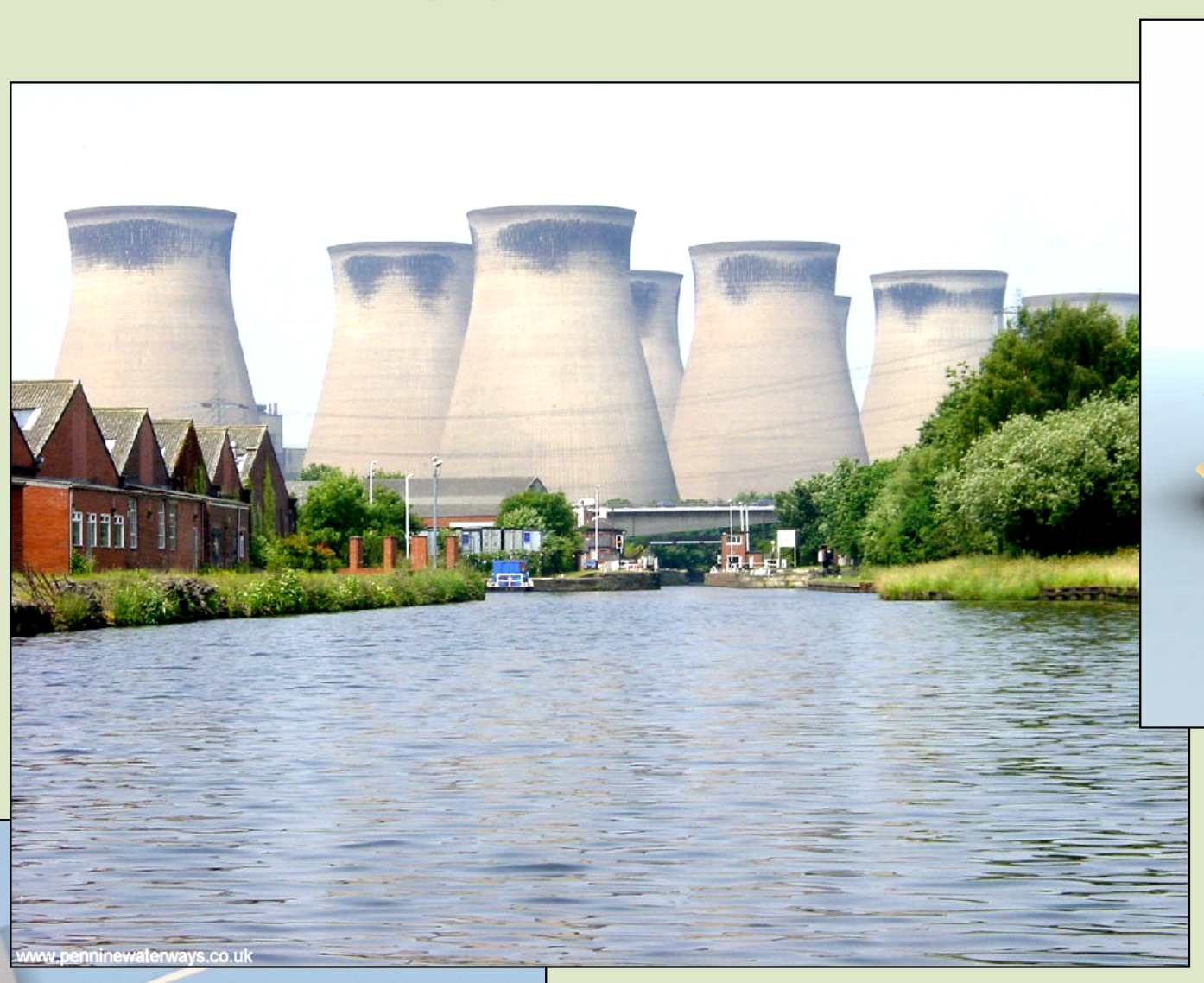
The Ectokernel Approach: A Composition Paradigm for Building Evolvable Safety-critical Systems from Unsafe Components

Tarek Abdelzaher (PI), Marco Caccamo, Lui Sha

Department of Computer Science, University of Illinois

Safety-Critical Applications



Power

Automotive

Smart Cities

Medical

Mass Transit

Avionics

Agriculture

Future CPS

Increased size and automation
Computational intractability to verify all code
Increased criticality, coupling and potential for failure cascades
Increased societal dependence and more dramatic consequences of software failure

Challenge: How to build software for safety-critical CPS where majority of code cannot be verified?

