Proceedings of the ASME 2013 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference
IDETC/CIE 2013
August 4-7, 2013, Portland, United States

DETC2013-13043

DESIGN OF HYBRID-ELECTRIC VEHICLE ARCHITECTURES USING AUTO-GENERATION OF FEASIBLE DRIVING MODES

Alparslan Emrah Bayrak
Mechanical Engineering
University of Michigan
Ann Arbor, Michigan, 48109
Email: bayrak@umich.edu

Yi Ren*
Mechanical Engineering
University of Michigan
Ann Arbor, Michigan, 48109
Email: yiren@umich.edu

Panos Y. Papalambros
Mechanical Engineering
University of Michigan
Ann Arbor, Michigan, 48109
Email: pyp@umich.edu

ABSTRACT

Several hybrid-electric vehicle architectures have been commercialized to serve different categories of vehicles and driving conditions. Such architectures can be optimally controlled by switching among driving modes, namely, the power distribution schemes in their planetary gear (PG) transmissions, in order to operate the vehicle in the most efficient regions of engine and motor maps. This paper proposes a systematic way to identify the optimal architecture for a given vehicle drive cycle, rather than parametrically optimizing one or more pre-selected architectures. An automatic generator of feasible driving modes for a given number of PGs is developed. For a powertrain consisting of one engine, two motors and two PGs, this generator results in 1116 modes. A heuristic search is then proposed to find a near-optimal pair of modes for a given driving cycle and vehicle specification. In a study this process identifies a dual-mode architecture with an 8% improvement in fuel economy compared to a commercially available architecture over a standard drive cycle.

1 Introduction

A variety of hybrid-electric vehicle (HEV) architectures have been commercialized to date. For a given architecture, one or more "operating modes" of the planetary gear (PG) transmission determine the distribution of power among engine, motors and output shaft. Optimal fuel efficiency and drivability can be achieved by controlling mode-shifting, engine and motor outputs so that the engine and motors operate in their most efficient map regions. For example, the Chevrolet Volt model uses one engine, two motors and one PG in four operating modes for different driving conditions such as launching (low-speed high-torque) and highway (high-speed low-torque) drives.

Existing HEV architectures include 1-PG [9, 16], 2-PG [12, 11, 4, 3, 2] and even 3-PG [14, 10] systems. For a given vehicle and associated duty cycles, determining the best architecture is an open question. For example, the most fuel efficient architecture for a military combat vehicle [8] is different from that for a city-driving truck [6]. Selecting the best HEV architecture is a combinatorial problem; for example, Zhang et al. discussed a 4-mode Toyota Prius architecture with one PG [16] and 6 clutches; Liu and Peng analyzed 288 feasible modes yielded from a 2-PG system by permuting the connection of powertrain components to the PGs [8]. A "manual" examination of candidate modes was employed to identify the feasible modes and then test them. An automated process to select the best architecture would be desirable.

This paper describes such a design automation process. We introduce a bond graph representation of HEV modes and propose a mode generation algorithm to enumerate all feasible modes, i.e., modes with two degrees-of-freedom, given a set of powertrain components. For example, we show that 1116 modes are generated for one engine, two motors, two PGs and one ground. A heuristic search identifies the best architecture, i.e., the combination of modes that optimizes fuel consumption for a

*Address all correspondence to this author.

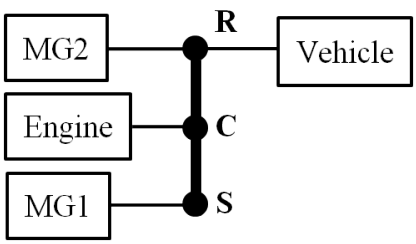


FIGURE 1. The Toyota Prius Hybrid System in the lever analogy representation; four powertrain components (the engine, two motors and the vehicle output shaft) are connected to PG nodes.

given drive cycle, with the optimal control strategy determined for each mode using a nested algorithm.

In the remainder of the paper, we introduce the bond graph representation in Section 2, assuming basic familiarity with bond graph terminology. Section 3 describes the algorithm for automatic mode generation. Section 4 discusses the vehicle component models used for simulation and the settings of the equivalent consumption minimization strategy algorithm used for optimal control. Section 5 describes the optimal design process in a case study. We conclude and discuss future directions in Section 6.

2 HEV Modes as Bond Graphs

We introduce the definition of an HEV mode and show that each mode can be represented with a bond graph, employing the "classical" Toyota Prius powertrain system as an illustration, Figure 1. The system is composed of four power sources (or sinks depending on direction convention), namely, the engine, the vehicle and the two motor/generators. These four sources are connected together by means of a planetary gear (PG), shown as the lever in the figure, with nodes "R" for the ring, "C" for the carrier and "S" for the sun gear. We use $\omega_{1,2,3,4}$ and $T_{1,2,3,4}$ to represent the speed (rads/s) and torques (Nm) that go out from the PG to the engine, the vehicle, Motor 1 and Motor 2, respectively. We assume engine and motor inertias are negligible compared to the vehicle shaft. Therefore $\omega_{1,3,4}$ and $T_{1,3,4}$ are equivalently the engine and motor speeds and torques, while ω_2 and T_2 are the speed and torque demands, rated by the final drive ratio. For a certain PG ratio ρ, the speed relationship of the system is given by

$$\begin{aligned} (\rho+1)\omega_1 &= \omega_3 + \rho\omega_2, \\ \omega_4 &= \omega_2; \end{aligned} \tag{1}$$

and the torque relationship is given by

$$\begin{aligned} -T_1 &= T_3(1+\rho), \\ T_1\rho + T_4(1+\rho) &= T_2(1+\rho). \end{aligned} \tag{2}$$

