

# Holistic Virtual Testing and Analysis of a Concept Hybrid Electric Vehicle Model

Jonathan Spike<sup>1</sup> Dr. Johannes Friebe<sup>1</sup> Dr. Chad Schmitke<sup>1</sup>  
Dr. Christian Donn<sup>2</sup> Michael Folie<sup>2</sup> Valerie Bensch<sup>2</sup>  
Christine Schwarz<sup>3</sup>

<sup>1</sup>Maplesoft, Waterloo, Ontario, Canada, {jspike, jfriebe, cschmitke}@maplesoft.com

<sup>2</sup>IPG Automotive GmbH, Karlsruhe, Germany,

{christian.donn, michael.folie, valerie.bensch}@ipg.de

<sup>3</sup>ISKO engineers AG, Leonberg, Germany, Christine.Schwarz@isko-engineers.de

## Abstract

In this paper; the development, integration, and analysis of a hybrid electric vehicle (HEV) using system level virtual test will be presented. The work will discuss how a Modelica-based Parallel HEV powertrain model developed using MapleSim™ is integrated into industrial vehicle modeling software tool (IPG CarMaker®) using the Functional Mockup Interface (FMI) standard; and how, using API commands, virtual testing and analysis was performed with an optimization tool (Noesis Optimus®). The acausal modeling of the HEV powertrain was done using Modelica 3.2.1, allowing the flow of energy to be inferred from the operating characteristics and controller design. The multidomain model uses components from the electrical and mechanical libraries, including commercialized library components from MapleSim's Driveline Component Library and Battery Component Library.

**Keywords:** Hybrid Electric Vehicle, Powertrain, FMI, Driveline, Modelica, MapleSim, CarMaker, Optimus

## 1 Introduction

Powertrain hybridization assists an internal combustion engine to operate with optimal efficiency and enables the recuperation of kinetic energy during braking. This increases a vehicle's fuel efficiency and reduces its exhaust emissions. Additionally, powertrain electrification offers many possibilities for increasing longitudinal and lateral vehicle dynamics (Appel, Sterzing-Oppel, *et al.*, 2015). However, in view of the wide range of variants and concepts of hybrid electric vehicles, finding optimized setups often poses a challenge due to the varying boundary conditions, different cases of application, as well as interdependent vehicle subsystems.

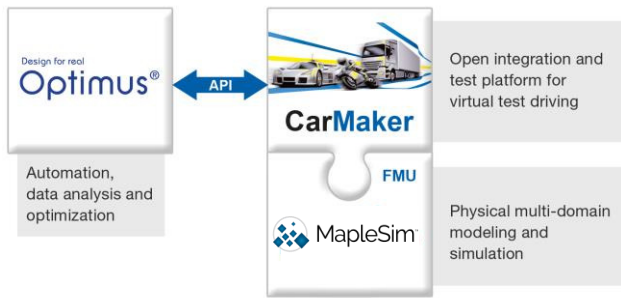
Although the process starts with simulation runs to investigate vehicle concepts and operating strategies using different powertrain topologies or components, it is crucial to examine the performance of the overall system, as well as the functionality and interaction of all relevant subsystems, in realistic scenarios and

conditions in order to meet the final development targets. This is where optimization processes and tools can assist – to find the best compromise, taking into account all the various design constraints.

In this paper, an open integration and test platform is used for the multi-objective optimization of the powertrain concept of a hybrid vehicle. This was done for different driving scenarios and driver types while taking into account longitudinal and lateral vehicle dynamics. A comparative study of fuel efficiency and performance for a hybrid-electric powertrain with different battery sizes and operating strategies was carried out, using the FMI approach to integrate a detailed vehicle powertrain model into a comprehensive full-vehicle model driven by a virtual driver on a virtual road.

## 2 Modeling and Simulation Environment

The comparative study of fuel efficiency and performance of different hybrid electric vehicle powertrain concepts carried out in this work was conducted using three development tools. The powertrain model was developed in MapleSim™, the multidomain modeling and simulation environment from Maplesoft™. It was then converted into a Functional Mockup Unit (FMU) for integration with the test environment using IPG CarMaker®, an open integration and testing platform (Kobayashi, Donn, 2015). The virtual vehicle, road and driver were set up using CarMaker, which then linked the entire virtual environment with Optimus®, a Process Integration and Design Optimization (PIDO) platform from Noesis, to perform comprehensive multi-objective optimization. Figure 1 shows an overview of the modeling and simulation environment.



