Proceedings of the ASME 2007 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference
IDETC/CIE 2007
September 4-7, 2007, Las Vegas, Nevada, USA

DETC2007-34718

A LEARNING ALGORITHM FOR OPTIMAL INTERNAL COMBUSTION ENGINE CALIBRATION IN REAL TIME

Andreas A. Malikopoulos*
Department of Mechanical Engineering

Panos Y. Papalambros
Department of Mechanical Engineering

Dennis N. Assanis
Department of Mechanical Engineering

University of Michigan, Ann Arbor, Michigan 48109, U.S.A.

ABSTRACT

Advanced internal combustion engine technologies have increased the number of accessible variables of an engine and our ability to control them. The optimal values of these variables are designated during engine calibration by means of a static correlation between the controllable variables and the corresponding steady-state engine operating points. While the engine is running, these correlations are being interpolated to provide values of the controllable variables for each operating point. These values are controlled by the electronic control unit to achieve desirable engine performance, for example in fuel economy, pollutant emissions, and engine acceleration. The state-of-the-art engine calibration cannot guarantee continuously optimal engine operation for the entire operating domain, especially in transient cases encountered in driving styles of different drivers. This paper presents the theoretical basis and algorithmic implementation for allowing the engine to learn the optimal set values of accessible variables in real time while running a vehicle. Through this new approach, the engine progressively perceives the driver's driving style and eventually learns to operate in a manner that optimizes specified performance indices. The effectiveness of the approach is demonstrated through simulation of a spark ignition engine, which learns to optimize fuel economy with respect to spark ignition timing, while it is running a vehicle.

Keywords: Markov Decision Process (MDP), learning algorithms, sequential decision-making under uncertainty, simulation-based optimization, reinforcement learning, internal combustion engine calibration, fuel economy

1. INTRODUCTION

The growing requests for better performance and fuel economy, and reduced emissions, have motivated continued research in advanced internal combustion engine technologies. These technologies, such as fuel injections systems, variable geometry turbocharging, variable valve actuation, and exhaust gas recirculation, have increased the number of accessible engine controllable variables, and our ability to optimize engine operation. In particular, the determination of the optimal values of these variables, referred to as engine calibration, have been shown to be especially critical for achieving high engine performance and fuel economy while meeting emission standards. Consequently, engine calibration is defined as a procedure that optimizes one or more engine performance indices, e.g., fuel economy, emissions, or engine performance with respect to the engine controllable variables. Engine calibration generates a static correlation between the optimal values of the controllable variables and the corresponding steady-state engine operating points to coordinate optimal performance of the specified indices. This correlation is incorporated into the electronic control unit (ECU) of the engine to control engine operation, so that optimal values of the specified indices are maintained.

Despite the advanced engine technologies, however, continuously optimal engine operation has not yet been possible. State-of-the-art engine calibration methods rely on dynamometer static correlations for steady-state operating points accompanied by transient vehicle testing. However, the calibration process, its duration, and its cost grow exponentially with the number of controllable variables and optimal

* Author of correspondence, Phone: (734) 647-1409, Fax: (734) 764-4256, Email: amaliko@umich.edu

 Copyright © 2007 by ASME

calibration for the entire feasible engine operating domain cannot be guaranteed. Even for engines with simple technologies, achievement of the optimal calibrations may become impractical [1]. In addition, current calibration methods cannot guarantee optimal engine operation in transient cases encountered in driving styles of different drivers [2].

To pre-specify the huge number of transient operations is impractical and, thus, calibration cannot generate optimal static correlations for all transient cases *a priori*. Transient operation constitutes the largest segment of engine operation over a driving cycle compared to the steady-state one [3, 4]. Emissions during transient operation are extremely complicated [4], vary significantly with each particular driving cycle [5, 6], and are highly dependent upon the calibration [6, 7]. Engine operating points, during the transient period before their steady-state value is reached, are associated with different Brake-Specific Fuel Consumption (BSFC) values, depending on the directions from which they have been arrived, as illustrated qualitatively in Figures 1 and 2. Pollutant emissions such as NO_x, and particulate matters, demonstrate the same qualitative behavior, as shown by Hagena *et al.* [8].

The main objective of calibration methods is to expedite dynamometer tests significantly using a smaller subset of tests. This subset is utilized either in implementing engine calibration experimentally or in developing mathematical models for evaluating engine output. Using these models, optimization methods can determine the engine calibration static correlations between steady-state operating points and the controllable engine variables [9]. Design of Experiments (DoE) [10-12] has been widely used as the baseline method. Major applications include catalyst system optimization [13], optimization of variable valve trains for performance and emissions [14-17], implementation of dynamic model-based engine calibrations [18, 19], and optimization of fuel consumption in a spark ignition engine with dual-continuously controlled camshaft phasing with respect to valve timing [20].

DoE-based calibration systems are typically used to reduce the scope of the experiments required to derive the optimal engine calibration correlation under steady-state operating conditions. Dynamic model-based calibration, however, utilizes high-fidelity dynamic or transient engine modeling. The data required to develop the engine model are obtained by operating the engine through a set of transient dynamometer tests while the engine calibration is perturbed in real time by a reconfigurable rapid prototyping control system. The predictive engine model produced in this fashion utilizes a combination of equation-based and neural network methods. DoE-experimental calibration is well suited only for steady-state engine operation over some driving cycle. In contrast, dynamic modeling produces a transient or dynamic engine model capable of predicting engine operating cycle. The steady-state optimal engine calibration can be produced from the transient engine model as a sub-set of the transient engine operation. Rask *et al.* [1] developed a simulation-based calibration method to rapidly

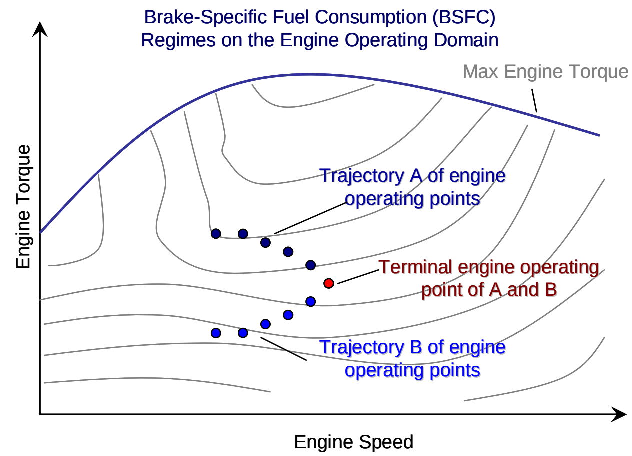

Figure 1. Two trajectories A, and B, of engine operating points ending at the same operating point.

