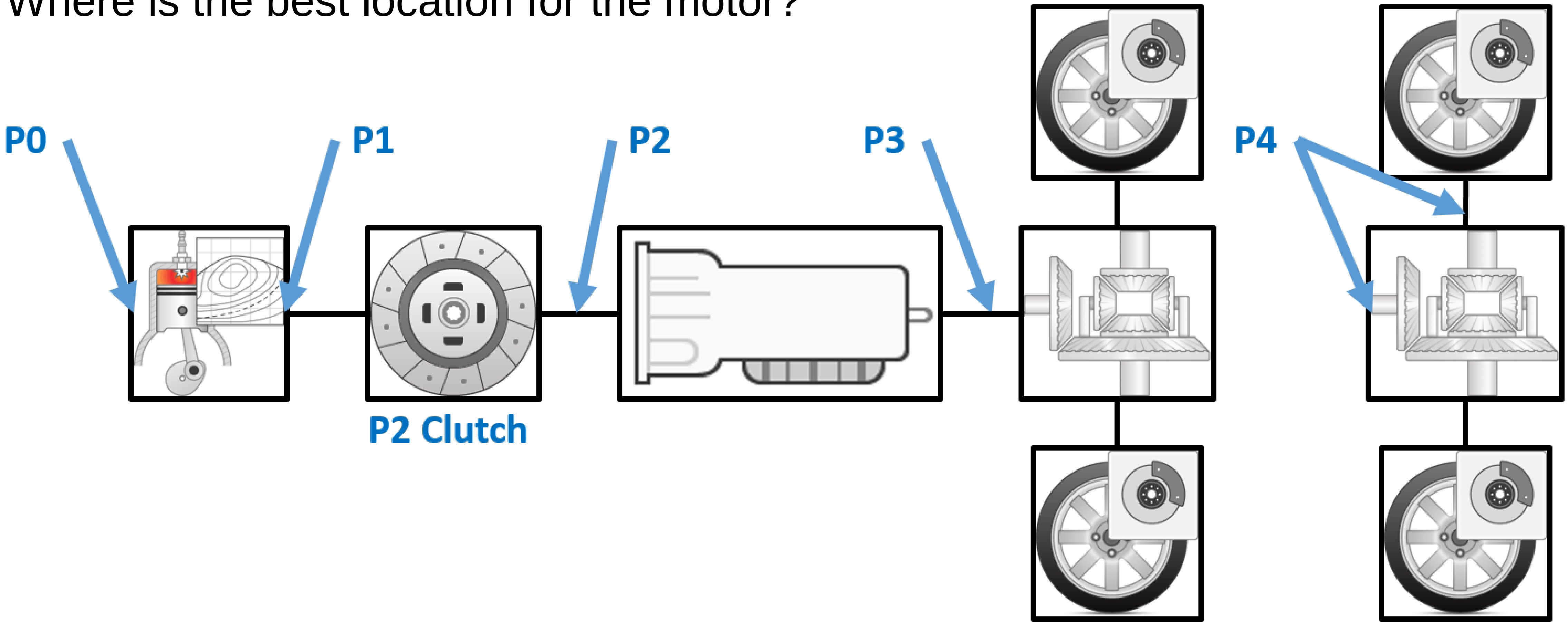


Agenda

- Context
- Case study description
- Tools used
- Plant model and controls
- Results
- Next steps

Electrified Powertrain Selection

- Considering variants of single motor, parallel hybrids
- Where is the best location for the motor?



Problem Statement

- Maximize:
  - Fuel economy (l/100km for drive cycles Highway, City, US06)
  - Acceleration performance ($t_{0\text{-}100km/h}$)

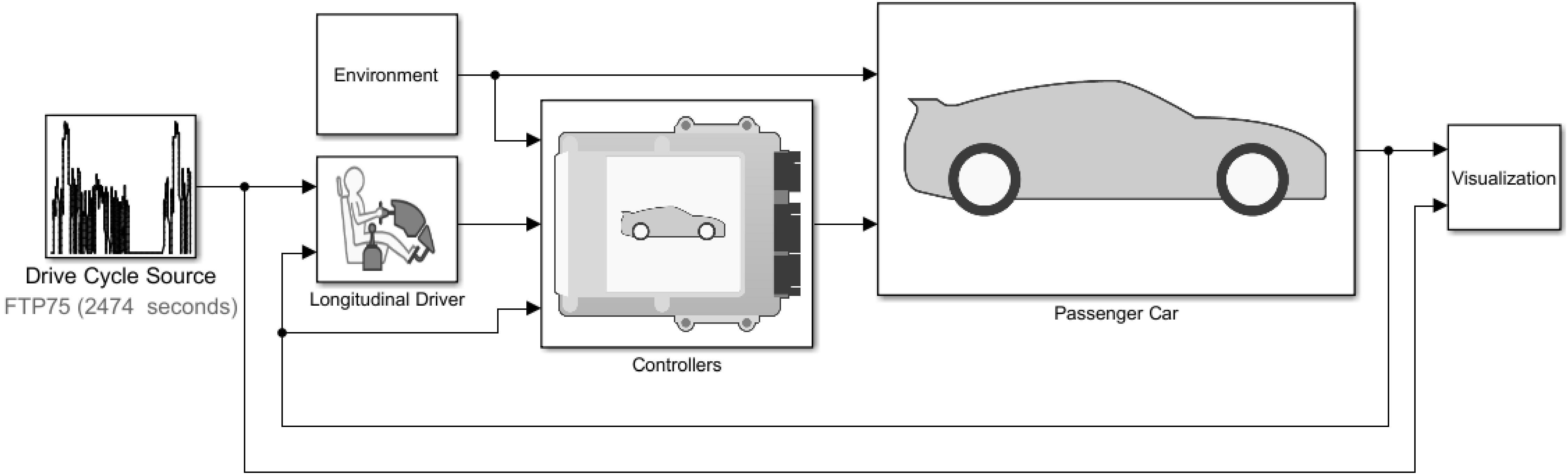
- Subject to:
  - Actuator limits for motor & engine
  - Velocity within 3,2 km/h window of drive cycle target velocity
  - SOC within [SOC_{low}, SOC_{high}]
  - $|SOC_{final} – SOC_{init}| < tol$ → **requires iteration on supervisory control parameter**

Agenda

- Context
- Case study description
- Tools used
- Plant model and controls
- Results
- Next steps

Powertrain Blockset

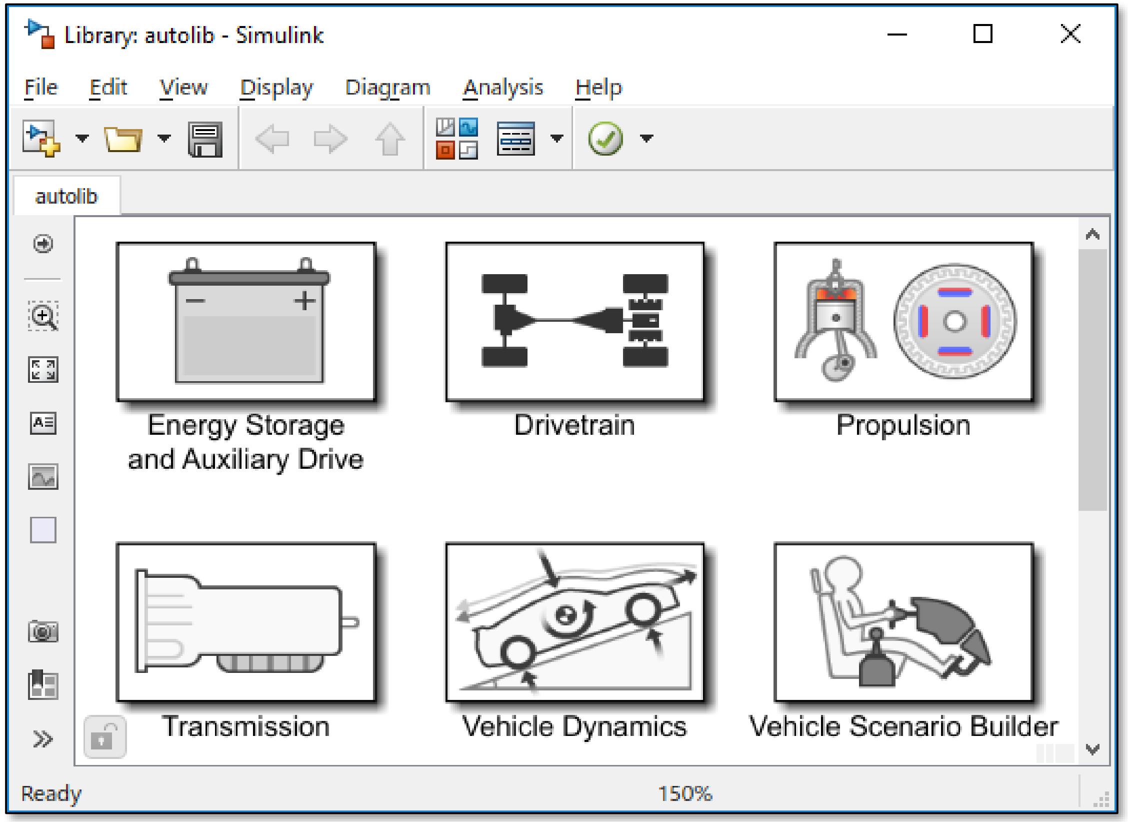
- Goals:
  - Provide starting point for engineers to build **good plant / controller models**
  - Provide **open** and documented models
  - Provide very **fast**-running models that work with popular HIL systems



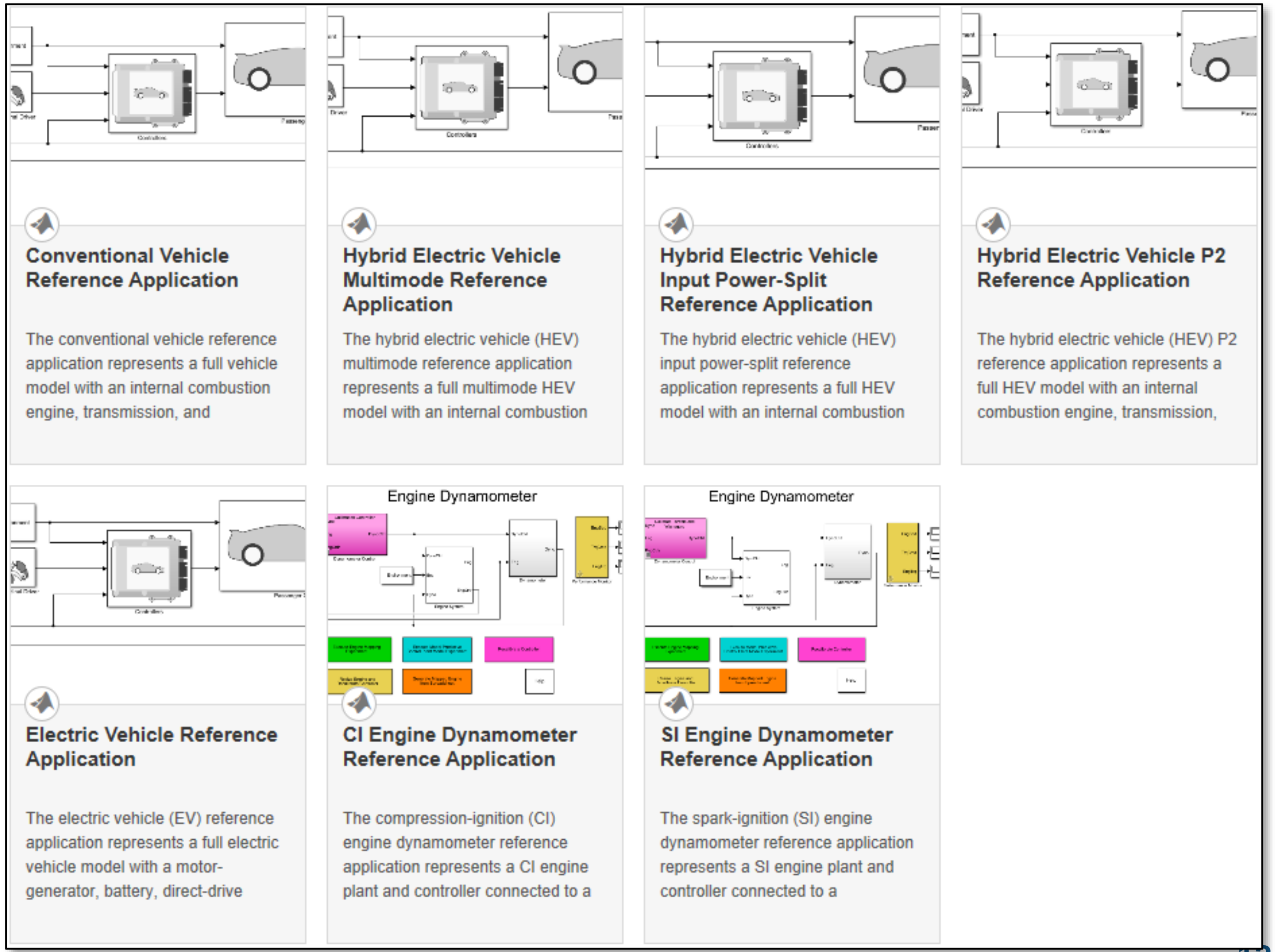
Lower the barrier to entry for Model-Based Design

