Guest Editors’ Introduction: Special Section on Science of Design for Safety Critical Systems

Sandeep K. Shukla, Senior Member, IEEE, and Jean-Pierre Talpin

The idea of this special section dawned on us during various discussions on the recent trends in computer system design throughout the 2008-2009 academic year when one of the editors spent a sabbatical year at INRIA hosted by the other editor. Cyber Physical System (CPS) was the most recent buzz word replacing the ‘hybrid systems’, and the ‘Science of Design’ (SoD) was the other buzz word on its way out to the perished land of unfashionable terminologies. In the realm of cyber physical systems there had been a lot of foundational developments under the guise of hybrid systems since the mid-nineties. The science of design, another terminology coined at the US National Science Foundation (NSF) somehow remained within the traditional programming language design community - and did not get a wider acceptance. Within cyber physical systems, however, there are special classes of systems which are safety critical such as avionics, automotive, space mission systems, missile control, smart grid, industrial process control or SCADA etc. This is the class of systems that interested us the most. We realized that since many of these are domain specific, the engineers who design them are not necessarily computer scientists, and they could be from any other engineering field such as aerospace, electrical, mechanical, power systems, control and so on. The question that naturally comes up as to how they collaborate with the computer scientists who develop the foundations of system design - especially systems that have digital control with analog environments - which are very common in most safety-critical systems. So we appropriated the term “Science of Design” and termed the foundational aspect of such design as the science, and the domain specific engineering as the application of science.

Next we talked to some of our colleagues - who were involved in designs of unmanned vehicle systems. It was hoped that a strong contribution to this special issue could be obtained from such colleagues. One would assume that major requirements on the cyber components of such unmanned vehicles must be low power consumption, small form-factor, reliability, verifiability, etc. A paper describing how these requirements interplay with their system design approach, and how the physical system design influences the requirements of the cyber parts and the control algorithms, would have been a great contribution. It turned out that no integrated approach was followed by these designers. Intel x86 processors were purchased (a power hungry one), and an off-the-shelf real-time Linux was used as an execution environment, while MATLAB based control algorithm models were provided to C programmers to create the software. Disappointed by the lack of an integration of science of design into the engineering, we spoke to a number of researchers at a number of defense labs, and contractors, and heard very similar ‘separation of concern’ stories.

The safety-critical systems that we were concerned with had strong coupling and interactions between one or more physical environments and a number of cyber or computing components. Evolution of the physical environments over time and space, described by their trajectories in continuous state spaces, are modeled by parameters whose evolution is best captured with continuous dynamical systems. Some of these parameters are controllable by the cyber components, and some evolve based on the dynamics of the physical worlds. The cyber components usually sample some or all of these parameters based on Nyquist criteria, and calculate robust feedback control over controllable parameters. This is often called digital control because the continuously varying parameters are sampled and discretized, while the control algorithms process the information to create control actions in the form of discrete signals. The feedback control affects the trajectory in the physical state space. Specifically, robust control algorithms make sure that the planned trajectories are tracked by the physical system as accurately as possible, regardless of various uncertainties and exogenous disturbances.

Before digital computers were cost effective, much of the control in many such systems were analog and mechanical in nature. This meant that the control components and the physical world together formed a complex dynamical system. The analysis of such system was within the realm of continuous mathematics. However, CPS systems have a dichotomy (between the continuous and the discrete) which poses challenges to their algorithmic development, proof of stability and robustness, etc. On the other hand, there is a tremendous opportunity due to the exponential effects of Moore’s law, making computing exponentially faster, cheaper, and smaller in size. However, the implementation of the control algorithms in hardware and/or software is often distributed in nature (digital signal processing, control computation for a large number of controllable parameters, and real time requirements may necessitate the use of a large number of processors, e.g., a modern automotive vehicle has more than 80 microcontrollers and processors). To make such hardware/software optimized and correct, one has to take care of concurrency issues, timing issues, power vs. performance trade-offs, and most importantly eliminate any redundant sampling or computation. Unfortunately, since such systems are often safety-critical (avionics, automotive,
power grid protection systems) it is very important that they are functionally correct, often at the cost of optimization opportunities. However, correctness and optimality can coexist if proper design methodologies are developed and applied.