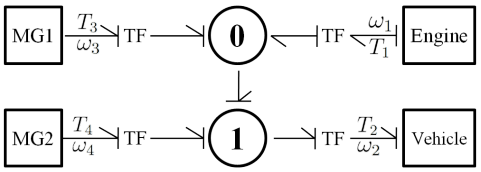


FIGURE 2. The bond graph of the Prius System.

From Equations (1) and (2), the engine speed and torque are decoupled from the vehicle speed and torque demands. This is made possible by operating Motor 1 at a certain speed so that the combined speed of Motor 1 and the engine output from the PG will meet the speed demand. Meanwhile, the output torque from the PG is combined with Motor 2 torque to meet the torque demand. Therefore the required power is split and met by two sources. This power-split design allows the engine to run at its own efficient region (speed - torque map) and thus improves fuel economy. We call Equations (1) and (2) the state-space output equations of a "mode" of the Prius "architecture".

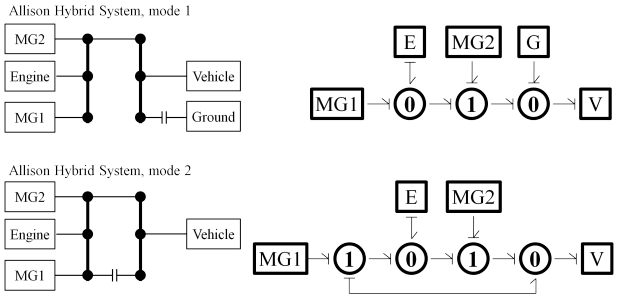
We now introduce bond graphs as a more general representation of HEV modes. To illustrate, Figure 2 presents the Prius bond graph. We use squared boxes for external sources and the circles with 0 and 1 for "0" and "1" junctions. The sign "TF" represents a transformer that scales torque and speed simultaneously for constant power before going into "0" or "1" junctions. For brevity, in the rest of this paper we will remove the transformer signs on bonds and assign weights to bonds.

The bond graph representation allows us to create different modes by changing the graph structure. Sample structures representing existing HEV modes from the literature are shown in Figure 3. Some architectures, such as the Alison Hybrid System (AHS) [13] and the Ai and Anderson architecture [2], contain multiple modes, and mode switching is enabled through clutches. For example, in AHS Mode 1, the clutch between PG2 and ground is engaged, resulting in higher output torque at vehicle launch. When the vehicle reaches a high speed, the two PGs are connected through a second clutch, leading to a more fuel efficient high speed mode. We refer to [7] for more details on the AHS.

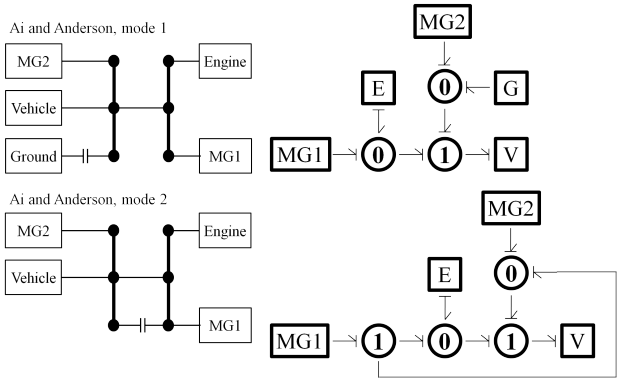
3 Mode Generation

We now describe how we can generate all possible modes for a given set of powertrain components. We use the term "external" junctions for the power sources, and 'internal" ones for "0" and "1" junctions. The number of junctions depends on the number of PGs we use and on whether a PG node is grounded. To be specific, without grounded nodes, the number of junctions is twice the number of PGs; we will have one fewer junctions with one grounded PG node. This can be observed in Figure 3. A formal proof is given in Subsection 3.2.

In the following, we elaborate on the steps of mode gener-



(a) Allison Hybrid System (AHS) [13]



(b) Ai and Anderson [2]

FIGURE 3. Bond graphs of modes in existing architectures. "E" stands for engine, "V" for vehicle output shaft, "G" for ground and "MG1" and "MG2" for the two motor/generators.

ation: (i) All possible undirected graphs are enumerated for a given number of PGs; (ii) for each graph, internal junctions are labeled by "0" or "1" junction types and causality strokes are assigned to bonds; (iii) bond graph generation is completed by assigning bond weights; (iv) the resulting bond graphs are then processed to generate the corresponding state space output equations.

3.1 Graph enumeration

We discuss the enumeration procedure of all undirected graphs for J^{ext} external junctions and J internal ones. We introduce some bond graph properties of interest first.

