*EVS27*
*Barcelona, Spain, November 17-20, 2013*

# Using Trip Information for PHEV Fuel Consumption Minimization

Dominik Karbowski,[1] Vivien Smis-Michel, [2] Valentin Vermeulen[2]
[1]*Argonne National Laboratory, USA, dkarbowski@anl.gov*
[2]*Mines-ParisTech, France*

## Abstract

*When driven past their all-electric range, plug-in hybrid vehicles (PHEVs) must use their engines. Numerous theoretical studies showed that the conventional control strategy, i.e. all-electric mode followed by a charge-sustaining mode, is not the most energy-efficient control strategy. Better strategies require knowledge of the trip ahead. In this paper, we present a method of predicting a trip for a given itinerary (vehicle speed, stop time, and grade) defined by using a geographical information system (GIS). For each segment of the itinerary, a vehicle speed profile is generated through a Markov process, defined by transition probabilities extracted from a large database of real-world trip records. Ten trip predictions are then generated from a single itinerary for evaluation of an optimal control strategy for a short-range power-split PHEV by using Autonomie, a powertrain modeling environment. The baseline controller uses rules and optimal operating point look-up tables when in charge-sustaining mode. The optimal controller uses the Pontryagin's Minimization Principle (PMP), the performance of which heavily depends on the choice of one scalar parameter, the equivalence factor. Finally, we demonstrate the fuel-saving potential of the PMP controller, using the aforementioned trip predictions.*

*Keywords: PHEV, control, optimization, GPS, prediction*

## 1 Introduction

Detailed maps of the road network, increased on-board computing capabilities, connectivity to cloud-based computing resources, and ever-increasing inclusion of global navigation satellite systems (GNSS) make route prediction more and more conceivable. One of the main applications of route prediction is energy efficiency: knowledge of future driving conditions, if used effectively, can contribute to improving the efficiency of advanced vehicles, such as hybrid electric vehicles (HEVs) and plug-in HEVs (PHEV).

Several optimization techniques that use some form of route prediction as an input already exist. The most theoretical is dynamic programming [1], which provides a global optimum. It is highly computer-intensive and is hardly implementable because of the very nature of the algorithm, which runs backwards (i.e., starting from the end). Stochastic dynamic programming [2] uses a probabilistic distribution of drive cycles, rather than a single cycle. Another technique is mixed-integer linear programming [3].

An alternative and easier to implement online optimization technique relies on the Pontryagin's Minimization Principle (PMP) [4][5], which under certain assumptions can be simplified to an Equivalent Consumption Minimization Strategy (ECMS) method [6][7][8]. In this case, global optimality is not guaranteed, but it generally leads to good results. However, the outcome of using this method highly depends on one constant, the initial co-state in one case, or equivalence factor in the other, that is chosen for the online implementation. Finding the optimal factor for a given trip can be done by predicting the route.

Many studies assume the speed profile is given and do not explore two major hurdles of real-world implementation of trip-based control optimization: (1) how to predict the vehicle speed profile for the trip and (2) how well the optimization works when the predicted and the actual speed differ – which invariably can happen in the real world, given drivers unpredictability.

One approach to trip prediction is to model vehicle speed as a Markov Chain [9][10]. This model relies on a database of real-world vehicle speeds, from which transition probabilities can be computed. This allows the generation of stochastic speed profiles. However, this method *per se* does not provide a prediction for a particular itinerary. Wu *et al.*[3] proposed predicting the speed for a particular itinerary by combining macroscopic average traffic speeds with a disturbance generated by EPA's MOVES.

We presented in [11] a method to generate a speed profile for a given itinerary based on information provided by a geographical information system (GIS), ADAS-RP, published by HERE (a NOKIA company). The prediction assumed the trip was made in sections of constant speed, constant acceleration, or constant deceleration. This process is available in the public version of Autonomie (version R12) [12], an automotive systems modeling environment developed by Argonne National Laboratory.

In this study, we present a method of obtaining a stochastic vehicle speed profile for a given itinerary using real geographical data and then demonstrate how this could be used for PHEV optimization by using a PMP-based controller. Our guiding principle is to accurately represent the causalities and uncertainties of the real world, from the powertrain chosen to the flow of information.

# 2 Trip Prediction

## 2.1 Markov Chain Generation Under Constraints

A Markov chain is a random process characterized as memory-less: the next state only depends on the current state and not on the sequence of past events. This type of mathematical model is good for representing vehicle speed. Ivanco ([9]) and later T.-K. Lee ([10]) used speed and acceleration as the states of the process, leading to positive results. We chose to use the same state definition in this work.

The transition from one state to another is governed by a transition probability, which only depends on those two states. Their collection forms the transition probability matrix (TPM), which can be built by processing all the data points of a real-world trip database. In our case, we used data from the 2007 Chicago Metropolitan Agency for Planning (CMAP) database [13] of approximately 6,000,000 data points that were filtered, processed, and quality-checked before being used to build the TPM.

One fundamental aspect of the "classic" Markov chain is that the outcome is stochastic, and the only control over the result is the time at which we stop the Markov chain generation. If the itinerary is given – for example the driver selects the destination on his navigation unit – there is also a deterministic aspect to speed prediction: there are stops, speed limits and historical average speeds at spatially defined points of the trips. There will be of course stochastic variations of speed around those determined conditions. To combine those two aspects, we created an algorithm which consists of generating stochastic speed profiles until a result with characteristics "close" enough to the deterministic prediction emerges. Information about the trip, or the "target" (such as distance, average speed) ahead can be provided by a GIS. This process is illustrated in Figure 1.

