Design Automation of Electronic Systems: Past Accomplishments and Challenges Ahead

By ROBERT BRAYTON, *Fellow, IEEE*
Guest Editor

LUCA P. CARLONI, *Senior Member, IEEE*
Guest Editor

ALBERTO L. SANGIOVANNI-VINCENTELLI, *Fellow, IEEE*
Guest Editor

TIZIANO VILLA, *Member, IEEE*
Guest Editor

I. INTRODUCTION

Design automation of electronic systems [electronic design automation (EDA)] is the engineering science that derives software and hardware tools for the design of integrated circuits and systems based on abstraction, design methodologies, and software implementations of sophisticated algorithms for verification and synthesis. What makes EDA unique is the continuous interplay between theory and applications. Computer science, mathematics, and physics offer the foundations upon which EDA rests. However, EDA specialists must also be able to leverage their application domain knowledge to solve abstract problems that are well known to be intractable in general. Indeed, EDA is a cornucopia of interesting problems, some solved and some that are just appearing on the scene, where prominent results come from the tight interaction of mathematicians, computer scientists, physicists, and engineers. The relevance of EDA is also proven by the existence of a vibrant community of EDA companies and universities active in this field. We believe that the future has much to offer to young researchers and engineers.

This special issue provides an overview of and a perspective on the evolution of EDA, and offers a perspective on some of the principal avenues of future development.

The goal of this special issue[1] is twofold: first, to provide an overview of and a perspective on the evolution of EDA and then, to offer a perspective on some of the principal avenues of future development.

EDA started as a field in the 1960s when researchers of some leading academic and industrial labs conceived the first computer-aided design (CAD) tools for supporting engineers in the analysis and layout of circuits and boards whose complexity was growing dramatically. The early

[1]This special issue is based on the invited presentations at a Workshop in honor of the career and contributions to EDA by Robert Brayton held at the Design Automation and Test in Europe (DATE) Conference in 2013.

Digital Object Identifier: 10.1109/JPROC.2015.2487798

1970s saw the birth of SPICE, a package to analyze electronic circuits, which began as a class project at the University of California at Berkeley and became over time the *de facto* standard in every design environment. EDA witnessed a tumultuous growth leading in the 1980s to the foundation of successful companies such as Mentor, Cadence, and Synopsys who capitalized on the research from universities and research labs to provide complete suites of design tools. After a dynamic phase of consolidation, these EDA vendors became billion dollar companies.

At the end of the 1990s, EDA problems became bigger because of the need to address complexity and scalability issues for designs with over a billion transistors. These issues highlighted the need to raise the level of abstraction for hardware specification above register–transfer level (RTL) design. At the same time, the design of complex embedded systems and sensor networks spanning different areas of technology set new challenging objectives for design automation. System-level design among other challenges introduced the need of dealing with physical systems such as electromechanical systems governed by digital controllers (cyber systems). Cyber–physical systems (CPSs) are now on top of the research agenda of the United States and of the European Community because of their pervasive presence in present and future products and infrastructures.

In parallel with the CPS revolution, new technology areas appeared such as nanotechnology and synthetic biology that created the need of a new breed of design tools that could take into consideration the peculiar characteristics of these domains.

This special issue consists of 15 articles organized across six main themes: simulation and circuit design, physical design and timing analysis, equivalence checking and formal verification, microarchitectural and logic optimization, system design, and new frontiers of EDA: nanotechnology and biodesign automation.

II. OVERVIEW OF PAPERS

A. Simulation and Circuit Design

In the 1950s, Eiichi Goto and John von Neumann proposed an innovative scheme for digital computation that used the phase of undulating waveforms to encode logic. However, circuitry for phase logic has so far been difficult to miniaturize, or to run at room temperature. Roychowdhury shows how virtually any self-sustaining nonlinear oscillator can be used to implement phase logic, thus opening the door to the use of many kinds of nanoscale oscillators in diverse domains, such as microelectronics, nanodevices, and biology. Further, he shows that phase logic has inherent noise and robustness advantages over conventional level-based logic, arguing that these advances make phase logic a strong contender for robust, low-power next-generation computation. This work is an offshoot of theory and CAD tools that were originally developed for oscillator phase noise and injection locking prediction. As such, it shows how research in EDA techniques can enable progress in very different areas in surprising ways.

