

Model Based Engine Control Development and Hardware-in-the-Loop Testing for the EcoCAR Advanced Vehicle Competition

2011-01-1297

Published
04/12/2011

Robert Cooley
Ohio State Univ.

Davide Vezza
Politecnico di Torino

Shawn Midlam-Mohler and Giorgio Rizzoni
Ohio State Univ

Copyright © 2011 SAE International

doi:[10.4271/2011-01-1297](https://doi.org/10.4271/2011-01-1297)

ABSTRACT

When developing a new engine control strategy, some of the important issues are cost, resource minimization, and quality improvement. This paper outlines how a model based approach was used to develop an engine control strategy for an Extended Range Electric Vehicle (EREV). The outlined approach allowed the development team to minimize the required number of experiments and to complete much of the control development and calibration before implementing the control strategy in the vehicle. It will be shown how models of different fidelity, from map-based models, to mean value models, to 1-D gas dynamics models were generated and used to develop the engine control system. The application of real time capable models for Hardware-in-the-Loop testing will also be shown.

INTRODUCTION

Current literature has shown that a wide variety of engine simulation models have been developed and applied to many different engine research problems [1,2,3,4,5]. These models vary greatly in their fidelity and computational efficiency [1]. The methods which are used to build and calibrate the different models require varying amounts of empirical data and model calibration effort [6]. With a large number of model based resources available, a key question is how to

select and implement models which are best targeted for a project's goals.

This question was addressed by the Ohio State University EcoCAR Challenge team. The EcoCAR Challenge is a collegiate automotive design competition primarily sponsored by General Motors and the U.S. Department of Energy. The competition challenges 16 North American Universities to redesign a GM donated vehicle for increased fuel economy and decreased emissions while maintaining or improving the performance and consumer acceptability of the vehicle.

The Ohio State team designed a range-extending electric vehicle capable of 50 miles all-electric range via a 22 kWh lithium-ion battery pack, with range extension and limited parallel operation supplied by a 1.8 L dedicated E85 engine. This vehicle is designed to drastically reduce fuel consumption, with an estimated fuel economy of 89 miles per gallon gasoline equivalent (mpgge) while meeting Tier II Bin 5 emissions standards.

To extend the range of the vehicle's electric architecture, the team converted a compressed natural gas (CNG) engine to operate with E85 fuel. As compared to typical gasoline engines, this dedicated E85 engine sees a dramatic improvement in brake efficiency because of its increased compression ratio, 12.5:1 [7]. More information about the competition and vehicle design can be found in [8]. A more

detailed description of the work presented in this paper can be found in [9].

The conversion of the engine from CNG to E85 required the complete development and calibration of a new engine control system. With the aim of reducing cost, minimizing required resources, and maximizing the quality of the control system, the team utilized model-based development tools to create a family of models for software-in-the-loop (SIL) testing, hardware-in-the-loop (HIL) testing, and control algorithm development.

Similar to industry, one of the key motivators for the EcoCAR team is time. The competition's compressed vehicle development process, illustrated in Figure 1, necessitates engine data prior to final hardware and calibrations being available. The Ohio State University (OSU) team has relied on a variety of engine models to fill in this gap and accelerate the development process. Engine simulation models and model based development allowed the team to move development early in the design cycle and minimize experimentation and calibration effort. The use of models allowed the team to conduct development tasks in parallel rather than serially which greatly streamlined the process.

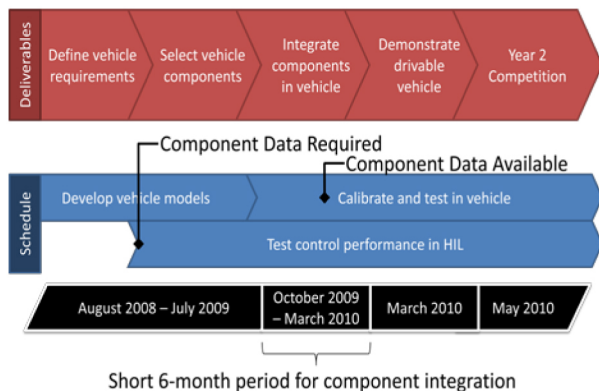


Figure 1. Competition Timeline

MODELING PROCESS

Before the team began to develop models and perform experiments on the engine, a modeling process was defined. By defining a process up-front, the team was not only able to understand the task better but was also able to understand what needed to be done in each stage of development. The team first looked at how models had been developed in previous iterations of the collegiate vehicle design competition. This process can be seen in Figure 2.

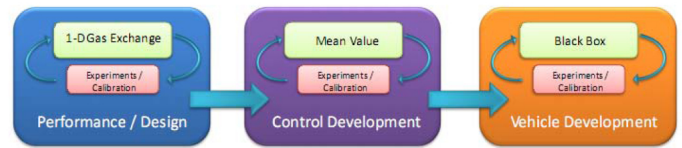


Figure 2. Linear Modeling Process

This linear process of model development allows different design groups to develop different models for different purposes. Each group works separately to develop the experimental requirements for each model's calibration. Also, each group can pass information to the others about how changes to the individual models may affect the others. This process does allow for some level of parallel model development; however, the team found that the process is too segmented for a small collegiate design group. Also, since the models are developed with independent experimentation, the team recognized that this type of process may lead to unnecessary amounts of repeated experimentation.

A more optimal process was found by moving away from a linear process and looking for a circular process. The process that the team decided to use is illustrated in Figure 3. This method works by dividing development into two processes, modeling and experimentation. There are also two loops in the process. An external loop connects the modeling process to the experimentation process. An internal loop connects the data generated by the different model types to one another.

