NordDesign 2018
August 14 – 17, 2018
Linköping, Sweden

Automated Generation of Vehicle Architectures and Derivation of Modular Systems within the Vehicle Front

Matthias Felgenhauer, Markus Lienkamp

Technical University of Munich
felgenhauer@ftm.mw.tum.de

Abstract

To reduce complexity while offering high variance to customers, vehicle manufacturers build multiple vehicles based on modular systems. These require unified architectural standards and cross-vehicle modules.

However, the identification of architectural standards and modules for multiple vehicle models and performance classes is difficult, due to a high uncertainty and a vast solution space. For a holistic identification of architectural standards and modules, concept engineers need to generate and dimension as well as compare hundreds of thousands of vehicle architectures. The unmanageable quantity of solutions as well as lack of time and resources make it impossible to investigate all conceivable vehicle architectures.

This indicates the need for a new method and software tool. Using MATLAB and CATIA, the "Parametric Automotive Concept Engineering" (PACE) method and tool holistically identifies architectural standards and modules within the vehicle front. Hereby, the tool considers multiple component variants, including combustion, hybrid and electric drivetrains. In addition, it includes different position variants, such as longitudinal and transverse engine installation.

The concept development department of AUDI AG uses the tool at an early stage to identify architectural standards and modules. It further allows concept engineers to assess the feasibility of modular systems and the solution of conflicts of interests through the fast variation of input requirements. Consequently, the method and software tool decrease both the time and the resources for the identification of architectural standards and modules.

1 Introduction

1.1 Fundamentals of Modular Systems

In line with the megatrends of globalization, mass customization and environmental protection, customers demand vehicles with different shapes, performance characteristics and drivetrains. To fulfill customer requirements and increase revenues, vehicle manufacturers continuously increased the number of vehicle models and the number of component variants, over the last decades (Felgenhauer et al. 2017b). The high number of models and variants increases complexity in development and production.

To overcome this downside of high variance, vehicle manufacturers avoid the individual design of vehicles. Instead a modular system defines the foundation for multiple vehicles of various segments. The concept of a modular system is based on the unified definition of architectural standards, such as the engine installation position and the strut mounting, as well as the cross-vehicle use of modules (Schuh et al. 2012; Fuchs 2014). This reduces the internal variance within all vehicles to a minimum, while allowing a high external variance. Examples for the application of the concept are the “Modularer Längsbaukasten (MLB)” of AUDI AG and Volkswagen’s “Modularer Querbaukasten (MQB)”.

Since the introduction, two approaches for the derivation of modular systems have evolved. The single-drivetrain approach designs modular systems separately for combustion and electric vehicles. This concept allows for adaption to the drivetrain characteristics. On the other hand, the multi-drivetrain approach derives one modular system for all drivetrain types. Here, the benefit is the flexible response in production to the customer demand for particular drivetrains.

1.2 Process for the Definition of Modular Systems

The development of a modular system starts about five years before the start of production of the first vehicle. After the definition of the number of vehicles as well as concept and customer requirements, the vehicle concept development department is responsible for the definition of the modular system.

Hence, the target is the identification of architectural standards and of cross-vehicle modules for multiple vehicles. To define the architectural standards and modules, concept engineers generate and dimension as well as compare architectures of all vehicles.

The variation of component and position alternatives creates multiple vehicle architectures. The dimensioning of component sizes and distances between them, for all considered component and position alternatives, is based on concept and customer requirements, such as acceleration and vehicle type. Through the comparison of the required installation space of each architecture with the available installation space of each vehicle, concept engineers identify the feasible architectures. By comparison of all vehicles’ feasible architectures, concept engineers derive architectural standards and cross-vehicle modules. In the event that a modular system is unfeasible, the process restarts with the refinement of the component sizes and distances. Further options involve the adaptation of the requirements.

1.3 Problem Description

The main challenges in defining a modular system are the complicated investigation of the vast solution space and iterations due to conflicts of interests.

For one architecture, concept engineers need to define hundreds of dimensions for the component sizes and the distances between components. Those dimensions depend on a multitude of concept and customer requirements. As these vary further for the component and position alternatives, the coherences between the requirements and the dimensions are often unknown at this early stage of the development (Felgenhauer et al. 2018). Furthermore, interdependencies between the requirements and/or the dimensions cause iterations within the dimensioning.

To overcome the high uncertainty and interdependencies within the dimensioning of components and distances, concept engineers assume the dimensions and investigate predecessor and competitor vehicles, as a starting point. If existing components do not qualify, investigations, simulations and iterations are both time and resource consuming.

Despite the high effort for the dimensioning of one vehicle architecture, the holistic generation of a modular system requires the generation and dimensioning of hundreds of thousands of vehicle architectures, depending on the number of vehicles and the considered component and position alternatives. In addition, concept engineers need to identify unified architectural standards and modules from this vast number of solutions. This makes a holistic consideration not only time and resource consuming, but humanly unmanageable.

As a result, concept engineers predefine architectural standards, such as the drive type and the engine installation position as well as modules as concept boundaries based on predecessor vehicles and benchmarks. These boundaries decrease the number of possible architectures and therefore the complexity. However, the reduction in the solution space, prevents the identification of architectural standards and modules best suited to the vehicle dimensions and requirements.

