SCANNING THE ISSUE

Design Automation for Cyber–Physical Systems

By QI ZHU, *Member IEEE*
Guest Editor

ALBERTO SANGIOVANNI-VINCENTELLI, *Fellow IEEE*
Guest Editor

SHIYAN HU, *Senior Member IEEE*
Guest Editor

XIN LI, *Fellow IEEE*
Guest Editor

I. INTRODUCTION

Cyber–physical systems (CPSs) are characterized by the seamless integration and close interaction of cyber components (e.g., sensors, computation nodes, communication networks) and physical processes (e.g., mechanical devices, physical environment, humans). The cyber components monitor, analyze, and control the physical processes, and react to their changes through feedback loops. A classic example of CPSs is autonomous vehicles. These vehicles collect information of the surrounding physical environment via heterogeneous sensors such as cameras, radar, and LIDAR; process and analyze the multi-modal information at real time with advanced computing devices such as GPUs, application-specific SoCs and multicore CPUs; automatically make planning and control decisions; and continuously actuate the corresponding mechanical components. The cyber components of autonomous vehicles are much more intelligent and complex than those of traditional vehicles, and interact more directly and closely with the physical environment.

The papers in this special issue discuss challenges and present promising solutions in the modeling, simulation, synthesis, validation, and verification of cyber–physical systems.

The advancement of CPSs, such as autonomous vehicles, industrial robots, wearable devices, smart buildings, and smart infrastructures, has shown great promises. However, the design and operation of CPSs face serious challenges stemming from system scale and functional complexity that is constantly growing; the adoption of distributed and networked architectural platforms; the close interaction with dynamic physical environment and human activities; and the stringent and diverse requirements on performance, safety, security, fault tolerance, extensibility, energy consumption, etc. For example, the development of new features in vehicles has led to a drastic growth of functional complexity, with software code increasing from around 1 million lines in 2000 to 100 million post-2010 [1]. The number of electronic control units (ECUs) has increased from under 50 to more than 100 [2], and cutting-edge GPUs have been developed for autonomous driving functions [3]. Automotive engineers now have to explore a much larger design space, address more functional and nonfunctional requirements, and validate the designs under more dynamic and uncertain scenarios for enabling autonomy.

Unfortunately, many key processes in current CPS design practices are *ad hoc* and manual, and are incapable of coping with the above challenges. We believe that now is a critical time for the CPS community, both academic researchers and industrial developers, to aggressively pursue design automation to develop a

Digital Object Identifier 10.1109/JPROC.2018.2865229

new set of methodologies, algorithms, and tools for improving CPS design quality, scalability, reliability, and productivity, and most importantly, to facilitate a bold move from *ad hoc* CPS design toward systematic and formal techniques. This special issue provides a comprehensive coverage of the broad area of design automation for CPSs.

II. OVERVIEW OF THE SPECIAL ISSUE

In this issue, leading research groups discuss challenges and present promising solutions in modeling, simulation, synthesis, validation, and verification of CPSs. They demonstrate the importance of these design automation techniques in a variety of application domains, including automotive and transportation systems, buildings, biochips, and mobile applications.

The first group of papers present methodologies and frameworks for addressing common design challenges across various CPS domains.

In particular, one major challenge to CPS design and analysis is the intrinsic heterogeneity of those systems. Today CPSs are often designed by leveraging existing solutions and by adding cyber components to an existing physical system, thus decomposing the design into two separate phases. In the paper "Codesign methodologies and tools for cyber–physical systems," Zhu and Sangiovanni-Vincentelli propose to codesign cyber and physical components of the system, i.e., to model, simulate, synthesize, and validate the sensing, control, computation, and communication algorithms; the software and hardware implementation platform; the mechanical components and processes; and the surrounding physical environment and human activities in a *holistic* environment. They present a number of codesign approaches, such as the Metronomy cosimulation framework [4] that integrates functional modeling in Ptolemy [5] and architectural modeling in Metro II [6], several cross-layer cosynthesis methods that are based on the exploration of timing contracts, and a collaborative functional verification and platform synthesis framework. The authors also discuss open challenges in CPS codesign and possible future directions for addressing them.

