Scientific Research Publishing

Journal of Transportation Technologies, 2022, 12, 804-832
https://www.scirp.org/journal/jtts
ISSN Online: 2160-0481
ISSN Print: 2160-0473

Powertrain Fuel Consumption Modeling and Benchmark Analysis of a Parallel P4 Hybrid Electric Vehicle Using Dynamic Programming

Aaron R. Mull, Andrew C. Nix, Mario G. Perhinschi, W. Scott Wayne, Jared A. Diethorn, Dawson E. Dunnuck

West Virginia University, Morgantown, USA
Email: armull@mix.wvu.edu, Andrew.Nix.mail.wvu.edu, Mario.Perhinschi@mail.wvu.edu, Scott.Wayne.mail.wvu.edu, jadiethorn@mail.wvu.edu, Dawson.Dunnuck@mail.wvu.edu

How to cite this paper: Mull, A.R., Nix, A.C., Perhinschi, M.G., Wayne, W.S., Diethorn, J.A. and Dunnuck, D.E. (2022) Powertrain Fuel Consumption Modeling and Benchmark Analysis of a Parallel P4 Hybrid Electric Vehicle Using Dynamic Programming. *Journal of Transportation Technologies*, **12**, 804-832.
https://doi.org/10.4236/jtts.2022.124045

Received: August 3, 2022
Accepted: October 11, 2022
Published: October 14, 2022

Abstract

The goal of this work is to develop a hybrid electric vehicle model that is suitable for use in a dynamic programming algorithm that provides the benchmark for optimal control of the hybrid powertrain. The benchmark analysis employs dynamic programming by backward induction to determine the globally optimal solution by solving the energy management problem starting at the final timestep and proceeding backwards in time. This method requires the development of a backwards facing model that propagates the wheel speed of the vehicle for the given drive cycle through the driveline components to determine the operating points of the powertrain. Although dynamic programming only searches the solution space within the feasible regions of operation, the benchmarking model must be solved for every admissible state at every timestep leading to strict requirements for runtime and memory. The backward facing model employs the quasi-static assumption of powertrain operation to reduce the fidelity of the model to accommodate these requirements. Verification and validation testing of the dynamic programming algorithm is conducted to ensure successful operation of the algorithm and to assess the validity of the determined control policy against a high-fidelity forward-facing vehicle model with a percent difference of fuel consumption of 1.2%. The benchmark analysis is conducted over multiple drive cycles to determine the optimal control policy that provides a benchmark for real-time algorithm development and determines control trends that can be used to improve existing algorithms. The optimal combined charge sustaining fuel economy of the vehicle is determined by the dynamic programming algorithm to be 32.99 MPG, a 52.6% increase over the stock 3.6 L 2019 Chevrolet Blazer.

Keywords

Hybrid Electric Vehicle, Dynamic Programming, Powertrain Modeling, Backwards Induction

1. Introduction

The objective of this research is to develop a hybrid electric vehicle (HEV) model that can be used to conduct a benchmark analysis by implementing a dynamic programming algorithm to identify the optimal control policy for a hybrid electric drivetrain on prescribed drive cycles. The results obtained from the benchmark analysis can be used to compare and improve existing hybrid supervisory control strategies by providing a frame of reference to the optimal performance of the hybrid drivetrain. Further, the methodology used in this analysis can be applied to a wide variety of HEVs.

Under conventional vehicle operation, the range of torques and speeds that correspond to the maximum efficiency of the ICE is narrow, and in most consumer automobiles, the ICE frequently operates outside of this envelope to honor the torque and power requests of the driver. The increasingly large number conventional vehicles used across the world have begun causing consequential problems for the environment and hydrocarbon resource supplies. Deteriorating air quality, global warming issues and depleting petroleum resources have forced regulatory entities to implement ever more strict emissions regulations for automotive manufacturers. Rising to the challenge of meeting these regulations, innovation in the automotive design field has influenced HEVs popularity more than ever across the world. HEVs utilize a combination of conventional vehicle components, such as an engine and transmission, and electric vehicle components, such as an electric motor and battery pack, to provide propulsive power to the wheels of the vehicle. By electrifying the powertrain, higher fuel efficiency and reduction in emissions can be achieved when compared to conventional vehicles [1].

When considering the control of the HEVs powertrain components, it is easy to assume that the problem is as simple as utilizing the electric motors as much as possible due to their high operating efficiencies. For HEVs equipped with charge depleting (CD) modes where the state of charge (SOC) of the energy storage system (ESS) starts at a high value and depletes as the vehicle is driven, this would be the case. However, from a charge sustaining (CS) point of view where the SOC is kept near a setpoint and within a low and high threshold, the control problem becomes far more complicated when attempting to achieve improved fuel economy over a conventional vehicle. An effective energy management strategy is essential to ensuring the efficient operation of the vehicle. Several families of energy management strategies have been investigated in existing literature. These strategies generally follow one of two trends, heuristic-based and mod-

el-based optimization methods [2]. Heuristic energy management strategies are primarily based on intuition and logical relationships between variables and thus little optimization occurs. Heuristic control is popular among automotive manufactures and is widely adopted in modern HEVs. Model-based optimization methods, also known as optimal control, make use of optimal control theory to derive the controller. Optimal control strategies are currently subject to research and are gradually being introduced in the industry [3]. Heuristic and optimal control strategies have two distinct subgroups as shown in **Figure 1**.

The most attractive characteristic of heuristic control methods is their effectiveness in real-time implementation. Since these strategies rely on predefined sets of rules and logic rather than minimization or optimization, heuristic controllers require little computational resources to make decisions. These types of controllers generally fall into two categories: rule-based and fuzzy logic [2]. Rule-based control is the traditional control methodology used in the automotive industry typically consisting of "if-then" and "switch" logic based on simulation data, intuition, or some other set of prescribed behavior based on constraints and conditions [4]. Energy management systems based on predefined rules have been widely researched and shown to be practical and successfully implemented to control hybrid powertrains [5] [6] [7] [8] [9]. In these works, the authors applied rule based strategies to a variety of powertrain configurations and found that a general set of rules could be developed to control HEVs, however, these rules required careful tuning to specific driving scenarios to achieve efficient performance. This major drawback is especially limiting in consumer applications due to the wide range of use cases.

The use of fuzzy set theory to control systems is referred to as fuzzy logic. Fuzzy control gives users the ability to implement expert knowledge through automation, provide robust nonlinear control and reduce development and maintenance time [10]. Many examples of fuzzy logic use in powertrain control are available in literature [11] [12] [13] [14] [15] that show improvements over

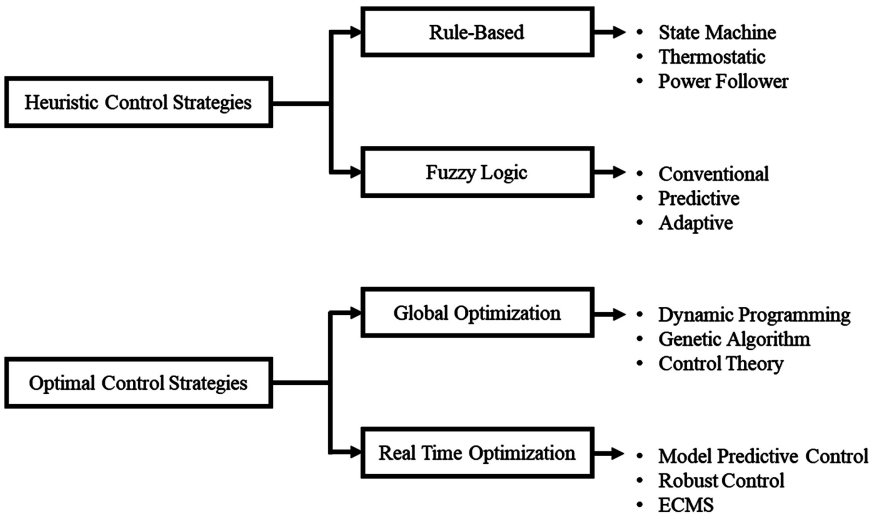

Figure 1. Control strategy classification [4].

traditional rule-based control strategies but remain limited in their ability to operate in an optimally efficient manor. This exhibits the fact that heuristic control strategies are sub-optimal as any optimization is conducted offline and used to design the governing statements of the controllers. Energy management of a HEV can also be posed as an optimization problem over a finite time horizon whose solution can be found from optimal control theory. The methods used are aimed at finding a control law for a given system such that a specific optimality criterion, usually defined as an integral performance index, is achieved [2]. In an optimal strategy, an appropriate cost function is created which is minimized at each timestep. There are two main areas of optimal control methods: real-time optimization and global optimization.

In real-time optimization, the cost function is minimized at each timestep of the online controller. In these controllers, simple mathematical models of the system are generally used to keep the execution time within a short time window. Potentially the most popular real-time optimal control strategy is Equivalent Consumption Minimization Strategy (ECMS) which operates on the premise that in a CS HEV the differences between initial and final SOC are small. This creates a specific cost for utilizing electrical stored energy and draws an equivalence between using a certain quantity fuel or stored electrical energy [2]. ECMS has been successfully implemented as an effective energy management strategy in literature [16] [17] [18] with authors boasting potential fuel consumption savings between 30% - 50% with the most significant savings occurring in urban driving scenarios. This optimization method is again limited by the requirement of careful tuning as well as limited computation capabilities onboard vehicles. Most of the ECMS work is still conducted in literature and has not yet been applied to consumer vehicles.

Alternatively, achieving a global optimal solution is generally directly correlated with highly complex and computationally expensive numerical solutions. One of the most popular methods for solving the optimal control problems for HEVs is Dynamic Programming (DP) which reduces a multi-step decision making problem into a series of single-step problems. These single-step problems may be solved either forward in time or backward from the last step to the first. The goal of DP is to minimize an incrementally increasing cost function at each step. DP offers dramatically reduced computation time compared to brute force methods of global optimization as it only searches over admissible state or control values. It is important to note however, DP still requires the storage of all valid state transition costs [4].

The results of the DP algorithm are used to determine appropriate control policies and the maximum efficiency a specific powertrain would be capable of achieving. DP algorithms typically require a highly simplified vehicle model due to the computational resources required to calculate the cost-to-go matrices throughout the simulation. DP is generally used to provide a benchmark to interpret the results of online heuristic controllers more appropriately. The bene-

fits of extracting data from DP to improve and assess online control strategies are well documented in literature [17] [19]-[24]. The authors of these works applied to a DP optimization to HEV control strategies, and found fuel consumption performance improvements of ranging from 27% - 70%.

In the following sections, the vehicle architecture applied in the EMC is discussed along with the formulation of the dynamic programming model. The results and conclusions of this study are then presented accompanied by future work suggestions. The final conclusions are based on results found in the study both from a fuel economy and computational perspective.

2. Vehicle Architecture

The vehicle modeled in this work was a P4 parallel HEV architecture that was selected for use in the West Virginia University EcoCAR Mobility Challenge (EMC) Advanced Vehicle Technology Competition (AVTC) as shown in **Figure 2**.

The performance metrics for the hybrid powertrain are shown in **Table 1**.