Even with the reduced complexity, the process might still require several iterations due to conflicts of interests. On the one hand, the high external variance can conflict with a minimum internal variance (Fuchs 2014). To fulfill the customers need for individualization, the vehicles of the modular system differ in their functional requirements, such as the vehicle acceleration, and the geometric requirements, in terms of exterior dimensions. This enables a high external variance. However, due to the dispersion of requirements it might be impossible to identify architectural standards and modules. On the other hand, it is possible that a conflict of interest exists between the functional requirements and therefore the component sizes as well as distances and the geometric requirements as the available installation space. It is possible, that the refinement of the component sizes and distances will immediately resolve these conflicts. Otherwise it requires a change of the variance, the functional requirements or the geometric requirements. Either way the conflicts of interests require process iterations that are both time and resource consuming.

2 State of the Art

Below, we describe the existing methods for the definition of vehicle architectures and the identification of modules and architectural standards.

2.1 Existing Approaches to Vehicle Architecture Design

Several approaches for the definition of vehicle architectures for passenger cars exist in the literature (Kuchenbuch 2012; Ried 2014; Fuchs 2014; Matz 2015). The scope ranges from the requirement definition and conversion, through the design of vehicle dimensions up to the generation and optimization of architectures. The methods either analyze the solution space within one architecture or generate and compare multiple architectures. All focus on hybrid or electric drivetrains. The component models for the generation and optimization of architectures, are mainly based on databases. For a few components, some authors use empirical and physical models to scale the component sizes. Felgenhauer et al. 2018 presented an overview of these methods.

Besides the consideration of either hybrid or electric drivetrains and the limited use of scalable component models, all presented methods consider just one architecture for one vehicle.

Förg et al. (2014) focus on portfolios for multiple commercial vehicles. They generate vehicle architectures based on customer profiles and sets of components. Starting with one vehicle and customer profile, the concept engineer manually defines the component positions as architectural standards. For subsequent customer profiles and vehicles, the tool proposes architectural standards based on the predefined architecture. In the event that it is impossible to use existing architectural standards, the user can define a new architecture. Then, the tool rates the overall portfolio in respect of the degree of standardization and identifies areas for improvement. However, the tool only supports the manual definition of architectures and the identification of architectural standards. The tool does not support the identification of the best solution. The resulting degree of standardization depends strongly on the first user-defined architecture.

Based on the ideas of Förg et al., Stocker et al. 2016 developed a tool for the automated derivation of portfolios for frame-mounted parts of commercial vehicles. After the generation of all feasible architectures, the tool defines architectural standards and modules for multiple vehicles and customer profiles, taking into consideration production volumes and costs. However, the components of the tool are based on a database. This makes a holistic investigation of the solution space impossible, as the tool is not adaptable to new requirements. In addition, they do not consider the vehicle front and the drivetrain, as the focus lies solely on frame-mounted parts.

Both, Förg et al. and Stocker et al. refer to portfolios instead of modular systems. Due to the high variance of commercial vehicles it is impossible to derive modular systems with fully unified architectural standards and modules.

2.2 Research Gap

Current methods lack either the consideration of different drivetrains, scalable models or architectural standards and modules. Therefore, no method for the automated generation of multiple vehicle architectures and the derivation of modular systems, with multiple drivetrains and scalable component models, exists. However, such a method would decrease the complexity and time and resources needed to define modular systems.

3 Method for the Automated Generation of Vehicle Architectures and Derivation of Modular Systems within the Vehicle Front

The authors have developed the "Parametric Automotive Concept Engineering" (PACE) method and software tool. PACE enables the holistic generation and the requirements-based, adjustable dimensioning of all conceivable architectures for the vehicle front. Based on these architectures, it automatically identifies unified modules and architectural standards for multiple vehicles.

Hence, the tool makes it possible to quickly derive modular systems for specific vehicle dimensions and requirements as well as single and multi-drivetrain approaches. In case a modular system is unfeasible, the tool enables the resolution of conflicts of interests by changing the input parameters and a fast repetition of the process. This reduces the time taken by iterations and therefore increases the efficiency during the derivation of modular systems at an early stage. After the description of the requirements and boundaries, we explain the method in detail.

3.1 Requirements and Boundaries

The main requirements divide into the following three categories:

- **Applicable:** the tool must be applicable even under high uncertainty during the early stage of the concept development. This limits the input parameters and defines the required level of detail.

- **Holistic:** besides combustion, hybrid and electric drivetrains the tool should be able to adjust to different requirements and consider the entire solution space of existing architectures. In addition, integrated models should be extensible.

- **Replicable:** the focus of the tool is not on the identification of completely new solutions. Instead, the target is to handle and investigate the solution space of current architectures. To do so, the process and the solution must be replicable and transparent.

The boundaries are defined by the focus of the method and tool on the vehicle front. Other areas of the vehicle are only partly (interior, cell) or not considered (rear). In addition, the tool is limited to the most important components of the vehicle front.

3.2 Overview of the Method

Using MATLAB and CATIA, the PACE method and tool include the definition of requirements (3.3), the generation of architectures (3.4) and the derivation of modular systems (3.5).
Within the definition of requirements, the user defines the most important concept and customer requirements as input parameters for multiple vehicles with several performance classes. A longitudinal dynamics model transforms these inputs into engine-gearbox combinations as performance requirements.
The generation of architectures is based on empirical and semi-physical geometric substitute models for the requirement-based scaling of component sizes and distances between components. The tool synthesizes and dimensions vehicle architectures by applying geometric substitute models to each combination of the engine-gearbox combinations and the permutation of component and position alternatives. In addition, the tool derives the available installation space, mainly based on the vehicle's exterior dimensions. Afterwards, it compares the required installation space of each architecture with the available installation space of each vehicle to identify feasible architectures. The tool iterates these steps in a fully automated manner for all vehicles and performance classes.
Subsequently, the derivation of modular systems identifies modules and architectural standards from all vehicles' feasible architectures. A parametric CATIA model is available to visualize the results. Figure 1 gives an overview of the presented method and tool.

