ORQA: Modeling Energy and Quality of Service within AUTOSAR Models

Borjan Tchakaloff, Sébastien Saudrais, Jean-Philippe Babau

ORQA: Modeling Energy and Quality of Service within AUTOSAR Models

Borjan Tchakaloff [1,2]
b.tchakaloff@estaca.fr

Sébastien Saudrais [1]
s.saudrais@estaca.fr

Jean-Philippe Babau [2]
babau@univ-brest.fr

[1] Embedded Systems team
CERIE, ESTACA
Laval, France

[2] MOCS, CACS team
Lab-STICC, UEB, UBO
Brest, France

ABSTRACT

Electric vehicles embed a low amount of energy, so their devices need to be managed efficiently to optimize the vehicle autonomy. A vehicle management is achieved by the embedded systems, modeled following the AUTOSAR standard. AUTOSAR covers most of the automotive concerns, but it lacks energy consumption and user-oriented Quality of Service models. This paper presents ORQA, a framework to model and manage the electric vehicle devices through energy consumption and user-oriented Quality of Service. At design time, the architects choose and tune the actual vehicle device models through their power requirements and, if appropriate, quality levels. The generated implementation is then embedded in the existing AUTOSAR models. Thus, at run-time, the vehicle's system is able to evaluate the global consumption of a trip and to propose the user a specific driving strategy. The optional devices are managed throughout the trip, based on the driver preferences. ORQA is illustrated with a classic use-case: a work to home trip.

Categories and Subject Descriptors

D.2.11 [**Software Engineering**]: Domain-Specific Architectures; D.2.13 [**Software Engineering**]: Reusable Software—*domain engineering, reuse models*; J.7 [**Computers in Other Systems**]: Consumer Products—*electric vehicle*

General Terms

Design, Management

Keywords

AUTOSAR, model driven architecture, energy consumption, Quality of Service

1. INTRODUCTION

The Electric Vehicle (EV) has now reached an industrial maturity. Though several models are available, its energy capacity remains low, limiting its purpose to a day-to-day usage (about a hundred kilometers autonomy).

At the present time, most of the existing EVs provide no complex energy management: they only limit the vehicle speed when the battery is getting low without any concern about the driver's intentions or destination. The driver takes benefit of a *full service* while there is a certain amount of energy left and a *restricted service* otherwise. In *full service*, the driver is not restrained in any way. In the *restricted service*, the vehicle is limited to reduced speeds and some devices are restrained to keep the energy consumption at its lowest. This policy neglects the driver's preferences and is not optimal for a given trip. It would be necessary to anticipate the imposed reduced speed to reach a destination. Also, the driver may want to express preferences and priorities on devices usage. The challenge is then to provide an adaptable and acceptable solution between the two extremes.

To perform an efficient and adequate global energy management, a full control of configuration of all the consuming devices (respecting the driver's preferences) is required. This control is possible through the software embedded in the network of control units composing the vehicle information system. Each consuming device such as the lamps, the air-conditioning and so on is managed by a dedicated control unit. The control software has to be integrated respecting the AUTOSAR [1] (AUTomotive Open System ARchitecture) standard constraints.

The worldwide AUTOSAR consortium gathers, and is used by, automotive manufacturers and equipment makers in need for a common methodology [7]. It aims at easing re-usability of Embedded Systems (ES) and contributing to a common basis, thus allowing easier project management and content sharing. The AUTOSAR methodology is based on models and relies on the software components paradigm. It permits designers to split ES modeling on different levels, from a system view down to the implementation code. But AUTOSAR models are architecture oriented and does not offer extra-functional properties support.

The energy consumption is an extra-functional property and, as such, is not taken into account in AUTOSAR models. In order to estimate the vehicle consumption, the consumption knowledge of every consuming device is required. Estimating what the vehicle should consume in certain conditions (the route type, the maximum vehicle velocity, the turned-on devices, etc.) allows to optimize the driving strategy. As the driver wants to reach his destination, the system should be able to offer at least one viable solution.

In this paper, we propose ORQA (mOdeling eneRgy and Quality of service in Autosar), a framework to model the vehicle devices consumption and user-oriented Quality of Service. They are used to fulfill the driver's expectation: to reach a destination using as much as possible all the devices. To assure the driver of his success, the vehicle energy consumption has to be predicted for the available routes and the best route proposed to the driver. These predictions rely on the energy consumption knowledge of the whole vehicle, that is on both compulsory devices (the engine, the lamps, ...) and non-critical devices (the air-conditioning system (A/C) and the heater in Figure 1). The framework presented in this paper takes into account both types of devices. Furthermore, an online control of devices usage has to be performed to ensure the strategy realization with regards to the driver preferences. This results in an embedded energy manager that manages the vehicle devices taking into account their energy consumption and the driver's goal.

The paper is organized as follows. We first provide an overview of the AUTOSAR standard in Section 2. Section 3 formulates the context and the problem the paper deals with. Section 4 then gives an overview of the ORQA process. The two following sections are dedicated to the specific models (Section 5) and to the architecture (Section 6). Then, in Section 7, we illustrate the use of ORQA. We discuss related works in Section 8 and finally conclude the paper.

2. AUTOSAR

AUTOSAR is a standard of automotive electrical and electronic engineering developed and used by car manufacturers and equipment makers all around the world. The software architecture, the control units hardware and configuration, the different network topologies are defined in a meta-model that supports the software development process from the design phase to the integration phase.

2.1 Concepts

Different concepts are used in AUTOSAR, from the software to the hardware point-of-view. The ORQA approach is based on the software component (SWC) concept.

The concept of SWC is the first-order element of an AUTOSAR system. SWCs communicate and interact through a virtual functional bus representing the future buses. SWCs are then mapped to distributed Electronic Control Units (ECUs), which will host them. As the result of a layered architecture, they can be transferred to other platforms without detracting from the individual functions. SWCs communicate using ports through their interfaces. A port is composed of data elements containing the exchanged messages. In AUTOSAR, SWCs can be either atomic or composite. A composition of several SWCs is called a composite and uses delegation ports. The internal behavior decomposes atomic SWC into runnable entities which represent the implementation, typically in C code. An atomic component could be functional or driver. A driver represents a physical device and the component can either be a sensor or an actuator.

