Knowledge Based Engineering to Support Automotive Conceptual Design and Automatic Control Software Development

Fengnian Tian and Mark Voskuijl

Keywords Multi-model generator · Knowledge based engineering · E/E system integration · Conceptual design · Electric vehicle

1 Background

The global motor vehicle production is rising steadily year by year. These vehicles have an increasing amount of electronic components and associated control software. As a result, the control software development becomes a key aspect and time consuming part of the design. In contrast to the rising production numbers, the number of safety recalls is in fact decreasing steadily [1]. However, it should be noticed that the number of vehicles being recalled because of failures in the electronics, either software or hardware, was in fact increasing significantly in the period 2000–2010, which is shown in Fig. 1.

To some degree, the increasing electronic malfunctions on automobiles can be explained by the fact that the development processes of vehicle and the development process of the related *Electric and Electronic Systems* (E/E systems) is not integrated. In other words, because almost all functions on the vehicles are electronically controlled nowadays, the complexity of overall E/E systems rise sharply along with the increasing number of vehicle variants. The established vehicle development processes which toward to efficiently create high quality mechanical systems cannot deal with the problem of high complexity of the E/E systems [3]. Moreover, the analysis and specification for the architecture of the logical system, the technical system and the software itself includes many repetitive processes in conventional

F2012-D03-005

F. Tian (✉) · M. Voskuijl
Delft University of Technology, Delft, The Netherlands
e-mail: f.tian@tudelft.nl

SAE-China and FISITA (eds.), *Proceedings of the FISITA 2012 World Automotive Congress*, Lecture Notes in Electrical Engineering 194,
DOI: 10.1007/978-3-642-33829-8_37,

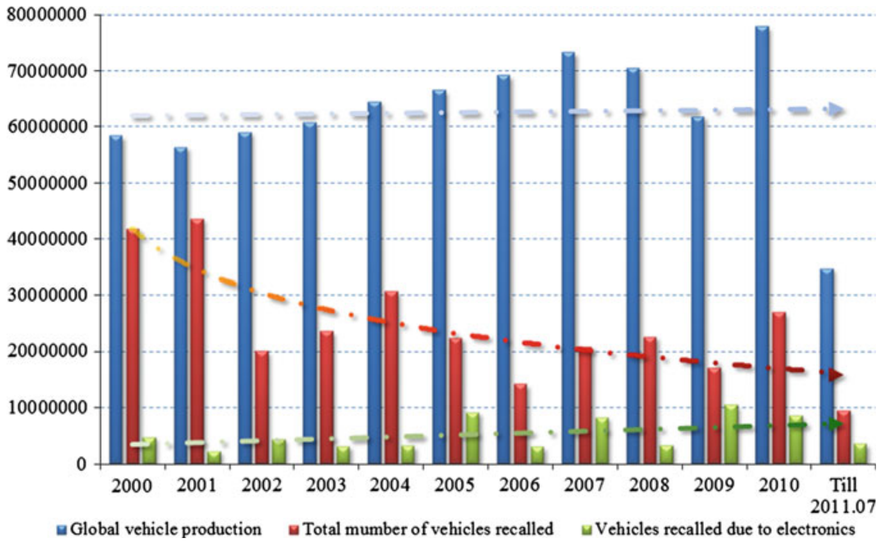

Fig. 1 Comparison of the global vehicle production, the potential number of vehicles recalled and the number of units affected due to electronics [1, 2]

automotive software development [4]. Those repetitive processes are time consuming and are prone to errors.

System engineering is introduced by the *Original Equipment Manufacturers* (OEMs) as a sustainable approach to the development of reliable E/E systems. Nevertheless, although it is well known that the logical system architectures, technical system architectures as well as software functions of the vehicle E/E systems should be integrated into the complete vehicle development [3], a number of issues have still to be solved in order to release the full system engineering potential.