**Figure 1.** Modeling and Simulation Environment

### 3 Powertrain Model

The HEV powertrain model was developed using MapleSim's Modelica-based library components. The Modelica physical modeling language has become widely-used and allows multidomain modeling within one model structure (Otter, Elmquist *et al*, 2007). The HEV powertrain model utilized MapleSim's own Driveline Component Library, Battery Component Library, as well as custom components to generate the model. The intuitive modeling process allows dragging-and-dropping of predefined components, connecting them together, and then specifying the operating parameters to obtain preliminary results. The model was then converted using the FMU code generation template for integration with vehicle modeling tools that comply with the FMI standard, such as IPG CarMaker. The automatically-generated code includes significant optimizations obtained by applying symbolic techniques. This results in fast execution time, making it suitable for both real-time applications and optimization solutions. The FMI export tool, used to integrate the HEV powertrain with CarMaker, provides the ability to generate FMI 1.0 or 2.0 along with both Model Exchange (FMU that uses the solver in the target environment) and Co-Simulation (FMU with embedded solver) configurations. The HEV powertrain model details are specified in the following section.

#### 3.1 Reference Model

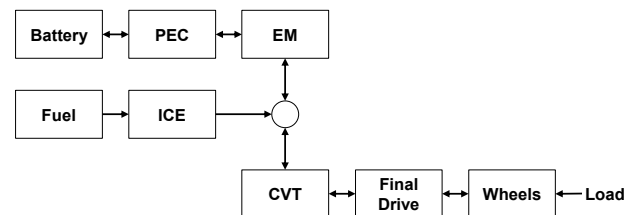
In order to meet the goal of a realistic HEV powertrain with a continuously variable transmission (CVT) and maximizing efficiency, a commercially-available powertrain concept was selected. The selected powertrain configuration is a Parallel HEV model with a CVT, inspired by the 2006 Honda Civic Integrated Motor Assist (IMA) powertrain (Hofman, Druten *et al*, 2005). The Honda IMA configuration uses the electric motor (EM) and internal combustion engine (ICE) in a parallel configuration therefore driving the same driveshaft. By adjusting the operating behavior of both the EM and ICE, an energy balance may be determined

to maximize efficiency. The EM connected in this configuration can provide engine balancing, function as the starter motor, and allow for energy flow to and from the driveshaft for either motor assisting or electrical regeneration. This parallel HEV powertrain configuration allows for six different modes of operation (Donn, Folie *et al*, 2015):

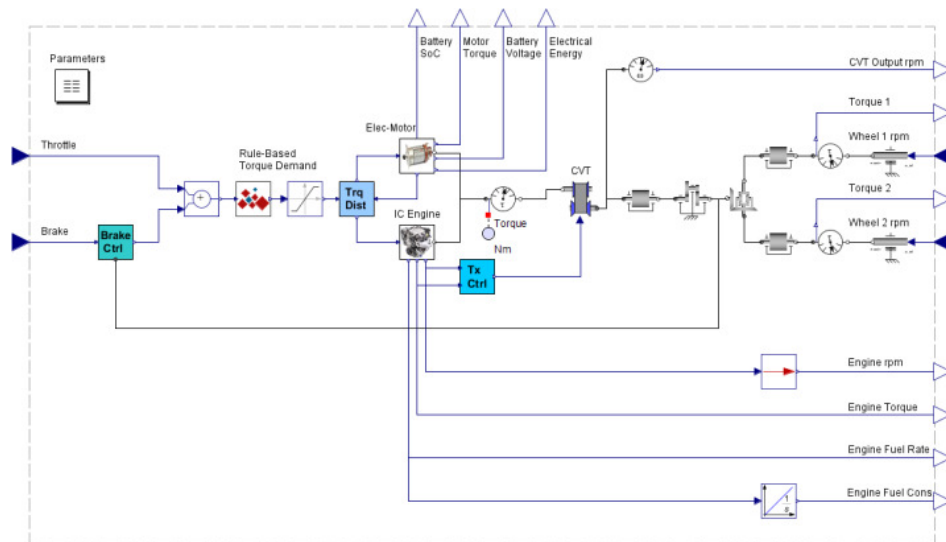
1. **Engine Driving (E):** The ICE is the only source for providing the power
2. **Motor Assist (MA):** The ICE and EM both deliver the requested driving power
3. **Motor Driving (M):** The EM is the only source for providing the power
4. **Charging (CH):** The ICE deliver power for both the vehicle and battery charge power demand.
5. **Brake Energy Recovery (BER):** During deceleration, part of decelerating power is recovered by the EM and stored in the battery
6. **Idle Stop (IS):** The ICE is stopped during full vehicle stops. Meanwhile, the auxiliaries are powered from the battery.