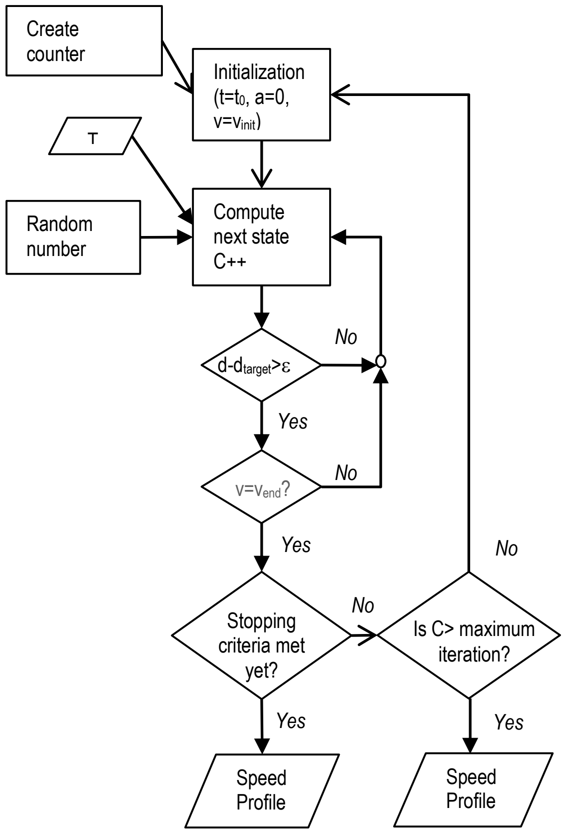


Figure 1: Stochastic vehicle speed generation under constraints

In a first loop, the Markov chain generation is stopped when the current distance is higher than, or close to the target distance and the speed is close to the target final speed (or equal to it if a stop is requested at the end of the segment). Once the candidate stochastic speed profile is generated, we check whether it satisfies a stopping criterion that depends on the target trip. If it does not, the algorithm computes a new vehicle speed profile.
The stopping criterion considers average speed, number of stops, excessive speed, and distance. It is given by the Performance Value (PV):

$$PV = w_1 \frac{|V_{avg} - V_{tgt}|}{V_{target}} + w_2 \frac{N_{stop}}{d_{seg}} + w_3 \sum_{t=T_1 \dots T_2} \max((V(t) - V_{lim}), 0)^2 + w_4 \frac{|d_{seg} - d_{tgt}|}{d_{tgt}} \quad (1)$$

where

- $(w_1, w_2, w_3, w_4)$ are constants;
- $V_{avg}$, $N_{stop}$, $d$ and $V$ are explanatory variables for the generated speed profile: $V_{avg}$ is the average speed, $N_{stop}$ is the number of stops, $d$ is the distance, and $V(t)$ is the speed at time *t*; and
- $V_{tgt}$, $V_{lim}$, $d_{tgt}$ are the constraints: $V_{tgt}$ is the target average speed, $V_{lim}$is the speed limit, and $d_{tgt}$ is the desired distance of the section.

This PV measures the capability of the generated speed profile to fit some constraints corresponding to the target trip: the speed average must be close to the traffic speed, the vehicle should avoid stopping for no reason (although we still allow unplanned stops), speed should not be higher than the speed limit, and the distance of the trip must be very close to the target distance.
Once the loop is exited and there is a speed profile that matches the stopping criteria, the synthesized speed profile does not match the target distance exactly. We use an algorithm that "stretches" or "shrinks" the vehicle speed profile accordingly, mainly by adding or removing bits of constant speed segments.
Figure 2 shows three vehicle speed traces for the same short target segment. Because of the stochastic nature of the method, no two synthesized speed traces are the same.

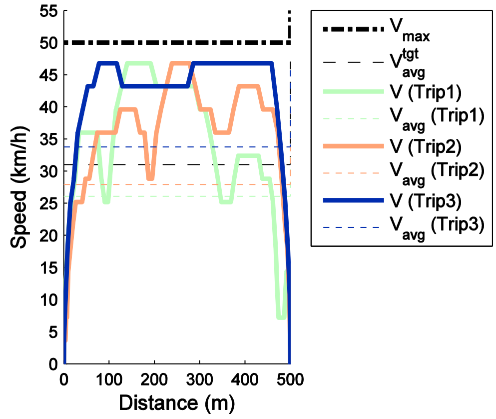


Figure 2: Three vehicle speed profiles synthesized by a Markov Chain under the same constraints

## 2.2 Speed profile generation using a GIS

The intent of this project was to generate a speed profile for an entire itinerary that could be tens of kilometers long. Along such a target itinerary, there are numerous stops and changes in average speed, speed limits, and other variables. Using the algorithm over the entire trip would not likely lead to a prediction that would fit the target well. Hence, the idea was to segment the target itinerary, and iteratively apply the algorithm to each of the segments. We also needed a way to obtain information about the trip.
To that end, we leveraged the capabilities presented in [11]. A process exists to define the itinerary in ADAS-RP (a GIS developed by HERE), process and format the data, segment the trip, and generate segments of constant speed or acceleration. The entire process is available in one of the public versions of Autonomie [12]. We re-used the itinerary acquisition and trip segmentation part of that tool.
Figure 3 shows a schematic view of the algorithm. The geographical data of the itinerary contain a large number of attributes that may or may not be available for each of the links. These raw data are therefore processed into three main outputs:

- Grade as a function of distance,
- Stop position and schedule, and
- Segmentation in segments of the same speed limit and average speed.

The grade does not need further processing, whereas the stop schedule and the segmentation are used in an iterative loop to generate stochastic vehicle speed profiles on segments until the final distance is reached.

