An Architectural Approach to Managing Heterogeneous Models for Automotive Control System Design

Position submitted to Auto-CPS 2010

Bruce H. Krogh[1], David Garlan[2], André Platzer[2]
[1]Dept. of Electrical and Computer Engineering
[2]School of Computer Science
Carnegie Mellon University
Pittsburgh, PA
{krogh|garlan|aplatzer}@cmu.edu
ph. 412-268-{2472|5056|1558}

Ken Butts, Prashant Ramachandra
Toyota Motor Engineering & Manufacturing
North America, Inc.
Ann Arbor, MI
{ken.butts|prashant.ramachandra}
@tema.toyota.com
ph. 734-995-2600

1. The Challenge: Heterogeneous Models

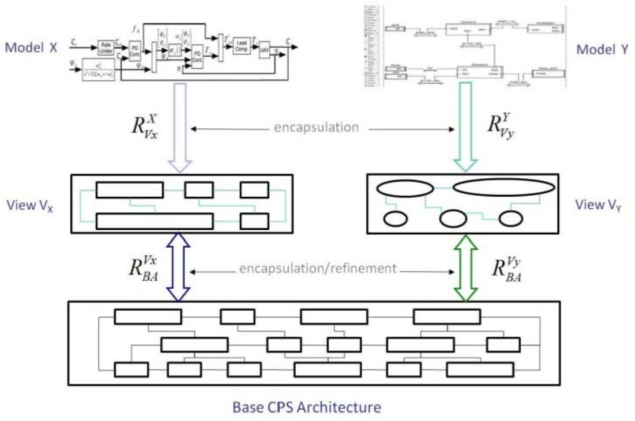
Automotive systems and other cyber-physical systems are designed and analyzed using a variety of modeling formalisms and tools. Each representation highlights certain features and occludes others to make analysis tractable and to focus on particular performance attributes. Typically a particular formalism represents either the cyber or the physical elements well, but not both. For example, differential equation models typically represent physical processes well, but do not represent naturally the details of computation, data communication, or digital control. On the other hand, discrete modeling formalisms such as Petri nets and automata are well suited for representing discrete behavior and control flow, but are not particularly useful for modeling continuous phenomena in the physical world. These different perspectives also reflect the wide range of engineering domains and technical expertise required to design and implement a system rich in both cyber and physical components. Thus, the heterogeneity of cyber-physical systems in many dimensions requires multiple heterogeneous models and formalisms to explore the complete design space.

Although the diversity of models and formalisms supports a component-based "divide and conquer" approach to cyber-physical system development, it presents a serious problem for verifying the correctness and safety of designs at the system level. Model-based design and verification of particular component properties and even global system properties is always done in the context of assumptions about system features that cannot be represented in the particular formalism being used. Each design and verification activity also leads to constraints and conditions that impinge on assumptions made in other models. The exchange of information, implications, and assumptions among the many groups of engineers performing design and verification in the development of a complex cyber-physical system is typically informal at best, and it is particularly difficult when the structure and semantics of the modeling formalisms differ significantly like in the gap between cyber and physical. Consequently, correctness of the design is inferred by a combination of engineering judgment supported by extensive testing of the final system. To achieve system-level verification in an explicit and principled way requires a framework that encompasses the complete system and is not prejudiced toward particular formalisms that capture well only cyber or only physical features.

2. An Architectural Approach

To support verification engineering at the system level, we propose the development of CPS architectural styles that provide a reference context for making meaningful associations between disparate architectural views induced by the many heterogeneous models used for system design. This architectural framework will make it possible to verify structural consistency and semantic correspondence between models and will be the context for carrying out system-level verification activities. The architectural framework would be supported by algorithms and tools that establish: consistency of the models with the overall system design; consistency of assumptions and abstractions used to construct various models; interconnectivity of verification results for different models; and coverage of the verification results from multiple models with respect to system requirements.