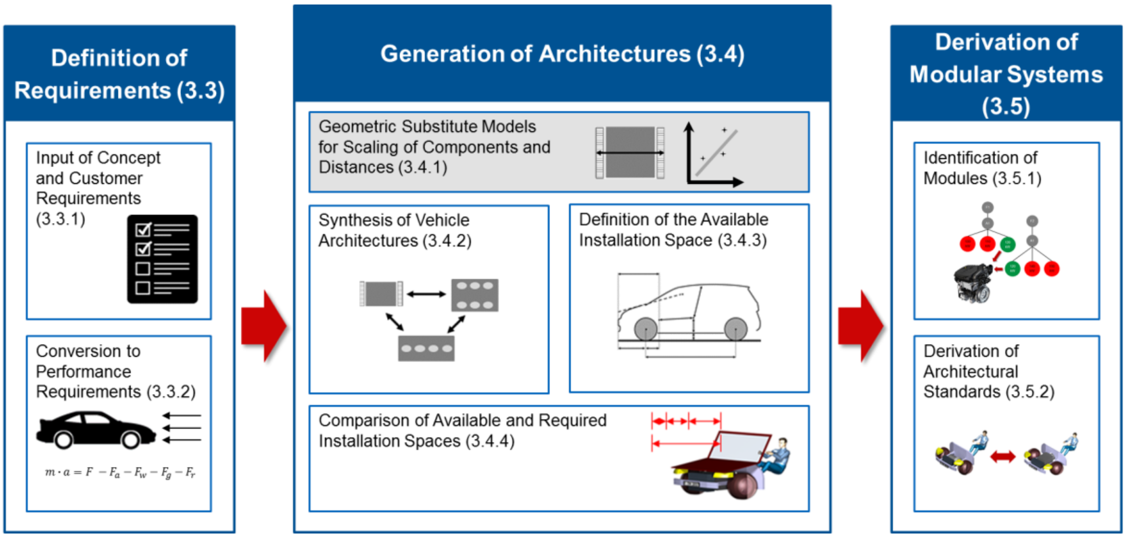


Figure 1. Overview of PACE

3.3 Definition of Requirements

At the start of tool, a concept engineer defines the concept and customer requirements for the modular system. Based on these inputs, a longitudinal dynamics simulation determines the performance requirements.

3.3.1 Input of Concept and Customer Requirements

Using a MATLAB user interface, the user defines general requirements for the modular system, such as the number of vehicles, and selects between the single- and multi-drivetrain approach. For up to five vehicles, the exterior dimensions (overhang, wheelbase, width) define the geometric concept requirements. The selection of the vehicle type (low-ground or high-ground) and the drivetrains (combustion, parallel hybrid and electric) are functional concept requirements. Each of the vehicles can have up to three performance classes (entry, volume and sports). The drive type (two-wheel or four-wheel drive), the weight, the load capacity and the trailer load are functional concept requirements for each performance class. The acceleration and the maximal velocity represent the functional customer requirements for each performance class. In this respect, the tool allows the inputting of concept and customer requirements of the performance classes separately for combustion/hybrid and electric drivetrains. Here, all parameters are adapted to the information available at an early stage of the development. For more detailed specification, the tool permits selection of the battery and headlight technology. Further optimization parameters are active pedestrian protection, extra load tires and an additional lower stiffener.

3.3.2 Conversion to Performance Requirements

The tool inputs do not include the engine and gearbox characteristics. Therefore, a longitudinal dynamics model converts concept and customer requirements into the engine power and torque as well as the number of gear speeds and their transmission ratios (Figure 2).

Figure 2. Conversion of concept into performance requirements

The longitudinal dynamics simulation involves combustion, parallel hybrid and electric drivetrains. For combustion engines, naturally aspired, supercharged gasoline and supercharged diesel engines apply. Parallel hybrids are a combination of combustion engines with permanent synchronous machines. For the electric machines, the tool considers permanent synchronous and asynchronous machines with low, medium and high rotational speeds.

To solve the undetermined problem, as neither engine nor gearbox are known, the tool scales load curves for all combustion engines, hybrid machines and electric machines from 40 to 400 kW in steps of 5 kW.

Each of the engine characteristics combines with one or six to nine speed gearboxes of the electric or combustion/hybrid drivetrains. The tool calculates the transmissions ratios, based on the climbing ability, the starting acceleration, the creep- and the maximum-velocity, for each of the engine and gearbox combinations. During the scaling of the load curves and the transmission ratio calculation, it is immediately possible to neglect some engines and gearboxes, because of insufficient eligibility.

Of the remaining engine-gearbox combinations, the tool identifies the ones complying with the requirements by means of an acceleration simulation.

Due to the high number of drivetrains, engine characteristics and gearboxes, the simulation runs over 850,000 combinations for all vehicle and performance classes. This includes an iteration for front and rear wheel drive types. As a combination of both, all-wheel drive vehicles do not require this iteration. As result, the tool must consider around 8000 feasible engine-gearbox combinations as performance requirements in the following steps.

The tire diameter for each vehicle and performance class is based on the vehicle weight and load capacity using an empirical model (3.4.1). Further parameters, such as the load distribution for the different drive types also originate from empirical evaluations. The vehicle dimension concept (3.4.3) defines the height of the vehicle.

3.4 Generation of Architectures

Based on the customer, concept and performance requirements, the tool generates and evaluates all possible vehicle architectures.