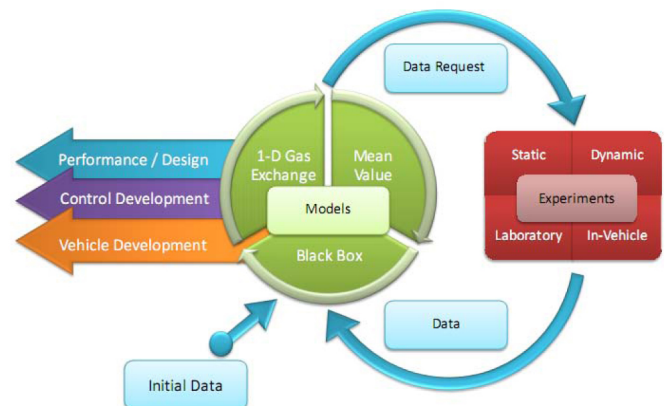


Figure 3. Circular Modeling Process

The key to this process is that all of the models are developed at the same time with the same information. For instance, the team enters the process with some amount of data about the system and then follows the progression in Figure 4 as many times as necessary.

1. Use known data to develop models
2. Use data generated by high fidelity models to fill in gaps in data coverage for lower fidelity models
3. Perform simulations to understand which portions of the models need improvement
4. Generate a request for more data and send to the experimentation process
5. Pass generated data from the experimentation process to the modeling process

Figure 4. Steps in Circular Modeling Process

Following this strategy allowed the different groups in the team to begin working at the same time and for all of the models to develop simultaneously. Also, sharing generated data in the modeling loop allowed the team to minimize the number of times that it was necessary to use the external experimentation loop. A more detailed diagram of the team's specific sequence of model development is shown in [Figure 5](#).

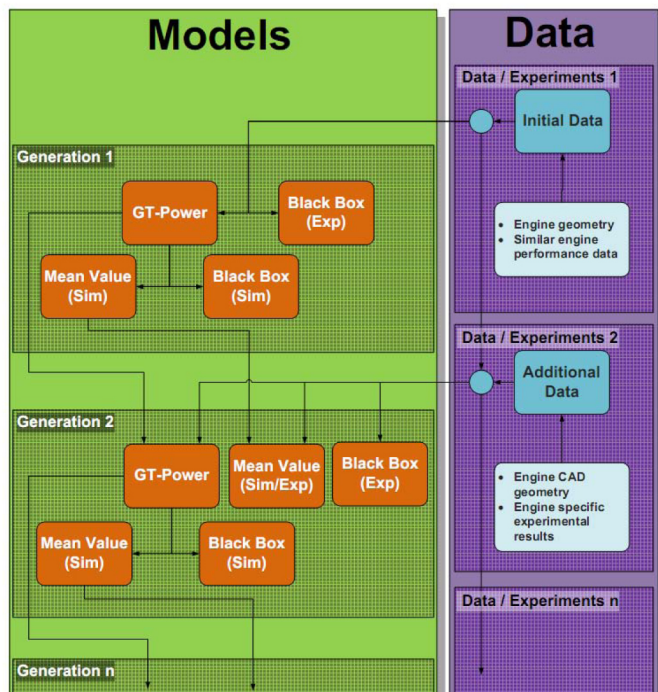


Figure 5. OSU Model Development Process

As shown in [Figure 5](#) the team started with a limited amount of data about the engine geometry and experimental performance characteristics of similarly sized engines. This information was used to generate both a GT-POWER simulation as well as a black-box model. The GT-POWER simulation was then used to generate simulation based version of both a mean-value and black-box model. This largely simulation-based model, although not well-validated,

was useful in other vehicle development tasks and was available long before the engine was available for experimentation. The second round of data collection and experimentation inserted more detailed engine geometry data as well as a limited number of preliminary experimental data points. This was used to increase the accuracy of the previous models and spawn a new generation engine models. The following sections will describe more in depth some of the model development and model results with different types of models.

BLACK BOX MODELS

The black-box engine models developed during year one and two of the EcoCAR competition are essentially lookup tables. Based on certain input conditions, the models lookup different output states from a static table. The lookup values are derived from steady state operation values which can be determined experimentally or estimated from similar engines or higher fidelity engine simulations. Based on certain engine operating speeds and torques, the associated efficiency and/or fuel consumption can be determined directly from an efficiency/fuel consumption surface.

The primary use of these models is to estimate fuel consumption in a vehicle simulator. Integrating the black-box engine model in to the OSU team's EcoSIM vehicle simulator allows time based fuel consumption to be estimated with engine torque demand and engine speed provided as inputs to the model. Engine speed is determined externally from this model because the vehicle architecture allows the engine to be coupled in series to a front electric motor or in parallel with the front electric motor and the road through a single speed transmission. The engine torque demand is determined by the vehicles supervisory control strategy. This strategy takes into consideration conditions like driver torque request and battery SOC to determine the optimal engine torque request.

The black-box model structure is fairly simple and is best explained by [Figure 6](#). With torque command as an input, the dynamics of torque request to production are simulated as a torque filter and torque saturation block. This lumps in the dynamics of throttle positioning, manifold air dynamics, *etc.* into a filter. The torque saturation block ensures that the torque of the engine does not exceed the maximum and minimum torque limits. The model then outputs an engine achieved torque which is translated to the other components of the vehicle. The fuel consumption of the engine can then be looked up based on the engine achieved torque and the imposed engine speed.