Foundations of CPS: Real-Time Scheduling

Sanjoy Baruah
Washington University in St. Louis
baruah@wustl.edu



Foundations of CPS:
Real-Time Scheduling

The **assignment** of **resources** to **activities over time**, subject to **constraints**, to **optimize** for specified objectives

Real-time scheduling theory is **application-driven**

Initially, **simple** and **closed** applications with **complete specifications**

The emergence of CPS

Jane Liu

Ted Baker



Th

to

Re

Ini

Th

CPS-Driven Scheduling Challenges

More complex software structure

More complex implementation platforms

Differentiated assurance

CPS-Driven Scheduling Challenges

More complex software structure

More complex implementation platforms

Differentiated assurance

Directed Acyclic Graph-based models

Demand elasticity

Self-suspension in threads

CPS-Driven Scheduling Challenges

More complex software structure

More complex implementation platforms

Differentiated assurance

Directed Acyclic Graph-based models; conditional DAGs

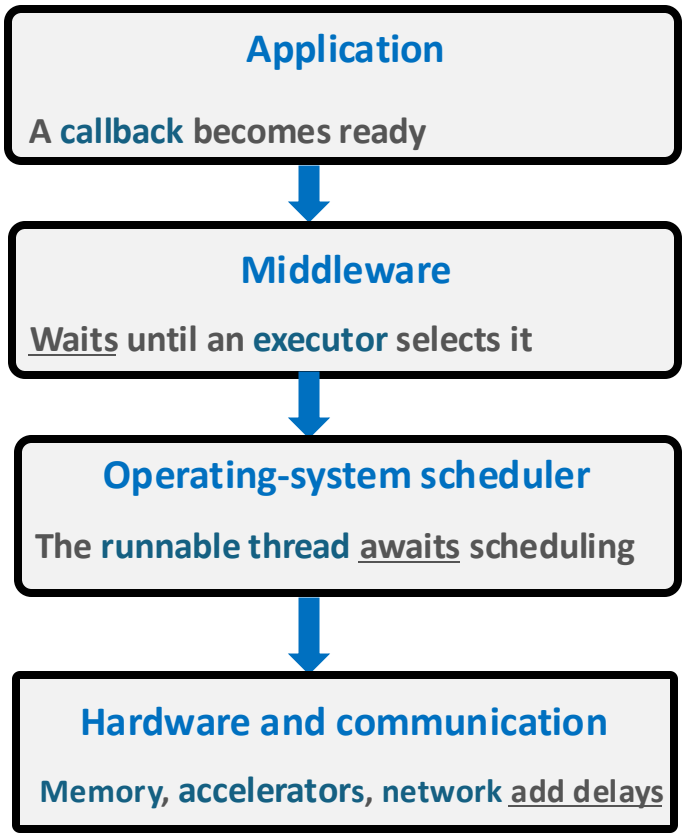
Demand elasticity

Self-suspension in threads

Parallel and heterogeneous platforms

May be distributed and networked

Access mediated by middleware



CPS-Driven Scheduling Challenges

More complex software structure

More complex implementation platforms

Differentiated assurance

Directed Acyclic Graph-based models; conditional DAGs

Demand elasticity

Self-suspension in threads

Parallel and heterogeneous platforms

May be distributed and networked

Access mediated by middleware

Integrated system architectures

Probabilistic specification of constraints

CPS-Driven Scheduling Challenges

More complex software structure

More complex implementation platforms

Differentiated assurance

Directe

Demar

Self-su

TAKE-AWAY

Richer models permit higher-fidelity representation.

Multiple notions of correctness are available.

Harder analysis problems have been solved.

Parallel and heterogeneous platforms

May be distributed and networked

Access mediated by middleware

Integrated system architectures: mixed criticalities

Probabilistic specification of constraints

Looking to the future…

[CPS 2.0]

The Impact of Agentic AI

Agents as assistants

Björn Brandenburg's PROSA Project

- a Rocq formalization of real-time scheduling theory
- Agentic AI tools to grow the Prosa corpus
- Agentic AI tools to automate the formalization of a paper proof
  - ProsaBuddy, PROVE-RT, …

The Impact of Agentic AI

Agents as assistants: the Formalization of RT Scheduling Theory

Agents as collaborators

A conjecture in real-time scheduling theory

AN EXAMPLE

Uniprocessor EDF schedulability verification is coNP complete

Conjecture: partitioned EDF schedulability verification is NP^{NP} complete

- Claude (Opus 4.8) failed to prove or refute this
- June: A human-generated proof (Nathan Fisher & Chris Harbourne)
- July 1st: Claude Fable 5 becomes available
- July 2nd: Claude Fable 5 obtains a proof

The Summing Up

CPS 1.0 (2006 – 2026)

The strong foundations of classical Real-Time Scheduling Theory have been broadened to meet CPS needs

CPS 2.0 (2026 –)

AI agents as assistants and collaborators

AI agents as oracles

An agent

reads system specifications

identifies properties needed for correctness

and provides a proof

What role remains for the human expert?

Foundations of CPS: Real-Time Scheduling

Collaborators include

Björn Brandenburg (MPI-SWS)
Pontus Ekberg (Uppsala)
Nathan Fisher (Wayne State)
Nan Guan (City U. Hong Kong)
Zhishan Guo (NC State)
Bryan Ward (Vanderbilt)

RTSS 2026 HOT TOPICS DAY WORKSHOP

Generative AI for Real-Time Scheduling Research

Discovery, Proof, Verification, and Research Integrity

December 8, 2026 · Yokohama, Japan · Full-day workshop