High-Assurance Design of Learning-Enabled Cyber-Physical Systems

Pierluigi Nuzzo, Department of Electrical Engineering, University of Southern California, Los Angeles, CA

Overview. Over the last decade, machine learning (ML) algorithms have achieved impressive results in solving large-scale perception problems for decision-making under uncertainty in autonomous cyber-physical systems (CPSs) (see, e.g., [3, 9, 12]). While autonomous systems heavily depend on situational awareness and ML components, their safety-critical nature raises concerns on the overall system safety and trustworthiness. In fact, even well-trained neural networks can be sensitive to small perturbations, e.g., small modifications can lead the network to misclassify the altered input with large confidence [17]. In systems with complex interactions, a misclassification error in one component may eventually compromise the entire functionality and cause accidents and fatalities, such as the ones recently reported for self-driving cars. While exhaustive testing is a prohibitive task [10], the complexity, heterogeneity, and highly dynamic nature of learning-enabled cyber-physical systems (LE-CPSs) make it challenging to even leverage existing model-based development techniques to effectively assess their correctness, dependability, and compliance with regulations. New rigorous design and analysis technologies are sorely needed to guarantee safety properties for these systems. Moreover, assumptions made at design time may break at run time, and correctness claims must often be based on evidence that can only be provided during system operation. *In this project, we pursue a holistic approach for rigorous design of LE-CPSs spanning design methodology, analysis and synthesis algorithms, and certified system architectures. Rather than focusing on ML components in isolation, we seek to systematically specify and analyze the behaviors of ML components within the context of a CPS and propose a methodology that can contextually generate safe architectures and their proofs of correctness.* We will develop a modeling framework, methods, and tools that can support the design process from system-level requirement capture, formalization, and analysis, to the generation, testing, and continual monitoring of software and hardware artifacts, including ML components, in feedback loop with physical processes.

Intellectual Merit. We will follow a contract-based approach [13, 15], which has shown to be promising for complex CPS design, since it facilitates the realization of modular architectures that are *correct by construction*, and leverages compositional reasoning to enhance the *scalability* of design and verification. A contract can be intuitively seen as a pair (A, G) of properties, where A specifies the assumptions that a component makes on the external environment and G the guarantees it provides under these assumptions. Contract-based design has emerged as a unifying, formal, compositional paradigm that can enable modular and hierarchical verification of system properties, rigorous stepwise refinement, and component reuse. Contracts have shown to be effective for specifying and reasoning about systems, especially when component models belong to the same level of abstraction (e.g., algorithm, software, architecture) or adopt the same formalism. However, there is no universal modeling formalism that can capture every aspect of complex, heterogeneous LE-CPSs, and guarantee, at the same time, tractable analysis. Designers usually "decompose" a complex system into semantically different models, by adopting the most convenient formalisms to represent different design viewpoints (e.g., system functionality, timing, safety, and reliability) at different abstraction levels. They then leverage the most suitable tools to analyze and synthesize these models separately. For example, the control law for a vehicle is typically derived and initially evaluated using control-oriented models, e.g., discrete-time linear models, in which details of the implementation and the physical dynamics are simplified or neglected. These details are usually modeled and evaluated using other formalisms and tools, e.g., by simulation of continuous-time differential algebraic equations. Each model represents some aspects of system design and occludes others, by making simplifying assumptions that are often undocumented or informally captured at best. A set of challenges then remain for model-based design flows when models are to be formulated and manipulated along the design flow, across different abstraction levels, and between design time and operation time. ML components further exacerbate these difficulties due to the lack of practical and reliable abstractions that can encompass the heterogeneity of ML algorithms and architectures (e.g., data-driven classifiers for perception and situational awareness, reinforcement learning for developing control policies, and algorithms for learning system dynamics) and the different nature of their performance and safety properties. *Our vision for LE-CPS design is that compositional abstractions should bridge heterogeneous formalisms and heterogeneous decomposition architectures, to make system analysis and*

synthesis tractable, to consistently combine different verification and synthesis methods at design time, and to provide seamless support for dynamic assurance at run time. We will pursue this vision via the following multi-pronged approach:

(1) *Hierarchical Heterogeneous Design Methodology.* We formalize the design as a layered process. At each abstraction layer, a representation of the design is built out of a library (collection) of components according to composition rules. The high-level requirements are formalized and a *design refinement* step called *mapping* is performed, where the requirements are mapped into the implementation library components. All the components and their aggregations will be specified by *contracts*. The contracts will be used to construct mathematical evidence that a design is correct and consistent with its physical implementation under certain assumptions on the external environment. We will develop libraries, algorithms, and tools for capturing requirements of LE-CPSs, for translating them into contracts, for analyzing and validating them using contract operations and relations, and for synthesizing design and verification artifacts from contracts.

(2) *Vertical Contracts.* While contracts are traditionally used to reason about "horizontal" compositions of "homogeneous" elements at the same abstraction level, we will develop a theory of *vertical contracts* as a foundation to fully support multi-layer design with heterogeneous models. Intuitively, if vertical contracts are satisfied, the mapping mechanism defined above can be used to produce design refinements that are correct. In model-based verification, abstractions between heterogeneous models were established in the past for specific pairs of formalisms, such as hybrid abstractions of nonlinear systems [8, 6], linear hybrid automata abstractions of linear hybrid systems [7], discrete abstractions of hybrid systems [2, 1, 4]. Our work aims to generalize these notions toward a formulation that is compositional, scalable, and systematically applies to virtually any pair of formalisms. We will use concepts from *abstract interpretation* as a means to build *conservative approximations* for verifying or enforcing safety properties in behavioral models. Abstract interpretation works by creating an abstraction of the model in which it is more efficient (i) to infer state invariants and behavioral properties, and (ii) to prove properties on the given model [5, 14]. Abstract interpretation will be combined with *model order reduction* techniques to provide a foundation for the verification of contract refinement as well as the generation of efficient runtime monitors.