The performance metrics for the P4 architecture supports three operating modes: front wheel drive (FWD) with opportunity charging, FWD with regenerative braking, and all-wheel drive (AWD). In FWD with opportunity charging, the engine produces excess torque to the front axle while the P4 traction motor "drags" the rear axle by producing negative torque. When producing negative torque, the electric motor is spun thus generating power that is stored

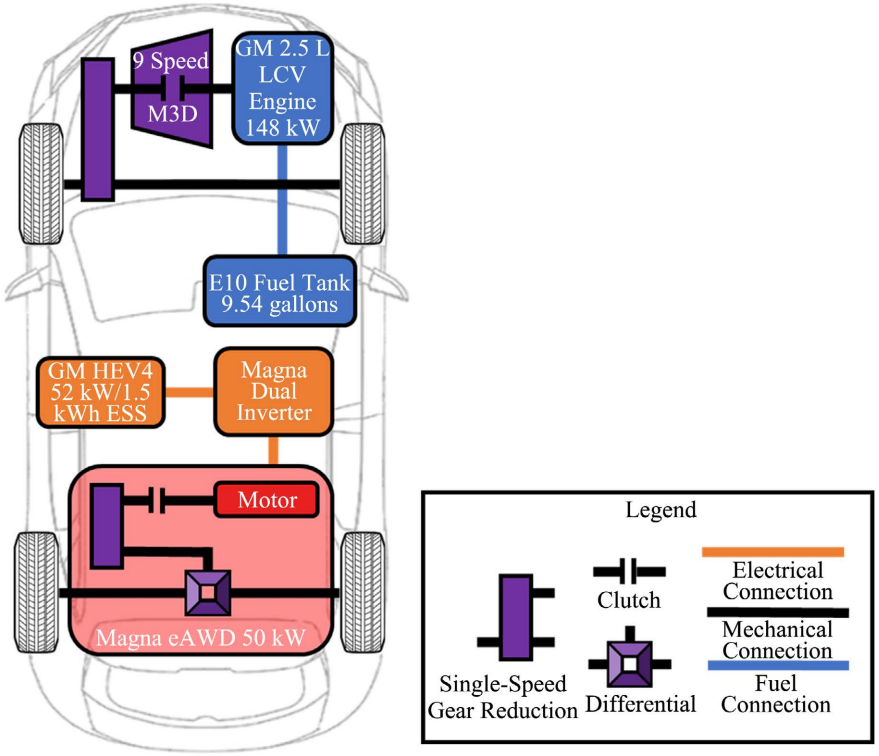

Figure 2. West Virginia university EcoCAR team competition vehicle architecture.

Table 1. West virginia university EcoCAR team competition vehicle architecture powertrain performance metrics.

Component	Specifications
Engine	• General Motors (GM) 2.5 L Naturally Aspirated LCV • Peak Power: 148 kW • Peak Torque: 255 Nm
Transmission	• GM M3D (9T50) 9-Speed Automatic
Fuel	• E10 Regular
Energy Storage System	• GM HEV4 • Peak Power: 50 kW • Energy Capacity: 1.5 kWh
Motor	• Magna Powertrain Electrified Rear Axle Drive (eRAD) • Peak Power: 50 kW • Peak Torque: 200 Nm • Integrated Gear Ratio: 9.17
Inverter	• Magna Dual Inverter

in the energy storage system (ESS). In FWD with regenerative braking, the engine is supplying all of the positive propulsive torque for the vehicle. In situations where the driver requests a deceleration event, the electric motor is used to produce negative torque to meet the braking needs of the vehicle, producing power that is stored in the ESS. In AWD both the electric motor and engine produce positive torque to meet the needs of the driver. Figure 2 illustrates the directional shift of the engine operating profile of torque and speed for both FWD with opportunity charging and AWD.

The operating mode of the vehicle can transition rapidly to meet the current driving conditions. Smooth transitions between these operating modes are a major consideration HEV controls development with respect to the ride quality of the vehicle.

In the WVU team competition vehicle, the engine (148 kW) is capable of producing nearly three times the power compared to the electric powertrain (50 kW). This limits the electric powertrain's ability to meet driver demands without the help of the engine. In addition, the selected ESS has a usable energy capacity of only 1.0 kWh with simulation results developed by the WVU EcoCAR team showing an all-electric range of only about 2 miles [25]. For these reasons, the team did not implement a charge depleting (CD) mode in the competition vehicle. Based on the available power from the electric powertrain, the team competition vehicle would operate exclusively in a CS mode with the electric motor augmenting the operation of the engine. Specifically, the electric powertrain would be used to shift the operating point of the engine to more efficient regions, as shown in Figure 3. It is also important to note that for fuel economy results of HEVs, the vehicle is evaluated on a known drive cycle with the starting and ending SOC within a specified bound. This is commonly referred to as CS fuel economy.

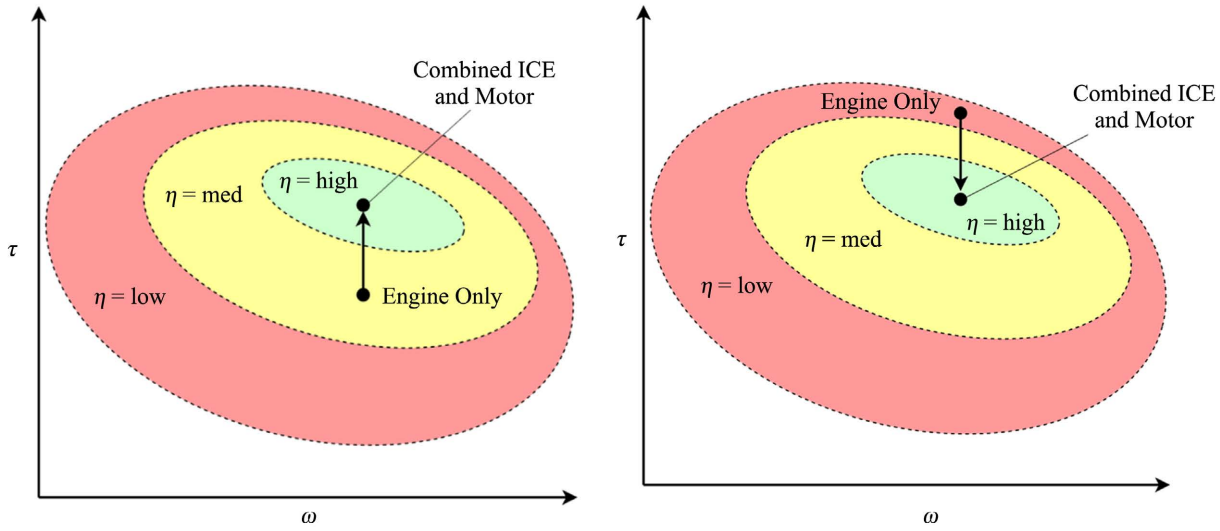

Figure 3. Example of engine operational point shifted with electric motor (left: FWD with opportunity charging; right: AWD).

3. Benchmark Analysis Methodology

3.1. The Basic Problem

Consider a discrete-time deterministic system, the states evolve over time as described by the transition function:

$$x_{k+1} = f\left(x_k, u_k\right), \left\{k = 0, \cdots, T\right\} \tag{1}$$

where x_k, the state variable at stage *k*, exists in a space S_k, and u_k, the control input at time *k* that modifies x_k, exists in space C_k.

The set of control inputs, termed a *policy*, consists of a sequence of functions:

$$\pi = \left\{u_0, u_1, \cdots, u_{T-1}\right\} \tag{2}$$

where each u_k is constrained to take values in a subset of C_k, depending on the current state x_k. The specific constraints applied are part of the formulation of a particular DP problem, and serve to eliminate infeasible control inputs. The set C_k is called an admissible policy.

Each transition of x_k between different values incurs a cost. The cost represents the effort of moving from one state to another and serves to differentiate the paths. In a literal path-search problem, the cost represents the distance required for each path step. The cost function describing the state transition and implementing constraints is given by $g_k\left(x_k, u_k\right)$. The cost of a given path is additive over time since it accumulates over each stage of the problem. The total cost of a path can be expressed as:

$$J\left(x_0\right) = g_T\left(x_T\right) + \sum_{k=0}^{T-1} g_k\left(x_k, u_k\right) \tag{3}$$

thus, the cost of following a given policy π starting from state x_0 is:

$$J_\pi\left(x_0\right) = g_T\left(x_T\right) + \sum_{k=0}^{T-1} g_k\left(x_k, \pi_k\right) \tag{4}$$

An optimal policy can be found, denoted π^*, that minimizes this cost, such that:

$$J_{\pi^*}(x_0) = \min\{J_\pi(x_0)\} \tag{5}$$

3.2. Bellman's Principal of Optimality

The cornerstone of DP is Bellman's Principal of Optimality, which states that "An optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision" [26].

This property suggests that if the following optimal policy:

$$\pi^* = \{u_0^*, u_1^*, \cdots, u_{T-1}\} \tag{6}$$

passes through the state x_i at time $k = i$, and with the desire to find the optimal policy to get from x_i to x_T, one would do so by minimizing the truncated cost function:

$$J_\pi(x_i) = g_T(x_T) + \sum_{k=i}^{N-1} g_k(x_k, u_k) \tag{7}$$

and would find that the optimal policy is simply the truncated policy:

$$\pi_i^* = \{u_i^*, u_{i+1}^*, \cdots, u_{T-1}\} \tag{8}$$

Practically, this means that the overall optimal control policy can be derived by sequentially determining the set of optimal policies for a series of smaller sub-problems. Applying this to a road trip example, if the fastest route from New York, New York to Los Angeles, California passes through Morgantown, West Virginia, then the principal of optimality implies that the Morgantown to Los Angeles section of the overall drive is also the fastest route from Morgantown to Los Angeles.

3.3. Dynamic Programming by Backwards Induction

In the HEV powertrain optimal control problem where knowledge of the drive cycle is known *a priori*, the optimal policy can be found with a DP algorithm that employs backward induction. This is where the problem is initially considered from the final, or terminal, point and the cost-to-go is calculated at each successive stage, working backwards in time toward the initial point.

First, the final stage ($k = T-1$) is considered, and an optimal control policy is determined for this step. This is referred to as the "tail sub-problem". Next, the optimal policy for the tail sub-problem involving the final two stages ($k = T-2 : T-1$) is determined. This process is continued until the policy for the full problem ($k = 0:1$) has been identified.

This type of problem can be efficiently solved using a recursive algorithm, as follows. Starting at the final stage ($k = T-1$), the minimum cost is found to be:

$$J_{T-1}^*(x_{T-1}) = \min\{g_{T-1}(x_{T-1}, u_{T-1})\} \tag{9}$$

The minimum cost of the next stage ($k = T-2$) is therefore:

$$J_{T-2}^*(x_{T-2}) = \min\{g_{T-2}(x_{T-2}, u_{T-2}) + J_{T-1}^*\} \tag{10}$$

Working backwards from here toward the initial stage $k=0$, the total cost can therefore be found by the recursive equation:

$$J_k^*\left(x_k\right)=\min\left\{g_k\left(x_k,u_k\right)+J_{k+1}^*\right\},\ \ k=\left\{0,\cdots,T-1\right\} \tag{11}$$

where the cost $J_0^*\left(x_0\right)$ is the optimal cost of the overall control policy π^*.

3.4. Statement of the Optimization Problem

For the benchmark analysis for the optimal control of a parallel P4 hybrid-electric 2019 Chevrolet Blazer, the basic objective function can be expressed as:

$$J=\min\sum_{k=0}^{T}\dot{m}_{fuel}\left(k\right) \tag{12}$$

where k is the discrete timestep of the drive cycle, $\dot{m}_{fuel}$ is the fuel consumption rate of the engine (quasi-static over each timestep k), and 0 and T are the beginning and end points, respectively. It is important to note that the minimization is not instantaneous fuel consumption, but instead the fuel consumption over the full drive cycle.