A valid bond graph must be simple and connected: A simple graph has no more than one bond between any two different junctions and does not contain loops on single junctions, see Figure 4; a connected graph is defined here as such that any two different junctions are connected regardless of bond directions. Each external junction has one bond connecting it to an internal junction, representing the input (or output) torque and speed flow from (or to) this external junction. We show below that for any junction with more than three bonds, there exists an equivalent graph containing junctions with only three bonds. Thus we set

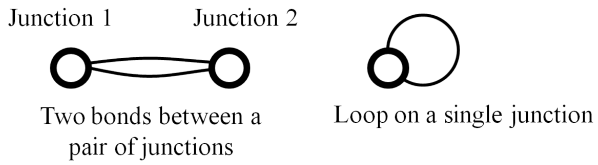


FIGURE 4. A simple graph has no more than one bond between any two different junctions and does not contain loops on single junctions.

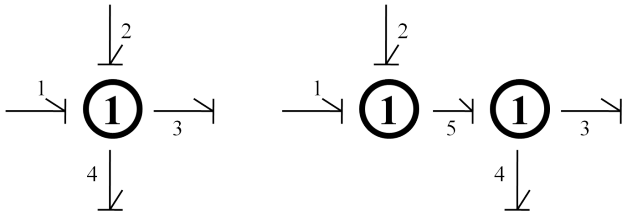


FIGURE 5. A bond graph with more than three bonds (left) and its equivalent counterpart with exactly three bonds (right).

the rule that each internal junction has three bonds.

Bond Graph Property 1. *A junction with more than three bonds can be replaced by a set of junctions with three bonds for each.*

Proof. Without loss of generality, we use "1" junction in this proof. In Figure 5, the bond graph on the left establishes the following torque (ω) and speed (T) relationship:

$$\begin{aligned} T_1 + T_2 - T_3 - T_4 &= 0; \\ \omega_1 = \omega_2 &= \omega_3 = \omega_4. \end{aligned} \tag{3}$$

This is equivalent to the relationship represented by the bond graph on the right, for which we have:

$$\begin{aligned} T_1 + T_2 - T_5 &= 0; \\ T_5 - T_3 - T_4 &= 0; \\ \omega_1 = \omega_2 = \omega_3 &= \omega_4 = \omega_5. \end{aligned} \tag{4}$$

More generally, one can always replace a junction with more than three bonds with a set of junctions with three bonds by introducing intermediate bonds.

We further show that a bond graph must have an even number of junctions, i.e., an odd number of external junctions must yield an odd number of internal ones and vice versa.

Bond Graph Property 2. *A bond graph must have an even number of junctions if each internal junction has three bonds and each external one has one bond.*

Proof. Following Property 1, a bond graph will have $(3J -$

$J^{\text{ext}})/2 + J^{\text{ext}}$ number of bonds. Therefore $J + J^{\text{ext}}$ must be even.

The external junctions considered in this study include the vehicle, the engine, two motors and optionally, the ground. When the ground is not engaged, the graph requires four external and four internal junctions, representing a mode with two PGs; when one of the PG nodes is engaged to the ground, five external and three internal junctions are required. This mode generation method can be extended to accommodate more than two PGs. However, we limit the number of junctions in this study for two reasons: (1) The computational cost of mode generation exponentially increases with the number of junctions, and (2) a mode with many PGs may have little practical value.

We now show how graphs are enumerated with a given number of junctions. The adjacency matrix of a graph can be denoted as $\mathbf{G} = [\mathbf{0}\ \mathbf{A}; \mathbf{A}^T\ \mathbf{B}]$, where $\mathbf{A}$ ($J^{\text{ext}} \times J$) and $\mathbf{B}$ ($J \times J$) are binary matrices. Element $A_{ij} = 1$ ($B_{ij} = 1$) if and only if the ith external (internal) junction and the jth internal junction are connected. Thus, by definition, $\mathbf{B}$ is symmetric. Also notice that external junctions are not connected with each other, thus the zero block in the adjacency matrix. A realization of $\mathbf{A}$ and $\mathbf{B}$ will determine a graph when the following constraints are satisfied:

1. Each external junction has one bond:

$$\sum_{j=1}^{n} A_{ij} = 1, \quad \forall i = 1, 2, ..., J^{\text{ext}}. \tag{5}$$

2. Each internal junction has three bonds:

$$\sum_{j=1}^{m} A_{ji} + \sum_{j'=1}^{n} B_{ij'} = 3, \quad \forall i = 1, 2, ..., J. \tag{6}$$

The enumeration procedure can be summarized in two steps: All feasible $\mathbf{A}$s are first generated; based on each $\mathbf{A}$, $\mathbf{B}$s can be created to satisfy Equation (6). The computation cost of this procedure is exponentially related to the degree of freedom of the system of equations from Equation (6). In practice though, this is not a significant difficulty since neither J^{ext} or J will take a large value.

