Automated dynamic modeling of arbitrary hybrid and electric drivetrain topologies

Citation for published version (APA):

Document license:
TAVERNE

DOI:
10.1109/TVT.2018.2834537

Document status and date:
Published: 01/08/2018

Document Version:
Publisher’s PDF, also known as Version of Record (includes final page, issue and volume numbers)

Please check the document version of this publication:
• A submitted manuscript is the version of the article upon submission and before peer-review. There can be important differences between the submitted version and the official published version of record. People interested in the research are advised to contact the author for the final version of the publication, or visit the DOI to the publisher's website.
• The final author version and the galley proof are versions of the publication after peer review.
• The final published version features the final layout of the paper including the volume, issue and page numbers.

Link to publication

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

• Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
• You may not further distribute the material or use it for any profit-making activity or commercial gain
• You may freely distribute the URL identifying the publication in the public portal.

If the publication is distributed under the terms of Article 25fa of the Dutch Copyright Act, indicated by the “Taverne” license above, please follow below link for the End User Agreement:
www.tue.nl/taverne

Take down policy
If you believe that this document breaches copyright please contact us at:
openaccess@tue.nl
providing details and we will investigate your claim.
Automated Dynamic Modeling of Arbitrary Hybrid and Electric Drivetrain Topologies

Wilco van Harselaar, Theo Hofman, and Markus Brouwer

Abstract—The optimal design regarding energy efficiency of hybrid and electric drivetrains is a problem that includes topology generation, topology optimization, component sizing, and control. One of the challenges of this design problem is that the topology defines the plant sizing and control parameters. Furthermore, the topology also determines the transmission model that is needed for the evaluation of the sizing and control problems. To enable automated component sizing and control optimization, a novel method is presented in this paper for the automated dynamic modeling of arbitrary hybrid and electric drivetrain topologies. In this paper, topologies are modeled at the level of (planetary) gears and clutches, making the method suitable for complex and unconventional drivetrain topologies. A generic transmission model is defined, for which the model parameters are automatically determined. The parameter determination is based on the analytic evaluation of the kinematic and kinetic properties of the components of the topology. All transmission modes are identified and classified, and infeasible and redundant modes are automatically excluded. As the method is analytic, the computation time to determine the model parameters is short (less than a second). The model, in this case, provides the rotational speeds and torques of all power sources as a function of the rotational speed and torque at the wheels, and the control variables. Using a case study, it is shown that the method can be used to automatically solve the control and sizing problems for complex drivetrain topologies.

Index Terms—Automated modeling, hybrid electric vehicles, powertrain design, drivetrain topology, transmission, multi-level optimal design, multi-mode, power-split, dynamic programming.

I. INTRODUCTION

The research and development of the automotive industry is currently focused on the reduction of pollutants, CO2 emissions, and fuel consumption. At the same time, vehicles must continue to meet high standards of comfort and performance. One of the solutions to this complex challenge is the development of hybrid and electric drivetrains. The design problem of such drivetrains includes the selection of the topology, and the sizing of the components. Here, the topology defines which and how many power sources the drivetrain contains, and how these power sources are connected with the output, i.e., the wheels of the vehicle. The component sizes include the maximum power of the power sources and transmission ratios.

The objective of a drivetrain design problem is often the minimization of electrical energy consumption in case of an electric vehicle, and the minimization of fuel consumption in case of a hybrid electric vehicle (HEV). To assess the energy or fuel consumption of a vehicle over a drive cycle, control is needed. This control can concern the selection of the transmission mode, and the selection of rotational speeds and torques of the power sources for every time step. Together, the topology generation, topology optimization, component sizing, and control form the system-level design (SLD) problem [1]. As the SLD problem of a hybrid drivetrain has a large design space, automated methods are needed to find the optimal design.

One of the challenges for the automated solving of the SLD problem is the influence of the topology selection on the component sizing and control problems. The topology defines which components the drivetrain includes, and thus need to be sized. Furthermore, to enable component sizing and control design, a transmission model is needed. This transmission model describes the kinematic and kinetic relations between the components for all transmission modes of the topology. Mathematical transmission models are a well explored research area [2], [3], yet in literature they are only rarely used for automated control of hybrid drivetrains. In [4] automated modeling, mode screening, mode classification, and control of a double planetary gear set (PGS) topology with one internal combustion engine (ICE), two electric machines (EMs), three clutches, and
one brake is presented. The connections of the components with the PGSs and the positions of the clutches and the brake are arbitrary, yet only topologies consisting of two PGSs are considered. A method for the model generation based on an object-oriented library is proposed in [5]. This method consists of collecting the topology equations, finding a causal flow of those equations, and selecting independent control variables, and is demonstrated for a series and for a parallel hybrid topology.

In this paper, a method is presented to automatically generate the transmission model for arbitrary hybrid and electric drivetrains. The proposed method consists of the definition of a generic transmission model, and the automated determination of the parameters for that model. This is schematically shown in Fig. 1. This figure also illustrates the contribution of this work to the SLD of hybrid drivetrains: it provides a solution to the influence of the topology selection on the component sizing and control sub-problems. For this research, an arbitrary drivetrain is defined as any drivetrain which is built from the library of components as shown in Table I, which will be further explained in the next section. The library of components shows two important facts: first that the developed method is suitable for topologies containing up to one ICE and two EMs, and second that topologies are modeled at the level of (planetary) gears and clutches, rather than viewing the gearbox as just one component. By modeling based on individual components, the presented method can be used for any possible drivetrain built from the components listed in Table I. This means that the presented method is also flexible to unconventional and complex (e.g., multi-mode) hybrid drivetrain topologies, and full electric topologies. As this research does not include the generation of topologies, the maximum number of instances does not need to be limited for every component type.

The remaining sections of this paper are organized as follows. In Section II hybrid drivetrain design is discussed by explaining the SLD problem, HEV functionalities, and topology modeling and representation. Section III presents the generic transmission model, and the method for automated parameter determination for that model is proposed in Section IV. The method is demonstrated by means of a case study in Section V, where the automated transmission modeling is used to optimize the control for two distinct hybrid drivetrain topologies. This paper concludes with a discussion in Section VI, and a conclusion in Section VII.

