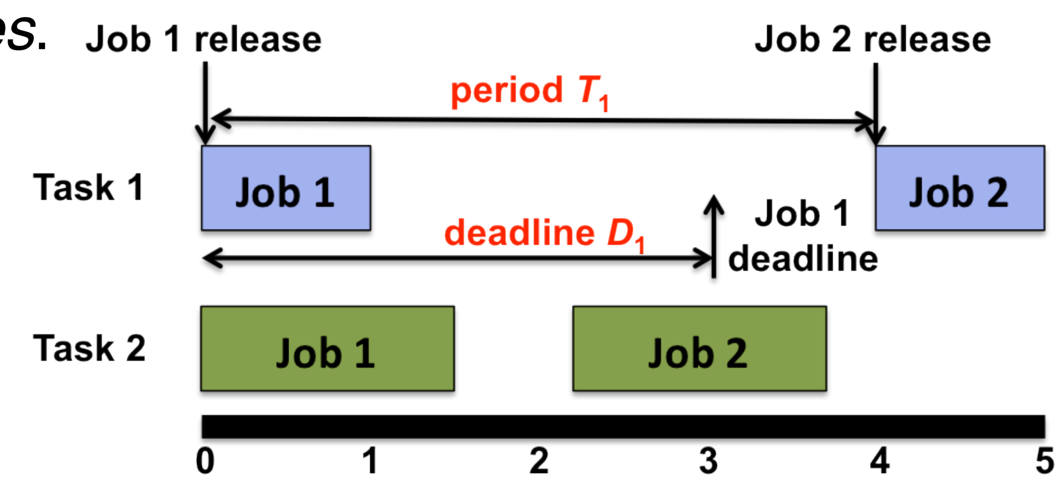
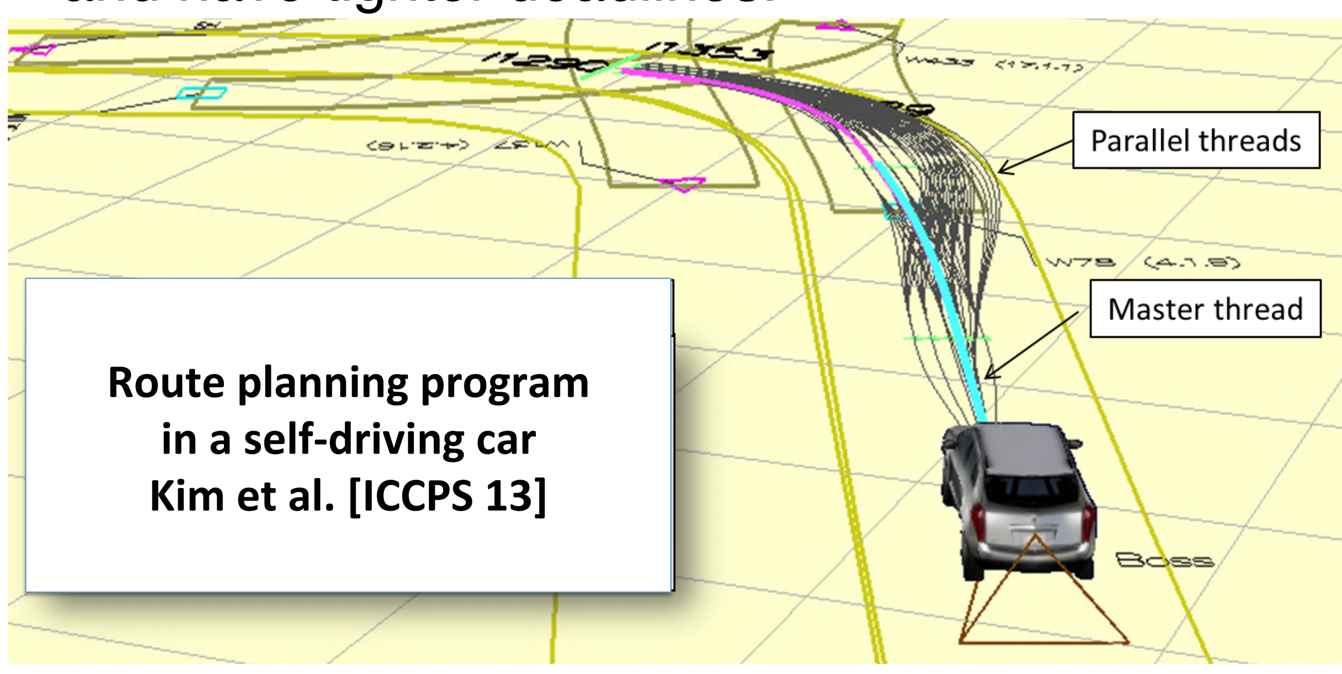
XPS: Real-Time Scheduling of Parallel Tasks

