CAREER: SOISTICe: Software Synthesis with Timing Contracts for Cyber-Physical Systems

CCF-1553757, 1/15/2016 – 12/31/2020, Qi Zhu, University of California, Riverside

Timing Challenges in Software Synthesis

- Timing has critical impacts on functional correctness and various design metrics (e.g., safety, security, control performance, extensibility, fault tolerance) in CPS.
- Synthesis of CPS software faces timing-related challenges:
  - ◇ **Diversity of timing requirements** from different design metrics,
  - ◇ **Complexity of timing analysis** under complex scale, hierarchy and concurrency of computation and communication,
  - ◇ **Uncertainty of timing behavior** from dynamic environment and data input.
- Timing constraints are often set in an ad-hoc fashion.
- Software synthesis is often conducted without holistic consideration of timing.

SOISTICe Framework

Theme A: Co-design and Design Refinement with Timing Contracts

A1. Multi-metric Co-design with Horizontal Timing Contracts Exploration
- *How to set timing constraints when considering multiple conflicting design metrics?*
- Identify critical timing factors for co-design and choose the right formalisms.
- Analyze how timing constraints and design variables affect system properties.
- Design co-design algorithms for exploring timing constraints and design variables.

A2. Hierarchical Design Refinement with Vertical Timing Contracts Exploration
- *How to assign timing "budget" for lower-level components during refinement?*
- Represent timing behavior and constraints across system hierarchy.
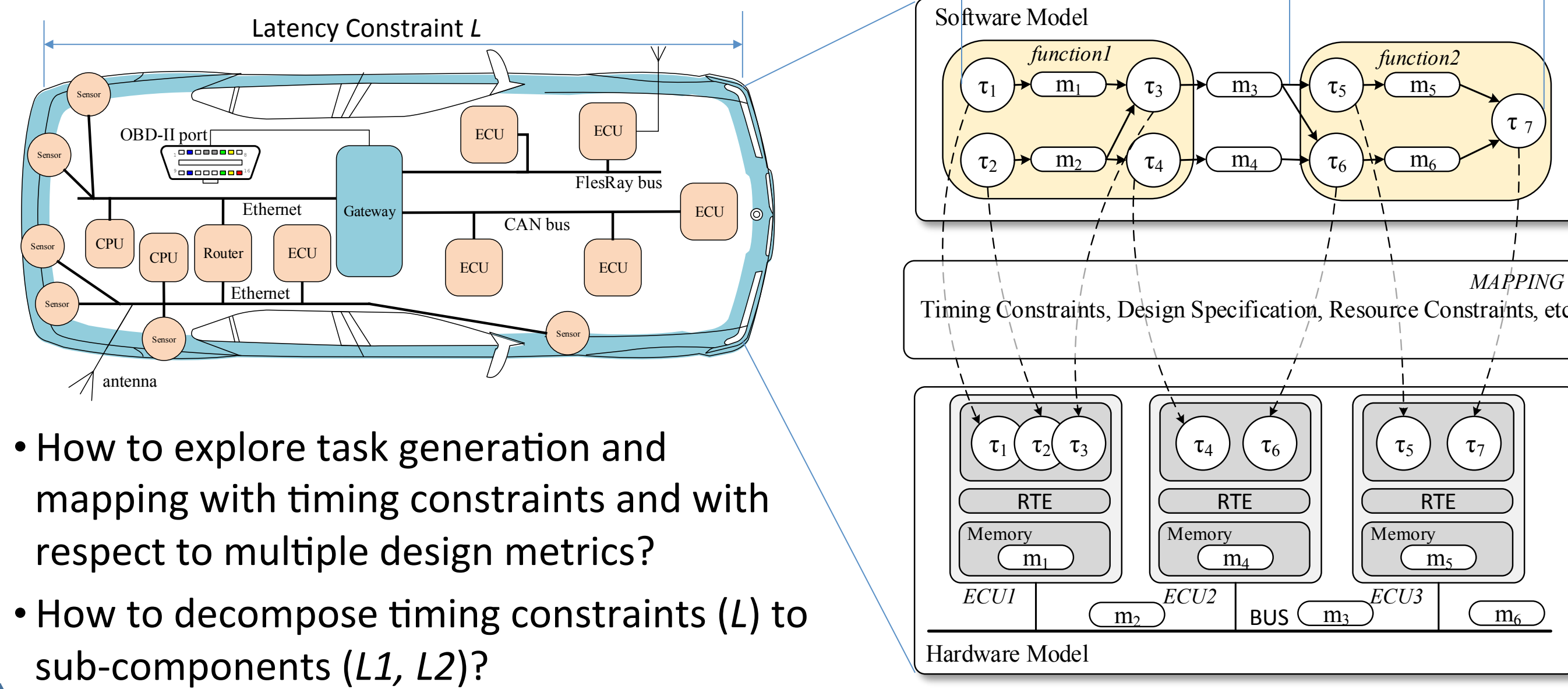- Efficiently estimate the timing complexity of subcomponents.

