

Using Model-Based Calibration Toolbox Multimodels for Cycle-Optimized Diesel Calibration

Joshua Styron, Ph.D.

Ford Motor Company

MathWorks Automotive Conference 2008

June 4, 2008





Outline

- Motivation
- Multimodel advantages and disadvantages
- Calibration process steps
 - Determine DoE points and cycle weights
 - DoE factors and bounds
 - Build and execute DoE test plan
 - Export multimodels and boundary models
 - Optimize calibration
 - Export and verify calibration





Motivation

- Current trends...
 - Emissions standards continue to tighten
 - Fuel consumption is becoming even more critical with rising fuel prices
 - Growing diesel engine customer base is demanding more refinement
- To meet these challenges technology is added to engines
 - Multiple injection strategies
 - Variable geometry turbochargers, series and parallel boost systems
 - Larger EGR coolers and cooler bypass valves
 - Variable swirl control
 - Variable valve events
 - Variable compression ratio
 - Etc...
- Calibration degrees of freedom continue to increase
 - **Model-based calibration has become absolutely necessary!**
- Margin to emissions standards continues to decrease
 - **Cycle-based emissions and fuel consumption optimization is required!**
 - Tradeoffs between various speed/load points
 - Optimum solution is not a collection of independent local optimizations
 - Requires global models at many speed/load points
- Model-Based Calibration Toolbox (MBC) offers tools to address these needs





Multimodel Advantages and Disadvantages

4

- Multimodels essentially switch between best local models at discrete speeds and loads
- Advantages
 - Reduces number of modeled inputs by two (speed and load)
 - Can select best model type for each response at each speed/load
 - Achieves maximum model fidelity at each speed and load
- Disadvantages
 - Compared to single-stage: Requires more data
 - Compared to conventional two-stage: Does not utilize data at other speeds and loads to determine trends
 - Cannot interpolate between speeds and loads