#### 3.2 Powertrain Component Modeling

The EM and ICE driveshaft is also coupled to the input of the CVT. The output from the CVT then drives the wheels as the torque is transferred through a differential. Since the EM motor is capable of both providing energy and extracting energy from the driveshaft, a Power Electric Controller (PEC) system is required to manage the energy flow to and from the EM and the battery. Figure 2 presents the energy flow between the different powertrain subsystems.



**Figure 2.** Energy Flow Diagram for a Parallel Hybrid Electric Vehicle Powertrain using a CVT.



**Figure 3.** FMU Generation Configuration of Hybrid Electric Vehicle Powertrain with a CVT

The acausal physical component modeling characteristics of the Modelica components allow the powertrain system to be developed based on the energy flow diagram of Figure 2. The fundamental physical equations within the Modelica components allow the system to provide/extract energy at the wheels as the vehicle is accelerating or decelerating without requiring special modeling structures or definitions for each component use case. Figure 3 shows an overview of the HEV powertrain model prepared for FMI export used in this work. The FMU uses input/output signals, unlike the physical ports in the Modelica components, thus requiring the model to be prepared in the form that is compatible with the FMI standard.

The MapleSim model was generated based on limited technical data for the 2006 Honda Civic IMA powertrain. Model data was obtained from multiple sources, and estimates were used when information was not available. Several of the key components (ICE, EM, Battery, PEC) were not explicitly available and approximate models were used to replicate the desired behavior. The following specifies additional details for each of the major subsystems.

**ICE System:** The ICE subsystem utilized engine performance data for a 1.36L engine. The engine map was implemented using MapleSim's Driveline engine component that related the engine throttle to the speed-torque tabular data. The ICE model was configured to allow the engine to be used with a stop-start system. The ICE model also provides instantaneous fuel rate, for fuel consumption calculation.

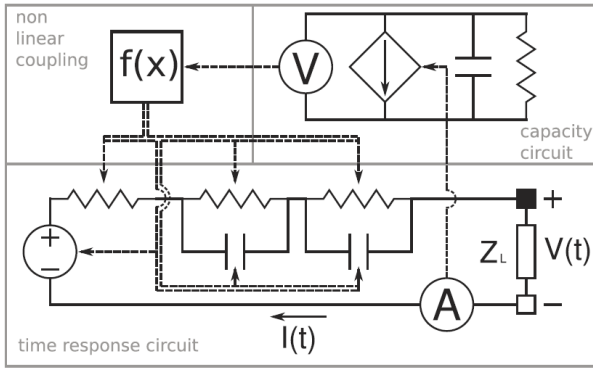
**EM, PEC, and Battery Systems:** The electric motor subsystem housed all three components (EM, PEC, and the battery). The EM selected was a permanent magnet DC motor with a rated power of 13.8kW. By using the DC-EM the PEC unit and control strategy can be relatively simple, as the PEC was not a primary focus for this investigation. However, the PEC did include features to protect the battery from being over charged or fully discharged and appropriately limit the electrical current. The battery model selected was the nickel-metal hydride (NiMH) equivalent circuit battery model from the MapleSim Battery Component Library. Further discussion on the battery is contained in Section 3.3.

**Transmission System:** The CVT component was directly obtained from MapleSim's Driveline Component Library. This component allows the transmission ratio to be adjusted based on the specified gear ratio. The CVT gear ratio operates within the continuous range of 0.45 and 2.6. The CVT was coupled to a final drive system allowing for further gear reduction. The control approach used with the CVT and ICE is discussed in Section 3.4.

### 3.3 Battery Pack

The Nickel Metal Hydride (NiMH) battery used in the reference vehicle powertrain provides the electrical energy storage required for the HEV and an additional electrical charge to operate the auxiliary components that would normally operate the 12V electrical systems. This battery supplies power during electrical motoring (M) or motor assist (MA) modes. During the charging (CH) or braking (BER) operating modes the energy is stored in the high voltage battery pack.

The NiMH battery model was selected from the MapleSim Battery Component Library to resemble the battery used in the reference vehicle. The battery library includes additional battery chemistry such as Lead Acid and Li-ion, each with available electro-chemical and equivalent circuit models. The NiMH battery model used was an equivalent circuit model that implements nonlinear resistors and capacitors. The values of the resistors and capacitors are determined as a nonlinear function of battery state-of-charge (SoC). Figure 4 illustrates the underlining equivalent circuit structure representing the battery. Equation (1) defines the exponential-polynomial function used (Chen, Rincón-Mora, 2006).



