

# Driving Pattern Recognition for Control of Hybrid Electric Trucks

CHAN-CHIAO LIN<sup>1</sup>, SOONIL JEON<sup>2</sup>, HUEI PENG<sup>3</sup>, AND JANG MOO LEE<sup>4</sup>

## SUMMARY

The design procedure for an adaptive power management control strategy, based on a driving pattern recognition algorithm is proposed. The design goal of the control strategy is to minimize fuel consumption and engine-out NOx and PM emissions on a set of diversified driving schedules. Six representative driving patterns (RDP) are designed to represent different driving scenarios. For each RDP, the Dynamic Programming (DP) technique is used to find the global optimal control actions. Implementable, sub-optimal control algorithms are then extracted by analyzing the behavior of the DP control actions. A driving pattern recognition (DPR) algorithm is subsequently developed and used to classify the current driving pattern into one of the RDPs; thus, the most appropriate control algorithm is selected adaptively. This “multi-mode” control scheme was tested on several driving cycles and was found to work satisfactorily.

## 1. INTRODUCTION

Over the past few years, significant research and development work was initiated, aiming to duplicate the success of the hybrid powertrain on passenger cars to light and heavy trucks. The 21st Century Truck program in the US, spearheaded by two government agencies, Department of Energy and Department of Defense, is one such example [1.]. It is widely believed that the 3-times fuel economy improvement demonstrated by several prototype hybrid passenger cars, produced under the sponsorship of the PNGV program, will be an unrealistic goal for hybrid trucks, especially if engine-downsizing is not an option. The downsized engine is generally not considered to be viable because the on-board energy storage element could be exhausted quickly for demanding truck operations. The recently announced emission rule for the US 2007 model year trucks makes it very clear that exhaust emission is also an important performance metric for hybrid trucks [2.].

---

<sup>1</sup> Ph.D. student, Department of Mechanical Engineering, University of Michigan

<sup>2</sup> Ph.D., School of Mechanical and Aerospace Engineering, Seoul National University

<sup>3</sup> Corresponding author: Associate Professor, Department of Mechanical Engineering, University of Michigan, hpeng@umich.edu

<sup>4</sup> Professor, School of Mechanical and Aerospace Engineering, Seoul National University

The objective of the power management control strategy is to develop a near-optimal power management strategy that determines the proper power split and transmission gear position to minimize the fuel consumption and emissions of the hybrid truck. In addition, the control strategy also needs to ensure that the power demand from the driver is satisfied and the state of charge (SOC) in the battery is maintained within pre-determined range under all driving conditions. The main challenge of the power management problem arises from the complex and coupling nature of sub-system efficiencies, together with the diverse driving scenarios. Many existing control strategies employ heuristic control techniques such as rules/fuzzy logic for the control algorithm development [3.-6.]. The majority of the heuristic approaches are based on single-element optimization concepts such as “load-leveling” for engines, charging/discharging efficiency for batteries, etc. For example, in the load-leveling concept, one attempts to operate the internal combustion engine around an optimal region, and uses the battery (or ultra-capacitor) as a load-leveling device. The obtained heuristic rules are usually tuned or optimized over a given driving cycle [7., 8.]. This approach can be used to design a control law quickly. The main drawback, however, is the fact that the product from this design approach has no guaranteed performance or level of optimality.

To ensure a known level of optimality, the control strategy can be obtained by using the dynamic programming (DP) technique [9.]. By discretizing the operation variables into grid points, global optimal control action can be obtained, up to the grid point accuracy. Although the optimal operating policy is derived for a given cycle, it provides an insight into how improvement is achieved, based on which a sub-optimal control strategy can be designed.

In this paper, a multi-mode control algorithm for the fuel economy and engine-out emission optimization of a parallel hybrid truck is presented. First, the design procedure for constructing sub-optimal control schemes was developed for six representative driving patterns (RDP). These rule-based, sub-optimal control schemes were obtained by learning the behavior of the DP optimal control laws under each of these driving modes. In other words, a systematic procedure was developed to obtain rule-based control algorithms that approximate the performance of the theoretically optimal DP results. It was found that the optimal DP results can be approximated by parameters associated with a power-split ratio curve, which makes it easy to extract sub-optimal control rules.

A real-time driving pattern recognition (DPR) algorithm is then developed to switch between these six rule-based control strategies, with the assumptions that (i) driving pattern does not change fast and thus historical pattern is likely to continue into the near future; and (ii) the sub-optimal control strategies are different enough that selecting a proper one among them will result in significant performance improvements [10.]. The DPR algorithm is developed based on the idea that driving scenarios can be differentiated by objective measures such as average propulsion power, braking energy, and the ratio of stop time to total time. It is determined that

these measures can be extracted accurately by using data over a historical window, for example, during the past 150 seconds. Two kinds of representative driving patterns can be constructed for the development of the real-time DPR algorithm--imaginary RDP and partial driving cycles [11.]. The imaginary RDP maneuvers are used in this paper. The DPR algorithm is trained to ensure a high success rate in identifying the driving pattern. Finally, six driving cycles that the DPR algorithm has never experienced before, are used to assess the overall performance of the proposed control system.

The remainder of this paper is organized as follows: in Section 2, the development of mode-specific sub-optimal controls by using dynamic programming is presented. The results obtained in Section 2 are used as “baseline single-mode control” and compared with the multi-mode control strategy presented in Section 3. Simulation results are given in Section 4. Finally, conclusions are given in Section 5.