### Nomenclature

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>a</td>
<td>Column vector from matrix A</td>
</tr>
<tr>
<td>g</td>
<td>Inequality constraint</td>
</tr>
<tr>
<td>h</td>
<td>Equality constraint</td>
</tr>
<tr>
<td>i</td>
<td>Transmission ratio</td>
</tr>
<tr>
<td>i</td>
<td>Vector of transmission ratios</td>
</tr>
<tr>
<td>j, k</td>
<td>Index variables</td>
</tr>
<tr>
<td>l</td>
<td>Number of inequality constraints</td>
</tr>
<tr>
<td>m</td>
<td>Number of equality constraints</td>
</tr>
<tr>
<td>r</td>
<td>Row vector</td>
</tr>
<tr>
<td>s</td>
<td>Slope</td>
</tr>
<tr>
<td>t</td>
<td>Time</td>
</tr>
<tr>
<td>v</td>
<td>Velocity</td>
</tr>
<tr>
<td>x</td>
<td>Optimization parameter</td>
</tr>
<tr>
<td>z</td>
<td>Vector of optimization parameters</td>
</tr>
<tr>
<td>A</td>
<td>Matrix of a set of linear equations</td>
</tr>
<tr>
<td>B</td>
<td>Set of power sources</td>
</tr>
<tr>
<td>C</td>
<td>Element of C</td>
</tr>
<tr>
<td>C</td>
<td>Set of power source combinations</td>
</tr>
<tr>
<td>D</td>
<td>Set of power sources</td>
</tr>
<tr>
<td>E</td>
<td>Edge</td>
</tr>
<tr>
<td>E</td>
<td>Set of edges</td>
</tr>
<tr>
<td>J</td>
<td>Cost function</td>
</tr>
<tr>
<td>J</td>
<td>Set of indexes</td>
</tr>
<tr>
<td>J</td>
<td>Set of topologies</td>
</tr>
<tr>
<td>L</td>
<td>Library of components</td>
</tr>
<tr>
<td>P</td>
<td>Power</td>
</tr>
<tr>
<td>P</td>
<td>Power</td>
</tr>
<tr>
<td>Q</td>
<td>Domain of x</td>
</tr>
<tr>
<td>Q</td>
<td>Number of instance</td>
</tr>
<tr>
<td>R</td>
<td>Number of index variables</td>
</tr>
<tr>
<td>S</td>
<td>Set of topologies</td>
</tr>
<tr>
<td>T</td>
<td>Topology</td>
</tr>
<tr>
<td>V</td>
<td>Node</td>
</tr>
<tr>
<td>V</td>
<td>Set of nodes</td>
</tr>
<tr>
<td>(\xi)</td>
<td>State</td>
</tr>
<tr>
<td>(\tau)</td>
<td>Torque</td>
</tr>
<tr>
<td>(\phi)</td>
<td>Component type</td>
</tr>
<tr>
<td>(\omega)</td>
<td>Rotational speed</td>
</tr>
<tr>
<td>(\omega)</td>
<td>Vector of rotational speeds</td>
</tr>
<tr>
<td>A</td>
<td>Drive cycle</td>
</tr>
<tr>
<td>(\Psi)</td>
<td>Mode type</td>
</tr>
<tr>
<td>((\uparrow))</td>
<td>Maximum</td>
</tr>
<tr>
<td>((\downarrow))</td>
<td>Minimum</td>
</tr>
<tr>
<td>Subscripts</td>
<td></td>
</tr>
<tr>
<td>c</td>
<td>Control</td>
</tr>
<tr>
<td>ca</td>
<td>Carrier</td>
</tr>
<tr>
<td>cb</td>
<td>Clutches and brakes</td>
</tr>
<tr>
<td>cdv</td>
<td>Control decision variable</td>
</tr>
<tr>
<td>cl</td>
<td>Closed</td>
</tr>
<tr>
<td>com</td>
<td>Combinations</td>
</tr>
<tr>
<td>e</td>
<td>Element</td>
</tr>
<tr>
<td>em</td>
<td>Electric machine</td>
</tr>
<tr>
<td>g</td>
<td>Ground</td>
</tr>
<tr>
<td>gp</td>
<td>Gear pair</td>
</tr>
<tr>
<td>ice</td>
<td>Internal combustion engine</td>
</tr>
<tr>
<td>p</td>
<td>Plant</td>
</tr>
<tr>
<td>pgs</td>
<td>Planetary gear set</td>
</tr>
<tr>
<td>pr</td>
<td>Primary</td>
</tr>
<tr>
<td>ps</td>
<td>Power source / output</td>
</tr>
<tr>
<td>red</td>
<td>Reduced</td>
</tr>
<tr>
<td>ri</td>
<td>Ring</td>
</tr>
<tr>
<td>se</td>
<td>Secondary</td>
</tr>
<tr>
<td>su</td>
<td>Sun</td>
</tr>
<tr>
<td>vn</td>
<td>Virtual node</td>
</tr>
<tr>
<td>w</td>
<td>Wheels</td>
</tr>
</tbody>
</table>

### Table I: Library of Components

<table>
<thead>
<tr>
<th>Component type, (\phi)</th>
<th>Abbreviation</th>
<th>Max number of instances</th>
<th>Number of edges</th>
</tr>
</thead>
<tbody>
<tr>
<td>1: Internal combustion engine</td>
<td>ICE</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>2: Electric machine</td>
<td>EM</td>
<td>2</td>
<td>1</td>
</tr>
<tr>
<td>3: Wheels</td>
<td>W</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>4: Gear pair</td>
<td>GP</td>
<td>n.l.</td>
<td>2</td>
</tr>
<tr>
<td>5: Planetary gear set</td>
<td>PGS</td>
<td>n.l.</td>
<td>3</td>
</tr>
<tr>
<td>6: Clutch</td>
<td>C</td>
<td>n.l.</td>
<td>2</td>
</tr>
<tr>
<td>7: Clutch pair</td>
<td>CP</td>
<td>n.l.</td>
<td>3</td>
</tr>
<tr>
<td>8: Brake</td>
<td>B</td>
<td>n.l.</td>
<td>1</td>
</tr>
<tr>
<td>9: Ground</td>
<td>G</td>
<td>n.l.</td>
<td>1</td>
</tr>
<tr>
<td>10: Virtual node</td>
<td>VN</td>
<td>n.l.</td>
<td>3</td>
</tr>
</tbody>
</table>

Detailed explanation in Section II. n.l. = not limited
II. HYBRID DRIVETRAIN DESIGN

The automated transmission modeling method presented in this work is developed to contribute to solving the SLD problem of hybrid and electric drivetrains. As the functionalities of electric vehicles can be considered a subset of the functionalities of HEVs, this section focuses on hybrid drivetrains without the loss of generality.

A. The System-Level Design Problem

In general, an optimization problem consists of the optimization parameters $x$ with their domain $X$, cost function $J(x)$, inequality constraints $g(x)$, and equality constraints $h(x)$. For the SLD of a hybrid drivetrain, the optimization parameters can be divided in the topology variable $x_T$, component sizes $x_p$, and control variables $x_c$:

$$x(t) = [x_T \ x_p \ x_c(t)]^T$$

where subscripts $p$ and $c$ denote plant and control variables, respectively. Variable $x \in \mathbb{N}$ yields the discrete selection of a topology by indexing the feasible set of topologies $T^f$. Vectors $x_p$ and $x_c$ can contain both discrete and continuous variables. The general SLD problem for hybrid and electric drivetrains can be defined as:

$$\min_{\mathbf{x}(t) \in \mathbf{X}} J(\mathbf{x}(t), \Lambda(t))$$

s.t. $g_j(x(t)) \leq 0, \ j = 1, \ldots, l,$

$$h_k(x(t)) = 0, \ k = 1, \ldots, m,$$

with $l$ the number of inequality constraints, $m$ the number of equality constraints, and $\Lambda$ the drive cycle consisting of velocity $v$, and slope $s$ over time:

$$\Lambda(t) = [v(t) \ s(t)]^T$$

Dynamic optimization problems typically contain states, $\xi$, which can be expressed as:

$$\dot{\xi}(t) = f(\xi(t), x(t), t)$$

For hybrid drivetrain design, the state of charge (SOC) of the battery is an example of a controlled state. When the complete SLD problem is defined with detailed enough models for the simultaneous optimization of topology, component sizes, and control, it is not possible to solve the resulting optimization problem within a practical time limit. The main reasons for the high computation time are the fact that the problem is nonlinear and non-convex, the problem has a high number of optimization parameters, and a large design space [1], [6]. Theoretically, all parameters of the SLD problem can be optimized simultaneously. However, as this problem contains both time dependent and time independent optimization variables, simultaneous optimization is computationally inefficient. To avoid the need of simultaneous optimization, the problem can be divided in sub-problems. In most recent literature, the distinction is made between a plant design problem and a control problem, with cost functions $J_p$ and $J_c$, respectively. Furthermore, the design problem can be divided in topology selection and component sizing. These individual problems can be solved by applying one of the two main classes of distributed system optimization (i.e., coordination): nested optimization and alternating optimization [7]. As mentioned before, the drivetrain topology defines the optimization parameters of the sizing and control sub-problems. The control parameters influence the optimal solution for the design problem, yet they do not influence the structure of the optimization parameters of the design problem. This unidirectional coupling between the structures of the optimization parameters of the sub-problems allows for a nested optimization approach. In [8], [9], the coupling between plant and control optimization problems is specifically addressed.

The different layers of the SLD problem, together with the links between the levels are shown in Fig. 2. The highest level is the topology generation, where all feasible topologies $T^f$ are found from the set of possible topologies $T^p$. In [10] a computational design synthesis (CDS) framework is presented to automatically generate HEV drivetrain topologies and find the feasible ones using a constraint satisfaction problem (CSP). Topology generation is applied for the synthesis of gearbox designs in [11], [12]. Fig. 2 also shows the influence of the topology selection on the sizing and control problems. This paper presents a solution to that link between the topology optimization, component sizing, and control layers.

B. Functionalities of HEVs

As an HEV contains at least one ICE and at least one electric machine (EM) [13], more system-level functionalities are enabled compared to conventional (ICE) or electric vehicles. Which functionalities are enabled depends on the topology of the drivetrain. In [10] seven principle system-level modes, which can be enabled by a hybrid powertrain, are defined. In the current research, one extra mode is defined, and the definitions are
reformulated to ensure that series hybrid driving is also clearly defined. For the ease of reading, first only the modes that allow for a power flow (positive or negative) to the wheels are listed. Note that the battery power is defined positive for discharging. The order of listing matches the typical usage of the modes in order of the power demand at the wheels; from negative during regenerative braking to maximum power during motor assisted driving:

1) **Regenerative braking**: the process of recovering kinetic energy of the vehicle during deceleration and converting it to electric energy for direct use or storage in a battery.

2) **Electric only**: the situation in which the ICE does not deliver power, and one or more EMs are used for the propulsion of the vehicle. It is assumed that this mode requires the rotational speed of the ICE to be zero.