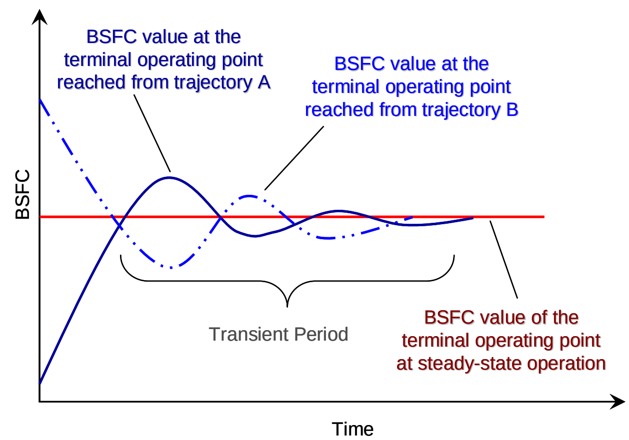

Figure 2. BSFC value of the terminal engine operating point as reached from trajectories A, and B.

generate optimized correlations for a V6 engine equipped with two-step variable valve actuation and intake cam phasing. Guerrier *et al.* [18] employed DoE and advanced statistical modeling to develop empirical models to advance the powertrain control module calibration tables. Stuhler *et al.* [19] implemented a standardized and automated calibration environment, supporting the complexity of gasoline direct injection engines, for an efficient calibration process using an online DoE to decrease the calibration cost. These engine models can predict engine output over transient operation. However, not all the correlations of optimal values of the controllable engine variables associated with the transient operating points can be quantified explicitly; to pre-specify the entire transient engine operation is impractical, and thus, engine calibration correlations cannot be optimized for these cases *a priori*.

Various approaches have been proposed for using artificial neural networks (ANN) to promote modeling and calibration of engines [21-26]. However, ANNs are application-specific and exhibit unpredictable behavior when previously unfamiliar data

Copyright © 2007 by ASME

are presented to them. These difficulties increase if a nonlinear dynamic presentation of a system is to be realized, because of the increasing number of possibilities related to the dynamic and the interactions between the input signals. ANNs are suited for formulating objective functions, evaluating the specified engine performance indices with respect to the controllable engine variables and, thus, deriving the engine calibration correlations. They are computationally efficient for optimization requiring hundreds of function evaluations. However, optimal engine calibration for the entire engine operating domain seldom is guaranteed even for steady-state operating points. Moreover, the correlations between optimal values of the controllable engine variables and the transient operating points, overall, cannot be quantified explicitly prohibiting *a priori* optimal engine calibration.

This paper introduces a method to make the engine an autonomous system that can learn its optimal calibration for the entire engine operating domain in real time while running a vehicle. Section 2 builds the general theoretical framework of considering the engine operation as a stochastic process, and introduces the predictive optimal stochastic learning algorithm. This algorithm predicts the optimal correlation between the engine controllable variables and the operating points (both steady-state and transient) based on observations of the engine outputs. The effectiveness of the approach is demonstrated in Section 3 through simulation of a Spark Ignition (SI) engine model; while the SI model is running, it progressively perceives the driver's conventional driving style and learns the optimal spark ignition values, as illustrated in Section 4. Finally, conclusions are presented in Section 5.

2. PROPOSED METHOD

Engine operation is described in terms of engine operating points and the evaluation of engine performance indices is a function of various controllable variables. In our approach, the engine performance indices are treated as random functions. Consequently, the engine is treated as a controlled stochastic system, and engine operation is treated as a stochastic process. The problem of engine calibration is thus reformulated as a sequential decision-making problem under uncertainty. The main objective towards the solution in this problem is to select the optimum values of the controllable variables for each engine operating point in real time that optimize the random functions (engine performance indices). The Markov Decision Process (MDP), extensively covered by Puterman [27], provides the mathematical framework for modeling sequential decision-making problems under uncertainty [28]; it is comprised of (a) a decision maker (engine), (b) states (engine operating points), (c) actions (controllable variables), (d) transition probability matrices (driver), (e) transition reward matrices (engine performance indices), and (f) optimization criteria (e.g., maximizing fuel economy, minimizing pollutant emissions, maximizing engine performance).

2.1 Markov Decision Process (MDP)

Formally, MDP is a discrete time stochastic control process defined as the tuple:

$$\mathcal{X}_n=\{\mathcal{S}, \mathcal{A}, \mathbf{P}(\cdot,\cdot), \mathbf{R}(\cdot,\cdot)\} \tag{1}$$

where $\mathcal{S}=\{s_i \mid i=1,2,...N\}, N\in\mathcal{N}$ denotes a finite state space, $\mathcal{A}=\bigcup_{s_i\in\mathcal{S}} A(s_i)$ stands for a finite action space, $\mathbf{P}(\cdot,\cdot)$ is the transition probability matrix, and $\mathbf{R}(\cdot,\cdot)$ is the transition reward matrix. The decision-making process occurs at each of a sequence of stages $\kappa=0,1,2,...M, M\in\mathcal{N}$. At each stage, the decision-maker observes a system's state $s_i\in\mathcal{S}$, and executes an action $\alpha\in A(s_i)$, from the feasible set of actions $A(s_i)\subseteq\mathcal{A}$ at this state. At the next stage, the system transits to the state $s_j\in\mathcal{S}$ imposed by the conditional probabilities $p(s_j \mid s_i,\alpha)$, designated by the transition probability matrix $\mathbf{P}(\cdot,\cdot)$. These conditional probabilities of $\mathbf{P}(\cdot,\cdot)$, $p:\mathcal{S}\times\mathcal{A}\rightarrow[0,1]$, satisfy the following constraint