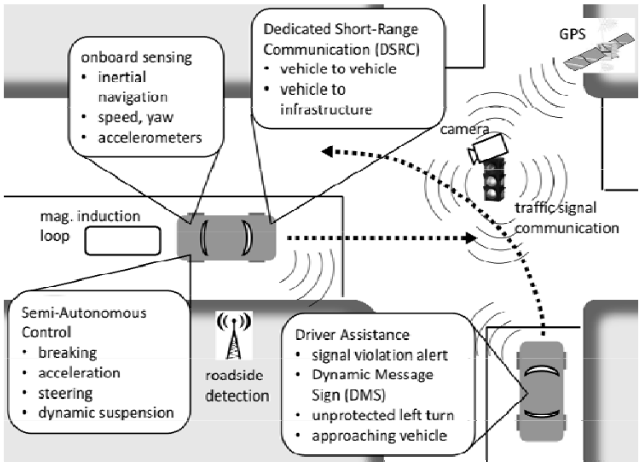
As illustrated in the figure to the right, each design model corresponds to an architectural view of a base architecture for the system. This approach does not aim to replace tools like Simulink/Stateflow that are used successfully today. The goal is to leverage existing approaches in a unified framework for system-level verification. The CPS architecture approach adopts a pragmatic perspective and combines fully formal verification with more informal engineering judgments, and exhaustive analysis with more selective simulation. The major technical challenge is how to create a sound framework and algorithms for managing relationships among heterogeneous models, formalisms, and verification results.



3. Application to Cooperative Intersection Collision Avoidance Systems (CICAS)

Many of the innovations in automobile technology are aimed toward making driving safer through advanced sensing and control, along with new methods to enhance the driver's situational awareness. Automobiles of the future will communicate with the traffic control infrastructure and other vehicles and provide drivers with safety as well as navigation advice. Ultimately, automobiles will be semi-autonomous, with the capability to enhance the driver's response to safety-critical situations and, when necessary, to take corrective action automatically to avoid collisions. We propose using cooperative intersection collision avoidance systems (CICAS), an emerging technology for semiautonomous functionality, as a case study for developing the proposed CPS architectural framework.

As illustrated in the figure below, CICAS augments traditional automobile control systems with real-time communication interfaces to the traffic and road-side infrastructure as well as to other vehicles. The full CICAS vision is a complex integration of cyber and physical



elements, including humans in the loop. Initial CICAS systems aim to reduce violations of intersection signals (stop sign and signal). Gap assist builds upon the violation concepts and technologies and adds mechanisms to assure safe distances between automobiles and intersections where other vehicles are entering the intersection at a stop sign or executing a left turn. Research has focused on cooperative systems that implement violation countermeasures through driver interfaces that provide alerts and suggestions. In current CICAS, the traffic light broadcasts its (future) state and each vehicle checks if it might possibly violate the traffic light. In future systems the intersection could provide an infrastructure-based countermeasure (e.g., adjust signal timing) resulting from the vehicle sending a warning notification to the traffic support infrastructure. In more advanced systems, the vehicle could exert some type of vehicular control countermeasure. Active control algorithms are currently being investigated to augment and correct human control actions when necessary. This provides a rich context for exploring the integration of heterogeneous models as multiple views of an underlying CPS architecture.

Author Bios

David Garlan is a professor in the School of Computer Science at Carnegie Mellon University, and is the Director of Professional Software Engineering Programs. His research interests include software architecture, pervasive computing, self-healing systems, cyber-physical systems**,** applied formal methods, and software development environments.

Bruce H. Krogh is a professor of electrical and computer engineering at Carnegie Mellon University. His current research interests include information processing and control for distributed systems, hybrid dynamic systems, and synthesis and verification of embedded control system designs.

André Platzer is an Assistant Professor in the Computer Science Department at Carnegie Mellon University. His interests include logic in computer science, hybrid systems, distributed hybrid systems, theorem proving, model checking, and verification of cyber-physical systems.

Ken Butts is an executive engineer, Powertrain and Chassis Division, Toyota Motor Engineering and Manufacturing North America in Ann Arbor, Michigan. In this position, he is investigating methods to improve in-vehicle control development productivity.

Prashant Ramachandra is a Principal Engineer, Powertrain and Chassis Division, Toyota Motor Engineering and Manufacturing North America in Ann Arbor, Michigan. He is currently working on improving efficiency of IC engine control systems development.