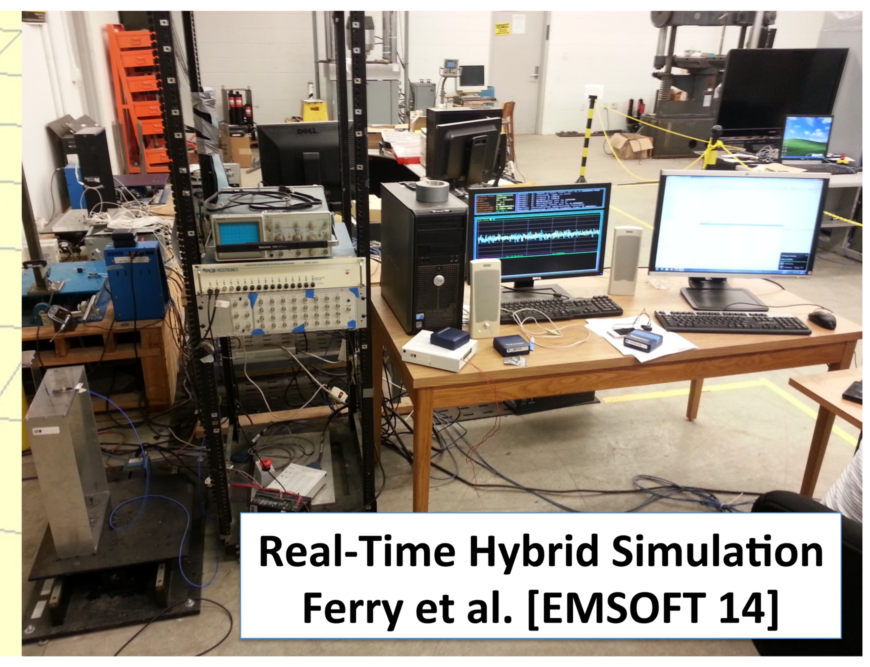
Motivation: Real-Time Systems Are Everywhere

- Real-time tasks are tasks with *deadlines*.
- They arise when computation interacts with physical world.
- Modern real-time applications are more computationally intensive and have tighter deadlines.

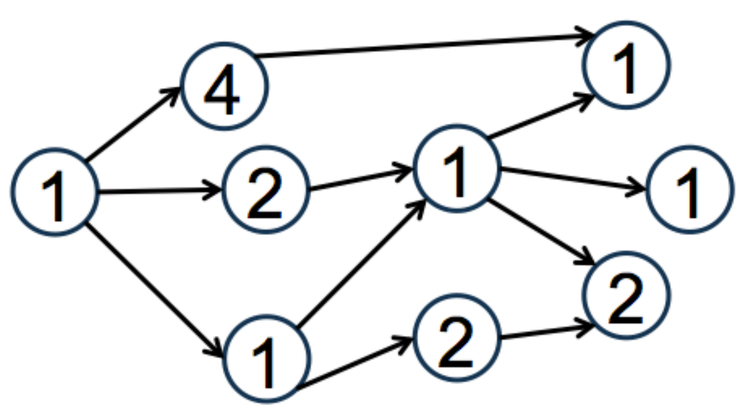




Route planning program in a self-driving car Kim et al. [ICCPS 13]


Real-Time Hybrid Simulation Ferry et al. [EMSOFT 14]