Theme B: Timing-centric Holistic Task Generation and Mapping

- *Given timing constraints, can we find feasible task generation and mapping? If yes, how good can the solutions be? If not, where are the bottlenecks and can we adjust certain timing constraints to produce feasible solutions?*
- Develop an **interactive task synthesis** approach: 1) quick assessment of feasibility and identification of design bottlenecks, 2) partial synthesis under incomplete constraints, 3) additive synthesis under updated constraints.
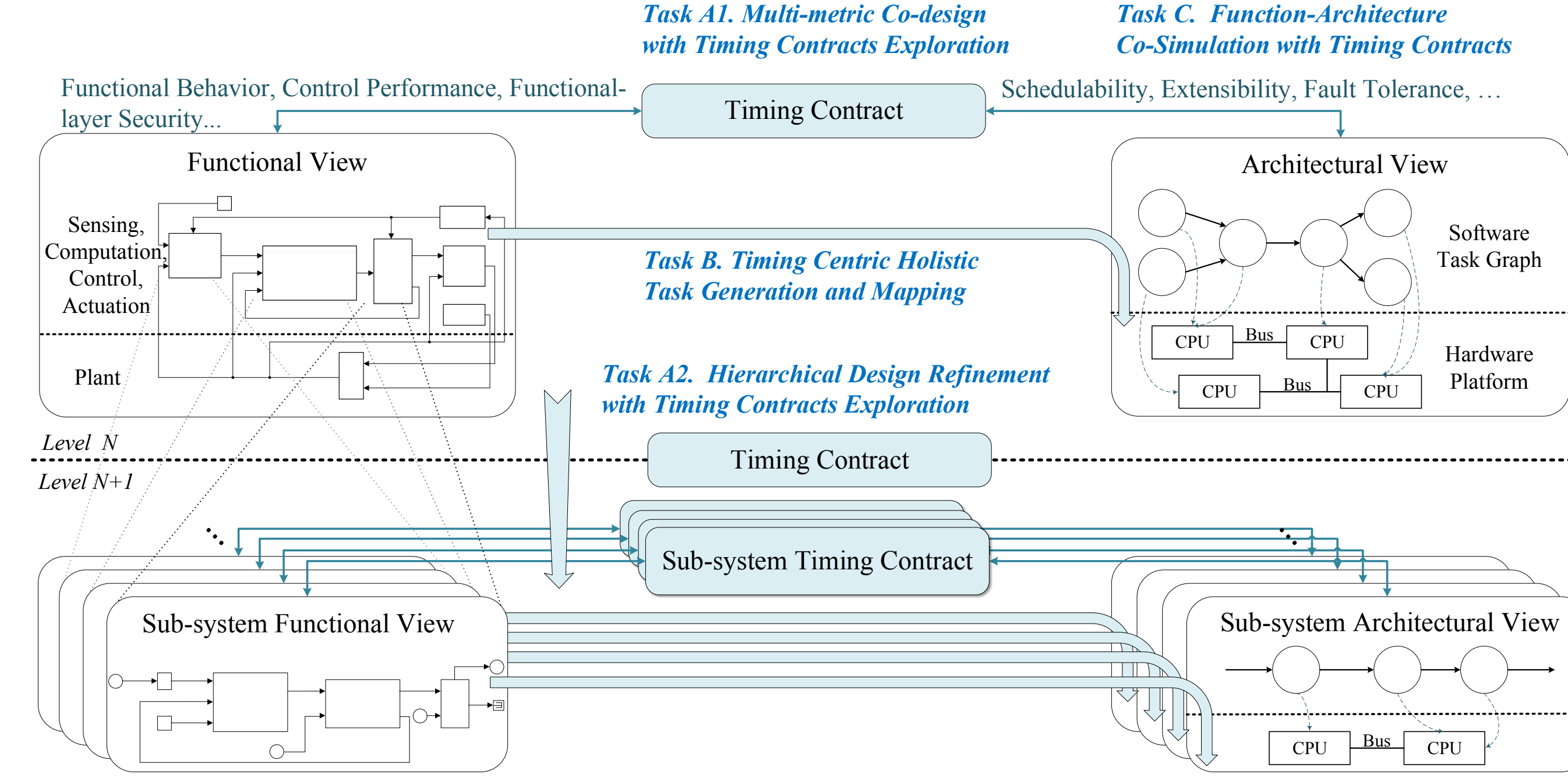- Task synthesis of heterogeneous and hierarchical functional models.

