Performance Comparison of Hybrid Vehicle Energy Management Controllers on Real-World Drive Cycle Data

Daniel F. Opila, Xiaoyong Wang, Ryan McGee, Jeffrey A. Cook, and J.W. Grizzle

Abstract— **Hybrid Vehicle fuel economy and drivability performance are very sensitive to the "Energy Management" controller that regulates power flow among the various energy sources and sinks. Many methods have been proposed for designing such controllers. Most analytical studies evaluate closed-loop performance on government test cycles. Moreover, there are few results that compare stochastic optimal control algorithms to the controllers employed in today's production hybrids. This paper studies controllers designed using Shortest Path Stochastic Dynamic Programming (SPSDP). The controllers are evaluated on Ford Motor Company's highly accurate proprietary vehicle model over large numbers of real-world drive cycles, and compared to a controller developed by Ford for a prototype vehicle. Results show the SPSDP-based controllers yield 2-3% better performance than the Ford controller on real-world driving data, with even more improvement on a government test cycle. In addition, the SPSDP-based controllers can directly quantify tradeoffs between fuel economy and drivability.**

I. Introduction

Hybrid vehicles have become increasingly popular in the automotive marketplace in the past decade. The most common type is the electric hybrid, which consists of an internal combustion engine (ICE), a battery, and at least one electric machine (EM). Hybrids are built in several configurations including series, parallel, and the series-parallel configuration considered here. Hybrid vehicles are characterized by multiple energy sources; the strategy to control the energy flow among these multiple sources is termed "Energy Management" and is crucial for good fuel economy. An excellent overview of this area is available in [13].

This energy management problem has been studied extensively in academic circles [7], [9], [11], [13]. There are many proposed methods available for both the non-causal (cycle known in advance) and causal (cycle unknown in advance) cases. It is rather unclear how much of this work is used by industry in actual production vehicles. Many papers show simulations on representative vehicles, although most of this work focuses on the certification test cycles. There are relatively few results showing how these algorithms perform in practice [2], [3], [8], [11] and how they compare to the existing industrial state of the art.

The controller design method studied here is Shortest Path Stochastic Dynamic Programming (SPSDP), which has been used in this application several times before [4], [6], [14]. Recent results [10] have developed this method so that real-world constraints on drivability and powertrain behavior can be incorporated. In addition to generating a class of optimal controllers, this method allows direct study of the tradeoffs between different performance goals, specifically drivability and fuel economy. The ability to easily generate Pareto tradeoff curves is perhaps just as interesting as a specific fuel economy benefit.

We believe that the SPSDP energy management formulation of [10] is the first model-based optimal energy-management strategy that can be used directly in a production vehicle with minimal manual tuning. To validate this hypothesis, real-world driving data is used to evaluate controller performance for typical drivers. A common customer complaint is that the fuel economy shown on the "window sticker" does not match the vehicle performance in practice. By using large numbers of real-world drive cycles, controller performance can be evaluated and optimized with respect to both average drivers and government certification.

The purpose of this paper is to study energy management controllers in a real-world scenario and compare them to an industrially-designed controller. It is hoped that these results can verify the usefulness of this algorithm and take these methods from academic research papers into industrial labs and onto the road.

This research is a collaborative effort of the University of Michigan and Ford Motor Company. This partnership allows broad access to proprietary in-house tools, albeit with some restrictions on the level of detail that can be published. This work uses Ford's high-fidelity vehicle simulation model [1], which is used to develop HEV control algorithms and evaluate fuel economy for production vehicles. The vehicle studied here is a prototype and does not match any vehicle currently on the market. As a benchmark, Ford supplied an energy management controller developed for this prototype.

This paper builds on previous work [10] and focuses on real-world simulation studies and comparison to industrial controllers, rather than detailed algorithm descriptions. Brief descriptions of the methods used are included here, but for more detail see [10].

D.F. Opila, J.A. Cook, and J.W. Grizzle were supported by a grant from Ford Motor Company.

Daniel Opila is with the Dept. of Mechanical Engineering, University of Michigan and is supported by a NDSEG fellowship. dopila@umich.edu

Xiaoyong Wang and Ryan McGee are with Ford Motor Company, Dearborn, MI.

Jeffrey Cook and Jessy Grizzle are with the Dept. of Electrical Engineering and Computer Science, University of Michigan. {jeffcook,grizzle}@umich.edu

This material is based upon work supported under a National Science Foundation Graduate Research Fellowship.

Special thanks to the University of Michigan Transportation Research Institute (UMTRI) for providing drive cycle data.

II. Vehicle

A. Vehicle Architecture

The vehicle model studied in this paper is a prototype series-parallel electric hybrid. A 2.4 L diesel engine is coupled to the front axle through a clutched 6-speed automated manual transmission. An electric machine, $EM1$, is directly coupled to the engine crankshaft, and can generate power regardless of clutch state. A second electric machine, $EM2$, is directly coupled to the rear axle through a fixed gear ratio without a clutch, therefore the electric machine is always rotating at a speed proportional to vehicle speed. Energy is stored in a 1.5 kWh battery pack. The system parameters are listed in Table I.