Goal: Parallel Real-Time Systems



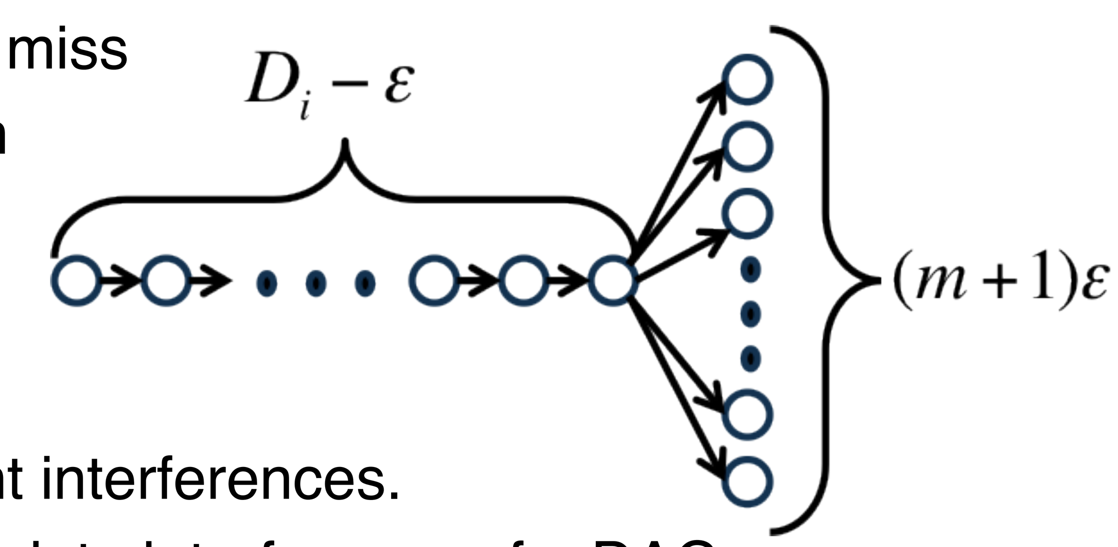
$C_i = 15$
$L_i = 6$
$D_i = 10$
$u_i = 1.5$

Execution time C_i
Critical-path length L_i
Utilization $u_i = C_i / D_i$

- Develop theoretical foundations and system implementations to support parallel real-time tasks.
- Task model: Directed Acyclic Graph (DAG).
- Nodes represent a sequence of instructions and edges represents dependences between nodes.

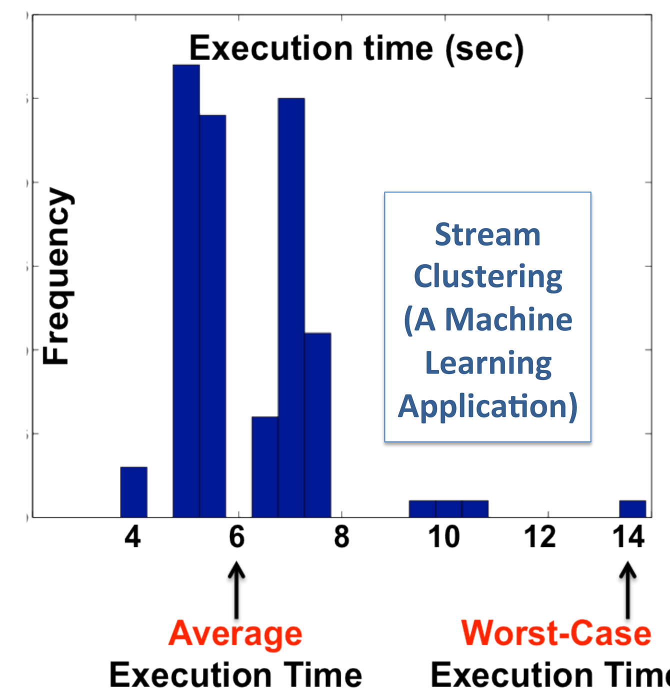
Challenges: Analyze Interference of DAG Tasks