In this work, CS operation is strictly enforced to allow direct comparison of fuel economy results without the need for a correction factor. CS operation is enforced due to the convenience afforded by the necessity to pick the starting state when determining the optimal control policy. It is important to note that the explicit selection of the CS SOC as the starting point may artificially skew the optimization as the optimal policy from a different starting point with an included energy conversion may yield higher fuel economy. To implement this requirement, the change in ESS SOC between the beginning and end of the drive cycle must be kept at zero, regardless of the level of SOC variation during the drive. Note that this restriction is required for comparison and would not normally be present in real-world operation. This generates the constraint:

$$\sum_{k=0}^{T}P_{batt}\left(k\right)=0 \tag{13}$$

A key constraint is imposed by the driver torque request $\tau_{w,req}$; the powertrain cannot produce any more torque at the wheels than the driver commands in order to perfectly follow the drive cycle. In order to represent the operating regimes of positive or negative acceleration the following constraints are used.

$$\tau_{w,req}\left(k\right)=\tau_{w,ICE}\left(k\right)+\tau_{w,mot}\left(k\right),\ \ \tau_{w,req}\left(t\right)>0 \tag{14}$$

$$\tau_{w,req}\left(k\right)=\tau_{w,brk}\left(k\right)+\tau_{w,mot}\left(k\right)+\tau_{w,ICE}\left(k\right),\ \ \tau_{w,req}\left(t\right)<0 \tag{15}$$

Additional constraints are imposed on the optimization due to the physical limitations of the vehicle powertrain. There are finite limits to the amount of instantaneous power the ESS, electric motor, and ICE can supply at a given vehicle speed, stemming from the limits on both torque output and rotational speed of the components. These constraints are as follows where the subscript c identifies a component torque:

$$\mathrm{SOC}_{min}\le\mathrm{SOC}\left(k\right)\le\mathrm{SOC}_{max} \tag{16}$$

$$\tau_{c,ICE,min}\le\tau_{c,ICE}\left(k\right)\le\tau_{c,ICE,max} \tag{17}$$

$$\omega_{ICE,min} \le \omega_{ICE}(k) \le \omega_{ICE,max} \tag{18}$$

$$\tau_{c,mot,min} \le \tau_{c,mot}(k) \le \tau_{c,mot,max} \tag{19}$$

$$\omega_{mot,min} \le \omega_{mot}(k) \le \omega_{mot,max} \tag{20}$$

3.5. Problem Formulation

As backward induction dynamic programming requires the use of a discrete-time system, the powertrain and vehicle are modeled as such. The drive cycle is discretized to a time resolution of 1 second per step. Each timestep, referred to as a "stage" in DP terminology, is represented by the variable *k*. The SOC is selected as the state variable, thus:

$$x_k = f(\mathrm{SOC}) \tag{21}$$

The motor wheel torque and the current transmission gear are selected as the control variables, thus:

$$u_k = \left[\tau_{w,mot}(k), N_{gear}(k)\right] \tag{22}$$

These control inputs influence the following state transition function:

$$\mathrm{SOC}_{k+1} = \frac{-P_{batt}\Delta t}{\text{Battery Capacity}} + \mathrm{SOC}_k \tag{23}$$

where P_{batt} is the battery current required to provide the requested motor torque, Δt is the timestep of the simulation in hours and Battery Capacity is the total energy capacity in Whr of the ESS. The negative sign is used to preserve the flow direction convention of positive current represents charge of the ESS and negative current represents discharge of the ESS.

4. Benchmarking Model

4.1. Drive Cycle Data and Shift Schedule

The benchmarking model is a backwards facing model, which means component speeds and torques are propagated from the wheels through the drivetrain and to the powertrain components. These torques and speeds are based on the drive cycle selected for this analysis. First, the drive cycle must be resampled to the discrete timestep selected for the DP algorithm. In this analysis, the original drive cycle data has a timestep of 0.1 seconds. The drive cycle is resampled at 1 second by selecting the values of the drive cycle that correspond to the 1 second timestep and ignoring the other entries.

Once the drive cycle has been resampled to 1 sec, the roadload polynomial equation is used to determine the force at the wheels at each vehicle speed. The coastdown test for the 2019 Chevrolet Blazer competition vehicle resulted in the following polynomial to describe the roadload force:

$$F_{roadload}(k) = 118.56 + 3.54V(k) + 0.54V(k)^2 \tag{24}$$

where $F_{roadload}(k)$ is the roadload force at the wheels, in N, for each timestep

and $V(t)$ is vehicle speed. Applying Newton's law for rigid bodies, the total force at the wheels of the vehicle can be determined as:

$$F_{total}(k) = m\dot{V}(k) + F_{roadload}(k) \tag{25}$$

where $F_{total}(t)$ is the total force at the wheels for each timestep, m is the mass of the WVU EcoCAR team competition vehicle, and $\dot{V}(t)$ is the acceleration of the vehicle at each timestep. The wheel torque, $\tau_{wheel}(t)$, of the vehicle is then calculated as follows:

$$\tau_{w,req}(k) = F_{total}(k) * r_{wheel} \tag{26}$$

where r_{wheel} is the rolling radius of the tires of the 2019 Chevrolet Blazer. The rotational wheel speed must also be known for each timestep. The wheel speed is calculated from the linear velocity of the vehicle as follows:

$$\omega_{wheel}(k) = \frac{V(k)}{r_{wheel}} * \frac{30}{\pi} \tag{27}$$

where $\omega_{wheel}(k)$ is the wheel speed in RPM.

As part of the benchmark analysis, the constraint of the conventional shift schedule is implemented to capture the optimal fuel economy that can be achieved with the current transmission shift schedule. The gear shift schedule is determined based on the current vehicle speed and accelerator pedal position. The speed of the vehicle is inherently known *a priori* due to DP's reliance on a predefined drive cycle. The accelerator pedal position is determined by implementing the 2.5 L LCV and M3D pedal map which is a function of vehicle speed and accelerator pedal position.

4.2. Dynamic Programming Algorithm

The following steps describe the operation of the developed DP algorithm. In this model, i indicates the state grid position at k, while j indicates the state grid position at stage $k+1$. The number of stages is indicated by T. To initialize the model, the transition cost from $k = N-1$ to $k = N$ is calculated. Figure 4 is included as a visual representation of this initialization step in the benchmarking model.

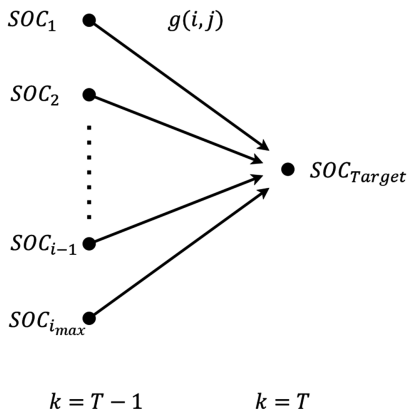

Figure 4. Initial stage calculation of the benchmarking model.

4.2.1. Electric Powertrain Model

The transition cost from each SOC in the state grid to the target SOC is calculated as follows. Rearranging the state transition equation to solve for battery power yields the following:

$$P_{batt}(i,j) = \frac{\left[\mathrm{SOC}(j) - \mathrm{SOC}(i)\right]\text{Battery Capacity}}{\Delta t} \tag{28}$$

In the initial stage, *j* represents the target SOC while *i* is each SOC in the state grid. Note that a negative battery power is a discharge from a higher SOC (at *i*) to a lower SOC (at *j*). The electrical system power limit constraints are imposed in the following piecewise equation:

$$P_{batt} = \begin{cases} \mathrm{NaN} & \text{if } P_{batt} \le P_{batt,min} \\ P_{batt} & \text{if } P_{batt,min} < P_{batt} < P_{batt,max} \\ \mathrm{NaN} & \text{if } P_{batt,max} \le P_{batt} \end{cases} \tag{29}$$

where $P_{batt,min}$ and $P_{batt,max}$ are the minimum and maximum battery power limits, respectively. The value "NaN" is useful in the DP model as it simplifies the removal of infeasible solutions as the value of "NaN" will persist through any mathematical operation and thus the transition will never be considered by the DP algorithm.

With the entirety of the drive cycle known *a priori*, the rotational speed of the electric motor is determined from the rotational speed of the wheels of the vehicle as follows:

$$\omega_{mot} = \omega_{wheel} * GR_{diff,P4} \tag{30}$$

where ω_{mot} is the P4 electric motor speed at stage *k* and $GR_{diff,P4}$ is the differential gear ratio of the P4 differential. The torque produced by the electric motor is determined by a lookup table generated from the powerloss data of the eRAD P4 electric. The electric motor component torque produced for the transition from state *i* to state *j*, $\tau_{c,mot}$, is a function of the battery power and electric motor speed for that same state transition:

$$\tau_{c,mot} = f\left(-P_{batt}, \omega_{mot}\right) \tag{31}$$

Note that the negative sign is used to preserve the battery power convention where negative power represents power discharged from the battery to produce positive propulsive torque from the electric motor.

4.2.2. Conventional Powertrain Model

With the produced motor torque known and the vehicle wheel torque requirements known *a priori*, the remaining torque produced by either the ICE or friction brakes is determined as follows:

$$\tau_{w,remain} = \tau_{w,req} - \left(\tau_{c,mot} * GR_{diff,P4}\right) \tag{32}$$

where $\tau_{w,remain}$ is the remaining wheel torque after subtracting the wheel torque of the electric motor and $\tau_{w,req}$ is the required wheel torque given by the road-load equation. The turbine speed of the torque converter is propagated from

wheel speed through the differential and transmission as follows:

$$\omega_{turb} = \omega_{wheel} * GR_{trans} * GR_{diff,ICE} \tag{33}$$

where ω_{turb} is the turbine speed of the torque converter. The locking state and clutch status of the torque converter is determined as follows:

$$Stat_{TC,Lock} = \begin{cases} 0\left(\text{unlocked}\right) & \text{if } \omega_{turb} < \omega_{idle,ICE} \\ 1\left(\text{locked}\right) & \text{if } \omega_{idle,ICE} \le \omega_{turb} \end{cases} \tag{34}$$

$$Stat_{TC,Clutch} = \begin{cases} 0\left(\text{disengaged}\right) & \text{if } \tau_{w,req} = 0 \\ 1\left(\text{engaged}\right) & \text{if } \tau_{w,req} \ne 0 \end{cases} \tag{35}$$

where $Stat_{TC,Lock}$ is the status of the torque converter lockup condition and $Stat_{TC,Clutch}$ is the status of the torque converter clutch. The speed of the ICE is then determined:

$$\omega_{ICE} = \begin{cases} \omega_{ICE,idle} & \text{if } Stat_{TC} = 0 \\ \omega_{turb} & \text{if } Stat_{TC} = 1 \end{cases} \tag{36}$$

where $\omega_{ICE,idle}$ is the target idle speed of the ICE. The speed ratio of the torque converter is determined by dividing the turbine speed by the impeller (ICE shaft) speed as shown:

$$\phi_{TC} = \frac{\omega_{turb}}{\omega_{c,ICE}} \tag{37}$$

where ϕ_{TC} is the speed ratio of the torque converter. Using data provided by GM, the torque ratio can be determined as a function of speed ratio by implementing a 1D lookup table:

$$\psi_{TC} = f\left(\phi_{TC}\right) \tag{38}$$

The remaining wheel torque is then transformed into a component torque request for the internal combustion engine by multiplying by the transmission gear ratio, differential gear ratio, transmission efficiency, and torque ratio, GR_{trans}, $GR_{diff,ICE}$, η_{trans}, and ψ_{TC} respectively:

$$\tau_{c,remain} = \tau_{w,remain} * GR_{trans}\left(N_{gear}\right) * GR_{diff,ICE} * \eta_{trans}\left(N_{gear}\right) * \psi_{TC} \tag{39}$$

Next, a series of 1D lookup tables are used to identify the maximum and minimum admissible operating points of the ICE where the speed of the ICE is the independent variable as follows:

$$\tau_{c,ICE,max} = f_1\left(\omega_{ICE}\right) \tag{40}$$

$$\tau_{c,ICE,min\ run} = f_2\left(\omega_{ICE}\right) \tag{41}$$

$$\tau_{c,ICE,min\ FCO} = f_3\left(\omega_{ICE}\right) \tag{42}$$

where $\tau_{c,ICE,max}$ is the maximum torque of the ICE, $\tau_{c,ICE,min\ run}$ is the minimum running torque of the ICE, and $\tau_{c,ICE,min\ FCO}$ is the deceleration fuel cut off torque of the ICE. The ICE component torque produced is implemented in the MATLAB model as a piecewise function, but due to the length of the function, a flow diagram shown in **Figure 5** is instead used to illustrate the logic.