Powertrain Blockset Features

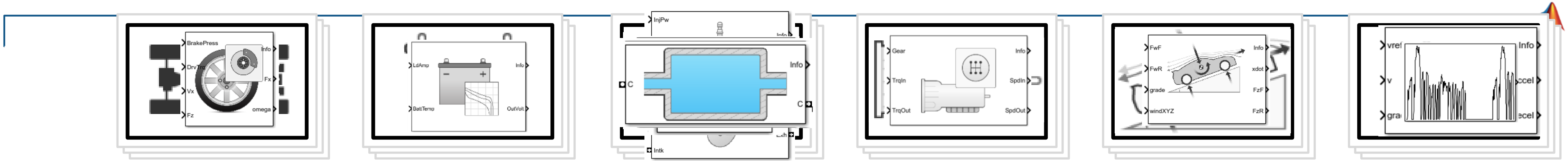
Library of blocks



Pre-built reference applications



MathWorks



Drivetrain

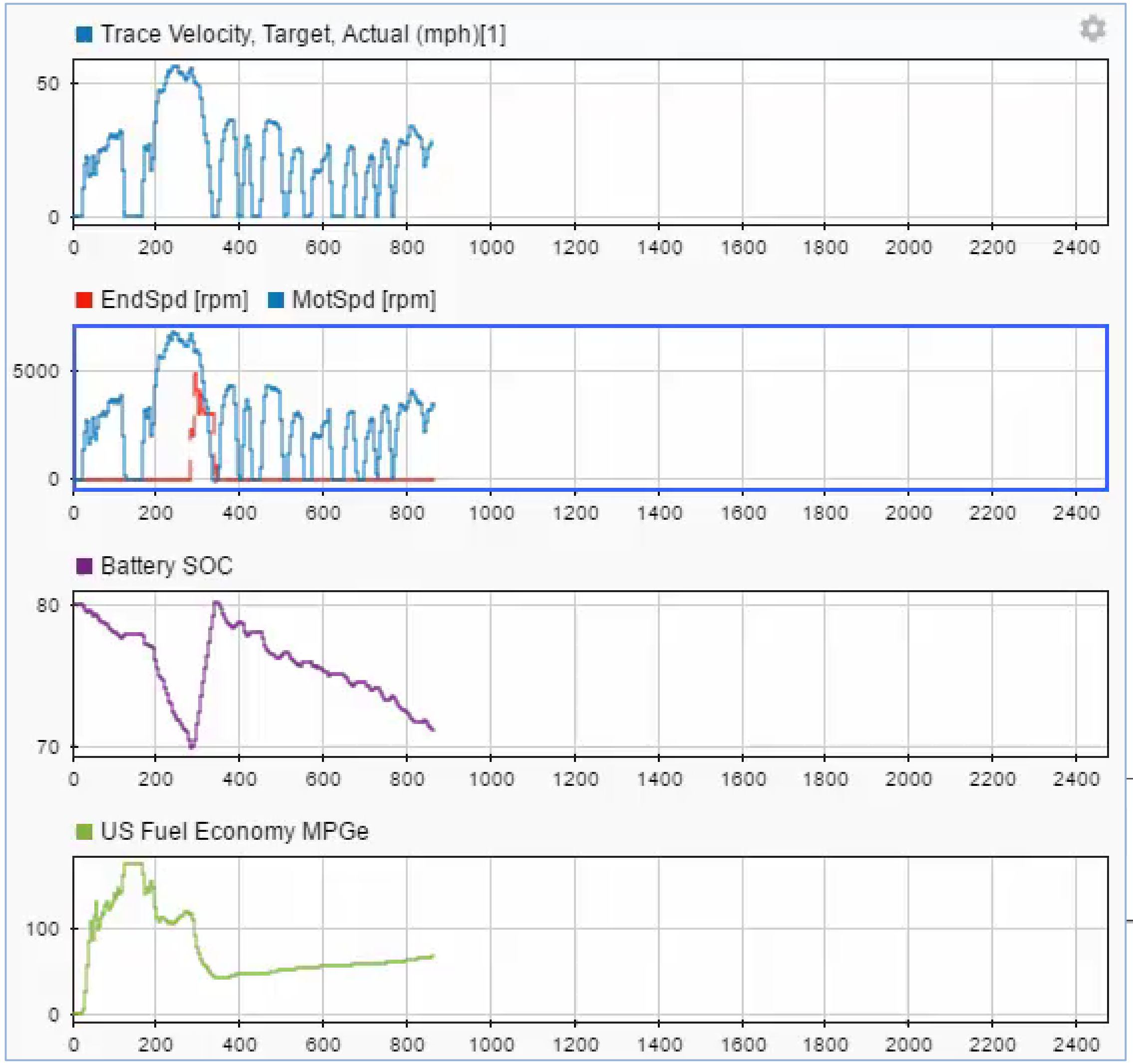
Energy Storage and Auxiliary Drive

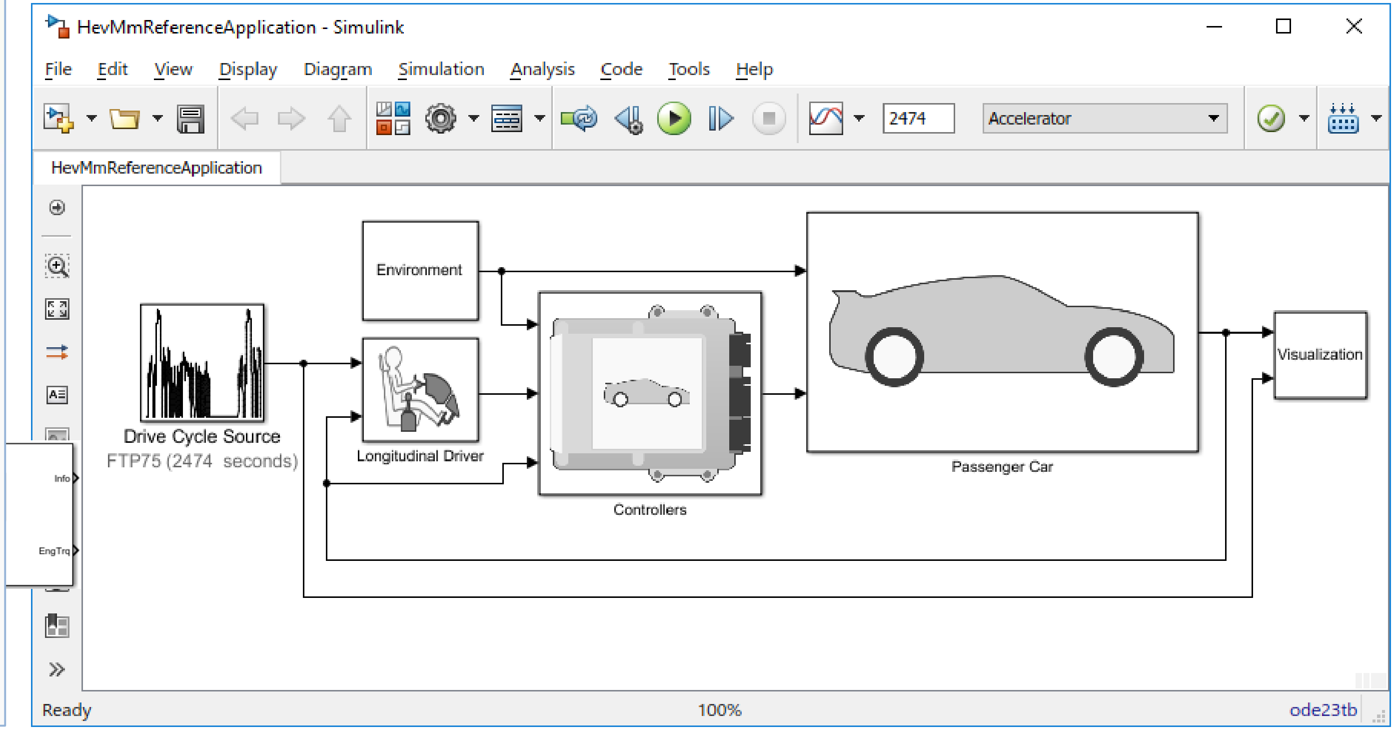
Propulsion

Transmission

Vehicle Dynamics

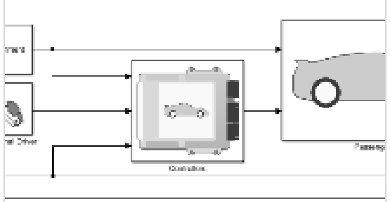
Vehicle Scenario Builder





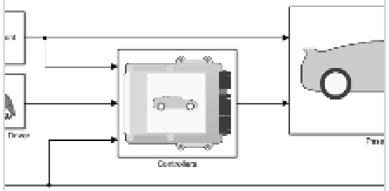
Reference Applications

Full Vehicle Models



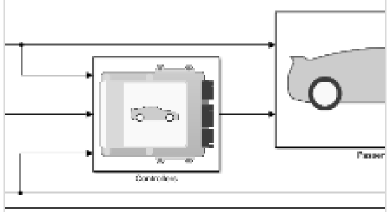
Conventional Vehicle Reference Application

The conventional vehicle reference application represents a full vehicle model with an internal combustion engine, transmission, and



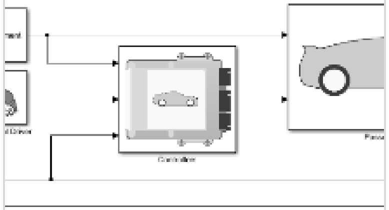
Hybrid Electric Vehicle Multimode Reference Application

The hybrid electric vehicle (HEV) multimode reference application represents a full multimode HEV model with an internal combustion



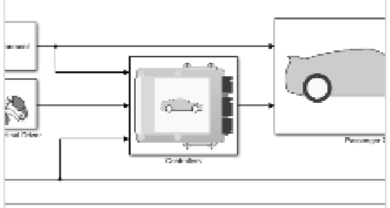
Hybrid Electric Vehicle Input Power-Split Reference Application

The hybrid electric vehicle (HEV) input power-split reference application represents a full HEV model with an internal combustion



Hybrid Electric Vehicle P2 Reference Application

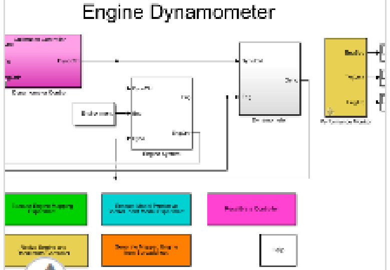
The hybrid electric vehicle (HEV) P2 reference application represents a full HEV model with an internal combustion engine, transmission,



Electric Vehicle Reference Application

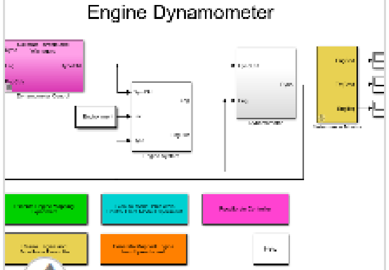
The electric vehicle (EV) reference application represents a full electric vehicle model with a motor-generator, battery, direct-drive

Virtual Engine Dynamometers



CI Engine Dynamometer Reference Application

The compression-ignition (CI) engine dynamometer reference application represents a CI engine plant and controller connected to a

