VIENNA 2018 TRA
TRANSPORT RESEARCH ARENA
A digital era for transport
solutions for society, economy and environment

Proceedings of 7th Transport Research Arena TRA 2018, April 16-19, 2018, Vienna, Austria

Energy Consumption Optimization and Vehicle Dynamics Performance Improvement for a Scalable P-HEV e-AWD Power Split Architecture to be Validated on a B-Segment Vehicle

P. Prieto[a], E. Trancho[a], I. Real[a], M. Allende[a], D. Roiu[b], R. Marco[b] and V. Ravello[b]

[a]*Tecnalia Research and Innovation, Industry and Transport Unit*
Parque Científico y Tecnológico de Bizkaia, c/ Geldo, Edif. 700, 48160 Derio, Spain
[b]*CRF S.C.p.A – Vehicle Research & Innovation, Strada Torino 50, Orbassano(TO), Italy*

Abstract

In this paper, a novel high level control-management strategy and architecture for plug-in hybrid complex hybrid vehicles energy consumption optimization and vehicle dynamics improvements is presented. A virtual Hardware in the Loop strategy is used for Vehicle Framework modelling, ensuring a high fidelity forward vehicle model. The propulsion architecture consists of a combination of two electric motor based e-drives and one Internal Combustion Engine (ICE). One e-motor is coupled with the ICE at the front axle by means of a modified belt system while the other one powers the rear axle through an integrated single speed transmission with disconnect. This architecture allows multiple operation modes: combinations of serial/parallel/split hybridization and full electric, providing e-4WD capability. Hybrid control strategies combining heuristic methods and optimization techniques are proposed to manage the power distribution between the Internal Combustion Engine and the electrical machine. In addition, an electronic horizon consisting of a smart energy management algorithm to enable predictive control hybrid strategies is presented. This uses information from vehicle sensors together with web services to extract information like traffic, weather, route inclination etc. All the information is used a-priori to optimize the energy management by using the best hybrid powertrain strategy and/or control large power electrical loads.

Keywords: Electric Vehicle; hybrid; energy optimization; Hardware in the Loop; electronic horizon; strategies.

1. Introduction

The main purpose of the ECOCHAMPS project is to improve the competitiveness of hybrid thermal-electric powertrains for Light Duty (LD) and Heavy Duty (HD) vehicle applications. Within the project Work Package 3, several European partners are collaborating to study an electric All Wheel Drive (eAWD) plug-in hybrid powertrain for LD application, to be validated on a FIAT 500X prototype vehicle. The approach adopted is to try to achieve a favorable consumption and CO_2 reduction (20% powertrain efficiency improvement in respect of the best in class same segment state of art vehicle) versus the extra added costs (a maximum 10% cost premium on the conventional model on which the demonstrator is based) by taking advantage of 1) high voltage e-components (e-machines, power electronics and batteries), 2) add-on approach for the front axle hybrid powertrain and the integrated rear electric axle, 3) a hybrid architecture that exploits the advantages of the dual clutch transmission and compensates for its limitations through the hybridization, 4) a wide usage flexibility (pure ICE, parallel, split or series hybrid and pure electric modes), 5) pure electric range based on the battery sizing (> 25 km), 6) advanced energy management, and 7) a solution modularly applicable to different vehicle segments.

2. Control System Development Framework

A control system development framework (DynaSim) is presented using high fidelity forward vehicle model and electric/electronic models allowing control engineers to run and validate their hybrid control algorithms at very early development stages, even before the actual ECUs (Engine Control Units) hardware were available. Driving scenarios and extended tests can be carried out for the complete control systems using DynaSim allowing HIL (Hardware in the Loop) and SIL (Software in the Loop) techniques for real time simulation. Some of the major outcomes of the HIL and SIL testing activities will be (1) validation, calibration, optimization and performance testing of selected control functions, (2) integration and validation of the drivetrain control systems together with the chassis control system of the demonstrator platform, (3) assessment and validation of safety functions, (4) pre-assessment of drive cycle efficiency (before the measurements in the vehicle), and (5) virtual test driving using realistic test scenarios (Varela (2009))[1].

2.1. Vehicle Architecture

The architecture chosen is a combination of two electric motor based e-drives and one ICE, as shown in Figure 1. One e-motor is coupled with the ICE at the front axle by means of a modified belt system; the other e-motor powers the rear axle through an integrated single speed transmission with disconnection technology (allowing disconnection at high speed to decrease the mechanical losses). This vehicle architecture allows multiple operation modes: combinations of serial/parallel/split hybridization and full electric only. In addition, e-4WD capability is provided.