$$\sum_{j=1}^{N} p(s_j \mid s_i,\alpha)=1. \tag{2}$$

Following this state transition, the decision-maker receives a reward associated with the action α, $R(s_j \mid s_i,\alpha), R:\mathcal{S}\times\mathcal{A}\rightarrow\mathcal{R}$ as imposed by the transition reward matrix $\mathbf{R}(\cdot,\cdot)$. A two-state MDP problem with the conditional probabilities and rewards corresponding to all possible transitions is illustrated in Figure 3. The states of an MDP possess the Markov property, stating that the conditional probability distribution of future states of the process, given the present state and all past states, depends only upon the current state and not on any past states, i.e., it is conditionally independent of the past states (the path of the process) given the present state.

Mathematically, the Markov property requires that

$$\begin{aligned} &p(X_{n+1}=s_j \mid X_n=s_i, X_{n-1}=s_{i-1},..., X_0=s_0)= \\ &=p(X_{n+1}=s_j \mid X_n=s_i). \end{aligned} \tag{3}$$

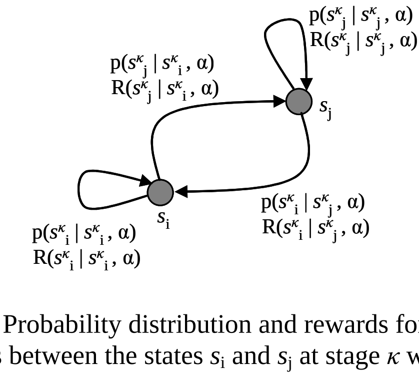

Figure 3. Probability distribution and rewards for all possible transitions between the states s_i and s_j at stage κ when action α is taken

Copyright © 2007 by ASME

The solution to a MDP can be expressed as a policy $\pi = \{\mu_0, \mu_1, ...\mu_M\}$, which provides the action to be executed for a given state, regardless of prior history; μ_κ is a function mapping states s_i into actions $\alpha = \mu_\kappa(s_i)$, such that $\mu_\kappa(s_i) \in A(s_i)$. Such policies are addressed as admissible. Consequently, for any initial state at stage κ = 0, s_i^0, and for any finite sequence of stages $\kappa = 0, 1, 2, ...M, M \in \mathcal{N}$, the expected accumulated value of the rewards of the decision maker is

$$V_\pi(s_l^M) = E\{R(s_j^0 \mid s_i^0, \mu_0(s_i^0)) + \sum_{\kappa=1}^{M} V_\pi(s_j^\kappa)\}, \qquad (4)$$
$$\forall s_i, s_j, s_l \in \mathcal{S}.$$

In the finite-horizon context the decision maker should maximize the accumulated value for the next M stages; more precisely, an optimal policy π^* is one that maximizes the overall expected accumulated value of the rewards

$$V_{\pi^*}(s_l^M) = \max_{\pi \in \mathcal{A}} V_\pi(s_l^M). \qquad (5)$$

Consequently, the optimal policy $\pi^* = \{\mu_0^*, \mu_1^*, ...\mu_M^*\}$ for the M-stage sequence is

$$\pi^* = \arg\max_{\pi \in \mathcal{A}} V_{\pi^*}(s_l^M). \qquad (6)$$

The finite-horizon model is appropriate when the decision-maker's "lifetime" is known, namely, the terminal stage of the decision-making sequence. However, in most real-life problems this is not the case; these problems are modeled in the infinite-horizon context, where the overall expected reward is the limit of the corresponding M-stage overall reward as $M \to \infty$:

$$V_{\pi^*}(s_l^M) = \lim_{M \to \infty} \max_{\pi \in \mathcal{A}} V_\pi(s_l^M). \qquad (7)$$

This relation is extremely valuable for various MDP problems, where the terminal stage is unknown; Eq. (7) holds under certain conditions [29].

A large class of sequential decision-making problems under uncertainty is solved by using classical dynamic programming algorithms, originally proposed by Bellman [30]. Algorithms, such as value iteration, policy iteration, and linear programming are employed to find optimal solution of MDPs. However, the computational complexity of these algorithms in some occasions may be prohibitive and can grow intractably with the size of the problem and its related data. In addition, dynamic programming algorithms require the realization of the transition probability matrix, $\mathbf{P}(\cdot,\cdot)$, and the transition reward matrix, $\mathbf{R}(\cdot,\cdot)$. For complex systems with large state space, the transition probability and reward matrices can be either impractical or impossible to compute.

Viable alternatives for approaching these problems through a simulation-based stochastic framework have been primarily developed in the field of Reinforcement Learning (RL) [31, 32]. RL methods aim to provide effective near-optimal solutions to complex problems of planning and sequential decision-making under uncertainty. Wheeler *et al.* [33] proposed a learning method in sequential stochastic games under certain properties; Sutton *et al.* [34] introduced a class of incremental learning procedures specialized for prediction, using past experience with an incomplete known system to predict its future behavior; Watkins [35] developed an algorithm for systems to learn how to act optimally in controlled Markovian domains amounting to an incremental method for dynamic programming imposing limited computational demands. These aforementioned classic RL methods and algorithms have been utilized successfully in various applications, e.g., robotics, control, operations research, games, human-computer interaction, economics/finance, and marketing.