A contribution from the Massachusetts Institute of Technology (MIT) by Zhang and White describes an advance in simulation when implicitly defined Toeplitz-plus-Hankel (TPH) matrices are involved (as when analyzing electromagnetic effects in layered media). To overcome the problem that explicit matrix values are often needed for numerical computations, the authors show that all elements of an implicitly defined TPH matrix can be recovered by sampling four or five carefully selected columns, and then using a linear least squares scheme to recover the rest of the matrix.

B. Physical Design and Timing Analysis

With every new complementary metal–oxide–semiconductor (CMOS) technology node, the share of circuit delay and dynamic power attributable to interconnect (including repeaters) has been growing. This trend highlights the importance of placement to the entire backend flow. Extensive research studies performed over the last 50 years addressed numerous aspects of global and detailed placement. The objectives and the constraints dominant in placement optimization have been revised many times to reflect the changing landscape of physical design, and continue to evolve. The increasing scale of placement instances additionally affects the algorithms of choice for high-performance EDA tools. Markov *et al.* survey the history of placement research, the progress achieved up to now, and outstanding challenges.

Logic synthesis consists of technology-independent optimizations followed by technology-dependent optimizations. Technology-independent optimizations restructure logic netlists without relying on any specific technology, whereas technology-dependent optimizations restructure netlists using information from a technology library. In his overview of technology-dependent optimization, Murgai first reviews the problem of technology mapping: given as input an optimized netlist produced by technology-independent optimization, generate a netlist consisting of gates from the technology library. Various objective functions are useful, such as minimum area, minimum delay, and minimum area subject to the timing constraints. The paper surveys algorithms and major breakthroughs for each of these objectives. Despite these advances, state-of-the-art mappers have several shortcomings and generate suboptimal netlists. This leads to the second part of the paper, namely post-mapping optimizations, to improve the quality of the mapped netlist, by operations such as local restructuring and remapping, gate sizing, gate cloning, buffering, and pin permutation.

C. Equivalence Checking and Formal Verification

Research in EDA has a rich history of attacking intractable problems with the goal of developing algorithms that are effective in practice. Boolean

satisfiability is a fundamental intractable problem in computer science that has received much theoretical and practical attention. It has many applications in EDA, notably in synthesis and verification. While it has been studied in many areas of computer science, e.g., artificial intelligence, the EDA community has contributed the algorithms that have made the biggest impact. The GRASP and Chaff SAT solvers developed in this community form the basis of the key ideas used in all modern SAT solvers. GRASP introduced conflict-driven clause learning, and Chaff introduced locality-based search and efficient unit propagation. Based on these ideas, modern SAT solvers can often handle practical instances with millions of variables and constraints. Vizel *et al.* trace the important contributions made to modern SAT solvers by the EDA community and discuss its EDA applications. A prime application has been in model checking where SAT solving is the basis for almost all the significant algorithms in this area. The paper presents a modern overview of this area and illustrates some new directions of research that combine two of the major methods.

The paper by Seshia presents sciduction: a formal methodology for verification and synthesis that integrates inductive inference (learning) from examples with traditional deductive reasoning using hypotheses about system structure. It generalizes some very effective approaches developed in the field of EDA, such as counterexample-guided abstraction refinement. This novel theoretical framework formalizes some of key connections between verification and synthesis, including the concept of verification by reduction to synthesis, and so it provides a common methodology to tackle both problem domains. This approach addresses some major challenges in formal verification and correct-by-construction synthesis, such as environment modeling, incompleteness in specifications, and the complexity of underlying decision problems. The paper presents examples of practical applications of the methodology and outlines directions for future work.