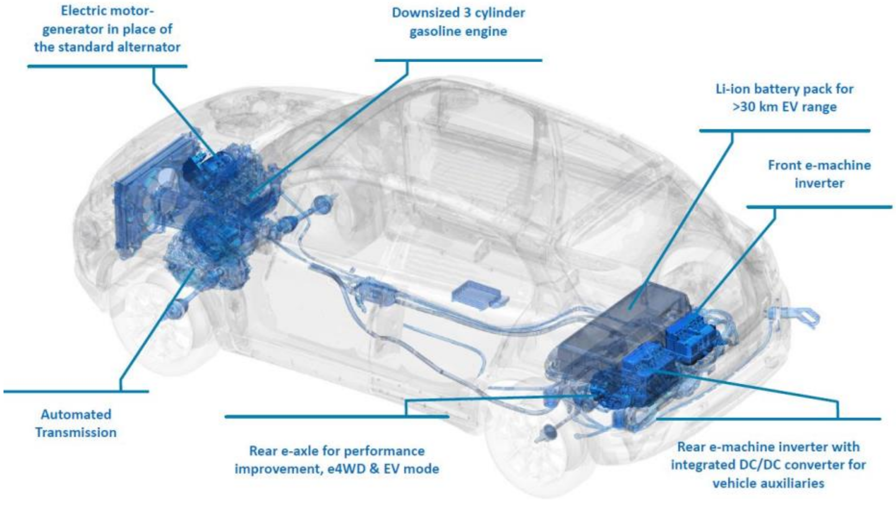

Fig. 1 Vehicle architecture, ghost view

2.2. Vehicle Framework Model Structure

The Vehicle Framework Model (VFM in advance) developed, is a high fidelity multidomain model, including the main components of the vehicle prototyped; two e-Motors (front axle and rear axle), Internal Combustion Engine, High Voltage Battery, Low Voltage Battery, front and rear transmission with integrated differential, DC/DC (from HV Battery to LV Battery), front and rear DC-AC (Motor controllers) and DYNACAR (Vehicle Dynamics mechanical multibody) (Pena (2012))[2].

The Hybrid Vehicle Plant Model is developed in the Matlab Simulink environment using a model-based design approach. This methodology is key to reduce powertrain modeling, enhance quality, and shorten the development cycle time. It allows using the same models throughout the development process for design, analysis, simulation, automatic code generation and verification. This approach allows for abstraction from the hardware side, enabling the control strategies developed to be used in any other hardware in the future, thus ensuring production continuity.

This VFM could be consider as a SIL/HIL simulator for testing automotive control systems and can be used during the entire development process. Starting from the model architecture generated at project early stages, all the necessary components to be modelled, according to the described vehicle, are identified.

The framework development has two different phases. The first one is related with the no real-time modelling and testing (computer phase, SIL) while the second one deals with real time communication and data exchange (real time phase, HIL (Short, (2005))) [3].

2.2.1. SIL Environment stage

A Simulink framework is developed with all the necessary models to emulate the vehicle's behavior. As shown in Figure 2, each system modeled is isolated in one block where the physical model and the control logic simulates real behavior. Due to the fact that it is a SIL environment, all the signals exchanged between blocks are virtual wires.

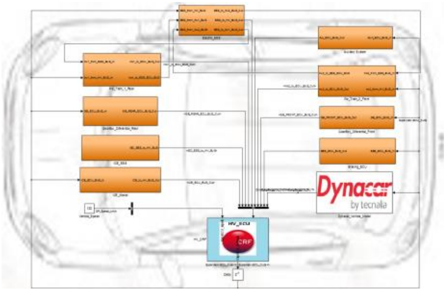

Fig. 2 SIL environment model

2.2.2. HIL Environment stage

This stage implies the virtual models to be deployed into real time systems. All the signals exchanged virtually, now are sent through real physical communication used in the vehicle (CAN bus). Two Real Time systems are needed to deploy the models to fulfill the simulation requirements, due to the different models execution times. The first one runs all the component models that are part of the vehicle plant and the second one runs the vehicle dynamics model (DynaCar). The vehicle virtualization includes co-simulated environments (electrical, mechanical and thermal) feeding the vehicle dynamics model which all together constitutes DynaSim control system simulator.