The rigorous mathematical assumptions required by the majority of existing RL algorithms to converge to optimal solutions impose limitations in efficiently employing these algorithms to solve engineering problems. For the engine calibration problem built upon the MDP theoretical framework described above, a learning process as employed in RL [32] and a new predictive optimal stochastic control algorithm (POSCA) are developed. The learning process is applied to the engine to progressively perceive the conventional driving style of a driver designating the transition probability matrix, $\mathbf{P}(\cdot,\cdot)$. In addition, during this process the desired engine performance indices, e.g., fuel economy, pollutant emissions, engine performance, are represented by the elements of the transition reward matrix, $\mathbf{R}(\cdot,\cdot)$. The intention of the algorithm is to predict the optimal policy (values of the engine controllable variables) $\pi^* \in \mathcal{A}$, thus optimizing the expected accumulated value of the rewards in the infinite-horizon context.

2.2 The Learning Process of the Engine

The learning process transpires while the engine is running the vehicle and interacting with the driver. Taken in conjunction with assigning values of the controllable variables from the feasible action space, $\mathcal{A}$, this interaction portrays the progressive enhancement of the engine's "knowledge" of the driver's driving style with respect to the controllable variables. More precisely, at each of a sequence of stages $\kappa = 0, 1, 2, ...M$, as $M \to \infty$, the driver introduces a state $s_i^\kappa \in \mathcal{S}$ to the engine, and on that basis the engine selects an action, $\alpha^\kappa \in A(s_i)$. This state arises as a result of the driver's driving style corresponding to particular engine operating points. One stage later, as a consequence of its action, the

Copyright © 2007 by ASME

engine receives a numerical reward, $R^{\kappa+1} \in \mathscr{R}$, and transits to a new state $s_j^{\kappa+1} \in \mathcal{S}$ as illustrated in Figure 4.

At each stage, the engine implements a mapping from the Cartesian product of the state space and action space to the set of real numbers, $\mathcal{S} \times \mathcal{A} \to \mathscr{R}$, by means of the rewards that it receives. Similarly, another mapping from the Cartesian product of the state space and action space to the closed set [0,1] is executed, $\mathcal{S} \times \mathcal{A} \to [0,1]$, satisfying Eq. (2). The latter essentially perceives the conventional driving style as expressed by the incidence in which particular states or particular sequences of states arise. The implementation of these two mappings aims to disclose the optimal policy π^* (optimal set values of the controllable variables) of the engine designated by the predictive optimal control algorithm. This policy is expressed by means of a mapping from states to probabilities of selecting the actions, resulting in the highest expected accumulated value of the rewards.

A challenge in the learning process is the trade-off between exploration and exploitation of the action space. Specifically, the engine has to exploit what is already known regarding the correlation involving the driving style and the values of the controllable variables that maximize the rewards, and also to explore those actions that have not yet been tried for this driving style to assess whether these actions may result in higher rewards. This exploration-exploitation dilemma has been extensively reported in the literature; Iwata *et al.* [36] proposed a model-based learning method extending Q-learning and introducing two separated functions based on statistics and on information by applying exploration and exploitation strategies; Ishii *et al.* [37] developed a model-based reinforcement learning method utilizing a balance parameter, which is controlled based on variation of action rewards and perception of environmental change; Chan-Geon *et al.* [38] proposed an exploration-exploitation policy in Q-learning consisting of an auxiliary Markov process and the original Markov process; Miyazaki *et al.* [39] developed a unified learning system realizing the tradeoff between exploration and exploitation.

The exploration-exploitation dilemma is closely related to the type of problem along with the decision-maker's "lifetime" problem [40]: The longer the lifetime, the worse the consequences of prematurely converging on a sub-optimal solution. This could result from not fully exploring the entire feasible action space for each state. In our case, the objective is to make the engine learn its optimal calibration for the driver's driving style in the infinite-horizon context. Consequently, the engine has to explore the entire action space for any state being visited by the particular driving style. In particular, it is assumed that for any state $s_i \in \mathcal{S}$ corresponding to the driving style, all actions of the feasible action set $\alpha \in A(s_i)$ are selected at least once. This may result in sacrificing the engine performance indices in the short run; however, the ultimate goal for the engine is to progressively perceive the driving style and learn the optimal policy in the long run.

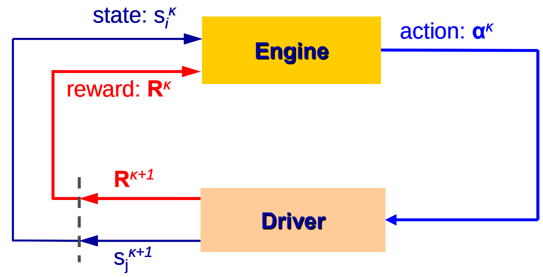

Figure 4. The learning process during the interaction between the engine and the driver

2.3 The Predictive Optimal Stochastic Control Algorithm (POSCA)

The learning process of the engine transpires at each stage κ in conjunction with actions $\alpha \in A(s_i)$ taken for each state $s_i \in \mathcal{S}$. At the early stages and until full exploration of the action set $A(s_i), \forall s_i \in \mathcal{S}$ occurs, the mapping from the states to probabilities of selecting the actions is constant; namely, the actions for each state are selected randomly with the same probability

$$p(\alpha \mid s_i) = \frac{1}{|A(s_i)|}, \forall \alpha \in A(s_i), \forall s_i \in \mathcal{S}. \tag{8}$$

Exploration of the entire feasible action set is important to evade sub-optimal solutions when the exploration phase is done. The predictive optimal stochastic control algorithm (POSCA) is thus used after the exploration phase. The main objective of POSCA is to realize the action at each stage which is optimal not only for the current state, but also for the subsequent states over the following stages. The subsequent states are predicted by POSCA by means of the conditional probabilities $p(s_j \mid s_i, \alpha), p: \mathcal{S} \times \mathcal{A} \to [0,1]$ of the transition probability matrix $\mathbf{P}(\cdot,\cdot)$. The expected accumulated value of the rewards for the subsequent states is perceived in terms of the magnitude, $\tilde{T}(s_j)$, defined as the maximum average future reward. Suppose that the current state is s_i and the following state given an action $\alpha \in A(s_i)$ is s_j with probability $p(s_j \mid s_i, a)$. The average future overall reward will be

$$\tilde{T}(s_j) = \max_{a \in \mathcal{A}} \left(\frac{\sum_{l=1}^{N} p(s_l \mid s_j, a) \cdot R(s_l \mid s_j, a)}{N} \right), \quad \forall s_j \in \mathcal{S}. \tag{9}$$