It should be noted that replicate modes (graphs) with different ordering of internal junctions can be created by this procedure, as is demonstrated in Figure 6. We consider graph $\mathcal{G}_1$ a replicate of graph $\mathcal{G}_2$ when the matrix $\mathbf{G}_1$ can be transformed into $\mathbf{G}_2$ by reordering its rows and columns simultaneously, i.e., $\exists \mathbf{P},\ \mathbf{P}^T\mathbf{P} = \mathbf{I},\ \mathbf{P}^T\mathbf{G}_1\mathbf{P} = \mathbf{G}_2$. It can be shown that $\mathcal{G}_2$ is a replicate of $\mathcal{G}_1$ if and only if $\mathbf{G}_1$ and $\mathbf{G}_2$ have the same eigenvalues. We filter out replicated graphs using this criterion.

Throughout the discussion below we will use $\mathcal{V}$ ($|\mathcal{V}| = J$) and $\mathcal{E}$ ($|\mathcal{E}| = (3J - J^{\text{ext}})/2$) for the internal junction and bond sets, respectively. The ith element in $\mathcal{V}$ ($\mathcal{E}$) is denoted as $\mathcal{V}_i$ ($\mathcal{E}_i$). We use the mapping $e(j_1, j_2) = \mathcal{E}_j$ to return the ith bond in $\mathcal{E}$ connecting junctions j_1 and j_2. For simple graphs, the mapping $e(j_1, j_2)$ returns a unique index when $G_{j_1 j_2} = 1$.

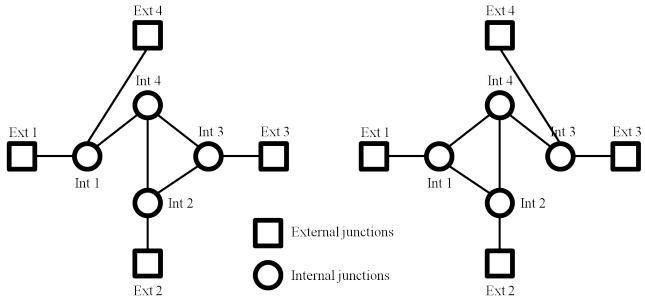


FIGURE 6. An example of replicates with different ordering of internal junctions. After assigning junction types and bond weights, these two will generate multiple identical state space output equations. Therefore, one should be removed before proceeding to the next step.

3.2 Junction type assignment

The next step is to enumerate feasible junction types and causal strokes on bonds. Similar to graph enumeration, the procedure can be summarized as finding feasible binary solutions for a system of linear equations. With the same convention as in Section 2, we use external junctions 1 through 5 to represent the engine, the vehicle output shaft, Motor 1, Motor 2 and the ground, respectively. The engine and the vehicle are defined as the flow (speed) sources while the motors as the effort (torque) ones. The ground is a special flow source where the flow is always zero.

Denote the junction type of internal junction j as t_j. We have $t_j = 1$ when the junction type is "1" and $t_j = -1$ when it is "0". We assign $c_{j_1 j_2} = 1$ to the bond connecting junctions j_1 and j_2 when the causal stroke of this bond is at junction j_1, implying $c_{j_2 j_1} = -1$. According to bond graph conventions, we have the following equality constraint on internal junction j_1 with respect to variables t_{j_1} and $c_{j_1 j_2}$:

$$-t_{j_1} + \sum_{j_2 : G_{j_1 j_2} = 1} c_{j_1 j_2} = 0, \quad \forall j_1 = 1, 2, .., J. \tag{7}$$

Since the number of binary variables in Equation (7) is $|\mathcal{V}| + |\mathcal{E}|$ while the number of equations to be satisfied is $|\mathcal{V}|$, multiple solutions exist. Notice that the solutions to Equation (7) have the following property:

Bond Graph Property 3. *The numbers of "0" and "1" junctions are equal when J is even, while the number of "0" junctions is one more than that of "1" junctions when J is odd.*

The proof is given below where we denote by J_0 and J_1 the numbers of "0" and "1" junctions, respectively, satisfying $J_0 + J_1 = J$.

Proof. Let J be even. From Equation (7) we have:

$$\sum_{j_1=1}^{J} -t_{j_1} + \sum_{j_2:G_{j_1 j_2}=1} c_{j_1 j_2} = 0. \tag{8}$$

Notice that the causality strokes on bonds associated with external junctions are given and their stroke values sum up to zero since two of them are effort sources and the other two are flow sources. Further, in Equation (8), for each $c_{j_1 j_2}$ and for all j_1 and j_2, there exists $c_{j_2 j_1}$ such that $c_{j_1 j_2} + c_{j_2 j_1} = 0$. Therefore by summing up all equations from Equation (8) we can get:

$$\sum_{j_1=1}^{J} t_{j_1} = 0, \tag{9}$$

which requires $J_0 = J_1$ for an even J. The same proof applies when J is odd, with the stroke value on the bond associated with the ground being -1 (a flow source).

This property implies that a graph with $J = 2k$ will model a k-PG mode and a graph with $J = 2k + 1$ will model a $(k + 1)$-PG mode where one PG node is engaged to the ground.

