Networked Embedded Systems

Pieter J. Mosterman

I. Position Statement

A. Background

After the tremendous proliferation of embedded computing power, as witnessed over the last several decades, a new wave of technological achievement in the form of ubiquitous communication can now be observed. Where initially wired networks were the main driver, more recently communication modalities such as software defined radio and wireless networks have added to the impetus of the field.

For example, in the automotive field, the use of a controller area network (CAN) has become essential to support the variety of electronics systems that modern-day vehicles are equipped with. Figure 1 illustrates the use of networked communication for a power window controller. The embedded controller to command the window up and down movement is located close to the lift mechanism and the dc motor that actuates the lift, as well as the current sensor that measures the current drawn by the dc motor and that is used for feedback control purposes.

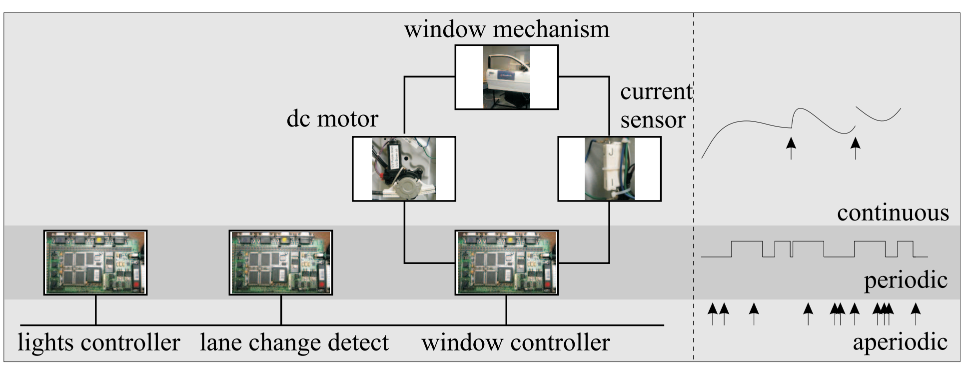

Fig. 1. A model of a power window control system.

The supervisory commands to initiate window movement are issued by the driver and passenger by pressing buttons located elsewhere in the vehicle. These buttons are connected to a microprocessor that translates the voltages corresponding to button up and down actions into

P.J. Mosterman is with The MathWorks, Inc., Natick, MA, USA. E-mail: pieter.mosterman@mathworks.com.

messages to be sent to the power window controller over the network. In the meantime, the network is shared by other systems such as a lane departure warning system and a head light position control system.

The architecture in Fig. 1 contains the basic elements of typical embedded systems, a physical plant, a sensor, an actuator, and a controller. Different architectures can be de developed around these, for example the controller may be connected to the sensors and actuators by means of a network or only a physical system and sensors may be present.

B. Problem Statement

The design of such networked embedded systems has presented challenges that are not inherently addressed by the design methods for embedded control systems. In particular, networked communication engenders idiosnycratic behavior that is not amenable to synthesis methods.

An important aspect of communication is uncertainty at many levels:

- Uncertainty in temporal behavior, for example in terms of jitter in latency.
- Uncertainty in message ordering, for example because of dynamic routing of data.
- Uncertainty in arrived data, for example because of intermittent failures leading to error rates.

In addition, networks consist of a huge discrete state space and discrete event space, which leads to complications, for example because of rare events. Moreover, the dynamic nature prohibits the use of proven methods such as static rate-monotonic scheduling and the time-triggered architecture in many applications.

Computational simulation is often the only technology that allows analysis and insight into a networked embedded system design, especially since closed-form solutions for anything but trivial network configurations and protocols do not exist. Computational simulation of network communication can be of high computational complexity, though, and dedicated solvers for the different parts and problems to be addressed are required. For example, discrete event solvers can be used to simulate the behavior of a network topology and routing algorithm, with respect to the source to destination specifics of a packet. The ambient network traffic, however, may be better represented as a differential equation model to serve as an abstraction of the discrete events, and to manage the volume of these events. This presents a need to integrate the different computational solutions to handling time-driven and event-driven execution.