3) **Charging mode**: refers to hybrid driving with a negative battery power. In this situation, the ICE delivers power for both the propulsion of the vehicle and to charge the battery. This is also referred to as load point shifting.

4) **Engine only**: the situation in which the ICE delivers power for the propulsion of the vehicle, and the battery power is zero. In this mode the power of the individual EMs does not need to be zero, yet the summation of the electric power of the EMs is zero.

5) **Motor assist**: refers to the situation in which both the ICE and the battery are delivering power for the propulsion of the vehicle, also referred to as boosting.

The difference between the hybrid driving modes 3, 4, and 5 is determined by the battery power. Series hybrid driving is also included in these three modes, as a mechanical connection between the ICE and the wheels is not required to fulfill the specifications of these hybrid driving modes. In addition to the above explained modes, three special modes are defined which do not describe a power flow to or from the wheels:

6) **Start-stop**: refers to the functionality of switching off the ICE during standstill, and turning it back on when needed.

7) **Recharge**: is a special mode referring to the process of using an energy grid to recharge the battery of the vehicle.

8) **Charging during standstill**: allows to charge the battery of the vehicle with the ICE through an EM without providing power for propulsion.

This last mode is extra compared to [10]. Enabling some or all of the listed system-level modes potentially decreases the fuel consumption of the vehicle. Furthermore, the comfort might also be improved. The electric only mode, for example, can improve the comfort compared to a conventional ICE vehicle by removing the ICE noise and vibrations.

### C. Topology Modeling and Representation

To enable automated modeling for hybrid and electric drivetrain topologies, these topologies need to be defined in a unified way. A method to define a drivetrain topology as an undirected connected fine graph is presented in [10]. In this method, a topology $T$ consists of nodes, $V$, and edges, $E$. Nodes represent components, and edges the connections between components.

All components considered in this research are listed in the library of components in Table I. Every node type has a fixed number of edges. A gear pair, for example, has two edges: one representing its primary side and one its secondary side. An EM has one edge, as its rotor is the only mechanical output. To connect more than two edges together, node type virtual node (VN) ($\phi = 10$) is defined [10]. Next to the VN, the ground ($\phi = 9$) is the second node type that does not directly translate to a physical component and represents a fixed connection with the transmission housing. A clutch pair ($\phi = 7$) is a set of two clutches of which only one can be closed at the same time.

Each node $V \in V$ is characterized by its type $\phi$, as listed in Table I, and its number of instance $\iota$. This information is included as subscript to each node: $V_{\phi,\iota}$. Each edge $E \in E$ is a two-element subset of the set of nodes ($E \subseteq V$). Although edges are sets of two elements, it is chosen to denote edges with an $E$ that is not bold to enable the usage of $E$ for the set of all edges of a topology. In the current research, only mechanical edges (i.e., connections) are considered, yet the described method allows to extend this by also assigning other types of ports to components, e.g., electrical and thermal ports. Fig. 3(a) shows a graph diagram of a single gear electric drivetrain, consisting of four nodes and three edges. Applying the explained definition, this topology is written as:

$$T = (V, E)$$

with

$$V = \{V_{2,1}, V_{4,1}, V_{4,2}, V_{3,1}\}$$

$$E = \{\{V_{2,1}, V_{4,1}\}, \{V_{4,1}, V_{4,2}\}, \{V_{4,2}, V_{3,1}\}\}$$

As the level of modeling of the drivetrain topology differs compared to [10], the library of components also differs. In [10], the gearbox is modeled as one component with two edges, whereas in the current research the sub-components of the gearbox (e.g., gear pairs and clutches) are modeled individually without declaring a part of the topology as gearbox. Despite this difference, the same method can be applied to mathematically define the topology. Furthermore, applying the method presented in this paper, the transmission model can also be automatically generated for topologies resulting from the work of [10] when the gearbox is defined in its subcomponents.

---

$^1$Examples of components that are not included in the library of components for this research are chains, belts, one-way clutches, stepped pinion PGSs, Ravignieux PGSs, and variators. Furthermore, the front and rear axles of the vehicle are merged together in component type wheels ($\phi = 3$). The method presented in this work can be extended to also enable the inclusion of these components. The influence of including a variator to the control problem is addressed in the discussion section of this paper.
Topologies can also be represented by a stick diagram. Fig. 3(b) shows the stick diagram of the topology defined by (5). Let us define an element as a part of the drivetrain which could rotate independently when the restrictions of gears, clutches, and brakes are disregarded. Then, in graph notation, every edge which does not connect a VN represents a unique element. Furthermore, every VN merges three edges into one element. Therefore, the number of elements $N_e$ of a topology can directly be extracted from $T$:

$$N_e = |E| - 2 \cdot N_{vn}$$

(6)

where $|E|$ is the cardinality of $E$, and $N_{vn}$ the number of VN’s. To enable the construction of linear equations, as discussed in the next section, the elements of the topology need to be numbered. An example of the division of a drivetrain into elements, and the numbering of these elements is visualized in Fig. 4(a). Note that the numbering of elements is arbitrary and does not influence the resulting transmission model.

III. THE GENERIC TRANSMISSION MODEL

A transmission model is needed for the evaluation of the cost functions of the component sizing and control problems. This model describes the torques and rotational speeds of the power sources as a function of the torque and speed demands at the wheels, and the control variables $x_c$. To obtain these transmission models for arbitrary topologies, a generic model is formulated. The topology-dependent parameters for this model are automatically determined, as explained in the next section. Topologies that are built from the defined library of components can contain 14 different mode types, which are denoted by symbol $\Psi$, and listed in Table II.

**Definition 1:** Transmission mode types are defined by the dynamic behavior of the drivetrain system and subsequently by the required control decision variables (CDVs) for that mode type.

Every transmission mode can be classified to one mode type. When a topology has multiple distinct transmission modes, these modes can be of the same mode type or of several distinct mode types. The generic model includes the dynamic behavior of all 14 possible transmission mode types. To name the different mode types, the terms fixed gear (FG), electric variable transmission (EVT), parallel, and series are used. FG refers to a transmission mode in which all rotational speeds are linearly dependent on each other. EVT refers to the possibility of controlling rotational speeds for one or multiple power sources for a given rotational speed of the wheels $\omega_w$. The terms parallel and series refer to the common classification of hybrid electric drivetrains, see e.g., [6].

For the generic model it is defined that the first control variable, $x_{c,1}$, is a discrete variable that determines the transmission mode. All subsequent control variables are continuous variables which are scaled to be in the interval $[0, 1]$. Which quantities are controlled by the continuous control variables is dependent on the mode type. The number of continuous control variables is dependent on the set of mode types that the topology enables, and can vary from zero to three for topologies based on the library of components defined in Table I.

In case of a topology with only one transmission mode (e.g., the EV example in Fig. 3), control variable $x_{c,1}$ has only one discrete value. This theoretically leads to $x_{c,1}$ not being a control variable anymore, as it becomes a constant. To ensure that the model is generic, $x_{c,1}$ is kept in this case, and $x_{c,2}$ stays the first continuous control variable.

Table II lists the variables that are controlled by the continuous control variables for each transmission mode type. Note that for modes of type $\Psi \in \{7, 11, 12, 13\}$ control variable $x_{c,2}$ determines the output power of the ICE, $P_{ice} = \tau_{ice} \cdot \omega_{ice}$. In these four mode types, there is no mechanical connection between the ICE and the wheels. Therefore, the ICE power is converted to electric power by one or two EMs. A fixed gear charging mode with one EM ($\Psi = 11$) has principally two continuous control variables: the torque and the speed of the concerning components. For mode type $\Psi = 12$ one extra variable is added as the ICE torque can be divided over the two EMs. To reduce computational effort, let us discretize $P_{ice}$. Additionally, let the charging efficiency be defined as the efficiency of the conversion from chemical energy of the fuel to electrical energy delivered to the battery. For each discrete value of $P_{ice}$, all charging modes have one combination of component torques and speeds that leads to the highest charging efficiency. Additionally, for each value of

