1-2018

Software Safety and Security Risk Mitigation in Cyber-Physical Systems

Miklos Biro
Software Competence Center

Atif Mashkoor
Software Competence Center

Johannes Sametinger
Johannes Kepler University of Linz

Remzi Seker
Embry-Riddle Aeronautical University, sekerr@erau.edu

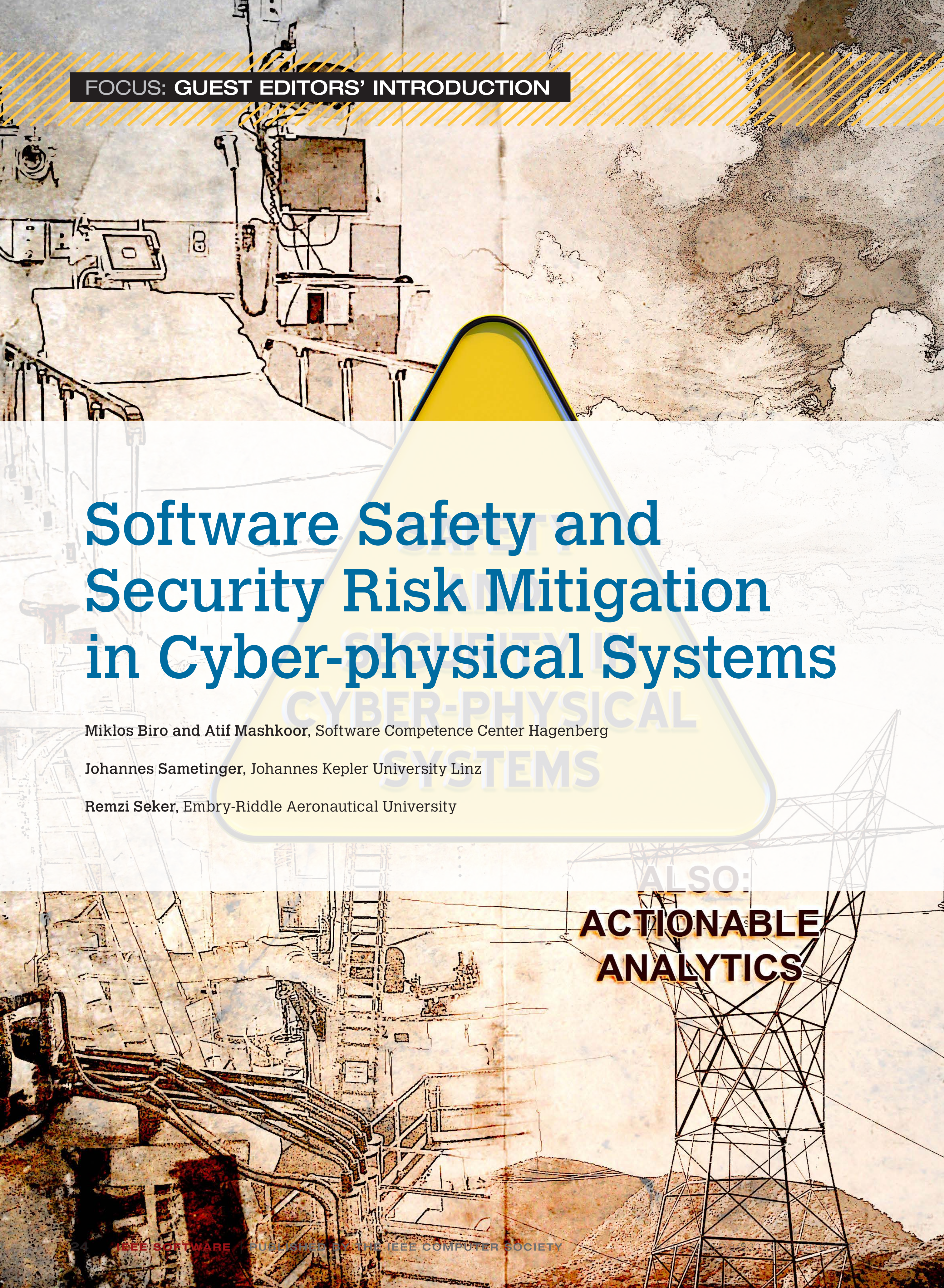
FOCUS: GUEST EDITORS' INTRODUCTION

Software Safety and Security Risk Mitigation in Cyber-physical Systems

Miklos Biro and Atif Mashkoor, Software Competence Center Hagenberg

Johannes Sametinger, Johannes Kepler University Linz

Remzi Seker, Embry-Riddle Aeronautical University



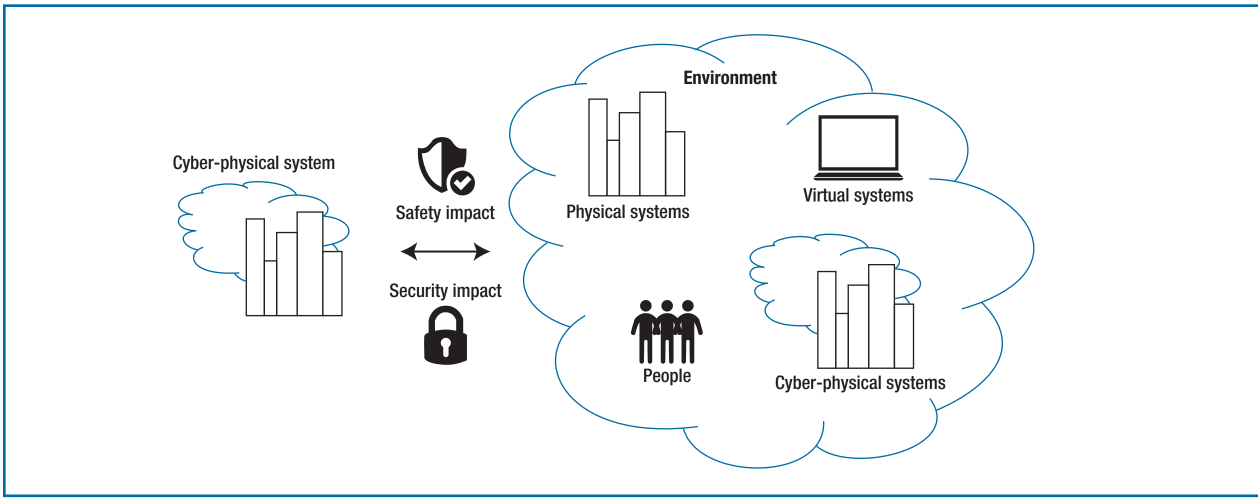


FIGURE 1. Cyber-physical systems influence each other's and their environments' safety and security.

CYBER-PHYSICAL SYSTEMS (CPSs) are smart systems including engineered interacting networks of physical and computational components.[1] Modern CPSs comprise systems of systems with heterogeneous components. These systems' manufacturers must address dynamic and uncertain environmental constraints. CPSs are often safety-critical; any system malfunction might seriously harm its users. The involved communicating peripherals also necessitate consideration of security issues so that cybersecurity threats don't affect a CPS's proper functioning.

As the number of features requested from existing systems grows, computation, communication, and control increasingly converge.[2] This evolution comes with challenges that must be met for these systems to provide the needed services with the desired quality attributes. CPSs span many domains and hence have a wide impact. These domains include biomedical and healthcare systems,[3] transportation systems,[4] the smart grid,[5] automotive systems,[6] and manufacturing systems.[7]

The Interplay between Safety and Security