2.2 Methodology

According to the AUTOSAR approach, the development process of a system is divided into six steps. The first step is to define the set of SWCs constituting the user software applications and realizing the desired functionalities. The SWCs are designed by abstracting the implementation concerns. SWCs are defined during the second step without consideration of the underlying hardware they will later run on. The communication between the components is then either an intra-ECU or an inter-ECU communication and is routed via the virtual bus. Steps 3, 4 and 5 consist of the mapping of SWCs to available ECUs. This phase requires some information about ECU (step 3) and system constraints (step 4). The methodology's last step consists of the development and integration of each ECU.

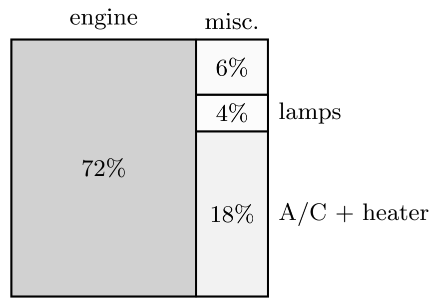


Figure 1: Mean Electric Vehicle devices consumption distribution in an typical urban environment

3. PROBLEM FORMULATION

The paper is based on the classical use-case *a driver wants to reach a destination point as soon as possible, using as much as possible the devices, saving as much as possible the battery.* To solve this problem, we first have to be able to evaluate a trip duration and the corresponding energy consumption for a certain vehicle usage. Then, we need to determine how to minimize duration and how to save energy. All these information are necessary to define optimization criteria in order to propose an adequate driving strategy.

We detail now how the duration and the energy consumption are computed according to the characteristics of the route and driving goals.

The trip duration T_{total} is basically related to the distance and to the velocity ($duration = distance \div velocity$). In our case, the distance is assumed to be constant (i.e. no change of route during the trip) but not the velocity so the duration is not fixed. The energy consumption E_{total} is the sum of all the devices consumption during the trip. As the energy is the product of power times duration, we state that:

$$E_{total} = P_{total} \times T_{total} \quad (1)$$

with E_{total} expressed in joules (J), P_{total} in watts (W) and T_{total} in seconds (s). The total required power P_{total} is divided in two parts: the mandatory (required) and the optional devices power requirements (respectively $P_{req.dev.}$ and $P_{opt.dev.}$). Devices are said to be mandatory if they are required to operate the system, they are otherwise said optional. The engine, the lighting system, the embedded systems and other security-related devices are mandatory for an electric vehicle. The air-conditioning system, the heater, the entertainment system and such are the optional devices of an EV. As stated in Figure 1, the most consuming device is the engine, which propels the vehicle. This figure presents the mean devices consumption distribution of an

EV driving the CADC-Urban[1] cycle which represents a typical urban environment in Europe[2]. We detail now how the most consuming devices of an EV are modeled based on mechanical and thermodynamical laws found in literature [4, 6, 8, 11, 15].

When traveling at a certain speed, the vehicle is subject to specific resistant forces: the aerodynamic force (2), the rolling (3) and the climbing resistive forces [11]. To simplify physical world modeling aspects, out of scope of this paper, mechanical losses are neglected and only longitudinal forces are considered. Moreover, the road is considered flat and with no facing wind. Because the road slope is null, the climbing resistive force is null and is therefore ignored. Depending on these assumptions, the different forces are:

$$F_{aero} = 1/2 \times \rho \times c_x \times A \times v^2 \tag{2}$$

$$F_{rr} = m \times g \times f_{rr} \tag{3}$$

where ρ is the air density (kg·m^{-3}), c_x the air penetration coefficient, A the active area of the vehicle (m^2), v the vehicle velocity (m·s^{-1}), m the vehicle mass (kg), g the gravity acceleration (m·s^{-2}), and f_{rr} the rolling resistive coefficient. The forces unit is the newton (N), equivalent to kg·m·s^{-2}. The propelling force F_{prop} is the resulting force from the vehicle's drive-chain. So the vehicle dynamics equation is:

$$a \times m = F_{prop} - F_{aero} - F_{rr} \tag{4}$$

where a is the vehicle linear acceleration (m·s^{-2}).

The engine drives the wheels by the mean of the transmission axle with a specific efficiency η_T. The choice of a single-gear or a multigear transmission depends on the motor characteristics, the two types can be design to offer equivalent acceleration and gradeability [6]. To simplify our modeling, we will assume the use of a single-gear transmission defined by a multiplier K_T. The engine delivered power $P_{delivered}$ is then computed with the following equation:

$$P_{delivered} = \frac{F_{prop} \times v}{\eta_T \times K_T} \tag{5}$$

The powers unit is the watt (W), equivalent to N·m·s^{-1}. Using the former equations (2) to (5), $P_{delivered}$ is given by:

$$P_{delivered} = \frac{\rho \cdot c_x \cdot A}{2 \cdot \eta_T \cdot K_T} \cdot v^3 + \frac{m}{\eta_T \cdot K_T} \cdot a \cdot v + \frac{m \cdot g \cdot f_{rr}}{\eta_T \cdot K_T} \cdot v \tag{6}$$

The electric engine has different efficiency whether it propels the vehicle (*motor mode*) or whether it recovers energy by regenerative braking (*generator mode*):

$$P_{engine} = \begin{cases} P_{delivered} \div \eta_{motor} & \text{in } \textit{motor mode} \\ P_{delivered} \times \eta_{generator} & \text{in } \textit{generator mode} \end{cases} \tag{7}$$

where η_{motor} and $\eta_{generator}$ are the engine efficiencies.

It is important to notice that regenerative braking allows to get back energy from the vehicle kinetic energy through the electric engine [11]. In *generator mode*, P_{engine} is negative while in *motor mode* it is positive. Regenerative braking is the most useful in an urban environment, where the driver has to accelerate and brake every now and then. In typical urban areas, up to 25% of the energy consumed to propel the vehicle can be recovered owing to the regenerative braking [8, 15].

