Adaptive Software Component for Predicting Energy Consumption of an Electric Vehicle on a Planned Route

Dmytro Humennyi[1]

[1]N-iX Ltd.; Kyiv National University of Construction and Architecture
Povitryanykh Sil avenue, 31, Kyiv, 03037
[1]apollo.d.g@gmail.com, https://orcid.org/0000-0001-6736-0543

Received 20.08.2024, accepted 01.11.2024
https://doi.org/10.32347/uwt.2024.15.1205

Abstract. This paper presents the development of an adaptive software component for forecasting the energy consumption of electric vehicles along planned routes. It includes a conceptual framework, architectural design, and modular software composition, introducing mathematical models for managing energy consumption with a model-based design approach for precise predictions and optimization. The importance of data sources such as route information, vehicle condition, and driver behavior is emphasized to create a comprehensive state vector for energy optimization.

Following ISO 26262 and A-SPICE 3.1 standards, the implementation uses a model-based approach with Simulink and aligns with the V-Model for rigorous validation. The methodology details segmenting routes and optimizing energy consumption for each segment, considering driving style and environmental conditions. The gradient search method adjusts energy consumption to minimize usage while maximizing comfort and ensuring route completion.

This research lays the groundwork for future advancements in predictive energy management systems for electric vehicles, with potential real-world applications. Future work will focus on refining predictive models, exploring machine learning for improved accuracy, and integrating real-time data from connected vehicle technologies for dynamic optimization.

Keywords. Electric vehicles, energy consumption prediction, model-based design, energy management system, route optimization, driving behavior, adaptive software, ISO 26262, A-SPICE 3.1, gradient search, Simulink, vehicle state vector, modular architecture, predictive models, machine learning, real-time data integration, connected vehicle technologies, battery management, comfort optimization.

Dmytro Humennyi
Ph.D., Engineering Manager at N-iX, Associate Professor of the Department of Cybersecurity and Computer Engineering

INTRODUCTION

Modern cars are complex networks of computing and control devices, each with its own topology that imposes specific properties, limitations, and norms. This distributed system for data collection, communication, and control enables the formation of a detailed state vector, accurately describing the car's condition for effective technical management decisions.

Simultaneously with the development of control means and internal communication of the car, its energy system also develops. This is particularly relevant for electric vehicles equipped with mobile power plants or electric batteries.

The advantages of using electric power elements for vehicle movement are significant and varied. Firstly, electric vehicles (EVs) help reduce or even eliminate fuel costs, as electricity is cheaper and more stable in price compared to gasoline [1], [2]. Secondly, EVs contribute to environmental conservation by producing fewer emissions than their gasoline counterparts, thus aiding in the fight against climate change [3], [4]. Additionally, electric vehicles support energy independence, reducing reliance on imported oil and promoting the use of locally sourced renewable energy [1], [2].

However, there are still unresolved issues related to the vehicle's range on electric traction. Modern vehicles are limited in their range to approximately 100-450 kilometers (about 279.62 mi) per charge, depending on the model and battery capacity [3], [12].

The issue of increasing the range is addressed by several approaches, including improvements in battery technology, development of more efficient power management systems, and the expansion of charging infrastructure [2], [4]. Another approach to increasing the vehicle's range is to introduce methods of energy consumption prediction with the search for an optimal consumption strategy for each energy consumer. This strategy also considers the dependence of expenses on driving style.

Therefore, predicting the energy consumption of a vehicle for moving along a given trajectory is the subject of this study. The object of the research is the development of application-level software (business logic) for predicting energy consumption and forming a system of recommendations for the driver.

The task of minimizing a vehicle's energy consumption, as well as the task of predicting its range, is primarily a task for battery engineers and component designers of the car both in its details and as a finished product. But secondly, it is also a task for companies that develop software, especially in connection with the concept of building a Software-defined vehicle. Thus, N-iX corporation, within which these studies were conducted, approached the described subject through the prism of Predictability of energy consumption and energy management system, which is a software solution for vehicle energy management.

PROBLEM STATEMENT

The objective of this research is the development of a composition of software tools and individual software components for a vehicle that enables the optimization of the electrical consumption scenario of each critical node of the vehicle. This ensures the achievement of the planned route with the maximum level of comfort.

It should be noted that the problem involves solving a multi-criteria optimization task of a bi-criteria optimization of a multi-parameter function with nonlinear dependencies between parameters.

Two criteria are proposed:

1. Ride comfort (minimizing the deviation between the ride properties set by the driver and the actual battery capacity allocated to each consumer).

2. Minimization of energy consumption (minimizing the energy consumption of each analyzed vehicle energy consumer).

Regarding energy consumers, the aim is to create a software composition that allows for the customization of energy consumers. The basic composition should manage the main consumers, namely Powertrain, HVAC, Suspension, Battery Climate Control System, and expenses related to driving style. Additional consumers can be added by creating corresponding software components and adding them to the composition.

Regarding information sources, several information sources are needed for energy management, which can be grouped as follows:

- Information about the route;
- Information about the vehicle's condition;
- Information about the driver and driving style.

All three basic information sources create a state vector of the vehicle. This state vector, together with the criteria for optimizing energy consumption, forms the problem statement for nonlinear optimization. Solving this problem leads to obtaining an optimal energy consumption scenario.

MATERIALS AND METHODS

According to the defined object and subject of the research, as well as the aim of the work, the outcome is the creation of a software composition that can be applied as business logic and function as an application within the ECU. The composition should meet the design, documentation, construction, testing, and integration requirements according to norms, standards, frameworks, and guidelines applicable in the automotive sector.

This composition includes an ASIL A class component. The SDLC V-Model and ISO 26262 [15] and ISO 33050 [16] standards were applied for its development. Project measures, as well as processes during preparation, development, and support, complied with the A-SPICE 3.1 framework. Technologically, they correspond to level 3 (Established). The teams applied the SAFe 6 project management methodology.