TABLE I
Vehicle Parameters

Engine Displacement	2.4 L
Max Engine Power	120 kW
Electric Machine Power $EM1$ (Front)	15 kW
Electric Machine Power $EM2$ (Rear)	35 kW
Battery Capacity	1.5 kWh
Battery Power Limit	34 kW
Vehicle Mass	1895 kg

B. Vehicle Models

The work presented in this paper uses two different dynamic models to represent the same prototype hybrid vehicle. The first model is quite simple; it has a sample time of 1s, uses lookup tables, and has very few states. It is used primarily to design the controller and do the optimization, and is called the "control-oriented" model.

The second model comes from Ford Motor Company and uses its in-house modeling architecture. This sophisticated model is used to test fuel economy and controller behavior by simulating controllers on drive cycles. This model is referred to as the "vehicle simulation" model in this paper [1].

This combination of models allows the controller to be designed on a simple model that keeps the problem feasible, while providing accurate fuel economy results on a complex model.

C. Control Model

When using Shortest-Path Stochastic Dynamic Programming, the off-line computation cost is very sensitive to the number of system states. For this reason, the model used to develop the controller must be as simple as possible. The vehicle model used here contains the minimum functionality required to model the vehicle behavior of interest on a second-by-second basis. Dynamics much faster than the sample time of 1s are ignored. Long-term transients that only weakly affect performance are also ignored; coolant temperature is one example.

The vehicle hardware allows three main operating conditions:

1) **Parallel Mode**-The engine is on and the clutch is engaged.
2) **Series Mode**-The engine is on and the clutch is disengaged. The only torque to the wheels is through $EM2$.
3) **Electric Mode**-The engine is off and the clutch is disengaged; again the only torque to the wheels is through $EM2$.

The model does not restrict the direction of power flow. The electric machines can be either motors or generators in all modes.

The dynamics of the internal combustion engine are ignored; it is assumed that the engine torque exactly matches valid commands and the fuel consumption is a function only of speed, ω_{ICE}, and torque, T_{ICE}. The fuel consumption F is derived from a lookup table based on dynamometer testing,

$$Fuel\,flow = F(\omega_{ICE}, T_{ICE}).$$

The automated manual transmission has discrete gears and no torque converter. The transmission is modeled with a constant mechanical efficiency of 0.95. Transmission gear shifts are allowed every time step (1s) and transmission dynamics are assumed negligible. When the clutch is engaged, the vehicle is in **Parallel Mode** and the engine speed is assumed directly proportional to wheel speed based on the current transmission gear ratio R_g,

$$\omega_{ICE} = R_g \omega_{wheel}.$$

The electric machine $EM1$ is directly coupled to the crankshaft, and thus rotates at the engine speed ω_{ICE},

$$\omega_{EM1} = \omega_{ICE}.$$

In **Parallel Mode**, the engine torque T_{ICE} and $EM1$ torque T_{EM1} transmitted to the wheel are assumed proportional to wheel torque based on the current gear ratio R_g and the transmission efficiency η_{trans}. The rear electric machine $EM2$ torque T_{EM2} transmitted to the wheel is proportional to the constant $EM2$ gear ratio R_{EM2} and rear differential efficiency η_{diff}. The total wheel torque T_{wheel} is thus the sum of the ICE and $EM1$ torques to the wheel $\eta_{Trans} R_g (T_{ICE} + T_{EM1})$ and the rear electric machine $EM2$ torque to the wheel $\eta_{diff} R_{EM2} T_{EM2}$,

$$\eta_{trans} R_g (T_{ICE} + T_{EM1}) + \eta_{diff} R_{EM2} T_{EM2} = T_{wheel}. \tag{1}$$

The clutch can be disengaged at any time, and power is delivered to the road through the rear electric machine $EM2$. This condition is treated as the "neutral" gear 0, which combines with the 6 standard gears for a total of 7 gear states. If the engine is on with the clutch disengaged, the vehicle is in **Series Mode**. The engine-$EM1$ combination acts as a generator and can operate at arbitrary torque and speed. The $EM1$ command is a speed rather than a torque in **Series Mode**. If the engine is off while the clutch is disengaged, the vehicle is in **Electric Mode**.

The battery system is similarly reduced to a table lookup form. The electrical dynamics due to the motor, battery, and power electronics are assumed sufficiently fast to be ignored.

TABLE II

VEHICLE MODE DEFINITIONS.

Gear State	Clutch State	Engine State	Mode
0	Disengaged	Off	Electric
0	Disengaged	On	Series
1-6	Engaged	On	Parallel
1-6	Engaged	Off	Undefined/not used

The energy losses in these components can be grouped together such that the change in battery State of Charge (SOC) is a function $\bar{\kappa}$ of Electric Machine speeds ω_{EM1} and ω_{EM2}, torque T_{EM1} and T_{EM2}, and battery SOC at the present time step,

$$SOC_{k+1} = \bar{\kappa}(SOC_k, \omega_{EM1}, \omega_{EM2}, T_{EM1}, T_{EM2}). \quad (2)$$

Assuming a known vehicle speed, the only state variable required for this vehicle model is the state of charge (SOC). Changes in battery performance due to temperature, age, and wear are ignored. During operation, the desired wheel torque is defined by the driver. If we assume the vehicle must meet the torque demand perfectly, then the sum of the ICE and EM contributions to wheel torque (1) must equal the demanded torque T_{demand},

$$T_{wheel} = T_{demand}.$$

This adds a constraint to the control optimization, reducing the 4 control inputs to a 3 degree of freedom problem. In **Parallel Mode** the control inputs are *Engine Torque*, $EM1$ *Torque*, and *Transmission Gear*. In **Series Mode**, the electric machine command becomes $EM1$ *Speed*.