<table>
<thead>
<tr>
<th>Mode Type, $\Psi$</th>
<th>$x_{c,2}$</th>
<th>$x_{c,3}$</th>
<th>$x_{c,4}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>ICE only</td>
<td>$-1$</td>
<td>$-1$</td>
<td>$-1$</td>
</tr>
<tr>
<td>EM only</td>
<td>$-1$</td>
<td>$-1$</td>
<td>$-1$</td>
</tr>
<tr>
<td>Hybrid</td>
<td>$-1$</td>
<td>$-1$</td>
<td>$-1$</td>
</tr>
<tr>
<td>5: FG parallel 1 EM</td>
<td>$\tau_{e,c} (x_{c,2})$</td>
<td>$\omega_{e,m} (x_{c,2})$</td>
<td></td>
</tr>
<tr>
<td>6: FG parallel 2 EMs</td>
<td>$\tau_{e,m} (x_{c,2})$</td>
<td>$\omega_{e,m} (x_{c,2})$</td>
<td></td>
</tr>
<tr>
<td>7: Series</td>
<td>$P_{e,c} (x_{c,2})$</td>
<td>$-1$</td>
<td></td>
</tr>
<tr>
<td>8: EVT 1 EM</td>
<td>$\omega_{e,c} (x_{c,2})$</td>
<td>$-1$</td>
<td></td>
</tr>
<tr>
<td>9: EVT 2 EMs 2 CDVs</td>
<td>$\tau_{e,c} (x_{c,2})$</td>
<td>$\omega_{e,c} (x_{c,2})$</td>
<td></td>
</tr>
<tr>
<td>10: EVT 2 EMs 5 CDVs</td>
<td>$\omega_{e,c} (x_{c,2})$</td>
<td>$\tau_{e,c} (x_{c,2})$</td>
<td>$\omega_{e,m} (x_{c,4})$</td>
</tr>
<tr>
<td>Charge (No propulsion possible)</td>
<td>$P_{e,c} (x_{c,2})$</td>
<td>$-1$</td>
<td>$-1$</td>
</tr>
<tr>
<td>Neutral</td>
<td>$-1$</td>
<td>$-1$</td>
<td>$-1$</td>
</tr>
<tr>
<td>11: FG 1 EM</td>
<td>$P_{e,c} (x_{c,2})$</td>
<td>$-1$</td>
<td></td>
</tr>
<tr>
<td>12: FG 2 EMs</td>
<td>$P_{e,c} (x_{c,2})$</td>
<td>$-1$</td>
<td></td>
</tr>
<tr>
<td>13: EVT 2 EMs</td>
<td>$P_{e,c} (x_{c,2})$</td>
<td>$-1$</td>
<td></td>
</tr>
</tbody>
</table>
as explained

\[ P_{\text{ICE}}(1) \Psi_0 \quad \text{for electric and} \quad x + \tau, \ \text{Distinct series modes cannot be combined in} \quad \tau_x \in \{ \Lambda \Psi, \ \text{and ICE torque ratio} \quad \text{modes} \quad i_{\text{ICE}} \cdot \text{ICE}(8) \]

\[ \Psi(\Lambda) = \max (x, \Psi) \quad \text{as defined for modes of EVT} \quad (\Lambda) \]

\[ \Psi(\Lambda) = \min (x, \Psi) \quad \text{if} \quad \tau \]

\[ \tau \quad \text{being a function of} \quad \tau_{\text{ICE}} \quad \text{and} \quad \text{ICE} \quad \text{torque ratios are fixed.} \quad \tau \]

\[ \tau \quad \text{with} \quad \tau = \min (\tau_{\text{ICE}}(\omega_{\text{ICE}}(x_{e,1})), \omega_{\text{w}}(\Lambda), i_{\text{ICE}}(x_{e,1}), x_{e,2})) \quad \text{(8)} \]

\[ \tau = \max (\tau_{\text{ICE}}(\omega_{\text{ICE}}(\Psi(x_{e,1})), \omega_{\text{w}}(\Lambda), i_{\text{ICE}}(x_{e,1}), x_{e,2})) \quad \text{(9)} \]

and ICE transmission ratio \( i_{\text{ICE}} \), and ICE torque ratio \( i_{\tau,\text{ICE}} \). The first case of (7) relates \( \tau_{\text{ICE}} \) with \( i_{\text{ICE}} \) to \( \tau_w \) for ICE only modes in which the ICE has a fixed gear ratio to the wheels, with \( \tau_w \)

depending on the drive cycle \( \Lambda \). Both \( \tau_{\text{ICE}} \) and \( \Psi \) are dependent on \( x_{e,1} \), as this control variable directly determines the transmission mode. The second case of (7) states that \( \tau_{\text{ICE}} = 0 \) for electric and neutral modes. The third and fourth cases of (7) show that \( \tau_{\text{ICE}} \) is directly controlled by a continuous control variable in modes of type \( \Psi \in \{5, 6, 9, 10\} \), as also defined in Table II. The fifth case of (7) defines \( \tau_{\text{ICE}} \) as a function of \( P_{\text{ICE}} \) in series hybrid and charge modes, with \( P_{\text{ICE}} \) being a function of \( x_{e,2} \) as explained in the previous paragraphs. The last case of (7) is similar to the first, but with torque ratio \( i_{\tau,\text{ICE}} \) as defined for modes of EVT type \( \Psi = 8 \), where the torque ratios are fixed.

The definitions of the torques and speeds of all power sources are not provided here for the reasons of space, yet are for the interested reader further elaborated on in [14].

IV. AUTOMATED MODEL PARAMETER DETERMINATION

A novel method is developed to automatically determine the parameters for the generic transmission model presented in the previous section. The method is based on the kinetic and kinematic properties of the components of the topology, and divided in three steps: (1) pre-processing, where sets of linear equations are constructed that describe the kinetic and kinematic relations of the topology, (2) processing, where the kinematic and kinetic relations are used to identify mechanical connections, identify CDVs, and exclude infeasible modes, and (3) post-processing, where the feasible transmission modes are classified and redundant modes are excluded. This process is visualized in Fig. 5. On the right side of the flow chart, the development of an elementary example topology along the process is shown.

A. Pre-Processing

The first step of the parameter determination is the pre-processing. The main results of this step are the sets of linear equations that describe the kinetics and kinematics of the drivetrain. As shown in Fig. 5, the pre-processing step is divided in...
three sub-steps. The first of these sub-steps is the determination of the feasible clutch combinations, step (1a). A clutch combination defines which clutches and brakes are closed, which means that every clutch combination can define a mode of the transmission. Note that a feasible clutch combination does not necessarily lead to a feasible transmission mode. The feasibility of transmission modes is analyzed in the processing step. As the clutch combination influences the kinematics of the transmission, every transmission mode has its unique sets of equations. The number of variables for the kinematic equations is equal to the number of elements \( N_e \). To make sure the system is not overdetermined in case an output speed is given, the maximum number of kinematic equations is \( N_e - 1 \). Each gear pair, PGS, and ground provides one kinematic equation, independent from the clutch combination. Therefore, for a given topology, the number of fixed kinematic equations is known. As every clutch or brake which is closed adds a kinematic equation, the maximum number of clutches and brakes that can be closed at the same time \( N_{cb,cl} \) can be determined:

\[
N_{cb,cl} = N_e - N_{gp} - N_{pgs} - N_g - 1
\]

with number of gear pairs \( N_{gp} \), number of PGSs \( N_{pgs} \), and number of grounds \( N_g \). As a result, the number of clutch combinations \( N_{com,cb} \) that do not lead to an overdetermined system can be determined using the binomial coefficient:

\[
N_{com,cb} = \binom{N_{cb,cl}}{j}
\]

where \( N_{cb} \) is the total number of clutches and brakes of the topology. Subsequently, clutch pairs (\( \phi = 7 \)) are taken into account. From the two clutches of a clutch pair, only one can be closed at the same time. After excluding clutch combinations where both clutches of a clutch pair are closed, the remaining combinations form the set of feasible clutch combinations. For the example topology displayed in Fig. 5, there are two feasible clutch combinations: (1) no clutch closed, and (2) clutch C1 closed, as shown at step (1a) in that figure.