## 2. SINGLE-MODE SUB-OPTIMAL CONTROL

The baseline truck studied in this paper is a Class VI, 7.3L diesel engine truck (International Truck, 4700 series), mainly used for urban delivery tasks. The model and hybridization of this truck has been presented in previous publications [12., 13.], in which a parallel hybrid configuration of the truck with a smaller engine (5.5L) and a 49KW electric motor was developed. Key parameters of this vehicle are given in Table 1. In this section, the dynamic programming technique for the design of a mode-specific sub-optimal control strategy is described. Given a specific driving schedule, the optimal control policy is solved by the dynamic programming technique and an implementable rule-based strategy can be extracted from the optimal DP results. The term, “single-mode,” refers to the fact that the control rule is obtained from a specific driving pattern. This design procedure will serve as the basis for the “multi-mode” control to be described in the next section.

Table 1: Basic vehicle parameters

Component	Basic specification
DI Diesel Engine	V6, 5.475L, 157HP/2400rpm
DC Motor	Maximum Power: 49 kW
	Maximum Torque: 274 N-m
Lead-acid Battery	Capacity: 18 Ah
	Number of modules: 25
Automatic Transmission	Nominal voltage: 12.5 (volts/module)
	4 speed, GR: 3.45/2.24/1.41/1.0
Vehicle	Curb weight: 7504 kg

## 2.1 Dynamic Programming (DP) Based Optimization

The control of hybrid electric vehicles is formulated as an optimal control problem in the dynamic programming approach [9.]. The goal is to find the control actions (engine power, motor power, and gear selection) of the hybrid powertrain to minimize a cost function, which consists of the weighted sum of fuel consumption and engine-out emissions over a defined driving cycle:

$$J = \sum_{k=0}^{N-1} [fuel(k) + \mu \cdot NOx(k) + \nu \cdot PM(k)] + \alpha (SOC(N) - SOC_f)^2 \quad (1)$$

where  $k$  is the index of the time step,  $N$  is the duration of the driving cycle,  $\mu$  and  $\nu$  are the weighting factors for the engine-out NOx and PM emissions,  $\alpha$  is the weight for terminal battery charge deviation, and  $SOC$  is the battery state of charge. The terminal constraint on SOC ( $SOC_f$ ) is imposed to ensure the control strategy is not charge depleting.

Once the driving cycle is given, the required wheel torque and wheel speed can be calculated. The torque and speed information becomes known inputs to the vehicle system. The DP optimization is also governed by the powertrain dynamics,

$$g_x(k+1) = \begin{cases} 4, & g_x(k) + shift(k) > 4 \\ 1, & g_x(k) + shift(k) < 1 \\ g_x(k) + shift(k), & \text{otherwise} \end{cases} \quad (2)$$

$$SOC(k+1) = SOC(k) + f(SOC(k), \tau_m(k), \omega_m(k)) \quad (3)$$

where  $g_x$  is the gear number, the control,  $shift$ , to the transmission can be  $-1$ ,  $0$ , or  $1$ , representing downshift, hold and up-shift, respectively, and  $\tau_m$ ,  $\omega_m$  are the torque and speed of the motor, respectively. In addition, the following inequality constraints are also imposed to ensure safe operation of the engine/battery/motor,

$$\begin{aligned} \omega_{e\_min} &\leq \omega_e(k) \leq \omega_{e\_max} \\ \tau_{e\_min}(\omega_e(k)) &\leq \tau_e(k) \leq \tau_{e\_max}(\omega_e(k)) \\ \tau_{m\_min}(\omega_m(k), SOC(k)) &\leq \tau_m(k) \leq \tau_{m\_max}(\omega_m(k), SOC(k)) \\ SOC_{min} &\leq SOC(k) \leq SOC_{max} \end{aligned} \quad (4)$$

where  $\omega_e$  is the engine speed, and  $\tau_e$  is the engine torque.

A powerful algorithm to solve the above optimization problem is the dynamic programming technique. DP has the advantage of finding the true optimality within the accuracy of computational grids [14.]. However, the computation efficiency of DP is low due to the “curse of dimensionality”. Several techniques including pre-computed look-up tables and vectorized operation are adopted to accelerate the computation speed in the MATLAB environment [15.].

To study the trade-off between fuel economy and emissions, the weighting factors in equation (1) are varied as follows:  $\mu \in \{0, 5, 10, 20, 40\}$  and  $\nu \in \{0, 100, 200, 400, 600, 800, 1000\}$ . The case  $\mu = \nu = 0$  corresponds to the optimal fuel economy scenario. The weighting factor,  $\alpha = 5 \cdot 10^6$ , is used to assure the terminal battery SOC constraint is satisfied at the end of the cycle. Detailed optimization results were presented in [9.]. It was found that significant reduction in NOx and PM emissions can be achieved at the price of a small increase in fuel consumption. Hence, the case  $\mu = 40, \nu = 800$  is chosen, which achieves a reduction of NOx and PM by 17.3 % and 10.3% respectively, at a 3.67% increase on fuel consumption compared to the  $\mu = \nu = 0$  case. The fuel economy and emission results from DP are shown in

Table 2. The “new control” results are from the sub-optimal algorithm to be described in the next sub-section.