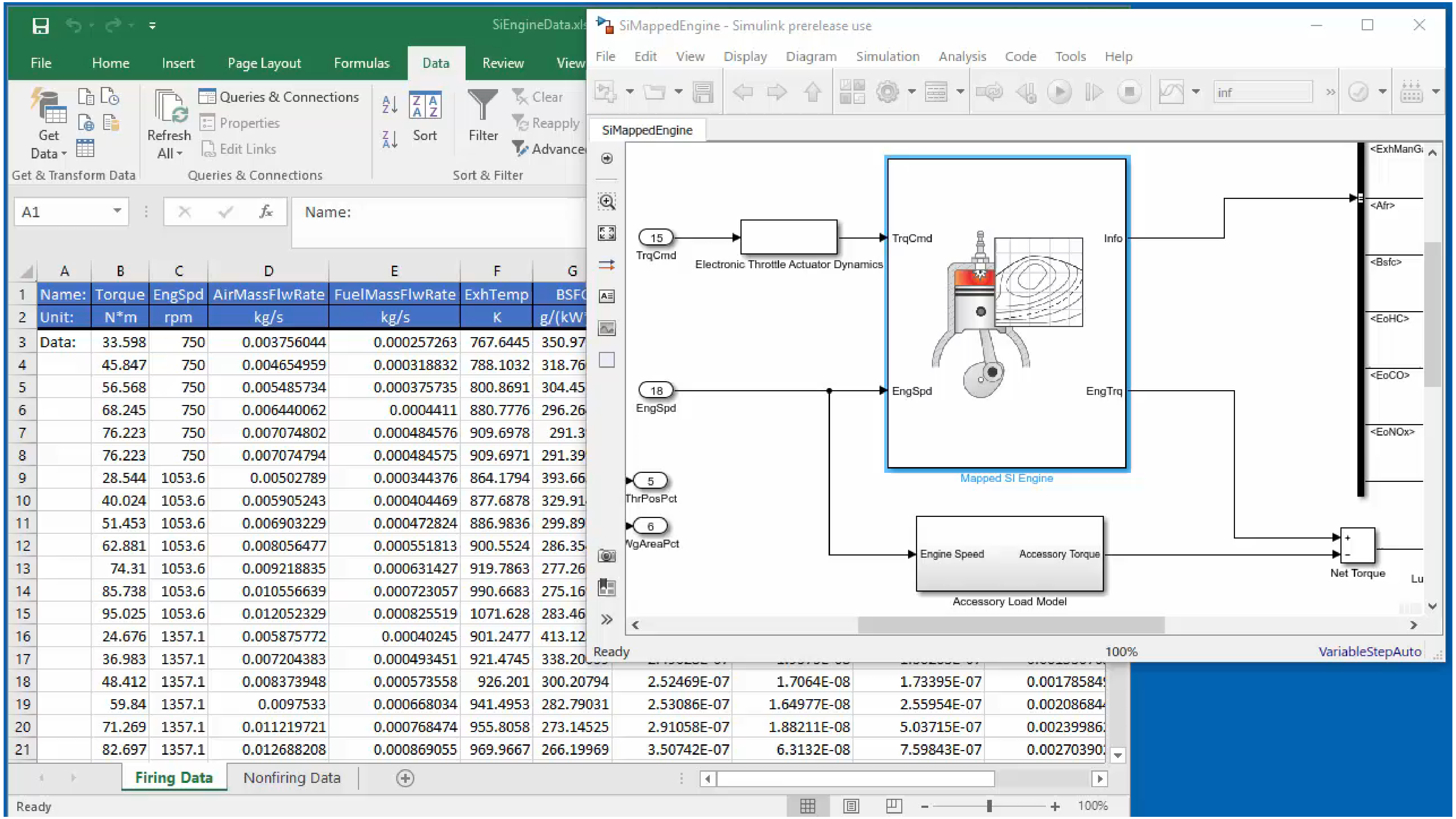


SI Engine Dynamometer Reference Application

The spark-ignition (SI) engine dynamometer reference application represents a SI engine plant and controller connected to a

What's New in R2018b ?

Engine Test Data Import

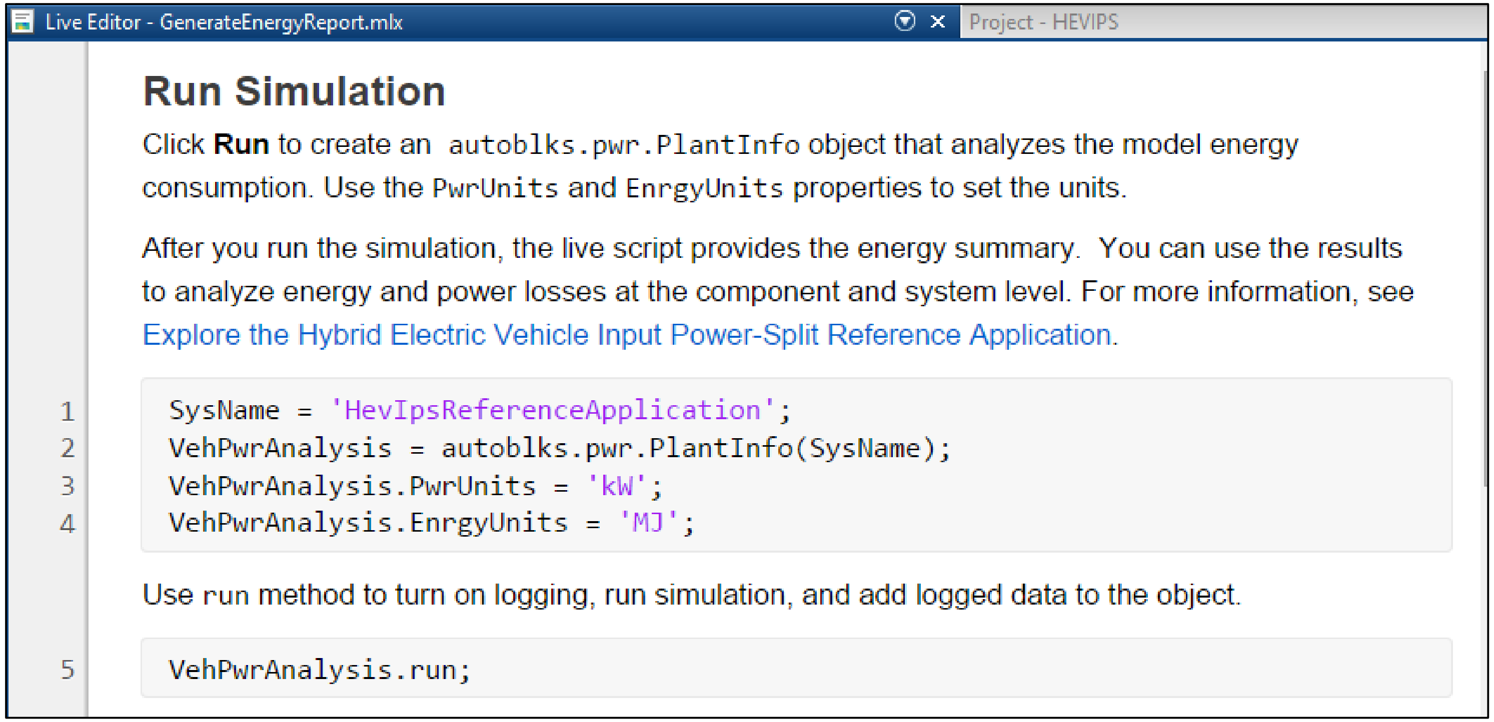


	A	B	C	D	E	F	G	H	I	J	K
1	Name:	Torque	EngSpd	AirMassFlwRate	FuelMassFlwRate	ExhTemp	BSFC				
2	Unit:	N*m	rpm	kg/s	kg/s	K	g/(kW				
3	Data:	33.598	750	0.003756044	0.000257263	767.6445	350.97				
4		45.847	750	0.004654959	0.000318832	788.1032	318.76				
5		56.568	750	0.005485734	0.000375735	800.8691	304.45				
6		68.245	750	0.006440062	0.0004411	880.7776	296.26				
7		76.223	750	0.007074802	0.000484576	909.6978	291.3				
8		76.223	750	0.007074794	0.000484575	909.6971	291.39				
9		28.544	1053.6	0.00502789	0.000344376	864.1794	393.66				
10		40.024	1053.6	0.005905243	0.000404469	877.6878	329.91				
11		51.453	1053.6	0.006903229	0.000472824	886.9836	299.89				
12		62.881	1053.6	0.008056477	0.000551813	900.5524	286.35				
13		74.31	1053.6	0.009218835	0.000631427	919.7863	277.26				
14		85.738	1053.6	0.010556639	0.000723057	990.6683	275.16				
15		95.025	1053.6	0.012052329	0.000825519	1071.628	283.46				
16		24.676	1357.1	0.005875772	0.00040245	901.2477	413.12				
17		36.983	1357.1	0.007204383	0.000493451	921.4745	338.20				
18		48.412	1357.1	0.008373948	0.000573558	926.201	300.20794	2.52469E-07	1.7064E-08	1.73395E-07	0.00178584
19		59.84	1357.1	0.0097533	0.000668034	941.4953	282.79031	2.53086E-07	1.64977E-08	2.55954E-07	0.00208684
20		71.269	1357.1	0.011219721	0.000768474	955.8058	273.14525	2.91058E-07	1.88211E-08	5.03715E-07	0.00239986
21		82.697	1357.1	0.012688208	0.000869055	969.9667	266.19969	3.50742E-07	6.3132E-08	7.59843E-07	0.00270390

What's New in R2019a ?

Energy Accounting and Reporting

- Simulate
  - Turn on logging
  - Run simulation
  - Check conservation of energy


Live Editor - GenerateEnergyReport.mlx
Project - HEVIPS

Run Simulation

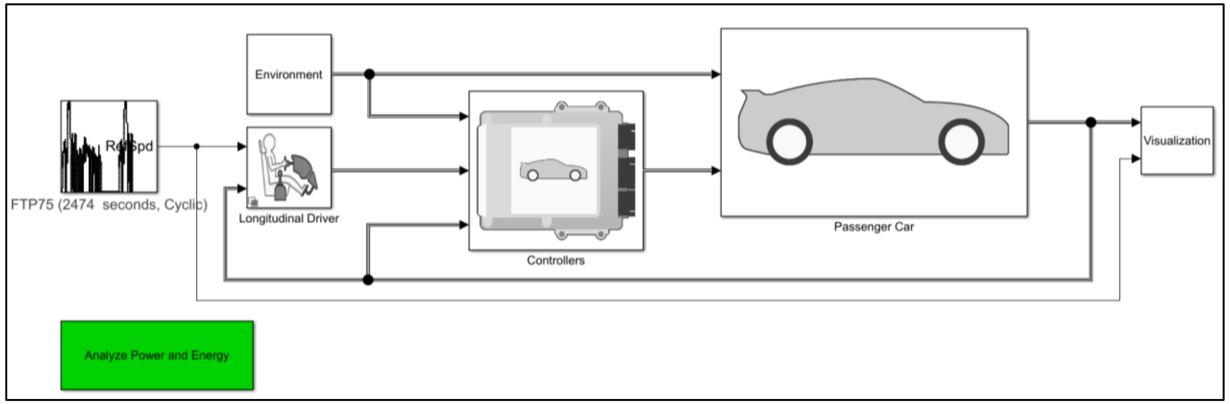
Click **Run** to create an `autoblks.pwr.PlantInfo` object that analyzes the model energy consumption. Use the `PwrUnits` and `EnrgyUnits` properties to set the units.

After you run the simulation, the live script provides the energy summary. You can use the results to analyze energy and power losses at the component and system level. For more information, see Explore the Hybrid Electric Vehicle Input Power-Split Reference Application.

```
SysName = 'HevIpsReferenceApplication';
VehPwrAnalysis = autoblks.pwr.PlantInfo(SysName);
VehPwrAnalysis.PwrUnits = 'kW';
VehPwrAnalysis.EnrgyUnits = 'MJ';
```

Use `run` method to turn on logging, run simulation, and add logged data to the object.