Step (1b) of the parameter determination is to construct the sets of kinetic and kinematic equations for every clutch combination. Besides the fixed kinematic equations from PGSs, gear pairs, and grounds, additional kinematic equations are given by closed clutches and closed brakes. These equations are defined using the example topology displayed in Fig. 4(a); let \( z_{su} \), and \( z_{ri} \), be the tooth counts of the sun gear and ring gear of planetary gear set PGS1 in that figure. Furthermore, let \( \tau_{gp} \) be the transmission ratio of gear pair GP1, and note that a ground provides the same equation as a closed brake. Then, PGS1, GP1, C1 closed, and B1 closed, provide the following kinematic relations:

\[
-z_{su} \omega_4 - (z_{ri} + z_{su}) \omega_3 + z_{ri} \omega_2 = 0
\]

\[
\omega_4 - \tau_{gp} \omega_5 = 0
\]

\[
\omega_1 - \omega_2 = 0
\]

\[
\omega_2 = 0
\]

The set of kinematic equations of a clutch combination is represented in matrix form by defining matrix \( A_c \):

\[
A_c \omega = 0
\]

where \( \omega \) is the vector with the rotational speeds of all elements of the transmission:

\[
\omega = [\omega_1 \omega_2 \ldots \omega_{N_e}]^T.
\]

The kinetic equations of a clutch combination are also represented in matrix form by defining matrix \( A_r \):

\[
A_r \tau = 0
\]

where \( \tau \) contains the torques of the edges of PGSs, gear pairs, the ICE, EMs, and the wheels (i.e., output). Let \( N_{ice} \) and \( N_{em} \) be the numbers of ICEs and EMs, respectively. Then, the number of entities in vector \( \tau \), \( N_r \), is:

\[
N_r = 3N_{pgs} + 2N_{gp} + N_{ice} + N_{em} + 1
\]

where the factors 3 and 2 correspond to the number of edges of PGSs and gear pairs, respectively. For the topology in Fig. 4(a), this vector becomes:

\[
\tau = [\tau_{pgs, su} \tau_{pgs, ca} \tau_{pgs, ri} \tau_{gp, pr} \tau_{gp, se} \tau_{ice} \tau_{em1} \tau_{em2} \tau_w]^T
\]

where subscripts pr and se denote primary and secondary, and subscripts su, ca, and ri denote sun, carrier, and ring.

For the kinetic equations, each PGS provides two equations and each gear pair provides one. Consider again PGS1 with \( z_{su} \), and \( z_{ri} \), and GP1 with \( \tau_{gp} \) from Fig. 4(a). These components provide the following kinetic equations:

\[
\tau_{pgs, su} + \tau_{pgs, ca} + \tau_{pgs, ri} = 0
\]

\[
z_{ri} \tau_{pgs, su} + z_{su} \tau_{pgs, ri} = 0
\]

\[
\tau_{gp, pr} + \tau_{gp, se} = 0
\]

Additional kinetic equations are dependent on the states of clutches and brakes. Let edges be grouped by VNs and closed clutches. For every group of edges, the sum of the torques is zero. External torques to the system can come from power sources, the wheels, grounds, and closed brakes. The torques of grounds and closed brakes are a function of the torques of the power sources and the wheels. Therefore, the torques of grounds and closed brakes do not need to be specified to obtain the torques of the power sources as a function of \( \tau_w \) and \( x_c \), and are not included in \( \tau \) as shown by (20). For groups of edges that are not subject to the torque of a ground or closed brake, an equation is added that states that the sum of the torques of that group is zero. As an example, consider the topology of Fig. 4(a) with C2 closed, and C1 and B1 both open. Then, (18) becomes:

\[
\begin{bmatrix}
1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & \tau_{ri} & 0 & z_{su1} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{bmatrix}
\tau = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}
\]
with $\tau$ as defined in (20). The first three rows match (21) to (23). The fourth row states that $\tau_{\text{ice}}$ is zero, as the edge of the ICE forms a group on its own. The fifth row states that the sum of torques of the sun, carrier, primary side of GP1, and EM2 is zero, as they are grouped by the closed C2. The sixth row states that the sum of the torques of the ring and EM1 is zero. Finally, the seventh row states that the sum of the torques of the secondary side of GP1 and the wheels is zero.

Subsequent to the construction of the sets of equations, step (1c) is to identify non-functional CDVs, and formulate additional equations to exclude them. Two types of CDVs are distinguished: kinematic CDVs and kinetic CDVs.

**Definition 2:** a non-functional CDV occurs when an element of the transmission can have an arbitrary rotational speed or torque while that speed or torque does not support any of the system-level modes.

An example of a non-functional kinematic CDV is displayed in Fig. 4(a): when clutch C1 is open, the kinetic equations define zero as the only solution for the ICE torque, as shown by (24). Yet, the rotational speed of the ICE is a CDV according to the defined set of kinematic equations. An example of a non-functional kinetic CDV would be a power source that is directly connected to a closed brake. An example is EM1 in Fig. 4(a) when B1 is closed. The kinematics fix the rotational speed of the power source to zero, yet its torque is a CDV as the brake can theoretically compensate for any torque generated by the power source.

**Definition 3:** dependent CDVs are the rotational speeds or torques of multiple elements which are linearly dependent on each other and together are a CDV. Independent CDVs are the rotational speeds or torques of multiple elements, which are all CDVs and are not linearly dependent on each other.

Non-functional CDVs are identified by analyzing the dependent and independent CDVs of the topology. To analyze whether $\omega_j$, the rotational speed of an arbitrary element $j$, is a CDV, $\omega_j$ is defined as the $j$th column of $A_\omega$, as in the work of [2]. Subsequently, $\omega$ is reduced for element $j$ by defining:

$$\omega_{\text{red}_j} = [\omega_1 \ldots \omega_{j-1} \omega_{j+1} \ldots \omega_n]^T$$

Equally, $A_{\omega,j}$ is defined as $A_{\omega}$ without the $j$th column. This is used to rewrite (16) to:

$$A_{\omega,j} \omega_{\text{red}_j} = -\omega_{j} \omega_j$$

If $\omega_j$ is a CDV, the set of equations must be solvable for a given non-zero value of $\omega_j$. For $\omega_j = 1$, (26) becomes:

$$A_{\omega,j} \omega_{\text{red}_j} = -\omega_{j}$$

This set of equations is solvable, i.e., $\omega_j$ is a CDV, if-and-only-if the following condition holds [15]:

$$A_{\omega,j}^\dagger A_{\omega,j} \omega_{j} = -\omega_{j}$$

where $A_{\omega,j}^\dagger$ is the Moore-Penrose pseudoinverse of $A_{\omega,j}$ [16]. The pseudoinverse is used as $A_{\omega,j}$ does not need to be square, and matrix inverse $A_{\omega,j}^{-1}$ is only defined for square matrices. If condition (28) does not hold, then the only solution for $\omega_j$ in (16) is zero. Note that if $A_{\omega,j}^\dagger A_{\omega,j}$ returns the identity matrix, then the rows of $A_{\omega,j}^\dagger$ are linearly independent. This is however not necessary for condition (28) to hold.

The described method is also deployed to evaluate if the torques of the power sources and the output are CDVs by setting $\tau_j = 1$ and evaluating:

$$A_{\tau,j}^\dagger A_{\tau,j} (-a_{\tau,j}) = -a_{\tau,j}.$$  

If-and-only-if this condition holds, is $\tau_j$ a CDV.

To analyze whether the rotational speeds of an arbitrary set of elements are independent CDVs, let us define $\omega_S$ as the vector with the rotational speeds of the elements of arbitrary set $S$:

$$S = \{j_1, \ldots, j_n\}$$

$$\omega_S = [\omega_{j_1} \ldots \omega_{j_n}]^T$$

In addition, let matrix $A_{\omega,S}$ denote the negative matrix that combines the columns of $A_{\omega}$ that correspond to the elements of $S$:

$$A_{\omega,S} = [-a_{\omega,j_1} \ldots -a_{\omega,j_n}]$$

Matrix $A_{\omega,S}$ reduced by the $S^\text{th}$ columns and vector $\omega_s$ reduced by $\omega_S$ are denoted by $A_{\omega,\text{red}_S}$ and $\omega_{\text{red}_S}$, respectively. Then, for any set $S$, (16) can be rewritten to:

$$A_{\omega,\text{red}_S} \omega_{\text{red}_S} = A_{\omega,S} \omega_S$$

Using these definitions, the rotational speeds of an arbitrary set of elements, $\omega_S$, are independent CDVs if-and-only-if the following condition holds:

$$A_{\omega,\text{red}_S} A_{\omega,\text{red}_S}^\dagger A_{\omega,S} = A_{\omega,S}$$

To enable convenient notation, let us define function $\Phi(V_{\phi,i})$ that identifies the component type of node $V_{\phi,i}$ as listed in Table I:

$$\Phi(V_{\phi,i}) = \phi$$

Furthermore, function $\Upsilon(S)$ returns a logical true if the rotational speeds $\omega_S$ are independent CDVs:

$$\Upsilon = \begin{cases} 1, & \text{if } A_{\omega,\text{red}_S} A_{\omega,\text{red}_S} \omega_S = A_{\omega,S} \\ 0, & \text{otherwise} \end{cases}$$