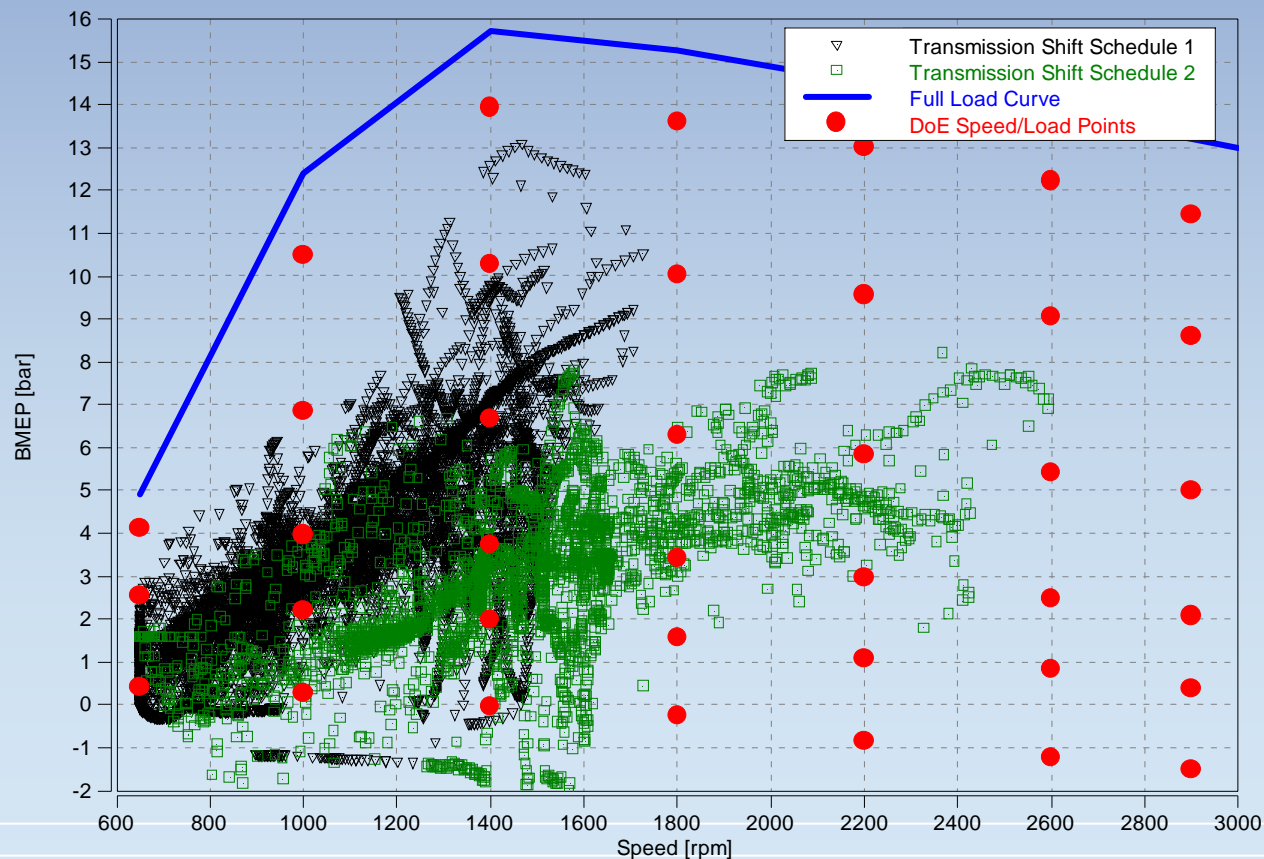
Calibration Process Steps

- Identify designed experiment (DoE) operating points
 - Select load axis
 - Cascade vehicle drive cycles to engine operating points
 - Select speed/load (S/L) break points for DoE
 - Calculate weighting factors for each S/L for each drive cycle
- Determine DoE factors, bounds, and resolution
 - Identify factors to be varied
 - Select ranges for each factor at each S/L
 - Build DoE runs based on desired model complexity
- Collect and prepare data for MBC
- Build multimodels for engine responses with boundary models
- Optimize calibration
 - Independent S/L optimization
 - Cycle-based optimization
- Export calibration and verify engine responses