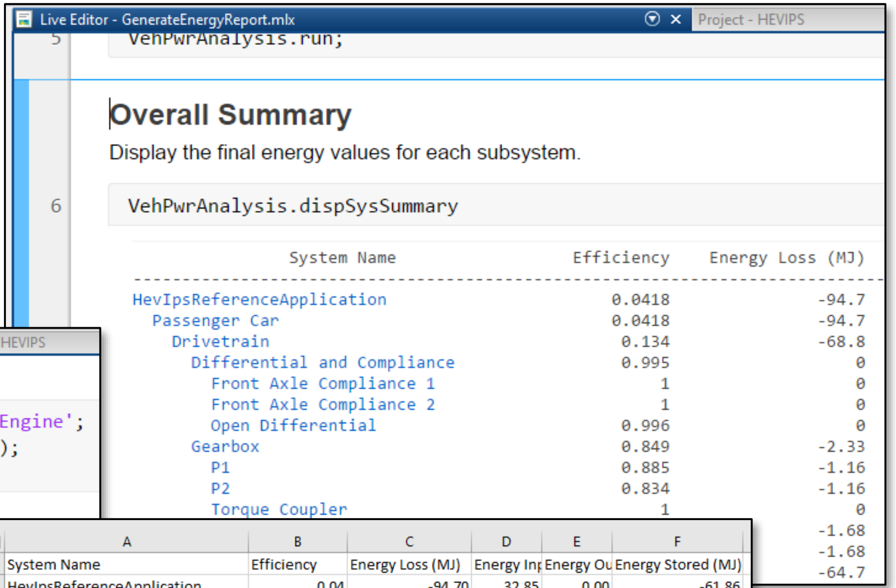
```
VehPwrAnalysis.run;
```

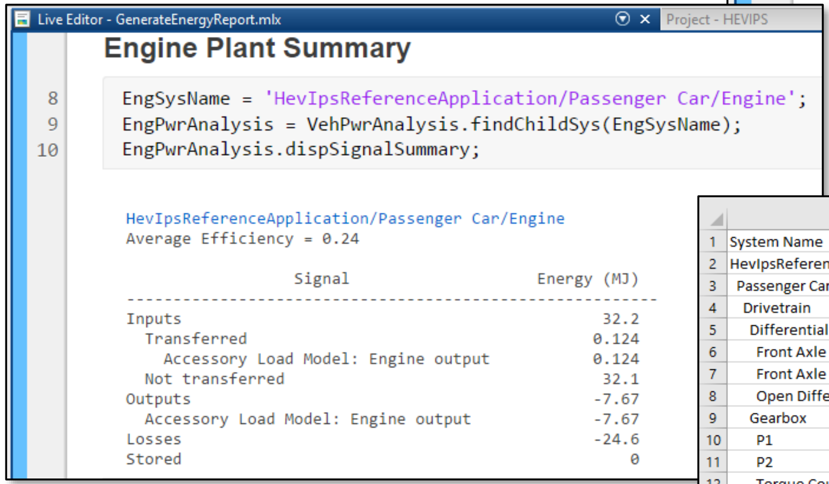


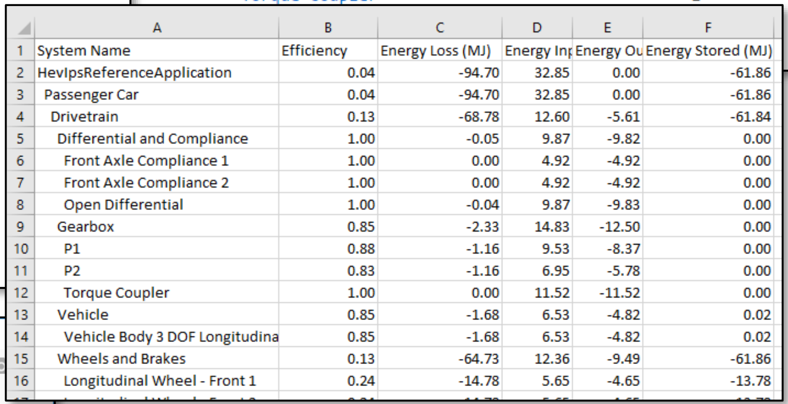
What's New in R2019a ?

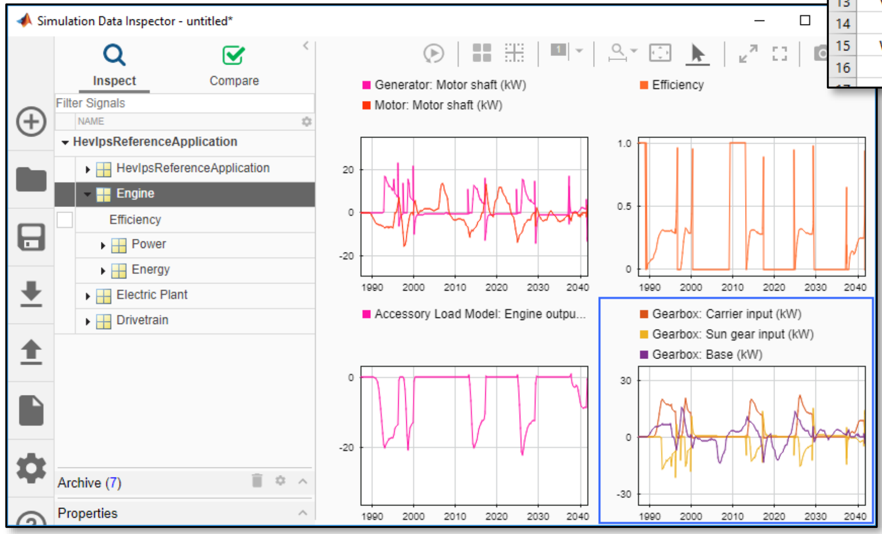
Energy Accounting and Reporting

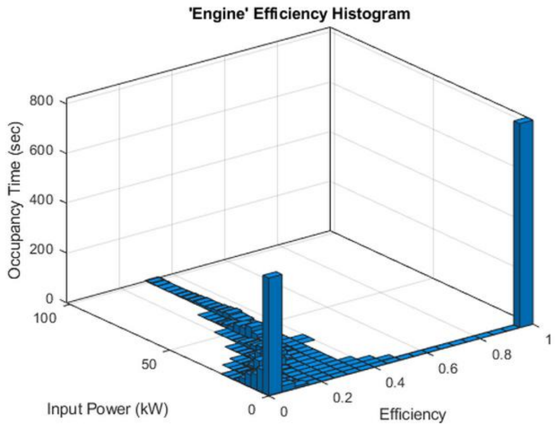
- Simulate
  - Turn on logging
  - Run simulation
  - Check conservation of energy
- Report results
  - System level summary
  - Subsystem detailed view
  - Excel export
  - Efficiency histogram
  - Time trace plots









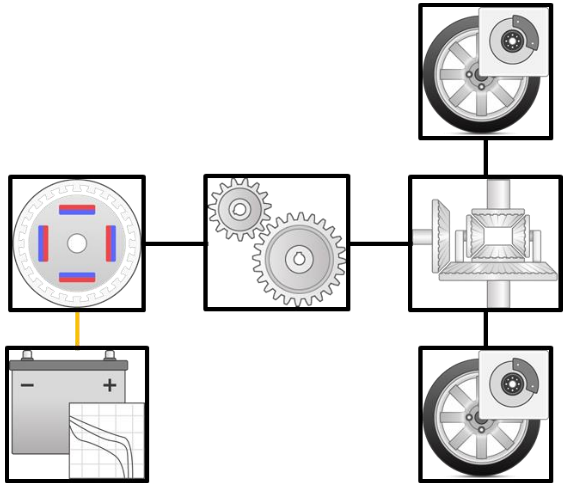


Agenda

- Context
- Case study description
- Tools used
- Plant model and controls
- Results
- Next steps

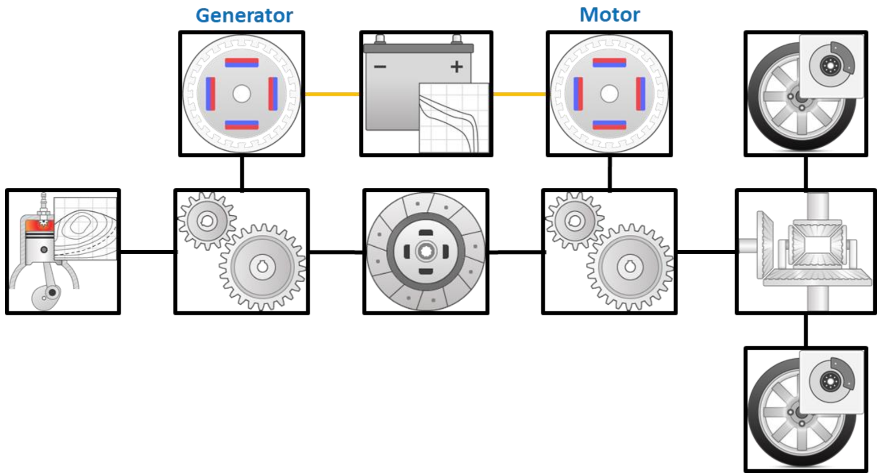
EV / HEV Configurations Shipping with Powertrain Blockset

Pure EV



- Released in: R2016b
- Similar powertrains:
  - Nissan Leaf
  - Tesla Roadster
  - Chevy Bolt

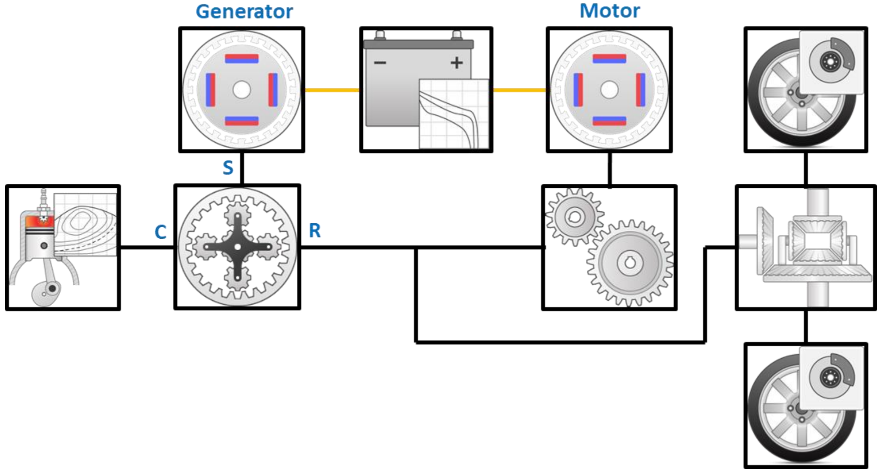
Multi-mode HEV → P1/P3



- Released in: R2016b
- Similar powertrains:
  - Hybrid Honda Accord

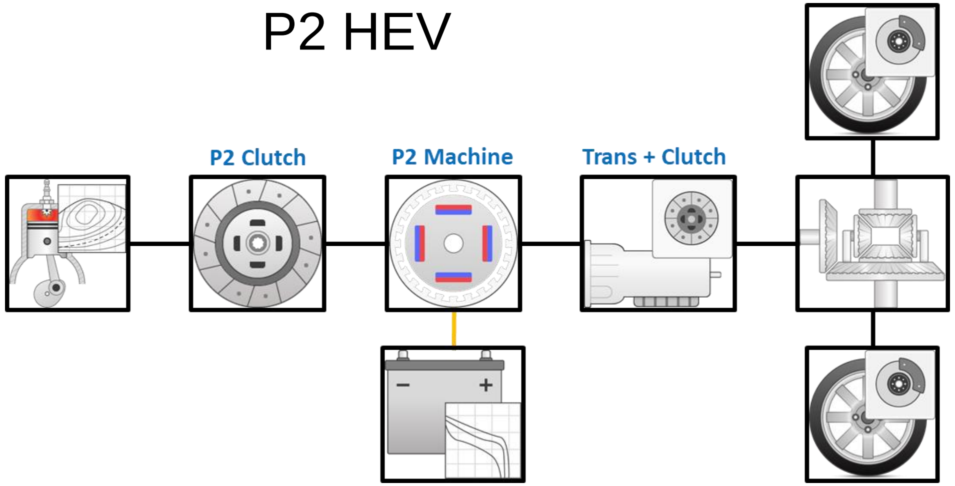
EV / HEV Configurations Shipping with Powertrain Blockset

Input Power-Split HEV



- Released in: R2017b
- Similar powertrains:
  - Toyota Prius
  - Lexus Hybrid
  - Ford Hybrid Escape

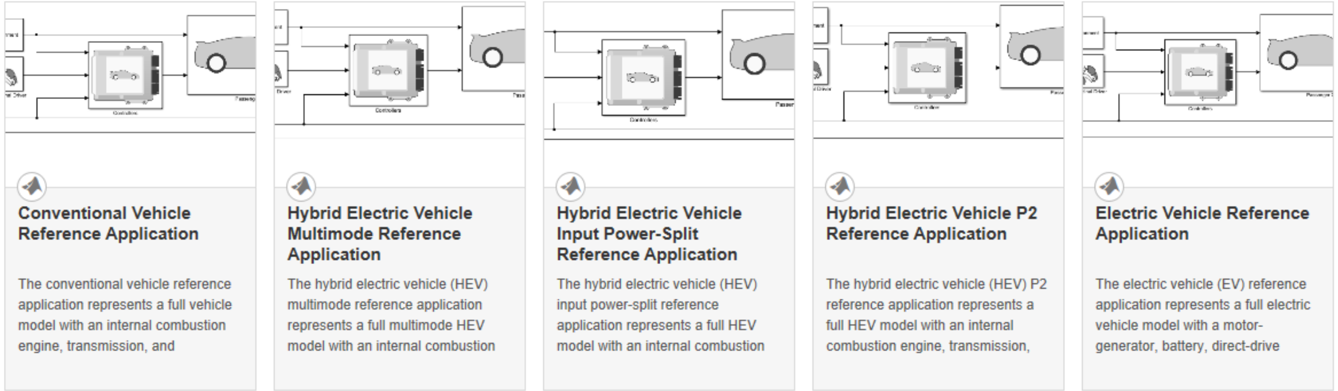
P2 HEV



- Released in: R2018b
- Similar powertrains:
  - Nissan Pathfinder
  - Hyundai Sonata
  - Kia Optima