Optimization using the control-oriented model assumes a "perfect" driver. The desired road power is calculated as the exact power required to drive the cycle at that time. Now, given vehicle speed, demanded road power and this choice of control inputs, the dynamics become an explicit function κ of the state *Battery SOC* and the three control choices as shown in Table III,

TABLE III

VEHICLE DYNAMIC MODEL

State	Control Inputs
Battery Charge (SOC)	Engine Torque
	$EM1$ Torque (Parallel) or Speed (Series)
	Transmission Gear

$$SOC_{k+1} = \kappa(SOC_k, T_{ICE}, T_{EM1}, Gear). \quad (3)$$

In **Series Mode**, T_{EM1} is replaced with ω_{EM1}. The engine fuel consumption can be calculated from the control inputs.

Operational Assumptions:

This control-oriented model uses several assumptions about the allowed vehicle behavior.

1) The clutch in the automated manual transmission allows the diesel engine to be decoupled from the wheels. This allows the engine to shut off during forward motion.
2) There is no ability to slip the clutch for starts.
3) There are no traction control restrictions on the amount of torque that can be applied to the wheels.
4) **Series Mode** is not used. The engine is off if the clutch is disengaged.

D. Vehicle Simulation Model

As part of this project, Ford provided an in-house model used to simulate fuel economy. It is a complex, MATLAB/Simulink based model with a large number of parameters and states [1]. Every individual subsystem in the vehicle is represented by an appropriate block. For each new vehicle, subsystems are combined appropriately to yield a complete system.

This vehicle simulation model contains the baseline controller algorithm. To generate simulation results using this controller, a target drive cycle is provided to the existing model with no modifications.

To use the vehicle simulation model with the algorithm developed here, the SPSDP controller is implemented in Simulink by interfacing appropriate feedback and command signals: Battery SOC, Vehicle Speed, Engine State, Gear Command, etc. The vehicle simulation model can then be "driven" by the SPSDP controller along a given drive cycle.

III. DRIVABILITY CONSTRAINTS

A. Motivation

Drivability is a rather vague term that covers many aspects of vehicle performance including acceleration, engine noise, braking, shifting activity, shift quality [12], and other behaviors. All of these contribute to consumer perception of the vehicle, which is crucial in purchasing decisions. This research addresses the "basic" drivability issues of gear selection and when to start or stop the internal combustion engine.

Current academic work in hybrid vehicle optimization primarily focuses on fuel economy. These tools are somewhat less useful to industry because of drivability restrictions in production vehicles, which fuel-optimal controllers usually violate. If these fuel-optimal controllers are used, drivability restrictions are typically imposed as a separate step.

In this paper we investigate the usefulness of optimizing for fuel economy and drivability simultaneously. By including these real-world concerns, one can generate controllers that improve performance and are one step closer to being directly implementable in production. Specifically, these results validate the real-world performance of the SPSDP algorithm and compare it to an industrial controller.

B. Chosen Penalties

In the context of the overall system, two significant characteristics that are noticeable to the driver are the basic behaviors of the transmission and engine. These are included in both vehicle models presented in Section II. To effectively design controllers, qualitative drivability requirements must be transformed into quantitative restrictions or metrics. Drivability experts at Ford Motor Company were consulted to assist in developing numerical drivability criteria. Two baseline metrics are used to quantify behavior for a particular trip. The first is *Gear Events*, the total number of shift events on a given trip. The second metric is *Engine Events*, the total number of engine start and stop events on a trip.

By definition, engine starts and stops are each counted as an event. Each shift is counted as a gear event, regardless of the change in gear number. A $1^{st} - 2^{nd}$ shift is the same as a $1^{st} - 3^{rd}$ shift. In this paper, the transmission is constrained to one step shifts (i.e. $1^{st} - 2^{nd}$) to match the transmission restrictions of the baseline controller. Gear shifts that occur while in neutral (clutch disengaged) are not counted. Engaging or disengaging the clutch is not counted as a gear event, regardless of the gear before or after the event.

Despite the relative simplicity of these metrics, simulations have shown that they capture a wide range of vehicle behavior and are well correlated with more complicated metrics.

IV. Shortest Path Stochastic Dynamic Programming

A. Cost Function

In order to design a controller with acceptable drivability characteristics, the optimization goal over a given trip of length T would ideally be defined as

$$\begin{gathered} \min \textstyle\sum_0^T Fuel\, flow \\ \text{such that} \\ \textstyle\sum_0^T GE \leq GE_{max} \,,\, \sum_0^T EE \leq EE_{max} \end{gathered} \tag{4}$$

where GE and EE are the number of Gear and Engine Events respectively, and GE_{max} and EE_{max} are the maximum allowable number of events on a cycle.

This constrained optimization incorporates the two major areas of concern: fuel economy and drivability. Constraints of this type cannot be incorporated in the Stochastic Dynamic Programming algorithm used here because the stochastic nature of the optimization cannot directly predict performance on a given cycle. Instead, the drivability events are included as penalties, and the weights are adjusted until the outcome is acceptable and meets the hard constraints.