In the paper "Model and tool integration platforms for cyber–physical system design," Sztipanovits *et al.* address the heterogeneity in CPS model libraries and design tools with two *integration platforms*. The Model Integration Platform leverages the General Modeling Environment (GME) [7], the model integration language CyPhyML [8], and the formal specification language FORMULA 2.0 [9] to represent components, design spaces and designs, cross-domain interactions, composition constraints, data model interfaces, models of engineering process, and model transformation. It enables precise representation of semantic interfaces among modeling domains. The Tool Integration Platform features the DESERT tool [10] for automated design space exploration, and integrates methods for formal verification, reliability analysis, and uncertainty quantification.

The paper "A component architecture for the Internet of Things" addresses heterogeneity for those CPSs that leverage internet technology for interactions between the cyber world and the physical world. It presents a design pattern called *accessors* to serve as a proxy for any "Thing" or service that may be local or remote (analogous to the role of a web browser proxy in representing a remote service). The accessors enable the integration of heterogeneous and distributed components for Internet-of-Things (IoT) applications. They are defined with an adapted *actor* model in Ptolemy, and interact with each other based on a timed discrete-event model of computation. Brooks *et al.* also present CapeCode, a design environment in Ptolemy that can be used to compose accessors and facilitate the modeling, debugging, and design space exploration of various IoT applications.

Another major challenge in CPS design is to manage the continuous change and evolution of the systems and their operation environment. In the paper "Platform-centric self-awareness as a key enabler for controlling changes in CPS," Möstl *et al.* define *self-awareness* as a system's ability to recognize its own state, possible actions, and the result of these actions on itself and its environment. They present two frameworks, controlling concurrent change (CCC) and information processing factory (IPF), for building self-aware CPSs that have the capabilities of self-modeling, self-configuration, and monitoring. In particular, CCC addresses in-field changes (both at-runtime and at-down-time) in automotive systems, with a focus on ensuring system safety and availability, while IPF focuses more on runtime feedback control for MPSoC-based CPSs.

The second group of papers present techniques for formalizing the modeling, synthesis, and verification of CPSs.

In the paper "Building a hybrid systems modeler on synchronous languages principles," Benveniste *et al.* present a modeling language for hybrid systems that is built on the synchronous language principles and compilation techniques. The proposed language combines traditional synchronous language constructs with ordinary differential equations (ODEs) and zero-crossing events, to support the modeling of both discrete time and continuous time in hybrid systems. It also provides a runtime that delegates the model execution in continuous-time phases to an off-the-shelf numerical solver. The approach has been implemented and evaluated in the academic tool Zelus and its industrial sister SCADE Hybrid.

The paper "Real-time decision policies with predictable performance" introduces the usage of declarative streaming languages, in particular StreamQRE, for modeling and analyzing real-time streaming applications that process sequences of data items under constraints on memory, processing time, and energy consumption. The approach is based on the formalism of quantitative regular expressions (QREs), and its evaluation algorithm can guarantee constant cost (memory, runtime, energy) per data item and calculate the upper bounds on the per-item cost. The paper uses cardiac arrhythmia monitoring as the driven application to demonstrate the ideas of StreamQRE.

The paper "Layering assume-guarantee contracts for hierarchical system design" presents a method to algorithmically decompose system-level temporal logic specifications for CPSs into lower level specifications for individual subsystems (components), in the form of assume-guarantee contracts. The automated process ensures that the generated component specifications are implementable and simpler for further development, based on a formalized definition of realizability and a parametric analysis approach for finding what variables can be hidden while preserving realizability and ensuring correct composition. The method also includes an algorithm to convert the generated specifications from binary decision diagrams to more readable formulas over integer variables.

In the paper "SMC: Satisfiability modulo convex programming," Shoukry *et al.* present a satisfiability modulo convex programming (SMC) framework that enables efficient reasoning of Boolean and convex constraints at the same time. Such capability is particularly important for CPS design and verification, where the system heterogeneity often brings both types of constraints. The proposed framework leverages a lazy combination of satisfiability (SAT) solving and convex programming, to provide a satisfying assignment or determine that the problem is unsatisfiable. Through case studies in spacecraft docking mission control, robotic motion planning, and secure state estimation, the authors demonstrate that the framework outperforms state-of-the-art satisfiability modulo theory (SMT) and mixed integer convex programming (MICP) solvers on problems with both complex Boolean structures and large number of real variables.