Preliminaries

Motivation

- Interactive complexity increasingly plagues the design and execution of large systems
- High complexity generates unexpected interaction patterns and hard-to-find bugs
- Building more reliable systems entails taming interactive complexity
- This project offers:
  - Architectural support for reducing interactive complexity
  - Algorithms for verifying safety guarantees in the presence of unverified code
  - Algorithms for diagnosing root causes of performance problems

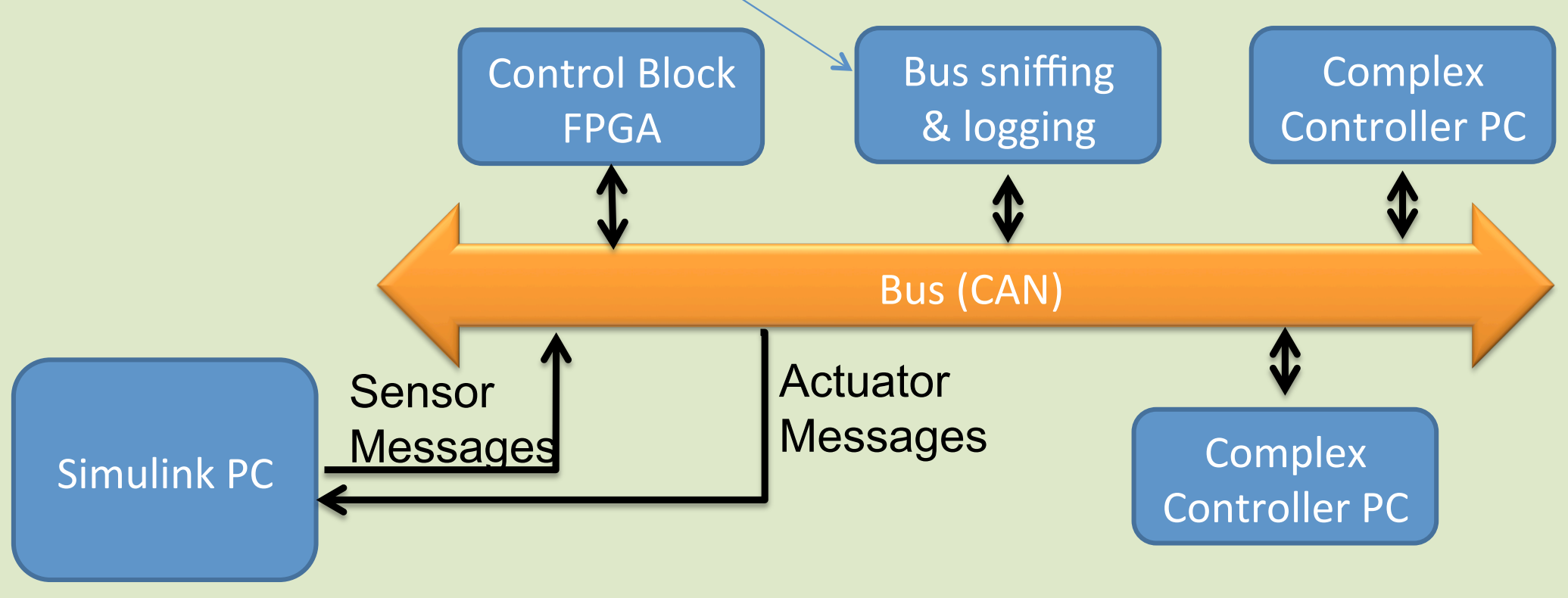
Main Ideas

- When safety properties are about to be violated, a verified control block performs "fail-over" to a verified component, disconnecting appropriate unverified ones.
- Physical connections (e.g., power grid distribution buses) in large scale CPS provide "fail-safe" remote state estimation for running local safety controllers
- A diagnostic system troubleshoots the unverified component problems
- Challenges:
  - Design of the switch logic in the "fail-over" control block
  - Efficient verification of core functions used for fail-over
  - Diagnosing root causes of bugs in the remaining unverified code