A Model-based approach based on Simulink (license 41148027) was used. This approach was employed from the requirement engineering stage to the integration stage. The software component includes means for interacting with the driver. For demonstrating the interaction, a virtual reality solution based on Carla Simulation was created. Physical steering wheels and pedals from Logitech were used for driver input. C99 and Python layers were used to integrate Carla with Simulink and the steering wheel and pedals with Simulink.

For obtaining route data and information about road temperature and quality, Google Maps services were used. Python scripts were employed to link Google Maps with Simulink.

The MathWorks ecosystem, including Simulink, System Composer, Requirement Toolbox, Embedded Coder, Advisor, Simulink Test, Powertrain, and Instrument Control Toolbox (license No. 41148027), was used for building the composition, components, integration, and requirement engineering.

THE TASK OF FINDING THE OPTIMAL ENERGY CONSUMPTION SCENARIO

There are several approaches to formulating the task of finding the optimal energy consumption scenario. For example, [8], [9] define it as minimizing energy expenditure. In contrast, [10], [11] use Markov chains to predict step-by-step expenses.

At N-iX Corporation, we approached this task creatively. We decompose the entire vehicle route into segments (the length of the segment is chosen dynamically, depending on changes in its characteristics. For example, if the road is straight, without significant changes in road angle or temperature, we consider this part of the route as one segment). After decomposition, we request the required current. This request is made to the consumption model, which we describe for each consumer. Thus, we estimate consumption for each segment of the route we need to cover. Then we request the driver's profile. Based on statistical data, we understand the driver's driving style. We interpret this style as an over-expenditure coefficient for each consumer. Next, having the consumption values for all consumers for all route segments, we can estimate whether the battery charge will be sufficient to cover this route.

Obviously, the route can be covered quickly or slowly, with the air conditioner at maximum power or turned off, using adaptive suspension, or switching it to passive mode. And it is also obvious that all this changes energy consumption. Therefore, the task of finding the optimal energy consumption scenario is to find the operating modes of each consumer so that we can cover the planned route.

It is important to note that the best chance of reaching the planned route is to turn off everything that is possible and switch the powertrain to the highest economy mode. But this is not a comfortable ride. Therefore, we introduced another criterion – maximizing comfort.

Thus, if we understand that we can reach the destination, and if the air conditioner operates in normal mode (provided that the driver wants it to be turned on in normal mode), we will propose to turn it on in this mode. If we cannot reach the destination in this mode, we will propose a balanced combination of modes for all consumers to evenly reduce comfort so as not to critically reduce it anywhere.

And, obviously, if we estimate the impossibility of covering this route at all, then after the appropriate calculation, we notify the driver.

Thus, the mathematical model for finding the optimal energy consumption scenario is proposed as follows:

1. Minimization of energy consumption (E):
   a. Minimization of the total energy consumption of each component.

b. We define the function E as the sum of the energies consumed by each component:

$$E = E_{powertrain} + E_{HVAC} + E_{suspension} + E_{batteryCC} + E_{driver}$$

2. Maximization of comfort (C):
   a. Minimization of the deviation between the ride properties set by the driver and the actual battery capacity allocated to each consumer.
   b. We define the comfort function (C):

$$C = - \sum_{i}^{\max(i)} \left(\dot{E_\iota} - E_i\right)^2,$$

where $\dot{E}i$ - is the desired energy consumption set by the driver for the i-th component, and Ei - is the actual energy consumption for the i-th component.
Then, the Lagrange function for this task can be written as follows:

$$L(E,\lambda) = E + \lambda(C - \alpha)$$

where λ - is the Lagrange multiplier, and α - is the constant that determines the balance between energy consumption and comfort.

We form a system of equations. So, to find optimal values, it is necessary to find the partial derivatives of the Lagrange function with respect to each variable and set them to zero:

$$\frac{\partial L}{\partial E_i} = 0, \ \frac{\partial L}{\partial \lambda} = 0 \quad (1)$$

For the segment of the route, we form a current request:

$$I_{segment} = \sum_{i}^{\max(i)} \frac{E_i}{U_i}$$

where U_i - is the network voltage for the i-th component.
Driver profile consideration is carried out as follows:

$$E_{driver} = \beta \sum_{segment}^{all\, segments} E_{segment}$$

where β - is the coefficient that accounts for the driver's driving style.

OPTIMIZATION OF CONSUMPTION FOR EACH SEGMENT OF THE ROUTE

The objective function will be as follows:

$$MIN\, E = E_{powertarin} + E_{HVAC} + E_{suspension} + E_{driver}$$

with constraints:
- Constraints on energy consumption for each component:

$$E_i \leq E_{max,i}, \ i = 1, \dots, n$$

- Constraints on total energy consumption:

$$\sum_{i=1}^{n} E_{i \leq E_{total}}$$

- Constraints on comfort reduction:

$$C = -\sum_{i=1}^{n} \left(E_i - E_{\dot{total}}\right)^2 \geq C_{min}$$

Finding optimal values E, λ is based on solving equation (1), which can be presented in the following form:

$$\frac{\partial L}{\partial E_i} = \frac{\partial}{\partial E_i}\left(E + \sum_{j=1}^{m} \lambda_j(E) - b_i\right) = 0 \quad (2)$$

for constraints: $g_j(E) - b_j = 0, \ j = 1, \dots, m$.

This completes the formulation of the optimization task, ensuring a balanced approach to minimizing energy consumption while maximizing driver comfort.

FORMATION OF THE SOFTWARE COMPOSITION

According to the A-SPICE framework, the software development process includes a series of preliminary stages. These stages will be omitted in this publication as they fall outside the scope of its scientific interest. The system engineering process (SYS A-SPICE group) is partially considered here, as well as the ENG group, in accordance and proportionate to the relevance of the subject of the work. The SUP and MNG groups are completely omitted in the publication.