Controllers based only on fuel economy and drivability completely drain the battery as they seek to minimize fuel. An additional cost is added to ensure that the vehicle is charge sustaining over the cycle. This SOC-based cost only occurs during the transition to key-off, so it is represented as a function $\phi_{SOC}(x)$ of the state x, which includes SOC. The performance index for a given drive cycle is

$$J = \sum_0^T Fuel\, flow + \alpha \sum_0^T GE + \beta \sum_0^T EE + \phi_{SOC}(x_T). \tag{5}$$

The search for the weighting factors α and β involves some trial and error, as the mapping from penalty to outcome is not known a priori. Note that setting α and β to zero means solving for optimal fuel economy, subject to a charge sustaining penalty.

Now, to implement the optimization goal of minimizing (5), a running cost function is prescribed as a function only of the state x and control input u at the current time

$$c_{full}(x, u) = F(x, u) + \alpha \mathbf{I}_{GE}(x, u) + \beta \mathbf{I}_{EE}(x, u) + \phi_{SOC}(x) \tag{6}$$

where the function $\mathbf{I}(x, u)$ is the indicator function and shows when a state and control combination produces a Gear Event or Engine Event. Fuel use is calculated by $F(x, u)$. The SOC-based cost $\phi_{SOC}(x)$ still applies only at key-off, when the systems transitions to the key-off absorbing state. Many other vehicle behaviors can be optimally controlled by adding appropriate functions of the form $\phi(x, u)$; a typical example is limiting SOC deviations during operation to reduce battery wear.

B. Problem Formulation

To determine the optimal control strategy for this vehicle, the Shortest Path Stochastic Dynamic Programming (SPSDP) algorithm is used [6], [14]. This method directly generates a causal controller; characteristics of the future driving behavior are specified via a Markov chain rather than exact future knowledge. The system model is formulated as

$$x_{k+1} = f(x_k, u_k, w_k),$$

where u_k is a particular control choice in the set of allowable controls U, x_k is the state, and w_k is a random variable arising from the unknown drive cycle. Given this formulation, the optimal cost $V^*(x)$ over an infinite horizon is a function of the state x and satisfies

$$V^*(x) = \min_{u \in U} E_w[c(x, u) + V^*(f(x, u, w))], \tag{7}$$

where $c(x, u)$ is the instantaneous cost as a function of state and control; (6) is a typical example. The optimal control u^* is any control that achieves the minimum cost $V^*(x)$. This equation represents a compromise between minimizing the current cost $c(x, u)$ and the expected future cost $V(f(x, u, w))$. Note that the cost $V(x)$ is a function of the state only. This cost is finite for all x if every point in the state space has a positive probability of eventually transitioning to an absorbing state that incurs zero cost from that time onward.

In order to use this method, the driver demand is modeled as a Markov chain. This “driver” is assigned two states: current velocity v_k and current acceleration a_k, which are included in the full system state x. A probability distribution is then assigned to the set of accelerations at the next time step. This means estimating the function

$$P(a_{k+1} | v_k, a_k) \tag{8}$$

for all states v_k, a_k. This Markov chain captures the uncertainty in the problem, which is represented in (7) by the random variable w. The specific realization of w determines a_{k+1} in (8),

$$a_{k+1} = g(v_k, a_k, w_k) \tag{9}$$

$$P(a_{k+1}|v_k, a_k) = P(w : g(v_k, a_k, w_k) = a_{k+1}). \tag{10}$$

The transition probabilities (8) are estimated from known drive cycles that represent typical behavior, dubbed the "design cycles." The function g represents system dynamics. The variables v_k, a_k, and a_{k+1} are discretized to form a grid. For each discrete state $[v_k,\ a_k]$ there are a variety of outcomes a_{k+1}. The probability of each outcome a_{k+1} is estimated based on its frequency of occurrence during the design cycle. See [14] for more detail.

In addition to fuel economy, it is desirable to study the drivability characteristics of the vehicle. The metrics chosen are gear shifts and engine events as described in Section III. To track these metrics, two additional states are required: the *Current Gear* (0-6) and *Engine State* (on or off).

Bringing this all together, the full system state vector x contains five states: one state for the vehicle (*Battery SOC*), two states for the stochastic driver (v_k, a_k), and two states to study drivability (*Current Gear* and *Engine State*). This formulation is termed the "SPSDP-Drivability" controller. A summary of system states is shown in Table IV. The control u contains the three inputs *Engine Torque*, $EM1$ *Torque/Speed*, and *Transmission Gear*, as described in Section II and Table III.

TABLE IV
VEHICLE MODEL STATES

State	Units
Battery Charge (SOC)	[0-1]
Vehicle Speed	m/s
Current Vehicle Acceleration	m/s^2
Current Transmission Gear	Integer 0-6
Current Engine State	On or Off

V. Drive Cycle Data

A major goal of this paper is to demonstrate the real-world potential of the proposed SPSDP algorithm. Controller performance is often demonstrated on standard test cycles (FTP, NEDC, US06) for comparison and relevance to government certification. A common complaint among customers is that the fuel economy on the window sticker does not match the actual fuel economy they get in practice. There are two potential reasons: either the controllers are tuned primarily for the test cycles, or real-world driving is fundamentally less fuel-efficient than the test cycles. The real-world data used in this paper allows a better evaluation of controller robustness and performance in the "off-cycle" real world.