It is necessary to consider complexity layers or domains to build the hybrid vehicle plant. For each component, the physical layer and the control layer (virtual ECU) have been developed. As an example, the electrical power system (physical) had been modeled using Simscape Power System toolbox together with the BMS (Battery Management System) virtual ECU.
The vehicle dynamics simulation consists of a rolling chassis mechanical model based in multibody formulation, which is formed by a series of solids that model the vehicle. The component description for the whole chassis is

provided by the original equipment manufacturer (OEM). This is of vital importance because the model needs a comprehensive description of suspension and tires. A 3D definition road surface is used to calculate the wheel contact points and the forces involved.

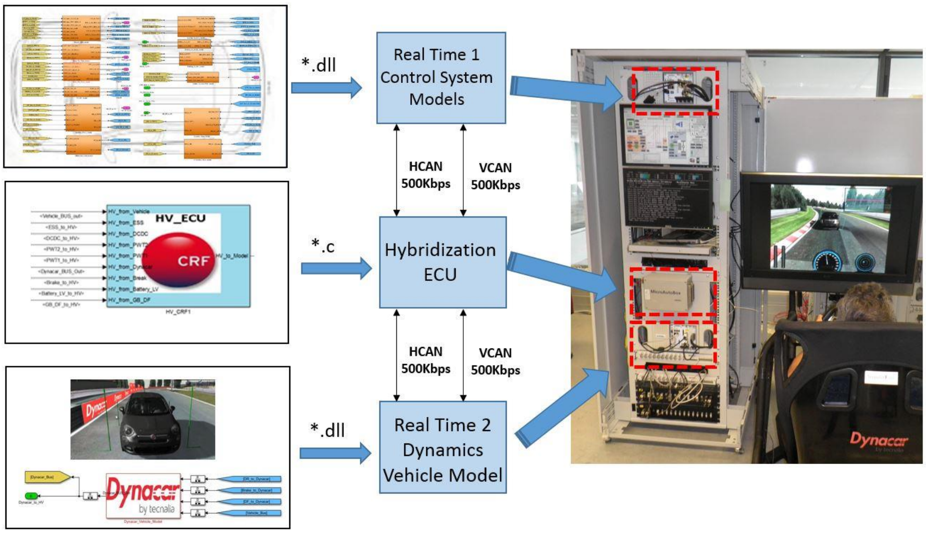

Fig. 3 Vehicle framework model

Custom made tools have been developed, allowing monitoring and set-up all the framework, as for example disabling virtual ECUs when plugged the real one. It also enables using the DynaSim for test rigs applications without any further effort.

3. Energy Management strategies implemented in the hybridization ECU.

The Hybrid Control Unit (HCU) is the supervisor of the vehicle functions; it allows to integrate the standard vehicle E/E and network architecture with the management of new components installed on the demo-car to provide plug-in hybrid functions. The HCU has been developed with a Logics module that has a general signals interface, allowing to be tested and used in different environment (simulations, HIL or vehicle validator) or different targets (rapid prototype ECUs). The main objective was to develop a general signals interface that comprised a set of Inputs/Outputs independently from the physical components and signals present on the vehicle (CAN, analogue or digital signals). In this way, it is possible to use the same Logics module with different simulation platforms, just by means of specific I/Os adaptation blocks and a set of parameters, which is configurable according to the simulation environment or for real vehicle implementation.

The entire HCU dSPACE SW architecture, with I/Os subsystems and Logic module, used for the simulation and HIL activities. For the integration in the Vehicle Validator, it a dSPACE unit has been used, a rapid prototyping platform; code is generated through a model based approach. The chosen HW is a 1401-1511 platform, because vehicle architecture requires 4 CAN connections and a large number of Digital and Analogue I/Os.

The implemented hybrid control strategy combines heuristic methods and optimization techniques to manage the power distribution between the ICE and the e-drives. The vehicle operating modes and the interactions between the modes are determined by event or time based conditions. The power split inside an operating mode could be determined by either heuristic methods or by the optimization techniques. In this strategy, a target performance metric that consists of energy consumption and emissions is minimized over permissible ICE and EM E-Machine) torque requests. The solution is then adjusted according to battery SOC (State Of Charge) considerations.

The main logic present inside the HCU SW is:

- Hybrid Supervisor: vehicle states supervision and transitions management among powertrain states.
- Limits Calculation: component’s status acquisition (High Voltage Battery, front and rear e-drives and ICE) and limits evaluation.