D. Microarchitectural and Logic Optimization

Fujita discusses a unification of synthesis and verification, as the synthesis results are correct up to the amount of how much we can verify, and designs are basically interactive processes: repetitions of design, verification, and debugging. The paper defines a general functional fault/error model, called functionally observable faults (FOF), where a set of subcircuits in the given combinational circuits may change the logic functions that they realize to any ones with the same sets of inputs. As any change of logic functions inside the subcircuits is targeted, it can deal with both general manufacturing faults as well as logic design errors (bugs), assuming that the errors are inside the subcircuits. Then, the author shows that automatic test pattern generation (ATPG) for FOF can be formulated as a two-level quantified Boolean formula (QBF). The two-level QBF problem is essentially an incremental SAT problem, and a very efficient ATPG method can be derived based on incremental SAT solvers. The numbers of generated test patterns is very small even for circuits having hundreds of primary inputs and even allowing multiple faults. They can be used not only for testing manufacturing faults, but also for formal verification, where checking 100% logical correctness of the partial circuits can be achieved by simulating with the generated small number of test vectors. Therefore, ATPG methods for FOF and their extensions can be used for various types of related synthesis and verification problems including debugging, engineering change order (ECO), and partial resynthesis of circuits.

Cortadella *et al.* review the evolution of logic synthesis techniques since the early days of EDA until the recent advances in automatic pipelining. The first steps in automation introduced two-level and multilevel combinational synthesis techniques to transform gate netlists within the boundaries of the sequential elements. A new generation of techniques, called sequential logic synthesis, enabled the crossing of the sequential boundaries and introduced new optimizations that could change the timing behavior of the state signals while preserving the behavior at the primary outputs. All the previous transformations still preserved a cycle-accurate behavioral equivalence of the system as observed at inputs and outputs. Maintaining the cycle accuracy imposes severe constraints on the type of optimizations that can be used in a circuit. Elasticity emerged as a new paradigm to overcome these limitations, enabling the design of systems that are tolerant to the dynamic changes of the computation and communication delays. The concept of elasticity was largely used in asynchronous circuits where timing is controlled by handshaking events. Later, it was introduced in synchronous designs by discretizing the timing in which events could occur, thus coining the term latency insensitivity. Elasticity opens the door to a new avenue of correct-by-construction behavior-preserving transformations for optimizing systems that cannot be systematically applied in the nonelastic context. These transformations enable an automatic exploration of pipelined microarchitectures that preserve the order of execution of output events (but not cycle accuracy) and offer potential optimizations in area, performance, and power consumption.

Kravets revisits a classic algorithm for algebraic factoring to establish a stronger connection to the functional intent rather than to the structural implications of a design description within synthesis. Then, he analyzes logic factoring problems, taking a fresh look at classic algorithms like fast-extract and presents a scalable factoring algorithm that reduces its dependence on two-level minimization. A case study is used to assess the algorithms within a complete design closure flow, showing substantially improved quality of results. He shows how the algorithm can be parallelized leading to almost linear speedup.

E. System Design

EDA tools have enabled the integrated circuit industry to sustain exponentially increasing product complexity growth to the present day, while maintaining consistent product development timeline and costs. Current digital flows are extremely modular based on a powerful well-understood abstraction hierarchy. They are decomposed into three main steps dealing, respectively, with user functional specification, RTL description, and physical implementation. Design frontends produce standardized descriptions that compile into invocations of components without regard to their implementation technology. Designers extensively use libraries of component models that fit standard simulation tools. Sifakis argues that the success of EDA-based design relies on the application of four interrelated principles: separation of concerns, component-based design, semantic coherency, and correctness by construction. Then, he discusses to what extent the rigorous very large scale integration (VLSI) design paradigm can be extended to general mixed hardware/software system design, in particular through the application of these principles. He points out main differences of system design versus pure software and hardware design and proposes a concept of system correctness characterized as the conjunction of two types of requirements: trustworthiness (the designed system can be trusted that it will behave as expected despite any kind of hazards resulting from logical design errors, hardware failures, and interaction with potential users and physical environment) and optimization requirements (quantitative constraints involving resources such as time, memory, and energy dealing with performance and costs). Finally, he advocates that moving away from empirical to rigorous design is not only of paramount importance for building cost-effectively complex trustworthy systems, but also it is a huge intellectually challenging and culturally enlightening endeavor.