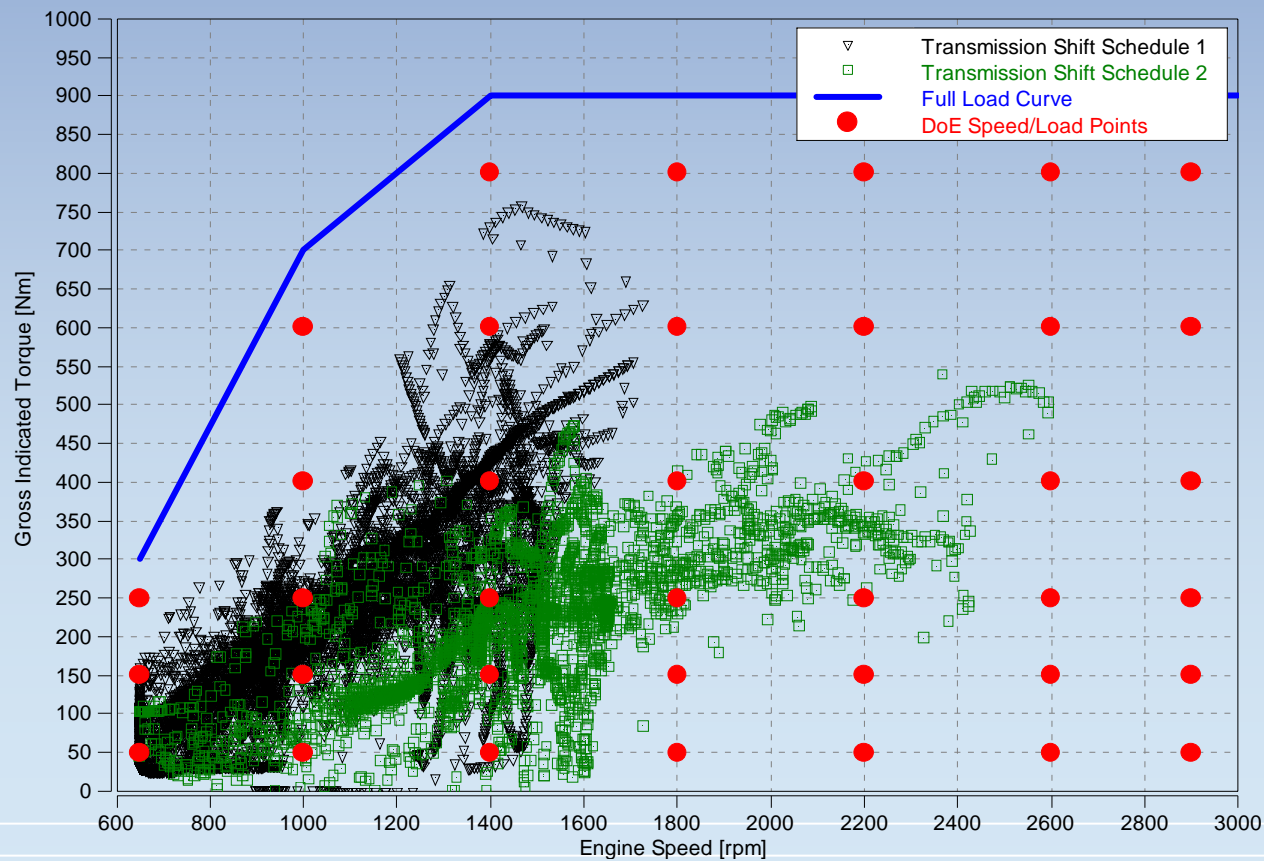
DoE Speed/Load Operating Points

- Cascade vehicle drive cycles to engine operating speed and load
- Requires either vehicle data or detailed vehicle models



DoE Speed/Load Operating Points