Firstly, *Computer Aided Design* (CAD) tools, which are widely used by the conventional mechanical dominated vehicle development processes, are not suitable for developing the logical and technical system architecture of the E/E systems. For example, the logical system architecture is usually described by graphical notations, such as block diagrams and state machines [4], which is not available in the CAD tools. Moreover, after the specification of the logical system architecture, the technical system architecture must consider all constraints of a technical and economic nature, as well as those concerned with organizational structure and manufacturing technology [4]. Only a few of those constraints are related to about geometry. Most of them are engineering disciplines, which are difficult to capture by CAD software. Secondly, the software had to be tuned several times due to any changes from the logical or technical system architecture, which is also difficultly supported by the CAD tools. Usually, the inputs of software functions are parameters and variables which describe the physical nature of a component or an assembly. For instance, it is well known that the axle load distribution has effects on the handing stability, *Anti-*

lock Braking System (ABS), etc. Nevertheless, such information cannot be read directly from the geometry model but requires computational analysis.

Therefore, the objective of the current research study is the development of new design methods and tools that allows the designer to take the development of the E/E systems including logical system architecture, technical system architecture and control software, into account already in the conceptual design stage of novel vehicles. The aim of these new methods and tools is to reduce the development time of these novel more electric vehicles and to create more consistent control software. The proposed design methods and tools can in principle be applied to any dynamic system with a high level of software integration, such as e.g., unmanned aerial vehicles.

2 Methodology

The methodology used in this research is based on the *Design and Engineering Engine* (DEE) concept, which has been developed by La Rocca [5]. The DEE is a modular computational design system to support distributed multidisciplinary design and optimization. It can support the design process of complex products and accelerate the process through the automation of non-creative and repetitive design activates [6]. Previously, the DEE is developed mainly for aircraft design. In this research, we expand the border of DEE to the automotive conceptual design, including E/E systems. A schematic representation of the proposed DEE is shown in Fig. 2.

As can be seen, the DEE is composed of several software modules. The first module is the *Initiator*, which is responsible for the initialization of the values of input parameters for the subsequent modules, such as wheelbase, track, weight, engine type, etc. It can work both in knowledge mode or custom mode. Under the knowledge mode, the user inputs general product planning, like vehicle type or number of passengers, and then the *Initiator* searches the design database to find similar existing designs, and produces the parameters automatically. The user could also directly select available assemblies from the component database to build a prototype vehicle under the custom mode.

The next element of the system, named *Multi-Model Generator* (MMG) is the heart of the DEE. The MMG is a *Knowledge Based Engineering* (KBE) application, which is able to model different automobile configurations and configurations' variants. Besides modeling high fidelity geometric models, the MMG can also output specific data for various analysis tools directly and automatically. For example, the MMG can generate the complex surface model of the car body, and then translates that model into clouds of points or panels which are required by the aerodynamic analysis tool.

The model abstractions produced by the MMG are analyzed by corresponding analysis tools. If the results meet the requirements in each disciplinary area, the process is stopped and the output is a final design for the detailed design stage; otherwise, the DEE will generate a new vehicle configuration or variant and perform iteration again. An optimization process can be included in the iteration.