- Power Calculation: driver requests and vehicle status acquisition and power calculation for Traction, Regenerative and Auxiliaries.
- Optimizer Mode Evaluation: Most efficient vehicle mode evaluation, guaranteeing requested performances without affecting vehicle drivability.
- Functions Manager: functions activation definition and functions priority discrimination.
- Mode Manager: HMI (Human-Machine Interface) input acquisition mode and activation/deactivation of a vehicle mode management.
- Torque Manager: torque split definition on the basis of vehicle mode, taking into account vehicle performance, drivability and stability.
- DC/DC Manager: energy flow management, from High Voltage to Low Voltage battery.
- Components Manager: Components logics management (command sequences, startup etc.).
- Additional Logic, as stand-alone modules (Plug-in recharge manager, thermal manager, HMI manager).

The core of the HCU SW architecture are the Hybrid Supervisor and the Functions Manager.

- The Hybrid Supervisor manages the vehicle status and transitions: its outputs are used by components modules to react coherently to the supervisor request. For example, if the supervisor requests a transition to the EV mode, the engine control must react by stopping the ICE, the rear transmission module must react by opening the clutch and the rear e-machine must guarantee the driver power requests, according to the activated function. The Functions Manager, on the basis of driver inputs and chosen vehicle mode, activates or deactivates some functions (defining priority and determining consequently vehicle status). The Functions Manager logic depends (directly or indirectly through other modules elaborations) on:
    - Selected vehicle mode.
    - ICE and transmission data.
    - Electric motors data.
    - Battery system data.
    - Driver Request.
    - Other vehicle data.

- The Function Manager is made of different modules, each of them constitutes a modular function, and by a function selector that defines priority among functions and allows activation/deactivation of each function. Each function has two outputs: one is the function request, an enable/disable signal, and one is the Power request related to the function. In Figure 4, the high-level software block diagram of a generic hybrid function is shown.

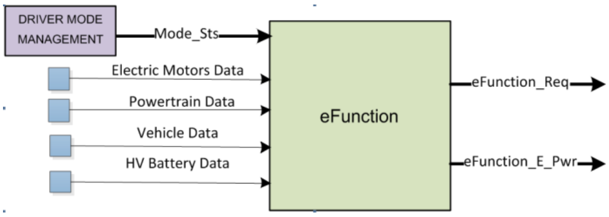

Fig. 4 High-level software block diagram of a generic hybrid function

The implemented hybrid functions are detailed below. The front and rear electric machines can be used both in traction mode and in regenerative mode, even if priority is to use the rear e-machine and manage the front one for specific functions, such as engine re-cranking and generator mode.

- eGENERATOR: Enhanced smart alternator. Capability to generate electric power (i.e. recharge battery). E-machines are used as generators. ICE is increasing the torque in order to keep the same output torque, as requested by the driver.
- eCOASTING: Regenerative deceleration. It simulates the ICE braking while recovering energy during vehicle deceleration without brake pedal action. The rear e-motor is providing a small negative torque, in order to recharge slightly the battery.
- eBRAKING: Regenerative braking. It allows energy recovery during vehicle deceleration with the brake pedal pressed. The rear e-motor is providing higher negative torque, in order to recharge the battery.

- eCRANKING: Automatic cranking performed through the front electric machine installed on the engine auxiliary belt.
- eCREEPING: Active when driver releases the brake pedal without pressing the accelerator pedal from a stop condition. It can be used for electric queuing, it simulates automatic gear-shift creeping, but using the rear electric machine. It is always available, also with the engine running.
- eTRACTION: The ICE/e-motor torque ratio as well as speed are managed to provide the requested power at wheels. The eTRACTION is possible in all the vehicle modes with different logics (AUTO Mode, AWD Mode, SPORT Mode, EV Mode and CHARGE Mode).

4. Use of control systems development framework

4.1. Validation, calibration, optimization and performance testing of the Hybridization ECU.

DynaSim main purpose is the validation, calibration and optimization of the HCU system developed by CRF. For this task a custom monitoring tool has been developed, allowing checking in real time the vehicle different parameters. The following pictures show the monitoring tool (Figure 5), where graphical energy flow are combined with engineering charts, allowing quick data analysis.