We consider a CPS with valuable services to be security-critical if it communicates with the outside world, because the communication channel might be opening up an attack vector against the CPS. On the other hand, a CPS is considered safety-critical if it can harm its environment—for example, a malfunctioning pacemaker might harm its patient.

Figure 1 shows how safety and security overlap and affect each other. A CPS can be safety-critical, security-critical, or both. However, security is especially a high priority in safety-critical systems because vulnerabilities might lead to safety-critical incidents.

Contemporary systems and software engineering methods and approaches often prove inadequate for the high-confidence design and manufacturing of CPSs. The conventional approach for engineering safety- and security-critical systems is to address safety and security in separate subsystems. However, owing to trends in CPSs toward openness, increased communication, and multicore architectures, this separation approach is no longer feasible. We need methods that deal simultaneously with functional safety and cybersecurity.

So, how do we model and analyze this interplay? For example, how do we ensure that software features on the same module don't interfere with each other? How do we guarantee that, if a security breach occurs, other system functions aren't at risk? How do we develop guidelines that provide concrete technical advice for designing and deploying safe, secure systems instead of focusing on safety- and security-critical features in isolation? How do we avoid negatively impacting assurance and compliance in safety-critical systems when we address cybersecurity challenges associated with them?

Dealing with Complexity

CPSs usually comprise various heterogeneous systems—for example, mechanical, electrical, electronics, and software systems. It's not uncommon for CPSs to be networked and interconnected; this connectivity introduces additional software and hardware for communication and control. So, CPSs' heterogeneity makes them extremely complex.