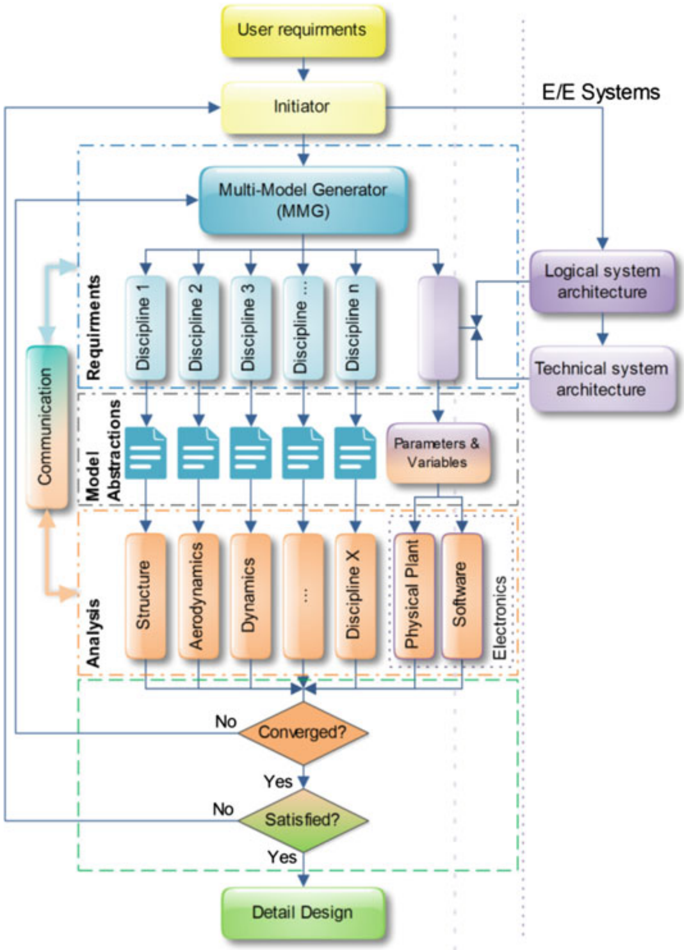

Fig. 2 Overview of the DEE

However, that is not the focus of the current study. It should also be noticed that there is an agent based communication model [5] connecting the MMG and the multidisciplinary analysis tools in Fig. 2. The communication model not only exchanges the data and information between various DEE models, but also controls the overall process. In this paper, a communication model is established between the MMG and MATLAB for the purposes of dynamic and control system simulation, which is shown in test case section. The communication model makes it possible to have elements of the DEE on different computers, which can be located even in different companies/institutes. Thus it also allows collaborative design efforts.

The DEE for automotive conceptual design is developed in the Genworks' *General-Purpose, Declarative, Language* (GDL), which is based on the ANSI

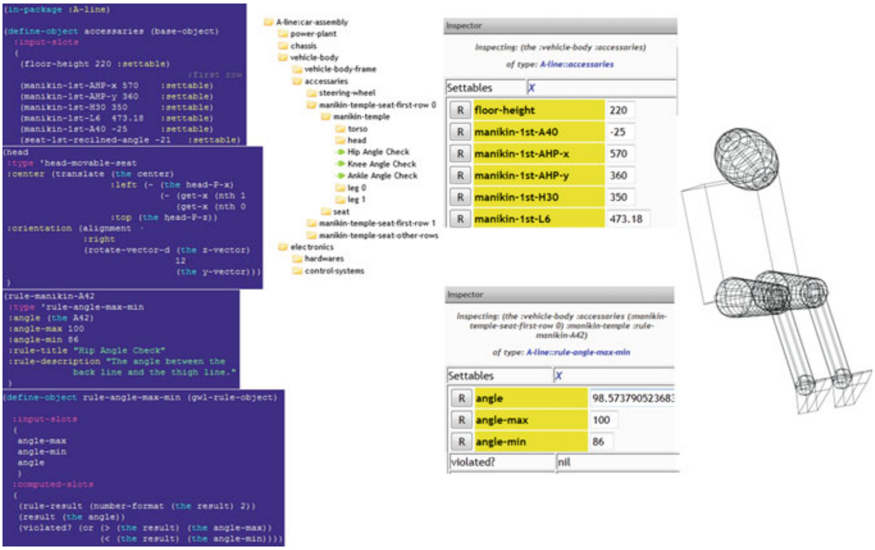

Fig. 3 Correspondence between source code and web-centric user interface for the MMG

standard version of Common LISP. GDL is particularly effective at representing complex systems, including three-dimensional geometric models and design process [7].

3 Integration Automotive E/E Systems Development into Vehicle Conceptual Design

3.1 A Brief Introduction of MMG for Automotive Conceptual Design

A *Multi-Model Generator* (MMG) for automotive conceptual design has been developed in this work. Besides complex geometry modeling, the MMG is able to capture the methods, process as well as various engineering disciplines. It can not only provide various models for multidisciplinary analysis but also support the automation of repetitive processes. In general, the MMG which is a tree structure is composed of a component library and a rule base.