- Select load axis – brake torque, indicated torque, etc.
- Select DoE S/L operating points
- Calculate time weights





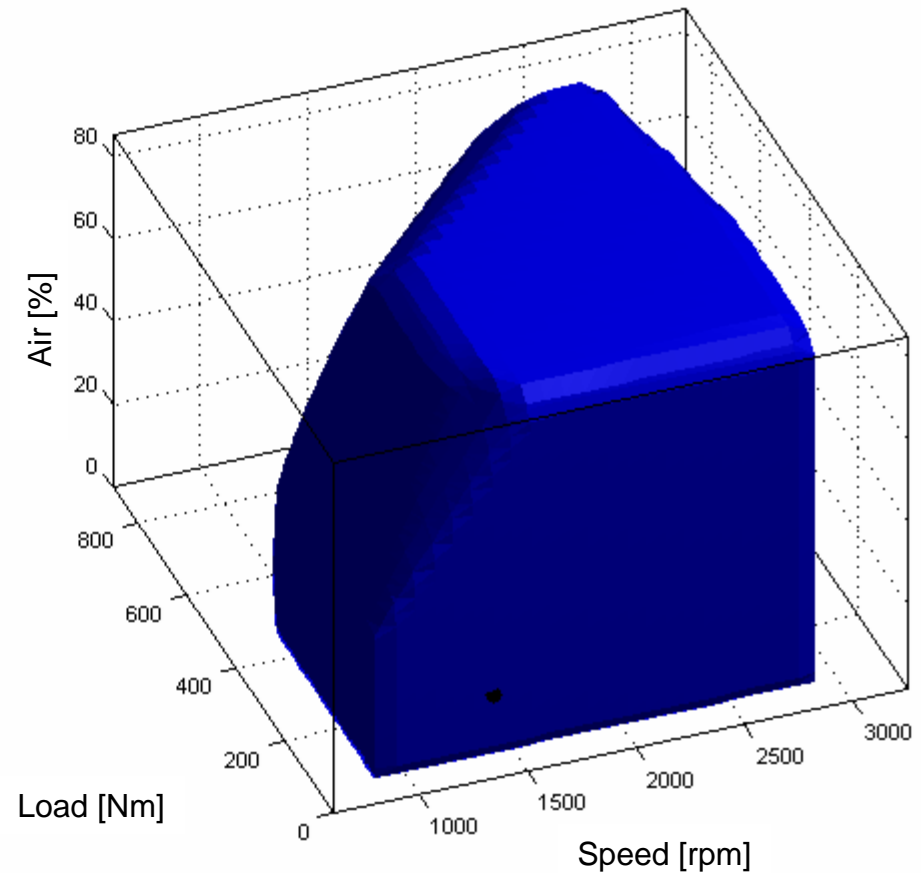
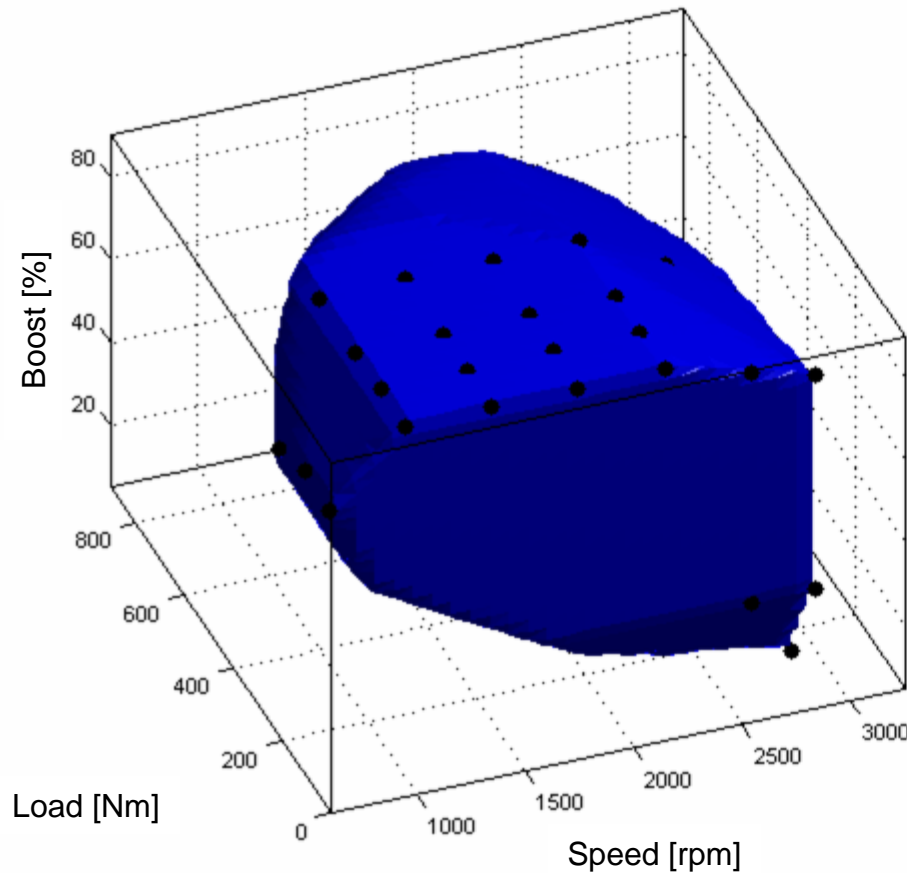
DoE Factors and Bounds