Flexible Modeling Framework

1. Choose a vehicle configuration
   – Select a reference application as a starting point



2. Customize the plant model
   – Parameterize the components
   – Customize existing subsystems
   – Add your own subsystem variants

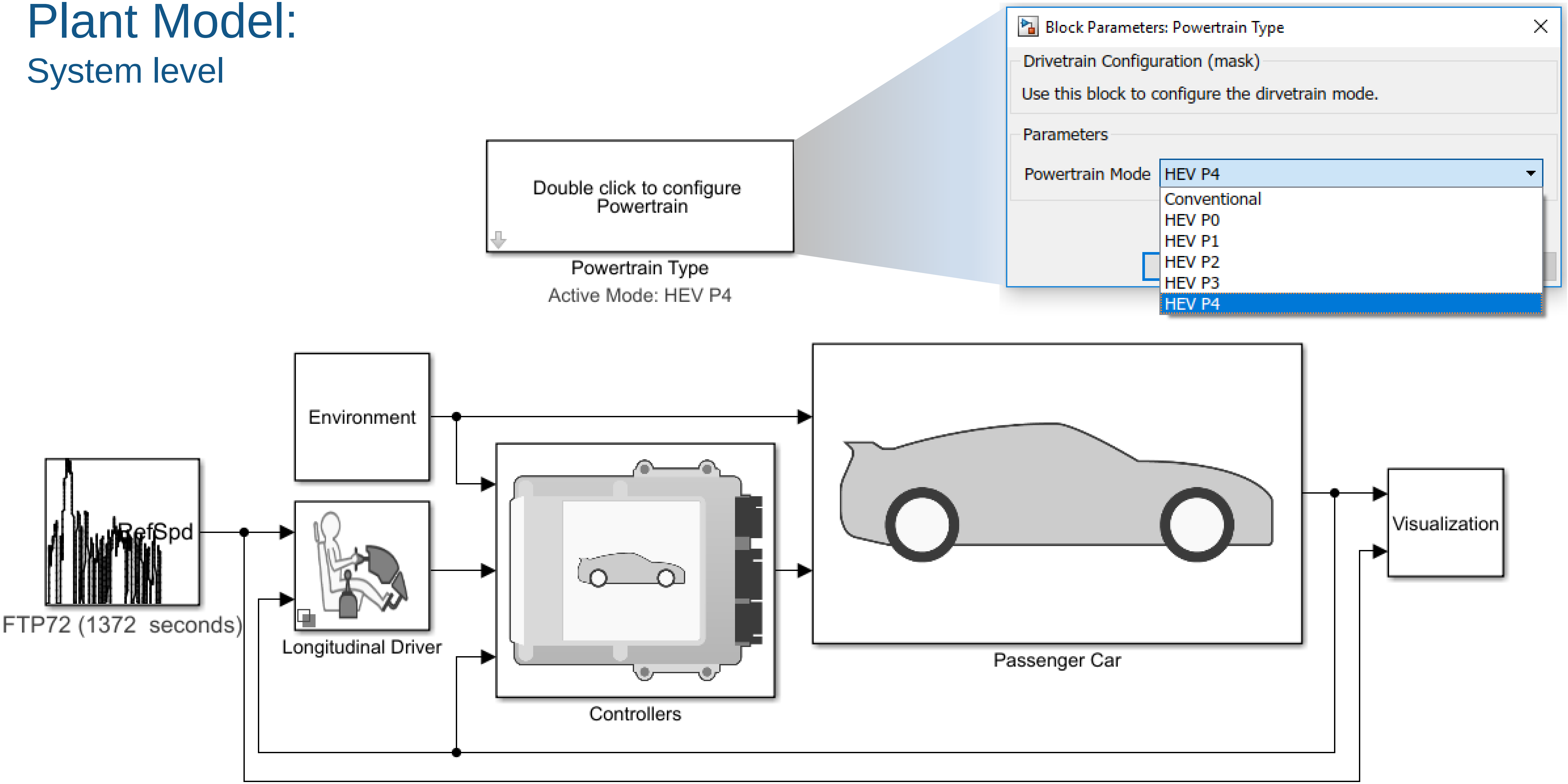
3. Customize the controllers
   – Parameterize the controllers
   – Customize supervisory control logic
   – Add your own controller variants

4. Perform closed-loop system testing
   – Sensitivity analyses
   – Design optimization
   – MIL / SIL / HIL testing

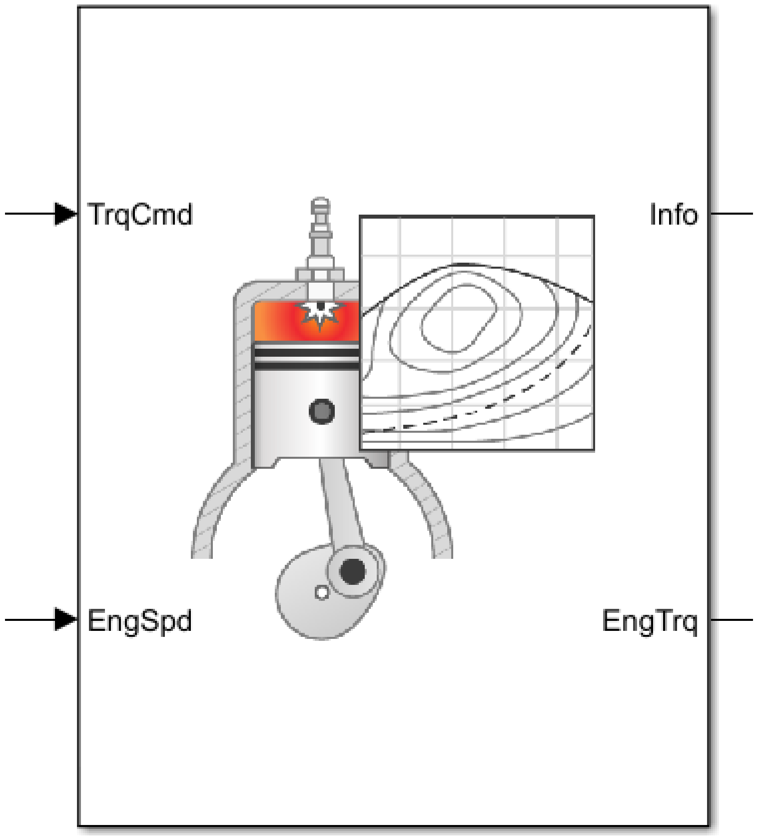
EcoCAR: Mobility Challenge

- What is it?
  - Student competition for 12 North American universities
  - Collaboration of industry, academia and government research labs
  - Improve fuel economy through hybridization and enable level 2 automation capabilities
- MathWorks provided Powertrain Blockset reference applications:
  - Plant models for P0 – P4 architectures
  - Supervisory controller
- Work reused as starting point for powertrain for this work