Table 2: Results over the UDDSHDV cycle [9.]

	FE (mi/gal)	NOx (g/mi)	PM (g/mi)	Performance Measure*
Baseline Control	13.11	5.770	0.460	843.96
New Control	12.81	4.866	0.435	793.16
DP ( $\mu = 40, \nu = 800$ )	13.24	4.642	0.399	739.56

\* Performance Measure:  $fuel + 40 \cdot NOx + 800 \cdot PM$  (g/mi)

## 2.2 Development of Sub-Optimal Rule-Based Control

Although the Dynamic Programming approach provides an optimal solution, the resulting control policy is not implementable under real-time driving conditions because it requires the knowledge of future speed and load profile, and is computationally heavy. Its behavior, on the other hand, is a good benchmark that other control strategies can be compared to or learn from. Therefore, the second step of the HEV control design procedure involves knowledge extraction from DP results

to obtain implementable rule-based control algorithms. Overall, the behaviors to learn include the transmission gear-shift strategy, the power-split strategy, and the charge-sustaining strategy. Here we assume that the regenerative braking strategy is simple—use as much regenerative braking as possible, subject to the current/power limit of the generator/battery. If the driver demands a harder deceleration than the regenerative braking can provide, the difference will be supplied by the friction brake. This simple regenerative braking strategy assumes that the vehicle handling stability is not an issue.

The transmission gear-shift strategy was found to be crucial for the fuel economy of hybrid electric vehicles [16.]. From DP results, the gear operational points are plotted on a slightly modified transmission shift-map (Figure 1). It can be seen that the gear positions are largely separated into four regions and the boundary between adjacent regions represents optimal gear shifting thresholds. A DP-inspired gear-shift map was obtained after a hysteresis function is added to the shifting lines.

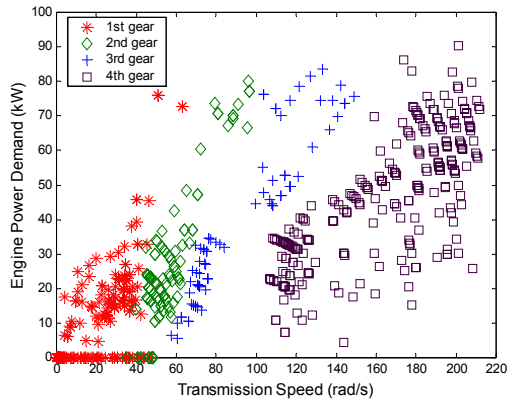


Figure 1: Gear operating points of DP optimization.

A power-split-ratio,  $PSR = P_{eng} / P_{dem}$ , is defined to quantify the optimal power split of the hybrid powertrain used by the DP algorithm. The optimal (DP) behavior was found to closely follow a simple curve when we plot the optimal PSR values versus the power request over the transmission input speed, which is equivalent to torque demand at the torque converter output shaft (see Figure 2). This figure shows that the optimal policy uses the recharging mode ( $PSR > 1$ ) in the low torque region, the engine-only mode ( $PSR = 1$ ) in the middle torque region, and the power-assist mode ( $PSR < 1$ ) in the high torque region. A least-square curve fit is then used to approximate the optimal PSR, shown as the solid line in Figure 2.

It should be noted that the power split control scheme described above can not assure the battery SOC will operate within a desired operating range. A charge-sustaining strategy should be developed to maintain the battery energy. More aggressive rules of spending battery energy can be used when SOC is high and more conservative rules can be used when SOC is low. These adaptive PSR rules can be learned from the DP policy by specifying different initial SOC points in the simulation [9].

The above new gear shifting control, power split control and charge-sustaining strategy are incorporated to construct a vehicle-level rule-based control strategy. This improved rule-based controller is evaluated using the original UDDSHDV cycle. A linear SOC correction procedure is used to calculate fuel economy and emissions [17., 18.]. The simulation results are shown in

Table 2. It can be seen that the new rule-based control system improves the combined fuel and emission performance (the “performance measure”) over the original, intuition driven rule-based control law [12.] and is only slightly worse than the performance of the DP result which is optimal for the UDDSHDV cycle. The improved rule-based control is obtained from the optimization result over one specific driving cycle. It may not perform satisfactorily under other driving scenarios. This motivates the multi-mode control study to be presented in the next section.

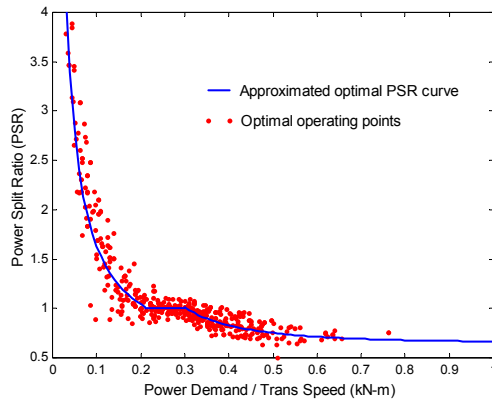


Figure 2: DP power split behavior (UDDSHDV cycle).