3.4.1 Geometric Substitute Models for Scaling of Components and Distances

For the requirement-based, adjustable scaling of component sizes and the distances between them, the authors created empirical and semi-physical geometric substitute models for over 25 components and distances of the drivetrain, the cooling, the chassis, the body and the passenger ergonomics. These models serve as a basis for the scaling of component sizes and distances between components throughout the tool. With the requirements forming the input, the model outputs the geometry.

The empirical substitute models can either be regression functions or constant values. In the latter case, the empirical data does not show any deviation for different requirements. Depending on the modelling granularity by subdivision of components, regression functions and constants apply for different subcomponents. Figure 3 shows the empirical substitute model for the length of in-line combustion engines, as a function of the maximum engine torque. This model consists of multiple sub-models for the engine swept volume, the cylinder volume, the bore diameter, the cylinder spacing and the overhang. For the empirical models, up to 300 data sets originate mainly from the automotive benchmarking database A2MAC1 EURL as well as CAD-sectional-drawings from the “Global Car Manufacturers Information Exchange” (GCIE).

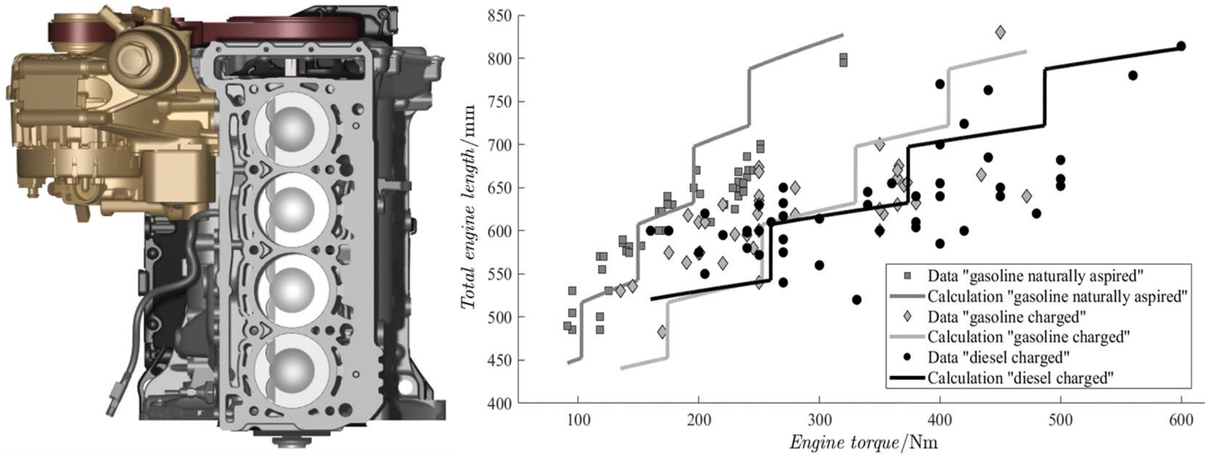


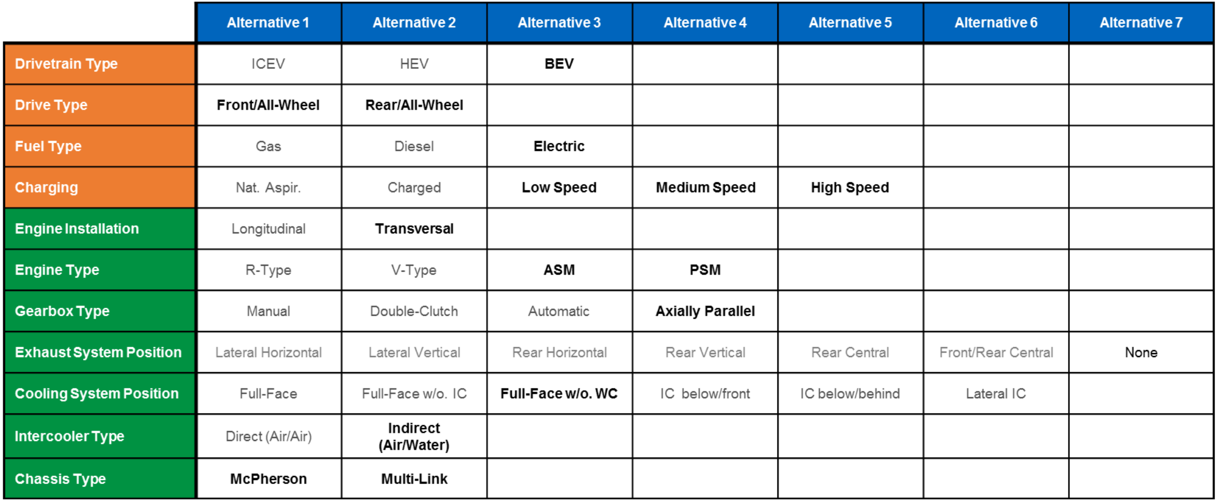
Figure 3. Empirical model for the length of in-line combustion engines with jumps in the number of cylinders

Besides empirical models, a semi-physical model, combining physical equations and empirical regression functions or constants, exists for the one-speed gearbox and the distance between the bumper beam and the cooling system.

The accuracies of the models range from 80 to 97 %. Therefore, the model accuracies are sufficient for the early stage of vehicle development. The authors described the process and the models in detail in Felgenhauer et al. (2018).