(3) *Abstraction of Learning Components.* Learning components are usually required to achieve high accuracy, integrate data from several sensors, and run efficiently on embedded processors. Learning techniques are heterogeneous, depending on their architectural structure (hierarchical, decentralized, or centralized), varying environment conditions, and the tight accuracy, latency, and energy constraints. We will develop theories and tools for abstraction and analysis of heterogeneous learning systems that can be used to inform architectural decisions and trade performance with safety, robustness, and computation time.

(4) *Scalable Computational Tools for Verification and Synthesis.* The robust control of continuous dynamical systems often involves the reduction of the control problem to a convex optimization problem. On the other hand, the design and verification of discrete systems involves the use of Boolean satisfiability (SAT) [11] solving. We will build upon a nascent framework that combines both SAT solving and convex optimization, termed as Satisfiability Modulo Convex programming (SMC) [16], to develop efficient computational tools to jointly reason about discrete and continuous dynamics, and support constraint solving problems that arise in the verification and synthesis of LE components and systems.

During the first phase of the project, we will consider scenarios from autonomous driving in unknown urban environments to prototype and evaluate our methodology and tools. We can then leverage state-of-the-art traffic simulators (e.g., PTV VISSIM) to test the real-time performance and safety of our solutions. As we target industrial-strength platforms, we will also partner with the new Center of Advanced Manufacturing and the Robotics and Autonomous Systems Center at USC, and with industry (e.g., Toyota, United Technologies Inc.) to identify appropriate scenarios in robotics, autonomous vehicles, and industrial automation, and corresponding testbeds to validate the proposed modeling infrastructure, methodology, and algorithms.

Broader Impact. In addition to disseminating the results through publications and workshops, we will collaborate with industry partners on transitioning the research findings into practice. Moreover, the theories, software, and hardware prototypes realized during this project will be leveraged for the development of an interdisciplinary education program targeting K-12, undergraduate, and graduate students, and focused on principled design and deployment of autonomous LE-CPSs as well as the impact of learning-enabled technologies on our life and safety. This effort will also include the development of a new course on autonomous CPS design together with the course textbook.

References

[1] Alur, R., Dang, T., and Ivančić, F. Counterexample-guided predicate abstraction of hybrid systems. *Theoretical Computer Science 354*, 2 (2006), 250–271.

[2] Alur, R., Henzinger, T., Lafferriere, G., and Pappas, G. Discrete abstractions of hybrid systems. *Proc. IEEE 88*, 7 (July 2000), 971–984.

[3] Blum, A. L., and Langley, P. Selection of relevant features and examples in machine learning. *Artificial intelligence 97*, 1 (1997), 245–271.

[4] Chutinan, A., and Krogh, B. Verification of infinite-state dynamic systems using approximate quotient transition systems. *IEEE Trans. Automatic Control 46*, 9 (Sep 2001), 1401–1410.

[5] Cousot, P., and Cousot, R. Abstract interpretation: a unified lattice model for static analysis of programs by construction or approximation of fixpoints. In *POPL* (1977), ACM, pp. 238–252.

[6] Dang, T., Maler, O., and Testylier, R. Accurate hybridization of nonlinear systems. In *Proc. Hybrid Systems: Computation and Control* (New York, NY, USA, 2010), HSCC '10, ACM, pp. 11–20.

[7] Frehse, G. PHAVer: algorithmic verification of hybrid systems past HyTech. *Int. J. Software Tools for Technology Transfer 10* (2008), 263–279.

[8] Henzinger, T., Ho, P.-H., and Wong-Toi, H. Algorithmic analysis of nonlinear hybrid systems. *IEEE Trans. Automatic Control 43*, 4 (Apr. 1998), 540–554.

[9] Jia, Y., Shelhamer, E., Donahue, J., Karayev, S., Long, J., Girshick, R., Guadarrama, S., and Darrell, T. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia* (2014), ACM, pp. 675–678.

[10] Kalra, N., and Paddock, S. M. Driving to safety: How many miles of driving would it take to demonstrate autonomous vehicle reliability? *Transportation Research Part A: Policy and Practice 94* (2016), 182–193.

[11] Malik, S., and Zhang, L. Boolean satisfiability from theoretical hardness to practical success. *Communications of the ACM 52*, 8 (2009), 76–82.

[12] Michalski, R. S., Carbonell, J. G., and Mitchell, T. M. *Machine learning: An artificial intelligence approach*. Springer Science & Business Media, 2013.

[13] Nuzzo, P., Sangiovanni-Vincentelli, A., Bresolin, D., Geretti, L., and Villa, T. A platform-based design methodology with contracts and related tools for the design of cyber-physical systems. *Proc. IEEE 103*, 11 (Nov. 2015).

[14] Passerone, R., Burch, J. R., and Sangiovanni-Vincentelli, A. L. Refinement preserving approximations for the design and verification of heterogeneous systems. *Formal Methods in System Design 31*, 1 (Aug 2007), 1–33.

[15] Sangiovanni-Vincentelli, A., Damm, W., and Passerone, R. Taming Dr. Frankenstein: Contract-Based Design for Cyber-Physical Systems. *European Journal of Control 18-3*, 3 (2012), 217–238.

[16] Shoukry, Y., Nuzzo, P., Sangiovanni-Vincentelli, A. L., Seshia, S. A., Pappas, G. J., and Tabuada, P. SMC: Satisfiability modulo convex optimization. In *Proc. Hybrid Systems: Computation and Control* (2017), pp. 19–28.

[17] Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199* (2013).