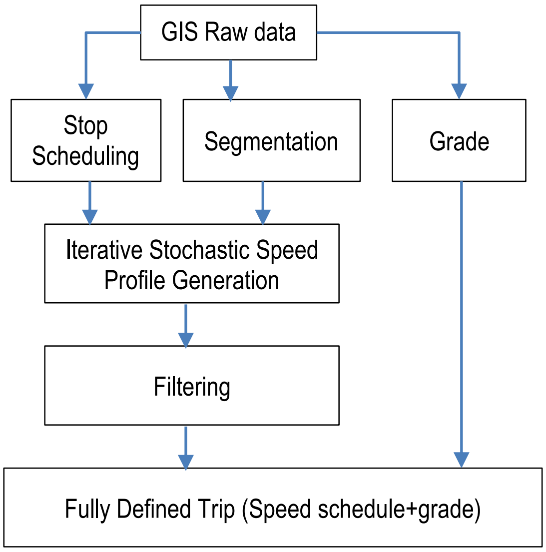


Figure 3: Schematic view of the trip generation process

During the iterative speed profile generation, several measures are applied to ensure convergence and continuity of the vehicle speed profile between segments:

- *Initial speed definition:* The initial speed for each segment is set to the final speed reached at the previous segment.
- *Final speed computation:* If there is a stop at the end of the segment, the final target speed is zero. Otherwise, we compute final speed bounds that take into account average target speed and speed limit for the current and next segment.
- *Adjustment of average speed:* We adjust the target average speed to take into account the possible scheduled stop time at the beginning, during, and at the end of the segment. We also make sure that the target average speed is realistic, given the speed limit and the initial and final speeds.

Finally, the speed profile is filtered to remove quantization—in generating a Markov chain, the speed variable is composed of discrete values (from 0 to 38 m/s, with a step of 1 m/s). This rule will ensure that we end up with a realistic speed profile, suitable for fuel consumption simulation. Figure 4 shows an example of an entire trip generated by this process, along with the targets for each segment.

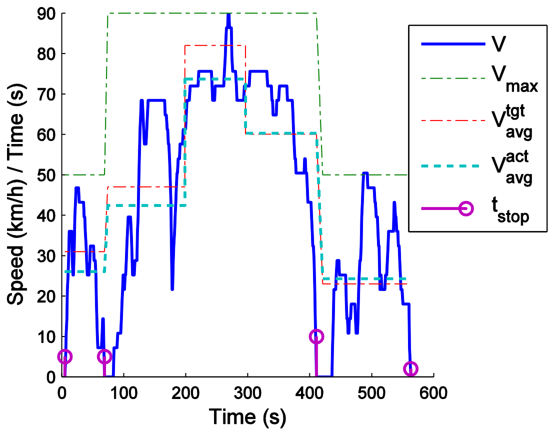


Figure 4: Target speeds/stops and synthesized trip

## 2.3 Trips Used for Study

To evaluate the optimal route-based control described in section 4, we selected an itinerary in ADAS-RP. The origin was BMW World in Munich, Germany, and the destination is Possenhofen, 36 km to the southwest. The speed limit was overwritten and set to 100 km/h, to allow EV operations at each point of trip. Figure 5 shows the trip on a map, and Figure 6 presents its characteristics.


Figure 5: Itinerary for a trip from Munich to Possenhofen
Source: http://here.com/

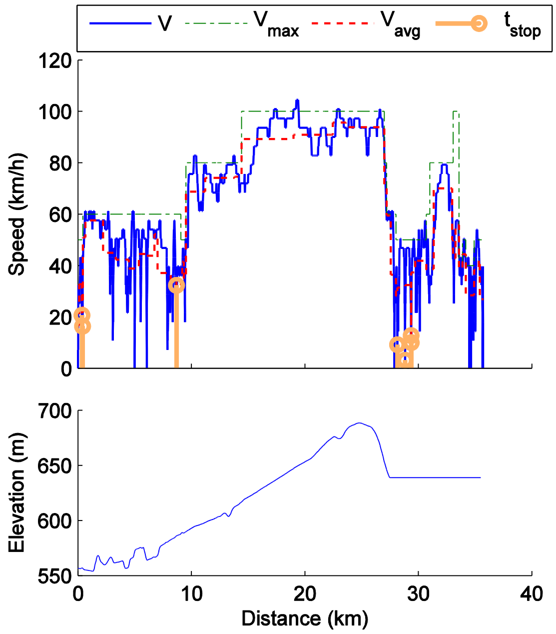


Figure 6: Characteristics of the Munich-Possenhofen itinerary, as well as one possible speed prediction

A total of 10 stochastic trips were generated for the chosen itinerary. The resulting travel time was between 39 and 42 minutes.

# 3 Baseline Vehicle Model

## 3.1 Baseline Vehicle Definition

A PHEV with a medium all-electric range (AER) is probably the best candidate to demonstrate route-based control. It is more likely to be driven past its AER, and therefore more likely to require engine use. Among the existing or planned vehicles, the Toyota Prius PHEV corresponds to such a profile.

The vehicle powertrain model follows the known specifications of the actual 2012 Prius PHEV. It has a 200-V, 21-Ah Li-ion battery with 168 cells. The top speed in all-electric mode is 100 km/h. The AER of the vehicle is 26 km using the JC08 test cycle, according to Toyota (23 km for our model). Since many key specifications were not available, we made assumptions, which explains the differences in the AER.

We modeled the vehicle in Autonomie (Figure 7). It is important to note that this is a forward-looking model and replicates the causality of the real world. In particular, the vehicle controller in the model uses the same type of inputs as those available from sensors in a vehicle.