The non-engine devices power requirements are specified by the manufacturers. They can be static, as for the lighting system (8), or parametrized by one or several variables in the same way that the engine does.

$$P_{lamps} = \begin{cases} 0\text{W} & \text{no lamps} \\ 50\text{W} & \text{day-time lamps} \\ 200\text{W} & \text{low-beam lamps} \\ 250\text{W} & \text{high-beam lamps} \end{cases} \tag{8}$$

The vehicle compartment temperature is influenced by several disturbances. The most important ones are the sun radiation ((in)directly heating the vehicle), the vehicle velocity (changing the convection on the outer side of the vehicle) and the outer temperature. Climate Control units gathering both the air-conditioning and the heater systems are standard in new vehicles. We propose a simplified equation to compute the required power P_{CC} of such a unit:

$$\begin{aligned} P_{CC} = \;& C_1 \cdot (T_{out} - T_{in}) + C_2 \cdot (T_{obj} - T_{in}) \\ & + C_3 \cdot v + P_{sun} + C_4 \end{aligned} \tag{9}$$

where the constants C_i are to be defined, T_{in}, T_{out} and T_{obj} are respectively the inside, the outside and the objective temperatures (°C), and P_{sun} is the power retrieved from the sun (W).

The entertainment system regroups basic functions (the auto-radio, the electric plugs) and amusement functions (the passengers video screens). These devices are not grouped so they can be managed independently.

Some devices powers are aggregated to a constant because they are not optional, their power do not fluctuate much and/or their power is negligible in front of the other ones. Thus, the ES (about 50W at full load), the indicator lamps and the dashboard (about 50W), the windshield wipers (up to 50W) powers are gathered into P_{misc}, tuned at the design phase according to the installed devices worst-case energy consumption.

The vehicle characteristics are either constants (ρ, c_x, A, g, η_T, K_T) or approximated to constants (f_{rr}, m). Then, the engine delivered power can be simplified to a function of the vehicle acceleration and velocity:

$$P_{delivered} = C_1 \cdot v^3 + C_2 \cdot a \cdot v + C_3 \cdot v \tag{10}$$

The overall vehicle power P_{total} is given by:

$$P_{total} = P_{req.dev.} + P_{opt.dev.} \tag{11}$$

where the required and optional devices powers are:

$$P_{req.dev.} = P_{engine} + P_{lamps} + P_{misc} \tag{12}$$

$$P_{opt.dev.} = P_{CC} + P_{radio} + P_{plugs} + P_{video} \tag{13}$$

With the devices powers defined, one can compute the energy consumption of a vehicle based on its environment. The total duration can be computed from the vehicle velocity evolution during the trip. Using these metrics, we define three objectives for the driver: (a) the minimization of T_{total} the trip duration, (b) the minimization of E_{total} the trip energy consumption and (c) a trade-off between the trip duration and its consumption. In every case, the optional devices usage is to be optimized. ORQA is now introduced to help manage the consuming devices with regards to the driver's objective and expectations.

[1]The *Common* ARTEMIS *Driving Cycles* (CADC) are the new European driving cycles references.

[2]Results based on data mostly gathered from [4] and applied to an EV simulation in an urban environment.

4. OVERVIEW

The Orqa process is realized in two phases. The first phase is performed at design stage. It is the creation of the energy and QoS model. The designer defines the power requirements and (when applicable) the quality levels of the vehicle devices using specific models. To help the designer, the framework offers a library of pre-defined models for the engine and each devices, the lamps, the Climate Control unit, the entertainment system devices and smaller consumers gathered in one constant power requirement.

The second phase is performed at run-time and concerns devices usage. It operates as follow:

1. The user chooses an objective and a consumption policy, they will later be used to compute and select a strategy. In the home trip example, the user chooses his target destination point and selects a consumption policy. The departure point is defined as the current vehicle position.

2. The strategies are retrieved or generated from the system. There are two sub-steps in this particular example. Several routes can exist between the departure point and the destination. A GPS unit typically offers three types of routes: the fastest, the shortest and a trade-off between duration and distance. At most one of each route type is retrieved (sub-step 1). Velocity coefficients (reducing maximal allowed speeds to lower speeds) are applied to the set of available routes, creating a matrix of route trips to evaluate (sub-step 2).

3. The strategies are evaluated, that is the required energy are computed for each one of them. Energy consumption of mandatory devices and functions, and duration of a route with a specific velocity coefficient are computed for each route trip.

4. The best matching strategy is selected. First, the different strategies are filtered by a maximum allowed consumption, depending on the available energy and on a safety margin. Then, scores are assigned to the remaining strategies depending on the user objective and the best match is selected. The user is then informed of the chosen strategy.

5. An embedded energy manager (a special component communicating with the device drivers) is used to control the devices online. To support constraint systems (such as Autosar ones), architecture is static and cannot be modified at run-time. So the manager influences the devices behavior by the mean of brokers attached to their driving components.

The two phases process is based on a set of specific energy- and quality-oriented models.

5. MODELS

Orqa proposes specific models focusing on three concerns: the powers required by the devices to operate (*the devices requirements*), the user-comfort *quality levels* of the optional devices and *the user preferences*.

First, we assume that a device is characterized by a set of *operating states*. Each operating state corresponds to a specific *power function* (its consumption law). As a device has a finite number of operating states, it is modeled with a state

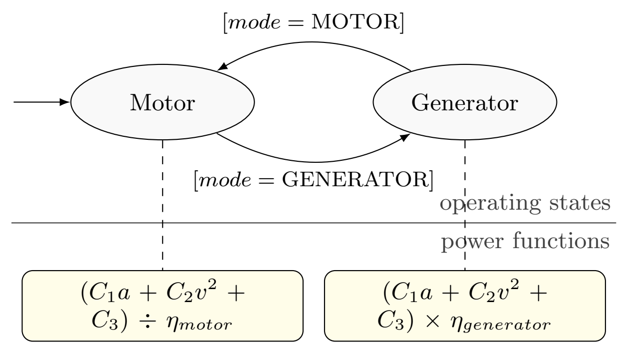


Figure 2: The engine operating states and their power functions

machine. Guarded transitions between operating states rely on environmental data values. Power functions associated to operating states allow Orqa to estimate the devices consumption when evaluating the different strategies.