3.3 Bond weight assignment

From the examples in Section 2, a pair of "0" and "1" junctions in a bond graph can be realized by a planetary gear where the gear ratio ρ is determined by the weights on the associated bonds. Here we assume a fixed ratio $\rho = 2$ for all planetary gear sets. Figure 7 enumerates all six combinations of bond weights for a "0" junction, where each combination has different assignments of the sun, ring and carrier nodes to the associated bonds. For example, the top-left combination uses node "1" as the sun, node "2" as the carrier and node "3" as the ring. Once the enumeration on these combinations is generated, the actual bond weights are calculated for each combination. When two "0" junctions share one bond, all bond weights associated with both junctions must be calculated based on the possibly different weights assigned on the shared bond. An example is shown in Figure 8.

3.4 State space output equation generation

With a bond graph created, we generate its state space output equations. Define the vector $\boldsymbol{\omega}^{\text{ext}} = [\omega_1, \omega_2, \omega_3, \omega_4]^T$ with its components being the angular velocities of the engine, the vehicle output shaft to the final drive and the two motors, respectively. Define also the vector of angular velocities on internal bonds as $\boldsymbol{\omega}$. Then we have

$$\mathbf{W}_0 \boldsymbol{\omega}^{\text{ext}} + \mathbf{W}\boldsymbol{\omega} = \mathbf{0}, \tag{10}$$

where $[\mathbf{W}_0, \mathbf{W}]$ can be calculated using bond weights and velocity relationships at "0" and "1" junctions. Notice that $\mathbf{W}$ has

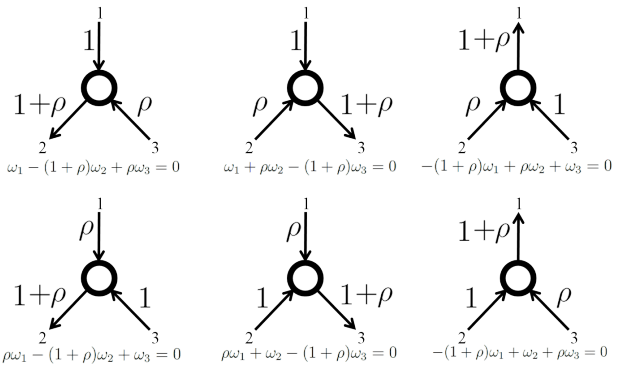


FIGURE 7. Six possible combinations for one "0" junction, switching roles of sun, ring and carrier of the three associated bonds.

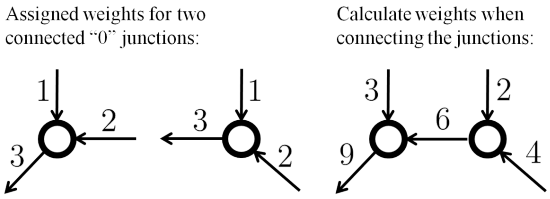


FIGURE 8. When two "0" junctions share one bond, the actual bond weights must be adjusted based on the assigned values.

$J_0 + 2J_1$ rows since each "0" junction has one equality constraint and each "1" junction has two. It also has $|\mathcal{E}|$ columns according to the length of $\boldsymbol{\omega}$. Using bond graph property 3, it is easy to verify that $J_0 + 2J_1 < (3J - J^{\text{ext}})/2$ when $J^{\text{ext}} > 1$. Therefore $\mathbf{W}$ has more rows than columns. When $\mathbf{W}$ has full column rank, Equation (10) can be rewritten as:

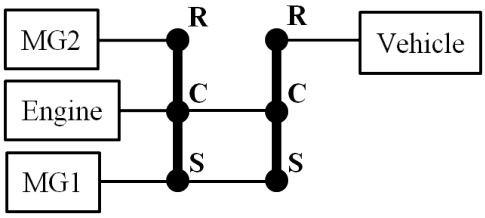
$$(\mathbf{W}_0 - (\mathbf{W}^T\mathbf{W})^{-1}\mathbf{W}^T\mathbf{W}_0)\omega^{\text{ext}} = \mathbf{0}. \tag{11}$$

The rank of $(\mathbf{W}_0 - (\mathbf{W}^T\mathbf{W})^{-1}\mathbf{W}^T\mathbf{W}_0)$ may range from 1 to 3, as the system can have one to three degrees of freedom. Here we focus only on two degree-of-freedom systems where motor speeds are governed by vehicle and engine speeds. We can further decompose $(\mathbf{W}_0 - (\mathbf{W}^T\mathbf{W})^{-1}\mathbf{W}^T\mathbf{W}_0)$ as $[\bar{\mathbf{W}}_1, \bar{\mathbf{W}}_2]$ where both $\bar{\mathbf{W}}_1$ and $\bar{\mathbf{W}}_2$ are 4-by-2 matrices. When both $\bar{\mathbf{W}}_2^T\bar{\mathbf{W}}_1$ and $\bar{\mathbf{W}}_2^T\bar{\mathbf{W}}_2$ are invertible, Equation (11) can be further rewritten as:

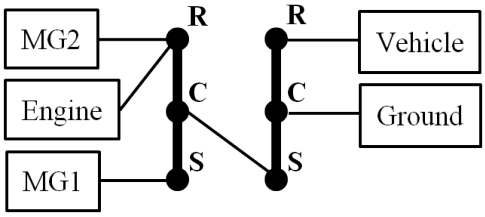
$$[\omega_3, \omega_4]^T = (\bar{\mathbf{W}}_2^T\bar{\mathbf{W}}_2)^{-1}\bar{\mathbf{W}}_2^T\bar{\mathbf{W}}_1[\omega_1, \omega_2]^T. \tag{12}$$

When the gear losses and engine and motor inertias are negligible, energy conservation requires that the energy input from engine and motors equals the energy output to the vehicle:

$$\omega_1 T_1 + \omega_3 T_3 + \omega_4 T_4 = \omega_2 T_2. \tag{13}$$



(a) Toyota Prius-like 2-PG mode



(b) Chevrolet Volt-like 2-PG mode, the PG on right serves as an extra final drive.