All the components necessary to complete a vehicle conceptual design are defined as objects in the component library. Every object is a parametric model with required inputs. The user can specify those inputs to get an object variant. Objects can be fitted together with designed methods to create a component. Several components can be put together to generate an assembly. For example, as one can see in Fig. 3, the manikin template included in the accessories (indicated by red arrow) is an assembly of head, torso and leg. The head contour is a component. The user can specify the length of the X, Y, and Z axes of the head contour in the custom input interface. The position and the reclined angle of the head contour are defined by the attributes*: center* and *:orientation* (marked by purple box) relative to the vehicle grid.

The rule base is designed to contain the requirements from various scientific areas. Those requirements are abstracted as one or a combination of the rules in terms of logic, math, geometry, configuration selection and communication. As mentioned before, besides geometry objects, there are also several rules integrated in the manikin template, such as hip angle check (pointed by green arrow). If we change the back angle A40 (displayed as manikin-1st-A40 in the red dash line box) from −25 to −27°, the 'angle' in the green dash line will increase from about 98.57 over 100° immediately, which exceeds the range of recommended comfort hip angle from 86 to 100° [8]. The value of 'violated?' will change from current 'nil' to 't' automatically in the interface, which means this rule is violated.

A part of the source code of the MMG and corresponding web-centric user interface is shown in Fig. 3. The GENWORKS GDL also provides the bottom layer connections from the source code compiler to the web-centric user interface. Such an interface can be loaded in any internet browser and keep updating with the compiler. As a KBE application, components and rules in the MMG are tightly integrated. If an input to the component is changed in the user interface, any rules and components which directly or indirectly depend on that input will automatically re-evaluate themselves and show results in the interface immediately. Rules and objects which are not affected by a modification will avoid the re-evaluation [9].

3.2 Integration Logical and Technical System Architecture of E/E Systems into the MMG

In general, the logical system architecture determines the performance the system will deliver but avoid the specific manner of its implementation [4]. In this study, the logical system architecture is summarized as a design discipline and included into the rule base of the MMG. Similarly, the physical parts of the technical system architecture are modeled as combination of objects included in the component library. The various constraints which have to be fulfilled by the technical system architecture are also defined as disciplines comprised by the rule base of the MMG.

3.3 MMG Supports for Software Function Development

Besides geometry, the MMG can also output specific data for various analysis tools, which is capable of supporting the validation of software functions. The

MMG can save the properties of the objects or pre-processing results in a common data format such as XML or ASCII, which can be used to set the values of parameters and variables needed by the software functions. For example, once a modification happens on the main assemblies, the MMG will re-calculate the axle load distribution immediately and send the data to the software functions. Then the results from the software validation could feedback to the MMG again to check whether some rules are violated. With the communication model of the DEE, such processes can be automatically finished repeatedly, which is suitable for getting the optimized solutions.

4 Test Case

4.1 A Novel Electric Vehicle Configuration

It is well known that the driving range of electric vehicles is always a critical technical specification for the designer, car manufacture and consumer. Nowadays, the driving range is limited by the battery pack equipped on the vehicle. If the aerodynamic drag and the curb weight can be lowered simultaneously, the driving range can be increased significantly, especially at high speeds. In the current research study, a novel electric vehicle configuration named A-line, where passengers are seated in line, is proposed. Compared with a conventional vehicle configuration, the advantages of this novel configuration are threefold. First of all, the vehicle will have a significantly lower aerodynamic drag due to a small frontal area. Second, the curb weight can be decreased because it is designed for two or three person seated in line. Finally, it will require a much smaller parking space because of its narrow track. In principle, several A-line vehicles can drive on the road side by side at the same time, which has the potential to alleviate traffic congestion. Besides all the advantages mentioned above, the vehicle is an excellent test-case for the novel design system proposed in this research study because; (1) it requires the integrated design of a novel E/E system, and (2) the concept is new, so one cannot rely on existing designs.