### 3. DRIVING PATTERN RECOGNITION (DPR)

#### 3.1 Multi-Mode Driving Control

The basic idea of a multi-mode controller based on Driving Pattern Recognition (DPR) technique was addressed in a previous paper [10.]. In a nutshell, this control concept assumes that we can use several Representative Driving Patterns (RDP) as basic templates, to represent all driving conditions. The switching will be determined by a DPR algorithm, which chooses one of the RDP to be the best approximation of current driving situation. This overall control algorithm assumes that the driving condition within a finite history window will continue into the near future.

Figure 3 shows the concept of the multi-mode driving control, where “ $T$ ” is the sampling time step for measuring vehicle input signals and generating control commands. Usually, the time steps for measurement and control are different from each other, but in this figure, they are assumed to be the same one to simplify the problem. The “ $pT$ ” is the duration (e.g. 150 seconds) of historical driving pattern which is sent to a buffer for DPR process. “ $fT$ ” is the duration (e.g. 1000 seconds) of RDP, “ $NT$ ” is the duration of control horizon (e.g. 5 seconds). The on-line procedures of the multi-mode driving control are described as follows.

First, characteristic parameters in the historical window ‘ $pT$ ’ are extracted, based on which the driving pattern over this historical window will be determined (see section 3.2). Next, the control algorithm will be switched to the sub-optimal control algorithm corresponding to newly identified RDP. Finally, the control actions will continue for the next  $NT$  seconds. The fact  $NT \ll fT$  (i.e., the control horizon is much shorter than the time duration used to predict the best RDP) is similar to that used in predictive controls or other receding horizon control algorithms.



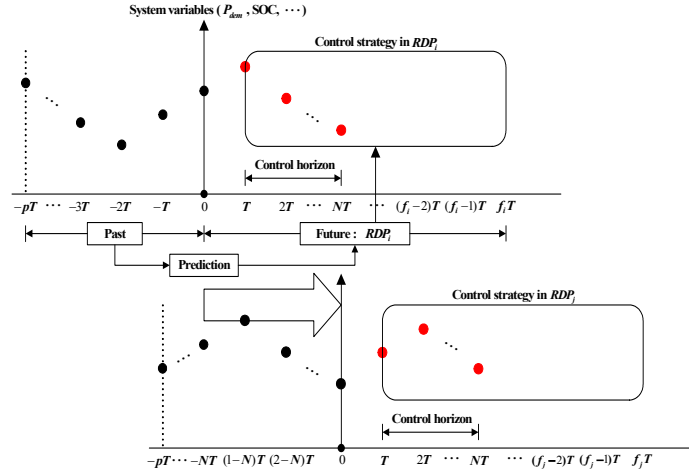


Figure 3: Concept of multi-mode driving control based on DPR.

### 3.2 Characteristic Parameters for DPR

To identify proper characteristic parameters for DPR, we pre-selected four candidates: the averaged positive power demand,  $P_{dem\_mean}$ , the standard deviation of positive power demand during driving,  $P_{dem\_std}$ , the averaged negative power demand,  $P_{dem\_neg\_mean}$  and the ratio of stop time against total driving time  $Stop\ time/Total\ time$ . These four parameters were selected because they are easy to compute, and that they have close correlation with the fundamental vehicle operation. We then investigate the possibility to reduce the number of characteristic parameters, so that the final DPR algorithm could be as simple as possible.

The last two parameters ( $P_{dem\_neg\_mean}$  and  $Stop\ time/Total\ time$ ) are good measures of the amount of regenerative braking energy and traffic congestion. Interestingly, these two parameters were found to have a moderate correlation with  $P_{dem\_mean}$  (see Figure 4 and Figure 5). These figures were obtained by simulating a hybrid passenger-vehicle over diverse driving cycles in ADVISOR [17.], that is it is possible to neglect the last two cycle parameters if  $P_{dem\_mean}$  is properly used in the DPR algorithm. Furthermore, the deceleration and stop situations that  $P_{dem\_neg\_mean}$  and  $Stop\ time/Total\ time$  represent belong to “passive” situations, which are not directly related to the control variables presented in Section 2, including the power-split ratio between engine and motor, and the sub-optimal gear ratio. Therefore, only two characteristic parameters were selected for the final design:  $P_{dem\_mean}$  and  $P_{dem\_std}$ .

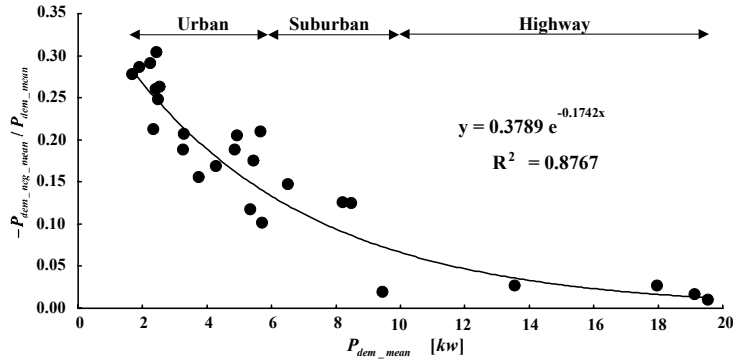


Figure 4: Correlation between  $P_{dem\_mean}$  and  $-P_{dem\_neg\_mean} / P_{dem\_mean}$ .