It's well known from systems engineering that as a system's complexity increases, guaranteeing the quality of service expected from the system becomes more difficult. The uncertainty in CPSs—and hence their complexity—is exacerbated as they incorporate additional capabilities throughout their lifecycle. As a CPS gains capabilities, its inherent entropy increases, making it a challenge to provide any assurance. Interesting research opportunities exist for assuring the desired level of quality attributes as CPS complexity increases.

> As a system's complexity increases, guaranteeing the system's expected quality of service becomes more difficult.

Potential Threats

Manufacturing used to be a single-shop-floor endeavor. Once manufacturing entered the distribution era, cybermanufacturing[8] became the next phase of evolution. Beyond the traditional manufacturing challenges, in cybermanufacturing risks have arisen through the addition of software and connectivity. Attack vectors that didn't previously exist have quickly become a priority. Manufacturers now must ensure that a design sent to a networked 3D printing device can't be extracted from network packets, a challenge they didn't have to consider until recently.

Traditionally, software safety has focused on safety-critical systems such as those in aviation,[9] medical devices,[10] transportation,[11] and nuclear power plants.[12] Development processes as well as risk assessment and mitigation activities often cover the safety aspects of software in safety-critical systems. As the connectivity of safety-critical systems or systems with a safety-critical core increases, new attack vectors for these systems surface.

Introducing additional features into systems with a safety-critical core expands their attack surface. The ease of attack vector generation requires careful consideration in changing a system's operating environment as well. For legacy systems, incorporating security needs within the original requirements might not be possible.[13]

Once a system with a safety-critical core is in service, the system's evolution begins; it continues as additional services or capabilities are implemented.[14] For example, once an automobile with a CAN (Controller Area Network) bus was built, one of the next services to be added was a tire-pressure-monitoring system (TPMS). When first deployed, the TPMS wasn't considered safety-critical and thus wasn't scrutinized as a critical system. If it had been, the widely reported exploitation of the TPMS[15] might not have been so easy or might not have happened at all.

Similarly, a particular fleet management system provided location information for its vehicles via the cell phone network.[16] The fleet management units were connected to the CAN bus of the trucks on which they were installed. At the time, the units had no protection for remote access. Once located, they were accessible over the Internet. It's not difficult to imagine an attack by malware, similar to Mirai, targeting an unprotected fleet management system and possibly other CPSs.

Some specialized CPSs rely on lack of access to specialized equipment and on laws for their protection. For example, the traffic collision avoidance system (TCAS), which aims to reduce the risk of mid-air collisions, has been used for decades. Despite incremental versions of the system, two fundamental assumptions remain:

- The equipment for operation is available only on aircraft.
- Illegal broadcasting in the TCAS frequency band won't occur.

It wouldn't be too difficult for an adversary to attack such systems.

Risk Management

Given CPSs' inherent complexity and their impact, risk assessment and reduction techniques are needed. These techniques must be easy to modify and adapt to accommodate changes in CPSs and their operating environment. Additionally, some CPSs might require systematic risk assessment and management to maintain a certificate of operation. In such cases, the tools and instruments developed for risk management must be rapid, cost-effective, and practical.

Some CPSs might not have any safety implications and thus might need only rudimentary risk assessment. Researchers will have to develop adaptable techniques that produce reliable results when, for example, two massive CPSs are interconnected to create a new system.

Formal Methods and Legacy Systems

Because some CPSs will need a degree of assurance about the correctness of certain functions, we need innovative approaches to applying formal methods in such complex systems. Given that CPSs might contain complex legacy systems that aren't well-documented, adopting formal methods to provide a degree of assurance for such systems becomes even more challenging. Tools and techniques for the use of formal methods in CPSs must meet the domain's assurance and compliance needs while scaling up to deal with very large and complex systems.

Legacy systems constitute most of our infrastructures. These systems were built to provide well-defined services. They still provide those services, but the environments in which they operate have changed considerably. This necessitates revisiting them to upgrade and protect them accordingly. Making changes to large, complex systems is costly, and the rapid pace of change in IT systems makes things even more challenging.

New Approaches Needed

Considering that we depend on CPSs for our societal well-being, we need innovative approaches for designing, maintaining, updating, and upgrading them. These approaches must be cost-effective and system-agnostic. So, a technique developed for CPSs in one application domain should be adaptable for CPSs in another domain without major adaptation costs.

These approaches must also address the challenges of combining services from multiple existing CPSs, including legacy systems. For example, interconnecting diverse systems might require accurate translation between different data types for assured operation of CPSs. Considering the various unfortunate events caused by faults associated with data types, such "minor" details in large-scale systems such as CPSs introduce additional challenges.