Two A-line models have been generated by the MMG, which is shown in Fig. 4. Their technical specifications are list in Table 1. It can be seen that both models have a smaller frontal area and curb weight than conventional configurations.

4.2 A Communication Model of the DEE

In order to validate handing stability and range of the A-line models above, a communication model has been established in this paper, as depicted in Fig. 5. Firstly, the MMG generate a *xxx.csv* file which includes the all the values of the

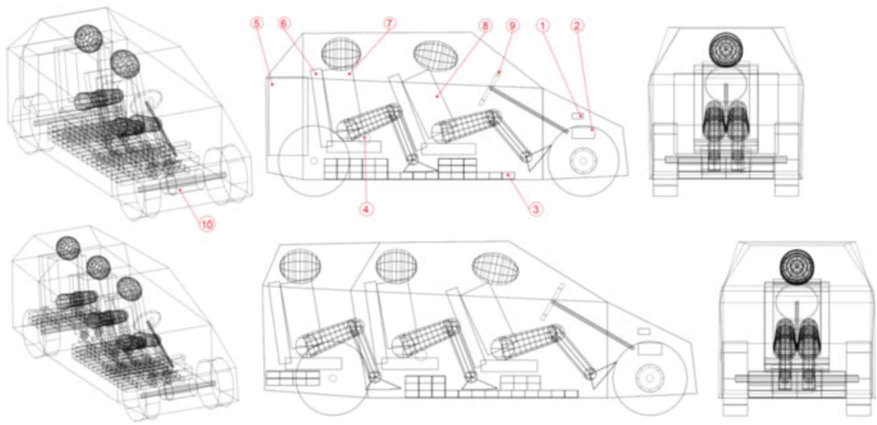

Fig. 4 The geometry of A-line generated by the MMG (*1*—electronic control unit (*ECU*), *2*-DC–DC—converter, *3*—battery pack, *4*–charger, *5*—cargo, *6*—seat assembly, *7*—passenger manikin template, *8*—driver manikin template, *9*—steer wheel, *10*—electric motor)

Table 1 Specifications of the A-line vehicles with two or three passengers

A-line	Model	Two passengers	Three passengers
Dimensions	Wheelbase [mm]	2,400	2,700
	Track F/R [mm]	1,050	1,050
	Overall length/width/height [mm]	3,250/1,250/1,455	3,700/1,250/1,485
	Ground clearance [mm]	150	150
	Frontal area [m^2]	1.54	1.57
Weight	Curb weight [kg]	855	1,110
	Axle load distribution	59 % (empty) 52 % (full)	60 % (empty) 48 % (full)
Performance	Pemax [kW]	80	100
	Temax [Nm]	220	240
	Vmax [km/h]	120	120
Battery	Type	LiFePO4 Lithium	LiFePO4 Lithium
	Number of cells	80	80
	Nominal capacity [Ah]	70	70
	Nominal voltage [V]	3.2	3.2
	Mass [kg]	200	200
Capacities	Seat capacity	2	3
	Cargo volume [L]	220	200 (rear seat up)
Accessories	Tire	205/55 R16	205/55 R16

parameters and variables required by MATLAB simulation. Then a server is opened in MATLAB, establishing a local host for the MMG (Our server is established based on a common lisp interface to MATLAB [10]). At this moment, MATLAB commands can be written in the source code compiler of the MMG to