DISTINGUISHED COMPOSITION COMPONENTS

The composition is divided into three groups of components. It is assumed that all components will be deployed on a single ECU.

First Group - Core Composition. Contains components that are not subject to customization. This group of components includes:

- Prediction SWC;
- Powertrain SWC;
- DriverProfile SWC.

Second Group – Plugins. Contains mathematical models of consumers. It allows for the current request to the power source. It also allows for modeling current consumption under different operating modes in search of the optimal combination. Each of the plugins has two subcomponents. The first subcomponent is used for iterating through energy consumption modes in search of an optimal configuration. The second subcomponent provides a current request in the selected mode depending on the operating conditions. This group's software components allow for customization according to the vehicle's specifications. Typical representatives of the second group of software components include:

- ClimatControl SWC;
- Suspension SWC;
- ExternalTemperature SWC;
- TirePressure SWC.

Third Group – Wrappers. They are used for software testing, debugging, and demonstrating its operation. Typical representatives of software components in this group include:

- GoogleMap SWC;
- HMI SWC;
- Modem SWC.

The proposed composition of software components is shown in Fig. 1.

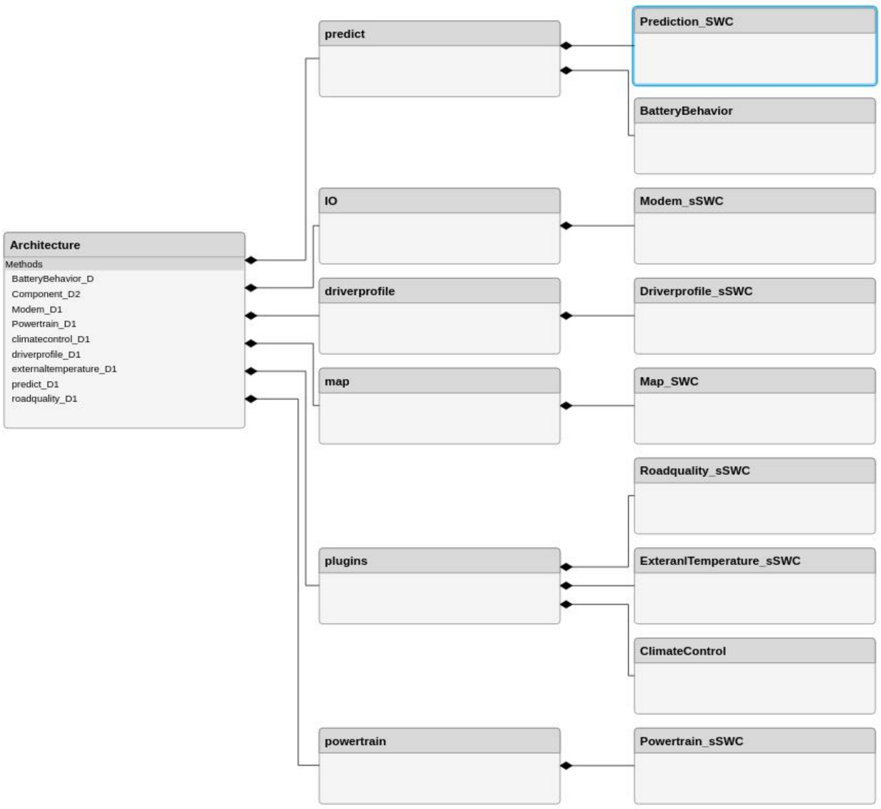

Fig. 1. Software Composition

While changing plugins should not affect the composition, the entire second group of software components is grouped into the Plugins SWC component, as shown in Fig. 2.

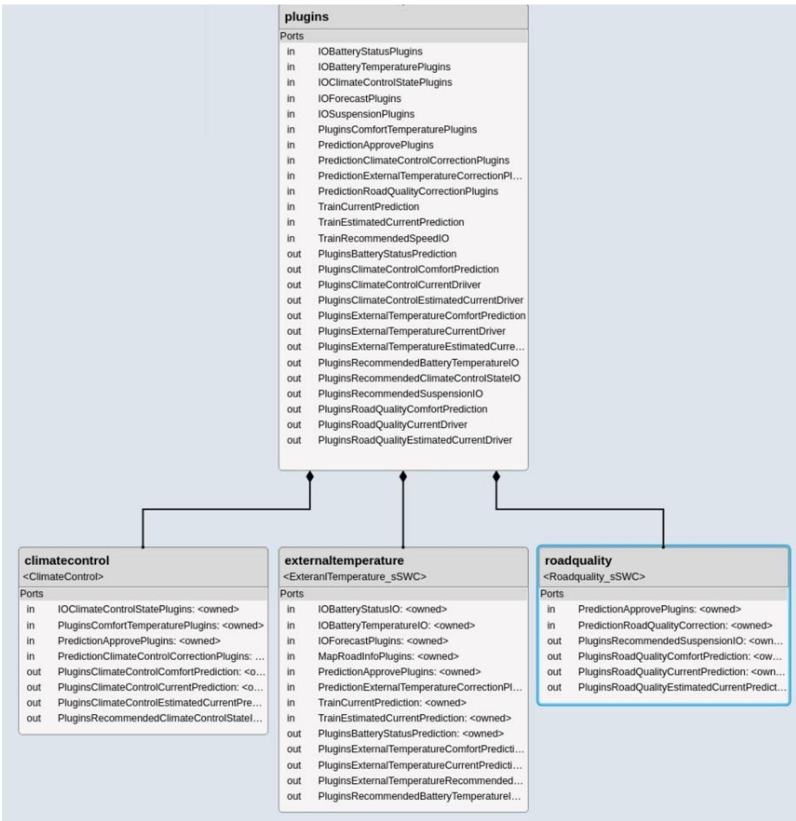

Fig. 2. Decomposition of Plugins into a Separate Group

A typical component of the composition is composed according to the MISRA AC GMG [13] or MAAB [14] guidelines. Its structure includes two subcomponents mentioned earlier (one for finding a new consumption plan using the gradient descent method, and the other for performing predictions of expenses for subsequent parts of the route). The typical structure of such a software component is shown in Fig. 3.