The immediate expected reward by transiting from state s_i to state s_j given an action $\alpha \in A(s_i)$ is

Copyright © 2007 by ASME

$$\tilde{t}(s_j \mid s_i, a) = p(s_j \mid s_i, a) \cdot R(s_j \mid s_i, a). \quad (10)$$

For the problem of optimal control of uncertain systems, which is treated in a stochastic framework, all uncertain quantities are described by probability distributions and the expected value of the overall reward is maximized. In this context, the optimal policy π^* realized by POSCA is based on the maxmin control approach, whereby the worst possible values of the uncertain quantities within the given set are assumed to occur. This is a pessimistic point of view which essentially assures that the optimal policy will result in at least a minimum overall reward value. Consequently, being at state s_i, POSCA predicts the optimal policy π^* in terms of the values of the controllable variables as

$$\pi^*(s_i) = \arg\max \left\{ \min_{\substack{s_j \in S \\ a \in A}} \left(\tilde{t}(s_j \mid s_i, a) + \tilde{T}(s_j) \right) \right\}, \quad \forall s_i, s_j \in S. \quad (11)$$

3. EXAMPLE

An example of real-time, self-learning optimization of the calibration with respect to spark ignition timing in a spark ignition engine is presented. In spark ignition engines the fuel and air mixture is prepared in advance before it is ignited by the spark discharge. The major objectives for the spark ignition are to initiate a stable combustion and to ignite the air-fuel mixture at the crank angle resulting in maximum efficiency, while fulfilling emissions standards and preventing the engine from knocking. Simultaneous achievement of the aforementioned objectives is sometimes inconsistent; for instance, at high engine loads the ignition timing for maximum efficiency has to be abandoned in favor of prevention of engine destruction by way of engine knock. Two essential parameters are controlled with the spark ignition: ignition energy and ignition timing. Control of ignition energy is important for assuring combustion initiation, but the focus here is on the spark timing that maximizes engine efficiency. Ignition timing influences nearly all engine outputs and is essential for efficiency, drivability, and emissions. The optimum spark ignition timing generating the maximum engine brake torque is defined as Maximum Brake Torque (MBT) timing [41]. Any ignition timing that deviates from MBT lowers the engine's output torque as illustrated in Figure 5. A useful parameter for evaluating fuel consumption of an engine is the Brake-Specific Fuel Consumption (BSFC), defined as the fuel flow rate per unit power output. This parameter evaluates how efficiently an engine is utilizing the fuel supplied to produce work

$$bsfc(g/kW \cdot h) = \frac{\dot{m}_f (g/h)}{P(kW)}, \quad (12)$$

where $\dot{m}_f$ is the fuel mass flow rate per unit time and P is engine's power output. Continuous engine operation at MBT ensures optimum fuel economy with respect to the spark ignition timing.

For a successful real-time, self-learning optimization of engine calibration in terms of spark ignition timing, the engine should realize the MBT timing for each engine operating point (steady-state and transient) dictated by the driving style of a driver. Consequently, by achieving MBT timing for all steady-state and transient operating points an overall improvement of the BSFC is expected. Aspects of preventing knocking are not considered in this example; however, they can be easily incorporated by defining the spark ignition space to include the maximum allowable values.

The software package enDYNA by TESIS [42] suitable for real-time simulation of internal combustion engines is employed. The software utilizes thermodynamic models of the gas path and is well suited for testing and development of electronic engine controllers. In the example, a four-cylinder gasoline engine is used from the enDYNA model database. The software's static correlation involving spark ignition timing and engine operating points is bypassed to incorporate the POSCA algorithm. This correlation is designated by the baseline calibration that enDYNA model is accompanied by, and is included in, the engine's ECU. In the context of the MDP problem, the states represent the pair of gas-pedal position and engine speed, and the actions denote the spark ignition timing; the rewards that the decision-maker (engine) receives correspond to the engine brake torque.

The engine model is run repeatedly over the same driving style represented by the pedal position. Every run over this driving style constitutes one complete simulation. To evaluate the efficiency of our approach in both steady-state and transient engine operation, the pedal position rate is chosen to represent an aggressive acceleration, as illustrated in Figure 6.

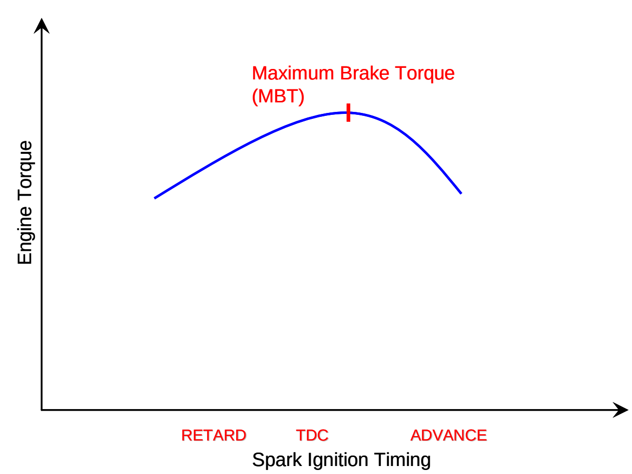

Figure 5. Effect of spark ignition timing on the engine brake torque at constant engine speed

Copyright © 2007 by ASME

Before initiating the first simulation of the engine model, the elements of the transition probability and reward matrix are assigned to be zero. That is, the engine at the beginning has no knowledge regarding the particular style and the values of the rewards associated with the controllable variables.