3.4.2 *Synthesis of Vehicle Architectures*

The generation of vehicle architectures is divided into two parts. One is the stepwise extension of the engine-gearbox combinations with conceivable component and position alternatives. The other is the dimensioning of component sizes and distances using geometric substitute models. Alternatives for the combustion engine types (in-line and V-type) and the installation positions (longitudinal or transverse) expand with the engine-gearbox combinations to preliminary architectures. Consequently, these preliminary architectures are a multiplication of the engine-gearbox combination for each of the engine types and the installation positions. Electric machines do not have a multiplication effect, as the engine-gearbox combinations already include the engine types (permanent synchronous and asynchronous machines) and as only transverse installation exists. Further stepwise extensions of the preliminary architectures with the exhaust system positions, gearbox types, cooling system positions as well as chassis types leads to over 250,000 vehicle architectures. Figure 4 displays the component and position alternatives already considered in the engine-gearbox combinations (orange) and ones added during the synthesis (green). However, individual component and position alternatives only partly combine because not all alternatives match with each other. For example, an electric machine does not need an exhaust system (grey alternatives in Figure 4 do not apply for BEV). In-between each expansion step the scaling of component sizes and distances takes place for the related components and distances. Hereby, the tool automatically rules out architectures which exceed technical boundaries, such as the number of cylinders or gear speeds.



	Alternative 1	Alternative 2	Alternative 3	Alternative 4	Alternative 5	Alternative 6	Alternative 7
Drivetrain Type	ICEV	HEV	**BEV**				
Drive Type	**Front/All-Wheel**	**Rear/All-Wheel**					
Fuel Type	Gas	Diesel	**Electric**				
Charging	Nat. Aspir.	Charged	**Low Speed**	**Medium Speed**	**High Speed**		
Engine Installation	Longitudinal	**Transversal**					
Engine Type	R-Type	V-Type	**ASM**	**PSM**			
Gearbox Type	Manual	Double-Clutch	Automatic	**Axially Parallel**			
Exhaust System Position	Lateral Horizontal	Lateral Vertical	Rear Horizontal	Rear Vertical	Rear Central	Front/Rear Central	None
Cooling System Position	Full-Face	Full-Face w/o. IC	**Full-Face w/o. WC**	IC below/front	IC below/behind	Lateral IC	
Intercooler Type	Direct (Air/Air)	**Indirect (Air/Water)**					
Chassis Type	**McPherson**	**Multi-Link**					

Figure 4. Overview of the component and position alternatives within the engine-gearbox combinations (orange) and the synthesis (green). Grey component and position alternatives do not apply for BEV.

3.4.3 *Derivation of the Available Installation Space*

The previous step creates a vast number of vehicle architectures. To assess the feasibility of these in the upcoming step, the tool derives the available installation space. In the vehicle front, the length, width and height of the engine compartment represent most of the available installation space.

From the outside to the inside, the derivation of the available installation space is based on dimensional chains of the Vehicle Dimension Design in x-, y- and z-directions. In x-direction, the length of the vehicle front minus the distance for pedestrian protection and sizes of body and pedal parts, is equal to the length of the engine compartment. The vehicle width, the maximal tire width and turning angle as well as the strut mounting are primarily responsible for the available space in y-direction. The height of the engine compartment is based on dimensions such as the ground clearance, the H30 dimension, the body height and the viewing angle (Figure

5). As input for the longitudinal dynamics model, the tool calculates the vehicle height based on the vehicle type and the seat reference point.

The derivation of the available installation space proceeds separately for each vehicle. However, with different weights and the existence of an underbody battery, the available space in one vehicle varies for combustion/hybrid and electric drivetrains. In contrast to a single drivetrain approach, a multi-drivetrain approach considers the minimum installation space of both combustion/hybrid and electric drivetrains in the vehicle.

The component sizes and distances in the derivation of the available installation space are also based on substitute models (3.4.1). In most cases they are empirical constants rather than regression functions. Semi-physical models do not apply.

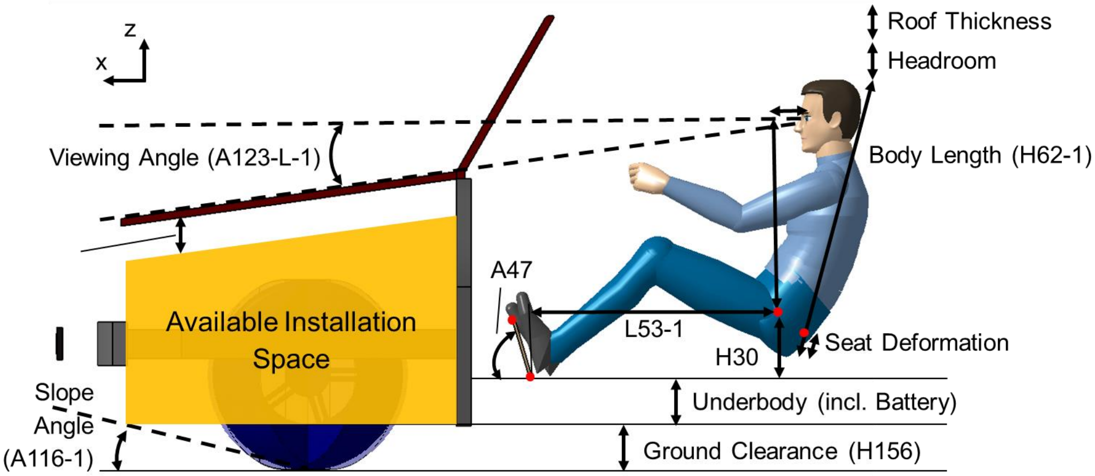


Figure 5. Dimensional chains of the Vehicle Dimension Design in z-direction

3.4.4 *Comparison of Available and Required Installation Space*

This step identifies the feasible architectures from the vast amount of generated vehicle architectures. Using dimensional chains of the Vehicle Architecture Design as described in Felgenhauer et al. 2017a, the tool compares the required installation space of each architecture with the available installation space from the inside to the outside (Figure 6).