FIGURE 9. 2-PG modes similar to Toyota Prius in (a) and Chevrolet Volt in (b).

Denoting $\mathbf{C} = (\bar{\mathbf{W}}_2^T\bar{\mathbf{W}}_2)^{-1}\bar{\mathbf{W}}_2^T\bar{\mathbf{W}}_1$, Equations (12) and (13) lead to

$$[\omega_1, \omega_2]\left(\mathbf{C}[T_3, T_4]^T - [-T_1, T_2]^T\right) = 0. \tag{14}$$

Since Equation (14) holds for any arbitrary ω_1 and ω_2, we have

$$[T_3, T_4]^T = \mathbf{C}^{-1}[-T_1, T_2]^T. \tag{15}$$

Equations (12) and (15) are used to calculate motor speeds (torques) based on engine and vehicle speeds (torques).

3.5 Results for 2-PG mode architecture

Following the mode generation procedure above, 1116 modes with unique state space output equations ($\mathbf{C}$ matrix) are generated considering (i) one engine, two motors and two PGs with fixed ratio $\rho = 2$, (ii) a two degree-of-freedom system only and (iii) with and without one PG node connected to the ground. All feasible 2-PG modes are found, including existing ones such as those in [2, 13, 5]. Notably, we also found the 2-PG counterparts of the Toyota Prius and the dual-motor mode of the Chevrolet Volt, as shown in Figure 9. The number of feasible modes derived here is more than that from [8], which does not allow two power sources to be connected to one PG node. It should be noted that the proposed method does not consider multiple grounded PG nodes or negative PG ratios used in compound planetary gear sets. Nonetheless, these can be incorporated when such alternatives are of interest.

Specification	Value
Vehicle Body Mass	1400 [kg]
Tyre Radius	0.3 [m]
Frontal Area	2 [m^2]
Aerodynamic Drag Coefficient	0.29
Battery Voltage	350 [V]
Battery Nominal Capacity	6.5 [Ah]
Electric Motor Output Power	50 [kW] (rated)
Engine Output Power	43 [kW] (1.5 [L])

TABLE 1. Vehicle Specifications

4 Vehicle Model and Control

A set of modes generated from the previous section forms an architecture and is "exported" to a vehicle model. The performance of an architecture is defined by its optimal fuel efficiency on a certain drive cycle. This efficiency is obtained by the Equivalent Consumption Minimization Strategy (ECMS) that finds the optimal control policy for engine and motor operations and mode shifting through the drive cycle. This section gives modeling details of the vehicle and describes our implementation of the ECMS algorithm.

4.1 Vehicle model

The generated modes are incorporated in a vehicle model for performance simulation. We simulate a mid-size passenger car with a high voltage battery, two interior permanent magnet synchronous motor/generators, an engine, a transmission and a vehicle body model. Model specifications are given on Table 1.

A static battery model is used to calculate battery losses and state of charge (SOC). We simplify the model by considering the battery output voltage and internal resistance as constants instead of functions of SOC, since the effect of their variation is negligible within the 40% to 80% SOC operation window of our interest. A battery maximum charging current limit of 100 A is set to limit the regenerative braking power. Battery transients are also neglected from the model.

Two equivalent motors/generators are used in the model. The efficiency and maximum/minimum torque maps follow those of a 2006 Toyota Prius. These maps are used to calculate the motor loss and the motor current input to the battery model. A fixed inverter efficiency of 95% is integrated in the efficiency map.

The 1.5L engine model uses a fuel consumption map with respect to engine torques and speeds. A maximum engine torque map is also used to limit the output torque of the engine.

The transmission model consists of a 2-PG transmission and

a final drive ratio. A set of state space matrices generated from Section 3 together with a mode shift signal are passed to the transmission model to calculate the torque and speed relationships among the engine, motors and the vehicle shaft based on a quasi-stationary approach where engine and motor inertias are neglected. Mode shifting loss is not modeled in this study. A final drive ratio of 3.9 is used.

4.2 Optimal control

The ECMS algorithm we use follows the idea from [1]. To start with, we use the change in SOC ($\Delta\mathrm{SOC}$) and the mode shift signal as control variables. The states are SOC, driving mode and the time step associated with certain vehicle output torque and speed demands. These torque and speed demands are pre-calculated from the drive cycle, provided the vehicle follows the drive cycle strictly. Recall that the two degree-of-freedom transmission model decouples the engine operation from the vehicle load. With output torque and speed demands known at each time step, iterating over a discrete set of feasible engine torque and speed values yields a Pareto set of battery power and corresponding engine fuel consumption, i.e., each point on this Pareto curve corresponds to an optimal engine and motor operating point. Note that such Pareto curves can be pre-calculated for all time steps and modes so that the optimal fuel consumption and battery power usage can be "looked up" for a given set of states and control variables.