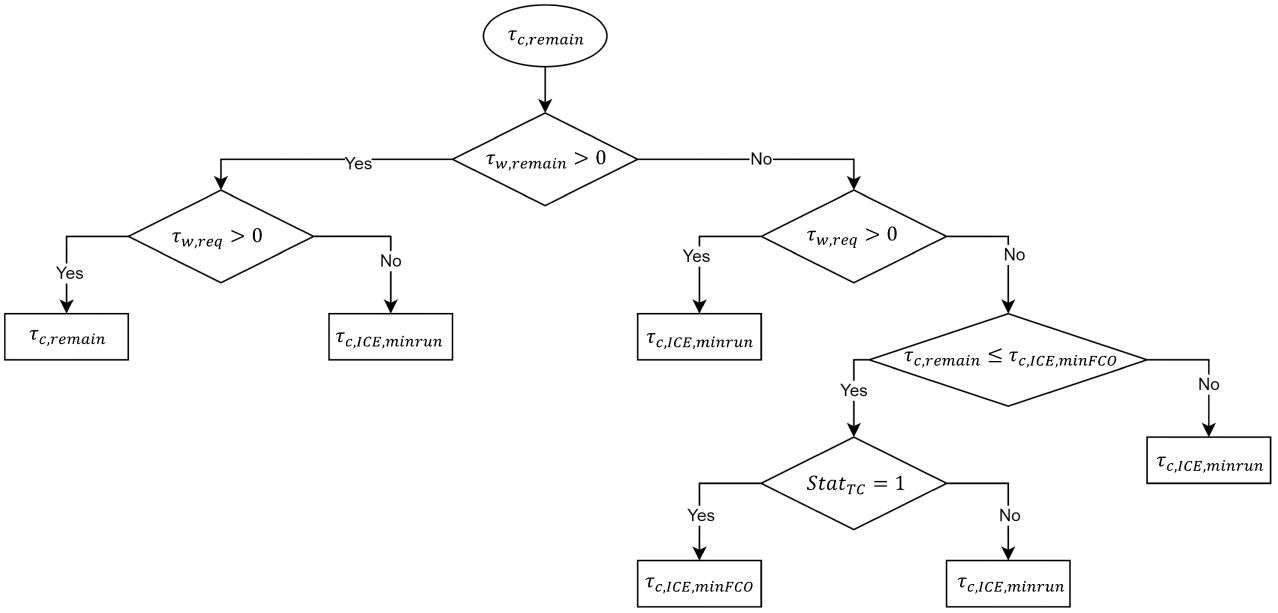

Figure 5. ICE torque determination logic.

4.2.3. Additional Constraints

With the operational points of the electric motor and engine known, two additional physical constraints must be imposed. The first is the requirement for mechanical braking to only apply "negative" torque to slow the vehicle down. Before calculating the braking torque, the component torques for both the electric motor and ICE must be converted to wheel torques by propagating the torques and speeds through each axle's driveline. The conversion from component to wheel torque for the P4 electric motor and ICE are as follows, respectively:

$$\tau_{w,mot} = \tau_{c,mot} * GR_{diff,P4} \tag{43}$$

$$\tau_{w,ICE} = \tau_{c,ICE} * GR_{diff,ICE} * GR_{trans} * \psi_{TC} * \eta_{trans} \tag{44}$$

The braking torque is then determined from the following relationship:

$$\tau_{w,brk} = \tau_{w,req} - \tau_{w,mot} - \tau_{w,ICE} \tag{45}$$

The second is the required drive cycle torque not met. This constraint is mathematically expressed by the following equation:

$$\tau_{w,req} = \tau_{w,mot} + \tau_{w,ICE} + \tau_{w,brk} \tag{46}$$

A violation of these constraints is imposed on the ICE torque value by assigning the location of the violation to be NaN.

4.2.4. Cost Function

With the operational points of the engine known, the fuel flow rate can be determined by interpolating the fuel flow rate map as follows:

$$\dot{m}_{fuel} = f\left(\tau_{c,ICE}, \omega_{ICE}\right) \tag{47}$$

where $\dot{m}_{fuel}$ is the fuel flow rate for the ICE torque ($\tau_{c,ICE}$) and ICE speed (ω_{ICE}). The DP algorithm is designed to assess the performance of the power-

train for a defined shift schedule which removes the additional dimension, N_{gear}, from the cost-to-go function, $g(i,j)$, resulting in a matrix defining the cost-to-go for the transition from state i to state j in the specified gear N_{gear} formulated as follows:

$$g(i,j) = \min\left(\dot{m}_{fuel} + \alpha\left(\left|\tau_{w,ICE}\right| + \left|\tau_{w,mot}\right| + \left|\tau_{w,brake}\right|\right)\right) \tag{48}$$

The cost function shown in Equation (49) applies a torque overproduction penalty based on the torque produced by the propulsion system with the tunable weight factor α. This effectively defines the goal of the DP algorithm to minimize the fuel consumed with the lowest possible production of torque. Finally, the total path cost for the initial step of the DP algorithm is simply equal to the cost-to-go matrix for every admissible transition from state i to j:

$$J(i,k) = g(i,k) \quad \text{for } k = T-1 \tag{49}$$

This concludes the initial stage of the DP algorithm.

4.2.5. Remaining Stages of the DP Algorithm

For each remaining stage of the DP algorithm ($k = T-1,\cdots,1$), calculations are carried out for every transition from state i to state j as shown in Figure 6.

For intermediate stages ($k = T-2,\cdots,1$) the total path cost, $J(i,k)$, is calculated according to Bellman's Principal of Optimality by the following relationship:

$$J(i,k) = \min\left[g(i,j) + J(j,k+1)\right] \tag{50}$$

Using this relationship, the total path cost is minimized by identifying the combination of the current cost-to-go from state i to state j and the total path cost leading to that transition with the lowest cost at stage k. This method can be visualized as shown in Figure 7.

The red path indicates the hypothetical selected transition path with the minimum total cost. This method implies that if at stage $k = T-2$ the optimized control actions transition from the state SOC_3 to state SOC_2, then at an

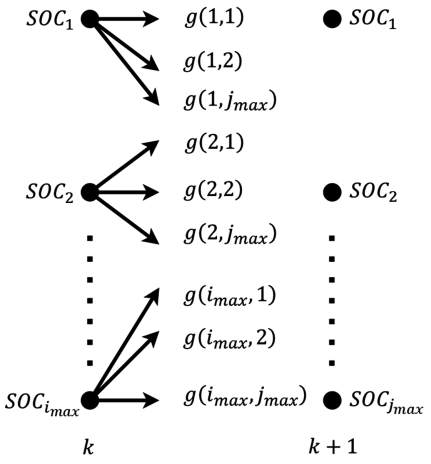

Figure 6. Intermediate stage calculations of the DP algorithm.

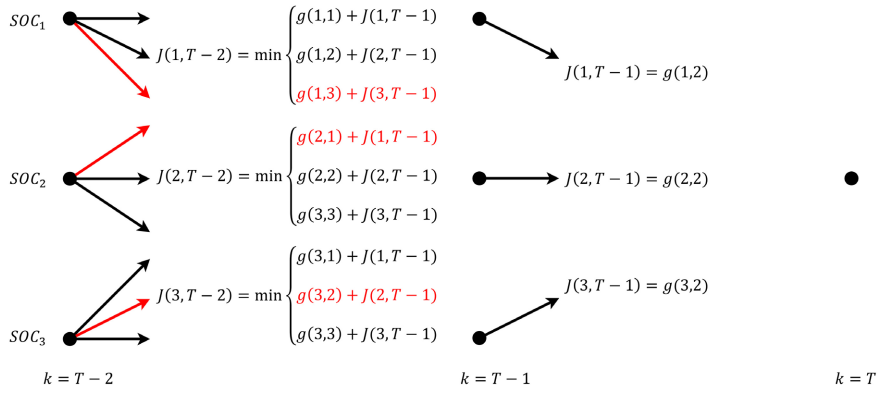

Figure 7. Example of intermediate stages of the DP algorithm at stage $k = T-2$.

earlier stage $k = T-3$, any transition from state SOC_i to SOC_3, the optimal control policy will include the previously determined control actions taken at stage $k = T-2$. This methodology is visualized in the continued example shown in **Figure 8**.

After reaching the final stage at $k = 1$, the total cost and control actions that constitute the global optimal solution have been calculated. In contrast with the final timestep, the starting SOC is difficult to enforce charge sustaining criteria. In this work, CS operation is imposed directly by selecting the initial state to be equal to the CS target SOC then following the optimal policy determined by the DP algorithm to the terminal SOC.

Once the optimal policy is found, the following set of equations are applied to determine the total efficiency for both components and total system.

$$\eta_{ICE} = \frac{P_{ICE}}{P_{fuel}} \tag{51}$$

$$\eta_{mot} = \frac{P_{mot}}{P_{loss}} \tag{52}$$

$$\eta_{total} = \frac{P_{wheel}}{P_{fuel}} \tag{53}$$

where η_{ICE}, η_{mot} and η_{total} are the efficiency of the ICE, electric motor, and total system, respectively.

By examining the torque split decisions of the optimal policy determined by the DP algorithm, the criteria for operating at specific torque split ratios can be assessed. The torque split ratio is a key part of energy management in parallel HEVs and is often communicated in various forms. For consistency, in this work, the torque split ratio is defined as follows:

$$\text{TSR} = \frac{\tau_{w,ICE}}{\tau_{Req}} \tag{54}$$

where TSR is the torque split ratio between ICE wheel torque and the required torque by the cycle. The torque split ratio defined as such represents the amount of required torque to follow the drive cycle that is produced by the ICE.

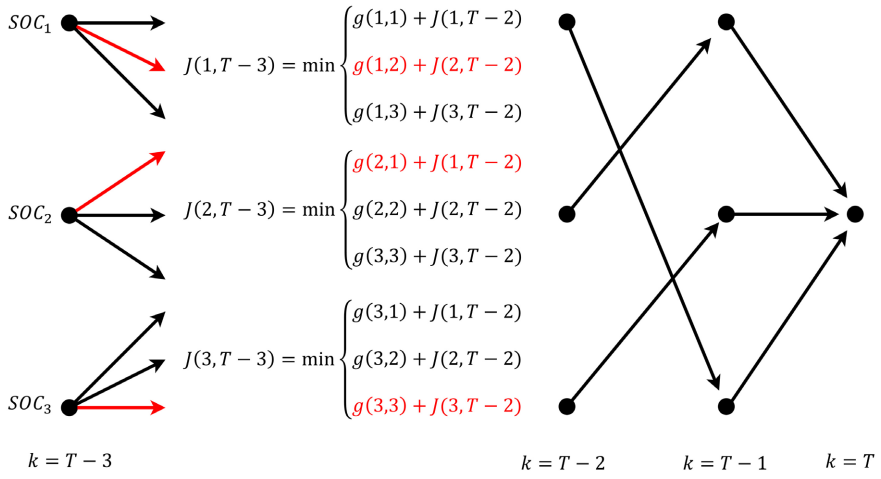

Figure 8. Example of intermediate stages of the DP algorithm at stage $k = T - 3$.