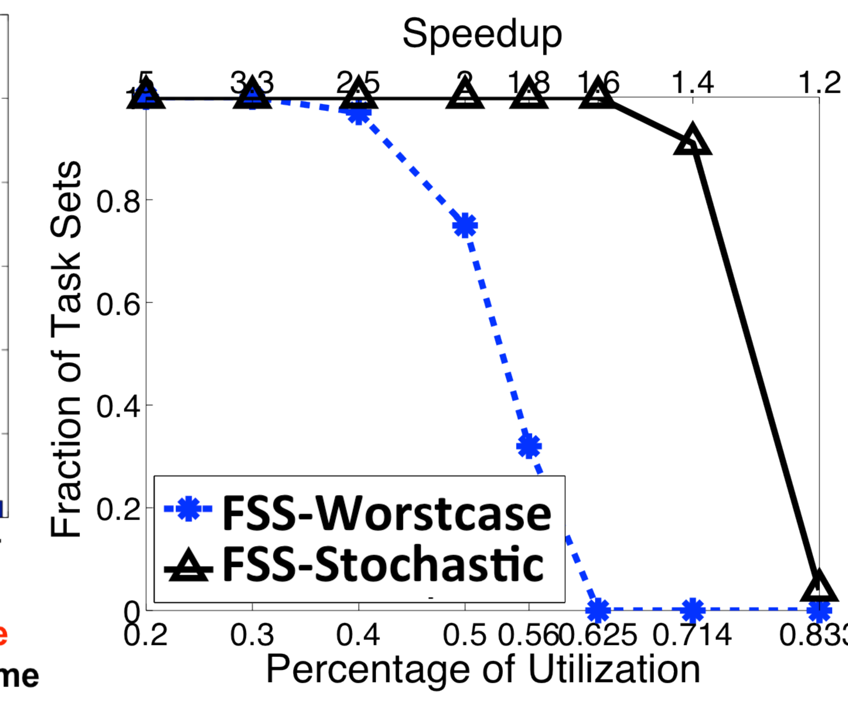
- Dhall's effect: There exist tasks which miss their deadline on *m* cores even though their utilizations are very close to 1.
- A task can generate jobs with different DAG structures.
- Different DAG structures have different interferences.
- It is computationally expensive to calculate interference of a DAG.



Stochastic Parallel Real-Time Tasks

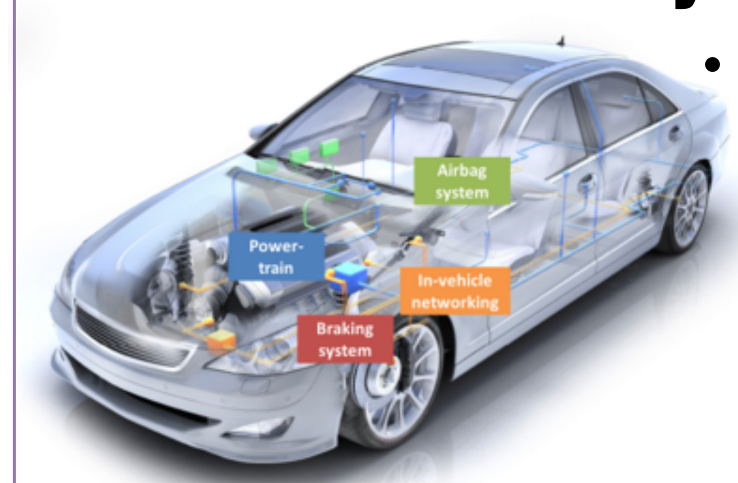
- If the task parameters have variation from one instance to another, we can model them as random variables and use the mean values for scheduling decisions.





- We generalized federated scheduling for this setting and proved the capacity augmentation bound of 2 [2].

Mixed-Criticality Parallel Tasks



- In autonomous cars, GPS navigation system has lower criticality than braking systems, yet both coexist in one "mixed-criticality" system.
- In normal state, we want to schedule both tasks based on their nominal workload, while in safety-critical state we need to guarantee braking system can still meet its deadline even with the pessimistic worst-case workload.
- Mixed-criticality federated scheduling algorithm has a capacity augmentation bound of 3.732 [6].