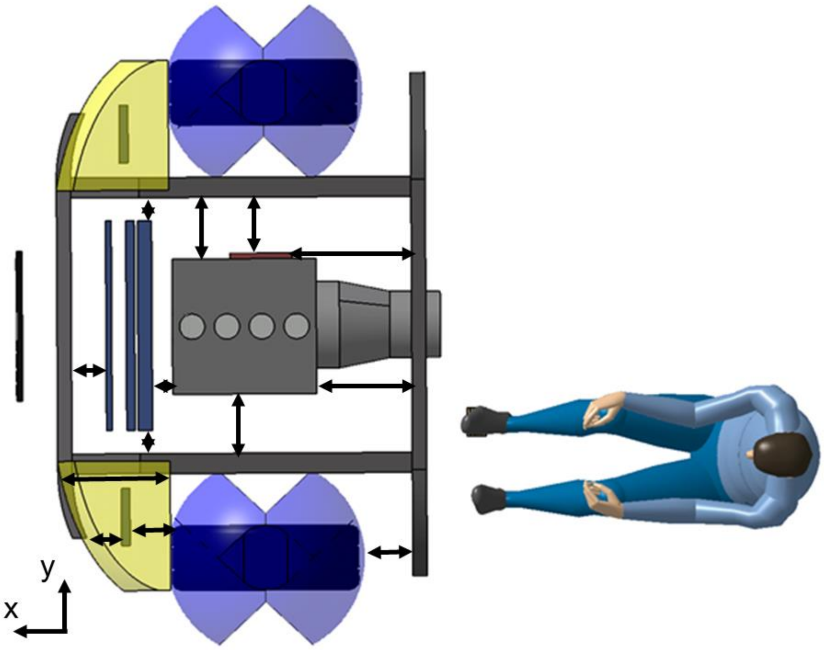


Figure 6. Dimensional chains of the Vehicle Architecture Design in x- and y-direction

Based on the available installation space within the engine compartment, these dimensional chains mainly assess the distances in x-, y- and z-directions. Further dimensional chains apply for the distance between the bumper beam or the headlight and the tire as well as the strut mounting and the engine. The distances mainly represent crash, assembly, vibration, thermal and other requirements. If just one distance or the sum of distances for the crash length is below its minimum specification, the architecture is not feasible. Other than the distance between the bumper beam and the cooling system as a semi-physical model, empirical substitute models (3.4.1) define the distance requirements.

3.5 Derivation of Modular Systems

At this point, the tool calculated the feasible architectures of all vehicles. Before the derivation of architectural standards for all feasible architectures, the tool identifies modules.

3.5.1 Identification of Modules

At this stage, multiple feasible vehicle architectures display the same external variance. Consequently, these only differ in their internal variants, such as the component type or the engine-gearbox combinations.

To select a vehicle architecture and thus decide on the internal variance and on one engine-gearbox combination the aim is to identify modules in all vehicles and performance classes.

Therefore, the tool subdivides the feasible architectures by the combinations of drive type, engine installation and chassis, which display the architectural standards. This is necessary to evaluate the modularization for each combination independently.

For each combination of architectural standards, the tool searches for performance characteristics of one component type, which are also feasible in different performance classes and vehicles. As for example the engine power of diesel in-line engines. An iteration runs for all power characteristics and component types that only display internal variants - in this particular case, diesel v-type engines. The tool compares the rates of modularization of the different internal variants and performance characteristics. The engine type and characteristics with the highest modularization rate qualifies for modularization. In vehicles and performance classes where the module applies vehicle architectures with other internal variants and performance characteristics get unnecessary (Figure 7). This process restarts for the remaining vehicles and performance classes, until no further modules are possible. The same procedure applies for other combinations of the architectural standards and other external variants of the engine. After the engine, the tool modularizes gearbox, hybrid machine and cooler. The sequence is based on the development and production costs, which are much higher for an engine than for a cooler.