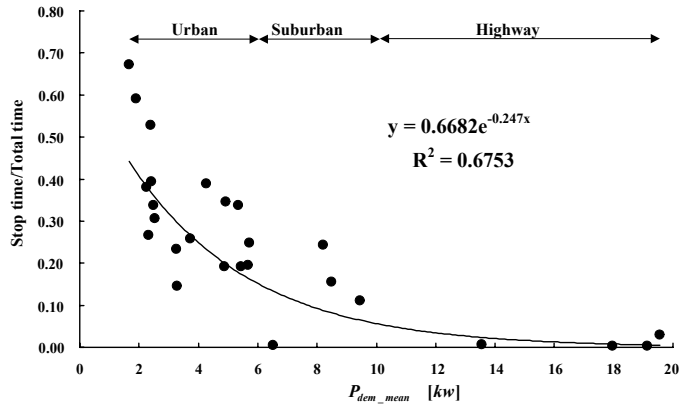


Figure 5: Correlation between  $P_{dem\_mean}$  and Stop time/Total time.

### 3.3 Selection of Representative Driving Patterns (RDP)

The next task is to choose RDP templates, which could be selected from commonly used driving cycles [10., 19.], or constructed from simple mathematical operations. We choose the second approach because of the fact that the characteristic parameters can be easily scaled.

Figure 6 presents the concept of creating the imaginary RDPs satisfying desired  $P_{dem\_mean}$  and  $P_{dem\_std}$ , where  $V_0$  is the initial vehicle speed,  $V_{i_i}$  ( $i=1, 2, 3, \dots$ ) are candidates of vehicle speed after  $\Delta T$ , and  $P_0$  and  $P_{i_i}$  ( $i=1, 2, 3, \dots$ ) are power

demands corresponding to  $V_0$  and  $V_{i1}$ . Four rules used in the RDP creation process are described below.

**Rule 1:** Once the desired  $P_{dem\_mean\_des}$  and  $P_{dem\_std\_des}$  are determined, the RDP creation process should proceed to achieve these desired values.

**Rule 2:** The sign of power demand is randomly selected. However, once the sign is selected, it is mandatory to continue accelerating or decelerating for a minimum time duration which is randomly selected among pre-defined candidates of minimum times.

**Rule 3:** In the case of acceleration ( $P > 0$ ), if the initial vehicle speed is given, the engine and motor speeds are calculated in a backward way by using a simplified gear shifting map depending on the vehicle speed. Subsequently, the maximum engine and motor powers ( $P_{e\_max}$  and  $P_{m\_max}$ ) corresponding to the given vehicle speeds are also determined. The new power demand after  $\Delta T$  is selected from grid points between 0 and  $P_{max}$ . In this process, the desired  $P_{dem\_mean\_des}$  and  $P_{dem\_std\_des}$  provide a guideline for this selection. For example, in Figure 6, if  $P_{14}$  is the closest grid point to  $P_{dem\_mean\_des}$  and if  $P_{dem\_std\_des}$  is small, one point within Region 1 ( $P_{14}$  and neighboring points in a small region) is randomly selected. If  $P_{dem\_std\_des}$  is large, one point within Region 2 ( $P_{14}$  and neighboring points in a larger region) will be selected.

**Rule 4:** In the case of deceleration ( $P < 0$ ), one grid point between  $P_{min}$  and 0 is randomly chosen.

Table 3 summarizes six RDPs created from the rules illustrated above. It is assumed that we only need to differentiate among a small number of RDPs. The six RDPs are classified into the low power demand RDPs (RDP1 and RDP2), the medium power demand RDPs (RDP3 and RDP4), and the high power demand RDPs (RDP5 and RDP6). Each class includes two RDPs with different standard deviations in power demand. For example, RDP1 (see Figure 7) represents typical urban driving patterns where the average power level is low but the variation in power is large due to frequent stop-and-go traffic conditions. RDP6 (see Figure 8), on the other hand, resembles suburban driving patterns where the average power level is high and the standard deviation in power is relatively small.

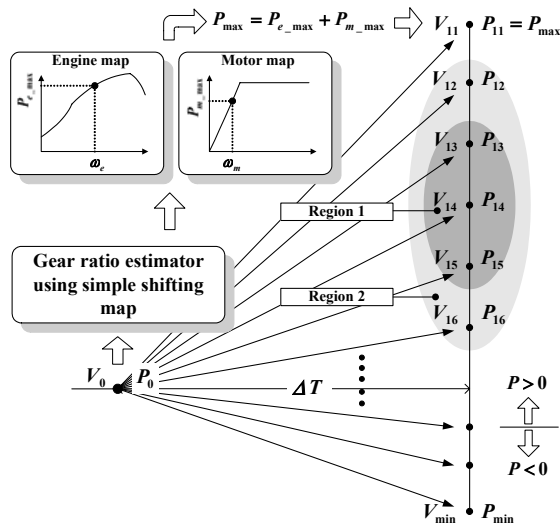


Figure 6: Process for defining imaginary RDPs.

Table 3: Six imaginary RDPs.