The drive cycle data used in this paper was collected by the University of Michigan Transportation Research Institute (UMTRI) [5]. The "source" data set supplied to us contains 2500 trips made by 87 drivers. Very short trips (less than 3 minutes or 0.5 km) are ignored. We randomly selected two sets of 100 drive cycles from this group. They are called "Ensemble 1" and "Ensemble 2."

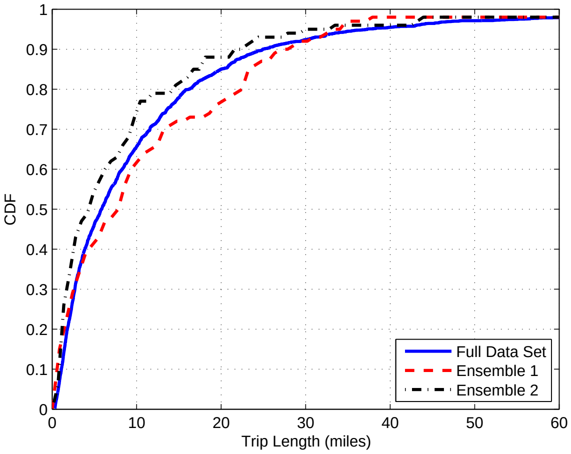

Fig. 1. Cumulative Distribution Function of Trip Distance for Source Data and Two Subsets. Mean Distances for the sets are: Full Data Set - 11.7 mi., *Ensemble 1* - 12.7 mi., *Ensemble 2* - 9.9 mi.

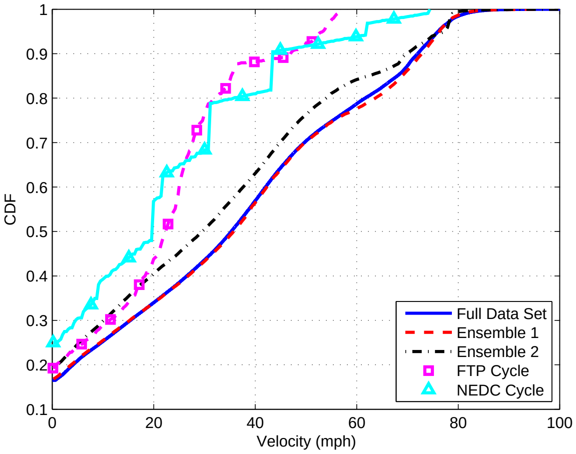

Fig. 2. Cumulative Distribution Function of Vehicle Velocity for Source Data, Two Subsets, and Two Government Test Cycles.

To gain some insight into the statistical nature of drive cycles, we briefly study the characteristics of the drive cycle distributions. The cumulative distribution functions (CDF) of trip distance for the source data and both subsets are shown in Figure 1. The statistics for the two ensemble sets are a reasonable match for the source data set.

A second statistic is the CDF of vehicle speed, as shown in Figure 2. This figure is computed by sampling vehicle velocity every second for the appropriate sets of trips and taking the CDF of the distribution. Two standard government test cycles are also shown, the Federal Test Procedure (FTP) and the New European Drive Cycle (NEDC). This yields five total curves in the picture: the Source Data, *Ensemble 1*, *Ensemble 2*, FTP, and NEDC.

There are three interesting things to notice in this figure. The first is that the government test cycles seem fundamentally different from the real-world data. The real-world cycles have substantially higher velocities in general. The second detail is the step-like nature of the NEDC cycle, which arises because it is completely contrived. The cycle is composed of perfect ramps to constant speeds and is obviously specified by hand. Lastly, *Ensemble 2* has lower velocities than *Ensemble 1*, which affects the fuel economy results presented in Section VII.

VI. Simulation Procedure

To study the effectiveness of this controller design methodology, a large number of controllers are simulated on a set of real-world driving data as discussed in Section V. Procedurally, this is conducted as follows:

1) A "family" of controllers is designed according to the methods of Section IV. A *family* is generated by fixing the model driving statistics and most parameters, and sweeping the 2 drivability penalties.
2) For each controller in the *family*, the controller is simulated on each of the 100 cycles in a particular ensemble using the vehicle simulation model.
3) The results for each ensemble set of 100 cycles are compiled to generate average or cumulative performance for that particular controller.

In the end result, each *family* of controllers contains a few hundred individual controllers which have each been simulated on 100 ensemble cycles. Each controller has average performance metrics (fuel economy and drivability) representing cumulative performance on the set of ensemble cycles. Note that studying 100 controllers on 100 cycles each means 10,000 simulations.

For additional comparison, controllers are simulated on a government test cycle, in which case there is only one simulation per controller.

Several results are presented which compare the SPSDP-based controllers to a baseline controller developed by Ford for this prototype vehicle. For proprietary reasons, all fuel economy numbers are normalized to the baseline Ford controller running the FTP cycle. Both controller design methods (Ford and SPSDP) use the same vehicle simulation model.

These simulations are all causal, so the final SOC is not guaranteed to exactly match the starting SOC. This could yield false fuel economy results, so all fuel economy results are corrected based on the final SOC of the drive cycle. This is done by estimating the additional fuel required to charge the battery to its initial SOC, or the potential fuel savings shown by a final SOC that is higher than the starting level. This correction is applied according to

$$\Delta Fuel = C_{Batt} \Delta SOC \frac{BSFC_{min}}{\eta_{max}^{Regen}} \tag{11}$$

where $\Delta Fuel$ is the adjustment to the fuel used, C_{Batt} is the battery capacity, ΔSOC is the difference between the starting and ending SOC, $BSFC_{min}$ is the best Brake Specific Fuel Consumption for the engine, and η_{max}^{Regen} is the best charging efficiency of the electric system.