Safety

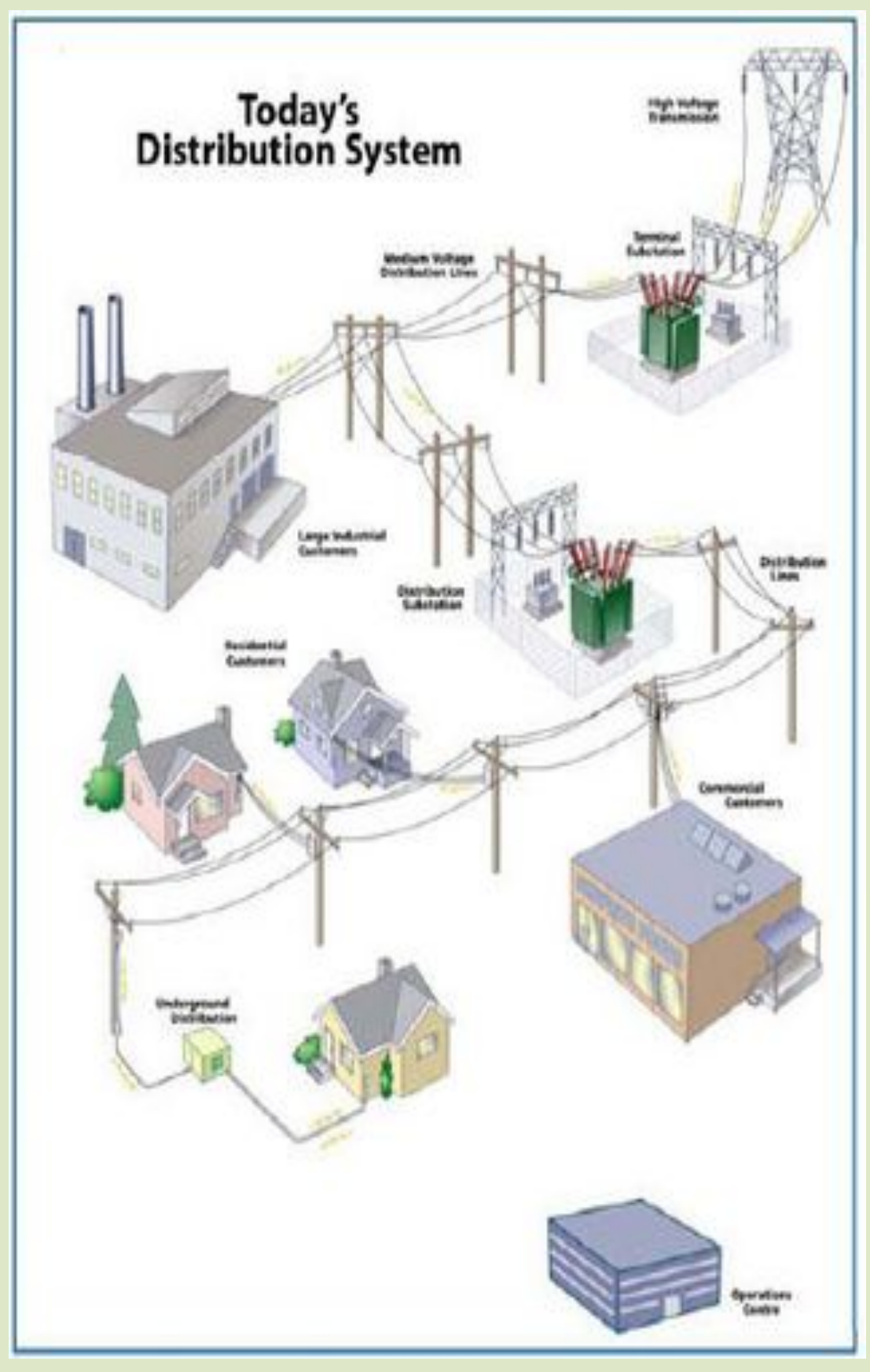
Emulation Architecture

- Bus sniffer allows to identify and locate hard to debug faults like transient frame loss due to a bad connector
- Sniffer supports differential and single ended receivers to identify transient faults
- Data logging is triggered by a specified event (like loss of differential signal on the bus)



Emulation Testbed

- Plant is simulated in Simulink Stateflow and the cyber-architecture and controllers run on physical hardware
- We can easily simulate noise and sensor failure
- Changing the plant model is easy
- We can use the proposed emulation environment for rapid prototyping, evaluation, and testing



Fault Resilient Architecture for Distributed CPS

The power grid, water distribution networks, etc. are all examples of distributed CPS characterized by a physical connection graph and a cyber one. When one or more cyber connections fail, physical connections (e.g., power grid distribution buses) can be used to estimate remote state variables and run a local safety controller.

Networked Control Switch logic:

The logic that switches controllers periodically compares cyber data (remote state variables) with their estimated values based on local physical measurements and the stability envelope in a networked control system.