- Identify factors to be directly varied
 - Do not have to match actual calibration inputs
 - Must offer stable engine operation with minimal baseline calibration
 - Prefer measured rather than modeled feedback
 - For position control, combine multiple actuators into single scale
 - Vane position and waste gate position => “Boost system position”
 - EGR valve and throttle valve => “Air system position”
- Measure responses that could be used as inputs
 - EGR rate, intake oxygen percentage, etc.





DoE Factors and Bounds

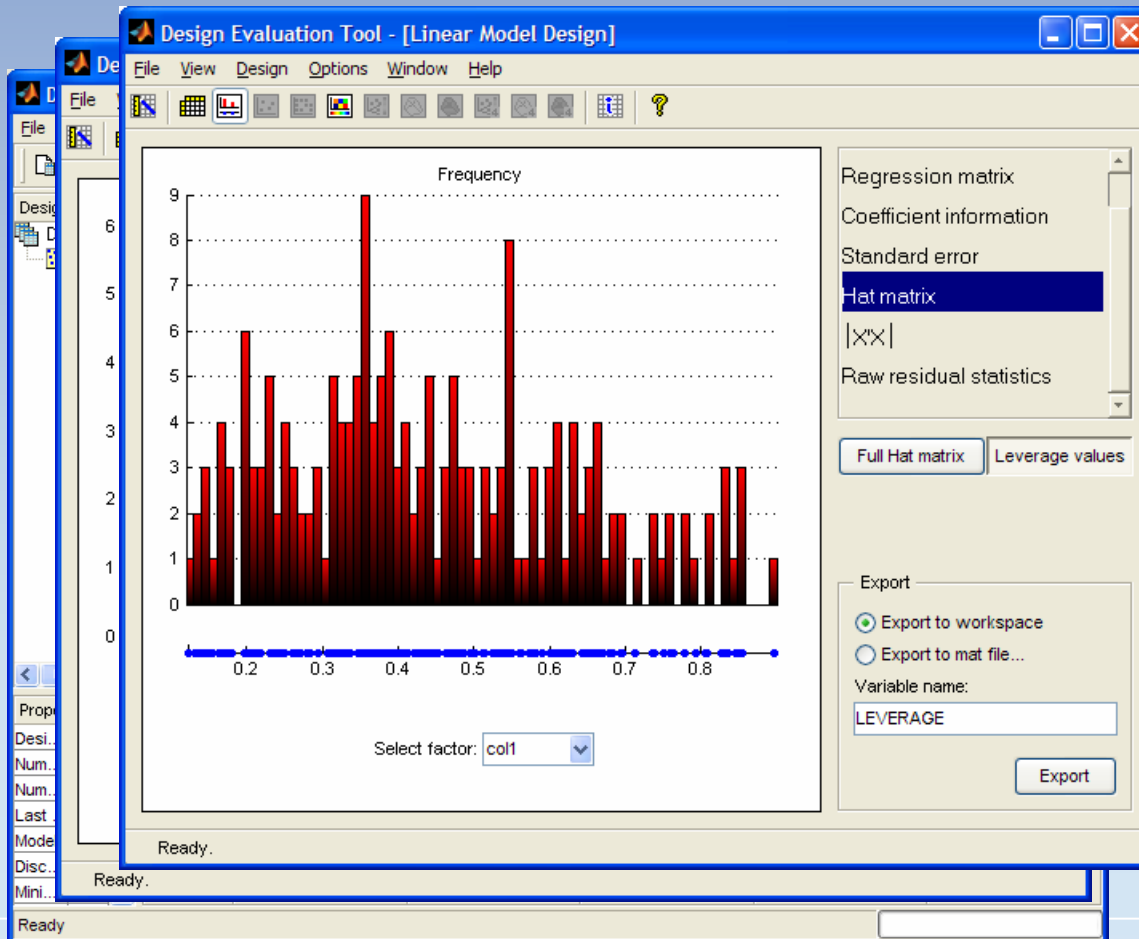


- Screening DoE for boost and air system limits
- Build boundary models for one-stage test plan
- Use simple CAGE optimization to find minimum and maximum for each speed and load



Build and Execute DoE Test Plan

- Generic DoE template (each factor 0 to 1)
- DoE for each speed/load based on bounds from screening



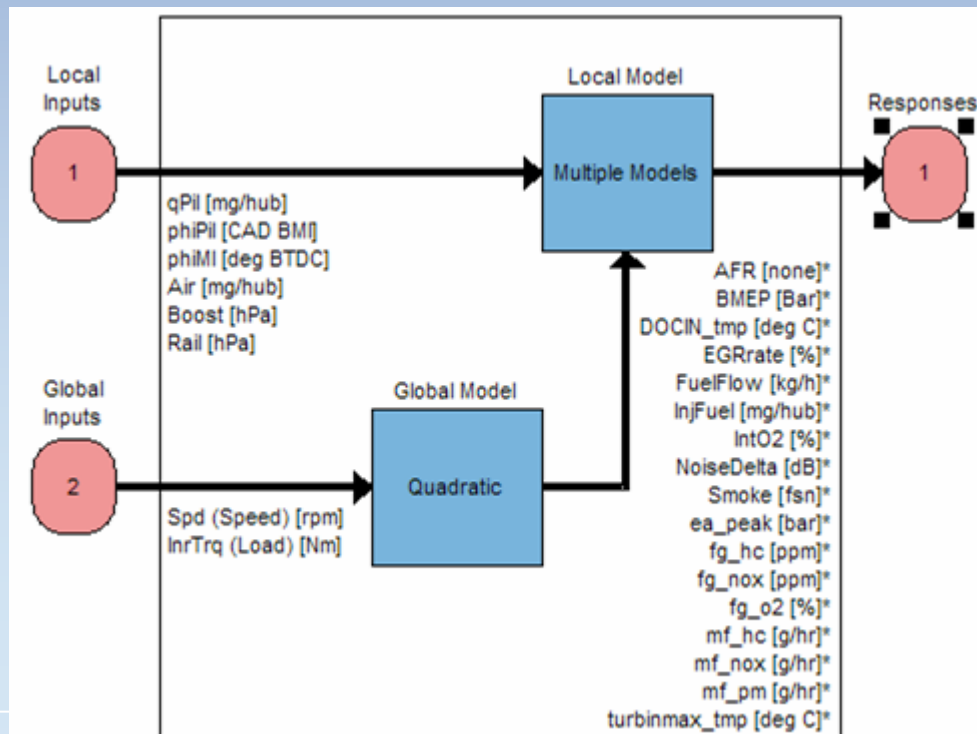
Example

- 190 point Halton sequence space-filling design for 6-input, 3rd order polynomial models
- Meets standard error coefficient guidelines of 0.7
- Meets Hat matrix leverage value guidelines of 0.9
- Real DoE should be augmented with additional ~25% of minimum point count (some points should fail if engine limits are being explored)