In CPSs, the interconnection of what in the past have been separate worlds (the cyber components and the physical subsystems) substantially increases the design and verification challenges. The realization of CPSs must increasingly rely on methodologies that propose to cope with their complexity and heterogeneity via a formalization of the design process with the goal of building correct implementations. Nuzzo *et al.* review methodologies, formalisms, and tools for CPS design and verification. In particular, they introduce a platform-based methodology that leverages abstraction and compositional reasoning to enable analysis, design space exploration, and correct-by-construction synthesis of system architecture and control. Design space exploration is carried out as a sequence of refinement steps from the initial specification to a final implementation by mapping higher level functional and nonfunctional models into a set of candidate solutions built out of a library of components at the lower level. In the methodology, assume–guarantee horizontal and vertical contracts formalize system requirements and provide consistent interfaces among specialized synthesis and verification frameworks that allow reasoning about different design aspects.

As integrated circuit technology progresses, the speed of global wires is not keeping up with the speed of ever-smaller transistors, so that the digital chip is now effectively a distributed system, breaking the synchronous paradigm assumption, on which mainstream CAD flows were based. Carloni overviews the principles and practice of latency-insensitive design (LID), a correct-by-construction design methodology developed primarily to address this challenge. LID introduced the protocols and shells paradigm, which offers several main benefits: modularity (by reconciling the synchronous paradigm with the dominant impact of global interconnect delays that characterizes nanometer process technologies), scalability (by making key properties of the design correct by construction through interface synthesis), flexibility (by simplifying the design and validation of a system through the separation of communication from computation), and efficiency (by enabling the reuse of predesigned components, thus reducing the overall design time). This establishes the foundation to bridge the gap between system-level design and physical design, a requisite to cope with the design complexity of future system-on-chip platforms.

An important step in the design of a complex system is its decomposition into a number of interacting components, of which some are given (known) and some need to be synthesized (unknown). Then, a basic task in the design flow is to synthesize an unknown component that when combined with the known part of the system (the context) satisfies a given specification. This scenario arises in several applications ranging from logic synthesis to the design of discrete controllers. Villa *et al.* review the existing formalisms, algorithmic solutions, and design automation tools to specify and synthesize unknown components in compositional finite-state systems.

F. New Frontiers of EDA: Nanotechnology and Synthetic Biology

Emerging technologies are characterized by the use of devices at the nanoscale, where new effects are important. As an example, an important class of devices supporting the use of controlled polarity gates are those based on silicon nanowires, carbon nanotubes, and graphene. Their logic abstraction is the equivalence (or difference) operation, which is intrinsically more expressive than the switch realized by standard transistors. Amarú *et al.* investigate the relation between logic synthesis and emerging nanotechnologies, and show how new logic synthesis techniques can enable the identification of the full potential of a given nanotechnology.

Advances in EDA have been crucial in the development of complex

electronic systems. The ability to independently specify, design, and assemble electronic systems at various abstraction levels has enabled tremendous growth in the semiconductor industry. As the field of synthetic biology (creating novel life forms from the ground up) grows, it is essential to introduce design methodologies that enable specification, design, and physical assembly while preventing potentially catastrophic side effects and emerging behavior. The paper by Vaidyanathan *et al.* outlines how specific approaches in EDA can be applied to synthetic biology, introducing a framework to address the challenges of applying logic synthesis techniques to genetic logic devices.

III. CONCLUSION

Our intention has been to provide the reader with an overview of the state-of-the-art of EDA from simulation, to physical design, from formal verification to logic optimization as well as of its evolution into system level design, nanotechnology, and synthetic biology. The field of EDA remains rich of challenges to overcome as the complexity of new products, services, and implementation platforms grows boundless. The reader is encouraged to browse not only the journals and magazines fully dedicated to these topics (e.g., the IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, IEEE DESIGN & TEST OF COMPUTERS, the *ACM Transactions on Design Automation of Electronic Systems*, and the *ACM Transactions on Embedded Computing Systems*) but also almost all other publications in engineering and computer science that feature papers describing tools for automation. The proceedings of the premier conferences in the domain [e.g., Design Automation Conference (DAC), International Conference on Computer-Aided Design (ICCAD), Design Automation and Test in Europe (DATE), and Asian-Pacific Design Automation Conference (ASPDAC)] are also a source of valuable information. The IEEE Council on Electronic Design Automation (CEDA) and the ACM Special Interest Group on Design Automation (SIGDA) include IEEE and Association for Computing Machinery (ACM) members with an interest in design automation. The Electronic Design Automation Consortium includes the companies that have substantial interests in EDA. The new areas where electronic design automation is of interest (embedded systems, CPSs, and synthetic biology) have their own groups, journals, conferences, and workshops that are too numerous to cite here.