More details:

F. Abdi, B. Robbins, and M. Caccamo, "A Fault Resilient Architecture for Distributed Cyber-Physical Systems", Proceedings of the IEEE conference RTCSA'12, Seoul, Korea, August 2012.

Jianguo Yao, Xue Liu, Guchuan Zhu, and Lui Sha, "NetSimplex: Controller Fault Tolerance Architecture in Networked Control Systems", accepted for publication, IEEE Transaction on Industrial Informatics

Diagnostics

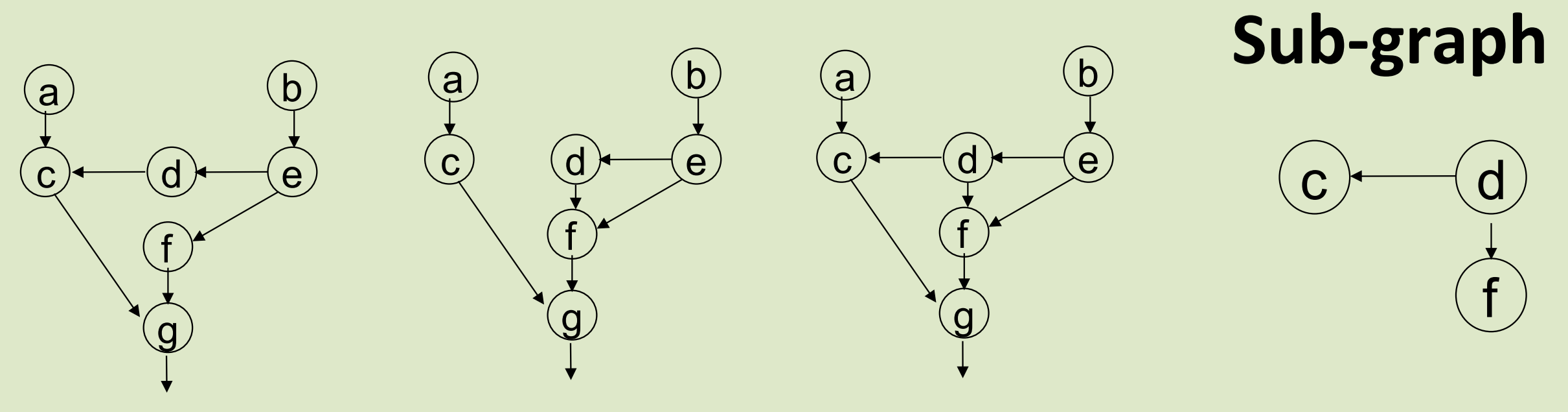
Log Collection

- For each reported event, log event ID and parameters
- Label distributed log as "well-behaved" or "bad" depending on whether errors were manifested

Discriminative Analysis

- Analyze the logs to identify the "discriminative" event sub-graphs causally correlated with "bad" behavior
- Report discriminative sub-graphs as probable causes of error

Example **Discriminative Sub-graph**



(a) Well-behaved (b) Well-behaved (c) Bad

More details:

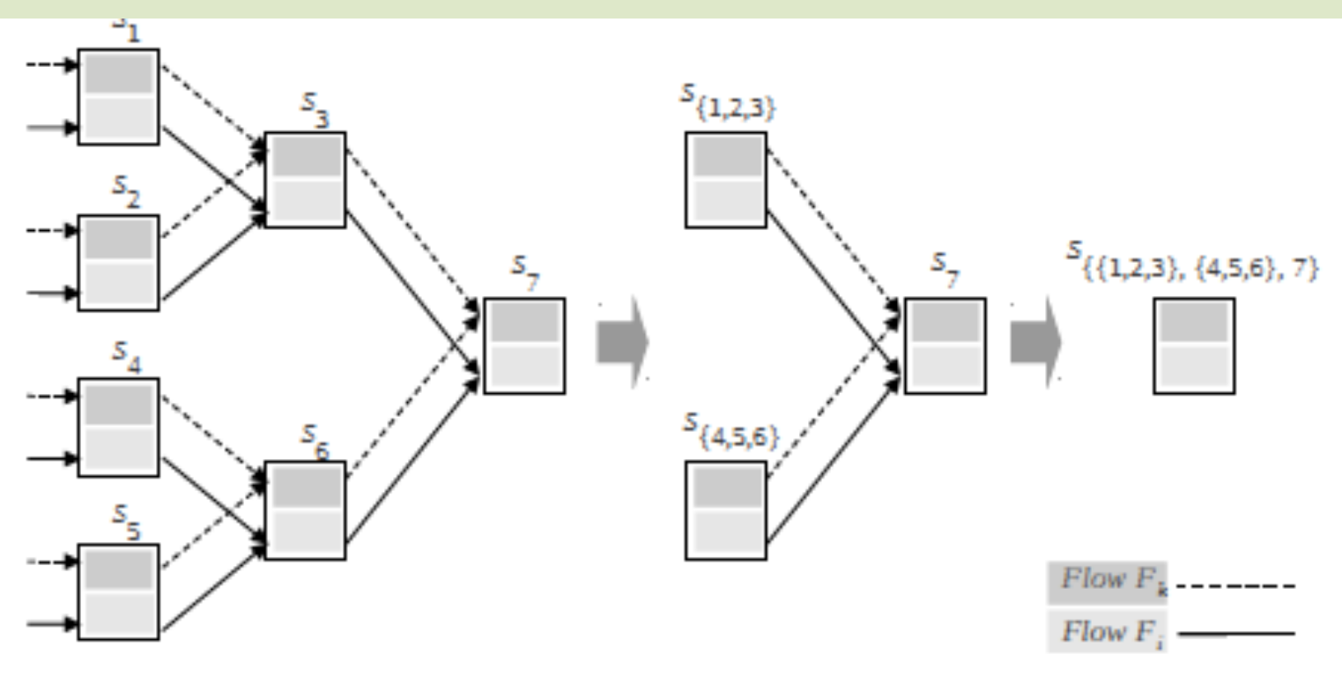
Eunsoo Seo, Mohammad Maifi Hasan Khan, Prasant Mohapatra, Jiawei Han and Tarek Abdelzaher, 'Exposing Complex Bug-Triggering Conditions in Distributed Systems via Graph Mining,' International Conference on Parallel Processing (ICPP), Taipei, Taiwan, September 2011.

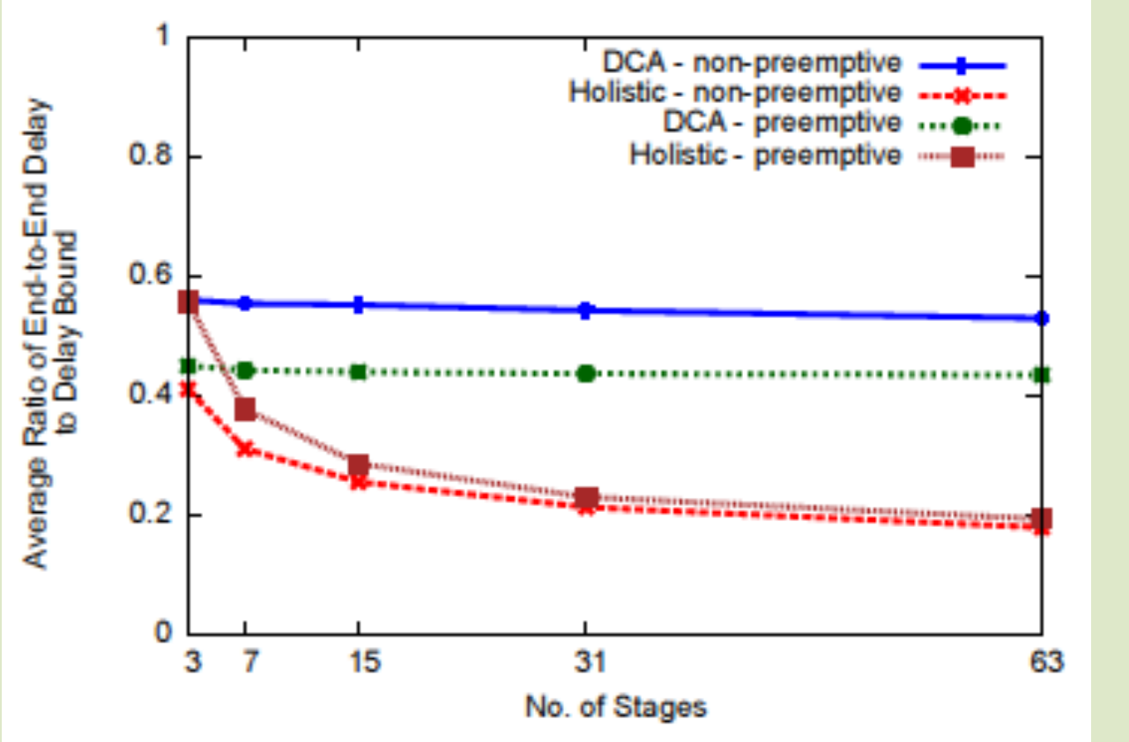
Jin Heo, Praveen Jayachandran, Insik Shin, Dong Wang, Tarek Abdelzaher, Xue Liu, 'OptiTuner: On Performance Composition and Server Farm Energy Minimization Application,' IEEE Transactions on Parallel and Distributed Systems, Vol. 22, No. 11, November 2011.

