# Uncertainty and Dependability in CPS

Daniel Mossé
Computer Science Department
University of Pittsburgh
*mosse@cs.pitt.edu*

Hakan Aydin
Computer Science Department
George Mason University
*aydin@cs.gmu.edu*

Much work has been done in embedded systems, real-time systems, control systems and wireless sensor networks. Cyber-Physical Systems emerges at the integration of these fields, to enable control of physical systems at a scale that has not yet been possible. CPSs have certain desirable properties, including dynamic behavior, scalability, self-* behavior, and guaranteed behavior (as defined by standards). CPS Safety and security, although beyond the scope of this position paper, is also a must since they interact with and control the physical world with humans in it; regulatory requirements must be put in place.

Of particular interest for this workshop is a vehicular network, encompassing the vehicle-to-vehicle, vehicle-to-infrastructure or infrastructure-to-vehicle communication. It is envisioned that thousands of sensors along highways will communicating with each other and each car as necessary, alerting, say, about a possible congestion down the road to enable re-routing. Also, quickly developing weather conditions (e.g., severe thunderstorms or a patch of black ice) should also be conveyed. The goals of such systems are not only reduction of congestion, faster throughput, but also a drastic reduction in the number of accidents and fatalities in roadways (there are more than 40,000 deaths yearly in US roads).

In this example, it becomes clear that **context-awareness** is an important issue, which is not present in most real-time or embedded systems. For example, driving in downtown areas is different from traveling in inter-state highways, and this information may enable the system to better use its resources and interpret the data more accurately. An instance of such information is a certain vehicle density, which may imply congestion in highways but maybe normal in within city limits. In the latter case, re-computing a path may be unnecessary (and may be misleading for other vehicles).

Issues that have been partially solved in the individual arenas become more overwhelming due to the intersection of the fields that have become CPSs. In particular, we believe that the **identification and treatment of faults** in the physical systems, as well as the **treatment of uncertainty** will be at the core of technologies that must be developed in order to achieve useful CPSs. This is clear from the use of CPSs for human and physical worlds, due to the intense interactions between engineering, information technology, and computer science.

Goals of CPSs include 24/7 reliability, with 100% availability, and 100% connectivity, in addition to the real-time response (deadlines defined by the system integrators). Other goals invisioned, but beyond the scope of this paper, is the ability to store information for online (stream) analysis

as well as post-mortem analysis.

In particular for highway transportation systems, cars will be moving at faster speeds with increased density, and therefore decreasing uncertainty and failures in the systems is a necessary condition for deployment. Whereas individual cars can adapt to problems in the physical world (e.g., shock absorbers), a vehicular network with enough sensors will be able to detect, identify, and report on problems of roads and traffic. This information can be immediately used by other cars in a convoy, can be used by vehicles planning to use a specific stretch of road, and can be used by maintenance crews.

The sensors distributed along the highways in addition to sensors located in cars will enable higher predictability of the environment where the cars are traveling, and will enable more streamlined algorithms, which will present faster response times and higher dependability. The environment will be more predictable due to the distribution of all data pertaining to a particular section/segment of a vehicle convoy. For that, **new scatter-gather mobility algorithms** will be needed, taking into account mobility of both the data collectors, potentially the routing nodes, as well as the destination nodes. We envision each node (car) being able to divulge its location and direction (in a secure manner, which is beyond the scope of this paper), and collect *relevant* information for its own purposes (e.g., a publisher-subscriber protocol, but the exact mechanism will depend on the application and its constraints).

In the same vein, that is, to better predict the environment, we need **better fault models**. By and large, current models have unrealistic assumptions, such as independence/uncorrelated causes of faults and unrealistic fault containment assumption (i.e., a transient fault happens in one or two components, it is contained, and the system "recovers nicely"). Energy efficiency of the electronics subsystem may be a concern in many CPS applications, but perhaps not the best example of major concern in transportation networks, due to its small contribution in the overall energy consumption. However, controlling peak power and aggregate power deserves attention due to the physical constraints of the environment that may increase the number and type of faults in the system.

Even in already deployed systems, such as satellites, there are few studies that enable developers to actually design algorithm that sample data and command actuators at the proper rate. Most practitioners, because they have no tools to deal with the unpredictability of the environment where their deployed systems operate, use a much higher rate than necessary. Dealing with **development of more sophisticated fault and recovery models** is essential for the deployment of CPS. In our current work, we are developing models of correlated interference in wireless networks, which is directly applicable to vehicular networks such as in smart highway systems.

Another issue that causes unpredictability is the ability to predict the behavior of software itself. Much work has been done in analysis of code for predicting worst-case execution time behaviors of software modules, and much work still remains, specially when considering advanced micro-architectures and the physical nature of CPS applications. Transient overloads may happen if the specifications are not accurate (and they are often not), and software is either designed to execute in very-low-utilization hardware or it has to adapt to overload situations. Given the physical nature of these CPS applications and the critical nature of some information, some issues may pose serious challenges, such as **determining priorities** perhaps through context-awareness and/or through criticality-based capabilities, and **containment of the effect of propagated timing errors**.

Although more predictable environments will lead to different (more streamlined) solutions applicable to CPSs, there is still a need for algorithms to deal with unpredictable situations. Two solutions emerge in such situations, namely the use of control theory and the use of fault tolerance. Control theory enables dynamic behavior in the systems, allowing for different input values to be treated according to the requirements of the application[1]. Fault tolerance enables application-oriented fault tolerance techniques should be developed for CPS, based on the needs and requirements of the applications. There may be some common techniques that can be used across the board, but **application-specific fault-tolerant integrated approaches** and **probabilistic approaches** that enable recovery even when it is not 100% guaranteed may be the most indicated.

These issues are specially true when we consider the interaction among smaller (sub)systems; such interactions are going from occasional to intensewhen CPSs take hold and start permeating modern societies. Uncertainty either in the form of faults or misspecifications of input data/parameters will require more robust designs to withstand more stringent CPS requirements.

[1]Other CPS researchers have been studying the needs for integrated control, cascaded control, and other forms of advanced control.