Build MBC Multimodels

- Import data into MBC
- Filter obvious outliers
- Build two-stage test plan with TWO inputs at the global level

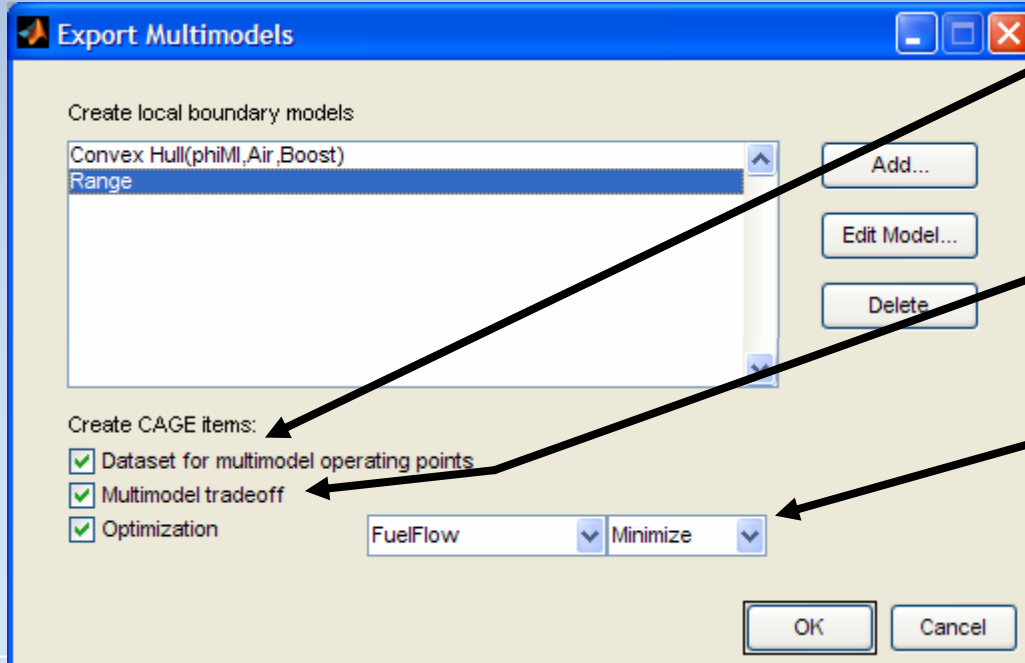


- Multiple models, i.e.
 - Third order polynomial
 - Radial basis function
 - Neural network
- Response models
 - Emissions
 - Fuel consumption
 - Critical engine limits
 - Optional control inputs
 - Combustion
 - Noise



Export Multimodels to CAGE

- New feature in R2008A
- Builds local boundary models
 - Range constraint
 - Convex hull with select critical parameters
 - Ability to preview boundary models is in development