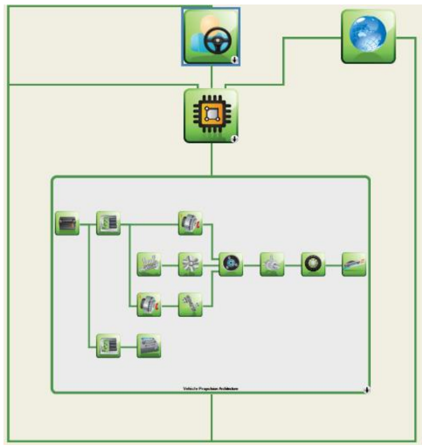

Figure 7: One-mode power-split PHEV model in Autonomie

## 3.2 Optimal Operating Lines

The Prius powertrain is a one-mode, power-split architecture. Thanks to two electric machines and a planetary gear set, the engine speed can be decoupled from the vehicle speed, thus introducing an additional degree of freedom. For a given engine and vehicle speed, the speeds of the electric machines are defined. Likewise, steady-state, electric machine torques are also defined if the gearbox output torque and engine torque are known. Engine torque and speed are therefore the natural control variables. There are two potential sources of losses in this system: operating the engine outside its efficiency area and energy recirculation (by which one electric machine generates current that is totally or partially used by the other, with the balance being taken to or from the battery). Minimizing losses can be done by using optimum operation maps. In our case, we made the battery output power a control variable as it is generally used for state of charge (SOC) control. The gearbox output speed and torque are givens at each time step – the latter comes from interpreting driver demand. An algorithm was developed to generate the optimal efficiency point for a given trio of gearbox output speed, gearbox output torque, and battery power. The two resulting 3-input look-up tables (one for engine speed, the other for engine torque) can then be used in the online controller. Figure 8 illustrates a subset of the optimal speed look-up table.

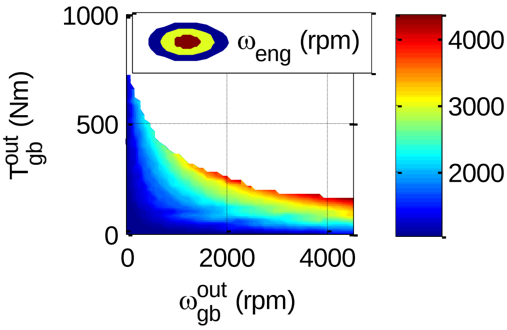


Figure 8: Optimal engine speed for a battery power of -10 kW

### 3.3 Baseline Controller

The baseline control strategy is to run the vehicle in EV mode as much as possible, at which point the vehicle starts operating in charge-sustaining (CS) mode, like a conventional HEV.
To provide a fair comparison between the optimal case and the reference case, we must ensure that the reference case is not poorly implemented. One way to approach this goal is to limit the differences between baseline and optimal controller. Therefore, we developed a baseline controller in which the high-level energy management aspect is clearly separated from other functions of the controller (such as engine speed control), which allowed us to easily swap strategies while keeping the lower level functions the same. Figure 9 shows the top level view of the controller in Simulink. Using the nomenclature in Figure 9, Block 3 is related to high-level energy management. As such, it is the only block that changes between the reference and baseline controller.

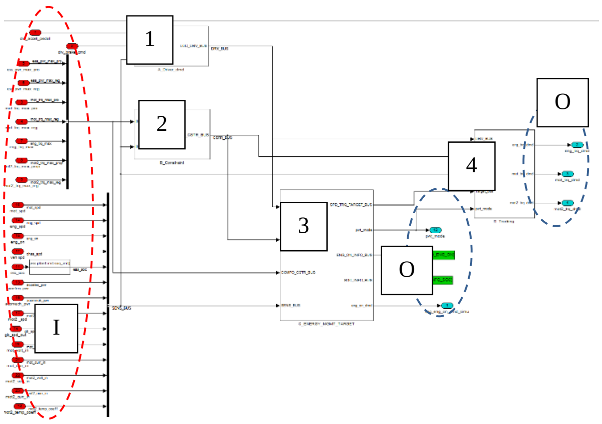


Figure 9: View of the vehicle-level controller with Inputs (I), Outputs (O), Driver Interpretation (1), Combined Constraints Computation (2), High-Level Energy Management/Target Generation (3), and Target Tracking (4)

The baseline control strategy is to run the vehicle in EV mode until the SOC reaches the discharge level, unless the vehicle speed is too high (above 100 km/h) or the power demand exceeds what the components can provide. Once a low SOC level is reached, the vehicle operates in charge-sustaining (CS) mode, like a conventional HEV.
In CS mode, the high-level energy management strategy follows commonly used rules. The engine is turned on (respectively shut down) when the driver power demand is higher (respectively lower) than an SOC-dependent threshold. When the engine is on, battery power demand is computed from an SOC-dependent look-up table: the battery is charged below the target SOC and discharged above the target SOC. The optimal operating point maps described previously are then used to generate the speed and torque targets. This strategy is schematically presented in Figure 10.