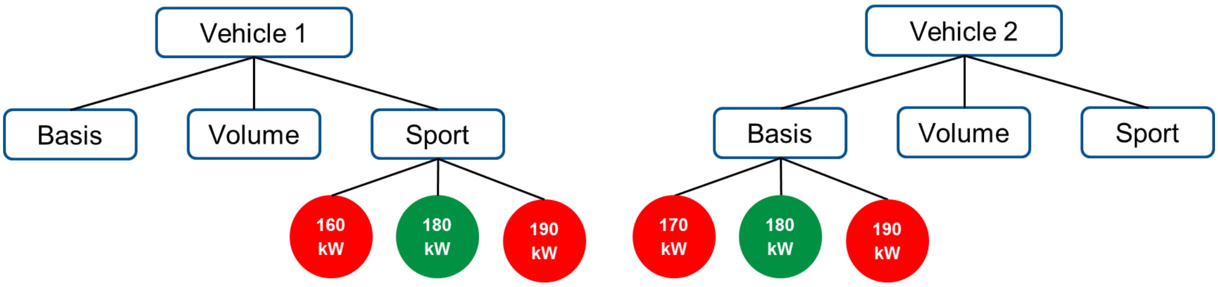


Figure 7. Modularization of 180 kW diesel engines

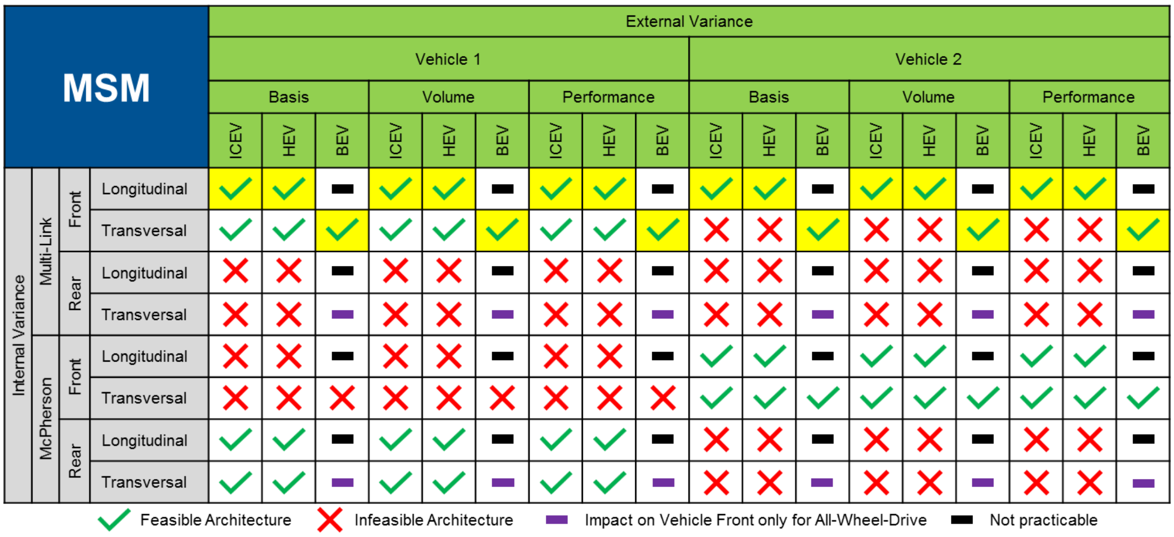
In some cases, a modularization is not possible. Hence, the tool selects the internal variant and engine-gearbox combination with the smallest component sizes and largest distances. A priority applies for the x-direction because here empty spaces increase the crash performance.

This step limits the number of architectures to a minimum. However, the tool always ensures the desired external variance.

3.5.2 Derivation of Architectural Standards

In this step, the tool indicates in the modular systems matrix (MSM) if vehicle architectures exist for the different combinations of external variants and architectural standards (Figure 8). Based on the MSM concept, engineers can determine if one combination of architectural standards (Figure 8, rows) includes feasible architectures and fulfils all external variants (Figure 8, columns). This indicates the general feasibility of a modular system. The architectural standards and the predefined modules would then form the modular system. The tool makes it possible to visualize all vehicle architectures within a parametric model of the vehicle front in CATIA. Should no modular system exist, concept engineers can use the matrix to identify the vehicles which require an adjustment of the functional or geometric requirements. The ability to restart the tool avoids time-consuming, manual iterations.

In the example in Figure 8, the first row indicates that multi-link suspension, front wheel drive and longitudinal engine installation as architectural standards enable vehicle architectures for all vehicles, performance classes and combustion and hybrid drivetrains. As longitudinal engine installation is not practicable for electric machines, the modular system could consider these with multi-link suspension, front wheel drive and transverse engine installation (Figure 8, row two). The combination of longitudinal combustion engines and transverse electric machines is feasible because the available installation space (3.4.3) in the multi-drivetrain approach is equal for one vehicle. The combination of longitudinal and transverse combustion engines would also be possible due to the equal installation space. However, this is not meaningful within a modular system as this limits the modularization.



MSM				External Variance																	
				Vehicle 1									Vehicle 2								
				Basis			Volume			Performance			Basis			Volume			Performance		
				ICEV	HEV	BEV	ICEV	HEV	BEV	ICEV	HEV	BEV	ICEV	HEV	BEV	ICEV	HEV	BEV	ICEV	HEV	BEV
Internal Variance	Multi-Link	Front	Longitudinal	✓	✓	▬	✓	✓	▬	✓	✓	▬	✓	✓	▬	✓	✓	▬	✓	✓	▬
			Transversal	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✓	✗	✗	✓	✗	✗	✓
		Rear	Longitudinal	✗	✗	▬	✗	✗	▬	✗	✗	▬	✗	✗	▬	✗	✗	▬	✗	✗	▬
			Transversal	✗	✗	🟪	✗	✗	🟪	✗	✗	🟪	✗	✗	🟪	✗	✗	🟪	✗	✗	🟪
	McPherson	Front	Longitudinal	✗	✗	▬	✗	✗	▬	✗	✗	▬	✓	✓	▬	✓	✓	▬	✓	✓	▬
			Transversal	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✓	✓	✓	✓	✓	✓	✓	✓
		Rear	Longitudinal	✓	✓	▬	✓	✓	▬	✓	✓	▬	✗	✗	▬	✗	✗	▬	✗	✗	▬
			Transversal	✓	✓	🟪	✓	✓	🟪	✓	✓	🟪	✗	✗	🟪	✗	✗	🟪	✗	✗	🟪

✓ Feasible Architecture ✗ Infeasible Architecture 🟪 Impact on Vehicle Front only for All-Wheel-Drive ▬ Not practicable

Figure 8. Feasible and infeasible vehicle architectures displayed in the modular systems matrix (MSM)

4 Conclusion and Outlook

The authors developed the PACE method and tool for the requirement-based, holistic and automatic identification of modules and architectural standards for a modular system. Due to the empirical and semi-physical substitute models, the tool is adjustable to different input requirements. In addition, combustion, hybrid and electric drivetrains can be considered.