The behavior of a complete network at various levels of detail requires models of a size that can only be obtained by the use of domain specific languages and automation of the model design.

Furthermore, the design for safety requires consideration beyond the traditional approaches such as, for example, a DO-178B certification, and novel algorithms and methodologies are needed. In particular, the design complexity of networked systems makes the correct design even in adverse conditions difficult to achieve. The dynamic character of networked systems renders coverage methods such as the Modified Condition/Decision Coverage (MC/DC) of DO-178B not applicable. Furthermore, the wireless receptiveness to unidentified signals makes wireless networked systems susceptible to unauthorized use.

To succesfully tackle the design complexity, network diagnostic approaches are required and need to be integrated with the design of a networked embedded system. Here, it is important to characterize the particularities of the networked embedded system, as different requirements may hold for each individual system.

II. A Roadmap

To handle the rapidly emerging complexity of networked embedded systems, novel algorithms, methodologies, and technologies need to be developed.

A. Algorithms

A number of algorithms need to be developed, in particular, to tackle the size of the problems posed by networked embedded systems:

- FDI algorithms, in particular concentrating on processing huge amounts of data.
- Nondeterministic simulation algorithms.
- Model checking algorithms that are effective for a huge set of states and decision points.

B. Methodologies

The domain of networked embedded systems presents a set of methodological challenges to address its extensive and uncertain nature:

- Domain specific languages need to be designed to allow efficient modeling of systems at the scale presented. This requires furthering developments in language design such as semantic anchoring.

- Because of the highly dynamic and extensive nature, automated modeling such as generative approaches are essential to effectively design models.
- The inherent uncertainties call for model-based diagnosis methodologies that can handle the intricacies of networked embedded systems, such as the intermittent character of errors.

C. Technologies

Technologies necessary to negotiate the complexity of networked embedded systems are to a large extent in the field of computational simulation. They include:

- Efficient integration of event-driven and time-driven execution engines.
- Facilitating accelerated simulation, for example by hardware emulation.
- Support for integration of C code to simulate network protocols implemented as such.

PLACE PHOTO HERE

Pieter J. Mosterman is a senior research scientist at The MathWorks, Inc. in Natick, MA. Before, he held a research position at the German Aerospace Center (DLR) in Oberpfaffenhofen. He has a Ph.D. degree in Electrical and Computer Engineering from Vanderbilt University in Nashville, TN, and a M.Sc. degree in Electrical Engineering from the University of Twente, Netherlands. His primary research interests are in Computer Automated Multi-Paradigm Modeling (CAMPaM) with principal applications in training systems and fault detection, isolation, and reconfiguration. For this, he designed several modeling and simulation environments such as the Electronics Laboratory Simulator (nominated for The Computerworld Smithsonian Award) and HyBrSim. He was awarded the IMechE Donald Julius Groen Prize for his paper "HyBrSim-A Modeling and Simulation Environment for Hybrid Bond Graphs".

Dr. Mosterman is program chair of the Model-Based Design for Embedded Systems track at the Design Automation and Test in Europe conference and exhibition and of the International Conference on High Level Simulation Languages and Applications in 2007. He is invited session chair for the 2006 IEEE International Symposium on Computer Aided Control System Design (CACSD). He co-chaired the 14th International Workshop on Principles of Diagnosis (2003) and the annual International Bellairs CAMPaM Workshop since 2004. He is currently chair of the IEEE CSS Action Group on Hybrid Dynamic Systems for CACSD, Editor-in-Chief of Simulation: Transactions of The Society for Modeling and Simulation International for the Methodology section, and Associate Editor of IEEE Transactions on Control System Technology and of Applied Intelligence. He was Mechatronics Area Editor of Simulation: Transactions of The Society for Modeling and Simulation International and Guest Editor of special issues of ACM Transactions on Modeling and Computer Simulation and IEEE Transactions on Control System Technology on the topic of CAMPaM.