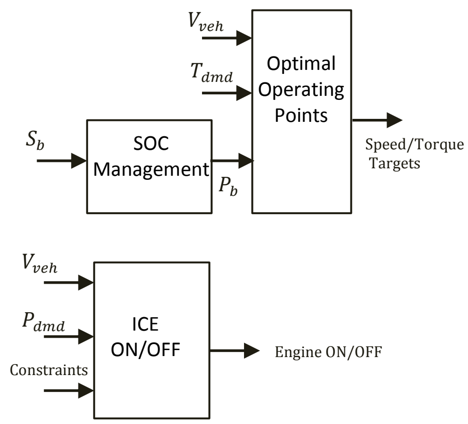


Figure 10: Schematic view of baseline high-level energy management

# 4 Optimal Control Using the Pontryagin's Minimization Principle (PMP)

## 4.1 Optimization Problem and PMP Theory

The goal of this project is to optimize a PHEV so that it uses less fuel energy than the baseline version. The vehicle must meet the driver demands, so we assume that vehicle speed and gearbox output torque are givens. The battery state of charge $S_b$ is the state of the vehicle. Since fixing the battery power $P_b$ is enough to compute an optimal operating point, we consider $P_b$ as the command variable. By using the look-up tables described previously, we can link the fuel power to the battery power such that $P_f = g(P_b, S_b, t)$. The battery SOC is linked to the battery power by the following dynamic equation:

$$\dot{S_b} = -\frac{P_b}{QV_n}\ \frac{2\frac{V_n}{V_{oc}}}{1+\sqrt{1-\frac{4P_bR}{V_{oc}^2}}} \quad (2)$$

where $Q$ is the battery capacity, $V_n$ is the battery nominal voltage, $V_{oc}$ is the open-circuit voltage, and $R$ is its internal resistance.

In equation (3), we introduce $\lambda_0$, a negative constant, and $\theta$, a scalar function of $P_b$ and $S_b$ (via $V_{oc}$ and $R$) close to 1 in the operating range of the system.

$$\lambda_0 = -\frac{1}{QV_n} \ \ ; \ \ \theta = \frac{2\frac{V_n}{V_{oc}}}{1+\sqrt{1-\frac{4P_bR}{V_{oc}^2}}} \quad (3)$$

As a result, equation (2) is equivalent to equation (4):

$$\dot{S_b} = \lambda_0\theta(P_b, S_b)P_b \quad (4)$$

Therefore, the optimization problem consists of finding successive optimal battery power demands that will minimize the fuel energy while reaching the target SOC $S_{tgt}$ at the end of the trip:

$$P_b^* = \underset{\substack{P_b \\ S_{b(T)}=S_{tgt}}}{\operatorname{argmin}} \left(\int_0^T P_f(P_b, S_b, t)dt\right) \quad (5)$$

The Hamiltonian $H$ of the system is:

$$H = P_f + p(t)\dot{S_b} \quad (6)$$

where $p$ is the co-state. The PMP states that a necessary condition of optimality is that the optimum command $P_b^*$ minimizes the Hamiltonian (equation (7)) with the co-state verifying equation (8), as well as boundary conditions, especially regarding SOC.

$$P_b^* = \underset{P_b}{\operatorname{argmin}}(\,H(P_b, S_b, p, t)) \quad (7)$$

$$\dot{p} = -p(t)\frac{\partial \dot{S_b}}{\partial S_b} \quad (8)$$

In this study, the characteristics of the battery ($V_{oc}$ and $R$) do not vary much as a function of the SOC. As a result, we will assume the co-state to be constant: $p(t) = p_0$. We also introduce in (9) the equivalence factor.

$$r_0 = \lambda_0 p_0 \quad (9)$$

This allows to rewrite (6) into (10), which then gives a physical interpretation of the Hamiltonian: it is the equivalent power used by the system at any given time.

$$H = P_f + r_0\theta(P_b, S_b)P_b \quad (10)$$

The practical implementation of the PMP requires to find the battery power that minimizes the Hamiltonian at each time step. However, there is another problem to solve, which is to find $r_0$ such that the final SOC is the target SOC (30%).

## 4.2 PMP Implementation

The PMP optimal controller is derived from the baseline controller. Only the high-level energy management is different. The other three blocks are the same. The PMP is implemented only for the charge-depleting mode (i.e., until the battery reaches the low SOC threshold used in the baseline control). Once that happens, the control is the same CS control as in the baseline control.

Figure 11 shows a simplified view of the high- level management block. It is a practical implementation of equation (7) and deals with the choice of operating point and the decision to turn the engine on or leave it off.

The first step is to compute the Hamiltonian for both the EV and engine-on modes. In EV mode, the computation is straightforward as there is only one way of controlling the vehicle in that mode. In engine-on mode, the computation of the Hamiltonian first relies on computing it for a vector of battery power demands and resulting fuel powers. The lowest Hamiltonian and the corresponding power demand are then selected.

The ON/OFF decision is based on the relative difference of the respective Hamiltonians in the EV and engine-on mode: in the ON/OFF logic block, we select the mode with the lowest Hamiltonian. Some filters are implemented to prevent an excessive number of state changes.

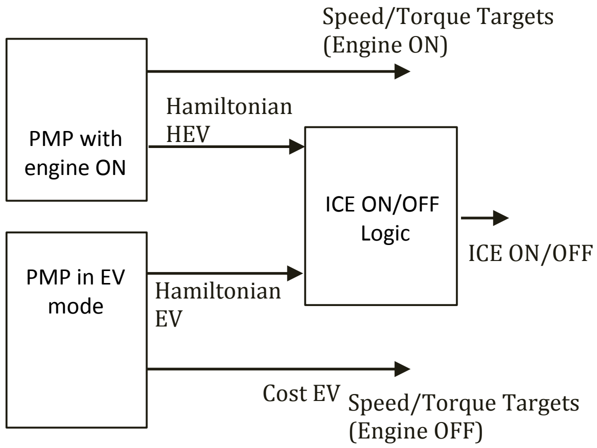


Figure 11: Schematic of the high-level energy management in the PMP controller

# 5 Benchmarking the Optimal Control by Using a Real-World Itinerary

## 5.1 Design of Experiments

Ten trips were synthesized by using the stochastic vehicle speed profile generation process described in section **Error! Reference source not found.**. All ten trips are based on the itinerary defined in section 2.3. The reference vehicle and the one with the PMP controller were run on each of those trips. For the vehicle with the PMP controller, equivalence factors of 2.795 to 2.83 were used, with a 0.005 step. We also assumed that the goal was to minimize fuel use (no fuel/electricity trade-off was sought), and that the battery was fully charged at the beginning of the trip.