The optional devices can be controlled automatically, the system does not need them to operate fully. Also, as the operating states of a device usually offer different QoS to the driver, the operating states of an optional device are qualified so as to allow quality comparisons between them.

Finally, user preferences allow the driver to inform the system on his preferences on the different optional devices.

The driver's comfort expectation is satisfied whenever possible but his trip objective is always favored to the optional devices.

5.1 Power functions

A power function corresponds to either a constant or to a function parametrized by environmental data. The energy consumption of a device for a specific operating state is computed with the multiplication of the bound power function and the usage duration.

For instance, the engine is a mandatory device which can either consume or restore energy to the battery depending on its mode. A suitable model is represented in Figure 2, with two operating states corresponding to the engine modes. The engine power functions are composite and depend on two environmental data (the vehicle velocity and acceleration) and on constants specific to the vehicle (Equation 10).

5.2 Quality levels and user preferences

The quality levels are attached to the optional devices operating states only. They are subjective data chosen by the designers and linked to operating states at the design phase. A higher value means a higher comfort for the passengers. As Orqa represents them as percentages, 100% is the best Quality of Service possible. Giving a quality level of zero means *no quality applicable*, as when a device is shutdown. The quality levels are harmonized by using percentages because it allows the system to compare different device usages by their QoS offered to the users. Two operating states can have equal levels, meaning they offer the same QoS. The designers set the quality levels once and for all when designing the system.

The user preferences are basically the optional devices ordering set by the driver and stored in the system. He ranks these devices by assigning a percentage to each of them. The sum of all of these values must equal 100%. The driver de-

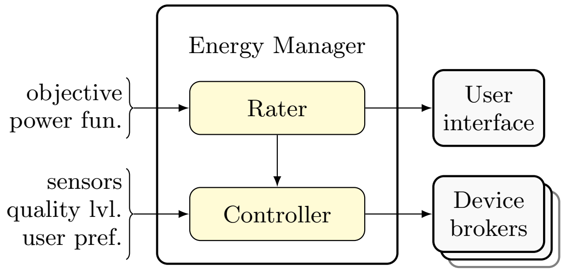


Figure 3: The Energy Manager component

fines which devices should be favored and in what proportion their QoS is important (the higher the percentage the more important the device). As an example, a driver preferring a better temperature control than listening to the auto-radio could set the Climate Control unit preference to 80% and the radio one to 20% (considering only these two devices as optional).

The system is then able to prioritize the optional devices with an overall QoS based on the driver's desires by combining quality levels and user preferences. With this information, the system is able to optimize the consumption, following user preferences. We describe now the ORQA architecture by presenting its components and its behavior.

6. ARCHITECTURE

The ORQA architecture is presented in Figure 3. The AUTOSAR system model is enhanced with two kinds of components: an Energy Manager and a set of device brokers. The Energy Manager component is the main component of ORQA. It requires environmental data from sensor components (like the in-vehicle temperature and the daytime sensor) or computed by other components (like the acceleration demand from the driver). All the required data already exists in the system and is available through the system buses.

The Energy Manager component is composed of the Rater and the Controller components. The Rater is in charge of selecting the best matching strategy with regards to the user objective. A strategy is the evolution over time of the mandatory devices operating states. The chosen strategy is then delivered to the user in an informative way out of scope of this paper. The Controller makes the strategy happen by managing the consuming devices. The management is realized through device brokers that are specially created to modify the devices controls.

We detail now the Rater and the Controller operations.

6.1 The Rater component

The Rater regroups the usage phase 2, 3 and 4 defined in the framework overview (Section 4). It selects the best matching strategy by evaluating the different possibilities weighted with the user's objective. There are three different objectives: (a) the minimization of the trip duration, (b) the minimization of the trip energy consumption and (c) a trade-off between the trip duration and its consumption.

Following our example, the Rater evaluates the available routes and selects the best matching one regarding the user's selected objective. We assimilate strategies with trips for the Electric Vehicles. Several trips are generated from each available route by applying velocity coefficients to it. The velocity coefficients are coefficients applying to the nominal (maximal allowed) road speeds. Some example coefficients

Table 1: Default velocity coefficients and corresponding maximal velocities

	Velocity coefficient (K_v)			
Road type	100%	90%	80%	60%
Highway	130km/h	117km/h	104km/h	78km/h
Extra-urban	90km/h	81km/h	72km/h	54km/h
Urban	50km/h	45km/h	40km/h	30km/h

are shown in Table 1 along with typical road types. A coefficient of 100% is for nominal velocities, 60% is most likely to be used in emergency mode and other coefficients are chosen as alternate steps. There is no limit to the number of different coefficients set in the framework and each coefficient will be applied to each available route.

A route is a set of steps defined by their initial, cruise and final velocities and distance. A step cruise velocity is the maximal allowed velocity on that very step. Nominal trajectory is computed for each step based on the vehicle characteristics set at design phase. The result of this computation is the evolution of the vehicle velocity and engine required power against elapsed time. So each step duration is known, and by applying the basic energy relation ($energy = power \times duration$) a step consumed energy is easily calculated. The overall route duration and energy consumption are the sums of the steps respective results.

The route selection rests on the route trips feasibility and scores. A route trip r is said to be feasible iff its estimated consumption E_r is less than the available energy left in the battery $E_{battery}$, considering a required minimum energy Δ_E that has to be kept. Δ_E is a safety margin against the route modeling precision and offers some spare energy that may be required at the arrival (e.g. to park the car).

$$\begin{cases} E_r \leq E_{battery} - \Delta_E & \text{feasible route trip} \\ E_r > E_{battery} - \Delta_E & \text{discarded route trip} \end{cases} \tag{14}$$

The remaining (feasible) route trips are then evaluated with scores. The higher the score, the better the trip adequacy to the consumption policy. The two objective parameters (trip duration and energy consumption) are weighted by w_T and w_E to build the score. w_T and w_E are integers between 0 and 100 and must sum up to 100. They represent the significance of the parameters given a consumption policy from which they are deduced: (a) the minimization of the trip duration sets w_T higher than w_E ($w_T \gg w_E$), (b) the minimization of the energy consumption does the contrary ($w_T \ll w_E$) and (c) the trade-off sets them equal ($w_T = w_E$). The weights possible values are defined at the design phase, example values are shown in Table 2. The score value S of a route r is given by:

$$S = \left(1 - \frac{T_r}{T^{max}}\right) \cdot w_T + \left(1 - \frac{E_r}{E^{max}}\right) \cdot w_E \tag{15}$$

where T_r, E_r are the route trip duration and energy consumption and T^{max}, E^{max} are the maximal values found while evaluating the available routes.