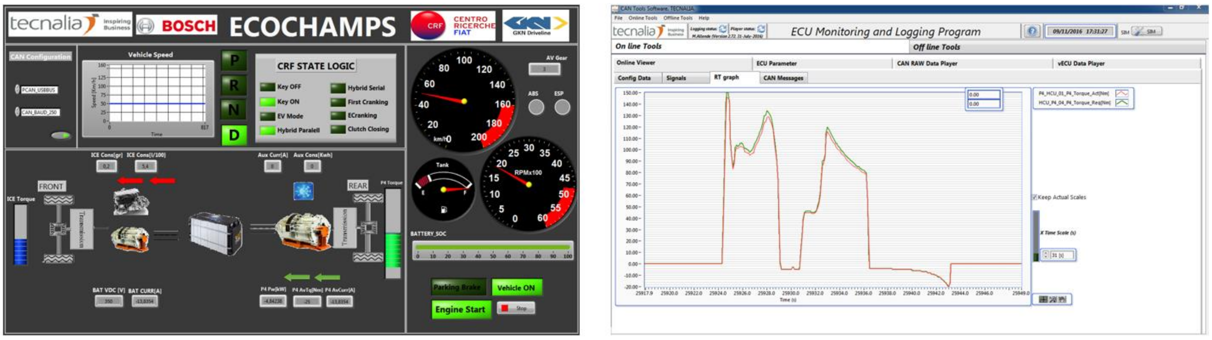

Fig. 5 Bespoke Monitoring Tool screen captures

The device under test, CRF hybrid control unit, is integrated to the DynaSim using real physical communications. DynaSim provides all the parameter emulating the real vehicle behavior to the unit (for example functionalities as start/stop, idling, brake blending, energy balance, emergency safety function etc.). This has allowed to perform a virtual validation test plan which includes 1) dynamics, 2) electric and 3) control logics. Examples of uses are 1) Rear axle stability assessment and initial energy recovery logic tuning using ISO maneuvers like Brake-in-a turn, throttle off, straight line braking etc. (Figure 6), 2) Energy optimization strategies using different driving cycles have been fine tuned to maximize performance and minimize consumption, 3) CAN Bus load test has allowed to validate the different networks supporting the vehicle integration stage, and 4) Battery power flows when hybrid modes are activated/deactivated to avoid overcurrent while charging/discharging.

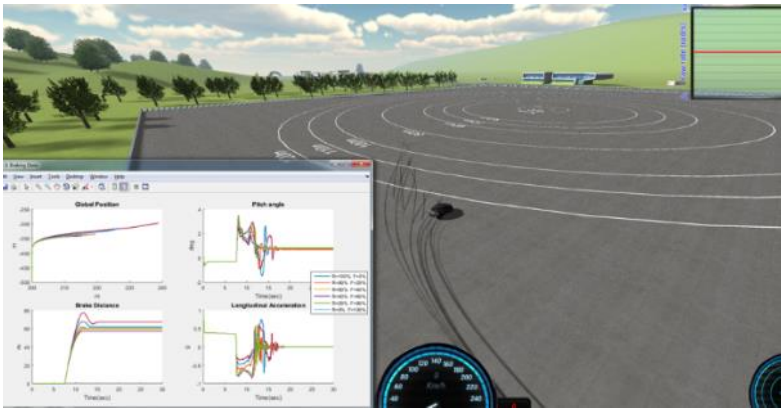

Fig. 6 Brake in a turn maneuver for energy recovery logic and stability assessment during braking

5. E-Horizon

5.1. Introduction:

A Smart Energy Management (SEM) adaptive-predictive control strategy is designed and implemented. The capability to pre-calculate the total trip energy needed and foresee the extra energy required for trip completion makes SEM system a key factor for hybrid powertrain strategies optimization. A diagram of the developed SEM strategy is shown in Figure 7.

A predictive backward Quasi-Steady State (QSS) vehicle simulation is used to predict on board the energy consumption of a real trip. A Web Service application (Google Maps) is used for that purpose, from where the trip's speed and gradient profiles are obtained. An energy management strategy is proposed to compensate energy deviations between the QSS and real driving, which includes the Driving Style (DS) estimation in its calculations due to its influence on the energy consumption (Schreibel, (2014))[4]. A more realistic scenario is achieved thanks to the driving style estimation, which influence is considered in the energy management strategy calculations. In addition, a GPS device to monitor the route is included. A geo-fence logic is implemented to ensure the driver is following the planned route and the correct information is used to predict the required energy to destination. In the following subsections, a detailed description of the control system is provided. Hardware and firmware implementation aspects are also discussed.