## 5.2 Operation with Optimal Choice of Equivalence Factor

In this section, we assume that the equivalence factor is optimally chosen. As shown in Figure 12, this means that the equivalence factors may be different from one trip to another.

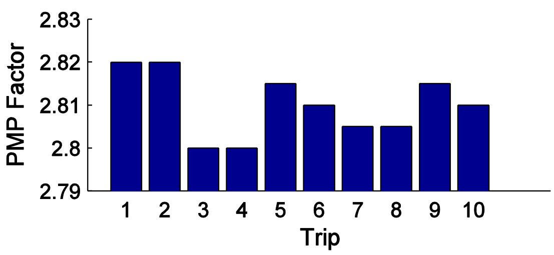


Figure 12: Optimal equivalence factor for each trip

The fuel consumption during each cycle with optimal equivalence factors is shown in Figure 13. There is ~10% difference between the most fuel intensive trip (#7) and the least intensive trip (#6) because of the stochastic nature of the process used to generate them. We observe that the PMP controller leads to fuel savings in all cases and reduces the consumption by as much as 5.8% (Trip #1) and by 4.6% on average.

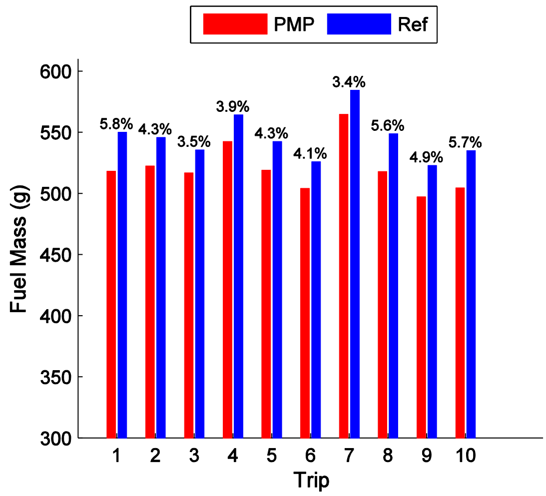


Figure 13: Mass of fuel consumed on each trip for the optimal PMP case and reference case; percentages are fuel savings in the PMP case compared to the reference case

Figure 14 shows several key descriptive parameters for both the reference case and the PMP case (see caption for detailed description). The engine is used less with the PMP controller, both in terms of energy (Parameter 1) or time (Parameter 4), and slightly more efficiently at that (Parameter 2). Parameter 5 is intended to give a sense of the level of recirculation, which occurs when part of the engine output is converted into electricity by motor 2 (or generator) and converted back to mechanical power by motor 1. Recirculation is a source of inefficiency, but it is at times necessary to preserve engine efficiency and system efficiency as a whole. It appears here that recirculation is lower with the PMP controller, which means more engine energy goes directly to the wheels. Better efficiency is gained thanks to more frequent engine state changes: the number of engine starts (Parameter 3) almost tripled.

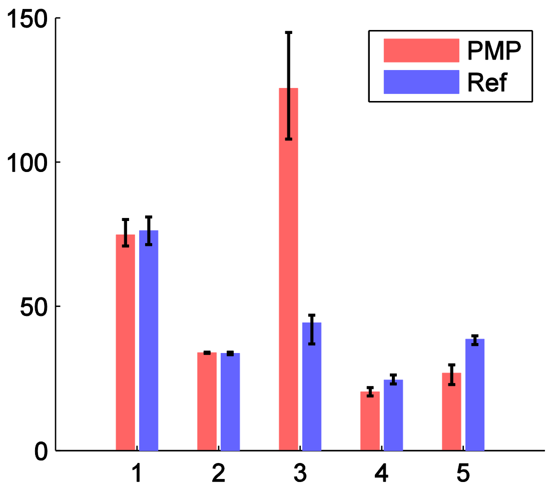


Figure 14: Key descriptive parameters for the PMP and reference case (error bars represent variability on trip):
1= Total engine mechanical energy (MJ)(x10)
2 = Engine efficiency (%)
3 = Number of engine starts
4= Share of time engine is ON (%)
5 = Ratio of motor2 energy over engine energy (counted when the engine is operating)

Figure 15 shows how the energy is used throughout one trip (trip 10, equivalence factor of 2.81).

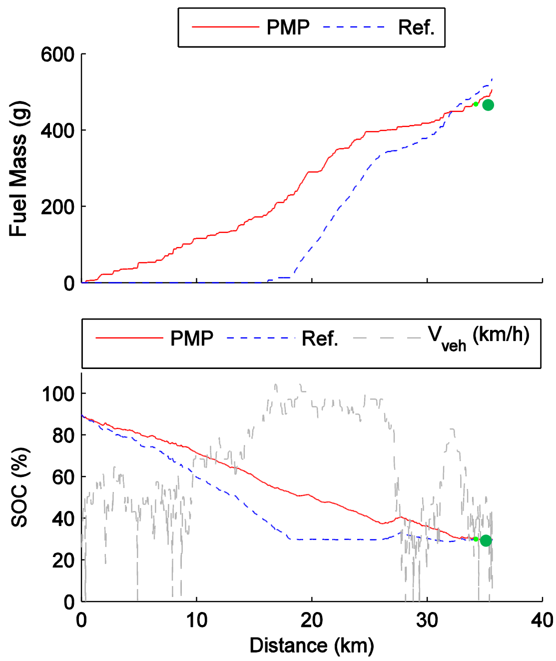


Figure 15: Operations of reference and optimized vehicles on trip 10, equivalence factor of 2.81

All through the trip, the engine and the battery are used in conjunction with the PMP controller, whereas the engine stays off during the first 18 km in the reference case (at which point it switches to CS mode). With the PMP controller, the CS mode is reached at 34.2 km (highlighted in Figure 15 by a green dot), about 2 km from the end of the trip.