# of RDP	$P_{dem\_mean}$ [kW]	$P_{dem\_std}$ [kW]
1	33.1	29.3 (H)
2	32.3	12.1 (L)
3	54.8	38.0 (H)
4	49.5	21.1 (L)
5	71.4	40.0 (H)
6	70.2	26.7 (L)

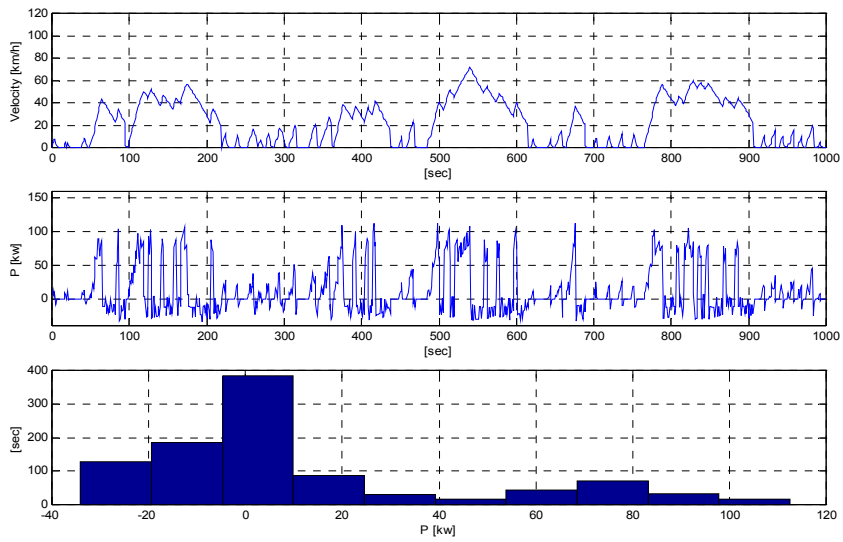


Figure 7: RDP 1

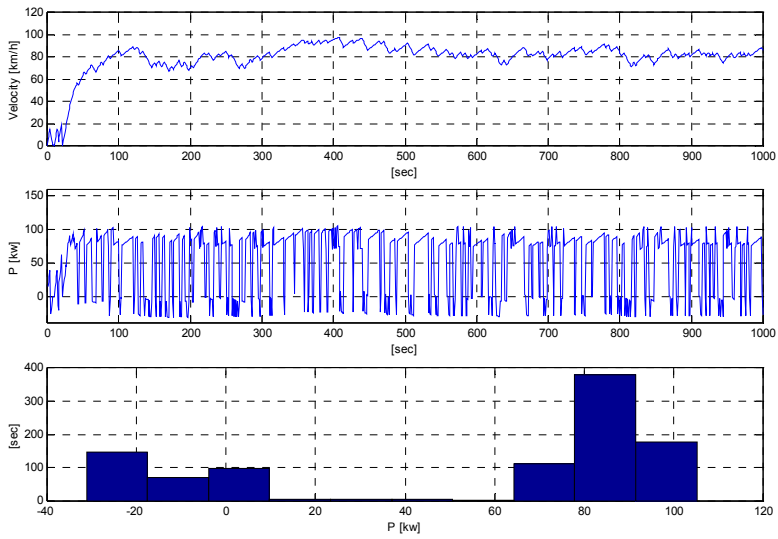


Figure 8: RDP6

#### 4. SIMULATION RESULTS

For each Representative Driving Pattern, the dynamic programming design procedure illustrated in Section 2 is applied to derive an implementable, sub-optimal controller. Since there are six RDPs, six respective control rules ( $C_1$ ~ $C_6$ ) are extracted. These six control rules are extracted off-line, and stored in the multi-mode control module for the on-line implementation. The on-line implementation of the multi-mode control strategy is illustrated in Figure 9. First, the historical values of the driver power demand are stored in a First-In-First-Out (FIFO) buffer. The buffer size is assumed to have  $pT$  seconds and the sampling time ( $T$ ) is chosen to be one second. As a result, the buffer always provides data in the most recent  $pT$ -second for driving pattern recognition. Every  $NT$  seconds, the DPR processor takes the stored power demand values in the buffer to calculate the necessary characteristic parameters (i.e., mean and standard deviation). The calculated parameters are then used to classify the current driving pattern into one of the six RDPs by the following algorithm.

$$RDP_i = \arg \min_i \sqrt{\left(P_{dem\_mean} - P_{dem\_mean,i}\right)^2 + \left(P_{dem\_std} - P_{dem\_std,i}\right)^2}, \quad i = 1, 2, \dots, 6 \quad (5)$$

where  $P_{dem\_mean,i}$  is the averaged positive power demand of RDP $_i$  and  $P_{dem\_std,i}$  is the standard deviation of positive power demand of RDP $_i$ . In general, a weighting factor could be added in Equation (5), e.g., to weigh the error in power standard deviation more than that of mean power. We chose not to add this weighting factor for simpler demonstration of the concept and the results. According to the classified RDP, the multi-mode control module switches to the corresponding control algorithm for the next  $NT$  seconds.

