To: Cyber-Physical Systems Workshop Program Committee
From: Tom Fuhrman, General Motors R&D
Subject: Position Paper for NSF Workshop on Cyber-Physical Systems
Date: September 8, 2006

Introduction: The Automotive Context

Software-based electronic control systems are increasingly being used in the automotive industry to provide convenience and safety features to vehicle drivers and passengers, with increasing levels of automation and control authority. A growing trend is to assist the driver in maintaining safe control over the motion of the vehicle in a variety of circumstances including, but not limited to, congested traffic conditions, adverse weather and road conditions, varying states of health of vehicle equipment, and varying skill levels of drivers. Previously such assist has been provided in the form of information or warnings to the driver, but increasingly such assist will be provided by actively manipulating actuators that control vehicle longitudinal acceleration and deceleration, lateral direction, and vertical displacement. The long term trend is towards partial or even fully autonomous operation of a single vehicle, or even of fleets of vehicles.

Three fundamental challenges of today’s cyber-physical systems

1. The first fundamental challenge of today’s automotive cyber-physical systems is how to guarantee the functional correctness of complex interactions between related vehicle features. Several convenience and safety features may, at various times and under various conditions, desire to exert control over actuators such as engines or wheel motors, brake calipers, road wheel steering angles, and suspension dampers. For example, numerous safety features (such as adaptive cruise control, forward and/or rear crash avoidance, and curve speed control) may desire to apply varying amounts of braking torque at various rates under various (possibly overlapping) conditions. Currently, unintended interactions between such interacting features are frequently discovered only upon physical integration into test vehicles, due to the difficulty of modeling these complex interactions in relation to the dynamic behavior of the vehicle’s motion, and also due to the difficulty of anticipating every possible driving scenario in which these interactions may occur.

2. The second fundamental challenge of today’s automotive cyber-physical systems is how to guarantee compliance with parafunctional requirements in distributed implementations of vehicle features. Once the functional behavior of a set of vehicle features is defined, this functionality needs to be allocated onto a (typically) distributed architecture consisting of sensors, computational nodes, and actuators, connected by hardwired signal paths or multiplexed serial data busses. The parafunctional characteristics of the resulting system, particularly synchronization (data coherency) and timing (latency and jitter), need to be evaluated for conformance to requirements. Currently, compliance with these parafunctional requirements is verified only upon physical integration into test vehicles, due to the unavailability of models, model parameters, and detailed design information earlier in the development
lifecycle. Early schedulability analyses for processing tasks and datalink messages are performed to the extent possible, but lack the fidelity to ensure that no problems are encountered during physical integration and testing.

3. The third fundamental challenge of today’s automotive cyber-physical systems is what types of affordably run-time architectures to deploy to maintain prescribed levels of functionality and performance (ranging from full to degraded to failsafe) in the presence of hardware and software faults. As described earlier, the level of criticality of software-based functionality in automotive embedded computing systems is increasing rapidly. Most current automotive embedded control systems assume that the driver always maintains complete responsibility for safe control of the vehicle’s motion via direct mechanical or hydraulic linkages between the driver’s input commands (such as accelerator pedal, brake pedal, and steering wheel) and the end effectors that control the vehicle’s motion (throttle plates, brake calipers or drums, and road wheel angles). But future generations of automotive convenience and safety features will insert their own layer of intelligence and decision making into the control path, possibly taking advantage of by-wire technologies to break the direct connection between the driver’s input command and the output devices being controlled.

**Three most important research challenges**

Each of the above three fundamental challenges leads naturally to its corresponding research challenge.

1. Simulation-based or formal methods of analysis are needed that can analyze the behavior of multiple interacting modal control systems that interact continuously with the dynamics of vehicle motion, with broad coverage of the possible driving scenarios that could be encountered during real-world driving situations. Such a comprehensive analysis methodology would need to combine multiple analysis domains, each of which is intractable individually, much less tractable in combination. For example, exhaustive formal verification of hybrid control systems remains elusive, and becomes even more intractable when combined with driver skill and driver workload modeling.

2. New layered or hierarchical scheduling frameworks are needed that can enable tractable analysis of schedulability and other parafunctional requirements such as latency requirements. Because individual scheduling policies for processing tasks and datalink messages interact in terms of their impact on overall system latencies and throughputs, new decoupling approaches (separation of concerns) are needed so that the analysis can become tractable. A future trend will be the automatic generation or synthesis of correct-by-construction implementation artifacts such as schedules. One specific example is the need for automatic generation of vehicle-wide static schedules for processing tasks and bus messages in time-triggered systems such as those implemented using FlexRay busses. Further, because asynchronous (CAN-based) and synchronous (FlexRay-based) communication links will coexist in
automotive architectures for the foreseeable future, schemes for the analysis of parafunctional attributes — and synthesis of provably correct implementations — are urgently needed for mixed asynchronous / synchronous architectures.

3. The increasing criticality of software-based control systems in high volume consumer and commercial products such as automobiles needs to be addressed by affordable architectures that can tolerate hardware, software, and communication faults to the required level of performance and functionality, in the presence of challenges such as unskilled or impaired drivers, aged and poorly maintained vehicles, adverse weather and road conditions, and congested traffic conditions. While existing techniques for fault tolerance from other industries such as aerospace may appear to be relevant, they may not be affordable for the automotive industry. At the same time, the exposure of a design fault such as a software bug in the automotive industry may be multiplied by the high volumes of products and the large number of passenger miles being driven by a given product line. Accordingly, new techniques for fault classification and affordable run-time fault detection, fault isolation, and fault mitigation architectures relevant to the automotive industry are needed.

Biographical sketch

Name: Tom Fuhrman
Affiliation: General Motors Research & Development
Title: Lab Group Manager, ECS Architecture Group, Electrical & Controls Integration Lab
Responsibilities: Develop next generation automotive electrical architectures, embedded software architectures, communication protocols and architectures, and safety / fault tolerant architectures
Email: thomas.e.fuhrman@gm.com
Phone: 586-986-1441