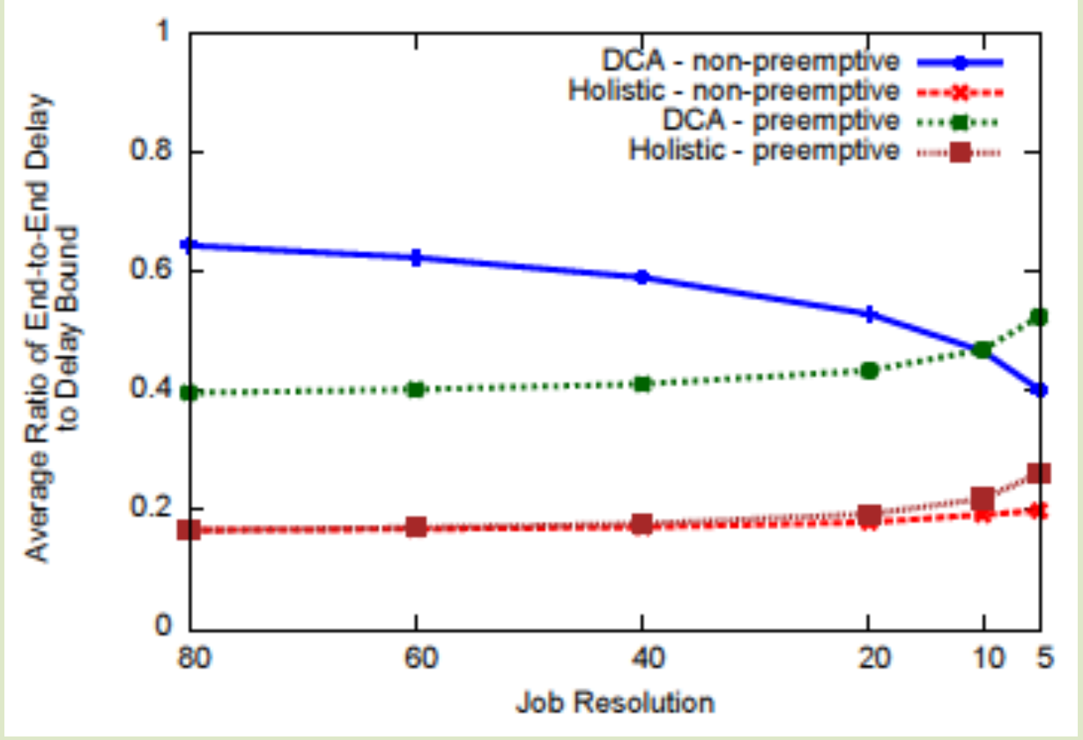
Schedulability Analysis

Schedulability analysis of distributed systems

- A compositional approach (originally developed under prior NSF funding)
- New: Tight capacity derivation for resource pipelines and data fusion topologies (extensions of utilization bounds)







More details:

Fatemeh Saremi, Praveen Jayachandran, Forrest Iandola, Yusuf Sarwar, Tarek Abdelzaher, Aylin Yener, 'On Schedulability and Time Composability of Multisensor Data Aggregation Networks,' In Proc. 15th International Conference on Information Fusion, Singapore, July 2012

Conclusions

- Proper architecture obviates the need to verify all code
- Unsafe conditions brought about by unexpected component interactions can be neutralized by cutting interaction chains that include culprit unverified code when safety is about to be violated
- Analysis of discriminative features in execution graphs (and hence, interaction patterns) can reveal root causes of bad behavior
- Analysis of discriminative features in execution graphs (and hence, interaction patterns) can reveal root causes of bad behavior
- The cyber system fault induced physical system instability is stablized by watch and controller switching using the stability envelopes of local subsystem and the networked control subsystem
- Status: Testbed is constructed, architecture is conceptualized, algorithms for verification and failover are derived, diagnostic discriminative mining tools are developed