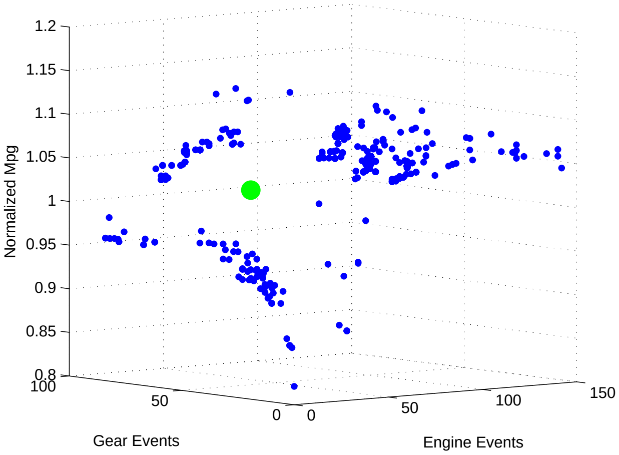

Fig. 3. Typical Simulation Results on FTP of a *family* of controllers on a 3-D scatterplot. The SPSDP controller *family* is shown by small blue dots. The Ford controller is shown as a large green circle. Fuel economy is normalized to the Ford controller on FTP for all figures. A response surface is fitted to raw data like these to generate isoclines of constant gear as in Figures 4-7.

VII. Main Results

The main goal of this research is to use the SPSDP method to tradeoff fuel economy and drivability requirements by using a class of optimal controllers, and validate the result against industrial design methods. The three metrics of interest during vehicle driving are the number of *Gear Events*, *Engine Events*, and the total fuel consumption corrected for SOC. These metrics yield conflicting goals and there is a distinct tradeoff among them. To study this tradeoff, several hundred controllers are designed with varying penalties assigned to each *Gear Event* and *Engine Event*. This creates one *family* of controllers as described in Section VI.

After simulation, the resulting data show the tradeoff between fuel economy and drivability. The typical result is a 3-D scatterplot of one *family* of controllers as shown in Figure 3. Each point represents a single controller driven on the cycle in question, FTP in this case. For figures like this one, the controllers are all driven on the same test cycle. As mentioned in Section VI, these points could also represent the average performance on a group of cycles. The combination of these points form a surface in 3-D space that shows the tradeoff surface for various operating conditions. This particular figure shows a *family* of controllers designed using FTP statistics running the FTP cycle.

These 3-D plots are difficult to interpret in a single figure, so the shape of the 3-D surface is presented as lines on a 2-D plot. A response surface is fit to the raw data and used to generate isoclines of constant gear as shown in Figure 4. This plot shows the performance on the *Ensemble 1* set for one *family* of SPSDP controllers designed on that same cycle set. Each line in the plot represents a constant number of Gear Events, while the horizontal and vertical axes show the number of Engine Events and normalized fuel economy respectively. The curves represent a large number of possible

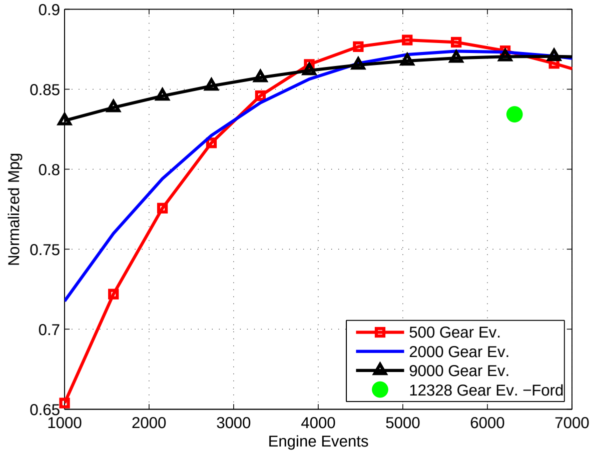

Fig. 4. Fuel Economy and Drivability Metrics shown as isoclines of constant gear on the *Ensemble 1* cycle set. The SPSDP controller *family* is designed on the *Ensemble 1* set. This figure is generated by fitting a response surface to raw data like that shown in Figure 3.

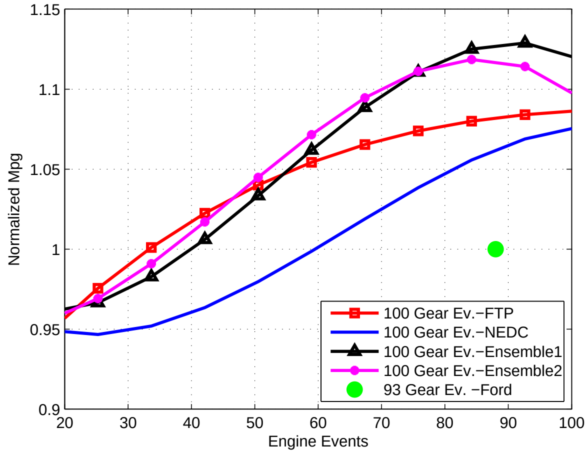

Fig. 5. Fuel Economy and Drivability Metrics on the FTP Cycle for 5 controller options. Controller *familes* are designed with statistics from FTP, NEDC, *Ensemble 1*, and *Ensemble 2*. All fuel economy figures are normalized to the Ford Controller performance on FTP, shown as a large green dot. The controllers are the same as those shown in Figures 6 and 7.

controllers, each with different penalty weights. The Ford controller performance on *Ensemble 1* set is still shown as a large green circle.