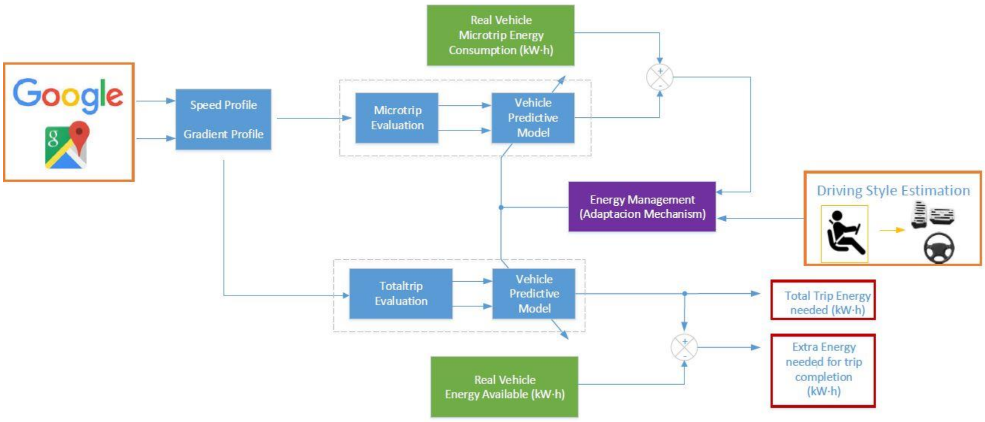

Fig. 7 SEM control strategy scheme

5.2. Control System

5.2.1. Web Service

A Web Service is used to obtain the corresponding drive cycle. This is achieved throughout an external driver interface display, in which the user indicates the desired destination. The SEM device connects to Google Maps® API to obtain the quickest route including traffic status. The system assumes the driver delegates the control of the route to follow to the Navigation System (Automatic driving). After a post processing stage, the system stores in memory the speed (speed vs distance) and gradient (elevation versus distance) profiles. The resulting route profiles are used by the QSS vehicle to estimate the energy required to go over the trip planned.

5.2.2. QSS vehicle

A vehicle simulation is used to estimate the energy flow and the energy consumption (Froberg, (2004)) [5]. In this sense, two main philosophies are distinguished to model specific vehicles:

- Forward looking model: The driver model modifies its commands in order to follow a desired vehicle speed/acceleration trace. A real driving reaction is emulated using this approach at the cost of developing complex component models which require high computational cost.
- Backward looking model: From a given speed profile, a simplified vehicle model determines the energy flow, from the vehicle back to the engine. Quasi-steady state models are commonly used when considering this approach and realistic control response is neglected.

In this work, an adaptive-predictive Backward QSS is proposed for vehicle modeling, due to its simplicity and low time consuming. It allows running in real time in a low cost embedded device.

For a precise cycle evaluation, the total route distance is divided in a defined number of iterations (microtrips), where the estimated energy is adapted using the real vehicle data by the Energy management logic. A total trip evaluation is carried out, determining the total energy required to complete the route. This information is then used by the hybridization strategy to optimize the energy management in the powertrain module.

5.2.3. Energy Management strategy

An Energy Management strategy is proposed to fine tune the QSS energy calculation. The error between the estimations and the real process is quantified in each iteration. The adapted power, at instant k, is given by the following expression:

$$P'_{mech,QSS}(k) = \delta(k) \cdot P_{mech,QSS}(k) \tag{1}$$

Being,

$$\delta(k) = W_{DS}(k) \frac{E_{microtrip,RV}(k)}{E_{microtrip,QSS}(k)} \sum_{i=1}^{k-1} \frac{E_{microtrip,RV}(i)}{E_{microtrip,QSS}(i)} \tag{2}$$

Where $\mathrm{P}_{\mathrm{mech,QSS}}$ is the required mechanical power calculated by the QSS, δ is the compensation factor, and $E_{microtrip,RV}$ and $E_{microtrip,QSS}$ are the energy per iteration (microtrip) corresponding to the real vehicle and QSS vehicle simulation respectively. The driving style is also taken into account using the brake, throttle and the steering inputs which directly impacts into the consumption. It uses a weighting function, $W_{DS}(k)$, which outputs three general types of driving style: aggressive, normal and calm, as expressed as follows:

$$W_{DS}(k) = W_{Aggresive}(k) + W_{Normal}(k) + W_{Calm}(k) \tag{3}$$

In this study, the DS identification has been carried out using Fuzzy logic (Langari, (2005))[6]. Average acceleration and standard deviation of acceleration over a specific range of samples are used to identify the driving style during each microtrip.