The third group of papers focus on specific application domains and present corresponding methodologies and tools.

In the paper "Design automation for smart building systems," Jia *et al.* present a platform-based design flow for smart buildings. The proposed flow maps high-level specifications of desired building applications to their physical implementations based on the platform-based design (PBD) paradigm. Three intermediate design platforms are defined for smart buildings, namely the virtual device platform (including high-level functions such as a virtual occupancy sensor), the module platform (including basic functions such as a sensing module and a data analytics module that distills the occupancy information from the sensor), and the implementation platform (including concrete software and hardware implementations such as building operation systems APIs and program code). Design space exploration is carried out when a design at higher level platform is mapped onto (refined into) a design at lower level platform.

The paper "Tools and methodologies for autonomous driving systems" introduces a standard reference architecture for connected and autonomous vehicles (CAVs), and presents a set of methodologies and tools for the modeling, design, development, and testing of CAV systems. The reference architecture includes sensors, V2X (e.g., vehicle-to-infrastructure, vehicle-to-vehicle, vehicle-to-pedestrian) communication interfaces, perception, planning and behavior modules, vehicle by-wire controls, embedded computing platform, etc. The tools include SysWeaver for model-based design, integration, and analysis of software architecture; SysAnalyzer for schedulability analysis; TROCS and AutoSim for hybrid emulation and simulation at system level and at application level, respectively; and a runtime diagnostics service for on-road tests.

In the paper "Cyber–physical digital-microfluidic biochips: Bridging the gap between microfluidics and microbiology," Ibrahim and Chakrabarty introduce a new synthesis methodology for digital-microfluidic biochips. The approach leverages on-chip integration of sensing systems, and uses realistic models of biomolecular protocols to address real-world microbiology applications through cyber–physical adaptation. More specifically, the paper presents a design and optimization framework to control multiple sample pathways in quantitative-analysis protocols such as the gene-expression analysis, a synthesis method for large-scale protocols with temporal constraints such as the real-time epigenetic analysis, and a synthesis method for protocols with indexed samples such as the type-driven single-cell analysis.

The paper "Oasis: A mobile cyber–physical system for accessible location exploration" considers mobile devices connected through wireless communication as a mobile CPS. It brings up the emerging concept of improving mobile user experience by appropriately modeling human mentality, wireless signal coverage, and their interplay. Based upon a real-world case study, Cheng *et al.* carefully analyze "null zones" and "hot zones," where data rate is not sufficiently high to facilitate delay-sensitive applications, and then develop a mobile CPS platform called Oasis, for guiding users to leave those zones and move to nearby locations with better mobile experience. The modeling of user satisfaction and user willingness to take a route is particularly interesting and important for CPS applications with strong human interaction.

III. SYSTEM DESIGN AUTOMATION FOR FUTURE CPS

Historically, electronic design automation (EDA) techniques have propelled the advancement of integrated circuits, tackling the ever-increasing circuit complexity with a wealth of automation, optimization, and validation tools. We believe that for enabling future advancements and innovations of CPSs, developing *system design automation* techniques will be similarly essential. The intrinsic heterogeneity of CPSs, from the differences between various cyber and physical components to the unique characteristics across different CPS domains, will likely make such development more challenging, but there are promising directions, as presented in this special issue.

We trust that the papers in this issue provide a broad and in-depth coverage of the needs, challenges, and solutions in design automation for CPSs; and we hope that they can stimulate future research and development to address the open challenges ahead.

REFERENCES

[1] *Connected Car, Automotive Value Chain Unbound*, McKinsey Company, New York, NY, USA, Sep. 2014.

[2] *Consolidation in Vehicle Electronic Architectures*, Roland Berger Strategy Consultants, Munich, Germany, Jul. 2015.

[3] *NVIDIA DRIVE Platform*. Accessed: Aug. 2018. [Online]. Available: https://www.nvidia.com/en-us/self-driving-cars/drive-platform/

[4] L. Guo, Q. Zhu, P. Nuzzo, R. Passerone, A. Sangiovanni-Vincentelli, and E. A. Lee, "Metronomy: A function-architecture co-simulation framework for timing verification of cyber-physical systems," in *Proc. Int. Conf. Hardw./Softw. Codesign Syst. Synth. (CODES+ISSS)*, Oct. 2014, pp. 1–10.