Theme C: Function-Architecture Co-simulation with Timing Contracts

- Timing contracts modeling and monitoring during co-simulation.
- Explicit and modular representation of task synthesis options.
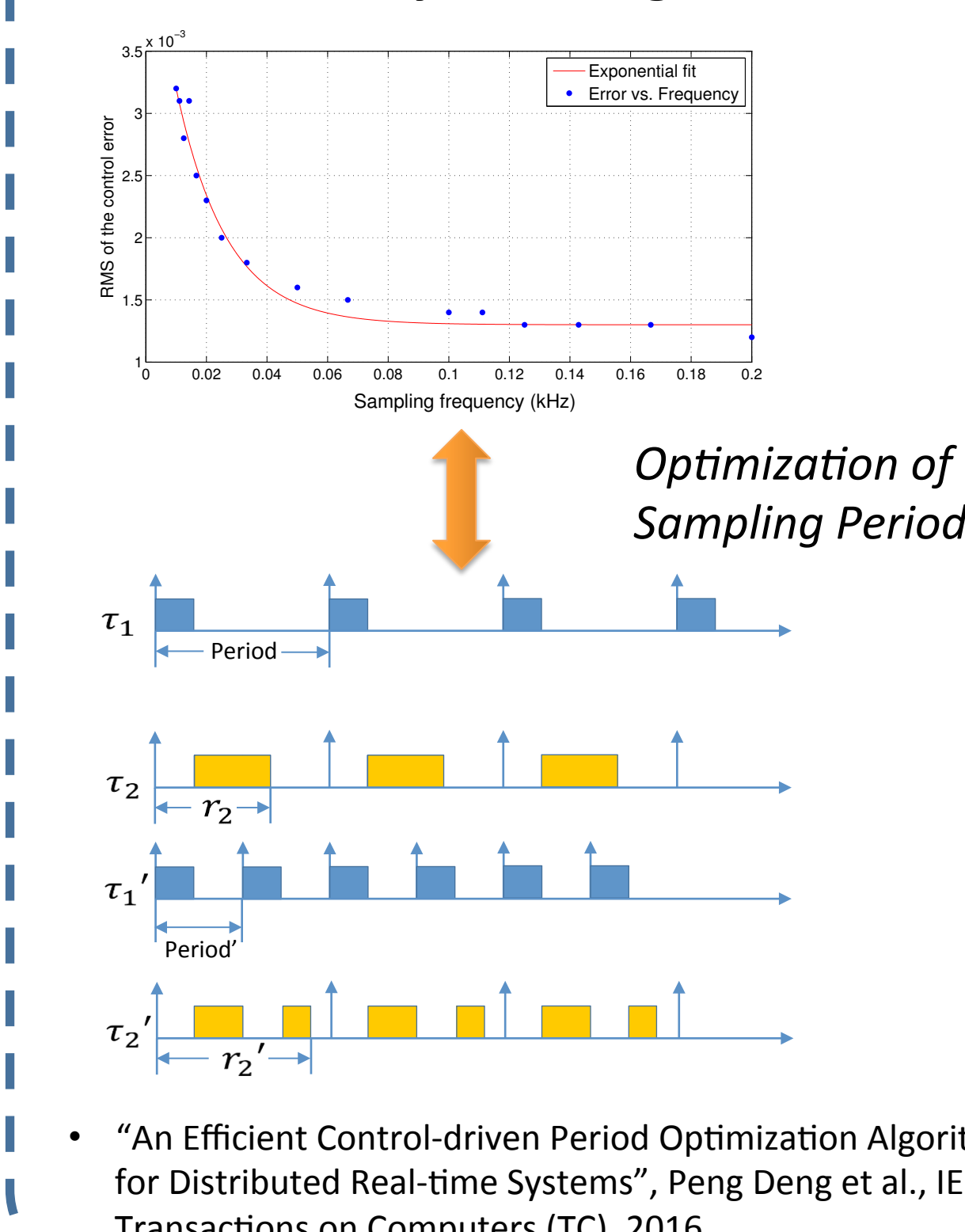- Integration of simulator and analytical algorithms.
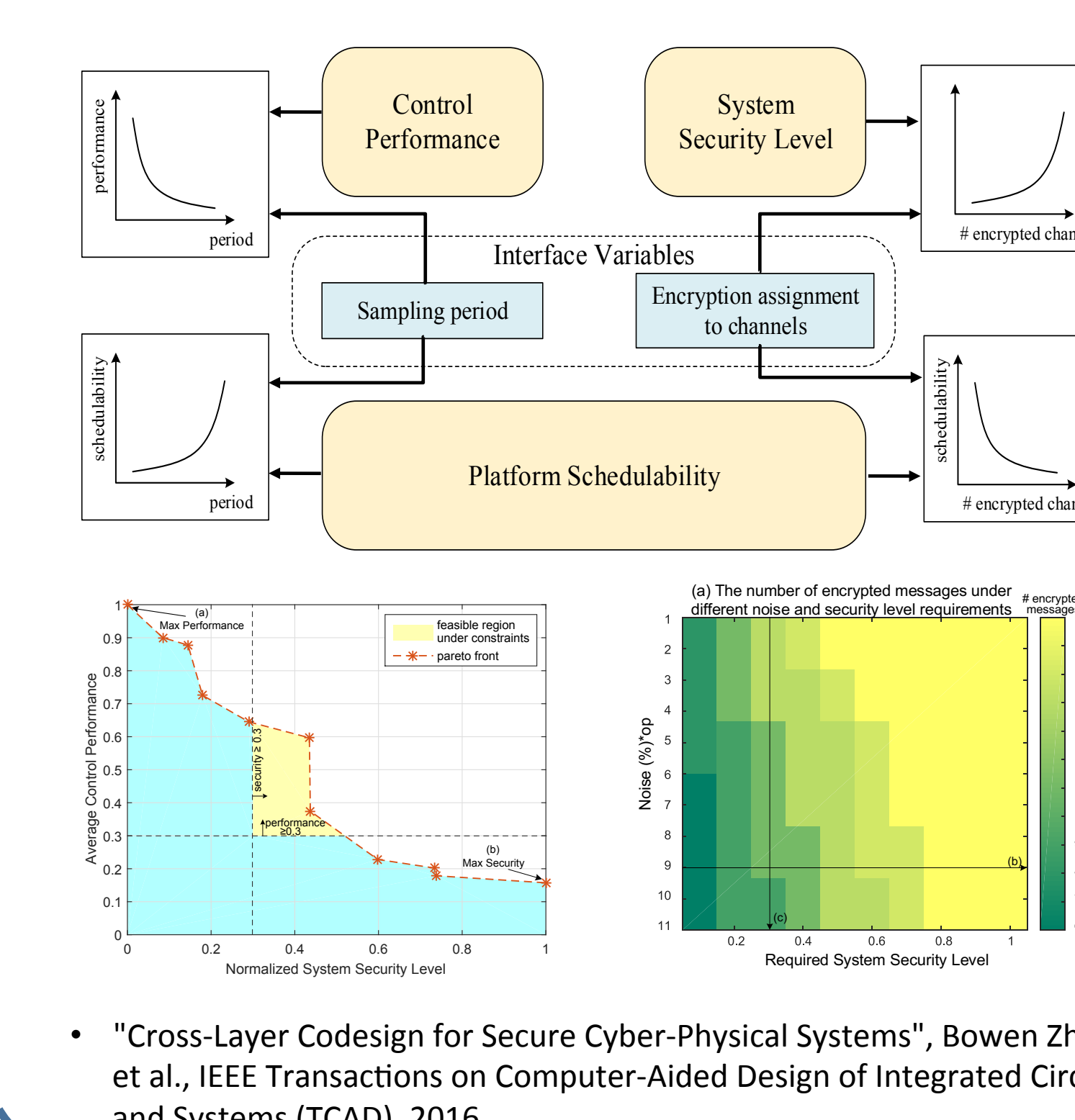
Software Architecture Synthesis