The previous definitions are used to obtain a set $D$ with as many power sources as possible that together are independent DOF. First, set $B$ is defined as the set containing only the power sources of topology $T$:

$$B = \{V \in V | \Phi(V) \in \{1,2\}\}$$

Subsequently, $C$ contains all combinations of $B$ of which the rotational speeds are independent CDVs as subsets:

$$C = \{S \in \mathcal{P}(B) | \Upsilon(S) = 1\}$$

where $\mathcal{P}(B)$ is the power set of $B$. All elements of $C$ are sets with power sources of the considered topology: $C \subseteq B \forall C \in C$. As with $E$, each $C \in C$ is a set but not written bold to make clear that it is an element of $C$. Note that in set builder expression (38) symbol $S$ is used as variable to emphasize that the elements of the domain are sets. Let set $C \subseteq C$ contain the
elements of $C$ with the highest cardinality of all elements of $C$. To find this set, we define $\bar{N}_C$ as the cardinality of $C \in C$:

$$\bar{N}_C = \max_{j=1, \ldots, |C|} \left( |C_j| \right)$$

$$\bar{C} = \{ S \in C | |S| = \bar{N}_C \}$$

Subsequently, $D$ is defined as one of the elements of $C$ and will be used for the identification of non-functional CDVs; $D \in C$. Note that which element of $C$ is selected to be $D$ does not matter for the process and results of the parameter determination. For every clutch combination, the described methods are employed in a four step process to identify and exclude the non-functional CDVs:

1) Non-functional kinematic CDVs of power sources and output: consecutively for the output and for each power source, denoted by index $ps$, it is checked whether the rotational speed is a CDV, while the torque of that power source or output is zero. This is done by verifying whether the following condition holds:

$$A_{r,red_{ps}} A_{r,red_{ps}}^T (-a_{\omega,ps}) = -a_{\omega,ps} \land$$

$$A_{r,red_{ps}} A_{r,red_{ps}}^T (-a_{\tau,ps}) \neq -a_{\tau,ps}$$

If this is the case, then $\omega_{ps}$ is a non-functional CDV and a row is added to matrix $A_\omega$ that defines $\omega_{ps} = 0$ in the context of (16).

2) Non-functional kinematic CDVs of all other elements: consecutively for each element $j$ it is checked whether its rotational speed is an independent CDV combined with the power sources in $D$. For this purpose, set $J$ is defined as $J = \{j\} \cup D$. Thereby, their rotational speeds are defined in one vector as $\omega_j = [\omega_j \omega_D]^T$. It is checked whether $\omega_j$ are independent CDVs by verifying whether the following condition holds:

$$A_{\omega,red_j} A_{\omega,red_j}^T A_{\omega,j} = A_{\omega,j}$$

If this is the case, then $\omega_j$ is a non-functional CDV and equation $\omega_j = 0$ is added to $A_\omega$.

3) Non-functional kinematic CDVs of EMs: one special case of a non-functional kinematic CDV could still be present after the first two steps: multiple EMs which are mechanically only connected to each other. To identify this case, consecutively for every EM, denoted by index $em$, it is checked whether its rotational speed is a CDV while at the same time that EM does not have a mechanical connection to either the ICE or the output:

$$A_{\omega,red_{em}} A_{\omega,red_{em}}^T (-a_{\omega,em}) = -a_{\omega,em} \land$$

$$\frac{\omega_{em}}{\omega_{ice}} = 0 \land \frac{\omega_{em}}{\omega_{w}} = 0$$

If this is the case, then $\omega_{em}$ is a non-functional CDV and equation $\omega_{em} = 0$ is added to $A_\omega$. Mechanical connections between elements are elaborated in the next section. Note that when an ICE and EM are mechanically only connected to each other and not to the output, this still enables the functionality of charging in standstill and series hybrid driving.

4) Non-functional kinetic CDVs: by the method applied to construct the kinetic equations, non-functional kinetic CDVs can only occur at the output and power sources. The identification of these CDVs is similar to step 1): consecutively for the output and for each power source, denoted by index $ps$, it is checked whether the torque is a CDV, while the rotational speed of that power source or output is zero. This is done by verifying whether the following condition holds:

$$A_{r,red_{ps}} A_{r,red_{ps}}^T (-a_{\tau,ps}) = -a_{\tau,ps} \land$$

$$A_{\omega,red_{ps}} A_{\omega,red_{ps}}^T (-a_{\omega,ps}) \neq -a_{\omega,ps}$$

If this is the case, then $\tau_{ps}$ is a non-functional CDV and a row is added to matrix $A_\tau$ that defines $\tau_{ps} = 0$ in the context of (18).

In the example displayed in Fig. 5, the rotational speed of the ICE is a non-functional kinetic CDV when clutch C1 is open, which is identified and excluded in the first step of the process above. Next to step (1c) in Fig. 5 it is shown that an extra row is added to $A_\omega$ to define $\omega_{ice} = 0$.

### B. Processing

When the sets of kinematic and kinetic equations are defined for all clutch combinations, these equations are used in the processing step of the automated parameter determination. The processing is divided in three sub-steps. The first of these sub-steps is to identify the mechanical connections of the topology, step (2a) in Fig. 5. The mechanical connections which are of interest are the connections between the wheels and power sources, limited to connections that allow torque transfer. There is a connection that allows torque transfer between a power source and the wheels if-and-only-if the transmission ratio from the power source to the output is non-zero. To determine the transmission ratios between any element $j$ and all other elements, column vector $i_j$ is defined as:

$$i_j = \omega/\omega_j$$

Additionally, row vector $r_{\omega,j}$ is defined as:

$$r_{\omega,j} = \begin{bmatrix} r_{\omega,j}^1 & r_{\omega,j}^2 & \cdots & r_{\omega,j}^{N_j} \end{bmatrix}^T$$

with $r_{\omega,j}^k = 0 \forall k \neq j$

$$r_{\omega,j}^j = 1$$

The set of kinematic equations, as in (16), can be complemented with equation $\omega_j = \omega_j$ using vector $r_{\omega,j}$:

$$\begin{bmatrix} A_\omega \\ r_{\omega,j} \end{bmatrix} \omega = \begin{bmatrix} 0 \\ \omega_j \end{bmatrix}$$

Dividing both sides of this equation by $\omega_j$ leads to:

$$\begin{bmatrix} A_\omega \\ r_{\omega,j} \end{bmatrix} i_j = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$
The values of \( i_j \) can now be determined using the Moore-Penrose pseudoinverse:

\[
i_j = \begin{bmatrix} A_{\omega} \\ r_{x,j} \end{bmatrix}^+ \begin{bmatrix} 0 \\ 1 \end{bmatrix}
\]  

(49)

Mechanical connections between the wheels and the power sources are found by determining \( i_w \), the column vector with the transmission ratios from every element, and thus also all power sources, to the wheels. For the example in Fig. 5, the EM is connected to the output in both transmission modes. The ICE is only connected to the output in the second mode, when C1 is closed.

Step (2b) is the determination of the number of CDVs in the sets of kinematic and kinetic equations. The number of kinematic CDVs can be found by comparing \( N_e \) with the rank of \( A_\omega \), and the number of kinetic CDVs can be found by comparing \( N_\tau \) with the rank of \( A_\tau \). During vehicle operation, the rotational speed and torque on the output element follow from the vehicle velocity and torque demand induced by the driver or velocity profile in the case of simulation. Therefore, \( \omega_w \) and \( \tau_w \) are not considered CDVs. To find the number of CDVs, excluding \( \omega_w \) and \( \tau_w \), the method from [2] is edited. The \( A \) matrices are complemented with an extra row that excludes the output element as CDV by fixing its speed, or torque respectively, to zero. For transmission modes which do not enable propulsion (i.e., no mechanical connection between the output and any power source), these extra equations do not influence the CDVs related to battery charging in standstill. Applying this method, the number of kinematic and kinetic CDVs, \( N_{cdv,\omega} \) and \( N_{cdv,\tau} \), are defined as:

\[
N_{cdv,\omega} = N_e - \text{rank} \left( \begin{bmatrix} A_{\omega} \\ r_{x,w} \end{bmatrix} \right)
\]

(50)

\[
N_{cdv,\tau} = N_\tau - \text{rank} \left( \begin{bmatrix} A_{\tau} \\ r_{x,w} \end{bmatrix} \right)
\]

(51)

With \( r_{x,w} \) according to (46), and \( r_{x,w} \) defined as:

\[
r_{x,w} = \left[ r_{x,w}^1 \ r_{x,w}^2 \ \ldots \ r_{x,w}^{N_r} \right],
\]

with \( r_{x,w}^k = 0 \ \forall \ k \neq N_r \),

\[
r_{x,w}^{N_r} = 1
\]

(52)

For the topology in Fig. 5, the numbers of CDVs are displayed in the same figure. In that topology, there are no kinematic CDVs, and in the second mode there is one kinetic CDV as the torque demanded at the wheels can be divided over the two power sources.