[5] J. Eker *et al.*, "Taming heterogeneity—The Ptolemy approach," *Proc. IEEE*, vol. 91, no. 1, pp. 127–144, Jan. 2003.

[6] F. Balarin *et al.*, "Platform-based design and frameworks: Metropolis and metro II," in *Model-Based Design of Heterogeneous Embedded Systems*, G. Nicolescu and P. Mosterman, Eds. Boca Raton, FL, USA: CRC Press, 2009.

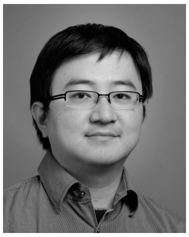
[7] J. Davis, "GME: The generic modeling environment," in *Proc. Companion 18th Annu. ACM SIGPLAN Conf. Object-Oriented Program., Syst., Lang., Appl. (OOPSLA)* 2003, pp. 82–83, doi: 10.1145/949344.949360.

[8] Z. Lattmann *et al.*, "Towards automated evaluation of vehicle dynamics in system-level designs," in *Proc. ASME Int. Design Eng. Tech. Conf. Comput. Inf. Eng. Conf. (IDETC/CIE)*, Chicago, IL, USA, Aug. 2012.

[9] *FORMULA 2.0: A Language for Formal Specifications*. Berlin, Germany: Springer, Aug. 2013. [Online]. Available: https://www.microsoft.com/en-us/research/publication/formula-2-0-language-formal-specifications/

[10] S. Neema, J. Sztipanovits, G. Karsai, and K. Butts, "Constraint-based design-space exploration and model synthesis," in *Embedded Software. EMSOFT* (Lecture Notes in Computer Science), vol. 2855, R. Alur and I. Lee, Eds. Berlin, Germany: Springer, 2003, pp. 290–305.

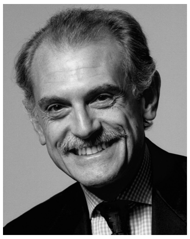
ABOUT THE AUTHORS



Qi Zhu (Member, IEEE) received the B.E. degree in computer science from Tsinghua University, Beijing, China, in 2003 and the Ph.D. degree in electrical engineering and computer science from the University of California at Berkeley, Berkeley, CA, USA, in 2008.

Currently, he is an Associate Professor at the Electrical Engineering and Computer Science Department, Northwestern University, Evanston, IL, USA. Prior to joining Northwestern, he was an Assistant Professor and later Associate Professor at the University of California at Riverside, Riverside, CA, USA, and a Research Scientist with Intel. His research interests include model-based design and software synthesis of cyber–physical systems, CPS security, embedded and real-time systems, and system-on-chip design.

Prof. Zhu received four best paper awards at the Design Automation Conference (DAC), the International Conference on Cyber–Physical Systems (ICCPS), and the *ACM Transactions on Design Automation of Electronic Systems* (TODAES); the National Science Foundation (NSF) CAREER award; and the IEEE Technical Committee on Cyber–Physical Systems (TC-CPS) Early-Career Award. He is an Associate Editor of the IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS (TCAD), and has served on the program committees for a number of conferences in design automation, cyber–physical systems, embedded systems, and real-time systems.



Alberto Sangiovanni-Vincentelli (Fellow, IEEE) is the Edgar L. and Harold H. Buttner Chair at the Electrical Engineering and Computer Sciences Department, University of California at Berkeley, Berkeley, CA, USA, where he has been a member of the faculty since 1976. He helped founding Cadence and Synopsys, the two leading companies in EDA. He is on the Board of Directors of Cadence, KPIT Technologies, Sonics, Expert Systems, and Cogisen. He is a member of the Investment Committee of Atlante Venture, the advisory board of Walden International and Xseed, and of the Executive Committee of the Italian Institute of Technology. He was the President of the Strategic Committee of the Italian Strategic Fund. He consulted for companies such as Intel, HP, Bell Labs, IBM, Samsung, UTC, Lutron, Camozzi Group, Kawasaki Steel, Fujitsu, Telecom Italia, Pirelli, GM, BMW, Mercedes, Magneti Marelli, ST Microelectronics, ELT, Unipol and UniCredit. He authored over 950 papers, 17 books, and two patents.