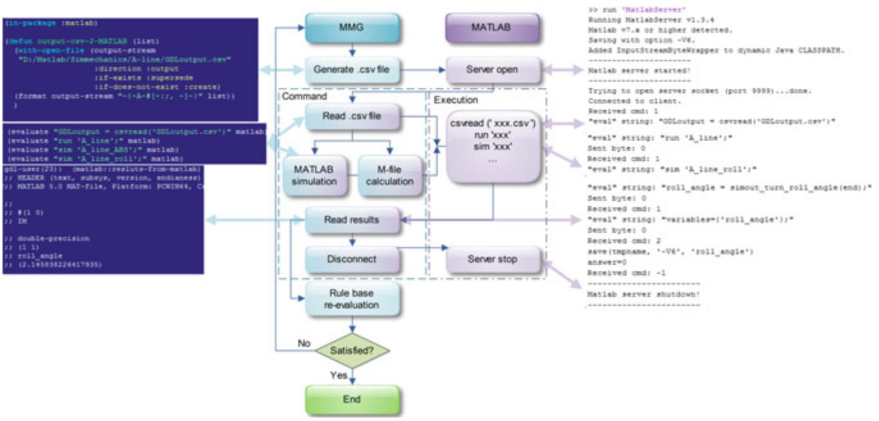

Fig. 5 Communication model between the MMG and MATLAB

control the objects in MATLAB, such as an M-file or a SimMechanics (multibody dynamics) model. After the execution of the M-files or MATLAB simulations, the results can be saved in any structure data type and read by the MMG. In the next step, the rule base will be re-evaluated again to check whether some rules are violated. If all the rules are satisfied, the whole process is finished. Otherwise, the MMG can start a new iteration.

4.3 Validation of Handing Stability

Because of the narrow front and rear track as well as a short wheelbase, the handing stability of the A-line vehicles needs validation. Usually, the roll angle of vehicle body should be around 3 degrees, no more than 7 degrees when the vehicle is turning at 0.4 g centripetal acceleration with a constant speed and turning radius. Furthermore, the pitch angle should be smaller than 3 degrees when the vehicle is braking at 0.4 g [11]. Therefore, those two rules are set as requirements included in the rule base of the MMG. We build a dynamic model of the A-line in SimMechanics of the MATLAB. Through the communication model in Fig. 5, all the values of parameters and variables needed by the dynamic simulations are provided by the MMG according to Table 1.

4.3.1 Roll Angle Validation

The initial condition of the roll angle validation simulation is the equilibrium position at standstill. After one second, the vehicle is accelerated and a turn is

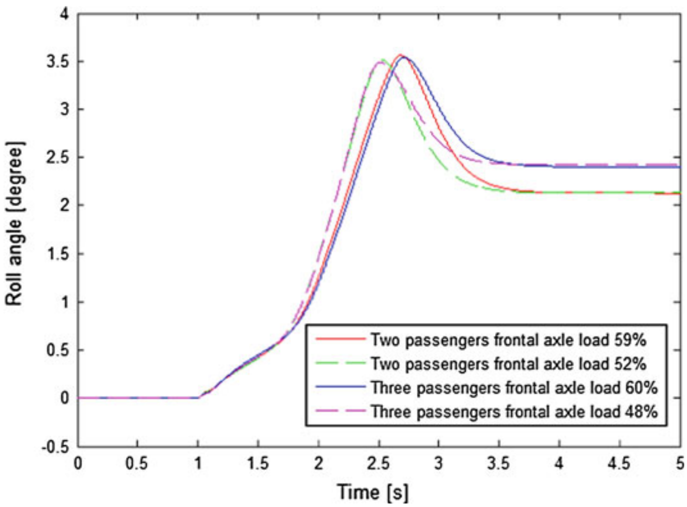

Fig. 6 Roll angle inspection for A-line vehicles

initiated until the desired condition (velocity 6 m/s and turn radius 10 m) is obtained. The resulting roll angle simulation for the A-line vehicles, with different loading conditions is shown in Fig. 6. It can be seen that both A-line configurations comply with the roll angle requirement for all loading or empty conditions.