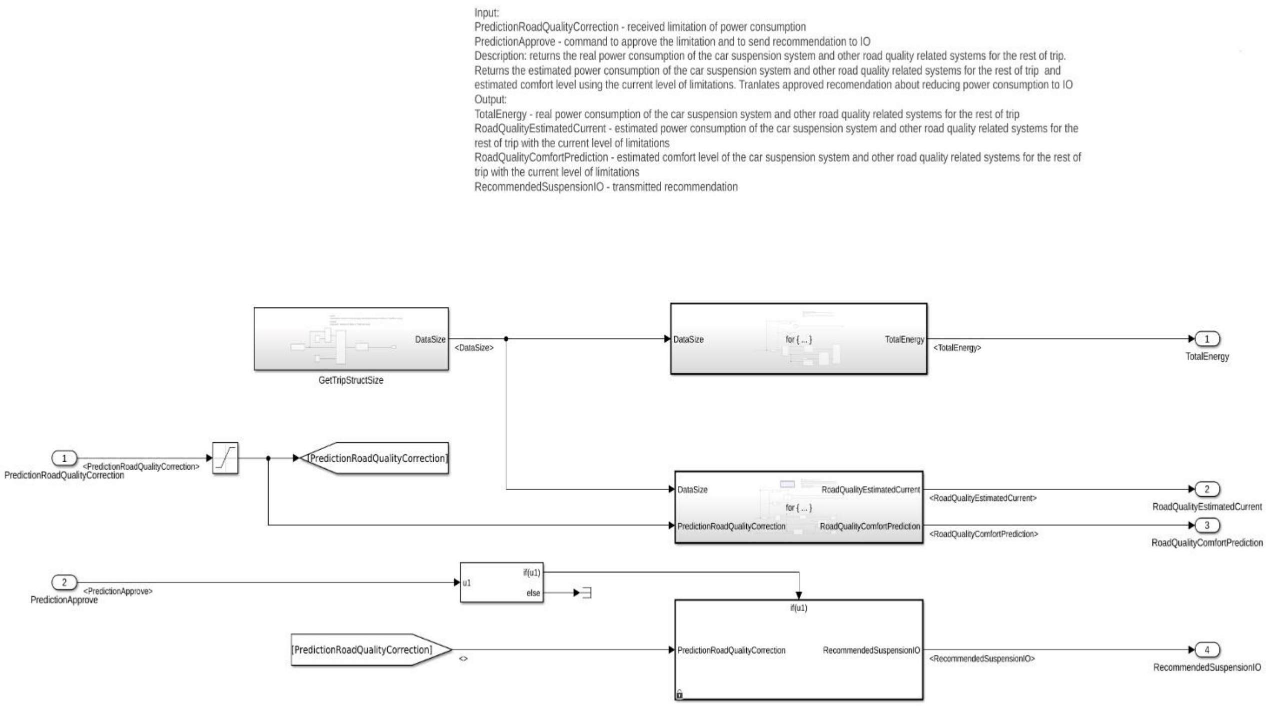

Fig. 3. Typical Structure of a Software Component with Two Separate Subcomponents

The Prediction SWC and DriverProfile SWC software components are shown in Figures 4 and 5, respectively.

The structure of the Prediction SWC includes two subcomponents. One performs the tasks of estimating costs for the entire route that the car still has to cover. The second subcomponent searches for the optimal energy consumption scenario according to system (2). The structure of the DriverProfile SWC includes logic for accruing and writing off "penalties" imposed on the driver for ignoring recommendations. Penalties are deducted for following these recommendations. The accrual and deduction model is hysteresis-based and is set by interpolation tables.

The mathematical model of DriverProfile SWC is shown below.

1. Penalty Accrual:

$$P_{t+1} = P_t + \alpha E_t$$

2. Penalty Deduction:

$$E_t = \frac{1}{2}\sum_{i=1}^{n}(r_i - a_i)^2$$

3. Backpropagation Update:

$$P_{t+1} = P_t - \beta R_t$$

$$\omega_{t+1} = \omega_t - \eta \frac{\partial E_t}{\partial \omega}$$

4. Hysteresis Effect:

$$P_{t+1} = \{\{P_t + \alpha E_t \;\; if \; E_t > \delta\} \; OR \; \{P_t - \beta R_t \; if \; E_t \le \delta\}\}$$