Note that though the trip duration is always positive ($T_r \geq 0$), the energy consumption *could* be negative. For example, if the trip is to drive down a hill, it is possible than more energy is harvested by the regenerative braking than consumed by the vehicle. The higher score related route trip is then selected by the Rater as the best match.

Table 2: Look-up table of the score weights from the selected consumption policy (example values)

Consumption policy	Weights (significance) duration: w_T	energy: w_E
min(duration)	80	20
min(consumption)	20	80
trade-off	50	50

6.2 The Controller component

The Controller manages the devices to implement the strategy chosen by the Rater. The chosen strategy contains the mandatory devices operating states evolution along the trip duration. By definition, the optional devices are not required. So their management is not included in the strategy computation but is ensured online by the Controller. The Controller implements the strategy by forcing the consuming devices to operate at certain states by the mean of device brokers. The brokers are components with a basic bypassing rule set by the Controller.

In this paper example, the selected driving strategy has been evaluated to require a certain amount of energy E_r. The remaining energy in the battery is dedicated for the optional devices usage:

$$E^{dedi.}_{opt.dev.} = E_{battery} - E_r - \Delta_E \tag{16}$$

Because r is a feasible route, its consumption is lower than the energy left plus the safety amount of energy (Equation 14), thus $E^{dedi.}_{opt.dev.} \geq 0$. If equal to 0, then first the driver is warned that no optional device should be used to complete the trip. Second, the optional devices brokers bypass the device drivers to operate them at their minimal consuming operating state (no matter their quality level). If strictly greater than 0, the devices *may* be used at a higher consuming operating state. We now explain the Controller reasoning in this case.

The total energy required to operate the optional devices for the (remaining) trip duration T_r is given by:

$$\begin{aligned} E_{opt.dev.} &= E_{CC} + E_{radio} + E_{plugs} + E_{video} \\ &= T_r \times (P_{CC} + P_{radio} + P_{plugs} + P_{video}) \end{aligned} \tag{17}$$

Periodically, the Controller computes the energy required to operate the devices at their different operating states for the estimated remaining duration. Then the maximum required energy of each devices are summed. If the sum is lower than the dedicated energy, the Controller sets its brokers to let the control flows intact and informs the driver that there is no restriction over the optional devices. Otherwise the devices operating states are ordered by their combined QoS. It represents the offered QoS regarding a device operating state quality level and a device user preference. A combined QoS is given by *quality level* $\times$ *user preference*. If two items have the same combined QoS, the precedence is given to the one bound to the device the user favors (i.e. the device with a higher user preference percentage). And if the two bound devices have the same user preference value, the item favored is the one which requires the least energy. The selection of the optional devices operating states then begins.

1. The optional devices are assigned their least consuming operating state. $E_{opt.dev.}$ is the sum of their computed consumption.
2. The first ranked item is evaluated against the allocated energy. If its additional consumption is greater than $E^{dedi}_{opt.dev.} - E_{opt.dev.}$, the item is discarded. Go to step 3. If not, the corresponding device is assigned the item operating state. $E_{opt.dev.}$ is updated with the device new operating state consumption. Any other items referring to this device are dropped.
3. If there are more items, loop back to step 2. Otherwise the optional devices have been set with their best possible operating state with considerations for both Quality of Service and user preferences.

7. USE-CASE

In this section, the use of ORQA is illustrated with a classic use-case. First, we present the used setup and how is implemented the framework. Second, the use-case follows the general framework process as presented in Section 4, with its modeling phase at design time and its usage phase at run-time. We finally compare the obtained result to those obtained for the other consumption policies.

7.1 Setup

The framework's specific meta-model of power requirements and quality levels has been designed with EMF, the Eclipse Modeling Framework [16] .

The link with the AUTOSAR tool-chain (the Vector AUTOSAR tool-suite in our case) is performed by model transformations from our EMF meta-model to an AUTOSAR EMF meta-model provided by the AUTOSAR Tool Platform (Artop) [10]. Artop is a special AUTOSAR user group of interest in AUTOSAR compliant tools. It provides the Artop software, based on the EMF technology, offering a common base for AUTOSAR development tools.

Table 3: Characteristics of our use-case Electric Vehicle and simulation properties

Vehicle characteristic		Value
m	mass (kg)	1200
A	frontal area (m^2)	2.75
c_x	air penetration coefficient	0.3
f_{rr}	rolling resistive coefficient	0.008
	engine rated power (kW)	50
	battery rated energy (kWh)	20
η_{motor}	motor efficiency	0.95
$\eta_{generator}$	generator efficiency	0.75
η_T	transmission efficiency	0.925
K_T	transmission multiplier	1
Simulation property		**Value**
T_{out}	outer temperature (°C)	30
T_{obj}	objective temperature (°C)	20
ρ	air density (kg/m^3)	1.204
g	gravity acceleration (m/s^2)	9.81
P_{misc}	miscellaneous devices power (W)	1500
$E_{battery}$	battery available energy (kWh)	2
Δ_E	required minimum energy (kWh)	0.1

Table 4: Pre-evaluated power functions of the engine

Operating state	Power function
Motor	$0.565 \times v^3 + 1593 \times a \cdot v + 125$
Generator	$0.402 \times v^3 + 1135 \times a \cdot v + 89$

The chosen use-case is a typical working-day trip in a mid-sized European city (Rennes[3], France). The user lives in a residential suburb in the northern area and drives from 12 to 20 kilometers to reach his office in the south-east zone. He has to cross a mix of urban- and freeway-driving environments, which proportions vary depending on the route taken. We assume that the driver uses an urban Electric Vehicle, like the Bolloré Bluecar, the Mitsubishi i-MiEV or the Renault Zoe. The characteristics of both the EV and the simulation are detailed in Table 3. The simulated vehicle's longitudinal acceleration a is set to a constant, either 0m/s^2 or $\pm 2\text{m/s}^2$. The required minimum energy Δ_E is set to 5% of the battery capacity (0.1kWh). We now use the ORQA process described in the Section 4.