Our take is that design automation is essential to the advancement of all engineering disciplines. We expect its importance to grow continuously as the design challenges offered by new technologies surface and engineers seek the assistance of automation tools that have solid foundations on rigorous analysis and sound algorithms. ■

ABOUT THE GUEST EDITORS

Robert Brayton (Fellow, IEEE) received the Ph.D. degree in mathematics from the Massachusetts Institute of Technology (MIT), Cambridge, MA, USA, in 1961.

He was a member of the Mathematical Sciences Department, IBM T. J. Watson Research Center, New York, NY, USA. In 1987, he joined the Electrical Engineering and Computer Science Department, University of California at Berkeley, Berkeley, CA, USA, where he is currently a Professor in the Graduate School.

Dr. Brayton is a recipient of the IEEE Emanuel R. Piore Award in 2006, the ACM Kanallakis Award in 2006, the European DAA Lifetime Achievement Award in 2006, the EDAC/CEDA Phil Kaufman Award in 2007, the D.O. Pederson Best Paper Award in 2008, the ACM/IEEE A. Richard Newton Technical Impact in EDA Award in 2009, the Iowa State University Distinguished Alumnus Award in 2010, the SRC Technical Excellence Award in 2011, and the ACM/SIGDA Pioneering Achievement Award in 2011. He also held the Buttner Chair and the Cadence Distinguished Professorship of Electrical Engineering at Berkeley. He is a member of the National Academy of Engineering.

Luca P. Carloni (Senior Member, IEEE) received the Laurea degree (*summa cum laude*) in electrical engineering from the Università di Bologna, Bologna, Italy, in 1995 and the M.S. and Ph.D. degrees in electrical engineering and computer sciences from the University of California Berkeley, Berkeley, CA, USA, in 1997 and 2004, respectively.

He is currently an Associate Professor with the Department of Computer Science, Columbia University, New York, NY, USA. He has authored over 100 publications and holds two patents. His current research interests include methodologies and tools for heterogeneous multicore platforms with emphasis on system-level design and design reuse, system-on-chip design, embedded software, and distributed embedded systems.

Dr. Carloni was a recipient of the Demetri Angelakos Memorial Achievement Award in 2002, the Faculty Early Career Development (CAREER) Award from the National Science Foundation in 2006, the Office of Naval Research (ONR) Young Investigator Award in 2010, and the IEEE Council on Electronic Design Automation (CEDA) Early Career Award in 2012. He was selected as an Alfred P. Sloan Research Fellow in 2008. His 1999 paper on the latency-insensitive design methodology was selected for the Best of ICCAD, a collection of the best papers published in the first 20 years of the IEEE International Conference on Computer-Aided Design. In 2010, he served as Technical Program Co-Chair of the International Conference on Embedded Software (EMSOFT), the International Symposium on Networks-on-Chip (NOCS), and the International Conference on Formal Methods and Models for Codesign (MEMOCODE). He was the Vice General Chair (in 2012) and General Chair (in 2013) of Embedded Systems Week (ESWEEK), the premier event covering all aspects of embedded systems and software. He is a Senior Member of the Association for Computing Machinery (ACM).

Alberto L. Sangiovanni-Vincentelli (Fellow, IEEE) received the Laurea degree (*summa cum laude*) in electrical engineering and computer sciences from the Politecnico di Milano, Milan, Italy, in 1971.