4. RESULTS

After completing the fourth simulation, POSCA specified the optimal policy in terms of the spark ignition timing, as shown in Figure 7, and compared with the spark ignition timing designated by the baseline calibration of the enDYNA model. The optimal policy resulted in higher engine brake torque compared to the baseline calibration as shown in Figures 8 and 9. This improvement indicates that the engine with self-learning calibration was able to operate closer to MBT timing. Having the engine operate at MBT timing resulted in an overall minimization of the BSFC, illustrated in Figure 10. Figure 11 compares the velocity of the two vehicles, one carrying the engine with the baseline calibration and the other with the self-calibrated one.

The two vehicles were simulated for the same driving style, namely, the same pedal-position rate. The vehicle carrying the engine with the self-learning calibration demonstrated higher velocity, since the engine produced higher brake torque for the same gas-pedal position rate.

Consequently, if the driver wishes to follow a specific vehicle's speed profile, this can now be achieved by stepping on the gas-pedal more lightly than required in the engine with the baseline calibration and, therefore, directly enabling in additional benefits in fuel economy.

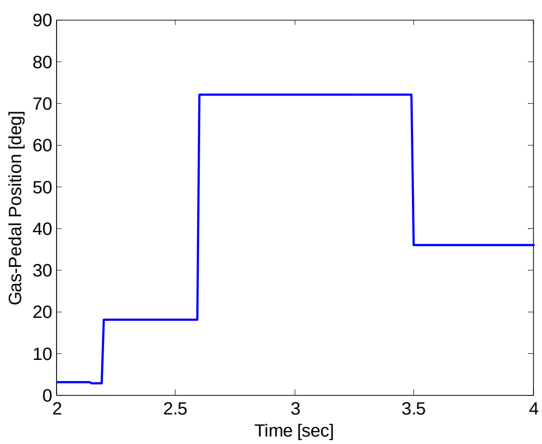

Figure 6. Gas-pedal position rate representing a driver's driving style

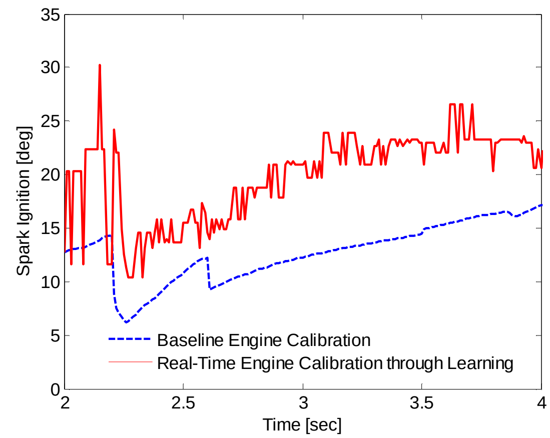

Figure 7. Spark ignition timing over the driving style

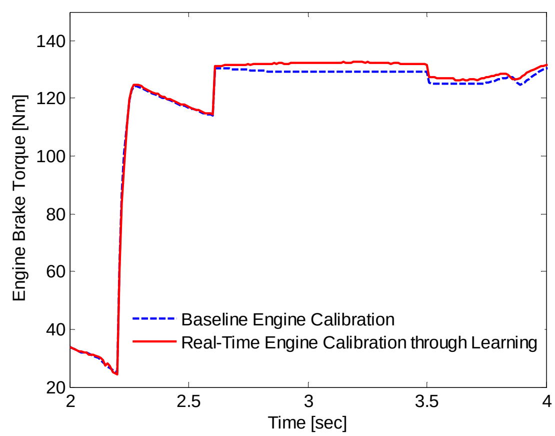

Figure 8. Engine brake torque

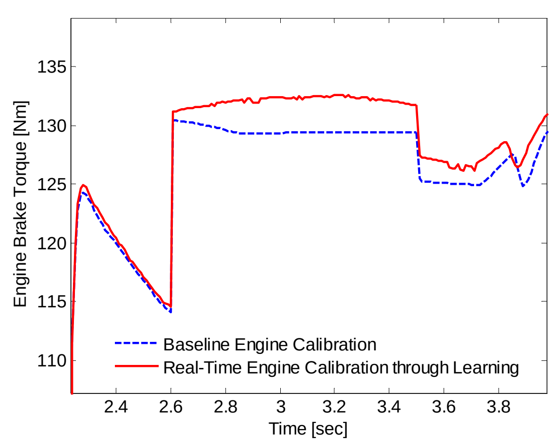

Figure 9. Engine brake torque (zoom-in)

Copyright © 2007 by ASME

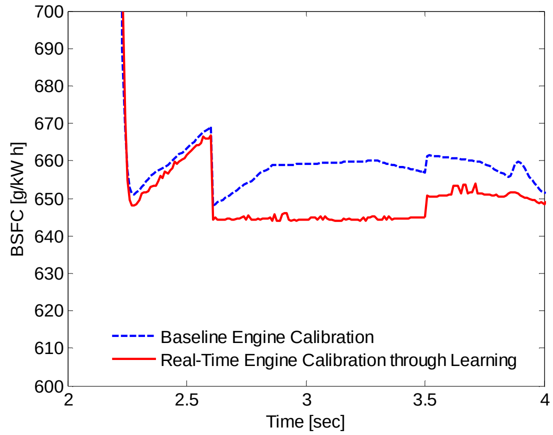

Figure 10. BSFC comparison between the baseline and self-learning calibration

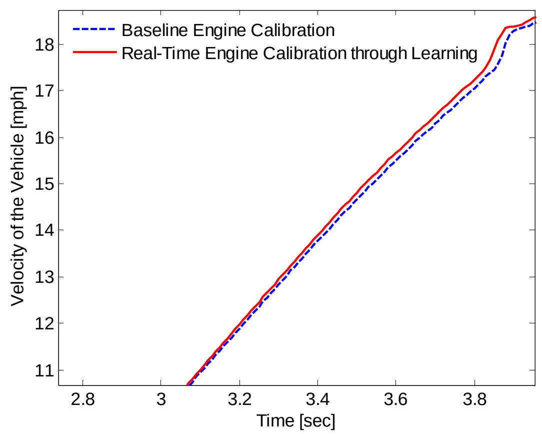

Figure 11. Velocity of the two vehicles carrying the engine with baseline and self-learning calibration