7.2 Modeling phase

The vehicle energy consumers are modeled at the design step. The engine modeling has already been illustrated in Figure 2 (Section 5.1), Table 4 contains the two engine pre-evaluated power functions.

The lighting system has static power requirements as the lamps are assumed to have constant powers. This system has four operating states depending on the lights switch position: no lamps (*off*), day-time lamps (*day-time*), low-beam lamps (*low-beam*) and high-beam lamps (*high-beam*). The required powers bound to the operating states are the followings: 0W for the *off* operating state, 50W for the *day-time* state, 200W for the *low-beam* one and 250W for the *high-beam* operating state.

The Climate Control unit offers the driver to define an objective temperature for the compartment. Bringing this desired temperature closer to the actual compartment temperature reduces the load on the CC unit, and so does its required power. Hence, two modes are defined for the CC unit: the auto mode and the eco mode. The auto mode fulfills the user demand (a specific objective temperature) whereas the eco mode uses a reduced temperature difference (the objective temperature is brought closer to the current temperature). The two modes obviously offer different Qualities of Service. As the auto mode respects the user directive, its QoS is maximal. The eco mode is arbitrarily set to offer half the QoS of the auto mode.

To simplify the CC power functions, we rely on data gathered from a campaign evaluating the average increase in consumption due to air-conditioning [2]. A fitting function is then made from power drawn of a mixed driving-cycle depending on objective and outer temperatures. This function parametrized by the temperature difference is used as the power function of the auto mode. The eco mode has an equivalent tuned function which reduces the difference between external and objective temperature by, for instance, 15%. Note that the gathered data offered a range of summer temperatures. The resulting power functions are only valid in this case. The whole model (operating states, power functions and quality levels) of the Climate Control unit is detailed in Table 5.

[3]The Rennes metropolitan area hosted 400'000 inhabitants in 2010.

Table 5: Operating states, power functions and quality levels of the Climate Control unit

Operating state	Power function	Quality level
Off	0	0%
Eco	$128(T_{out} - T_{obj}) + 116$	50%
Auto	$151(T_{out} - T_{obj}) + 116$	100%

Table 6: Operating states, power functions and quality levels of the entertainment system devices

Operating state		Power function	Quality level
Radio	Off	5	0%
	On	50	100%
Electric plugs	Off	0	0%
	On	25	100%
Video	Off	0	0%
	On	50	100%

The entertainment system regroups several optional devices: the auto-radio, the electric plugs and the passenger video screens. As stated in Section 3, all these devices are modeled individually (Table 6). So the Controller component will be able to manage them independently.

Also, in order to offer the best QoS possible to the driver, the framework requires the optional devices to be ordered. We set the user preferences as the following: 40% for the Climate Control unit, 30% for the auto-radio, 20% for the electric plugs and 10% for the video screens.

7.3 Usage phase

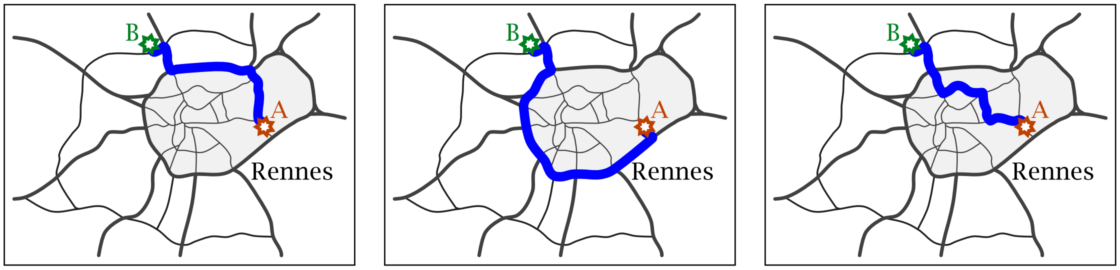
Now that the design is set and deployed in the embedded systems, the driver uses ORQA to help him choose the most suitable driving strategy for his trip.

1. The departure point is the user's office and the destination his home. As the remaining stored energy is low, he chooses an energy-saving policy, leading to $w_T = 20$ and $w_E = 80$ (Section 6.1 Table 2).

2. The GPS unit is here emulated by an online routing algorithm. Three different routes are selected. Along with the distance and estimated duration (from the routing algorithm), Table 7 presents the urban and extra-urban distribution added to illustrate the routes composition. Figure 4 illustrates the different route paths from point A to point B.

Table 7: The use-case selected routes details

	GPS data		Composition	
Route	distance	duration	urban	extra-urban
1	12.3km	15'	44%	56%
2	19.5km	18'	14%	86%
3	12.0km	22'	80%	20%



(a) route 1 (b) route 2 (c) route 3

Figure 4: The use-case routes selected by the online routing algorithm (standing for the GPS unit)

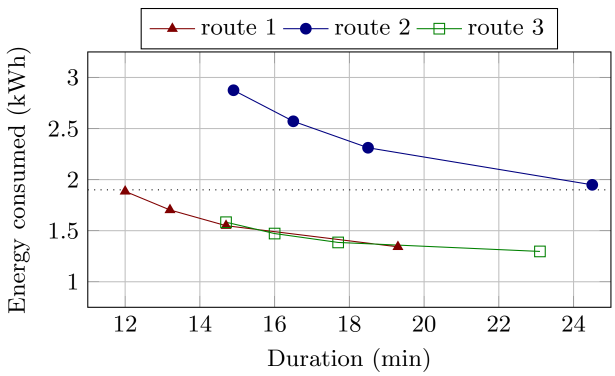


Figure 5: Evaluation results of the available routes

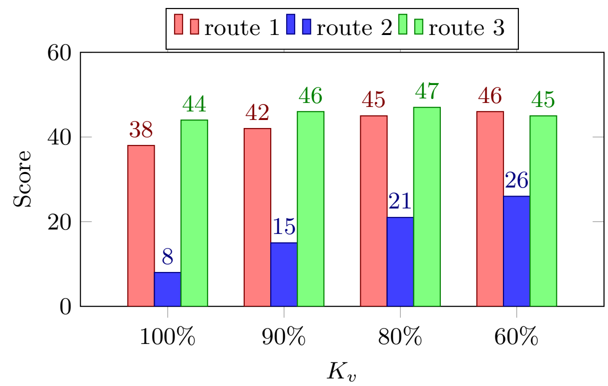


Figure 6: Evaluated scores of the available routes using an energy saving policy, the higher the better