This tool applies at an early stage of vehicle concept development. To overcome the high uncertainty, the tool is based on just a few input parameters. Concept engineers can use this tool to identify the architectural standards and modules that offer the required external variance for a minimum of internal variance, from over 250,000 architectures. In addition, the tool allows for the fast adaptation of the functional and geometric requirements to analyze the effects of changes and to solve conflicts of interests. Consequently, time-consuming manual iterations are avoided at an early stage using the tool.

However, the method and tool focus on the geometric feasibility of modular systems. An economic assessment would require the consideration of production volumes as well as costs of component alternatives. In addition, the tool is not able to generate never seen architectures. Instead it is meant to cope with the vast variance of existing solutions, using a systematic and replicable approach with dimensional chains. The empirical models output a statistical average. This describes a robust design and leaves leeway for further development. Consequently, the models do not consider the best-of-benchmark dimensions. Furthermore, due to the early stage consideration, the authors simplify styling influences such as the curvature of the bumper beam and the shape of the headlight. Lastly, the tool focuses on the vehicle front. This is reasonable, as one of the focus areas is on single- and multi-drivetrain approaches.

In the future the user will be able to deviate from the statistical average for selected components, by defining an optimization rate. In addition, it is planned to optimize the tool by replacing the vehicle weight as an input parameter with an empirical or semi-physical weight assumption model based on vehicle dimensions. Finally, the authors will evaluate the tool using CAD-data of existing vehicles and modular systems.

Contributions

As lead author, Matthias Felgenhauer identified the problem within the definition of modular systems and developed the presented method and tool. Markus Lienkamp contributed to the conceptual design, proofread the paper and fully supports the method.

Acknowledgement

This article includes results from the student theses of Amin Siala, Andreas Krohn, Florian Schneider, Dzemaludin Malkic, Raul Marksteiner, Frederik Massner, Christoph Mederer, Stephan Wagner, Stephane Levrat, Tim Schröder, Nico Trümper and Maximilian Zähringer. Furthermore, it includes conceptual designs from Frank Züge (Head of Department) and Frank Schöpe (Team Leader) of the Department for Front and Rear Vehicle Concepts MLB at AUDI AG. In addition, the authors would like to thank A2Mac1 EURL, in the person of Wilhelm König, for support through the access to the A2Mac1 automotive benchmarking database. AUDI AG and Technical University of Munich sponsor the project.

References

A2MAC1 EURL: A2mac1. Automotive Benchmarking. Version. Hary, France. Online verfügbar unter https://www.a2mac1.com/, zuletzt geprüft am 27.10.2017.

Felgenhauer, Matthias; Angerer, Christian; Marksteiner, Raul; Schneider, Florian; Lienkamp, Markus (2018): Geometric substitute models for efficient scaling of dimensions during the vehicle architecture design. In: Design Society (Hg.): Proceedings of the DESIGN 2018. 15th International Design Conference. Dubrovnik, Croatia, 21.05-24.05.2018.

Felgenhauer, Matthias; Schöpe, Frank; Bayerlein, Michaela; Lienkamp, Markus (2017a): Derivation, analysis and comparison of geometric requirements for various vehicle drivetrains using dimensional chains. In: Design Society (Hg.): Proceedings of the 21st International Conference on Engineering Design (ICED17). Vol. 4: Design Methods and Tools. Vancouver, Canada, 21.-25.08.2017, S. 189–198.

Felgenhauer, Matthias; Stocker, Johannes; Lienkamp, Markus (2017b): New approach for architecture design of high variable vehicle portfolios. In: ASME (Hg.): Proceedings of the ASME 2017 International Mechanical Engineering Congress and Exposition. Tampa, USA, 03.-09.11.2017.

Förg, Armin; Kreimeyer, Matthias; Lienkamp, Markus (2014): Architecting new vehicle concepts: virtual design and performance based assessment. 13th International Symposium on Heavy Vehicle Transportation Technology. International Forum for Road Transport Technology. San Louis, Argentinia, 2014.

Fuchs, Johannes (2014): Analyse der Wechselwirkungen und Entwicklungspotentiale in der Auslegung elektrifizierter Fahrzeugkonzepte. Dissertation. Technische Universität München, München. Institut für Maschinen- und Fahrzeugtechnik, Lehrstuhl für Fahrzeugtechnik.

Kuchenbuch, Kai (2012): Methodik zur Identifikation und zum Entwurf packageoptimierter Elektrofahrzeuge. Dissertation. Technischen Universität Carolo-Wilhelmina zu Braunschweig, Braunschweig. Institut für Konstruktionstechnik.

Matz, Stephan (2015): Nutzerorientierte Fahrzeugkonzeptoptimierung in einer multimodalen Verkehrsumgebung. Dissertation. Technische Universität München, München. Institut für Maschinen- und Fahrzeugtechnik, Lehrstuhl für Fahrzeugtechnik.

Ried, Michael (2014): Lösungsraumanalyse für Plug-In-Hybridfahrzeuge hinsichtlich Wirtschaftlichkeit und Bauraumkonzept. Dissertation. Universität Duisburg-Essen, Duisburg-Essen. Abteilung Maschinenbau und Verfahrenstechnik.

Schuh, Günther; Lenders, Michael; Nußbaum, Christopher; Rudolf, Stefan (2012): Produktarchitekturgestaltung. In: Günther Schuh (Hg.): Innovationsmanagement. Berlin, Heidelberg: Springer Berlin Heidelberg, S. 115–160.

Stocker, Johannes; Schmidt, Michael; Kreimeyer, Matthias; Lienkamp, Markus (2016): Development of market-oriented architectural standards by means of standardized vehicle layouts. In: Design Society (Hg.): Proceedings of NordDesign 2016. Trondheim, Norway 10.08.-12.08.2016.