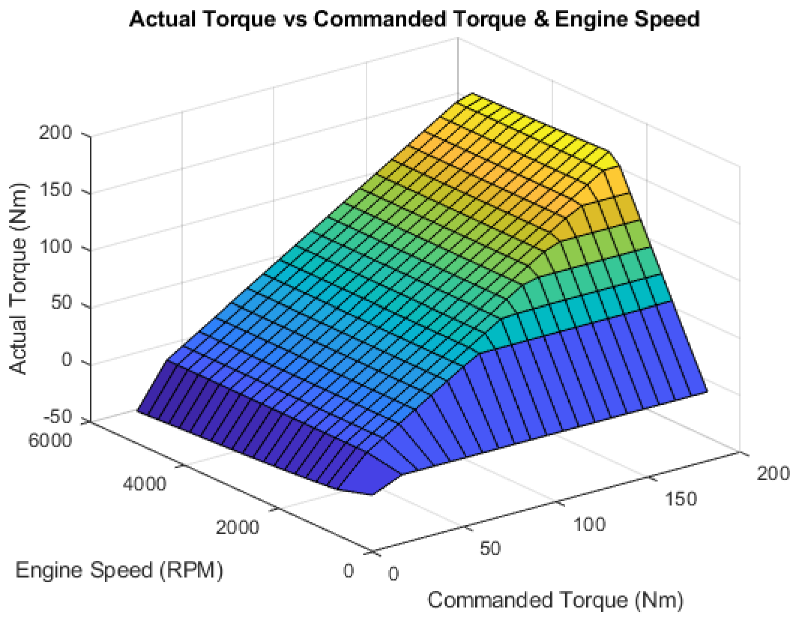
Plant Model:
System level

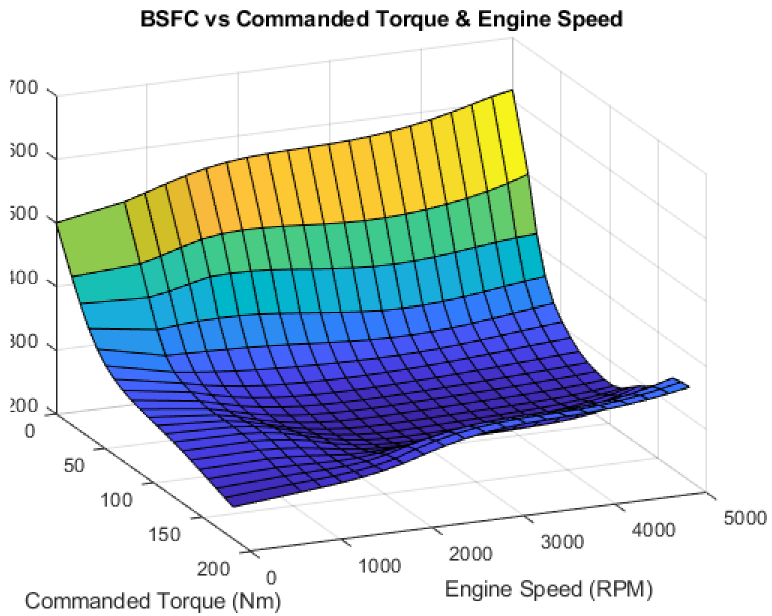


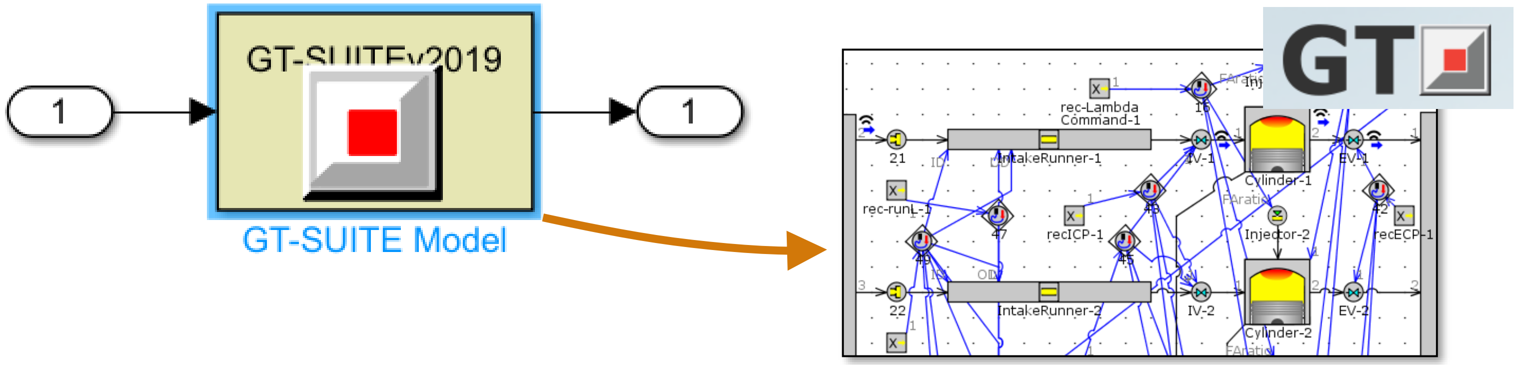
Plant Model:
Engine Subsystem



1.5L Gasoline Engine

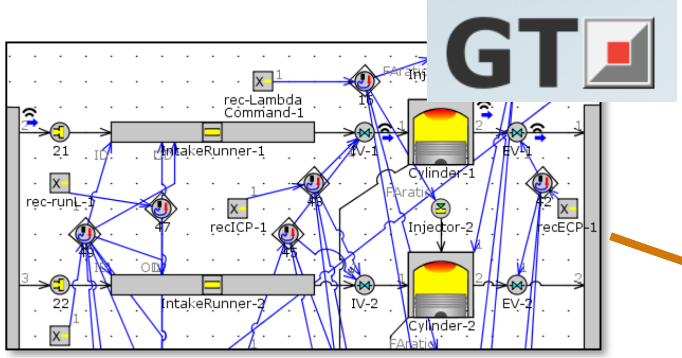




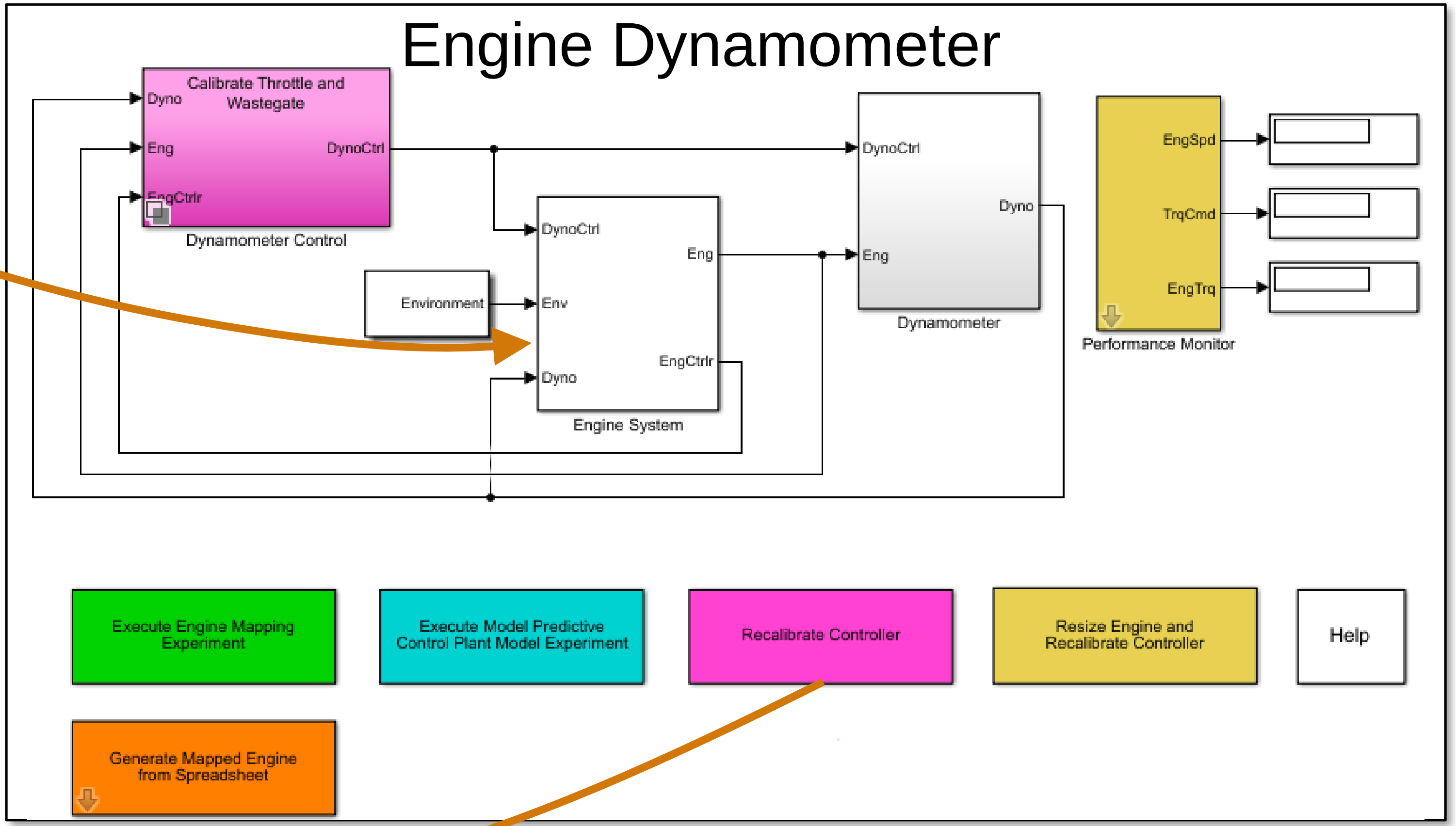


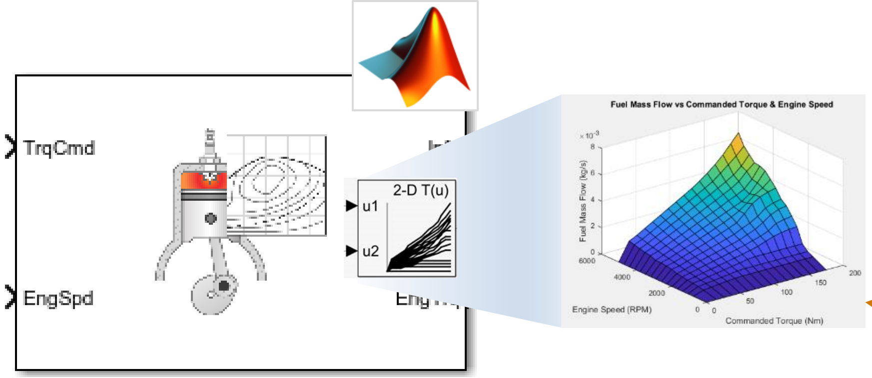
Maps generated from GT-POWER

Controls-oriented Model Creation



Detailed, design-oriented model

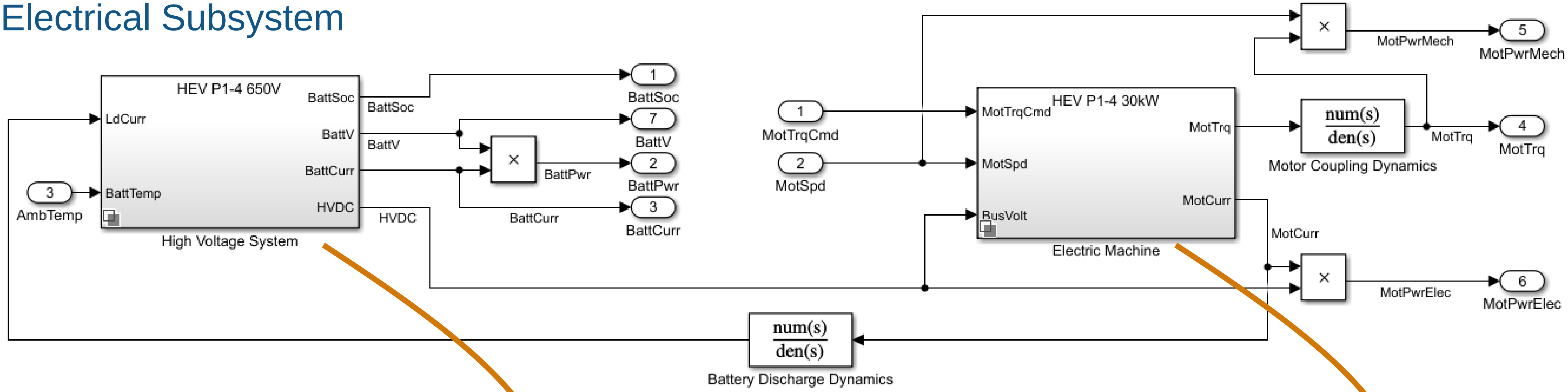


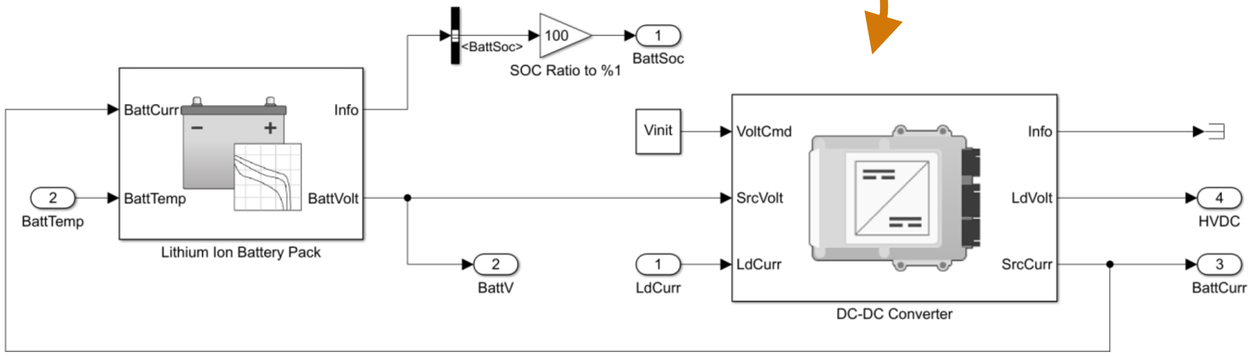


Fast, but accurate controls-oriented model

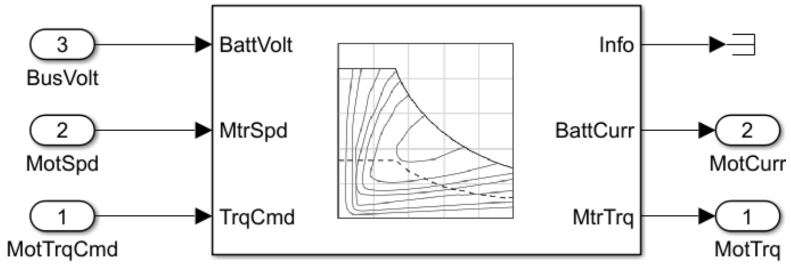
Plant Model:

Electrical Subsystem





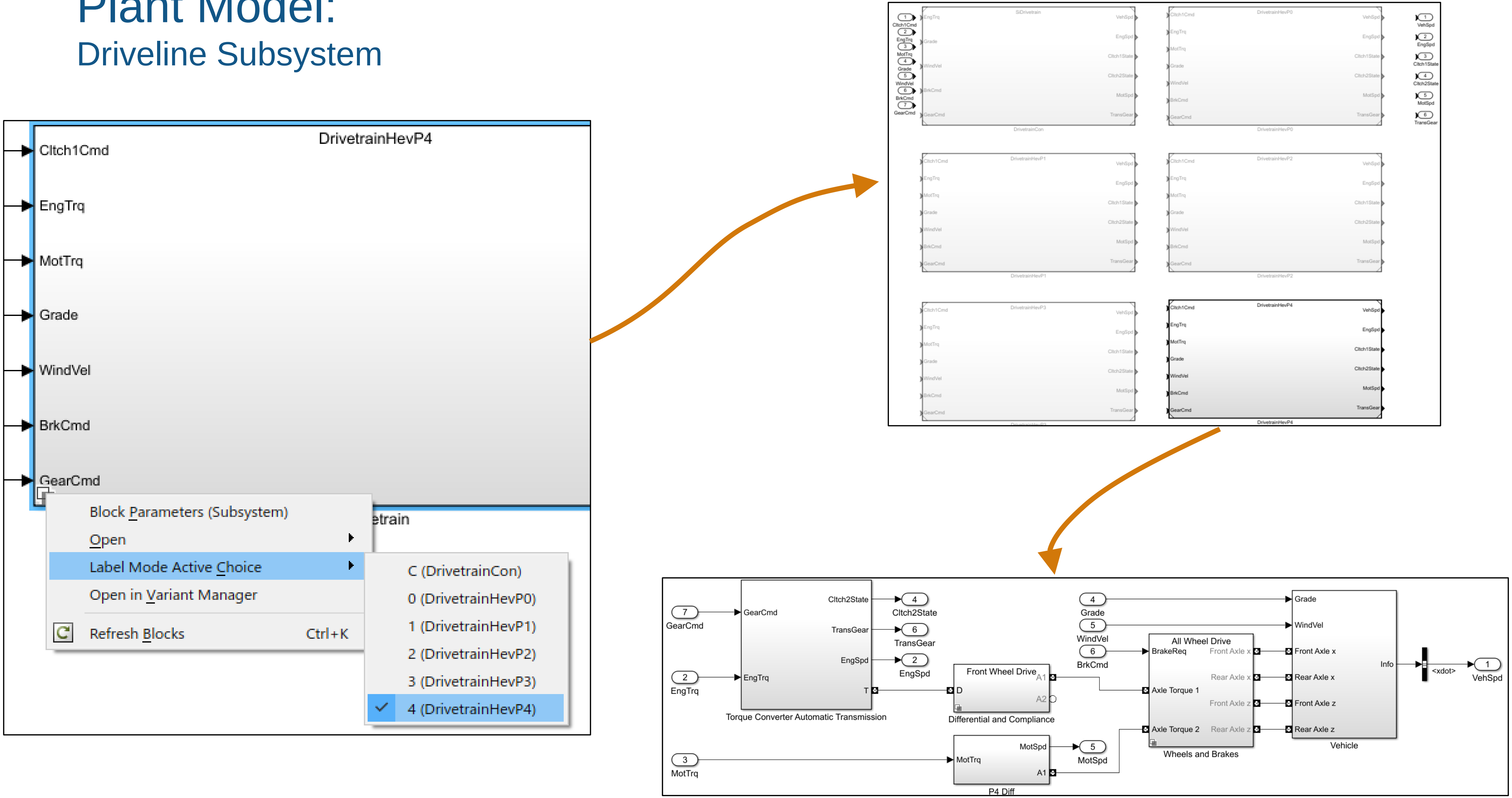
650 V Battery & DC-DC Converter
(smaller sizing for P0)