With this special issue, we wanted to consider this dilemma and the opportunities to apply advances in computing to control the problems of real physical engineering artifacts. So we solicited papers from all areas of engineering to describe case studies along with methodologies of any integrated design approach for designing safety-critical cyber physical systems such as automotive, avionics, industrial process control, etc. The agenda was to showcase how the cyber parts of such systems are designed along with the physical parts in an interdependent fashion. Most interestingly, after almost six months of advertisement for the special issue’s call for papers, only five papers were submitted, and only one was from the US. This is indicative of the fact that many university laboratories in robotics, unmanned aerial vehicle (UAV) design, aerospace, power systems, and mechanical engineering are not exploiting the research in computer engineering, especially on topics such as resource optimal computing (low power, low area, low latency), novel sensor capabilities and system architectures, and the overall modeling and code synthesis capabilities. Individual advances in hardware and software design are often not being integrated into the research done in these engineering labs. More problematic is that the overall system modeling and design sciences for computing, and design sciences for physical systems are not meeting each other head on. The physical systems designers are using MATLAB, LabView, etc., to model their control algorithms and simulation environments. While these tools are essential for simulation based validation of algorithms, they do not provide any design space exploration, hardware/software codesign, exploration of distributed vs. centralized computation, network latency models, etc. The choice of processors, execution environments, and software design methodology used in these labs is therefore not at the cutting edge of research in computer science/engineering. This also proliferates into industry as students from these labs go into industry, having been used to developing the physical and the cyber parts in isolation.

Eventually, out of five papers, we were able to select three papers which do not necessarily fulfill the vision we had regarding creation of a collection of reference papers within one special section that future designers can look at and learn collaboration models, as well as examples of integrated approach. Among the three papers we selected, the paper by Pedro Sánchez, Diego Alonso, Francisca Rosique, Bárbara Álvarez, and Juan A. Pastor “Introducing Safety Requirements Traceability Support in Model-Driven Development of Robotic Applications”, describes a methodology they developed in the process of designing of tele-operated services robots - a methodology that allows them to automatically trace back to initial safety requirements every time a change in the system is made. This traceability allows them to attribute any change to the initial requirements, as well as help mitigating any risk associated with such changes. Such methodology should be a part of our arsenal while designing safety-critical systems and hence the paper is important. The second paper, by Irem Y. Tumer, and Carol S. Smidts, titled “Integrated Design-Stage Failure Analysis of Software-Driven Hardware Systems” also have a similar outlook. Failure source identification and tracking it back to the source from the point of failure based on system level model is a required approach with the current trend in model based design of safety-critical systems. The third paper, by Rachid Hadjidj and Hanifa Boucheneb, “Efficient Reachability Analysis for Time Petri Nets”, is actually not so much on the integrated approach to system design, but actually on a specific reachability problem in timed systems.

We hope that in the future more integrated approaches and collaborative methodologies for engineering safety-critical CPS by virtue of strongly leveraging the science of design for such systems will develop and our special section effort will be one of the instigators for such approaches and collaboration.

Sandeep K. Shukla
Jean-Pierre Talpin
Guest Editors

Sandeep K. Shukla (SM’02, M’99) received the bachelor’s degree in 1991 from Jadavpur University, Calcutta, and the master’s and PhD degrees in computer science in 1995 and 1997, respectively, from the State University of New York at Albany. He is an associate professor of computer engineering at Virginia Polytechnic and State University in Blacksburg (Virginia Tech), where he has been a faculty member since 2002. He is also a founder and director of the Center for Embedded Systems for Critical Applications (CESCA) and director of the FERMAT research lab. He has published more than 150 articles in journals, books, and conference proceedings, and has published eight books. He was awarded the PECASE (Presidential Early Career Award for Scientists and Engineers) award for his research in design automation for embedded systems design, which in particular focuses on system level design languages, formal methods, formal specification languages, probabilistic modeling and model checking, dynamic power management, application of stochastic models and model analysis tools for fault-tolerant nano-scale system design, reliability measurement of fault-tolerant nano-systems, and embedded software engineering. Professor Shukla was elected a College of Engineering Faculty fellow at Virginia Tech in 2004. He is a distinguished visitor of the IEEE Computer Society, a distinguished speaker of the ACM, and a senior member of the IEEE and ACM. He worked at GTE labs and Intel Corporation between 1997 and 2001. He was a researcher at the Center for Embedded Computer Systems at the University of California at Irvine. In 2007, Professor Shukla received a Distinguished Alumni award from the State University of New York at Albany for Excellence in Science and Technology. In 2008, he received the Friedrich Wilhelm Bessel Research Award from the Humboldt Foundation in Germany.

Jean-Pierre Talpin received the PhD degree in 1993 from the University of Paris VI Pierre et Marie Curie. He is the director of research at INRIA Rennes-Bretagne-Atlantique. He worked at the European Computer-Industry Research Centre in Munich as a research associate prior to joining INRIA in 1995. He is the scientific leader of the Espresso project-team whose main area of research is embedded software design. He is also the coordinator of the ANR project OpenEmbeDD developing an open-source, model-driven, embedded system design platform. He served eight years as an elected member of INRIA’s evaluation commission. In 2005, he received the ACM SIGPLAN Award for the most influential POPL paper for his article with Mads Tofte.