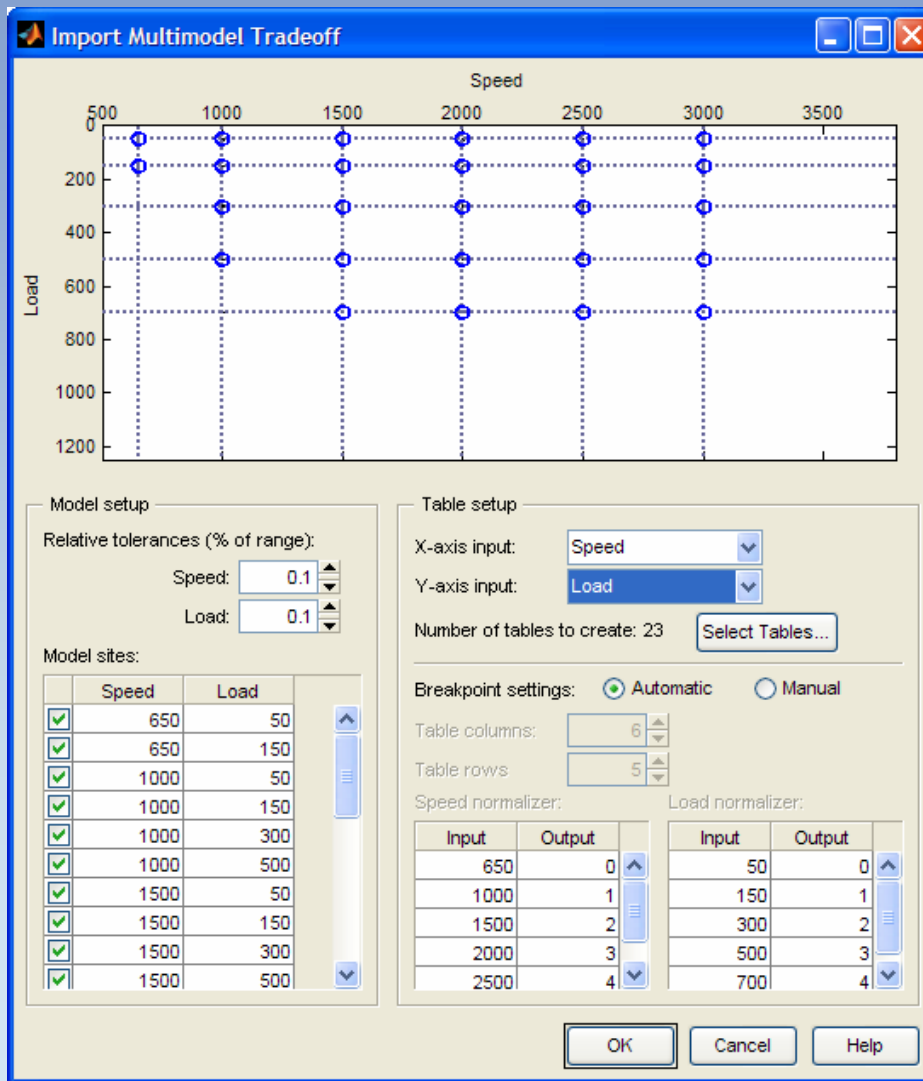
Creates CAGE dataset with average inputs for each speed and load

Creates CAGE tradeoff and builds calibration tables

Creates simple optimization with one objective and boundary models



CAGE Tradeoff Options










- Automatic table setup available
- Option to manually set up tables that match actual calibration break points
- Later, tables can be filled by optimization results





Optimization Step 1 – Independent Local Optimums

- Creates good starting point for cycle optimization
- Gives direction for off cycle speed/load points
- Objective
 - Minimize fuel flow
- Constraints
 - Boundary model
 - Basic emissions limits
 - Critical engine mechanical limits
 - Combustion noise

Objectives		
Name	Description	Type
 FuelFlow	FuelFlow(qPil, phiPil, phiMI, Air, Boost, Rail, Speed, Load)	Minimize
<div>< [] ></div>		
Constraints		
Name	Description	
 Fuel_Boundary	Boundary constraint of FuelFlow(qPil, phiPil, phiMI, Air, Boost, Rail, Speed, Load)	
 Smoke	Smoke(qPil, phiPil, phiMI, Air, Boost, Rail, Speed, Load) <= 2	
 Noise	-1.5 <= NoiseDelta(qPil, phiPil, phiMI, Air, Boost, Rail, Speed, Load) <= 1.5	
 HC	fg_hc(qPil, phiPil, phiMI, Air, Boost, Rail, Speed, Load) <= 300	
 PeakPres	ea_peak(qPil, phiPil, phiMI, Air, Boost, Rail, Speed, Load) <= 200	
 TurbineInlet	turbinmax_tmp(qPil, phiPil, phiMI, Air, Boost, Rail, Speed, Load) <= 750	