For driver from

$$D = \{d_1, d_2, \ldots, d_m\}$$

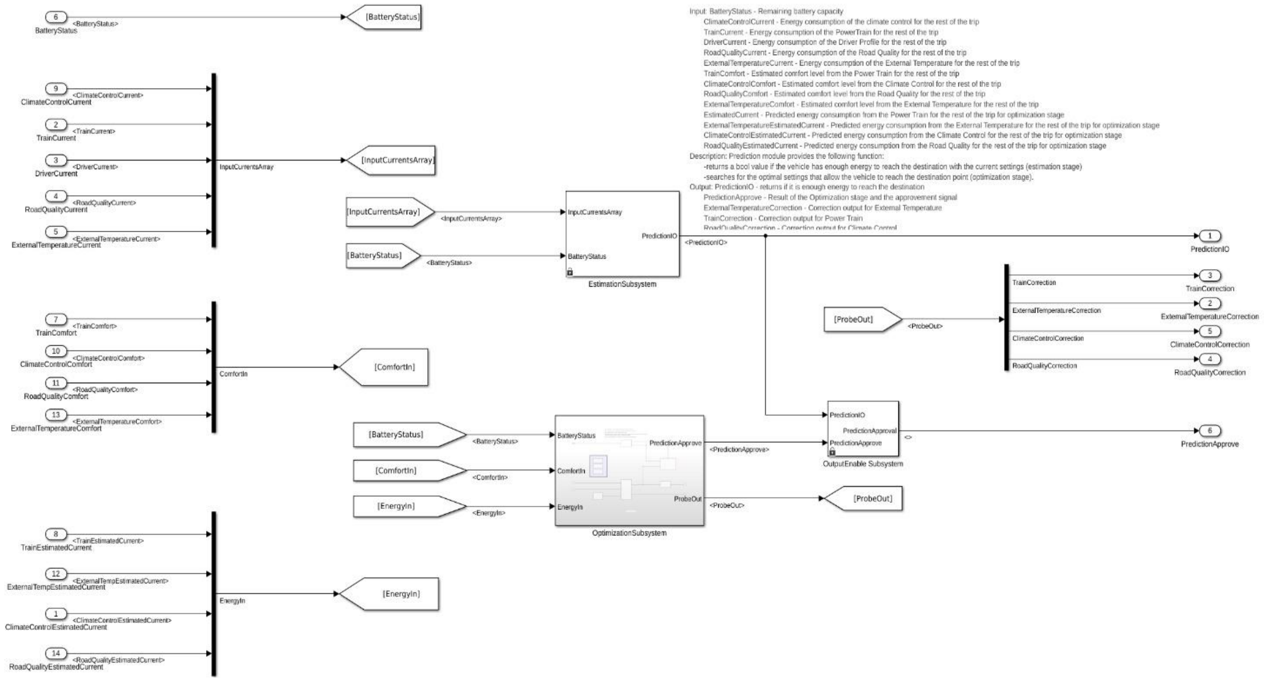

Fig. 4. Prediction SWC Software Component Model

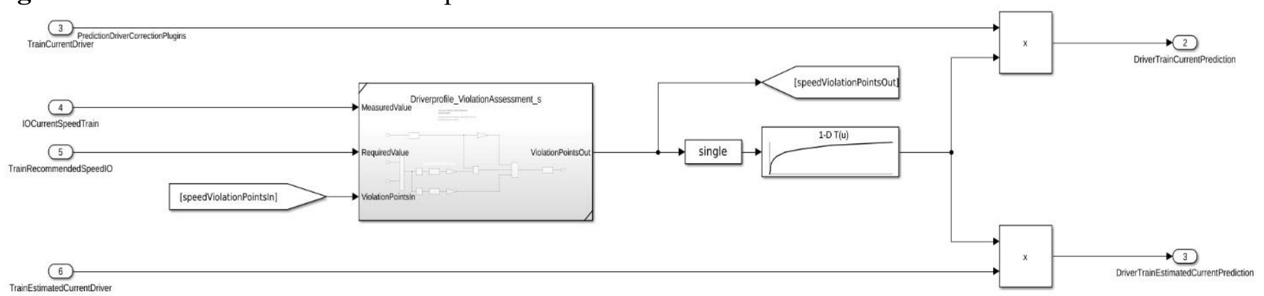

Fig. 5. DriverProfile SWC Software Component Model (the main subsystem only)

The mathematical model, which is part of the Powertrain SWC, cannot be disclosed due to NDA.

SETTING TRIGGERING TIME OF COMPONENTS WITH EACH OTHER

Time-sampling for all SWC components is crucial for ensuring synchronized and efficient system operation. Each SWC component triggers at specific intervals, which are recorded as time samples. These time samples help in understanding the interaction and behavior of components under different operating conditions.

In the histogram (Figure 6), we observe the distribution of triggering times for various components:

- Prediction SWC triggers primarily between 10-13 milliseconds.
- Powertrain SWC triggers mainly between 15-18 milliseconds.
- DriverProfile SWC triggers mostly between 19-22 milliseconds.
- ClimatControl SWC triggers predominantly between 25-28 milliseconds.
- Suspension SWC triggers primarily between 30-33 milliseconds.
- ExternalTemperature SWC triggers mainly between 35-38 milliseconds.
- TirePressure SWC triggers mostly between 40-43 milliseconds.
- GoogleMap SWC triggers predominantly between 45-48 milliseconds.
- HMI SWC triggers primarily between 50-53 milliseconds.
- Modem SWC triggers mainly between 55-58 milliseconds.

