Building Robust Automotive Systems through Separation of Concerns

Sibin Mohan
Dept. of Computer Science
North Carolina State University
Raleigh, NC 27695
smohan@cs.ncsu.edu

Johannes Helander
Microsoft Research
1 Microsoft Way
Redmond, WA 98052
jvh@microsoft.com

Abstract

Modern automotive systems utilize a large number of embedded processors to perform computation, control, monitoring, or simply to enhance the user experience. The design and development of such systems is not always an easy task, mainly due to their distributed nature being coupled with real-time constraints. Such processes and the systems they control also interact with each other and other external systems in a complex manner – at times in ways that were not expected or modeled in advance. Yet, such interactions must not digress from the fundamental functionality, timeliness guarantees and other important properties (such as security/privacy) that the system must provide for correct and safe operation. This paper discusses techniques to reason about, analyze, test and develop such systems from the ground up while still retaining the real-time properties of the system.

I. Motivation

The development of consumer applications (for instance mobile phones) and increasing trends in automation and inter-connections among systems that surround us (such as home automation) means that embedded systems interact with each other in ways that were not previously imagined. The increased use of embedded systems in automotive systems is one such application area where various embedded tasks must be developed in a manner amenable to distributed computation, which still retaining some real-time guarantees. Such applications have a variety of "*concerns*," such as functional correctness, temporal correctness, parallelism, security, etc. each of which adds an additional layer of complexity to the development and verification of the application. Another "*concern*" is that of scalability – the application may have to execute on a wide variety of hardware configurations – from simple microcontrollers to powerful multi-core processors. Another *concern* is that of interoperability – the application may have to interact and/or exchange information with other applications. For instance, if the user brings in her mobile phone into the car – the mobile phone and the car automation system may have to recognize each other and perhaps exchange data or even control actions. This must be done in a way so as to *(a)* not detract from the correct operation and timeliness guarantees of either system, and *(b)* guarantee security/privacy issues – e.g. mobile phones of users other than the owner of the automobile must not be allowed to interact with or inject malicious data/code into the automobile system.

II. Challenges

While developing applications for automobiles that must address each one of these *concerns* is a tedious task, the task of verifying the correctness, testing and providing guarantees for each

one is quite daunting, if not impossible. Existing analysis techniques are able to extract information on some of these *concerns* but make stringent assumptions that often fail to address real-world concerns or are unable to scale to new technologies (newer architectures, for instance). Hence, some important challenges to developing comprehensive cyber-physical automotive systems are:

1. Complexity of developing distributed automotive systems with various *concerns*
2. Provide guarantees for each individual concern as well as the combined operation of all concerns that make up the complete application
3. Verify the correctness and provide for ease of testing for each concern as well as the entire application
4. Analysis techniques that can reason about combinations of *concerns*

III. Development of Distributed Cyber-Physical Systems using Separation of Concerns

We have seen that utilizing single source of information or analysis techniques restricts the information that can be obtained from such analysis – a classic example is that of timing analysis for hard real-time systems (1). These techniques are unable to adapt to newer architectural features and hence restrict the choices available to developers of such systems. Hybrid analysis techniques (2), (3), have been a good start in utilizing multiple sources of information/analysis techniques to tackle this problem.

We believe that reasoning about *concerns* of distributed cyber-physical systems is the right way to develop cyber-physical systems in the future. Existing analysis techniques coupled with newer ones developed explicitly for this purpose would ensure the success and ease of development for such systems. Previous experience has shown us how to develop functional and temporal concerns for such systems from the ground-up (4) as well as how temporal information and parallelism can be extracted from existing distributed embedded applications (3). These experiences also provide insights on how these specific *concerns* can be addressed and verified individually and composed together, without loss of guarantees, to develop comprehensive applications.

This leads us to believe that a thinking of separate concerns is useful in the development of automotive cyber-physical systems. Some examples of such "*concerns*" are:

a. correct functionality
b. temporal properties, timeliness, etc.
c. services provided to the user
d. parallelism points in the system
e. security and privacy of the system and resident/in-transit data

We intend to study and propose analysis techniques for these and other related *concerns* that will address the problems presented in section II.

IV. Security *concern* – a case study

With the proliferation of embedded systems, security of the system and privacy of the data becomes a serious issue. Hence, there is a need to implement security and/or trust policies in such systems. There exist some limitations (5) to this process of integrating security policies into such systems, such as: *(a)* guarantees for meeting all deadlines in the system, *(b)* the periodic nature of real-time embedded systems, *(c)* worst-case timing guarantees for all tasks that must execute in the system, *(d)* limitations of processing power and memory in such systems, *(e)* level or comprehensiveness of security that can be achieved due to the above limitations.

Hence, security is a *concern* that requires separate analysis and development path in automotive embedded systems. But, as we see above, this concern must also operate within the strict limits of others in the system – hence there is a need to provide guarantees for not just the security part, but the other parts as well as the collective.

V. Author Biographies

Johannes Helander is a researcher at Microsoft with an interest in distributed embedded systems, scalable real-time, consumer-centric security, and context history based prediction and modeling. He has a long track record on real software projects, including the first Embedded XML Web Services, multiple operating systems, real-time scheduling, a remote shell for Vista, etc.

Sibin Mohan is a doctoral candidate at North Carolina State University where he has been working in the field of real-time and embedded systems for the last six years. He has been actively involved in the development of analysis techniques for modern processor architectures and embedded software, particularly with a focus on hybrid analysis techniques that use multiple sources of information, to characterize the true nature of such systems.

Bibliography

1. *The Worst-Case Execution Time Problem - Overview of Methods and Survey of Tools.* **R. Wilhelm, J. Engblohm, et. al.** s.l. : ACM, 2007, Vol. Transactions on Embedded Computing Systems.

2. *Hybrid Timing Analysis of Modern Processor Pipelines via Hardware/Software Interactions.* **S. Mohan, F. Mueller.** St. Louis : IEEE, 2008. Real-Time and Embedded Technology and Applications Symposium.

3. *Temporal Analysis for Adapting Concurrent Applications to Embedded Systems.* **S. Mohan, J. Helander.** s.l. : Microsoft Research, 2008. MSR-TR-2008-37.

4. *Adapting Futures: Scalability for Real-World Computing.* **Johannes Helander, Risto Serg, Margus Veanes, Pritam Roy.** s.l. : IEEE, 2007. Real-Time Systems Symposium.

5. *Worst-Case Execution Time Analysis of Security Policies for Deeply Embedded Real-Time Systems.* **Mohan, Sibin.** 2007. PhD Students Forum on Deeply Embedded Real-Time Computing, IEEE RTSS.

6. *Adapting the Auto to a New Tune.* **Helander, J., Preden, J.** Rio de Janeiro : IEEE, 2006. RTSS 2006 - Workshop on Models and Analysis for Automotive Systems.