**Figure 4.** Equivalent Circuit Model Diagram for the NiMH Battery

$$R_i = k_{1,Ri} \exp(k_{2,Ri} \text{SoC}) + k_{3,Ri} + k_{4,Ri} \text{SoC} + k_{5,Ri} \text{SoC}^2 + \dots$$

$$C_j = \frac{1}{R_j} (k_{1,Ri} \exp(k_{2,Ri} \text{SoC}) + k_{3,Ri} + k_{4,Ri} \text{SoC} + k_{5,Ri} \text{SoC}^2 + \dots)$$

$$i = \{0,1,2\}, \quad j = \{1,2\}$$
(1)

In order to prevent the battery from reaching non-physical conditions, the battery model can terminate the simulation if the battery is discharged past a minimum level or similarly if the battery is over charged. As a result the PEC implemented is tasked with maintaining the battery within the desired SoC range ensuring the proper battery operation.

The calculating the instantaneous battery SoC is determined using Equation (2).

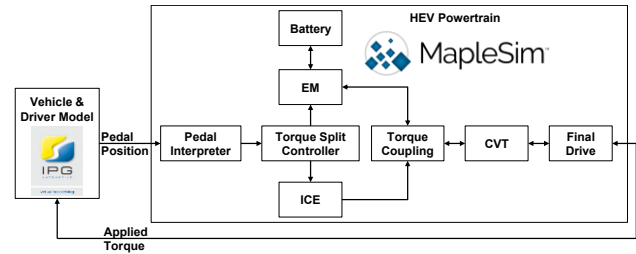
$$\text{SoC}(t) = \frac{Q(t)}{Q_{\max}} \quad (2)$$

SoC(t) and Q(t) are the instantaneous state of charge and battery electrical change, while  $Q_{\max}$  is the maximum electrical charge that the battery pack is capable of maintaining. Therefore, the value of SoC ranges between 1 (fully charged) and 0 (fully

discharged). More details about the battery model can be found in (Dao, Vyasarayani *et al*, 2012)

### 3.4 Powertrain Control Strategy

The hybrid powertrain energy management strategy determines the distribution of power between the EM, ICE and the battery while fulfilling the instantaneous power demand requirements. The implemented control strategy has a significant impact on the useable energy and performance of the powertrain (Donn, Folie *et al*, 2015). Figure 5 illustrates the energy flow chart, defining the block subsystems used in the control strategy. The powertrain is combined with the vehicle and driver model in which case the applied torque can either accelerate the vehicle or apply regenerative braking depending on the driver pedal input command.



**Figure 5.** Energy Management Flow Chart for the Parallel Hybrid Electric Vehicle

The powertrain's torque demand ( $T_d$ ) is calculated by converting information from the positions of the accelerator and brake pedals. The combined torque output from the EM and ICE is used to meet the torque demand. As a result, the distribution between the two systems – as determined by the torque split controller – provides an important part of the control strategy. In this work the torque split controller is a rule-based approach, determined by the torque demand and battery SoC.

Figure 6 illustrates an extract of the sample code that was used to define the custom Modelica torque splitter component. As a result, the powertrain operating mode is determined by the torque splitter output. When the battery SoC is at a sufficient operating point, the powertrain operating characteristics are as defined in Table 1. Otherwise, when the battery SoC is below the 10% minimum specified threshold, the torque splitter will ensure not to load the EM. Instead, the ICE will be expected to provide all of the accelerating torque demand.

```

1 model TorqueSplit
2   Modelica.Blocks.Interfaces.RealInput Td "Torque Demand"
3   annotation(/*...*/);
4   Modelica.Blocks.Interfaces.RealOutput TdENG "Torque Demand for Electric Motor"
5   annotation(/*...*/);
6   Modelica.Blocks.Interfaces.RealOutput TdEM "Torque Demand for Electric Motor"
7   annotation(/*...*/);
8   Modelica.Blocks.Interfaces.RealInput SOC "Battery State of Charge"
9   annotation(/*...*/);
10
11   parameter Real SOCmin = 0.1 "Minimum Battery State of Charge";
12   parameter Real RegenTrq = 15 "Electric Motor Regenerative Torque Load";
13   parameter Real TE = 25 "Torque Demand: Battery Regeneration Lower Limit ";
14   parameter Real TMA = 100 "Torque Demand: Motor Assist Starting Point";
15   parameter Real TRg = 65 "Torque Demand: Battery Regeneration Upper Limit ";
16   parameter Real ElecMARatio = 0.5 "Motor Assist Torque Distribution Ratio";
17
18 equation
19   TdEM =
20     if SOC < SOCmin and 0 < Td then 0
21     else if Td < 0 then Td
22     else if 0 <= Td and Td < TE then Td
23     else if TE <= Td and Td < TRg then -RegenTrq
24     else if TRg <= Td and Td < TMA then 0
25     else if TMA <= Td then ElecMARatio * (Td - TMA)
26     else Td;
27   TdENG =
28     if SOC < SOCmin and 0 < Td then Td
29     else if Td < 0 then 0
30     else if TE <= Td and Td < TRg then Td - TdEM
31     else if TRg <= Td and Td < TMA then Td
32     else if TMA <= Td then Td - TdEM
33     else 0;
34 end TorqueSplit;