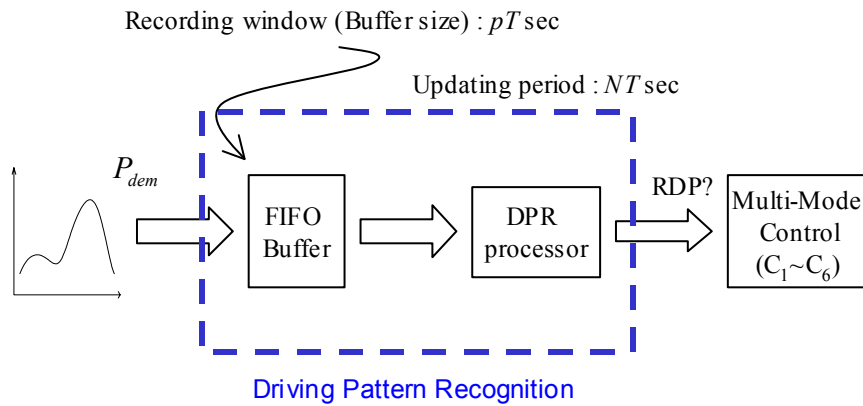


Figure 9: Diagram of multi-mode control strategy

The multi-mode control strategy developed above is evaluated by using the hybrid vehicle simulation model implemented in Simulink environment [9.]. The effectiveness of the multi-mode control is verified through driving cycles that have not been used in the selection of RDPs or experienced by the DPR algorithm. The six evaluation driving cycles were selected from the cycle library in the ADVISOR software [17.]. These cycles include city, suburban and highway driving. Furthermore, many of these cycles were designed for cars/light trucks and are somewhat aggressive for the medium truck we have. Note that none of these six cycles have exactly the same mean power or RMS power used in constructing the RDPs. They represent real driving a vehicle might experience.

The overall performance measure ( $FC + 40 \cdot NO_x + 800 \cdot PM$ ) of the multi-mode control is compared with that of the single-mode control which uses a sub-optimal control rule extracted only from the UDDSHDV cycle. Table 4 and Table 5 show the simulation results of the single-mode and multi-mode controller. For comparison purposes, the results achieved by the DP algorithm are also presented as the benchmark (performance limit). Note that the DP results are idealized (peek into future), and do not represent achievable performance. In these simulations,  $T=1$ sec,  $p=150$  and  $N=5$ . It can be seen that the multi-mode control achieves better results over most of the test cycles compared with the single-mode controller. It has worse performance only in UDDSHDV, the cycle that is the basis of the single-mode controller. It is important to point out that tuning of  $pT$  and  $NT$  is important for better results. Figure 10 shows the classification history when UDDSHDV cycle is tested. It can be seen that all modes were used, except mode '5', which is the high power, high standard deviation cycle.

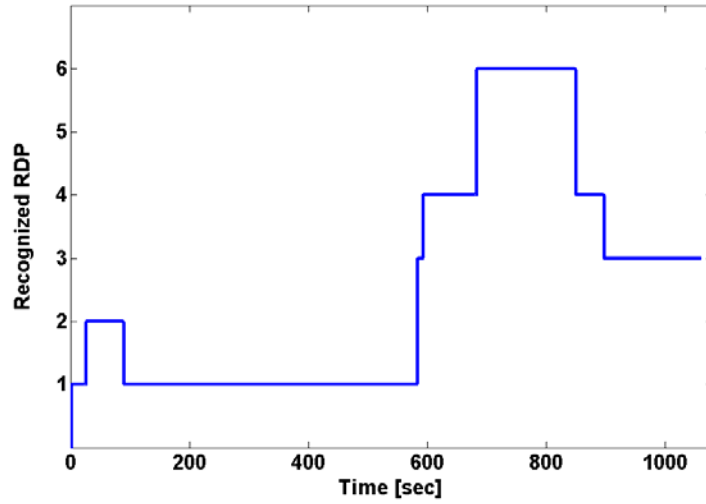


Figure 10: Results of driving pattern recognition over UDDSHDV

Table 4: Simulation results of multi-mode control ( $fuel + 40 \cdot NOx + 800 \cdot PM$  [g/mile]).

	UDDSHDV	WVUCITY	WVUSUB	WVUINTER
Single-mode	793.16	494.12	582.18	896.00
Multi-mode	801.64	468.38	576.39	897.22
DP	739.56	403.58	526.67	847.67

Table 5: Simulation results of multi-mode control ( $fuel + 40 \cdot NOx + 800 \cdot PM$  [g/mile]).

	NYCCOMP	NYCTRUCK	Manhattan
Single-mode	401.17	667.70	786.74



Multi-mode	381.89	659.63	771.00
DP	312.14	551.91	592.23

---

## 5. CONCLUSIONS

A multi-mode control strategy based on the driving pattern recognition scheme was developed for a hybrid electric truck to minimize fuel consumption and engine-out emissions over various driving scenarios. Six representative driving patterns with defined characteristics were selected to represent different driving modes. For each representative driving pattern, dynamic programming technique was utilized to determine the optimal power split and gear shift trajectory. Implementable, sub-optimal control algorithms associated with each representative driving pattern was then extracted by analyzing those optimal control actions. This design methodology by learning from the dynamic programming results has the clear advantage of being near-optimal, accommodating multiple objectives, and systematic.

A driving pattern recognition algorithm was developed, which used historical data (150 sec) in average and standard deviation of vehicle propulsion power to determine which representative driving pattern is closest to the current driving pattern. The sub-optimal control strategy for the identified mode was then used for a short future horizon (5 sec). The performance of the multi-mode control was evaluated by using six evaluation cycles. It was found that the multi-mode control achieves significant performance improvement in almost all the cycles we tested.

## ACKNOWLEDGEMENT

The work of Soonil Jeon and Jang Moo Lee at Seoul National University, Korea was supported in part by a grant from the BK-21 Program. The work of Chan-Chiao Lin and Huei Peng at the University of Michigan is supported by the U.S. Army TARDEC under the contract DAAE07-98-C-R-L008.