30 kW Motor
(10 kW for P0)

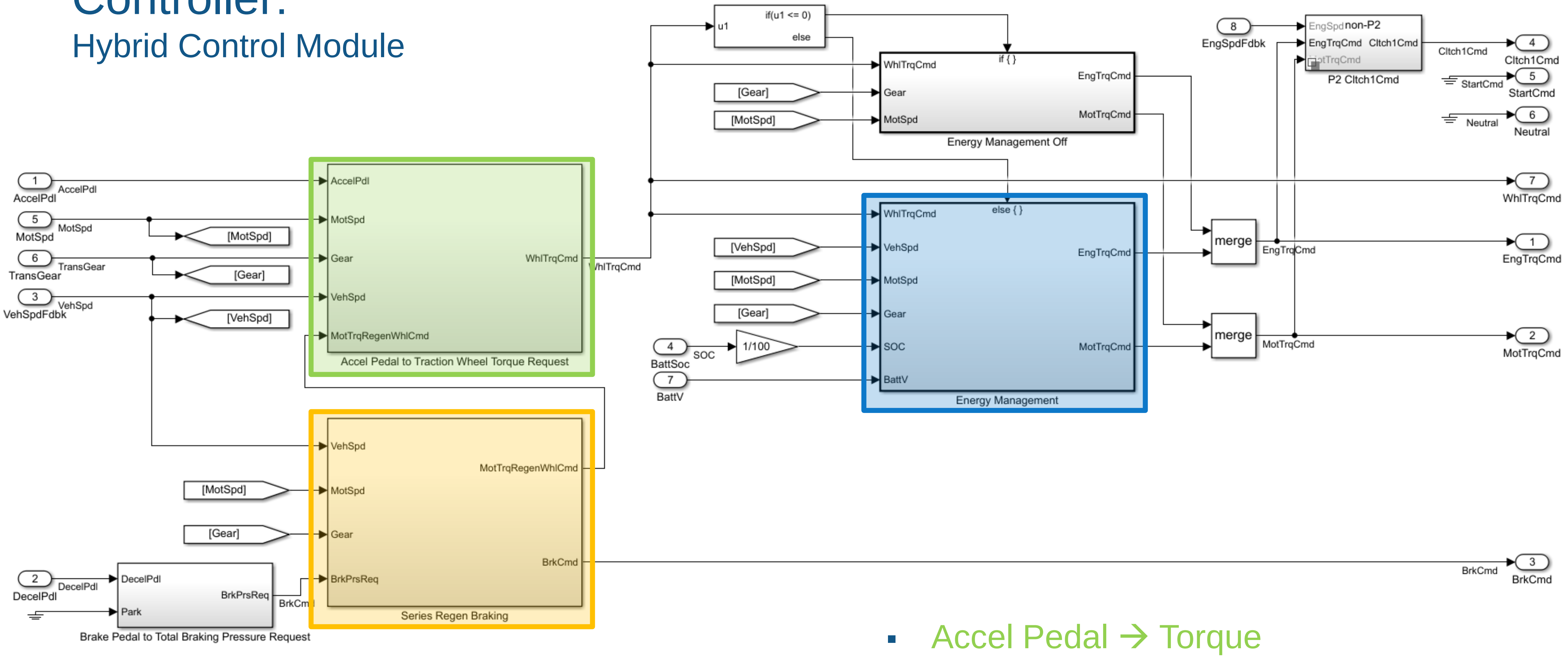
Plant Model:

Driveline Subsystem



Controller:

Hybrid Control Module



- Accel Pedal → Torque
- Regenerative Brake Blending
- Energy Management

Equivalent Consumption Minimization Strategy (ECMS)

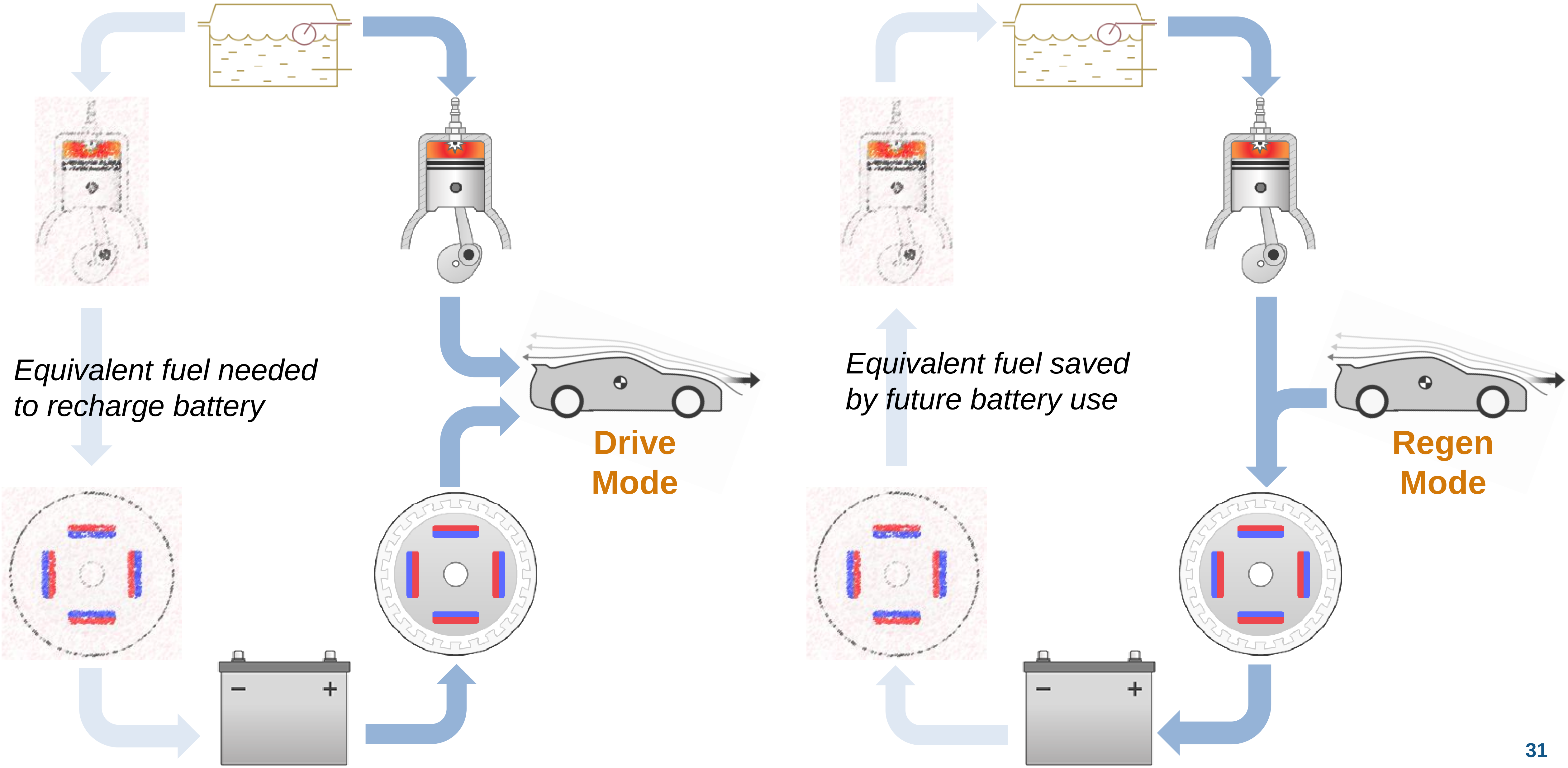
- What is ECMS?
  - Supervisory control strategy to decide when to use engine, motor or both
  - Based on analytical instantaneous optimization

- Why use ECMS?
  - Provides near optimal control if drive cycle is known a priori
  - Can be enhanced with adaptive methods (i.e. Adaptive-ECMS)

$$\min P_{equivalent}(t) = P_{fuel}(t) + s(t) \cdot P_{battery}(t),$$

where s(t) are the “equivalent factors”

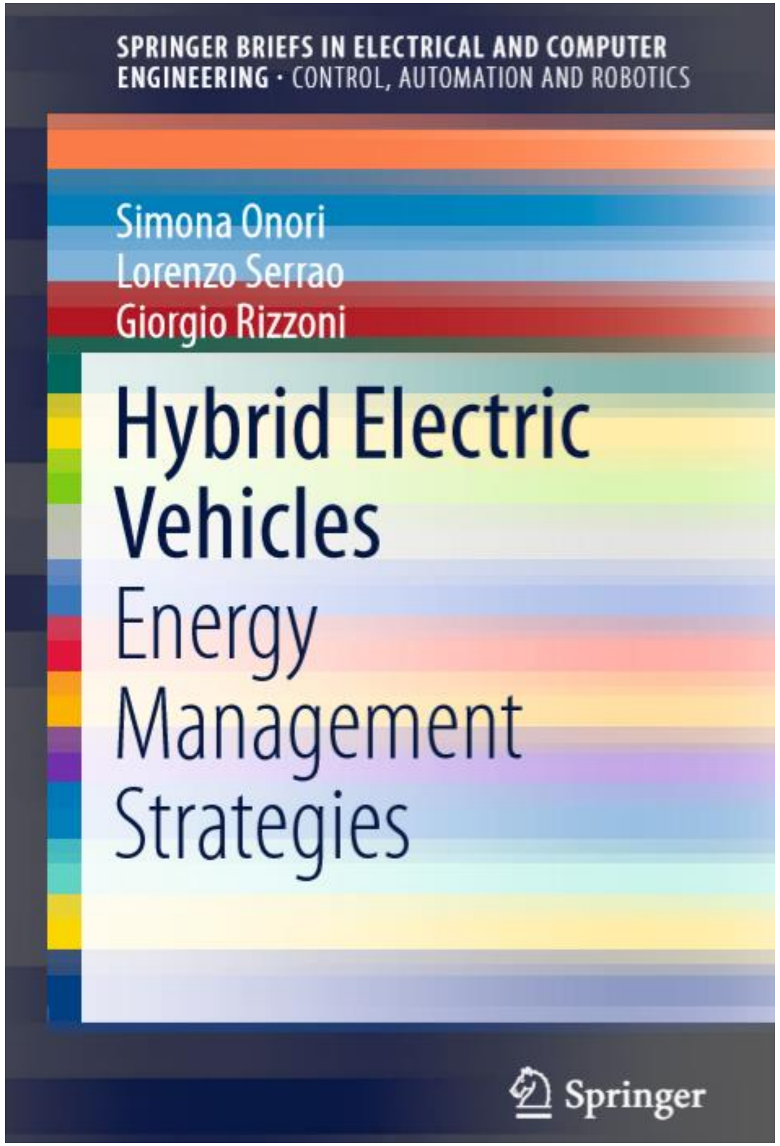
Equivalent Consumption Minimization Strategy (ECMS)



Equivalent Consumption Minimization Strategy (ECMS)

- Collaborated with Dr. Simona Onori from Stanford University

- For more information on ECMS, refer to:



https://www.springer.com/us/book/9781447167792

Agenda

- Context
- Case study description
- Tools used
- Plant model and controls
- Results
- Next steps