The third sub-step of the processing is the exclusion of infeasible transmission modes. A transmission mode is declared infeasible if the mode does not support any of the system-level functionalities as defined in Section II. This can be either due to the transmission being partly blocked or due to the lack of power paths between components, i.e., the sets of equations describing an over-defined or under-defined system. As non-functional CDVs are excluded in the pre-processing step, for all infeasible transmission modes the rotational speed of all elements will be fixed to zero. This property is used to identify infeasible modes by verifying the following condition:

\[
N_e - \text{rank} \left( A_\omega \right) = 0
\]

(53)

If-and-only-if this condition holds, the transmission mode is infeasible. An example of an infeasible mode can be found for the topology shown in Fig. 4(a), when clutch C2 and brake B1 are closed. In this situation, the rotational speeds of elements e2, e3, e4, and e5 are fixed to zero by the brake. As the rotational speed of e1 is a non-functional CDV, the equation \( \omega_1 \) is added in the pre-processing step, leading to the rank of the \( A_\omega \) matrix of this clutch combination being equal to the number of elements of the topology.

C. Post-Processing

The third and last step of the automated parameter determination is the post-processing. This step is divided in two sub-steps: mode classification (3a), and the exclusion of redundant transmission modes (3b). In the first of these two steps, all modes of the topology are classified to the mode types presented in Table II. The classification is based on four properties that are calculated in the processing step: the number of kinematic CDVs, the number of kinetic CDVs, the number of ICEs that are connected to the output, and the number of EMs that are connected to the output. Table III shows the values for the four mentioned properties for all 14 mode types. Note that for the neutral mode it must be verified whether the rotational speed of the output is a CDV in the initial \( A_\omega \) constructed in step (1b), before the non-functional CDVs are excluded in step (1c).

Step (3b) concerns the exclusion of redundant transmission modes. In step (2c) infeasible modes are excluded, yet among the leftover feasible modes there could still be redundant modes. Redundant modes arise when different clutch combinations lead to exactly the same power paths between power sources and the output. The identification of redundant transmission modes is not trivial because there can be multiple power paths between
two components at the same time (in the case of an EVT mode). Additionally, two modes that result in the same transmission ratio, yet have distinct power paths, are defined as not being redundant.

To identify redundant modes, the properties of prime numbers are used. The relevant transmission ratios are calculated for every mode with unique prime numbers as ratios of the individual gear pairs and PGSs. By the assignment of unique prime numbers, equal overall transmission ratios for distinct modes indicate equal power paths, and thus redundancy. Using the properties of prime numbers in this way is applied in computer science, e.g., in [17].

A special case of redundancy can be caused by the presence of a series hybrid mode, $\Psi = 7$. A series mode incorporates two separate mechanical circuits: one that enables propulsion, and one that enables charging of the battery. Both of these circuits could also form a feasible transmission mode on their own: a propulsion mode of type $\Psi = 2$ and a charging mode of type $\Psi = 11$, respectively. In this case, the modes of type $\Psi \in \{2, 11\}$ do not add functionality with respect to the series mode and are excluded as being redundant. Only the series mode is retained.

After the exclusion of redundant transmission modes, the final set of all feasible and non-redundant modes remains. This set, together with the types and ratios of each mode, form the parameters for the generic transmission model. This set also defines the optimization parameters of the control problem. For the topology in Fig. 5, for example, it is found that the domain of the first control variable is $x_{c,1} \in \{1, 2\}$, as the topology has two feasible transmission modes. Furthermore, there is one continuous control variable, $x_{c,2}$, due to the presence of a transmission mode of type $\Psi = 5$.

V. CASE STUDY

In order to demonstrate the potential of the automated modeling method, a case study is executed. The fuel consumption of two distinct hybrid drivetrain topologies is compared by solving the control problem for each over a drive cycle. By employing the methods presented in the previous sections, the transmission models are automatically generated and used to solve the control problem. The goal of the case study is to show that the presented method can be used to compare the electrical energy or fuel consumption of distinct drivetrain topologies in an automated way. Additionally, the computation times of the automated parameter determination and control optimization are addressed.

A. Control Optimization Method and Study Assumptions

To compare distinct design parameters, the control must be on the same level of optimality for the distinct designs. In practice this means that the control parameters must be optimal in order to find the optimal drivetrain design. In literature it is commonly shown that optimal control can only be achieved using non-causal control/simulation [18]. Dynamic programming (DP) is widely applied for the control of HEVs [19]–[23], and leads to the optimal solution up to numerical errors introduced by discretization and interpolation [6]. In [24], it is shown that causal equivalent consumption minimization strategy (ECMS) control leads to a 3% to 9% increased fuel consumption compared to non-causal DP control. In [21], it is specifically shown that the control strategy has an influence on the resulting optimal component sizes for a parallel hybrid topology.

For this case study, it is chosen to solve the control problem with DP. For the implementation of DP an existing Matlab function is used [25]. To reduce computation time, the problem is initially solved with a sparse battery SOC grid, i.e., state grid. The problem is then solved multiple times, and the state grid is iteratively refined around the solution of the previous iteration [26]. Efficiency maps are used to determine the fuel mass flow as a function of $\tau_{ice}$ and $\omega_{ice}$, and to determine the required electrical power of each EM as a function of $\tau_{em}$ and $\omega_{em}$. The control problem has two states: the on/off state of the ICE, and the battery SOC. The ICE on/off state is used to penalize starting the ICE during a driving cycle. Next to minimum and maximum values of the SOC for each time instance, a minimum is defined for the final value of this state variable to perform charge sustaining simulation.

This case study is subject to a number of assumptions that influence the results of the study. Losses of gears, clutches, and brakes are neglected, as is the required energy to actuate the clutches and brakes. The weight difference between the two drivetrain topologies and the rotational inertia of the drivetrain are not taken into consideration. Automated modeling of the mentioned losses, actuation energy, and inertia of the drivetrain is considered as future work. The model assumes that changing the transmission mode does not take time, i.e., that there is no interruption in torque flow to the wheels. Furthermore, exemplary component sizes are chosen which are equal for both topologies, yet not optimized for either of the two. Pollutant emissions are not considered. To take emissions into account, but also for a more accurate determination of the fuel consumption, the thermal states of the system (ICE and after-treatment temperatures) should be taken into consideration [27]. In case of a selective catalytic reduction (SCR) after-treatment system, AddBlue usage should additionally be considered as a cost [28].

B. Topology Comparison

For this case study, the drivetrain topology of the model year 2016 Toyota Prius is used [29]. This topology has one mode of type $\Psi = 9$. The second topology is based on the first, yet two clutches and a brake are added to create a multi-mode topology. Both drivetrain topologies are shown in Fig. 6. The two clutches and the brake enable a total of five modes, all of different mode types. The modes with their types and clutch combinations are listed in Table IV. The exemplary components are a 100 kW ICE, and two EMs of 20 kW and 100 kW, EM1 and EM2 respectively.

<table>
<thead>
<tr>
<th>Mode</th>
<th>Mode type, $\Psi$</th>
<th>C1</th>
<th>C2</th>
<th>B1</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2: EM only, FG 1 EM</td>
<td>open</td>
<td>open</td>
<td>open</td>
</tr>
<tr>
<td>2</td>
<td>3: EM only, FG 2 EMs</td>
<td>open</td>
<td>closed</td>
<td>open</td>
</tr>
<tr>
<td>3</td>
<td>5: Hybrid: FG parallel 1 EM</td>
<td>closed</td>
<td>open</td>
<td>closed</td>
</tr>
<tr>
<td>4</td>
<td>6: Hybrid: FG parallel 2 EMs</td>
<td>closed</td>
<td>open</td>
<td>open</td>
</tr>
<tr>
<td>5</td>
<td>9: Hybrid: EVT 2 EMs 2 CDVs</td>
<td>closed</td>
<td>open</td>
<td>open</td>
</tr>
</tbody>
</table>
Fig. 6. Stick diagrams. (a) 2016 Prius topology. (b) The therefrom derived multi-mode topology.

Fig. 7. Working points of the ICE. (a) 2016 Prius topology. (b) Multi-mode topology.

Fig. 8. Velocity and cumulative fuel usage over time.

The used vehicle parameters are from a full size luxury car, also referred to as F-segment. The control is optimized over the New European Driving Cycle (NEDC). Fig. 7 shows that the ICE operates in different working points for the two topologies. The velocity profile of the simulated driving cycle and the cumulative fuel usages are shown in Fig. 8. The multi-mode topology shows a reduction of 4% in fuel usage compared to the Prius topology.