## REFERENCES

- 1.. US Department of Energy, Office of Energy Efficiency and Renewable Energy: Technology Roadmap for the 21st Century Truck Program. December, 2000.

- 2.. US Environmental Protection Agency: Final Emission Standards for 2004 and Later Model Year Highway Heavy-duty Vehicles and Engines. EPA420-F-00-026, 2000.
- 3.. Wipke, K., Cuddy, M., and Burch, S.: ADVISOR 2.1: a user-friendly advanced powertrain simulation using a combined backward/forward approach. *IEEE Transaction on Vehicular Technology* 48(6) (1999), pp. 1751-1761.
- 4.. Baumann, B.M., Washington, G.N., Glenn, B.C., and Rizzoni, G.: Mechatronic Design and Control of Hybrid Electric Vehicles. *IEEE/ASME Transactions on Mechatronics* 5(1) (2000), pp. 58-72.
- 5.. Rahman, Z., Butler, K., and Ehsani, M.: A Comparison Study Between Two Parallel hybrid Control Concepts. *SAE Paper*, No. 2000-01-0994, 2000.
- 6.. Schouten, N., Salman, M., Kheir, N.: Fuzzy Logic Control for Parallel Hybrid Vehicles. *IEEE Transactions on Control Systems Technology* 10(3) (2002), pp.460-468.
- 7.. Bowles, P.D.: Modeling and Energy Management for a Parallel Hybrid Electric Vehicle (PHEV) with Continuously Variable Transmission (CVT). MS thesis, University of Michigan, Ann Arbor, MI, 1999
- 8.. Piccolo, A., Ippolito, L., Galdi, V., and Vaccaro, A.: Optimization of Energy Flow Management in Hybrid Electric Vehicles via Genetic Algorithms. *Proceedings of 2001 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, Como, Italy, July, 2001.
- 9.. Lin, C.C., Peng, H., Grizzle, J.W., and Kang, J.: Power Management Strategy for a Parallel Hybrid Electric Truck. *IEEE Transactions on Control Systems Technology* 11(6) (2003), pp. 839-849.
- 10.. Jeon, S.I., Jo, S.T., Park, Y.I., and Lee, J.M.: Multi -Mode Driving Control of a Parallel Hybrid Electric Vehicle Using Driving Pattern Recognition. *ASME Journal of Dynamic Systems, Measurement, and Control*, 124(1) (2002), pp. 141~149.
- 11.. Andre, M.: Driving Cycles Development: Characterization of the Methods. *SAE Paper*, No. 961112, 1996.
- 12.. Lin, C.C., Filipi, Z.S., Wang, Y., Louca, L.S., Peng, H., Assanis, D.N., and Stein, J.L.: Integrated, Feed-Forward Hybrid Electric Vehicle Simulation in SIMULINK and its Use for Power Management Studies. *SAE Paper* No. 2001-01-1334, 2001.
- 13.. Assanis, D.N., Filipi, Z.S., Gravante, S., Grohnke, D., Gui, X., Louca, L.S., Rideout, G.D., Stein, J.L. , and Wang, Y.: Validation and Use of SIMULINK Integrated, High Fidelity, Engine-In-Vehicle Simulation of the International Class VI Truck. *SAE Paper* No. 2000-01-0288, 2000.
- 14.. Bertsekas, D.P.: *Dynamic Programming and Optimal Control*. Athena Scientific, Belmont, MA, 1995.
- 15.. Kang, J., Kolmanovsky, I., and Grizzle, J.W.: Dynamic Optimization of Lean Burn Engine Aftertreatment. *ASME Journal of Dynamic Systems, Measurement and Controls*, 123(2) (2001), pp. 153-160.
- 16.. Lee, H.D., Sul, S.K., Cho, H.S., and Lee, J.M.: Advanced Gear Shifting and Clutching Strategy for a Parallel Hybrid Vehicle. *IEEE Industry Applications Magazine*, November/December, 2000.
- 17.. National Renewable Energy Laboratory: *Advanced Vehicle Simulator ADVISOR, ver 3.2*, 2001.
- 18.. Mckain, D.L., Clark, N.N., Balon, T.H., Moynihan, P.J., Lynch, P.J., and Webb, T.C.: Characterization of Emissions from Hybrid-Electric and Conventional Transit Buses. *SAE Paper* No. 2000-01-2011, 2000.
- 19.. Engstrom, J., and Victor, T.: Real-time Recognition of Large-scale Driving Patterns. *Proceedings of 2001 IEEE Intelligent Transportation Systems Conference*, Oakland, CA, August, 2001.

## APPENDIX

Table 6: Nomenclature

Symbol	Meaning
$g_x$	gear number
$P$	power (W)
$SOC$	Battery state of charge
$\tau$	Torque (N-m)
$\omega$	rotational speed (rad/s)
$\alpha$	weighting factor on final SOC
$shift$	gear shifting command
$\mu$	weighting factor on NOx
$v$	weighting factor on PM
<b>Subscripts</b>	
$e$	engine
$m$	motor
$x$	transmission
<b>Abbreviations</b>	
DP	Dynamic Programming
DPR	Driving Pattern Recognition
HEV	Hybrid Electric Vehicle
PSR	Power Split Ratio
RDP	Representative Driving Pattern
SOC	State of Charge