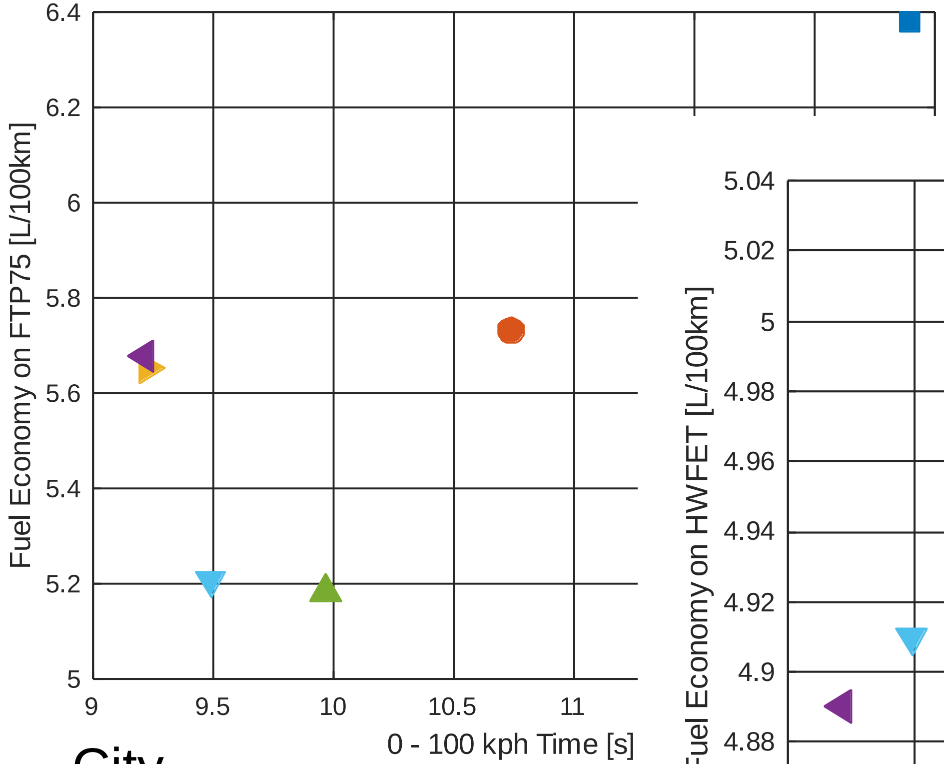
Methodology

- Generate Powertrain Blockset mapped engine from GT-POWER model

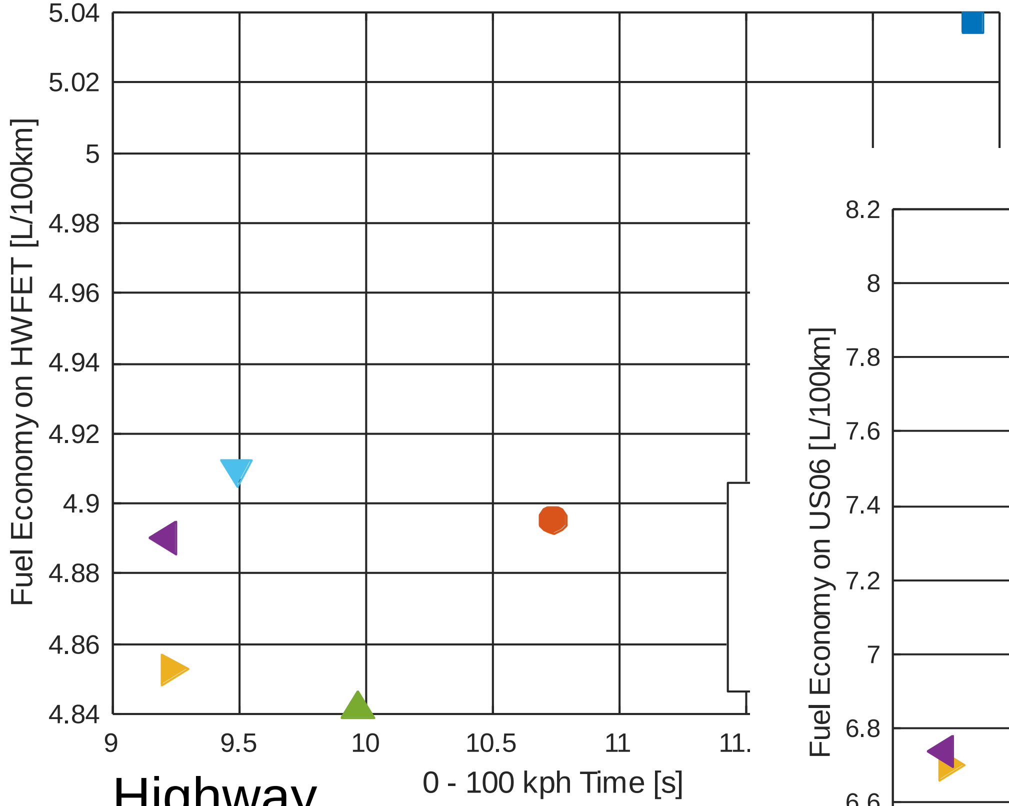
- For each P_i architecture:
  - Using mapped engine model, iterate on s (controller parameter) to achieve dSOC $\leq$ 1% across each drive cycle
  - Assess fuel economy on city, highway and US06 drive cycles
  - Assess acceleration performance on Wide Open Throttle (WOT) test

- Compare fuel economy and performance across P0 – P4 architectures

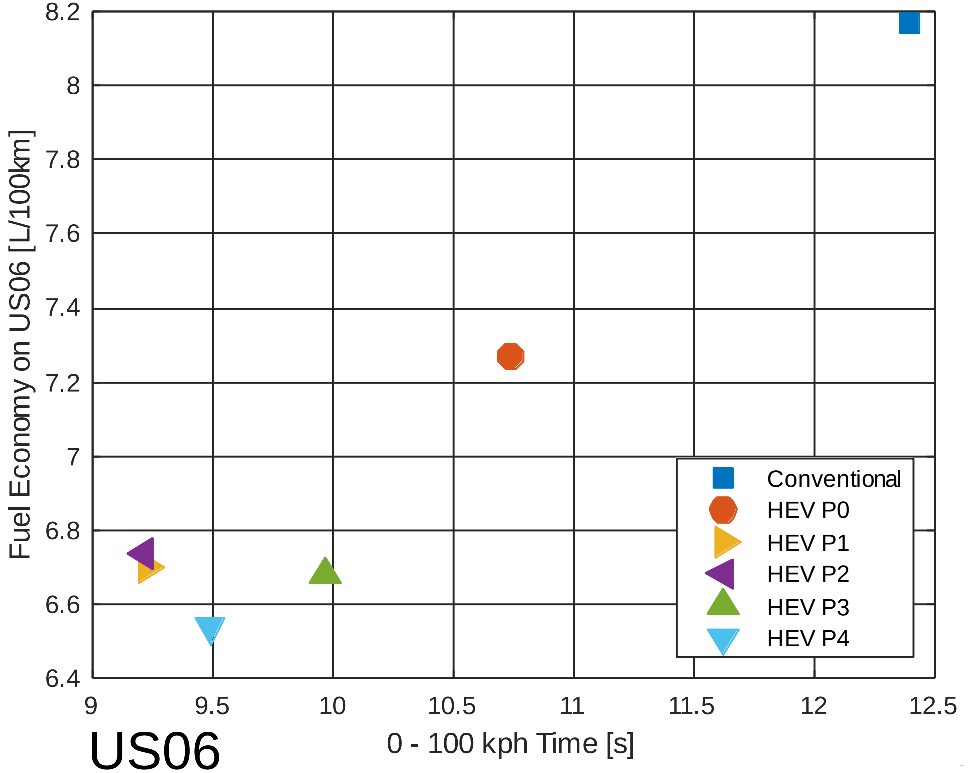
Results



City

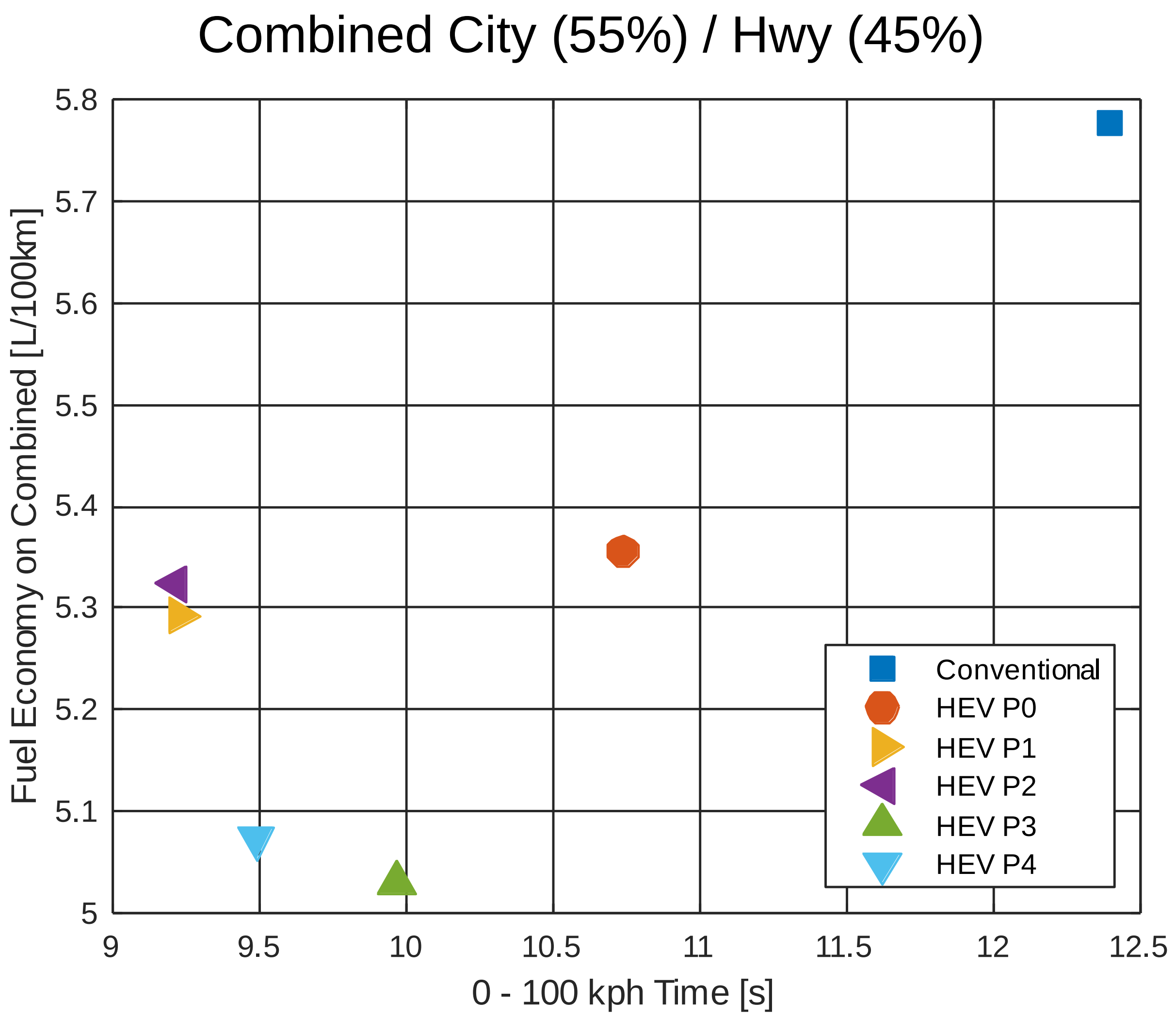


Highway



US06

Results



- ECMS provides a fair comparison of alternatives
- Placing motors closer to the drive wheel:
  - Improves fuel economy (better regen efficiency)
  - Degrades performance (lower mechanical advantage)
- Simulation allows you to quantify the tradeoff

Agenda

- Context
- Case study description
- Tools used
- Plant model and controls
- Results
- Next steps

Summary

- Assembled full vehicle simulation
  - Powertrain Blockset as framework for vehicle level modeling
  - Mapped engine models auto-generated from design-oriented engine model
  - ECMS for supervisory controls strategy applicable to all P0 – P4 variants

- Assessed fuel economy / performance across several variants
  - Iterated on controller parameter to identify charge neutral settings
  - Generated pareto curve to quantify tradeoff between variants

Next Steps

- Widen the scope of powertrain selection study
  - Include two-motor HEV's, with modified ECMS controls
  - Search over design parameters (final drive ratio, battery capacity, etc.)

- Conduct more in-depth analysis
  - Assess additional attributes of interest by including more design-oriented models (engine, aftertreatment, drivability, etc.)
  - Integrate control features from advanced development / production

- Continue along the V-cycle
  - Once field candidates are narrowed down to a few options, conduct more detailed electrification study (motor controls, battery design, etc.)
  - Once vehicle platform is selected, calibrate vehicle (drivability, etc.)

Thank You

Eva Pelster
Application Engineer
Eva.Pelster@mathworks.com