4.3.2 Pitch Angle Validation

The validation of the pitch angle requirement is related to the vehicle braking. In order to show the integration of the E/E systems and the MMG, an ABS module is built both in the MMG and SimMechanics. Because the focus of the current study is the integration of the two systems, the controller of the ABS module in the SimMechanics is simply set up based on the error between actual slip and desired slip. As described in Sect. 4.1, first, the logical system architecture of the ABS is defined as several rules in the MMG, like maximum braking distance and pitch angle of vehicle body. Second, the physical plant representations of the technical system architecture of the ABS are generated by the MMG, which includes wheel brakes, hydraulic modulator, sensors, etc. Third, together with the entire parameters list in Table 1, some physical plant attributes needed for the tuning of software functions are also transferred from the MMG to the SimMechanical plant through the communication model. For example, as one can see in Fig. 7, the pressure of hydraulic modulator set in the web-centric user interface is sent to the model in MATLAB by the *.csv* file.

The dynamic model starts to brake at 2 s with the initial speed 100 km/h. The results of the ABS simulations are shown in Fig. 8. Similar to the roll angle validation, all the configurations of the A-line vehicles satisfy to the pitch angle requirements.

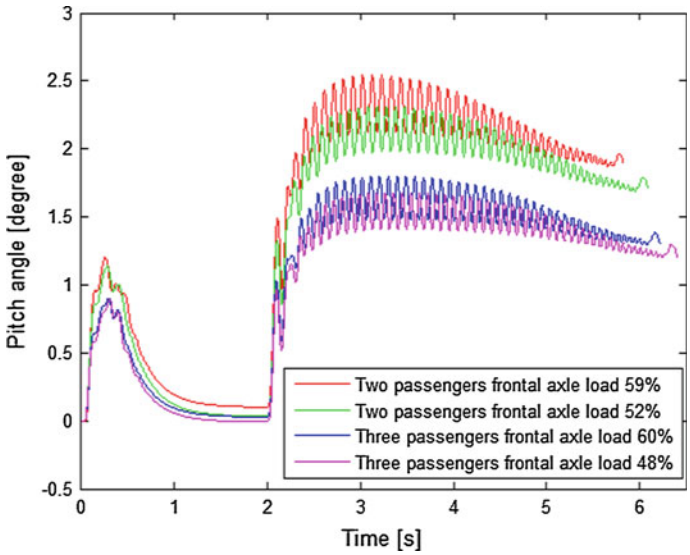

Fig. 7 The ABS module in the MMG

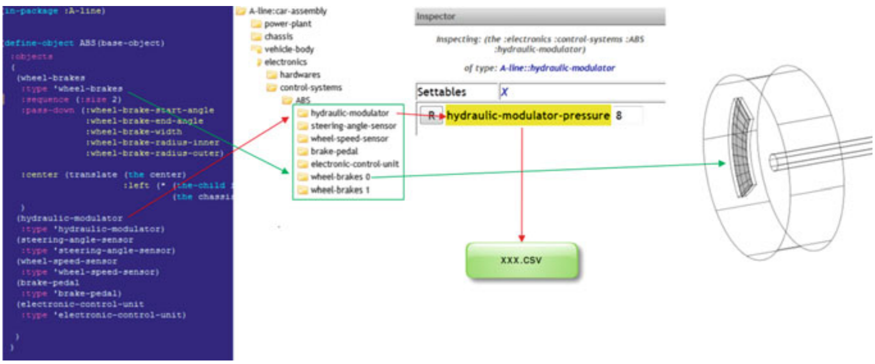

Fig. 8 Pitch angle inspection for A-line vehicles

4.4 Validation of Range

The main advantage of the A-line configuration is its large driving range. In order to demonstrate this, the driving range of the A-line vehicles is compared with the General Motors EV1, whose batteries were rated at 60 amp-hours (18.7 kWh) at 312 V [12]. To give a fair comparison with current batteries for which data is available, we use 70 amp-hour (17.9 kWh) Li batteries at 256 V. The range calculation method is based on the work of Larminie [13]. Both EV1 and A-line vehicles are tested in a simplified federal urban driving scenario. It is supposed that most electronic accessories are switched on, such as headlights, radio and heater.