3. The velocity coefficients 60%, 80%, 90% and 100% are applied to the three routes and the resulting trips are evaluated. The energy consumption and total duration of these evaluations are shown in Figure 5. The dotted line on the figure is the maximum energy a trip can consumed and still be feasible. The evaluated devices are the engine and the aggregated constant power of the miscellaneous devices. The lighting system is assumed to be in the *Off* state so it does not require any power. The optional devices are controlled online by the Energy Manager component and they will operate, in the most energy-saving case, at their least consuming state. Therefore, these minimal consumptions are also included into the computation.

4. Evaluated trips which consume more than allowed are discarded. Thus, every trip of the second route is discarded as they all consume more energy than allowed: $E_{battery} - \Delta_E = 1.9$kWh. For each remaining trip, a score is computed (Section 6.1 Equation 15). The score function is parametrized by the objective weights which arise from the user objective, in this use-case $w_T = 20$, $w_E = 80$. Figure 6 illustrates the different scores computed from the trips evaluations. Note that route 1 and route 3 have close scores which highly depend on the weights definition. A higher score indicates a better fitness to the user objective. The score related route trip is then selected as the best match possible. The route 3, when applied a velocity coefficient of 80%, obtains the higher score of 47. This is the route the driver is proposed to take, following the specific maximal velocities.

5. The last usage step is executed online, during the trip. The selected trip consumption has been estimated to 1385Wh, so the energy that is to be left is $E_{battery} - E_r = 615$Wh. The energy remaining for the optional devices usage (still considering the safety amount to be kept) is $E^{dedi.}_{opt.dev.} = 515$Wh (Section 6.2 Equation 16). The optional devices consumption are listed in Table 8. If they were to be operated at their maximum quality levels for the whole trip, the total energy required would be $E^{max}_{opt.dev.} = 555$Wh. As that is greater than the dedicated amount of energy, the system has to look for an optimal combination of the optional devices operating levels to keep the energy consumption within the allowed limit. The combined QoS are computed for each operating state by multiplying the devices operating states quality level to the devices user preferences. Table 8 also presents the devices respective combined QoS values. The operating states selection algorithm described previously (Section 6.2) is used. So the best selection of operating states is *eco* for the Climate Control unit (20% of combined QoS), *on* for the radio (30% of combined QoS), *on* for the plugs (20% of combined QoS) and also *on* for the passenger video screens (10% of combined QoS). This combination, possible within the limit of 515Wh, offers the best overall devices QoS possible of 80%.

To summarize, the Rater has elected the third route with a velocity coefficient of 80% and offers the driver to have an overall Quality of Service of 80% over the optional devices.

Table 8: Required energy and combined Quality of Service of the optional devices for the whole trip

Opt. device	State	Energy	Combined QoS
Climate Control	Off	0Wh	0%
	Eco	413Wh	20%
	Auto	518Wh	40%
Radio	Off	1Wh	0%
	On	15Wh	30%
Plugs	Off	0Wh	0%
	On	7Wh	20%
Video	Off	0Wh	0%
	On	15Wh	10%

7.4 Other results

We presented the results the driver would obtain with a consumption policy of energy-saving. Let us now detail the results obtained for the two other objectives.

Duration minimization.

Suppose that the driver wants to be home as soon as possible knowing that he only has 2kWh of energy left. The weights w_T and w_E have now the contrary values they had previously when the driver wanted to keep his consumption low, so $w_T = 80$ and $w_E = 20$. In this case, the best route selected is the first route with a velocity coefficient of 100% which have the higher score of 48. This route trip is 32% faster than the one chosen before (12' vs 17'42") but consumes 1885Wh. This allows only 15Wh for the optional devices. So the driver can only use the radio while the other optional devices should not be used at all. In this case, the overall QoS of the optional devices is of 30%.

Trade-off between duration and energy consumption.

Suppose now that the selected consumption policy is the trade-off one with $w_T = w_E = 50$. With a score of 43, the first route coupled to a velocity coefficient of 80% is the best possible choice. This route trip is completed in 14'7" and has an energy consumption of 1549Wh leaving 351Wh for the optional devices. With this amount of energy, the Climate Control unit is discarded but the other devices are allowed to operate normally. The overall devices QoS is thus of 60%.

8. RELATED WORKS

Several studies exist on optimal trip driving of vehicles, whether for internal combustion-driven engine or for electric vehicles. Mensing et al. [12] presents an eco-driving computation based on an optimal backward search. They offer optimal usage of the engine and gear selection to follow a driving cycle, noting that their solution is not suitable for online computation. Petit et al. [14] formulates an optimal control problem for an EV trajectory. They minimize the engine consumption with a generated optimal vehicle velocity scheme. Their inversion-based approach is stated to be of a low computation burden. Dib et al. [5] offers to minimize the vehicle energy consumption based on a dynamic-programming approach. Again, the objective is to provide velocities to lower the engine consumption for specific distance and prescribed duration. The authors state that their solution is not suitable for an online use. ORQA does not provide the driver optimal path and velocities, but tries to *estimate* what the trip will cost (both in energy and time) and how to complete it. Our route selection process is based on a direct method: a limited number of routes is explored with different velocities. As we rely on discretized values throughout the search process, our approach is stochastic. Thus, found solutions may not be optimal in the real (i.e. non-discretized) search-space and ORQA could benefit from a more sophisticated search process.