Opportunities Ahead

The CPS research area seems to offer various opportunities at different phases of the system lifecycle:

- *Design and implementation.* A CPS could be designed and implemented from scratch or around an existing system.
- *Combining existing CPSs.* Two or more existing CPSs could be merged into a new one for a specific purpose—for example, to make process control more efficient.
- *Adding capabilities to existing CPSs.* An existing CPS could incorporate additional sensor values to report, for finer-grained control.
- *Dismantling or decommissioning existing CPSs.* A CPS could be dismantled into its subsystems, and subsystems or functions that are no longer needed could be eliminated.

Each of these phases has unique challenges requiring researchers' attention.

Given CPSs' challenges and opportunities—especially regarding functional safety, cybersecurity, and their interplay, as well as the systems' impact on society—new methods and techniques are needed for CPS development and assurance. The articles in this theme issue aim to help address some of these challenges.

In This Issue

For this issue, we received 17 submissions from around the world. After thorough and stringent reviews, we selected three articles that represent key issues associated with functional safety and cybersecurity in CPSs.

In "Safe, Secure Executions at the Network Edge: Coordinating Cloud, Edge, and Fog Computing," Niko Mäkitalo and his colleagues introduce *action-oriented programming* (AcOP). This programming model has a framework that can dynamically adapt to edge and cloud computing according to the given environment and connectivity. Mäkitalo and his colleagues compare AcOP to mobile-app-based and cloud-based deployment of CPSs. They also propose a framework to enable secure coalition and dynamic management of collective executions. This research's strongest aspect is a proposed communication paradigm that addresses critical real-life situations, such as car accidents.

In "Probabilistic Threat Detection for Risk Management in Cyber-physical Medical Systems," Aakarsh Rao and his colleagues present a dynamic risk management and mitigation approach based on probabilistic threat estimation. This research's strongest aspect is the application of the results to solve critical issues regarding a smart-connected-pacemaker, a candidate system from a demanding area of healthcare.

In "Leveraging Software-Defined Networking for Incident Response in Industrial Control Systems," Andrés Murillo Piedrahita and his colleagues focus on how to respond to attacks targeting industrial control systems (ICSs). They show how software-defined networks and network function virtualization technologies can help designers architect automatic-incident-response mechanisms for ICSs. Such an infrastructure enables the implementation of a variety of automatic reactions.

Because CPSs are critical to sustaining and improving the quality of our lives, their safety and security are crucial. Thus, the topic requires greater attention from the engineering community.

Beyond the need for advances in engineering, awareness of this concern is growing among policymakers, as was evidenced in European Commission President Jean-Claude Juncker's State of the Union Address on 13 September 2017:

> *Cyber-attacks can be more dangerous to the stability of democracies and economies than guns and tanks. … Cyber-attacks know no borders and no one is immune. This is why, today, the Commission is proposing new tools, including a European Cybersecurity Agency, to help defend us against such attacks.*[17]

We hope this special issue serves as a drop in the ocean of knowledge on improving CPSs and the services they offer, especially regarding functional safety, cybersecurity, and their interplay.

Acknowledgments

The research presented in this article was supported partly by the Austrian Ministry for Transport, Innovation, and Technology; the Austrian Federal Ministry of Science, Research, and Economy; and the Province of Upper Austria in the framework of the COMET (Competence Centers for Excellent Technolo-

ABOUT THE AUTHORS

MIKLOS BIRO is a key researcher at Software Competence Center Hagenberg (SCCH) and a senior researcher at Johannes Kepler University Linz. His research interests are systems and software process improvement, software engineering, and decision support systems. Biro received a PhD in mathematics from Eötvös Loránd University. He's the founding president of the Software Quality Management Division of the John von Neumann Computer Society, the Hungarian national representative in IFIP TC-2 Software: Theory and Practice, and the representative of Upper Austrian Research and SCCH in the EARTO Security & Defense Research Group. Contact him at miklos.biro@scch.at.

ATIF MASHKOOR is the scientific head of the Rigorous Methods in Software Engineering research focus at the Software Competence Center Hagenberg. His research interests are formal methods, rigorous systems, and software engineering. Mashkoor received a doctorate in computer science from the University of Lorraine. Contact him at atif.mashkoor@scch.at.

JOHANNES SAMETINGER is an associate professor in the Department of Information Systems at Johannes Kepler University Linz. His research interests include software engineering and IT security, emphasizing software security and medical-device security. Sametinger received a Dr. techn. in computer science from Johannes Kepler University Linz. Contact him at johannes.sametinger@jku.at.