During a drive cycle, the equivalent consumption at each state can then be calculated as a sum of instantaneous engine fuel consumption and battery power consumption multiplied by a conversion factor [15]. At a given time step, the optimal control variables are determined by minimizing the equivalent consumption for the given states. We then look up the pre-calculated Pareto curves to find the engine and motor operations corresponding to these control variables. This instantaneous control strategy is proved to be able to calculate the fuel consumption close to the global optimum [15].

Notice that the choice of the conversion factor significantly affects the battery SOC variation over the drive cycle. Therefore in order to sustain the battery charge at the beginning and end of the drive cycle (at 60% SOC), an appropriate conversion factor needs to be searched for each architecture under evaluation. To this end, we use a simple root-finding algorithm (the secant method) that considers the final SOC as a function of the conversion factor: For a given architecture and drive cycle, the algorithm starts by running two simulations using ECMS with different conversion factors, yielding two final SOC values. A linear function of SOC with respect to the conversion factor is then created and a third conversion factor corresponds to the desired SOC is obtained from this linear interpolation and its true SOC can be simulated. The iteration continues with the interpolated conversion factor and another factor that is closest to the desired SOC from previous iterations until a final SOC is obtained within 0.2% error from the target SOC.

5 Optimal Design for a 2-Mode Architecture

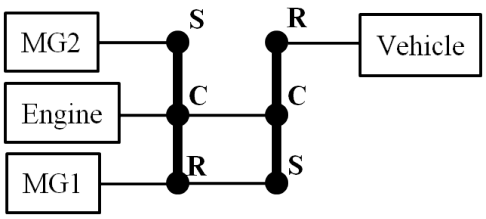
Here we elaborate on the design process of the case study on the UDDS drive cycle. All 1116 modes generated from Section 3 are considered as candidates from where we select two as an architecture. Pareto-optimal engine and motor/generator operating points for each mode over the UDDS drive cycle are pre-calculated using the vehicle model described in Section 4. These Pareto points form fuel consumption lookup tables to be used in ECMS.

The combinatorial problem of finding the optimal architecture, i.e., selecting two modes from a total of 1116 candidates, is intractable and a standart heuristic search such as genetic algorithm without insights will not work well. Here we employ a simple iterative approach: We start by using one mode that has the same state space equations as the Toyota Prius, called "Prius-like" as shown in Figure 9(a), and picking the other mode from the rest to find the best companion for the "Prius-like" to minimize the fuel consumption. This enumeration reduces the fuel consumption from 371.1 grams (64 miles per gallon (MPG)) of the Prius-like mode alone to 346.6 grams (68 MPG) with two modes over the UDDS drive cycle. We then remove the Prius-like from the pair, continuing the process with the new mode found in the previous iteration and searching for the most fuel efficient mode to operate with it. Since the Prius-like mode is in the candidate set, this design process guarantees to achieve fuel consumptions not worse than previous iterations. The process terminates when the same pair of modes are selected in two consecutive iterations. Specifically, the design process terminates in 4 iterations with the two modes shown in Figure 10, achieving a fuel consumption of 343.1 grams (69 MPG) on UDDS, an improvement of 8% from Prius alone.

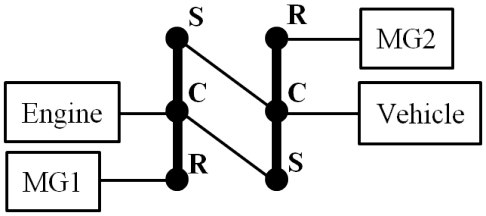
Figure 11 compares the fuel consumption during the drive cycle for Prius-like alone and the optimal architecture. We also show the SOC change of the optimal architecture to validate the tuning scheme for the conversion factor. One can see that the most significant difference in consumption between the two architectures happens during the period of 200 to 300 seconds into the drive cycle. In this region where the speed and torque demands are relatively high according to the drive cycle, we benefit from switching between two modes as opposed to sticking with a single mode.

6 Conclusions and Future Work

We introduced a systematic design process for selecting a hybrid electric powertrain architecture for a given vehicle and drive cycle. The case study showed that a simple heuristic search within 2-mode 2-PG architectures yields a powertrain design with 8% improvement in fuel consumption from a Prius-like design.



(a) Mode 1 in the optimal architecture



(b) Mode 2 in the optimal architecture

FIGURE 10. The optimal solution for 2-mode architecture starting from a Prius-like mode from Figure 9(a)