Figure 4 shows a distinct "knee" in the curves at 4000 Engine Events in where the fuel economy flattens out at its maximum. The fuel economy tradeoff is not as severe for larger numbers of gear events. The remainder of the figures in this paper compare several *families* of controllers, so only one isocline is drawn for each *family*. Each line still represents a portion of a 3-D surface like the one in Figure 3.

Simulations are conducted on FTP again in Figure 5 for SPSDP controller *families* designed on four different design cycles. This figure fixes the desired number of gear events at 100, where the tradeoff between fuel economy and engine events is not as severe. The controllers based on FTP, *Ensemble 1* and *Ensemble 2* show similar performance while the NEDC-based controllers show a noticeable difference.

To study performance in the real world, the controllers are tested on the set of ensemble cycles. The fuel economy for the ensemble cycle sets is calculated using the ratio of the total fuel used on all cycles and the total distance (sum of all 100 cycles). The fuel use is corrected for final SOC for each individual cycle, before the summation to yield total fuel. This result approximates the average consumer fuel economy over about 1000 miles, or 3 tanks of gas.

Figure 6 shows 5 different controller options running the *Ensemble 1* set. The controllers are the same as those in Figure 5, just running different cycles. As expected, the controllers based on the ensemble statistics yield the best performance. The same controller options are simulated on the *Ensemble 2* set as shown in Figure 7. The relatively slower driving of *Ensemble 2* as shown in Figure 2 yields slightly improved fuel economy compared to *Ensemble 1*.

In general, the SPSDP design methods are quite robust to drive cycle variation. They consistently beat an industrial controller both on a government test cycle and real-world cycles. The FTP and ensemble cycles show significant differences both in terms of fuel economy and the relative performance of different controllers, confirming the statistical differences noted in Figure 2. Regardless of the design statistics used, real-world driving does not approach the fuel economy on FTP, which has a normalized fuel economy of 1.0.

The Ford controller uses 12,328 Gear Events on the *Ensemble 1* set. These results depend on the exact definitions used for gear and engine events. In this work, any gear shifts that occur while the clutch is disengaged (including engine start and shutdown) are not counted as an event and incur no cost. The SPSDP-based controllers have this definition in mind when they are designed, but the Ford controller does not. The Ford controller uses a shifting strategy that does not directly account for the metrics used here.

VIII. Conclusions

The energy management controller for a hybrid vehicle is a major factor in the vehicle's overall performance. This paper studies controllers generated using Shortest Path Stochastic Dynamic Programming (SPSDP) and evaluates their performance and robustness on real-world drive cycles using a highly accurate simulation model. The SPSDP-based controllers use a statistical description of expected driving behavior to minimize a cost function that is a weighted sum of consumed fuel and drivability penalties, such as shift events and engine on-off events. By varying the weights, a control designer can systematically trade off fuel economy and drivability. These tradeoffs are optimal for given driving statistics. The performance of the SPSDP-based controllers was compared against an industrial-quality controller provided by Ford Motor Company that was designed by a team of engineers over several years.

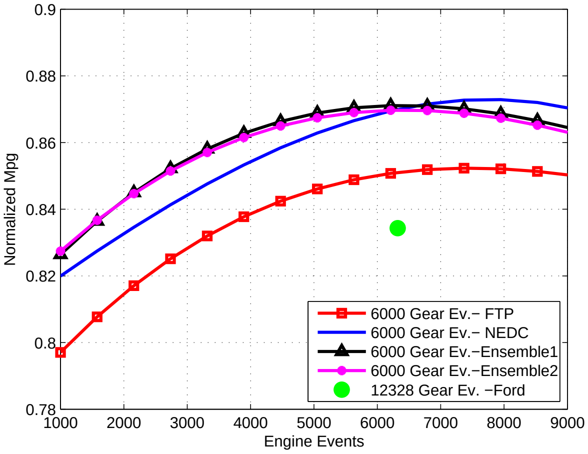

Fig. 6. Fuel Economy and Drivability Metrics on the *Ensemble 1* set for the Ford Controller and controller *familes* are designed with statistics from FTP, NEDC, *Ensemble 1*, and *Ensemble 2*. Fuel Economy, Gear Events, and Engine Events are cumulative for the whole cycle set, approximately 1000 miles. The controllers are the same as those shown in Figures 5 and 7. Results normalized to the Ford Controller on FTP (Fig. 5).