```

**Figure 6.** Modelica Code Sample of the Torque Splitter

**Table 1.** Powertrain torque split strategy

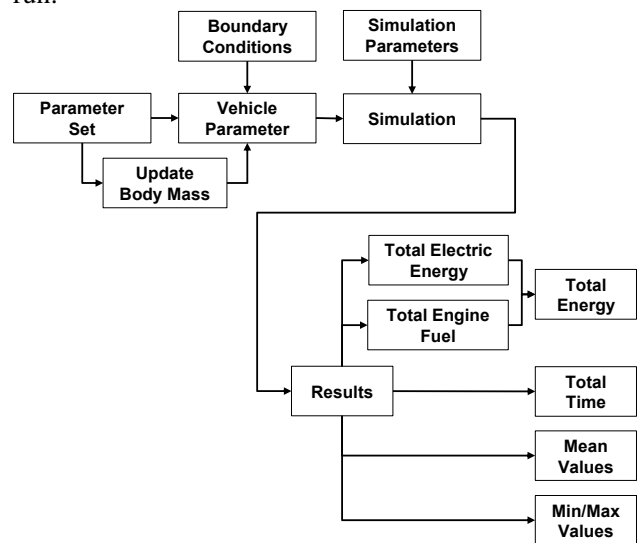
Mode	Torque Demand Range	Torque Split
BER	$T_d < 0$	$T_{ICE} = 0$ $T_{EM} = T_d - T_{Drag}$
M	$0 < T_d < TE$	$T_{ICE} = 0$ $T_{EM} = T_d$
CH	$TE < T_d < TRg$	$T_{ICE} = T_d + T_{EM}$ $T_{EM} = RegTrq$
E	$TRg < T_d < TMA$	$T_{ICE} = T_d$ $T_{EM} = 0$
MA	$T_d > TMA$	$T_{ICE} = T_d - T_{EM}$ $T_{EM} = 0.5(T_d - TMA)$

Once the torque distribution has been determined, the engine and transmission control systems are required to operate together. The approach selected assumes that the operating performance of the ICE is the major focus. The ICE and transmission control approach uses the minimal Brake Specific Fuel Consumption (BSFC) (Bai, Maguire *et al*, 2013). Both the engine map and optimal BSFC operating points have been determined in a previous analysis.

The powertrain utilizes two controllers that continuously regulate the engine throttle and the transmission ratio, adjusting the ICE instantaneous operating point to the BSFC optimal operating point. Additionally, when the vehicle is running at speeds below 20Km/h, the electrical regenerative braking is not used. Rather, only the hydraulic brake (in the CarMaker brake model) is used to stop the vehicle.

## 4 Process Integration and Optimization

The Optimus software tool provided an automated approach to performing a large number of simulations with parameter configuration changes. Optimus takes over the task of manually running simulations, obtaining, storing, and then analyzing the results. This automated approach is performed through direct communicating with CarMaker using API functions (alternatively the open ASCII interface communication can be used). Figure 7 shows the workflow deployed with the Optimus tool. Starting on the left, different parameter sets are selected, and substituted into CarMaker for simulation. Following that, the results are processed. For example, the total energy used in the simulation is determined based on the total electrical energy and fuel consumed during the entire simulation run.



**Figure 7.** Optimus Workflow of the Automated Process

One vehicle parameter, the vehicle body mass, was determined as a result of adjusting the capacity of the HEV powertrain battery. The base vehicle body mass of 1301kg is updated with the additional battery mass calculated based on (Noshin, Verbrugge *et al*, 2010) using an energy density of 80Wh/kg. Once the simulation is complete, the results can be used as inputs for an objective function, or as constraints during an optimization process.