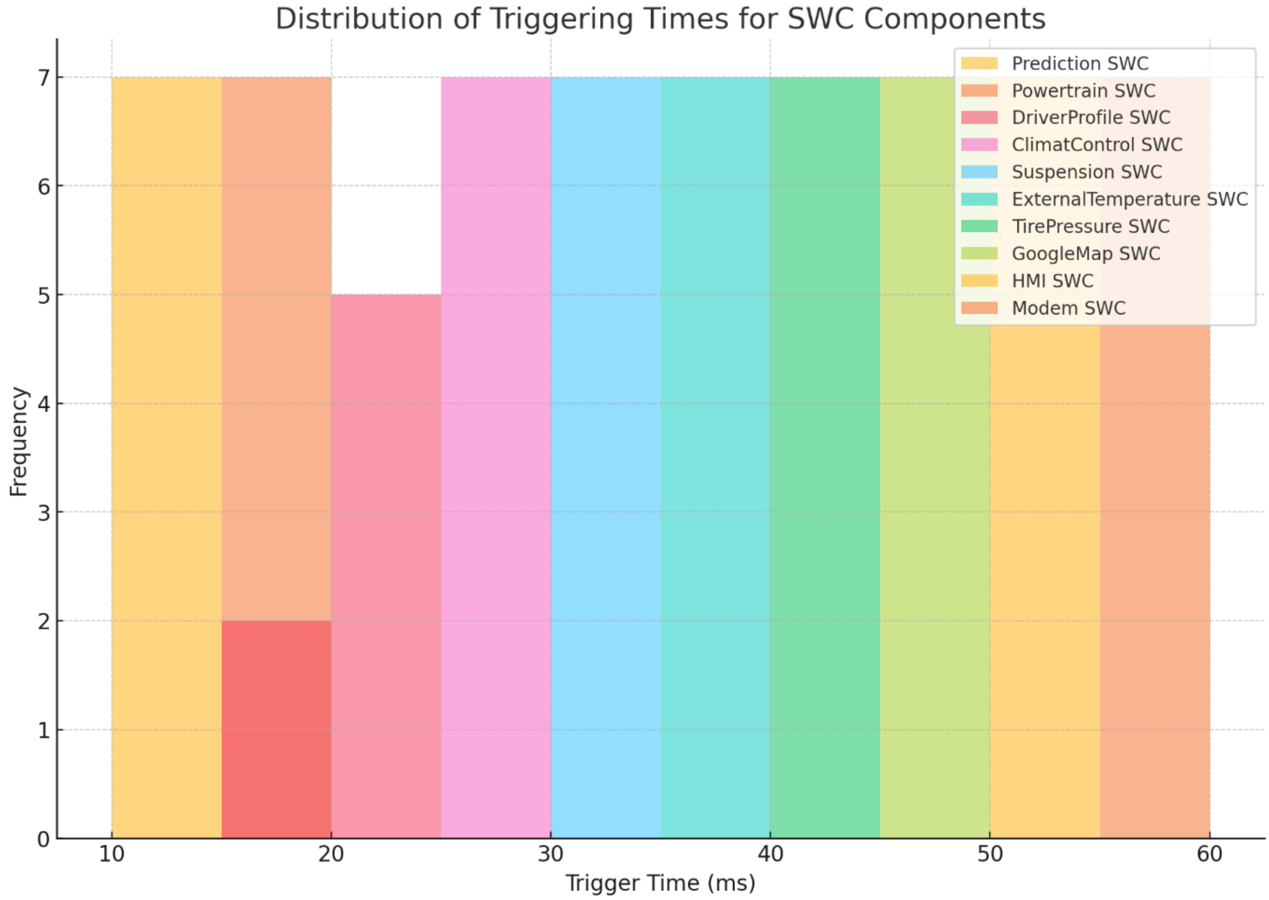

Fig. 6. Timesample histogram

This detailed time-sampling allows for precise coordination and optimization of component interactions, ensuring that the system operates smoothly and efficiently. By analyzing these time samples, we can identify potential areas for improvement in synchronization and performance, leading to a more robust and reliable system.

COMPONENTS FOR ENVIRONMENT VIRTUALIZATION AND STUB REALIZATION

The construction of these components significantly depends on the available software licenses. By basing solutions on Simulink, an alternative approach can be taken. Since this software section is more for demonstrating capabilities rather than being deployed as a product, certification is not required. Therefore, the use of third-party software is acceptable.

For external solutions such as GoogleMaps, steering wheel interfaces, Carla simulation, or practically any solution outside the MATLAB ecosystem, it is proposed to use the Instrument Control Toolbox, which includes a UDP toolkit. This type of task is effectively managed using sockets.

ACHIEVING REAL-TIME

Achieving real-time operation is challenging. Simulink does not inherently support real-time execution as the simulation time of the system can differ from physical time. Real-time performance can be achieved after compiling the software product and running it as an application.

TESTING AND INTEGRATION

According to the V-Model process, testing is inseparable from the development process. Building a testing process for ASIL-A according to ISO 26262-6 [15] includes five stages, conducted sequentially. Stages and completion criteria for this composition are provided in Table 1.

In accordance with ISO 26262-6 [15] recommendations, the processes are conducted using certified tools or subject to further verification. Therefore, in this work, it is proposed to use the verified Simulink Test toolkit with additional instruments such as Test Manager, Coverage, and Reports, to remain within a single software development ecosystem.

The completion criteria for the testing stages are proposed as indicated in Table 2.

The stages "Validation of composition on virtual target (ViL)" and "Testing on real ECU (ViL)" are not completed to date. Their impact on the product will be published additionally. Accordingly, integration as an ECU application is not the subject of the current publication.

Table 1. Testing stages of software composition

Stage	Metrics	Description
Static analysis of Simulink model	Diagnostic Coverage (DC), Single Point Fault Metric (SPFM)	Model analysis to identify defects and potential problems. Measuring code coverage and diagnostic efficiency for testing evaluation.
Model unit testing (SiL)	% of tests executed, Common Cause Fault Metric (CCFM)	Executing model-level tests based on a simulation environment. Measuring the percentage of successfully executed tests and the metric for partial failures.
Integration testing of components (HiL)	% of integrated components, Probabilistic Metric for random Hardware Failures	Testing interactions between different components at the hardware level. Evaluating the percentage of successfully integrated components and integration time.
Validation of composition on a virtual target (ViL)	DC, Failures In Time (FiT)	Testing the entire composition on a virtual platform. Measuring diagnostic efficiency and the number of failures per trillion hours.
Testing on a real ECU (ViL)	FiT, Latent Fault Metric (LFM)	Testing the composition on a real ECU to verify stability and identify defects. Measuring the number of failures per trillion hours and latent faults.