Optimization Step 2 – Cycle Optimization

15

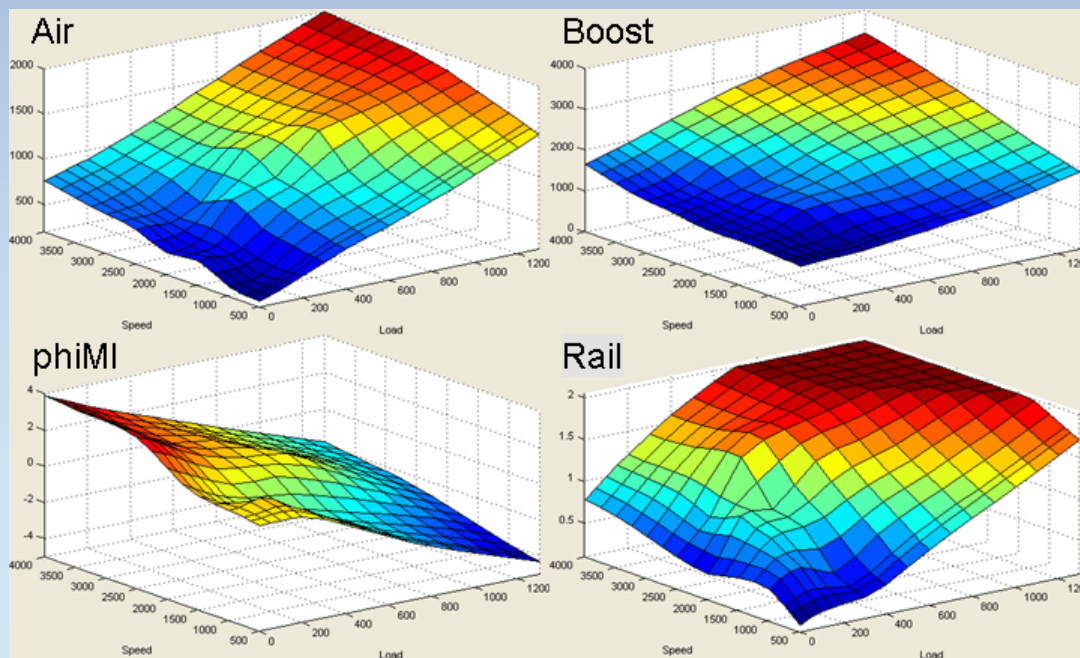
- Objective
 - Minimize cycle-weighted fuel flow
- Constraints
 - Boundary model
 - Basic emissions limits
 - Critical engine mechanical limits
 - Cycle-weighted emissions limits
 - Map gradients to ensure smooth calibration tables
 - Combustion noise

Objectives		
Name	Description	Type
FuelFlow	Weighted sum of FuelFlow(qPil, phiPil, phiMI, Air, Boost, ...	Minimize
< >		
Constraints		
Name	Description	
Fuel_Boundary	Boundary constraint of FuelFlow(qPil, phiPil, phiMI, Air, Boost, Rail, Spee...	
Smoke	Smoke(qPil, phiPil, phiMI, Air, Boost, Rail, Speed, Load) <= 2	
Noise	-1.5 <= NoiseDelta(qPil, phiPil, phiMI, Air, Boost, Rail, Speed, Load) <= 1.5	
HC	fg_hc(qPil, phiPil, phiMI, Air, Boost, Rail, Speed, Load) <= 300	
PeakPres	ea_peak(qPil, phiPil, phiMI, Air, Boost, Rail, Speed, Load) <= 200	
TurbineInlet	turbinmax_tmp(qPil, phiPil, phiMI, Air, Boost, Rail, Speed, Load) <= 750	
CycleNOx	Weighted sum of mf_nox(qPil, phiPil, phiMI, Air, Boost, Rail, Speed, Load)...	
CycleHC	Weighted sum of mf_hc(qPil, phiPil, phiMI, Air, Boost, Rail, Speed, Load) ...	
CyclePM	Weighted sum of mf_pm(qPil, phiPil, phiMI, Air, Boost, Rail, Speed, Load) ...	
Grad_phiMI	Gradient constraint of phiMI over (Load,Speed)	



Fill Calibration Tables

- Cycle optimization results can be used to fill calibration tables by interpolation and extrapolation
- Extrapolated results may violate engine limits



- Tables can be exported to calibration tool
- Engine must be mapped to verify results





Summary

- Multimodels are useful for complex engine calibrations
 - Select best local models
 - Allow use of cycle objectives and constraints
 - Allow use of gradient constraints
- Usability greatly increased in R2008A
 - Boundary model visualization is in development
- Engine verification is critical for interpolated and extrapolated points





Acknowledgements

- Ford base-engine calibrators
 - Brian Hallgren, Alex Wenzel, Brent Keppy, Lawrence Vaduva, Joe Acar, Ryan Johnson, Brian Baldwin, Bryan Van der Velde, and Gus Johnson
- Ford test cell automation and operation team
 - Rick Smith, Tom Beard, John Hancock, and Tim Gardner
- Peter Maloney at The MathWorks for facilitating communication with the MBC developers
- Thank you for your attention
- Questions?