The approach presented in [9] is to compute the best route to take considering the user driving mode (*balanced*, *eco* or *sport*) and a set compartment temperature for the heater unit. Grossard et al. offers an online capable solution which is independent of the embedded systems. Our approach is similar in the user-oriented objectives though we rely on the AUTOSAR standard now widely used. ORQA is integrated in the embedded systems, so it brings the consumption knowledge to the whole system. Also, we offer to take into consideration several devices with their power requirements and their QoS, not only the heater. On the other hand, the authors take into account the road elevation in their computation. For now, we are ignoring it. With enough knowledge of the road characteristics (in our case from the GPS unit), ORQA can be extended to evaluate routes considering their elevation.

Quality of Service in software engineering is usually associated with performance. Different levels of quality are defined for a service and modeled, for instance, in contracts [3] and in enhanced architectures [13, 17]. At run-time, passed contracts are controlled and if violated, lead to new contracts. ORQA also follows this architectural approach. Selecting a device operating state can be viewed as passing a contract with the device. A contract assures the system of a certain power requirement and assures the user of a certain quality from the device. We assume that if a device is actually in an operating state, both its power requirement and its QoS are experienced. Neither the power requirement or the user experience are controlled. We rely on the feedback given by the device broker stating the actual device operating state to manage the contract. We consider, as a possible extension, to embed probes measuring the actual devices consumption to get the consumption feedback.

9. CONCLUSION

In this paper, we proposed a framework to ensure an electric vehicle driver that he will reach his destination point. This framework is realized by embedding a system-wide energy manager. The manager is based on a pre-computed model of energy consumption for all devices, a set of user preferences and different levels of Quality of Service given for each optional device.

From these models, the framework searches for available routes and computes for each one of them the duration and the consumption for both nominal driving and reduced velocities. The framework relies on an energy model defining the consuming devices embedded in the vehicle to compute the global consumption. This model is introduced aside of the current embedded systems modeling done in AUTOSAR. It is used to generate an enhanced AUTOSAR model which still validates the standard, so it does not break the compatibility with existing tool-chains and can be integrated seamlessly with vendors solutions.

We are working on using the approach to optimize the different proposed levels and velocity coefficients by exploring different cases in different configurations. The final goal is to define optimal strategies to embed in the vehicle. A prototype implementation is currently under development and will be the object of a later publication. Also, we are working on bringing the overall optional devices Quality of Service sooner in the process, in the route trips evaluation. As it can be seen in our example, route trips can have close scores evaluating their fitness to the user consumption-policy. But some let more available energy for the optional devices, making their QoS better. The idea is to take into consideration the optional devices QoS sooner, so it can be optimized along the route path.

10. REFERENCES

[1] AUTOSAR Partnership. AUTOSAR - the worldwide automotive standard for E/E systems, 2011.

[2] S. Barbusse, D. Clodic, and J.-P. Roumégoux. Automobile air conditioning: effects in terms of energy and the environment. *Research - Transports - Security (Recherche - Transports - Sécurité)*, 60:3–18, 1998.

[3] A. Beugnard, J.-M. Jézéquel, N. Plouzeau, and D. Watkins. Making components contract aware. *Computer*, 32(7):38–45, 1999.

[4] R. Bosch. *Automotive Handbook*. John Wiley & Sons, 8th edition, may 2011.

[5] W. Dib, A. Chasse, D. Di Domenico, P. Moulin, and A. Sciarretta. Evaluation of the energy efficiency of a fleet of electric vehicle for eco-driving application. *Oil & Gas Science and Technolology - Rev. IFP Energies nouvelles*, 67(4):589–599, sep 2012.

[6] M. Ehsani, Y. Gao, and A. Emadi. *Modern Electric, Hybrid Electric, and Fuel Cell Vehicles: Fundamentals, Theory, and Design, Second Edition*. Power Electronics and Applications Series. CRC Press, 2nd edition, sep 2009.

[7] S. Fürst. Challenges in the design of automotive software. In *Proceedings of the Conference on Design, Automation and Test in Europe (DATE)*, 2010.

[8] Y. Gao, L. Chen, and M. Ehsani. Investigation of the effectiveness of regenerative braking for EV and HEV. *SAE transactions*, 108(6), aug 1999.

[9] M. Grossard, S. Kachroudi, and N. Abroug. An optimal energy-based approach for driving guidance of full electric vehicles. In *International Symposium on Industrial Electronics (ISIE), 2012 IEEE*, pages 1708–1713, may 2012.

[10] C. Knüchel, M. Rudorfer, S. Voget, S. Eberle, R. Sezestre, and A. Loyer. Artop – an ecosystem approach for collaborative AUTOSAR tool development. In *International Congress on Embedded Real Time Software and Systems*, 2010.

[11] J. Larminie and J. Lowry. *Electric Vehicle Technology Explained, 2nd Edition*. John Wiley & Sons, 2nd edition, 2012.

[12] F. Mensing, R. Trigui, and E. Bideaux. Vehicle trajectory optimization for application in ECO-driving. In *Vehicle Power and Propulsion Conference (VPPC), 2011 IEEE*, sep 2011.

[13] L. Palopoli, T. Cucinotta, L. Marzario, and G. Lipari. AQuoSA – adaptive quality of service architecture. *Software: Practice and Experience*, 39(1):1–31, 2009.

[14] N. Petit and A. Sciarretta. Optimal drive of electric vehicles using an inversion-based trajectory generation approach. In *Proceedings of the 18th IFAC World Congress, 2011*, sep 2011.

[15] R. Sehab and G. Feld. An online estimation of energy recovery in an electric vehicle using ARTEMIS mission profiles. In *Vehicle Power and Propulsion Conference (VPPC), 2012 IEEE*, pages 333–338, oct 2012.

[16] D. Steinberg, F. Budinsky, M. Paternostro, and E. Merks. *EMF: Eclipse Modeling Framework, 2nd Edition*. Eclipse Series. Addison-Wesley Professional, 2nd edition, dec 2008.

[17] J.-C. Tournier, J.-P. Babau, and V. Olive. Qinna, a component-based QoS architecture. In G. Heineman, I. Crnkovic, H. Schmidt, J. Stafford, C. Szyperski, and K. Wallnau, editors, *Component-Based Software Engineering*, volume 3489 of *Lecture Notes in Computer Science*, pages 107–122. Springer Berlin Heidelberg, 2005.