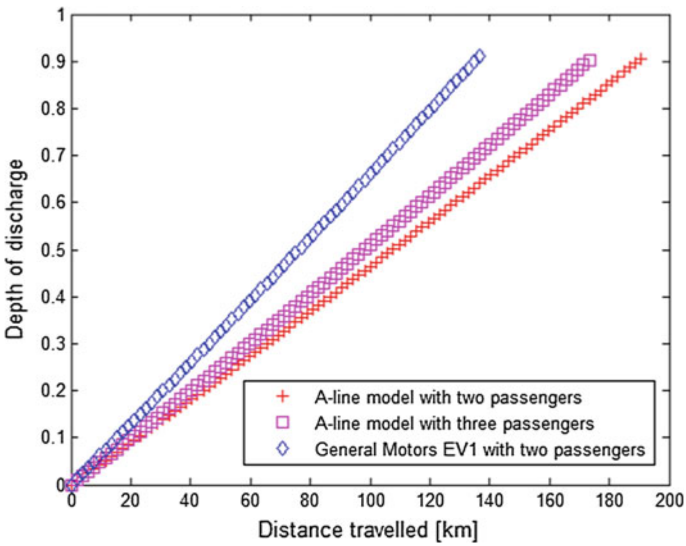

Fig. 9 A comparison of the distance travelled between the A-line vehicles and General Motors EV1

The drag coefficient is estimated 0.33 for A-line vehicles. Both two A-line models above are compared with the General Motors EV1, which is shown in Fig. 9.

It is apparent that when 80 % discharge is reached, the A-line model with two passengers has already travelled for 169.4 km, followed by the three passengers' model (155.9 km). The General Motors EV1 reaches 121.3 km, which is in agreement with the official driving range published for the EV1 [11].Thus, a 28 % driving range improvement is obtained by a change in the vehicle configuration whilst keeping other design parameters such as the type of batteries, motor, etc. identical.

5 Conclusion

The concept of DEE for automotive conceptual design has been proposed in this paper. The MMG which is a KBE application has been developed in this work. It has been proven that the MMG can generate models both for conventional and novel electric vehicle configurations and variants. Moreover, the MMG is able to support the development of automotive E/E systems from the logical, technical system architectures to the software functions, integrating the E/E systems into the automotive conceptual design. Finally, a novel electric vehicle configuration named A-line has been tested for handing stability and range. It has been validated that the A-line can drive much longer range than conventional electric vehicles.

References

1. OICA (2011) World Motor Vehicle Production[EB/OL]. [2012-02-01]. http://oica.net/wp-content/uploads/total-2011.pdf
2. safercar.gov (2011) Flat File Copies of NHTSA/ODI Databases[EB/OL]. [2012-02-01]. http://www-odi.nhtsa.dot.gov/downloads/.
3. Weber J (2009) Automotive development processes. Springer, Berlin, p 64
4. Schäuffele J, Zurawka T (2005) Automotive software engineering. SAE International, Warrendale, p 167
5. La Rocca G, van Tooren MJL (2007) Enabling distributed multi-disciplinary design of complex products: a knowledge based engineering approach. Design Res 5(3):341
6. van Tooren MJL, Nawijn M, Berends J et al (2005) Aircraft design support using knowledge engineering and optimisation techniques. In: 46th AIAA/ASME/ASCE/AHS/ASC Structures, structural dynamics and materials conference, Austin, Texas
7. GENWORKS. Genworks International[EB/OL]. [2011-09-20]. http://www.genworks.com
8. Fenton J (1998) Handbook of automotive body and systems design. Redwood Books, Great Britain, p 41
9. Cooper J (2004) Knowledge base techniques in web applications: a tutorial for genworks' GDL/GWL, p 46
10. Fenton J (1998) Handbook of automotive body and systems design. Wiley, London, p 41
11. Liu YX (2001) Automotive design. Tsinghua University Press, Beijing, p 40
12. Wikipedia. General Motors EV1[EB/OL]. [2011-09-01]. http://en.wikipedia.org/wiki/General_Motors_EV1
13. Larminie J, Lowry J (2003) Electric vehicle technology explained. Wiley, West Sussex, pp 203–204