5. CONCLUDING REMARKS

The POSCA algorithm allows an internal combustion engine to act as an autonomous system that can learn its optimal calibration for both steady-state and transient operating points in real time while running a vehicle. The engine progressively perceives the driver's driving style and, eventually, learns to coordinate optimal performance of several specified indices, e.g., fuel economy, pollutant emissions, engine performance, for this particular driving style. The longer the engine runs with a particular driving style, the better the specified indices will be. The engine's ability to learn its optimum calibration is not limited, however, to a particular driving style. The engine can learn to operate optimally for different drivers by assigning the transition probability $\mathbf{P}(\cdot,\cdot)$, and reward matrices $\mathbf{R}(\cdot,\cdot)$ for each driver. The drivers would indicate their identities before starting the vehicle to denote the pair of these matrices that the engine should employ. The engine can then adjust its operation to be optimal for a particular driver based on what it has learned in the past regarding his/her driving style.

The example presented the real-time, self learning calibration of a spark ignition engine with respect to spark ignition timing. The engine was able to realize the MBT timing for each engine operating point (steady-state and transient) designated by a driving style representing an aggressive acceleration and, thus, minimizing BSFC. Aspects of preventing knocking were not considered in this example; however, a potential extension is possible such as defining the spark ignition space to include the maximum allowable values ensuring engine operation without knocking. POSCA predicted efficiently the optimal control policy (spark ignition timing) for each state (engine operating point). It is left for future research to explore the impact of traffic patterns, and terrain, on the general applicability of having the engine learn its optimal calibration for an individual driving style. Future research should also investigate the potential of advancing POSCA in predicting the optimal policy of a number of controllable variables associated with different states and, thus, avoiding the enhancement of the problem's dimensionality. Increased dimensionality is a major challenge for learning algorithms.

ACKNOWLEDGMENTS

This research was partially supported by the Automotive Research Center (ARC), a U.S. Army Center of Excellence in Modeling and Simulation of Ground Vehicles at the University of Michigan. The engine simulation package enDYNA was provided by TESIS DYNAware GmbH. This support is gratefully acknowledged.

REFERENCES

[1] Rask, E. and Sellnau, M., "Simulation-Based Engine Calibration: Tools, Techniques, and Applications," SAE Transactions-Journal of Engines, v. 113, 2004, SAE 2004-01-1264.

[2] Atkinson, C. and Mott, G., "Dynamic Model-Based Calibration Optimization: An Introduction and Application to Diesel Engines," SAE World Congress, Detroit, Michigan, April 11-14, 2005, SAE 2005-01-0026.

[3] Rakopoulos, C. D., Giakoumis, E. G., Hountalas, D. T., and Rakopoulos, D. C., "The Effect of Various Dynamic, Thermodynamic and Design Parameters on the Performance of a Turbocharged Diesel Engine Operating under Transient Load Conditions," SAE 2004 World Congress and Exhibition, Detroit, Michigan, April 8-11, 2004, SAE 2004-01-0926.

[4] Wijetunge, R. S., Brace, C. J., Hawley, J. G., Vaughan, N. D., Horroocks, R. W., and Bird, G. L., "Dynamic Behavior of a High-Speed, Direct-Injection Diesel Engine," SAE Transactions-Journal of Engines, v. 108, 1999, SAE 1999-01-0829.

[5] Clark, N. N., Gautam, M., Rapp, B. L., Lyons, D. W., Graboski, M. S., McCormick, R. L., Alleman, T. L., and National, P. N., "Diesel and CNG Transit Bus Emissions Characterization by Two Chassis Dynamometer