Table 2. Completion criteria for testing stages

Testing stage	Completion criteria
Static analysis of Simulink model (FMEDA)	Completion of the analysis when DC ≥ 90% and SPFM ≥ 99%. All identified critical defects must be corrected and confirmed by repeated analysis.
Model unit testing (SiL)	Completion of testing when % of successful tests ≥ 95% and CCFM ≤ 1 FIT. All critical defects must be corrected and confirmed by repeated tests.
Integration testing of components (HiL)	Completion of testing when % of integrated components ≥ 95% and PMHF ≤ 10 FIT. All identified critical defects must be corrected and confirmed by repeated tests.
Validation of composition on virtual target (ViL)	Completion of validation when DC ≥ 95% and FIT ≤ 10 per trillion hours. All critical defects must be corrected and confirmed by repeated tests.
Testing on real ECU (ViL)	Completion of validation when DC ≥ 95% and FIT ≤ 10 per trillion hours. All critical defects must be corrected and confirmed by repeated tests.

IMPLEMENTATION AND PUBLICATION OF RESULTS

The results of the work have been implemented as a Proof of Concept (PoC) and have been presented at events such as MathWorks Automotive Conference 2024 [5], featured at the N-iX company booth; a MathWorks webinar titled “Energy Consumption Prediction for Electric Vehicles” [6]; and described in N-iX company publications [7].

THE AMPERE POC

From Figure 7(a), the structure of the Ampere project is evident, which is built according to the concept described in the publication. In this Proof of Concept, the focus is on cooperation with the semi-realistic environment. To achieve this, Carla, GoogleMaps, Steering, and an Android Tablet (driver authorization, destination coordinates determination, communication of optimal trip data to the driver) were used. Communication protocols UDP and ProtoBAFs were employed for this purpose.

Figure 7(b) shows the application as seen by the user.

CONCLUSION

This study details the development of an adaptive software component to predict energy consumption for electric vehicles on planned routes. It emphasizes a modular and scalable architecture, incorporating mathematical models to manage consumption levels. Key data sources such as route information, vehicle condition, and driver behavior form a comprehensive state vector for optimizing energy use.

The implementation adheres to ISO 26262 and A-SPICE 3.1 standards, using a model-based approach with Simulink, and aligning with the V-Model for rigorous testing. The gradient search method adjusts consumption to minimize energy use while maximizing comfort, ensuring the vehicle completes its route efficiently.

Components for environment virtualization and stub realization were developed using Simulink and third-party software, managing external solutions through the Instrument Control Toolbox and UDP sockets. Achieving real-time execution involved compiling the software product and running it as an application.

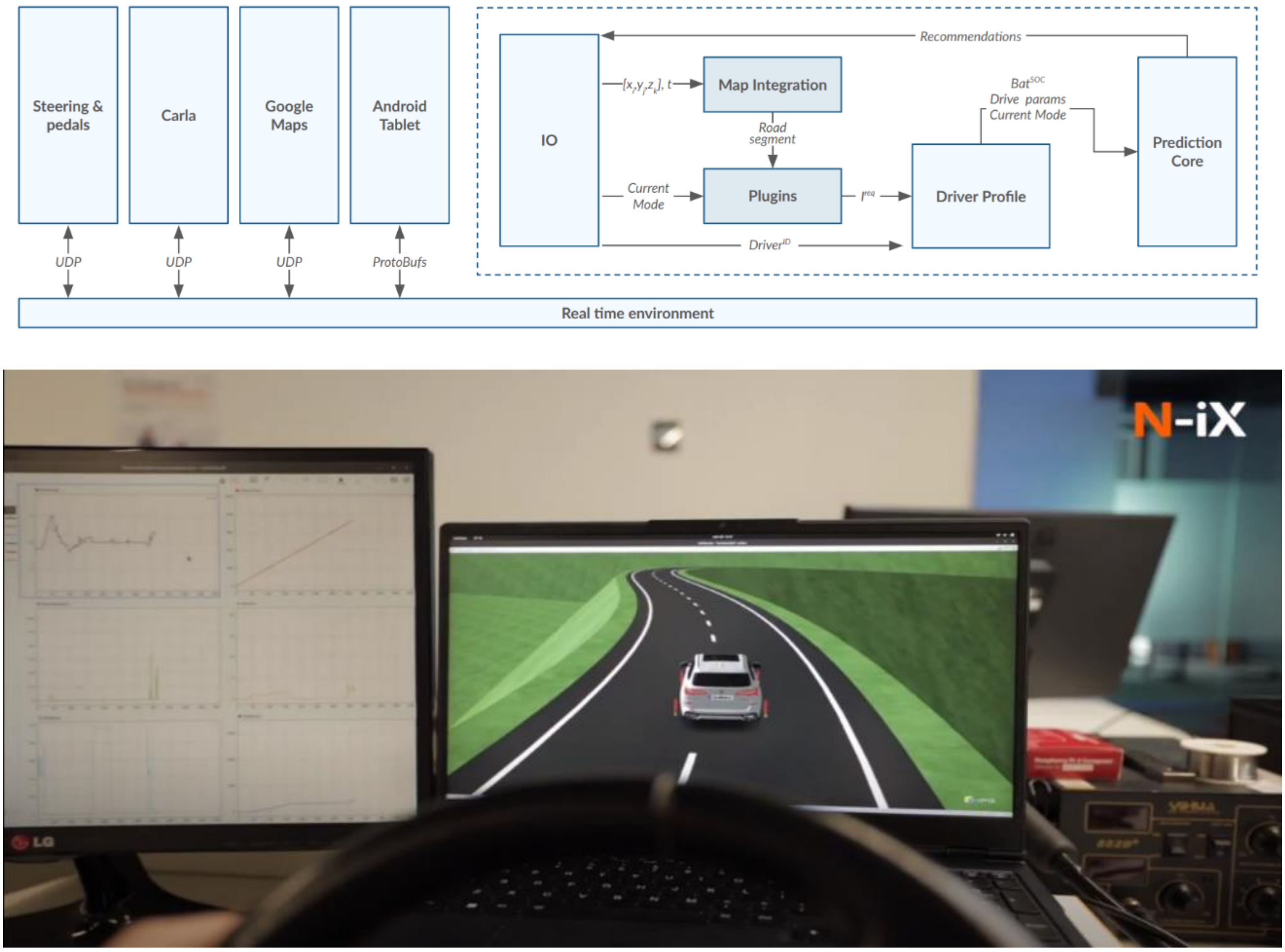

Fig. 7. The Ampere by N-iX. a - generalized view; b - user view

This Proof of Concept has been showcased at industry events, demonstrating the feasibility of advanced predictive models for energy management in electric vehicles. Future work will focus on refining predictive models, exploring machine learning for improved accuracy, and integrating real-time data from connected vehicle technologies to dynamically optimize energy consumption. These efforts aim to enhance the efficiency, reliability, and user-friendliness of energy management systems, supporting wider adoption and sustainability in the transportation sector.