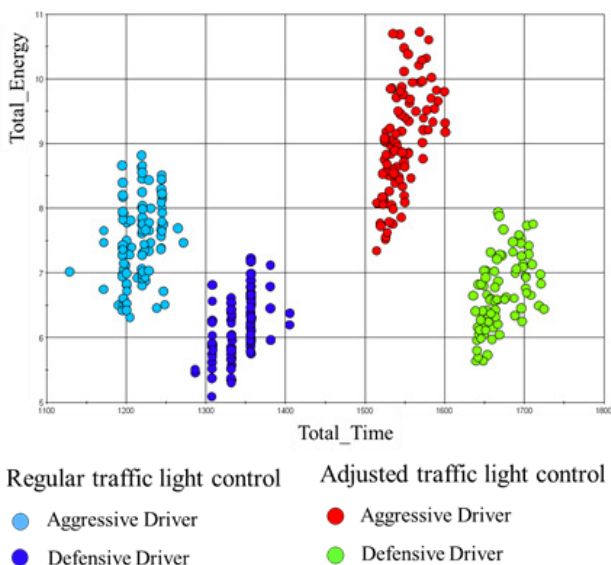
### 4.1 Design of Experiments

The design process, particularly for systems with many undetermined parameters, has the challenge of defining a suitable parameter set to use for optimization. This results in a large design space during the optimization process. One automated approach to reducing the design space is to perform a sensitivity analysis. The sensitivity analysis provides a method of determining the design variables that have the most significant



influence on the response. The most significant variables can be used in further analysis while simultaneously reducing the design space by excluding less significant factors.

A sensitivity analysis study was preformed to examine which of the parameters had the greatest influence on the response of the entire system performance. Figure 8 illustrates the results of the Latin Hypercube sampling with 100 experiments per traffic-light and driver-behavior configuration. One finding noted in the sensitivity analysis was that the traffic light timing considerably influenced simulation results, most notable for the defensive driver. This can be observed clearly by the vertical lines in the plot results with the regular traffic light control. To ensure each simulation would experience the same influence from each traffic light, the traffic lights were adjusted to ensure the vehicle would stop as it approached the intersection. The adjusted traffic light control plot points are obtained using this new traffic light timing. It can be noted that the aggressive driver was able to reliably complete the simulation run in less time than the defensive driver with the tradeoff appearing in higher total energy consumption. The adjusted traffic light control will be used for further discussion in this work.



**Figure 8.** DOE Results from Drive Cycle with Different Traffic Light Control and Driver Behavior

A set of six parameters were selected to be used in the optimization process with an interval range that remained within physically feasible limits. Table 2 shows the six parameters and the ranges selected to drive the optimization process. It also includes the vehicle body mass parameter determined by the battery capacity selected.

**Table 2.** Optimization Input Parameters

Parameter	Description	Min Value	Max Value
BatCap	Capacity of Battery	10 [Ah]/ 1.44 [kWh]	150 [Ah]/ 21.6 [kWh]
TE	Torque Demand: Battery Regeneration Lower Limit	10 [Nm]	50 [Nm]
TRg	Torque Demand: Battery Regeneration Upper Limit	25 [Nm]	100 [Nm]
TMA	Torque Demand: Motor Assist Starting Point	75 [Nm]	150 [Nm]
RegenTrq	Electric Motor Regenerative Torque Load	5 [Nm]	50 [Nm]
FinDrRatio	Final Drive Ratio	4 [-]	7 [-]
Body_mass	Vehicle Mass	1319 [kg]	1571 [kg]

The Latin Hypercube 100 sample experiments were used to determine the influence of selected parameters on the total response time and total energy consumption. Table 3 shows the output parameters monitored during the experiments.

**Table 3.** Optimization Output Parameters

Parameter	Description	Unit
Total_Time	Total Time for Drive Cycle	[s]
Total_Energy	Total Energy Consumption	[kWh]

The correlation denoted in Table 4 represents both the Pearson and Spearman rank correlation coefficients. For both, values close to +/- 1 indicate a significant linear or monotonic correlation between the input parameter and the output parameter. For example the input parameter TMA has been determined to have strong correlation with the total energy consumption particularly for the defensive driver.