According to the EMC rules, the combined fuel economy of the Chevrolet Blazer is determined from a combined euwation of both city and highway driving characteristcis. This relationship is defined using the following relationship:

$$\text{EMC Combined Fuel Economy} = \frac{1}{\frac{0.55}{\text{EMC City}} + \frac{0.45}{\text{EMC Highway}}} \tag{55}$$

5. Results and Discussion

The benchmark analysis examines 5 drive cycles to determine the optimal control policies and relevant parameters that are beneficial with the design of the control system. The drive cycles are the EMC City, EMC Highway, US06, UDDS, and HWFET. The performance of the powertrain over these cycles can give insight for how an online control should handle specific situations. The EMC City cycle is analyzed in detail to provide an in depth look at the performance of the benchmarking model while the results of the remaining cycles are summarized to provide additional data for this specific powertrain architecture.

5.1. EMC City Drive Cycle

The EMC City drive cycle is designed to be representative of city driving conditions with two long driving events with variable speed followed by several short acceleration and braking events representative of driving behavior between stop lights. The speed and time profile with associated shift schedule of the cycle is shown in Figure 9. Similar to the torque and power requirements, the shift schedule is relatively relaxed compared to the verification cycle. Relevant information for the EMC City drive cycle and DP algorithm initialization are shown in Table 2.

Discretizing 5000 points within the state grid strikes a balance between runtime and memory requirements. The constraints of the ESS state that the SOC should remain between 20% and 80% and although the optimal control policy

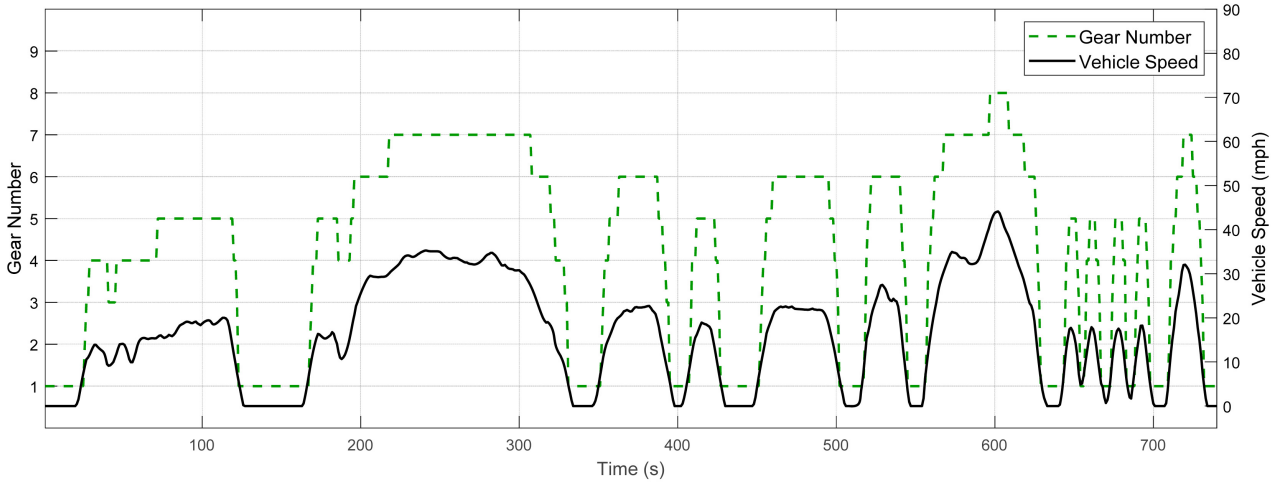

Figure 9. EMC city drive cycle speed and shift schedule.

Table 2. EMC city information and DP algorithm initialization parameters.

DP Algorithm		EMC City Drive Cycle	
Parameter	**Value**	**Parameter**	**Value**
Maximum SOC (%)	80	Distance (mi)	3.34
Minimum SOC (%)	20	Total Time (s)	740
Target SOC (%)	50	Sample Time (Hz)	1
Number of Grid Points	5000	Maximum Acceleration (m/s^2)	2.09
ΔSOC (%)	0.012	Maximum Deceleration (m/s^2)	-1.86

likely exists within a more narrow window of SOC, the entire feasible range of the ESS is included. The powertrain power output results are shown in **Figure 10**.

As the fuel consumption, power loss, and efficiency maps of both the ICE and P4 motor used in the benchmark analysis are protected by confidentiality agreements, the operating points of these components are shown overlaid with constant power lines and the maximum torque line as a frame of reference. It should be noted that the included line is a publicly available reported maximum torque determined from a 2016 Chevrolet Colorado [27] and differs from the maximum torque used to model the powertrain. The selected operating points for the powertrain components are shown in **Figure 11**.

The ICE can be identified operating at its minimum admissible running points such as minimum running power near the zero constant power line, and FCO in the negative region of torque. The ICE is shown to frequently operate around 2000 RPM with the majority of torque commands between 0 and 75 Nm of component torque produced. The ICE operates outside of this region a few times with speeds reaching up to 2500 RPM and torques of up to 100 Nm. The electric motor typically operates between ±60 Nm over a large range of speeds.

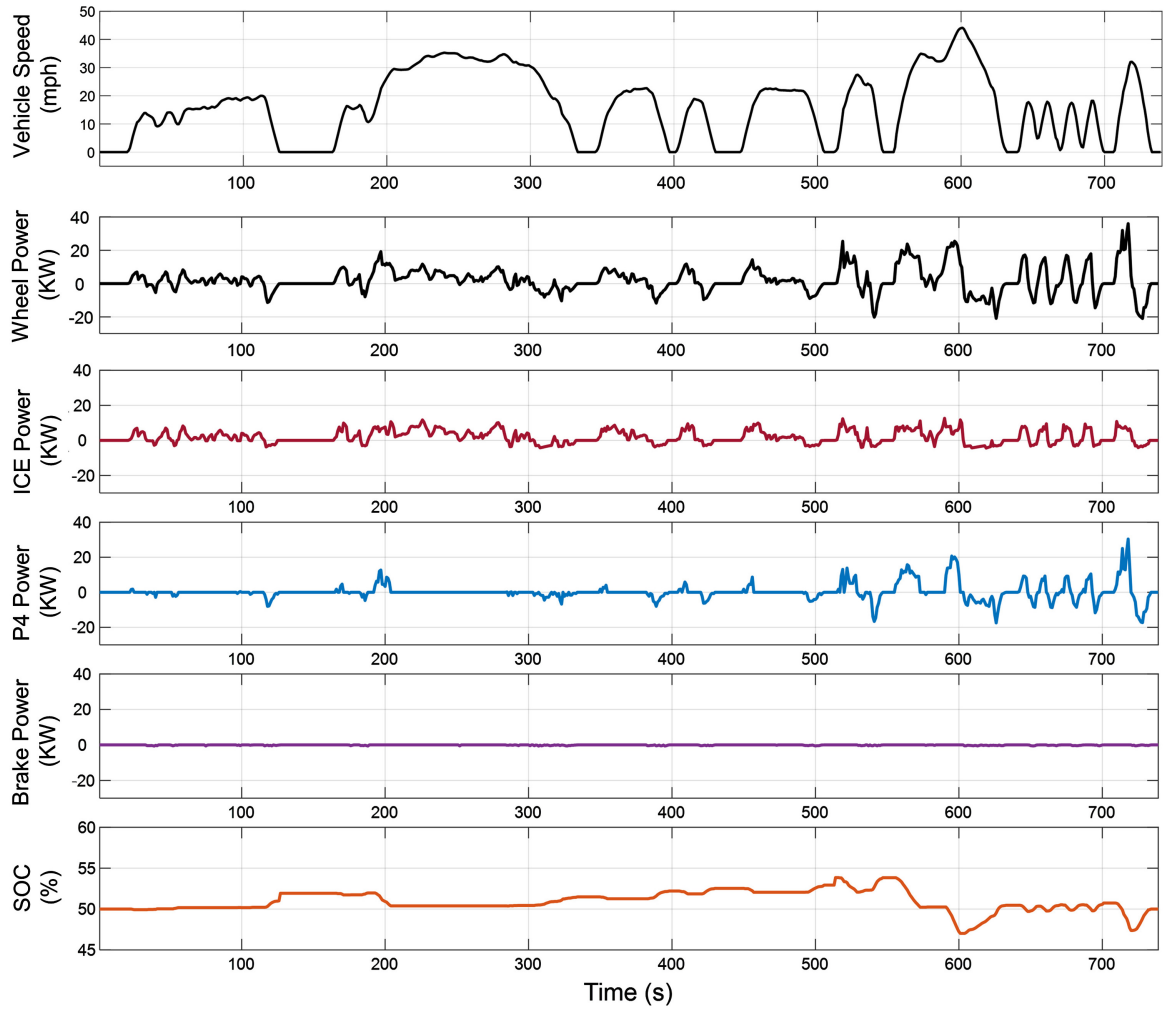

Figure 10. EMC city powertrain power output.

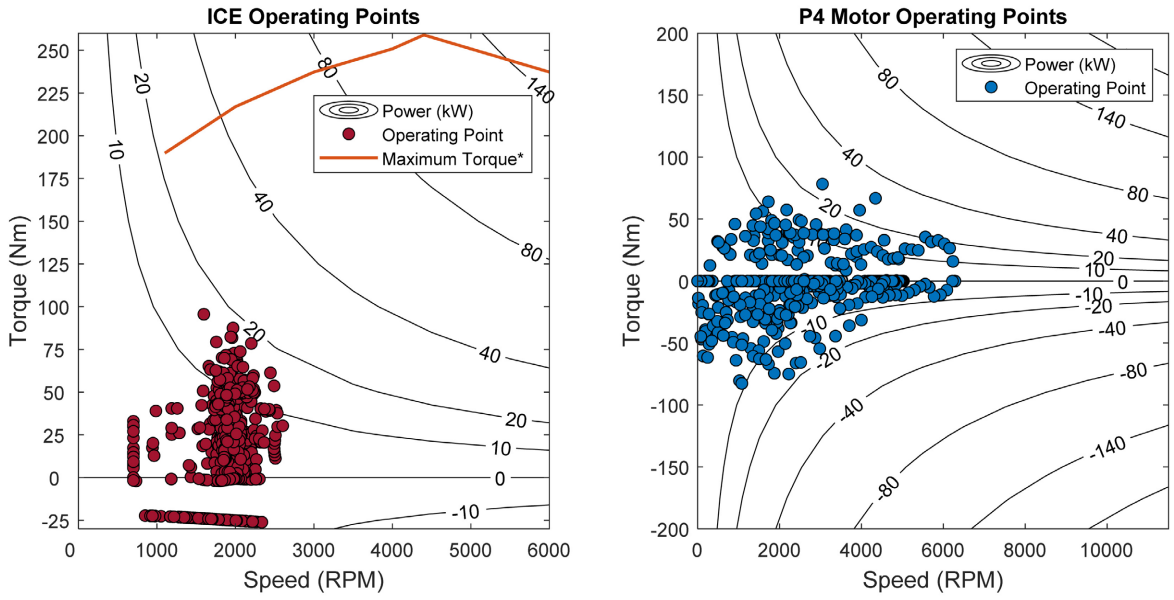

Figure 11. Operating points of ICE and P4 motor for EMC city drive cycle.

To examine the operating efficiency of the powertrain, a set of histograms were developed to look at the distribution of operating points with efficiency values originating from Equations (52)-(54). The ICE efficiency determined from ICE only operation of the vehicle over the EMC City drive cycle is used to demonstrate the efficiency improvements from implementing the electric drivetrain in urban conditions. The distribution of ICE efficiencies for ICE only operation over the EMC City drive cycle is shown in Figure 12.