Fig. 9 shows the SOC profile over time, and Fig. 10 shows the usage of the different mode types of the multi-mode topology.

With this topology comparison it is shown that the presented automated modeling method can be used to assess the fuel consumption of distinct hybrid drivetrain topologies.

C. Computation Time

Implementation of the presented method in the SLD framework as shown in Figs. 1 and 2 theoretically allows for the automated solving of the SLD problem. Yet, computation time shows to be an obstacle. Here, the computation times of the automated parameter determination and control optimization with DP are discussed.

The parameter determination method is analytic, resulting in relative short computation times. The computation times to determine the model parameters for the Prius and multi-mode topologies are less then half a second, as shown in Fig. 11(a). The strongest influence on the computation time of the parameter determination was shown by the number of clutches and brakes. This is displayed in Fig. 12, where the parameter determination times of 23 distinct topologies are shown. For all tested topologies, the division of the computation time over the pre-processing, processing, and post-processing is approximately 40%, 20%, and 40%.

Table V shows the usage of the system-level modes for both topologies.

<table>
<thead>
<tr>
<th>System-level mode</th>
<th>Prius</th>
<th>Multi-mode</th>
</tr>
</thead>
<tbody>
<tr>
<td>Regenerative braking</td>
<td>12%</td>
<td>12%</td>
</tr>
<tr>
<td>Electric only</td>
<td>35%</td>
<td>36%</td>
</tr>
<tr>
<td>Charging</td>
<td>16%</td>
<td>11%</td>
</tr>
<tr>
<td>Engine only</td>
<td>0%</td>
<td>0%</td>
</tr>
<tr>
<td>Motor assist</td>
<td>37%</td>
<td>41%</td>
</tr>
<tr>
<td>Charging in standstill</td>
<td>0%</td>
<td>0%</td>
</tr>
</tbody>
</table>

With all computations performed on an Intel Core i7-6820HQ CPU at 2.70 GHz with 16 GB RAM. For parameter determination, 10 tests have been performed per topology of which the average computation time is taken.
The computation time of the control optimization is highly dependent on the applied technique to solve it. The required times for control optimization with DP of the two considered topologies are shown in Fig. 11(b). For the control optimization of both topologies, the same discretization of the continuous control variables is applied. Therefore, the fact that the multi-mode topology has 5 times as much modes as the Prius leads to a 5 times as large total grid for the multi-mode problem. This larger grid causes a significant increase in computation time, as displayed in Fig. 11(b). Possible methods to reduce the control optimization time are discussed in the next section.

VI. DISCUSSION AND FUTURE WORK

Throughout this paper possible ways to improve and extend the presented method are mentioned. This section provides a brief overview of possible improvements and extensions that are not discussed yet. The library of components can be extended with a variator as found in push-belt continuously variable transmissions. This extends the control problem by one dimension, as the transmission ratios of every mode type \( \Psi \) can be variable for a fixed mode. For topologies containing a variator, \( x_{c,2} \) can be selected to define the variator ratio, in which case (7) changes to:

\[
\begin{align*}
\tau_{\text{ice}} &= \begin{cases} 
\frac{\tau_c (\Lambda)}{\tau_{\text{ice}} (x_{c,1}, x_{c,2})}, & \text{if } \Psi (x_{c,1}) = 1, \\
0, & \text{if } \Psi (x_{c,1}) \in \{2, 3, 4, 14\}, \\
\frac{\tau + x_{c,3}}{\tau + x_{c,2}} \cdot (7 - \tau), & \text{if } \Psi (x_{c,1}) = 5, \\
\frac{\tau + x_{c,4}}{\tau + x_{c,3}} \cdot (7 - \tau), & \text{if } \Psi (x_{c,1}) \in \{6, 9, 10\}, \\
\tau_{\text{ice}} (P_{\text{ice}} (x_{c,3})), & \text{if } \Psi (x_{c,1}) \in \{7, 11, 12, 13\}, \\
\frac{\tau_c (\Lambda)}{\tau_{\text{ice}} (x_{c,1}, x_{c,2})}, & \text{if } \Psi (x_{c,1}) = 8, \\
\end{cases}
\end{align*}
\]

(54)

with

\[
\tau = \min \left( \tau_{\text{ice}} (\omega_{\text{ice}} (\Psi (x_{c,1}), \omega (\Lambda), i_{\text{ice}} (x_{c,1}, x_{c,2}, x_{c,3}))) \right)
\]

(55)

\[
\tau = \max \left( \tau_{\text{ice}} (\omega_{\text{ice}} (\Psi (x_{c,1}), \omega (\Lambda), i_{\text{ice}} (x_{c,1}, x_{c,2}, x_{c,3}))) \right)
\]

(56)

The presented method can also be used for automated generation and evaluation of transmission design boundaries. Relevant examples are maximum rotational speeds of the planet gears of a PGS, and maximum clutch and brake torques. This is particularly interesting when the presented method is used in combination with automated component sizing.

The computation time of the control optimization has a large influence on the computation time of the complete SLD problem, which shows to be a limiting factor for the solving of that problem. In [5] methods are presented to further reduce the computation time of DP. An alternative is to formulate the control problem convex, and solve it analytically. For convex formulation, the problem may not have discrete variables, which would make the problem nonlinear. In [30], the control problem for a series HEV is formulated convex by replacing the ICE on/off control with an embedded problem (EP), and solving the problem using Pontryagin minimum principle (PMP). For the same type of HEV, the engine on/off control is solved with rule based (RB) control and the other control parameters are solved convex simultaneously with the component sizing in [23]. PMP is applied for the control of a single mode EVT topology in [31]. In these examples, the ICE on/off signal is the only discrete control variable. For topologies with multiple (discrete) transmission modes it is possible to formulate and solve the problem of the continuous control variables convex for each value of the discrete variable(s) [32], [33]. In this case, the convex solving of the continuous control variables is nested in a DP optimization of the discrete control variables, and the control problem must be formulated convex for every mode type \( \Psi \).

VII. CONCLUSION

A method for automated modeling of arbitrary hybrid and electric drivetrain topologies containing up to one ICE and two EMs is developed. This method contributes to the solving of the SLD problem of HEVs. A generic transmission model is defined, for which the topology-dependent parameters are automatically determined. The parameter determination is performed analytically, resulting in low computational demands. By modeling topologies at the level of (planetary) gears and clutches, the method is flexible to unconventional topologies. To allow for the integration with topology generation, topologies are modeled as undirected connected fine graphs. To show the potential of the method, a case study is performed: the control problem is solved with DP using automatically generated transmission models for two distinct hybrid drivetrain topologies, to enable comparison of their fuel consumption. Future work will include the automated modeling of transmission losses, and research on methods to reduce the computational effort of solving the control problem for arbitrary drivetrain topologies.
REFERENCES


Wilco van Harselaar was born in Tiel, The Netherlands, in 1992. He received the B.Eng. degree in automotive engineering from the Eindhoven University of Applied Sciences, Arnhem, The Netherlands, in 2015, and the M.Sc. degree (cum laude) in automotive technology from the Eindhoven University of Technology, Eindhoven, The Netherlands, in 2017. Since 2017, he has been working toward the Ph.D. degree at Daimler AG, Stuttgart, Germany, in cooperation with the Institute for Mechatronic Systems, Technische Universität Darmstadt, Darmstadt, Germany. His research interests include the design, modeling, and control of automotive propulsion systems.

Theo Hofman was born in Utrecht, The Netherlands, in 1976. He received the M.Sc. (with honors) and the Ph.D. degree from the Eindhoven University of Technology, Eindhoven, The Netherlands, in 1999 and 2007, respectively, both in mechanical engineering. From 1999 to 2003, he was a Researcher and a Project Manager with the R&D Department, ThalesCryogenics B.V., Eindhoven, The Netherlands. From 2003 to 2007, he was a Scientific Researcher with Drivetrain Innovations B.V., Eindhoven. From 2007 to 2009, he was a Postdoctoral Fellow with the Control Systems Technology group. Since 2010, he has been an Assistant Professor with the Control Systems Technology group. His research interests are modeling, design, and control of energy-efficient propulsion systems.

Markus Brouwer was born in Tokyo, Japan. He received the Dipl.-Ing. degree in mechanical engineering from RWTH Aachen University, Aachen, Germany, in 2007. Since 2008, he has been working as a Development Engineer with Daimler AG in Stuttgart, Germany, where his main focus is the design and analysis of advanced vehicular powertrains.