The proposed energy management gives a more realistic evaluation of the total route using predictive approach.

5.3. Geofence

A geo-fence based diagnosis strategy has been implemented to ensure a correct route tracking using GPS. Virtual barriers are set up to determine the route boundaries for each latitude-longitude location received from Google Maps® (Stevens, (2017)) [7]. Approximating the Earth's surface as a sphere with radius given by the WGS84 ellipsoid for a given altitude, a bounding box defined by the maximum latitude, maximum longitude, minimum latitude and minimum longitude is obtained as shown in Figure 8. Then, an alert is triggered if the actual GPS position exceeds the defined virtual boundaries. A new route will be calculated if the alert persists during long period, forcing a new energy simulation calculation.

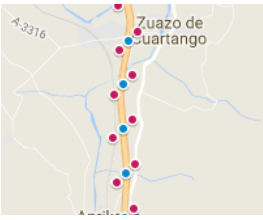

Fig. 8 Geofencing example, including the route reference (blue dots) and virtual barriers (red dots).

5.4. Implementation

5.4.1. Hardware platform

An electronic device powered by ARM® Cortex® processor running open software operation system (Linux) has been used for the SEM algorithm deployment. For the demonstrator, a fully open source hardware has been selected, Beaglebone Black Wireless platform. The BeagleBoard.org Foundation provides source of everything required to integrate into own product. The distribution used is Debian 8.7, known as Jessie. The device tree, data structure for describing hardware, has been modified to be able to use the two CAN Bus interfaces available on the TI AM335x processor (Molloy, (2015)) [8]. The device contains the Octavo Systems OSD335x System-in-Package which integrates the Texas Instruments Sitara ARM® Cortex®-A8 AM335x Processor, DDR3 memory, TPS65217C PMIC, TL5209 LDO, and all needed passive components into small package. This allows for a vastly simplified final system design.

In Figure 9 (a) the device is connected throughout CAN converted to a PC running a custom made tool which emulates real vehicle's messages for development purposes. The GPS, G-STAR IV, is plugged to the USB port.

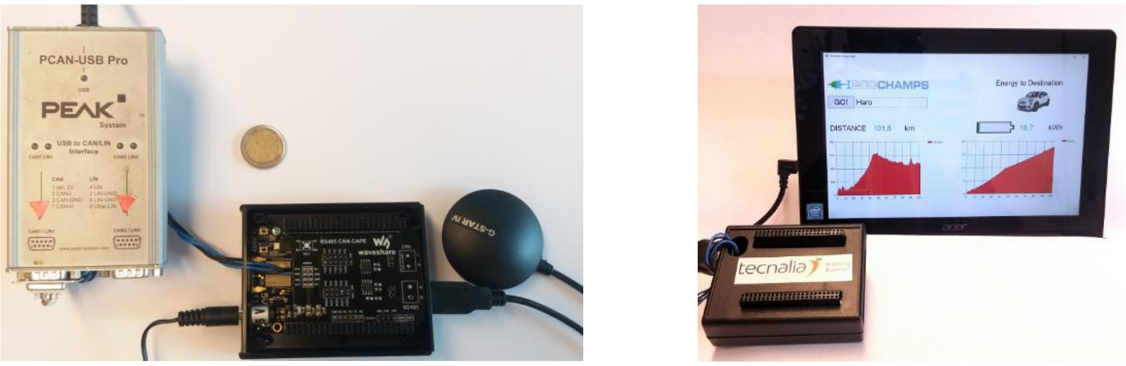

Fig. 9(a) Laboratory test hardware, (b) Driver interface showing one of the multiple screens available

A Human Machine Interface (HMI) display has been developed which interfaces with the SEM device using UDP protocol. The SEM sends in real time all the relevant information to be showed to the driver. A 10.1'' touchscreen tablet PC has been selected running a C# software custom made for the ECOCHAMPS project. In Figure 9 (b) an example screen can be appreciated, where route elevation profile and energy estimated charts are displayed after the user inputs the destination.
The SEM outputs the required energy estimation to the CAN bus for advanced predictive hybridization strategies.

5.4.2. Firmware

Model based design has been used to develop the proposed SEM algorithm. A diagram of the implemented framework is shown in Figure 10.