### 5.3 Sensitivity of PMP Controller to Equivalence Factor

In a potential real-world setting, the optimal co-state cannot be precisely known ahead of time because the predicted and actual speed will always differ. The results shown in the previous sections are therefore best-case scenarios. A more realistic scenario would be to obtain an equivalence factor based on a set of trip predictions and apply it to a different set, thus modeling the prediction/actuality discrepancy.
We do not propose a prediction method here; however, the analysis of whether the PMP controller will still bring benefits with sub-optimal equivalence factors will provide valuable guidance. Figure 16 shows the outcome of simulations over three different trips using the entire sweep of equivalence factor values, along with the reference case (in red).

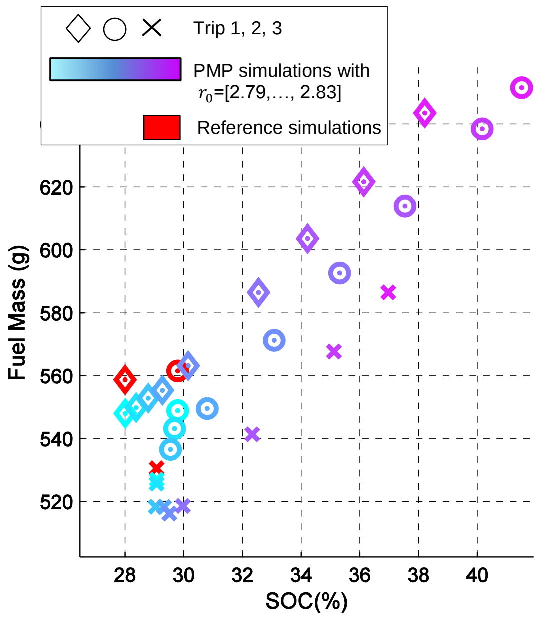


Figure 16: Final fuel mass and SOC for three different trips (different shapes), PMP different equivalence factors (cyan to purple colors), and Reference control (red).

If the equivalence factor is too high, battery energy is too expensive, and as a result the final SOC is too high: there is battery energy left at the end of the trip that could have been used to displace fuel consumption. In that case, the performance of the PMP controller vehicle is actually worse.

On the other hand, if the equivalence factor is too low, the battery will be discharged prematurely, and the fuel consumption will approach the reference case. As a result, the benefits will diminish. However, it is unlikely that it will do worse than the reference strategy.

Figure 17 provides another way to look at these results and shows fuel savings for all trips and equivalence factors. We adjusted the fuel mass value to account for differences in final SOC between the PMP and reference case for those PMP runs in which the CS mode was reached. We also included the average savings for any given equivalence factor, which peaks at 3.3%, for an equivalence factor of 2.805. Another curve shows the average over the best eight savings, which provides a slightly better result of 4.3%, for an equivalence factor of 2.81.

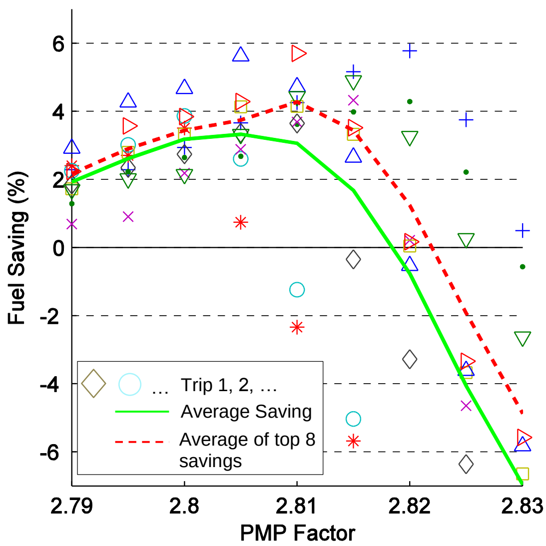


Figure 17: Fuel savings as a function of Equivalence factor for all 10 trips

Note that there is a high sensitivity to the equivalence factor for certain trips. For example, an equivalence factor of 2.81 can bring 5.7% savings on one trip while increasing the consumption by over 2% on another one. This suggests that the "one-size-fits-all" equivalence factor approach may not be the best for yielding the highest results; an adaptive algorithm could resolve this issue.

# 6 Conclusion

## 6.1 Summary

Trip prediction offers great opportunities for improving the energy efficiency of PHEVs, but it is also greatly challenging. In theory, while significant fuel savings are to be expected, few solutions have been proposed for real-world implementation.

To fully demonstrate the potential of this new data-driven technology, one must address three key questions:

- How can the future be predicted? Or more exactly, how can a usable prediction of future speed and grade be proposed?
- How can the prediction be used to improve energy efficiency?
- How can the benefits of this simulated technology be evaluated? Or, how can we adequately model the uncertainties and inaccuracies associated with prediction?

We believe the work presented in this paper tackles all three questions and proposes ideas that could eventually answer them and be implemented in the cars of tomorrow.

Our proposed tool for prediction combines deterministic and stochastic aspects. It is closely related to the chosen itinerary in terms of road topography, positions of stops, speed limits, and traffic speeds, among other parameters, by gathering information for that itinerary for that particular trip. The stochastic nature comes from synthesizing each segment with a Markov chain defined by probabilities extracted from a database of real-world trip values.