The distribution of operating efficiencies for hybrid vehicle operation is shown in Figure 13.

Comparing the ICE only operation to hybrid operation, there is a clear shift in ICE efficiency from a nearly uniform distribution to a significant skew to the right between 25% - 30% efficiency. There are noticeable peaks between 25% and 30% with few operating points achieving efficiency greater than 30%. It is critical to note that for 18% of the drive cycle, the engine was in FCO and as such no

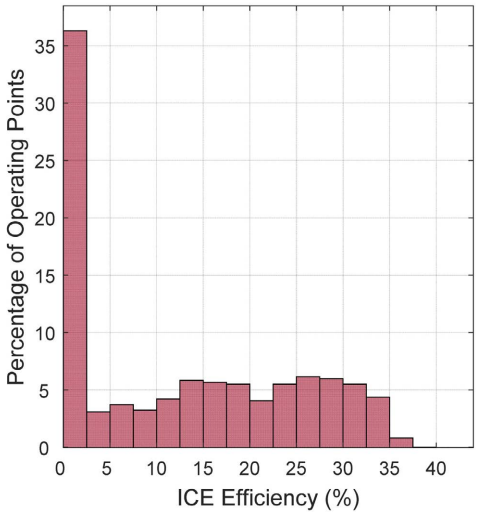

Figure 12. Efficiency distribution of ICE only operation for EMC city drive cycle.

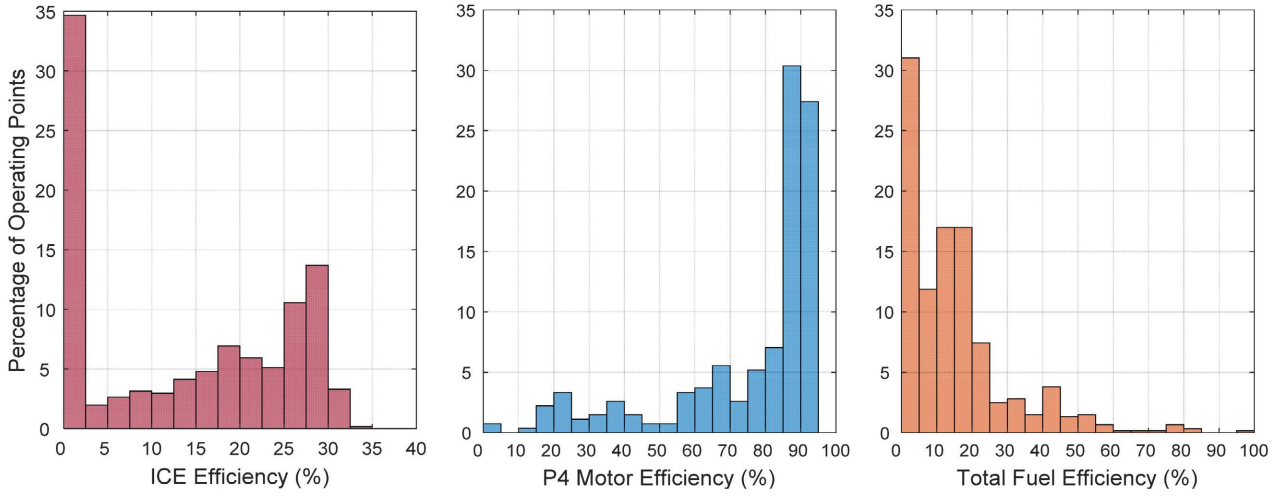

Figure 13. EMC city optimal operating efficiency distribution.

fuel was burned. These points are not captured in the histogram as there is no associated efficiency for these points. The efficiency distribution for motor operation is significantly skewed to the right with peaks between 85% - 90% efficiency that cover 57% of the operating points. It is important to note that for 62% of the drive cycle the motor is not used and thus no efficiency value is assigned to these points. Considering the system as a whole, the total fuel efficiency is skewed to the left with the majority of points operating below 20% efficiency. This distribution is expected due to the effects of losses from the driveline and energy conversions.

The TSR, defined previously in Equation (55), provides a distinct relationship between two distinct driving conditions in the EMC City drive cycle.

The first condition is a low-speed cruise with slight perturbations to the speed. **Figure 14** shows the torque production of the hybrid powertrain for the second cruise condition of the EMC City Cycle.

From the torque visualization of the torque split ratio of the cruise portion of the EMC City drive cycle, the electric motor is primarily used to assist the ICE at low speeds and high torque requirements. There is a corresponding significant increase in instantaneous fuel efficiency as the electric motor is used to provide positive propulsive torque. As speed increases, the ICE takes over to produce the entirety of required torque. For negative torque requirements, the ICE goes into FCO, as shown by the callout in **Figure 14**, while the electric motor supplies the remaining braking torque with regenerative braking.

During the stop and go traffic driving condition portion of the EMC City drive cycle, the torque requirements are much higher with large speed variation. The vehicle is quickly accelerated and decelerated multiple times throughout this portion of the drive cycle. The torque production analysis of the stop and go point of the EMC City drive cycle is shown in **Figure 15**.

Throughout the examined portion of the drive cycle in **Figure 15**, the electric motor is used significantly to produce both positive and negative torque to the wheels of the vehicle. The general trend during braking is that the ICE is pushed into FCO and the electric motor is used to capture regenerative braking. Throughout this portion of the drive cycle, there is significant increase in instantaneous fuel efficiency as the electric motor is heavily used.

Investigating the relationship between the power produced by the ICE and the torque split ratio reveals more about the control policy determined by the DP algorithm. The torque split ratio in the positive torque and power regime are shown in **Figure 16**.

As noted in the results in **Figure 16**, the TSR is 1 for low torque requests, specifically torque requests under 112 Nm. For higher torque requests there is a loose correlation with smaller values of TSR. In general, the TSR splits between 20% - 80% of torque between the ICE and electric motor with few operating points outside of that envelope. At very low power requests the ICE generally provides 100% of the torque request, but as the power requirement increases, there is a strong decreasing trend in the torque split ratio. This suggests that at

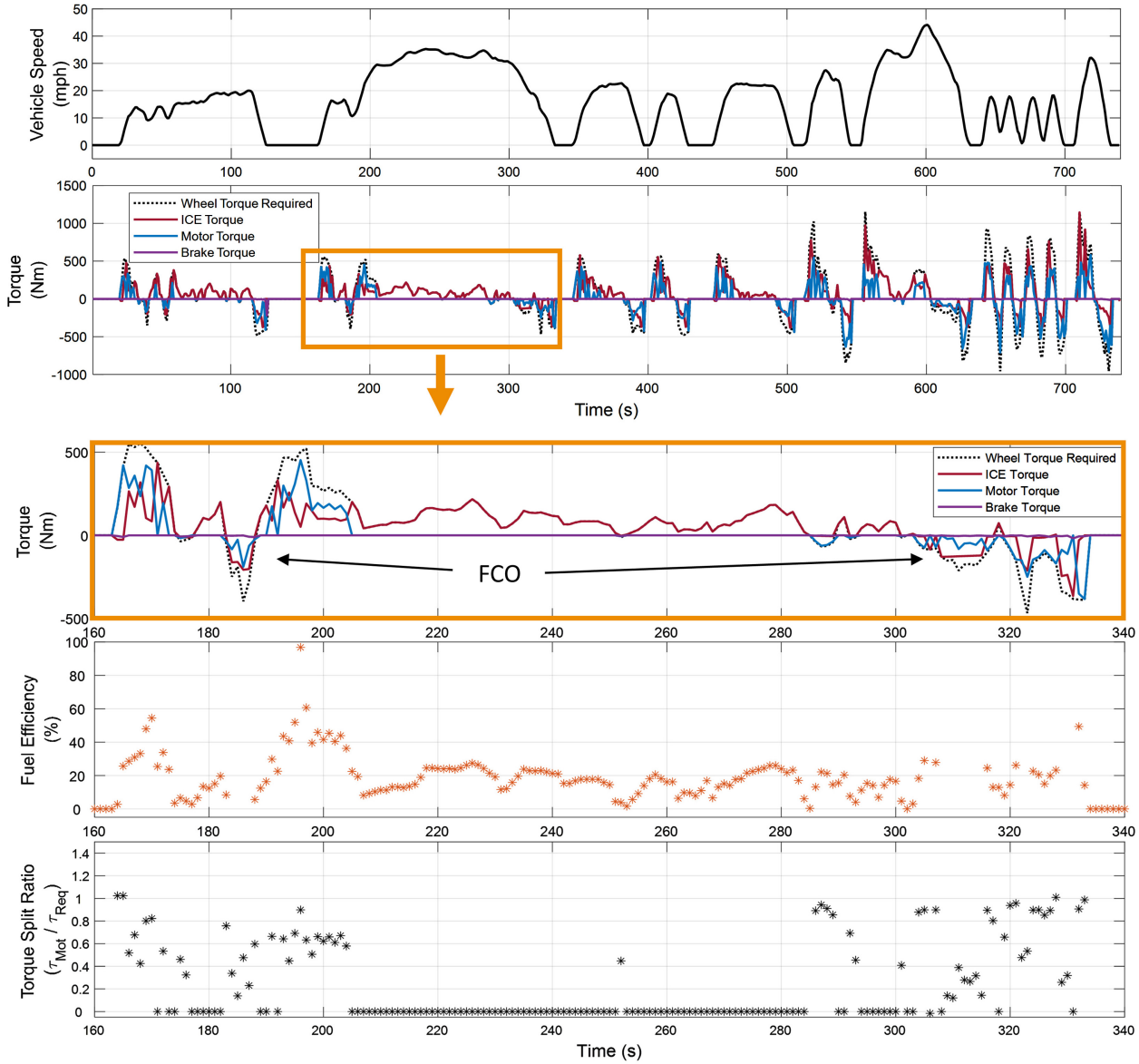

Figure 14. Torque production analysis of cruise portion of EMC city drive cycle.

high power requests, the electric motor plays a much larger role in the torque production in the optimal control policy.

The energy consumption of the vehicle for the EMC City drive cycle is shown in Figure 17.