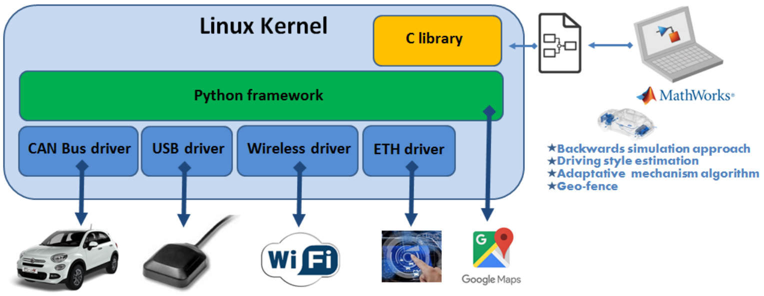

Fig. 10 SEM strategy implementation architecture

The framework has been developed in Python language which allows easily manage high level task like CAN bus and wireless communications, web services iteration (Google Maps®), GPS and user display.

The complete SEM algorithms (predictive backward QSS, the energy management strategy and geofence) have been developed under Simulink® and Stateflow® auto generated C code. The software runs a state machine, making code more efficient, easier to debug and helping to organize the program flow. One of its multiples advantages is the possibility to simulate and validate the algorithms before its integration in the real target.

The framework serves web service information, CAN bus parameters and GPS signal to the SEM strategy. This approach benefits of using a language like Python to manage high level complex task with a powerful-fast C code for the control systems algorithms.

6. Conclusions

A novel control system framework, DynaSim, has been developed allowing vehicle/powertrain engineers to run and validate their hybrid control algorithms at very early development stages, even before the actual ECUs hardware was available for ECOCHAMPS project. The main advantages of the proposed approach are 1) minimizing development and integration time, 2) ensuring quality, 3) reducing physical testing and validation through virtual driving, and 4) Fault injection capability for safety assessment. In addition, a smart energy management prototype has been developed enabling predictive control hybrid strategies and/or control large power electric loads (i.e. HVAC). The vehicle architecture allows multiple operation modes: combinations of serial/parallel/split hybridization and full electric, providing e-4WD capability. In order to manage the power distribution between the Internal Combustion Engine and the electrical machines the hybrid control strategies combining heuristic methods and optimization techniques have been proposed. The hybrid control unit, has been integrated to the DynaSim environment using real physical communications which provided all the parameters emulating the real vehicle behavior to the unit. This has allowed to perform a virtual validation test plan including dynamics aspects, electric behavior and control logics. The real vehicle testing and validation of the hybrid control logics and the smart energy management with predictive control will be performed in the Q1 of 2018.

7. Acknowledgment

This project has received funding from the European Union’s Horizon2020 research and innovation programme under Grant Agreement no. 653468.

8. References

1. Varela, J., et al, 2009, “Integrated Modeling Approach for Highly electrified HEV. Virtual Design and Simulation Methodology for Advanced Powertrain Prototyping”, EVS24.
2. Pena, A., et al, 2012, “Development and validation of Dynacar RT software, a new integrated solution for design of electric and hybrid vehicles”, EVS26, pp 1-7.
3. Short, M., et al, 2005, “Hardware in the loop simulation of embedded automotive control system”, Intelligent Transportation Systems, 2005.Proceedings. 2005 IEEE, DOI: 10.1109/ITSC.2005.1520052
4. Schreibel, V., 2014, “Range Prediction of Electric Vehicles”, 58th ILMENAU SCIENTIFIC COLLOQUIUM, pp 1-11.
5. Froberg, A., et al, 2004, “Dynamic Vehicle Simulation Forward, Inverse and Mixed Possibilities for Optimized Design and Control”, SAE Technical paper series SP-1826, 2004-01-1619.2004
6. Langari, R., et al, 2005, “Intelligent Energy Management Agent for a Parallel Hybrid Vehicle – Part I: System Architecture and Design of the Driving Situation Identification Process”. IEEEE Transactions on Vehicular Technology, vol 54, n3, pp 925-934
7. Stevens, M., N., et al, 2017 “Specification and Evaluation of Geofence Boundary Violation Detection Algorithms”, International Conference on Unmanned Aircraft Systems, pp 1588-1596
8. Molloy, D., 2015 “Exploring BeagleBone: Tools and Techniques for Building with Embedded Linux” editorial Wiley, ISBN-13: 9781118935125, ISBN-10: 1118935128.