We also demonstrated an implementation of the much-researched PMP in a vehicle-level controller for a forward-looking powertrain model. Besides using inputs and outputs similar to those available in the real world, it also takes into account the main dynamic aspects of control and addresses some driveability issues (e.g., unrealistic and excessive engine state changes). Furthermore, we deployed best efforts to ensure the baseline controller using the "EV+CS" strategy is itself efficient, to limit the effect of a "bad" reference control on the results. The PMP controller is definitely an attractive approach and yields positive results in our limited experiments, but its sensitivity to the choice of the equivalence factor can be a hurdle for real-world benefits, and further research is needed.

Finally, we addressed the question of evaluating the impact of the discrepancies between predicted and actual speed profiles in an indirect way by showing how an optimum parameter for one prediction works on another prediction for the same itinerary.

### 6.2 Future Work

Future work will continue to address the three questions of prediction, use of prediction in control, and accurate evaluation of its benefits.
We will work on improving the robustness and speed of our algorithm, possibly leading to its integration in future releases of Autonomie. We will also explore ways to better match transition probabilities with particular segments from the GIS.
On the control side, we will focus on fast equivalence factor prediction, as well as adaptive algorithms, so that we reduce the sensitivity to the equivalence factor. Better control and better prediction will help us further evaluate this promising technology.

## Acknowledgments

This work is funded by the U.S. Department of Energy (DOE) Vehicle Technology Office, and is under management of David Anderson.
We would like to acknowledge HERE (a NOKIA company) for helping in the use of their GIS platform, ADAS-RP.
The submitted manuscript has been created by UChicago Argonne, LLC, Operator of Argonne National Laboratory ("Argonne"). Argonne, a U.S. Department of Energy Office of Science laboratory, is operated under Contract No. DE-AC02-06CH11357. The U.S. Government retains for itself, and others acting on its behalf, a paid-up nonexclusive, irrevocable worldwide license in said article to reproduce, prepare derivative works, distribute copies to the public, and perform publicly and display publicly, by or on behalf of the Government.

## References

[1] D. Karbowski, A. Rousseau, et al., *Plug-in Vehicle Control Strategy: From Global Optimization to Real Time Application*, 22th International Electric Vehicle Symposium (EVS22), Yokohama, October 2006

[2] Moura, S.J., Fathy, H.K. et al., *A Stochastic Optimal Control Approach for Power Management in Plug-In Hybrid Electric Vehicles*, IEEE Transactions on Control Systems Technology, Volume 19, Issue 3, May 2011

[3] Guoyuan Wu, Kanok Boriboonsomsin et al., *Development and Evaluation of Intelligent Energy Management Strategies for Plug-in Hybrid Electric Vehicles*, 2012,University of California Transportation Center website, www.uctc.net/research/papers/UCTC-FR-2012-09.pdf, accessed on July 2012

[4] Namwook. Kim, S. Cha, and H. Peng, *Optimal Control of Hybrid Electric Vehicles Based on Pontryagin's Minimum Principle*, IEEE Transactions on Control Systems Technology, Vol. 19, Issue 5, pp. 1279 - 1287, Sept. 2011

[5] Daeheung Lee, Suk Won Cha et al., et al, *Optimal Control Strategy for PHEVs using Prediction of Future Driving Schedule*, EVS26, Los Angeles, May 6-9, 2012

[6] G. Paganelli, S. Delprat, et al., *Equivalent Consumption Minimization Strategy for Parallel Hybrid Powertrains*, 55th IEEE Vehicular Technology Conference, 2076–2081, 2002

[7] L. Serrao, S. Onori, G. Rizzoni, *ECMS as a realization of Pontryagin's minimum principle for HEV control*, American Control Conference - ACC , pp. 3964-3969, 2009

[8] Musardo, C., Rizzoni, G., Staccia, B., A-ECMS: *An Adaptive Algorithm for Hybrid Electric Vehicle Energy Management*, 44th IEEE Conference on Decision and Control, 2005

[9] Ivanco, A., Charlet, A., Chamaillard, Y., and Higelin, P., *Energy Management Strategies for Hybrid-Pneumatic Engine Studied on an Markov Chain Type Generated Driving Cycle*, SAE Technical Paper 2009-01-0145, 2009

[10] Tae-Kyung Lee, Zoran S. Filipi, *Representative Real-world Driving Cycles in Midwestern US*, Proceedings of Int. Scient. Conf. on hybrid and electric vehicles (RHEVE 2011), IFPEN, 2011

[11] Dominik Karbowski, Sylvain Pagerit, et al., *Energy Consumption Prediction of a Vehicle along a User-Specified Real-World Trip*, EVS26 Los Angeles, California, May 6-9, 2012

[12] *Autonomie*, www.autonomie.net, accessed on July 2013

[13] *Travel Tracker Survey*, www.cmap.illinois.gov/travel-tracker-survey, accessed on July 2013

## Authors

**Dominik Karbowski** is a principal research engineer at Argonne National Laboratory. Dominik leads research in vehicle control development and utilization of ITS to improve energy efficiency. Dominik has extensive experience modeling advanced powertrains such as hybrids or plug-in hybrids, both light- and heavy-duty, and contributes to the development of Autonomie. He holds a Master's degree in Science and Executive Engineering with a major in Energy Systems from Mines-ParisTech.

**Vivien Smis-Michel** is a graduate student at Mines-ParisTech, with a major in Applied Mathematics. Vivien worked as a research aide at Argonne, and helped link Markov chain generation with GIS.

**Valentin Vermeulen** is a graduate student at Mines-ParisTech, with a major in energy and mechanical engineering. As a research aide at Argonne, Valentin developed Markov chain generation algorithms.