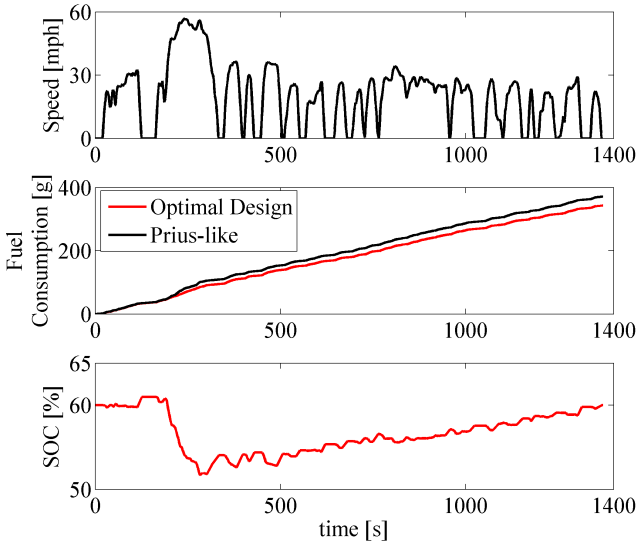


FIGURE 11. (Top) The UDDS drive cycle; (Middle) Comparison of the fuel consumption of Prius-like and the optimal architecture; (Bottom) Battery SOC variation for the optimal architecture

Three modeling issues require future investigation. First, the losses associated with mode shifting are not modeled, resulting in undesired frequent mode shifting. A potential remedy is to penalize mode shifting during the optimal control. Second, combining modes in an architecture requires a feasibility check. It is possible that a large number of clutches is needed to realize an architecture with the selected modes, making the design unrealistic. In fact, the 2-mode optimal architecture we derived in Section 5 requires 8 clutches for mode shifting, which may not be desirable in practice. To this end, a constraint on the number of clutches might be imposed during optimization. Third, the vehicle weight needs to be updated based on the architecture when evaluating different possibilities, and packaging feasibility must be checked in order to obtain more realistic results.

In addition, since this paper focuses on architecture design, the gear ratios are fixed at the beginning of the design process. While the proposed method can be followed by parameter design of the powertrain components, i.e., tuning PG gear ratios and motor sizes to further improve the fuel efficiency, the change in those parameters may affect the optimal architecture selection. Coordinating the solution of decomposing architectural and parametric design for a true system-optimal solution remains a challenge. It also remains to investigate how the optimal architecture will change for different types of vehicles and driving conditions.

7 Acknowledgement

This research was partially supported by the Automotive Research Center, a US Army Center of Excellence in Modeling and simulation of Ground Vehicle Systems headquartered at the University of Michigan, and by the University of Michigan - General Motors Collaborative Research Laboratory in Advanced Powertrains. This support is gratefully acknowledged. The authors would also like to thank Professor Huei Peng and Xiaowu Zhang from the University of Michigan and Dr. Kukhyun Ahn for their insightful advice. The opinions expressed here are solely those of the authors.

REFERENCES

[1] K. Ahn, S. Cho, and S. Cha. Optimal operation of the power-split hybrid electric vehicle powertrain. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 222(5):789–800, 2008.

[2] X. Ai and S. Anderson. An electro-mechanical infinitely variable transmisson for hybrid electric vehicles. 2005.

[3] A. Holmes, D. Klemen, and M. Schmidt. Electrically variable transmission with selective input split, compound split, neutral and reverse modes, Mar. 4 2003. US Patent 6,527,658.

[4] A. Holmes and M. Schmidt. Hybrid electric powertrain including a two-mode electrically variable transmission, Nov. 12 2002. US Patent 6,478,705.

[5] N. Kim, J. Kim, and H. Kim. Control strategy for a dual-mode electromechanical, infinitely variable transmission for hybrid electric vehicles. *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 222(9):1587–1601, 2008.

[6] C. Li and H. Peng. Optimal configuration design for hydraulic split hybrid vehicles. In *American Control Conference (ACC), 2010*, pages 5812–5817. IEEE, 2010.

[7] J. Liu. *Modeling, configuration and control optimization of power-split hybrid vehicles*. PhD thesis, The University of Michigan, 2007.

[8] J. Liu and H. Peng. A systematic design approach for two planetary gear split hybrid vehicles. *Vehicle System Dynamics*, 48(11):1395–1412, 2010.

[9] J. Liu, H. Peng, and Z. Filipi. Modeling and control analysis of toyota hybrid system. In *Advanced Intelligent Mechatronics. Proceedings, 2005 IEEE/ASME International Conference on*, pages 134–139. IEEE, 2005.

[10] M. Raghavan, N. Bucknor, J. Hendrickson, et al. Electrically variable transmission having three interconnected planetary gear sets two clutches and two brakes, Sept. 1 2006. WO Patent 2,006,091,324.

[11] M. Schmidt. Two-mode, input-split, parallel, hybrid transmission, Sept. 24 1996. US Patent 5,558,588.

[12] M. Schmidt. Two-mode, split power, electro-mechanical transmission, Nov. 26 1996. US Patent 5,577,973.

[13] M. Schmidt. Electro-mechanical powertrain, Aug. 10 1999. US Patent 5,935,035.

[14] M. Schmidt. Two-mode, compound-split electro-mechanical vehicular transmission, Aug. 3 1999. US Patent 5,931,757.

[15] L. Serrao, S. Onori, and G. Rizzoni. Ecms as a realization of pontryagin's minimum principle for hev control. In *American Control Conference, 2009. ACC'09.*, pages 3964–3969. IEEE, 2009.

[16] X. Zhang, C. Li, D. Kum, and H. Peng. Prius+ and volt-: Configuration analysis of power-split hybrid vehicles with a single planetary gear.