**Table 4.** Correlation table for the adapted traffic light control drive cycle

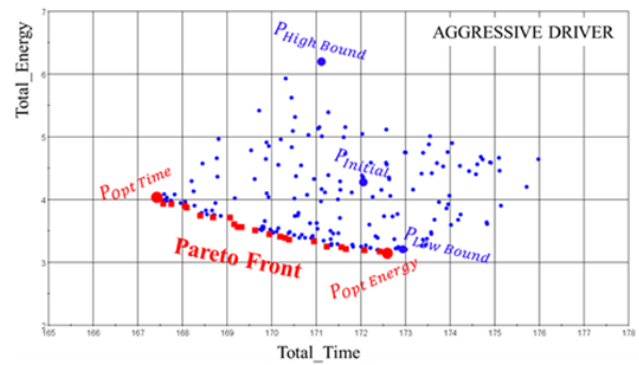
Driver Type	Parameter	Total Time Pearson (Spearman)	Total Energy Pearson (Spearman)
AGGRESSIVE DRIVER	BatCap	0.015 (0.205)	0.433 (0.434)
	TE	0.173 (0.191)	-0.333 (-0.308)
	TRg	-0.137 (-0.127)	0.196 (0.188)

DEFENSIVE DRIVER	TMA	0.331 (0.682)	0.682 (0.683)
	RegenTrq	0.097 (0.048)	0.327 (0.323)
	FinDrRatio	-0.053 (-0.259)	0.083 (0.072)
	BatCap	0.098 (0.193)	0.576 (0.574)
	TE	0.082 (0.118)	-0.345 (-0.331)
	TRg	-0.083 (-0.041)	0.245 (0.223)
	TMA	0.318 (0.723)	0.521 (0.524)
	RegenTrq	0.107 (0.076)	0.320 (0.322)
	FinDrRatio	-0.068 (-0.227)	0.121 (0.107)

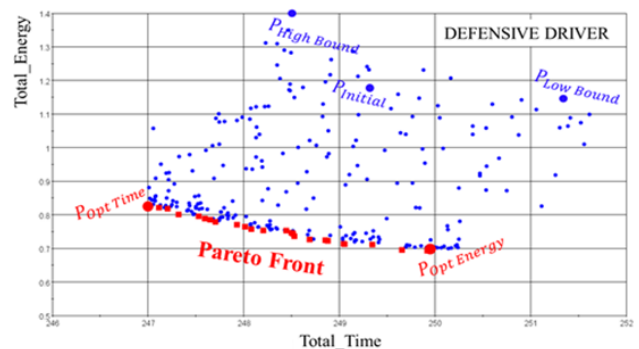
## 4.2 Multi-objective Optimization

The selection of a suitable strategy and algorithm for optimization can be influenced by multiple factors. Aside from parallelization possibilities and computer resources, the number of design variables and system behavior can have a great influence on the strategy deployed. In this case, the simulation time was not a limiting factor - with the simulation performance approximately 8-fold faster than real time. Additionally, because the HEV system under consideration is highly nonlinear, a global optimization algorithm such as an evolutionary strategy should be chosen.

This work considered two simultaneous objective functions: minimize drive cycle total time and minimize energy consumption. These two conflicting objectives will result in more than one optimal solution. As a result a Pareto front is determined defining the complete set of compromised optimal solutions. The evolutionary algorithm NSEA+ (Non-dominant Sorting Evolutionary Algorithm) was used to detect the Pareto points for the Nordschleife drive cycle. Figure 9 and Figure 10 show the optimal results for both the aggressive and defensive driver. The optimization process also considered the constraints  $TE < TRg$  and  $TRg < TMA$  as part of the requirement in the HEV powertrain controller.



**Figure 9.** Optimization Results: Nordschleife Drive Cycle and Aggressive Driver Behavior



**Figure 10.** Optimization Results: Nordschleife Drive Cycle and Defensive Driver Behavior

The five unique points denoted in Figure 9 and Figure 10 are defined as follows:

1.  $P_{initial}$ : the initial design response.  $P_{initial}$  was determined by using the mean value for each design variable considered in the optimization process.
2.  $P_{Opt\_Time}$ : the optimal solution considering the optimal time objective only
3.  $P_{Opt\_Energy}$ : the optimal solution considering the optimal energy consumption objective only
4.  $P_{High}$ : response with all design space variable using upper boundary values
5.  $P_{Low}$ : response with all design space variable using lower boundary values

For each of the objectives functions a considerable improvement was made compared to the initial design variable selection. It is worth noting that some of the optimal results were found at the boundaries, suggesting that increasing the boundary limits might further improve the optimal results.