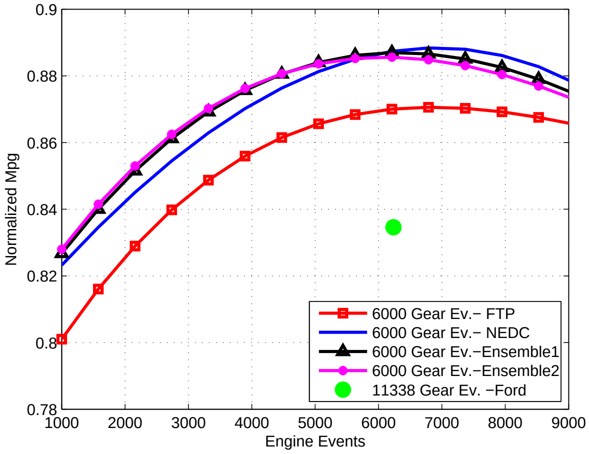

Fig. 7. Fuel Economy and Drivability Metrics on the *Ensemble 2* set for the Ford Controller and controller *familes* are designed with statistics from FTP, NEDC, *Ensemble 1*, and *Ensemble 2*. Fuel Economy, Gear Events, and Engine Events are cumulative for the whole cycle set, approximately 1000 miles. The controllers are the same as those shown in Figures 5 and 6. Results normalized to the Ford Controller on FTP (Fig. 5).

The SPSDP-based controllers deliver 2-3% performance improvement over the industrial controller on real world driving patterns, with even more improvement on a government test cycle. Moreover, the SPSDP design procedure can be highly automated. For example, for a fixed vehicle model and set of drive cycle statistics, one hundred SPSDP controllers representing various combinations of fuel consumption and weighted drivability penalties can be designed on the University of Michigan computing grid in four hours.

From an academic standpoint, these results are significant because they validate SPSDP as a reasonable design method and by extension lend credibility to other methods in the literature. Comparisons between industrial and academic controllers are quite rare. From an industrial perspective, this method has additional benefits. The speed and ease with which SPSDP controllers can be designed may result in significant labor savings, faster overall development time, or the ability to evaluate more hardware design tradeoffs in the prototyping phase. With the ability to generate optimal tradeoff curves among competing performance metrics, such as drivability and fuel economy as studied here, the manufacturer gains additional insight into the operating point selection process.

This analysis shows that Shortest Path Stochastic Dynamic Programming is a viable method for designing real-world controllers. The controllers can be implemented directly with little manual adjustment, and generate performance exceeding the current industrial state of the art.

References

[1] C. Belton, *et al.*, “A vehicle model architecture for vehicle system control design,” in *Proceedings of SAE 2003 World Congress & Exhibition.*, 2003.

[2] A. Boyali, *et al.*, “Modeling and control of a four wheel drive parallel hybrid electric vehicle,” in *Proc. IEEE International Conference on Control Applications*, Oct. 2006, pp. 155–162.

[3] L. Graham, M. Christenson, and D. Karman, “Light duty hybrid vehicles - influence of driving cycle and operating temperature on fuel economy and ghg emissions,” in *Proceedings of the IEEE EIC Climate Change Technology*, M. Christenson, Ed., 2006, pp. 1–6.

[4] I. Kolmanovsky, I. Siverguina, and B. Lygoe, “Optimization of powertrain operating policy for feasibility assessment and calibration: stochastic dynamic programming approach,” in *Proceedings of the American Control Conference.*, vol. 2, 2002.

[5] D. LeBlanc, *et al.*, “Road departure crash warning system field operational test: Methodology and results.” University of Michigan Transportation Research Institute, Tech. Rep. UMTRI-2006-9-1, June 2006, http://www-nrd.nhtsa.dot.gov/pdf/nrd-12/RDCW-Final-Report-Vol.1_JUNE.pdf.

[6] C.-C. Lin, H. Peng, and J. Grizzle, “A stochastic control strategy for hybrid electric vehicles,” in *Proceedings of the American Control Conference*, 2004.

[7] C.-C. Lin, H. Peng, J. Grizzle, and J.-M. Kang, “Power management strategy for a parallel hybrid electric truck,” *IEEE Transactions on Control Systems Technology*, vol. 11, no. 6, pp. 839–849, 2003.

[8] C.-C. Lin, H. Peng, J. Grizzle, J. Liu, and M. Busdiecker, “Control system development for an advanced-technology medium-duty hybrid electric truck,” in *Proceedings of the International Truck & Bus Meeting & Exhibition, Ft. Worth, TX, USA*, 2003.

[9] C. Musardo, G. Rizzoni, and B. Staccia, “A-ECMS: An adaptive algorithm for hybrid electric vehicle energy management,” in *Proceedings of the European Control Conference Decision and Control CDC-ECC.*, 2005.

[10] D. Opila, D. Aswani, R. McGee, J. Cook, and J. Grizzle, “Incorporating drivability metrics into optimal energy management strategies for hybrid vehicles,” in *Proceedings of 2008 IEEE Conference on Decision and Control*, 2008.

[11] G. Paganelli, M. Tateno, A. Brahma, G. Rizzoni, and Y. Guezennec, “Control development for a hybrid-electric sport-utility vehicle: strategy, implementation and field test results,” in *Proc. American Control Conference*, vol. 6, 2001.

[12] P. Pisu, K. Koprubasi, and G. Rizzoni, “Energy management and drivability control problems for hybrid electric vehicles,” in *Proceedings of the European Control Conference Decision and Control CDC-ECC.*, 2005.

[13] A. Sciarretta and L. Guzzella, “Control of hybrid electric vehicles,” *IEEE Control Systems Magazine*, vol. 27, no. 2, pp. 60–70, 2007.

[14] E. Tate, J. Grizzle, and H. Peng, “Shortest path stochastic control for hybrid electric vehicles,” *International Journal of Robust and Nonlinear Control*, vol. 18, pp. 1409–1429, 2008.