There is significantly more fuel energy consumed throughout the EMC City drive cycle. This result is expected as the ICE does not have engine start/stop functionality thus continuously burning fuel unless in FCO. As expected, the energy produced by the ICE is far lower than the total available fuel energy due to the inherent inefficiencies associated with the ICE. The produced ICE energy exceeds the net energy at the wheels due to the losses associated with drivetrain component efficiencies. Comparing hybrid performance with non-hybrid performance, the 0.73 MJ of energy recovered using regenerative braking throughout

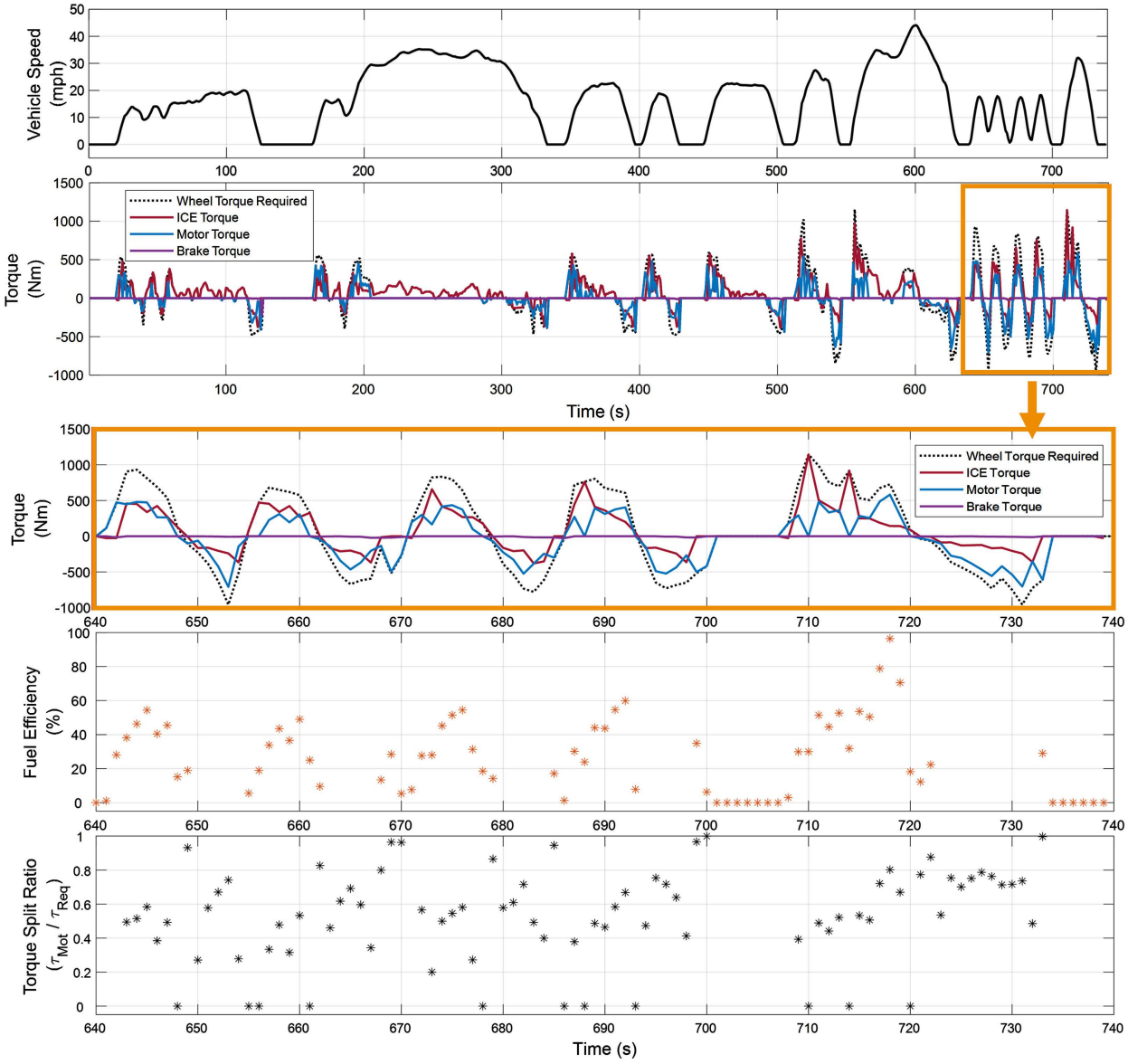

Figure 15. Torque production analysis of stop and go portion of EMC city drive cycle.

the drive cycle that is, in turn, used for propulsive torque saved a total of 3.0 MJ of fuel energy. The resulting fuel economy of the vehicle following the optimal control policy is 30.11 MPG.

5.2. Remaining Drive Cycles

In addition to the hybrid vehicle benchmarking, the non-hybrid fuel economy results are determined. Table 3 presents a summary of the performance results determined from the benchmark analysis.

Equation (56) defines the fuel economy calculaton according to the EMC rules, and with this relationship, the combined fuel economy of the hybrid 2019 Chevrolet Blazer developed by the WVU EcoCAR team is 32.99 MPG compared to the 21.6 MPG of the stock 3.6 L 2019 Chevrolet Blazer.

Table 3. DP algorithm powertrain performance results.

Parameter	EMC City	EMC Highway	US06	UDDS	HWFET
Non-Hybrid Fuel Economy (MPG)	24.64	33.17	19.15	23.26	29.16
Fuel Economy (MPG)	30.18	37.24	24.63	30.74	32.95
Fuel Used (g)	311.19	2238.5	914.20	681.61	875.57
Total Fuel Energy (MJ)	13.36	96.12	39.26	29.29	37.60
Total Wheel Energy (MJ)	3.87	21.89	14.43	9.46	9.61
Total ICE Energy (MJ)	3.27	26.36	13.77	7.56	12.15
Maximum SOC (%)	53.85	54.22	53.41	51.66	51.48
Minimum SOC (%)	47.02	46.09	30.07	33.74	43.57

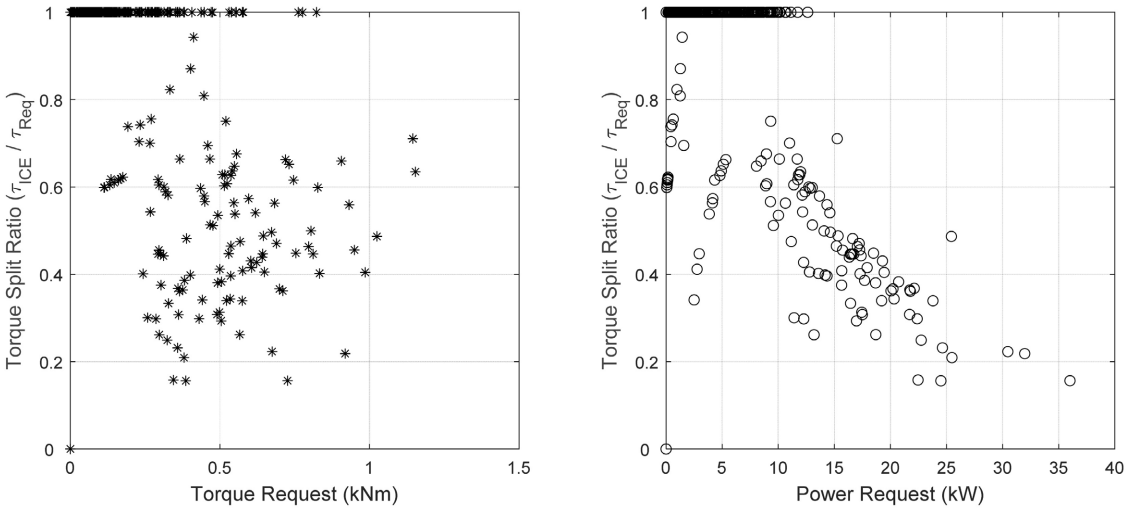

Figure 16. EMC city torque split ratio in positive torque and power regime.

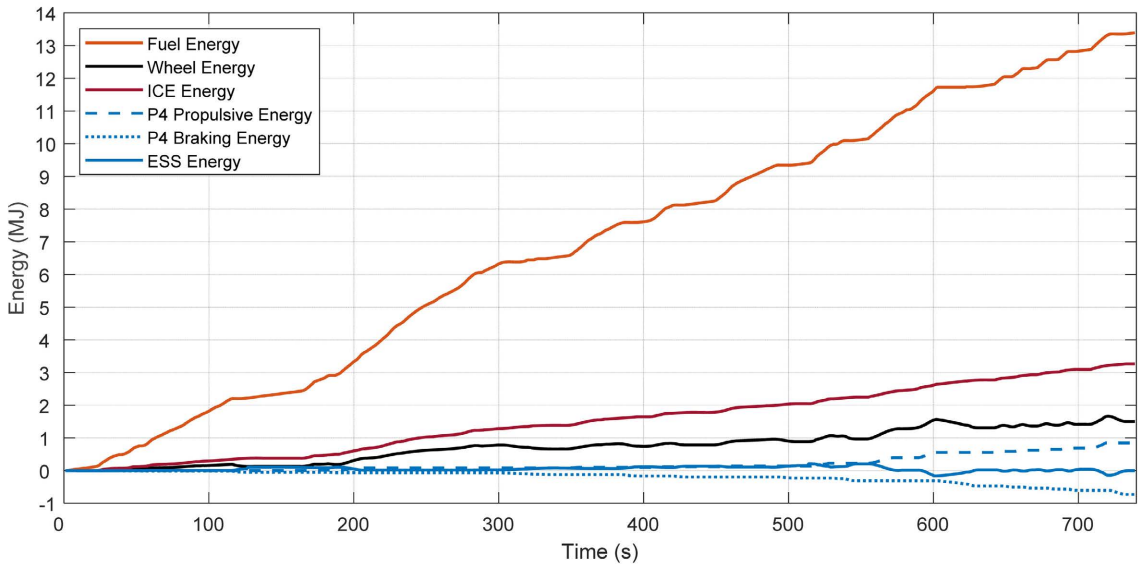

Figure 17. Energy analysis for EMC city drive cycle.

Throughout the benchmark analysis, the torque split ratio was examined as a function of torque request and power request. The relationship between the optimal torque split ratio and these parameters is often used to determine rule sets for heuristic controllers. To summarize the relationships determined from each drive cycle, a composite plot of the optimal torque split ratio is generated to identify overall control policies that may be useful for general operation of the vehicle. This composite plot is shown in **Figure 18**.

Several general control policies can be established from these composite figures. In the positive regime of **Figure 18**, the torque split ratio never drops much below 0.6 unless the torque request exceeds roughly 200 Nm. There is a clear relationship between power request and torque split ratio in both the positive and negative regimes. The data shown in **Figure 18** can be used to generate a quasi-optimal torque split ruleset by using curve fitting tools to generate trendlines for this data.

For urban driving conditions captured by the EMC City and UDDS drive cycles, the ICE operates over a range of low speeds and torques with the electric motor used to assist the engine in stop-and-go scenarios. In highway driving conditions captured by the EMC Highway and HWFET drive cycles, the ICE operates within a tighter envelope around 2000 RPM compared to urban driving with higher torque production. The electric motor is not heavily relied on for torque production in the highway driving scenarios. The seldom times that the electric motor does operate in this condition typically correspond to large power requests. The optimal control policy of the US06 drive cycle does not follow the trends of either urban or highway driving. The aggressive nature of this drive cycle results in wide operating envelopes of both powertrains. The ICE operates

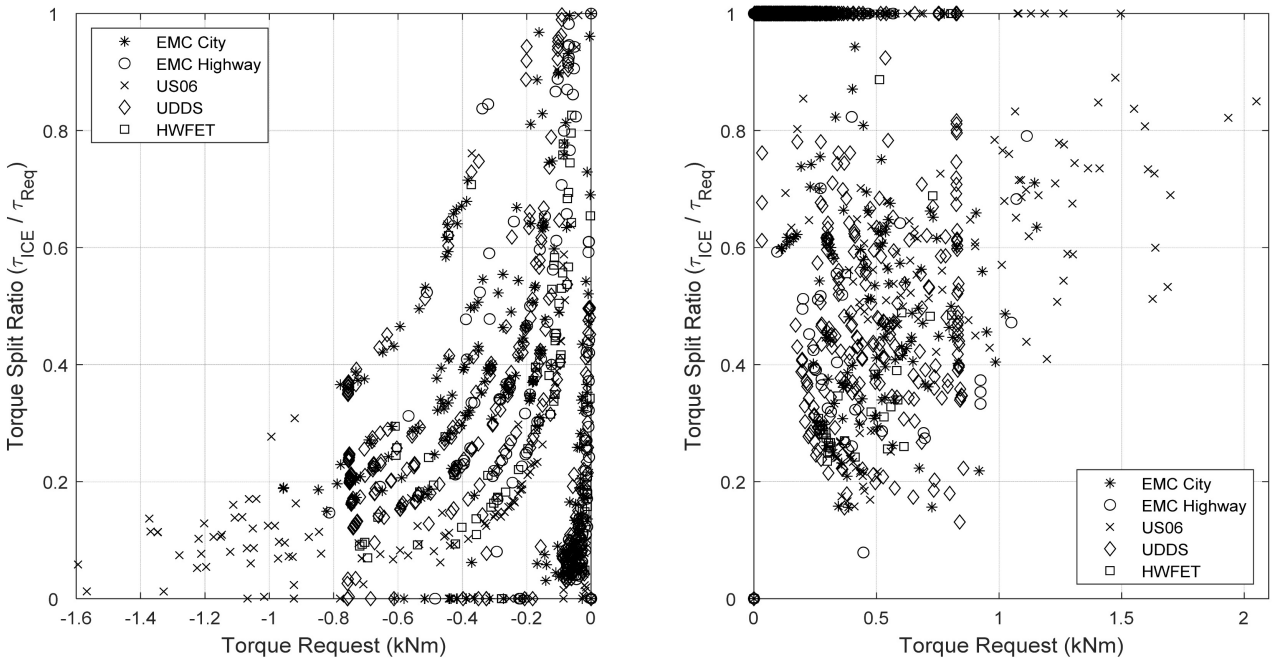

Figure 18. Optimal torque split ratio as a function of torque request for multiple drive cycles.

over nearly its entire admissible range of speeds and torques with several values reaching the maximum available torque from the ICE and electric motor. This operating strategy is likely due to the vehicle model attempting to keep up with the aggressive driving conditions of the cycle.