- How to explore task generation and mapping with timing constraints and with respect to multiple design metrics?
- How to decompose timing constraints (L) to sub-components ($L1, L2$)?

SOISTICe Framework

Control Performance and Schedulability Co-design

Optimization of Sampling Period

- "An Efficient Control-driven Period Optimization Algorithm for Distributed Real-time Systems", Peng Deng et al., IEEE Transactions on Computers (TC), 2016.

Control Performance, Security, and Schedulability Codesign

- "Cross-Layer Codesign for Secure Cyber-Physical Systems", Bowen Zheng et al., IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD), 2016.

CONVINCE Framework for Vehicular Networks Design

- "CONVINCE: A Cross-Layer Modeling, Exploration and Validation Framework for Next-Generation Connected Vehicles", Bowen Zheng et al., IEEE/ACM International Conference on Computer-Aided Design (ICCAD), 2016.
- "Next Generation Automotive Architecture Modeling and Exploration for Autonomous Driving", Bowen Zheng et al., IEEE Computer Society Annual Symposium on VLSI, 2016.

Scientific Impacts

- Explore timing constraints quantitatively throughout the software synthesis process to produce correct, efficient, and predictable CPS software implementation.
- Develop new methodologies for timing contracts definition and exploration, novel algorithms for timing-centric task generation and mapping, and a simulator with explicit timing contracts evaluation.
- Use automotive and transportation systems as primary case studies and provide new tools for automotive software development.

Broader Impacts and Education

- Enable fundamental advances in design automation for cyber-physical systems.
- Establish close industry collaborations and facilitate potential technology transfer.
- Leverage research findings to build an interdisciplinary education program for K-12, undergrad, and graduate students: 1) outreaching to K-12 schools with Lego Mindstorm labs development, 2) extending undergrad embedded systems course and advising senior design projects, 3) developing new graduate course on CPS, and 4) writing a textbook in collaboration with industry.

PI: Qi Zhu (qzhu@ece.ucr.edu)

Industry Collaborators: TOYOTA InfoTechnology Center, U.S.A., Inc.   United Technologies Research Center   SRI International   NATIONAL INSTRUMENTS

UNIVERSITY OF CALIFORNIA UC RIVERSIDE