By addressing these observations and incorporating the suggested revisions, the paper will improve in clarity, readability, and overall effectiveness.

REFERENCES

1. "Advantages of Electric Vehicles: Top Benefits Explained," EnergySage, accessed June 6, 2024, (2024). EnergySage.
2. "Experts Discuss Hurdles, Benefits of Electric Vehicles," Duke Today, accessed June 6, 2024, Duke Today.
3. "Electric Vehicles: Benefits, Challenges, and Potential Solutions for Widespread Adaptation," MDPI, accessed June 6, 2024, MDPI.
4. "Comprehensive Review of Electric Vehicle Technology and Its Impacts: Detailed Investigation of Charging Infrastructure, Power Management, and Control Techniques," MDPI, accessed June 6, 2024, MDPI.
5. MathWorks Automotive Conference 2024. Available at: https://www.mathworks.com/company/events/conferences/automotive-conference-stuttgart/2024.html
6. Webinar by MathWorks "Energy Consumption Prediction for Electric Vehicles." Available at: https://www.mathworks.com/company/events/webinars/upcoming/energy-consumption-prediction-for-electric-vehicles-4331190.html
7. Predictive Energy Management for EV. N-iX Publications. Available at: https://www.n-ix.com/predictive-energy-management-for-ev/

8. **Rombaut, E., Vanhaverbeke, L., & Coosemans, T.** (2023). Energy-Optimal Speed Control for Autonomous Electric Vehicles Up- and Downstream of a Signalized Intersection. *World Electric Vehicle Journal*, 14(2), 55. https://doi.org/10.3390/wevj14020055
9. **Martinez C. M., Hu X., Cao D., Velenis E., Gao B., Wellers M.** (2016). Energy management in plug-in hybrid electric vehicles: Recent progress and a connected vehicles perspective. *IEEE Transactions on Vehicular Technology*, Vol. 66, No. 6, 4534-4549.
10. **Hannan M. A., Azidin F. and Mohamed A.** "Hybrid electric vehicles and their challenges: A review". *Renewable and Sustainable Energy Reviews*, vol. 29, 135-150, 2
11. **Amjad S., Neelakrishnan S. and Rudramoorthy R.** (2010). "Review of design considerations and technological challenges for successful development and deployment of plug-in hybrid electric vehicles". *Renewable and Sustainable Energy Reviews*, Vol. 14, no. 3, 1104-1110.
12. **Sutton R. S., McAllester D., Singh S. and Mansour Y.** (1999). "Policy gradient methods for reinforcement learning with function approximation". *Advances in neural information processing systems*, Vol. 12.
13. MISRA. (2007). "Guidelines for the application of MISRA-C:2004 in the context of automatic code generation," MISRA AC GMG.
14. MathWorks. (2019). "Modeling Guidelines for Production Code Generation," MathWorks Automotive Advisory Board (MAAB).
15. International Organization for Standardization, "ISO 26262: Road Vehicles – Functional Safety," ISO 26262:2018.
16. International Organization for Standardization, "ISO 22050: Safety of Machinery – Physiological Effects of Whole-Body Vibration," ISO 22050:2007.

Адаптивний програмний компонент для прогнозування енергоспоживання електромобіля на запланованому маршруті

Дмитро Гуменний

Анотація. У даній роботі представлено розробку адаптивного програмного компонента для прогнозування енергоспоживання електромобілів на запланованих маршрутах. Він включає концептуальну основу, архітектурний дизайн і модульну композицію програмного забезпечення, вводячи математичні моделі для управління споживанням енергії з модельним підходом до проектування для точних прогнозів і оптимізації. Важливість джерел даних, таких як інформація про маршрут, стан транспортного засобу та поведінка водія, підкреслюється для створення комплексного вектора стану для оптимізації енергії.

Дотримуючись стандартів ISO 26262 і A-SPICE 3.1, реалізація використовує модельний підхід із Simulink і узгоджується з V-моделлю для суворої перевірки. Методологія детально розподіляє маршрути на сегменти та оптимізує споживання енергії для кожного сегменту, враховуючи стиль водіння та умови навколишнього середовища. Метод градієнтного пошуку регулює споживання енергії, щоб мінімізувати споживання, максимізуючи комфорт і гарантуючи завершення маршруту.

Це дослідження закладає основу для майбутніх досягнень у системах прогнозного керування енергією для електромобілів із потенційним застосуванням у реальному світі. Майбутня робота буде зосереджена на вдосконаленні прогнозних моделей, вивченні машинного навчання для підвищення точності та інтеграції даних у реальному часі з технологій підключених транспортних засобів для динамічної оптимізації.

Ключові слова. Електромобілі, передбачення енергоспоживання, модельний дизайн, система управління енергією, оптимізація маршруту, поведінка водіння, адаптивне програмне забезпечення, ISO 26262, A-SPICE 3.1, градієнтний пошук, Simulink, вектор стану автомобіля, модульна архітектура, прогнозні моделі, машинне навчання , інтеграція даних у реальному часі, підключені технології транспортних засобів, керування акумулятором, оптимізація комфорту.