**Table 5.** Optimization Results for Nürburgring-Nordschleife Drive Cycle

	BatCap [Ah]	TE [Nm]	TR <sub>g</sub> [Nm]	TMA [Nm]	RegenTrq [Nm]
P-Low	10	10	25	75	5
P-Initial	80	30	62.5	112.5	27.5
P-High	150	50	100	150	50
Aggressive Drive P <sub>Opt Time</sub>	11.9	31.5	43.4	76.0	25.0
Aggressive Drive P <sub>Opt Energy</sub>	10.2	43.6	49.8	75.9	10.1
Defensive Driver P <sub>Opt Time</sub>	16.7	49.9	52.1	75.5	20.6
Defensive Driver P <sub>Opt Energy</sub>	17.4	50.0	51.1	75.7	22.1
	FinDrRatio [-]	Body_Mass [kg]	Total Energy Consum. [kWh]	Total Time [s]	
P-Low	4	1319	1.15	251.4	
P-Initial	5.5	1445	1.17	249.3	
P-High	7	1571	1.46	248.6	
Aggressive Drive P <sub>Opt Time</sub>	6.8	1322	3.93	167.8	
Aggressive Drive P <sub>Opt Energy</sub>	4.3	1319	3.17	172.4	
Defensive Driver P <sub>Opt Time</sub>	6.1	1331	0.79	247.6	
Defensive Driver P <sub>Opt Energy</sub>	4.9	1332	0.72	248.9	

This example showed that a hybrid powertrain with a smaller battery was best for the aggressive driver on this particular driver cycle. As for the defensive driver, a higher torque limit (TE) for pure electric mode was more beneficial.

## 5 Conclusion

In this paper, a new and efficient approach for optimizing the design parameters of a complete powertrain model with a hybrid electric vehicle was shown. The goal was not only to integrate a complex powertrain model in an easy and intuitive way for a given vehicle, but also to detect coherences and perform advanced optimizations using virtual test driving in an automated loop without much user effort.

The powertrain model was created using MapleSim. It is a Parallel HEV model with a CVT, inspired by the 2006 Honda Civic IMA powertrain, and allows for six modes of operation. It was integrated with the CarMaker vehicle model using the Functional Mockup Interface (FMI) standard, and simulated under two scenarios – representing defensive and aggressive driver behavior. Design of experiment and multi-variable optimization were performed using Optimus.

While this work was focused on investigating the method and workflow of combining the different tools, the realistic results that were achieved indicate that this is an effective method for investigating and optimizing a hybrid powertrain concept. The combined effect of the different tools creates an easy-to-use and powerful environment for comprehensive development and optimization of a hybrid electric powertrain concept. The automated evaluation of multiple simulations allows good insight into a highly complex system to understand its dependencies. Applying evolutionary optimization algorithms helped to find the optimal settings required to meet pre-defined performance goals.

## References

- C. Appel, S. Sterzing-Oppel, J. Gerstenberg, C. Donn. Comprehensive and crossdomain vehicle simulation for the electrification of sports cars. Graz Symposium Virtual Vehicle, Graz, 2015.
- S. Bai, J. Maguire and H. Peng. Dynamic Analysis and Control System Design of Automatic Transmissions, Warrendale, Pennsylvania, USA: SAE International, 2013.
- M. Chen and G.A. Rincón-Mora. Accurate electrical battery model capable of predicting runtime and I-V performance. *IEEE Transactions of Energy Conversion*, Vol. 21, No. 2, 2006.
- T.-S. Dao, C.P. Vyasayani, J. McPhee. Simplification and order reduction of lithium-ion battery model based on porous-electrode theory. *Journal of Power Sources*, 01/2012, page 329–337, DOI: 10.1016/j.jpowsour.2011.09.034, 2012.



- C. Donn, M. Folie, V. Bensch, J. Friebe, J. Spike, P. Goossens and C. Schwarz. Concept analysis & system design of a hybrid electric vehicle with virtual test driving. 15th Stuttgart International Symposium Automotive and Engine Technology, Stuttgart, 2015.
- T. Hofman, R.M. van Druten, A.F.A Serrarens and J. van Baalen. A fundamental case study on the Prius and IMA drivetrain concepts. 21st Worldwide International Battery, Hybrid and Fuel Cell Electric Vehicle Symposium (EVS-21), Monaco, 2005.
- M. Kobayashi and C. Donn. Fusion of simulation and testing within the automotive development process using an open integration and test platform. 2015 JSAE Annual Congress, Yokohama, 2015.
- O. Noshin, B. Verbrugge, G. Mulder, P. van den Bossche, J. van Mierlo, M. Daowd, M. Dhaens and S. Pauwels. Evaluation of performance characteristics of various lithium-ion batteries for use in BEV application, Vehicle Power and Propulsion Conference (VPPC), IEEE, Lille, France, 2010.
- M. Otter, H. Elmqvist, S. E. Mattsson. Multidomain Modeling with Modelica. *Handbook of Dynamic System Modelling*, Chapman & Hall/CRC, chapter 36, pp. 36.1 - 36.27, 2007.