5.3. Computational Considerations

With a step size of 5000 steps, the memory requirement for the 5000 by 5000 arrays is 200 MB each. At each stage, 245,000 by 5000 matrices are computed with 150 smaller arrays and variables with the total memory corresponding to the length of the drive cycle. The shortest drive cycle is US06 at 600 seconds which has a memory requirement of 8.58 GB and the longest drive cycle is EMC Highway at 2962 seconds with a memory requirement of 9.97 GB. Doubling the step size to 10,000 steps substantially increases the runtime and memory requirements. A 10,000 by 10,000 matrix requires 800 MB each with US06 and EMC Highway requiring a total of 24.43 GB and 26.47 GB, respectively. A PC equipped with 32 GB of memory can execute the DP algorithm with 10,000 steps providing higher resolution in the state grid, however, the additional runtime is significant.

For the EMC City drive cycle, the runtime for 5000 steps is 26.5 minutes while 10,000 steps have a runtime of 105.0 minutes, nearly four times longer than 5000 steps. This difference in runtime is further recognized in the EMC Highway drive cycle with the 5000 and 10,000 step runtimes of 102 minutes (1.7 hours) and 436 minutes (7.3 hours), respectively. Examining the difference in fuel used results, the fuel used for the optimal policy determined for EMC City drive cycle with a step number of 5000 and 10,000 steps is 286.3 g and 284.6 g, respectively, with a percent difference of 0.55%. It is clear that for the considerable increase in runtime and memory requirements, the additional resolution of the optimal control policy is not necessary for the benchmarking activities presented in this work.

6. Conclusions and Recommendations

The objective of this work was to develop and execute a benchmark analysis for a hybrid 2019 Chevrolet Blazer using DP by backward induction. The end goal was to determine the global optimal control policy consisting of the TSR and transmission gear number for a variety of drive cycles to provide a frame of reference for the maximum possible performance of the powertrain. Throughout this work, several important takeaways and recommendations were discovered and are discussed in this section.

- The development of an appropriate backward facing model is critical to the efficient operation of a DP algorithm.
- The model must be of high enough fidelity to adequately capture the performance of a vehicle while being low enough fidelity to have fast execution time.

- Keeping as many of the calculations as possible in matrix form gave the DP algorithm incredible speed compared to using loops.
- Memory requirements are a key design parameter for the DP algorithm.
- Increased state grid resolution resulted in increased memory requirements and runtime.
- There are diminishing returns on increasing state grid resolution.
- Modifying the constraints can be useful for verification as well as generating additional benchmark data for the system.
- The cost function for the DP algorithm can include soft constraints.
- The cost function represents the overall goal of the DP algorithm optimal policy selection process.
- An appropriately selected cost function should mitigate the negative effects caused by leaking and uniform transition cost.
- The ability of the DP algorithm to select an appropriate set of control actions is highly dependent on the selection of a cost function.
- The ICE should be pushed into FCO as much as possible in deceleration events and the electric motor should be used to make up the remaining braking torque as regenerative braking.
- Mechanical brake usage should be minimized as it wastes braking energy that should be captured by the electric motor.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Ehsani, M., Gao, Y. and Emadi, A. (Eds.) (2017) Modern Electric, Hybrid Electric, and Fuel Cell Vehicles. 2nd Edition, CRC Press, Boca Raton. https://doi.org/10.1201/9781420054002

[2] Onori, S., Serrao, L. and Rizzoni, G. (2016) Hybrid Electric Vehicles. 1st Edition, Springer, London. https://doi.org/10.1007/978-1-4471-6781-5

[3] Guzzella, L. and Sciarretta, A. (2013) Vehicle Propulson Systems. Springer International Publishing, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-35913-2

[4] Kaban, S., Dong, Z. and Crawford, C. (2010) Performance Modeling and Benchmark Analysis of an Advanced 4WD Series-Parallel PHEV Using Dynamic Programming. University of Victoria, Victoria.

[5] Bowles, P., Peng, H. and Zhang, X. (2000) Energy Management in a Parallel Hybrid Electric Vehicle with a Continuously Variable Transmission. *Proceedings of the* 2000 *American Control Conference*, *ACC* (IEEE Cat. No.00CH36334), Vol. 1, Chicago, 28-30 June 2000, 55-59. https://doi.org/10.1109/ACC.2000.878771

[6] Butler, K.L., Ehsani, M. and Kamath, P. (1999) A Matlab-Based Modeling and Simulation Package for Electric and Hybrid Electric Vehicle Design. *IEEE Transactions on Vehicular Technology*, **48**, 1770-1778. https://doi.org/10.1109/25.806769

[7] Pisu, P. and Rizzoni, G. (2007) A Comparative Study of Supervisory Control Strategies for Hybrid Electric Vehicles. *IEEE Transactions on Control Systems Technolo-*

gy, **15**, 506-518. https://doi.org/10.1109/TCST.2007.894649

[8] Banvait, H., Anwar, S. and Chen, Y. (2009) A Rule-Based Energy Management Strategy for Plug-in Hybrid Electric Vehicle (PHEV). 2009 *American Control Conference*, St. Louis, 10-12 June 2009, 3938-3943. https://doi.org/10.1109/ACC.2009.5160242

[9] Phillips, A.M., Jankovic, M. and Bailey, K.E. (2000) Vehicle System Controller Design for a Hybrid Electric Vehicle. *Proceedings of the* 2000 *IEEE International Conference on Control Applications. Conference Proceedings* (Cat. No.00CH37162), Vol. 1, Anchorage, 27 September 2000, 297-302. https://doi.org/10.1109/CCA.2000.897440

[10] Driankov, D., Hellendoorn, H. and Reinfrank, M. (1993) An Introduction to Fuzzy Control. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-662-11131-4

[11] Hajimiri, M.H. and Salmasi, F.R. (2006) A Fuzzy Energy Management Strategy for Series Hybrid Electric Vehicle with Predictive Control and Durability Extension of the Battery. 2006 *IEEE Conference on Electric and Hybrid Vehicles*, Pune, 18-20 December 2006, 1-5. https://doi.org/10.1109/ICEHV.2006.352279

[12] Xu, B., Li, M., Yang, S., Guo, B. and Cui, H. (2010) Design and Simulation of Fuzzy Control Strategy for Parallel Hybrid Electric Vehicle. 2010 *International Conference on Intelligent System Design and Engineering Application*, Vol. 1, Changsha, 13-14 October 2010, 539-543. https://doi.org/10.1109/ISDEA.2010.287

[13] Majdi, L., Ghaffari, A. and Fatehi, N. (2009) Control Strategy in Hybrid Electric Vehicle Using Fuzzy Logic Controller. 2009 *IEEE International Conference on Robotics and Biomimetics* (*ROBIO*), Guilin, 19-23 December 2009, 842-847. https://doi.org/10.1109/ROBIO.2009.5420563

[14] Zeng, Q. and Huang, J. (2007) The Design and Simulation of Fuzzy Logic Controller for Parallel Hybrid Electric Vehicles. 2007 *IEEE International Conference on Automation and Logistics* (ICAL 2007), Jinan, 18-21 August 2007, 908-912. https://doi.org/10.1109/ICAL.2007.4338695

[15] Xu, S.J. (2011) Investigation of EMS Based on Fuzzy Logic Controller for an ICE/Battery/UC hybrid Electric Vehicle. 2011 2*nd International Conference on Artificial Intelligence*, *Management Science and Electronic Commerce* (AIMSEC), Deng Feng, 8-10 August 2011, 4041-4044.

[16] Sciarretta, A., Back, M. and Guzzella, L. (2004) Optimal Control of Parallel Hybrid Electric Vehicles. *IEEE Transactions on Control Systems Technology*, **12**, 352-363. https://doi.org/10.1109/TCST.2004.824312

[17] Harris, T.P., Nix, A.C., Perhinschi, M.G., Wayne, W.S., Diethorn, J.A. and Mull, A.R. (2021) Implementation of Radial Basis Function Artificial Neural Network into an Adaptive Equivalent Consumption Minimization Strategy for Optimized Control of a Hybrid Electric Vehicle. *Journal of Transportation Technologies*, **11**, 471-503. https://doi.org/10.4236/jtts.2021.114031

[18] Kazemi, H., Fallah, Y.P., Nix, A. and Wayne, S. (2017) Predictive AECMS by Utilization of Intelligent Transportation Systems for Hybrid Electric Vehicle Powertrain Control. *IEEE Transactions on Intelligent Vehicles*, **2**, 75-84. https://doi.org/10.1109/TIV.2017.2716839

[19] Lin, C.-C., Peng, H., Grizzle, J.W. and Kang, J.-M. (2003) Power Management Strategy for a Parallel Hybrid Electric Truck. *IEEE Transactions on Control Systems Technology*, **11**, 839-849. https://doi.org/10.1109/TCST.2003.815606

[20] Wang, R. and Lukic, S.M. (2012) Dynamic Programming Technique in Hybrid Electric Vehicle Optimization. 2012 *IEEE International Electric Vehicle Conference*, Greenville, 4-8 March 2012, 1-8. https://doi.org/10.1109/IEVC.2012.6183284

[21] Dokuyucu, H.I. and Cakmakci, M. (2012) Concurrent Design of Energy Management and Vehicle Stability Algorithms for a Parallel Hybrid Vehicle Using Dynamic Programming. 2012 *American Control Conference* (*ACC*), Montreal, 27-29 June 2012, 535-540. https://doi.org/10.1109/ACC.2012.6315397

[22] Patil, R.M., Filipi, Z. and Fathy, H.K. (2014) Comparison of Supervisory Control Strategies for Series Plug-In Hybrid Electric Vehicle Powertrains through Dynamic Programming. *IEEE Transactions on Control Systems Technology*, **22**, 502-509. https://doi.org/10.1109/TCST.2013.2257778

[23] Connelly, N.J., George, D.I., Nix, A.C. and Wayne, W.S. (2020) Generation and Analysis of Hybrid-Electric Vehicle Transmission Shift Schedules with a Torque Split Algorithm. *Journal of Transportation Technologies*, **10**, 21-49. https://doi.org/10.4236/jtts.2020.101003

[24] George, D. (2018) Hybrid Electric Vehicle Torque Split Algorithm for Reduction of Engine Torque Transients. West Virginia University Libraries, Morgantown.

[25] West Virginia University EcoCAR Team (2019) Architecture Selection Report. West Virginia University, Morgantown.

[26] Bellman, R.E. and Dreyfus, S.E. (1962) Applied Dynamic Programming. Princeton University Press, Princeton. https://doi.org/10.1515/9781400874651

[27] General Motors (2016) 2.5L LCV—Powertrain OEM Sales.