He currently holds the Edgar L. and Harold H. Buttner Chair of Electrical Engineering and Computer Sciences at the University of California at Berkeley, Berkeley, CA, USA. He was a cofounder of Cadence and Synopsys, the two leading companies in the area of electronic design automation (EDA), and the founder and Scientific Director of the Project on Advanced Research on Architectures and Design of Electronic Systems (PARADES) research center in Rome, Italy. He has been a member of the Board of Directors of Cadence, KPIT-Cummins, Sonics, and Expert Systems. He was a member of the ST Microelectronics Advisory Board for ten years. He was a member of the HP Strategic Technology Advisory Board (2005–2007) and the Science and Technology Advisory Board of General Motors (2003–2013), and he is a member of the Technology Advisory Council of United Technologies Corporation. He is also member of the Scientific Council of the Italian National Science Foundation (CNR). Since 2010, he has been a member of the Executive Committee of the Italian Institute of Technology. Since July 2012, he has been named Chairperson of the Comitato Nazionale Garanti per la Ricerca. He is an author of over 880 papers, 17 books, and three patents in the area of design tools and methodologies, large-scale systems, embedded systems, hybrid systems, and innovation.

Dr. Sangiovanni-Vincentelli has been an IEEE Fellow since 1982, a Member of the National Academy of Engineering since 1998, and an Association for Computing Machinery (ACM) Fellow since 2014. In 1981, he received the Distinguished Teaching Award of the University of California. He received the worldwide 1995 Graduate Teaching Award of the IEEE for ''inspirational teaching of graduate students.'' In 2002, he was the recipient of the Aristotle Award of the Semiconductor Research Corporation. He received numerous research awards including the Guillemin-Cauer Award (1982–1983), the Darlington Award (1987–1988) of the IEEE for the best paper bridging theory and applications, two awards for the best paper published in the IEEE Transactions on Circuits and Systems and Computer-Aided Design, five best paper awards, and one best presentation award at the Design Automation Conference. In 2001, he was given the Kaufman Award of the Electronic Design Automation Council for ''pioneering contributions to EDA.'' In 2008, he was awarded the IEEE/RSE Wolfson James Clerk Maxwell Medal ''for pioneering innovation and leadership in electronic design automation that have enabled the design of modern electronics systems and their industrial implementation.'' In 2009, he was awarded an honorary Doctorate by the University of Aalborg, Aalborg, Denmark, and he received the first ACM/IEEE A. Richard Newton Technical Impact Award in Electronic Design Automation to honor persons for an outstanding technical contribution within the scope of electronic design automation. In 2012, he was awarded an honorary Doctorate from the Royal Institute of Technology (KTH), Stockholm, Sweden, and he received the Lifetime Achievement Award from EDAA.

Tiziano Villa (Member, IEEE) received the B.Sc. degree in mathematics from the University of Milano, Milan, the Postgraduate Degree in computational mathematics from the University of Pisa, Pisa, Italy, and UK, DAMTP, Mathematical Tripos Part III from the University of Cambridge, Cambridge, U.K.

From 1980 to 1985, he worked as a Computer-Aided Design Specialist in the integrated circuits division of the CSELT Labs, Torino, Italy, and then from 1986 to 1996, he was a Research Assistant with the Electronics Research Laboratory, University of California at Berkeley, Berkeley, CA, USA. From 1997 to 2001, he was a Research Scientist with the PARADES Labs, Rome, Italy. From 2002 to 2006, he was an Associate Professor with the Department of Electrical, Industrial, and Mechanical Engineering (DIEGM), Università degli Studi di Udine, Italy. Since October 2006, he has been a Full Professor with the Department of Computer Science, Università degli Studi di Verona, Verona, Italy. His research interests include computer-aided design of digital circuits (especially logic synthesis), formal verification, cyber–physical systems, and automata theory. He coauthored the books: *Synthesis of FSMs: Functional Optimization* (Norwell, MA, USA: Kluwer, 1997), *Synthesis of FSMs: Logic Optimization* (Norwell, MA, USA: Kluwer, 1997), and *The Unknown Component Problem: Theory and Applications* (New York, NY, USA: Springer-Verlag, 2012); he coedited the book *Coordination Control of Distributed Systems* (New York, NY, USA: Springer-Verlag, 2015).

Dr. Villa was awarded the Tong Leong Lim Predoctoral Prize from the Electrical Engineering and Computer Science Department, University of California at Berkeley in May 1991.