Copyright © 2007 by ASME

Laboratories: Results and Issues," SAE Transactions-Journal of Fuels and Lubricants, v. 108, 1999, SAE 1999-01-1469.
[6] Samulski, M. J. and Jackson, C. C., "Effects of Steady-State and Transient Operation on Exhaust Emissions from Nonroad and Highway Diesel Engines," SAE Transactions-Journal of Engines, v. 107, 1998, SAE 982044.
[7] Green, R. M., "Measuring the Cylinder-to-Cylinder EGR Distribution in the Intake of a Diesel Engine During Transient Operation," SAE Transactions-Journal of Engines, v. 109, 2000, SAE 2000-01-2866.
[8] Hagena, J. R., Filipi, Z. S., and Assanis, D. N., "Transient Diesel Emissions: Analysis of Engine Operation During a Tip-In," SAE 2006 World Congress, Detroit, Michigan, April 3-6, 2006, SAE 2006-01-1151.
[9] Papalambros, P. Y. and Wilde, D. J., *Principles of Optimal Design: Modeling and Computation*, 2nd edition, Cambridge University Press, July 2000.
[10] Clarke, G. M. and Kempson, R. E., *Introduction to the Design and Analysis of Experiments*, Hodder Arnold, November 1996.
[11] Hicks, C. R. and Turner, K. V., *Fundamental Concepts in the Design of Experiments*, 5th edition, Oxford University Press, USA, March 1999.
[12] Diamond, W. J., *Practical Experiment Designs : for Engineers and Scientists*, 3rd edition, John Wiley & Sons, February 2001.
[13] Edwards, S. P., Grove, D. M., and Wynn, H. P., *Statistics for Engine Optimization*, 1st edition, John Wiley & Sons Canada, December 2000.
[14] Regner, G., Teng, H., Wieren, P. V., Park, J. I., Park, S. Y., and Yeom, D. J., "Performance Analysis and Valve Event Optimization for SI Engines Using Fractal Combustion Model," Powertrain and Fluid Systems Conference and Exhibition, Toronto, Ontario, Canada, September 16-19, 2006, SAE 2006-01-3238.
[15] Ghauri, A., Richardson, S. H., and Nightingale, C. J. E., "Variation of Both Symmetric and Asymmetric Valve Events on a 4-Valve SI Engine and the Effects on Emissions and Fuel Economy," SAE 2000 World Congress, Detroit, Michigan, March 6-9, 2000, SAE 2000-01-1222.
[16] Amann, M., Buckingham, J., and Kono, N., "Evaluation of HCCI Engine Potentials in Comparison to Advanced Gasoline and Diesel Engines," Powertrain and Fluid Systems Conference and Exhibition, Toronto, Ontario, Canada, September 16-19, 2006, SAE 2006-01-3249.
[17] Jankovic, M. and Magner, S., "Fuel Economy Optimization in Automotive Engines," Proceedings of the 2006 American Control Conference, Minneapolis, MN, USA, 2006.
[18] Guerrier, M. and Cawsey, P., "The Development of Model-Based Methodologies for Gasoline IC Engine Calibration," SAE Transactions-Journal of Engines, v. 113, 2004, SAE 2004-01-1466.
[19] Stuhler, H., Kruse, T., Stuber, A., Gschweitl, K., Piock, W., Pfluegl, H., and Lick, P., "Automated Model-Based GDI Engine Calibration Adaptive Online DoE Approach," SAE 2002 World Congress, Detroit, Michigan, March 3-7, 2002, SAE 2002-01-0708.
[20] Roepke, K. and Fischer, M., "Efficient Layout and Calibration of Variable Valve Trains," SAE Transactions-Journal of Engines, v. 110, 2001, SAE 2001-01-0668.
[21] Ayeb, M., Theuerkauf, H. J., and Winsel, T., "SI Engine Emissions Model Based on Dynamic Neural Networks and D- Optimality," SAE World Congress, Detroit, Michigan, April 11-14, 2005, SAE 2005-01-0019.
[22] Brahma, I., He, Y., and Rutland, C. J., "Improvement of Neural Network Accuracy for Engine Simulations," Powertrain and Fluid Systems Conference and Exhibition, Pittsburgh, Pennsylvania, September 27-30, 2003, SAE 2003-01-3227.
[23] Lowe, D. and Zapart, K., "Validation of Neural Networks in Automotive Engine Calibration," Proceedings of the Fifth International Conference on Artificial Neural Networks, Cambridge, UK, 1997.
[24] Meyer, S. and Greff, A., "New Calibration Methods and Control Systems with Artificial Neural Networks," SAE 2002 World Congress, Detroit, Michigan, March 4-7, 2002, SAE 2002-01-1147.
[25] Wu, B., Prucka, R. G., Filipi, Z. S., Kramer, D. M., and Ohl, G. L., "Cam-Phasing Optimization Using Artificial Neural Networks as Surrogate Models-Fuel Consumption and NOx Emissions," SAE 2006 World Congress, Detroit, Michigan, April 3-6, 2006, SAE 2006-01-1512.
[26] Wu, B., Prucka, R. G., Filipi, Z. S., Kramer, D. M., and Ohl, G. L., "Cam-Phasing Optimization Using Artificial Neural Networks as Surrogate Models~Maximizing Torque Output," SAE Transactions-Journal of Engines, v. 114, 2005, SAE 2005-01-3757.
[27] Puterman, M. L., *Markov Decision Processes: Discrete Stochastic Dynamic Programming*, 2nd Rev. edition, Wiley-Interscience, 2005.
[28] Sennott, L. I., *Stochastic Dynamic Programming and the Control of Queueing Systems*, 1st edition, Wiley-Interscience, September 1998.
[29] Bertsekas, D. P., *Dynamic Programming and Optimal Control (Volumes 1 and 2)*, Athena Scientific, September 2001.
[30] Bellman, R., *Dynamic Programming*. Princeton, NJ, Princeton University Press, 1957.
[31] Bertsekas, D. P. and Tsitsiklis, J. N., *Neuro-Dynamic Programming (Optimization and Neural Computation Series, 3)*, 1st edition, Athena Scientific, May 1996.
[32] Sutton, R. S. and Barto, A. G., *Reinforcement Learning: An Introduction (Adaptive Computation and Machine Learning)*, The MIT Press, March 1998.

Copyright © 2007 by ASME

[33] Wheeler, R. and Narenda, K., "Decentralized Learning in Finite Markov Chains," *IEEE Transactions on Automatic Control*, vol. 31(6), pp. 373-376, 1986.
[34] Sutton, R. S., "Learning to Predict by the Methods of Temporal Difference," *Machine Learning*, vol. 3, pp. 9-44, 1988.
[35] Watkins, C. J., Learning from Delayed Rewards, PhD Thesis, Kings College, Cambridge, England, May 1989.
[36] Iwata, K., Ito, N., Yamauchi, K., and Ishii, N., "Combining Exploitation-Based and Exploration-Based Approach in Reinforcement Learning," Proceedings of the Intelligent Data Engineering and Automated - IDEAL 2000, Hong Kong, China, 2000.
[37] Ishii, S., Yoshida, W., and Yoshimoto, J., "Control of Exploitation-Exploration Meta-Parameter in Reinforcement Learning," *Journal of Neural Networks*, vol. 15, pp. 665-87, 2002.
[38] Chan-Geon, P. and Sung-Bong, Y., "Implementation of the Agent Using Universal On-Line Q-learning by Balancing Exploration and Exploitation in Reinforcement Learning," *Journal of KISS: Software and Applications*, vol. 30, pp. 672-80, 2003.
[39] Miyazaki, K. and Yamamura, M., "Marco Polo: a Reinforcement Learning System Considering Tradeoff Exploitation and Exploration under Markovian Environments," *Journal of Japanese Society for Artificial Intelligence*, vol. 12, pp. 78-89, 1997.
[40] Kaelbling, L. P., Littman, M. L., and Moore, A. W., "Reinforcement Learning: a Survey," *Journal of Artificial Intelligence Research*, vol. 4.
[41] Heywood, J., *Internal Combustion Engine Fundamentals*, 1st edition, McGraw-Hill, April 1988.
[42] TESIS, <http://www.tesis.de/en/>.

Copyright © 2007 by ASME