Dr. Sangiovanni-Vincentelli is a Fellow of the Association for Computing Machinery (ACM), a member of the National Academy of Engineering, and holds two honorary Doctorates from Aalborg University (Denmark) and KTH (Sweden). He earned the IEEE/RSE Maxwell Award for groundbreaking contributions that have had an exceptional impact on the development of electronics and electrical engineering, the Kaufmann Award for seminal contributions to EDA, the EDAA Lifetime Achievement Award, the IEEE/ACM R. Newton Impact Award, the University of California Distinguished Teaching Award, the IEEE Technical Committee on Cyber–Physical Systems (TC-CPS) Technical Achievement Award for pioneering contributions and leadership in cyber–physical systems and design automation, the International Symposium on Physical Design (ISPD) lifetime achievement award, intended for individuals who have made outstanding contributions to the field of physical design automation over multiple decades, and the IEEE Graduate Teaching Award for inspirational teaching of graduate students.

Shiyan Hu (Senior Member, IEEE) received the Ph.D. degree in computer engineering from Texas A&M University, College Station, TX, USA, in 2008.

He is an Adjunct Full Professor at Michigan Technological University, Houghton, MI, USA, where he was an Associate Professor and Director of the Center for Cyber–Physical Systems. He was a Visiting Professor at IBM Research (Austin, TX, USA) in 2010, and a Visiting Associate Professor at Stanford University (Stanford, CA, USA) from 2015 to 2016. His research interests include cyber–physical systems (CPSs), CPS security, and data analytics, where he has published more than 100 refereed papers.

Prof. Hu is an Association for Computing Machinery (ACM) Distinguished Speaker, an IEEE Systems Council Distinguished Lecturer, a recipient of the 2017 IEEE Computer Society TCSC Middle Career Researcher Award, the 2014 U.S. National Science Foundation (NSF) CAREER Award, and the 2009 ACM SIGDA Richard Newton DAC Scholarship. His publications have been awarded several Best Paper Awards and Best Paper Finalists, which include the 2018 IEEE SYSTEMS JOURNAL Best Paper Award. His papers have also been highlighted as a Keynote Paper in the IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN in 2017, as Thomson Reuters ESI Highly Cited Paper, and as the Front Cover in the IEEE TRANSACTIONS ON NANOBIOSCIENCE in March 2014. He is the Chair for the IEEE Technical Committee on Cyber–Physical Systems. He is the Editor- In-Chief of *IET Cyber–Physical Systems: Theory & Applications*. He is an Associate Editor for the IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN, the IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS, the *ACM Transactions on Design Automation for Electronic Systems*, and the *ACM Transactions on Cyber–Physical Systems*. He is also a Guest Editor for eight IEEE/ACM journals such as the PROCEEDINGS OF THE IEEE and the IEEE TRANSACTIONS ON COMPUTERS. He has served as General Chair, Technical Program Committee (TPC) Chair, TPC Subcommittee Chair, Session Chair, and TPC Member for various IEEE and ACM conferences. He is a Fellow of the Institution of Engineering and Technology (IET).

Xin Li (Fellow, IEEE) received the Ph.D. degree in electrical and computer engineering from Carnegie Mellon University, Pittsburgh, PA, USA in 2005.

He is currently a Professor at the Electrical and Computer Engineering Department, Duke University, Durham, NC, USA and is leading the Institute of Applied Physical Sciences and Engineering at Duke Kunshan University, Jiangsu, China. His research interests include integrated circuit, signal processing, and data analytics.

Dr. Li is the Deputy Editor-in-Chief of the IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS. He was an Associate Editor of the IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, the IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING, *ACM Transactions on Design Automation of Electronic Systems*, IEEE DESIGN & TEST, and *IET Cyber–Physical Systems: Theory & Applications*. He was the General Chair of ISVLSI and FAC. He received the National Science Foundation (NSF) CAREER Award in 2012 and six Best Paper Awards from the IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, DAC, ICCAD and ISIC.