REMZI SEKER is a professor of computer science and the director of the Cybersecurity and Assured Systems Engineering Center at Embry-Riddle Aeronautical University. His research focuses on cybersecurity issues in aviation and aerospace systems. He serves on RTCA's Special Committee 216, which is authoring and updating two cybersecurity standards for commercial aircraft and their continued airworthiness. Seker received a PhD in computer engineering from the University of Alabama at Birmingham. Contact him at sekerr@erau.edu.

gies) center Software Competence Center Hagenberg.

References

1. "Framework for Cyber-physical Systems Release 1.0," Cyber-physical Systems Public Working Group, May 2016; pages.nist.gov/cpspwg.
2. S. Graham and P.R. Kumar, "The Convergence of Control, Communication, and Computation," *Proc. IFIP-TC6 8th Int'l Cont. Personal Wireless Communications* (PWC 03), 2003, pp. 458–475; doi.org/10.1007/978-3-540-39867-7_44.
3. A. Mashkoor and J. Sametinger, "Rigorous Modeling and Analysis of Interoperable Medical Devices," *Proc. 2016 Modeling and Simulation in Medicine Symp.* (MSM 16), 2016, pp. 800–807; dl.acm.org/citation.cfm?id=2962683.
4. A. Mashkoor and O. Hasan, "Formal Probabilistic Analysis of Cyber-physical Transportation Systems," *Proc. 12th Int'l Conf. Computational Science and Its Applications* (ICCSA 12), 2012, pp. 419–434.
5. X. Yu and Y. Xue, "Smart Grids: A Cyber-physical Systems Perspective," *Proc. IEEE*, vol. 104, no. 5, 2016, pp. 1058–1070.
6. S. Chakraborty et al., "Automotive Cyber–physical Systems: A Tutorial Introduction," *IEEE Design & Test*, vol. 33, no. 4, 2016, pp. 92–108.
7. R.F. Babiceanu and R. Seker, "Big Data and Virtualization for Manufacturing Cyber-physical Systems: A Survey of the Current Status and Future Outlook," *Computers in Industry*, vol. 81, 2016, pp. 128–137; dx.doi.org/10.1016/j.compind.2016.02.004.
8. J. Lee, B. Bagheri, and C. Jin, "Introduction to Cyber Manufacturing," *Manufacturing Letters*, vol. 8, pp. 11–15, 2016.
9. A. Mashkoor and J.-P. Jacquot, "Observation-Level-Driven Formal Modeling," *Proc. IEEE 16th Int'l Symp. High-Assurance Systems Eng.* (HASE 15), 2015, pp. 158–165.
10. A. Mashkoor and M. Biro, "Towards the Trustworthy Development of Active Medical Devices: A Hemodialysis Case Study," *IEEE Embedded Systems Letters*, vol. 8, no. 1, 2016, pp. 14–17.
11. A. Mashkoor and J.-P. Jacquot, "Domain Engineering with Event-B: Some Lessons We Learned," *Proc. 18th IEEE Int'l Requirements Eng. Conf.* (RE 10), 2010, pp. 252–261.
12. D.L. Parnas, G. Asmis, and J. Madey, "Assessment of Safety-Critical Software in Nuclear Power Plants," *Nuclear Safety*, vol. 32, no. 2, 1991, pp. 189–198.
13. R. Gauthier and R. Seker, "Addressing Operator Privacy in Automatic Dependent Surveillance Broadcast (ADS-B)," to be published in *Proc. 51st Hawaii Int'l Conf. System Sciences* (HICSS 18), 2018.
14. R. Messnarz et al., "Need for the Continuous Evolution of Systems Engineering Practices for Modern Vehicle Engineering," *Systems, Software and Services Process Improvement*, J. Stolfa et al., eds., Springer, 2017, pp. 439–452; link.springer.com/chapter/10.1007%2F978-3-319-64218-5_36.
15. C. Miller and C. Valasek, "Remote Exploitation of an Unaltered Passenger Vehicle," presentation at Black Hat USA 2015.
16. T. Fox-Brewster, "Warning over Truck Attacks as Fleet Control Tech 'Left Open' to Hackers," *Forbes*, 9 Mar. 2016; www.forbes.com/sites/thomasbrewster/2016/03/09/remote-control-truck-hacker-threat.
17. J.-C. Juncker, *State of the Union Address 2017*, European Union, 2017; europa.eu/rapid/press-release_SPEECH-17-3165_en.htm.