Work In Progress and Future Work

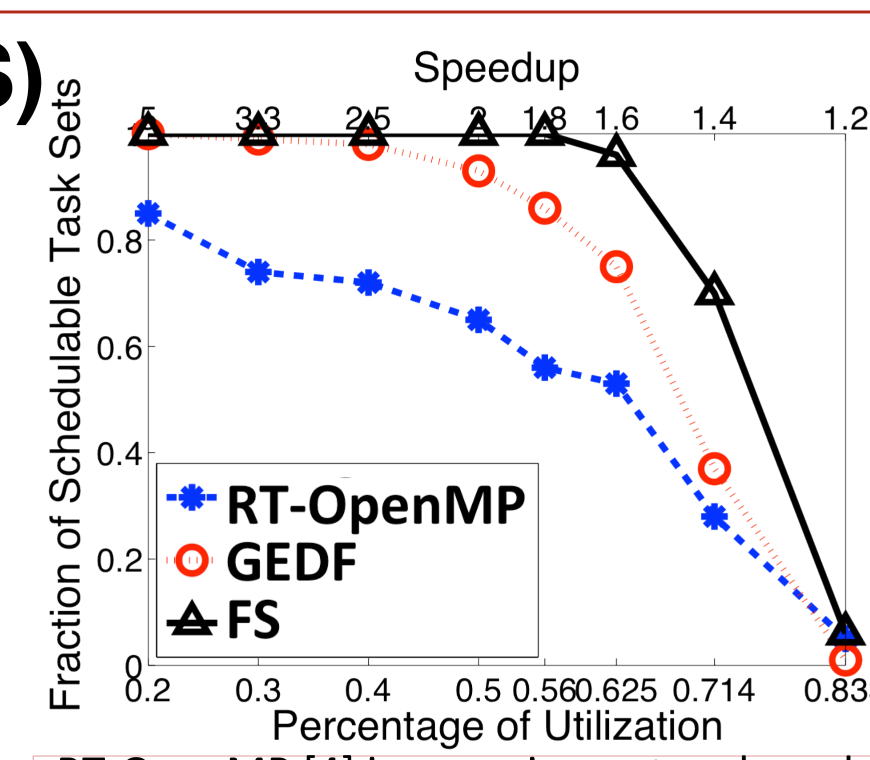
- Handling resource contention and critical sections.
- Providing soft real-time guarantees. e.g. tardiness.
- Analyzing and scheduling aperiodic tasks

Theoretical Bounds

- *Resource augmentation bound* of *b*: If an ideal scheduler can schedule a task set on *m* cores of speed 1, then scheduler *S* can schedule that task set on *m* cores of speed *b*
- *Capacity augmentation bound* of *b*: If a task set has total utilization of at most *m*/*b*, and each task has its critical-path length of at most 1/*b* of its deadline, then scheduler *S* that can schedule the task set on *m* cores.

Global Earliest Deadline First (GEDF) and Federated Scheduling (FS)

	Scheduling Algorithm	Resource Bound	Capacity Bound
GEDF [1, 5]	Global scheduler. A job can execute on all *m* cores. Prioritizes jobs with the earliest deadline. (Core 1, Core 2, Core 3, Core 4)	approximately 2 (upper & lower bound)	approximately 2.6 (upper & lower bound)
FS [1]	Partition-like $n_i = \left\lceil \frac{C_i - L_i}{D_i - L_i} \right\rceil$ (Core 1, Core 2, Core 3, Core 4)	approximately 2	approximately 2 (optimal)



RT-OpenMP [4] is our prior system based on decomposing parallel tasks.

Publications

1. J. Li, J.-J. Chen, K. Agrawal, C. Lu, C. Gill and A. Saifullah, Analysis of Federated and Global Scheduling for Parallel Real-Time Tasks, ECRTS, 2014
2. J. Li, K. Agrawal, C. Gill and C. Lu, Federated Scheduling for Stochastic Parallel Real-time Tasks, RTCSA, 2014.
3. D. Ferry, G. Bunting, A. Megareh, S. Dyk, A. Prakash, K. Agrawal, C. Gill and C. Lu, Real-Time System Support for Hybrid Structural Simulation, EMSOFT, 2014.
4. A. Saifullah, D. Ferry, J. Li, K. Agrawal, C. Lu and C. Gill, Parallel Real-Time Scheduling of DAGs, IEEE Transactions on Parallel and Distributed Systems, 2014.
5. J. Li, Z. Luo, D. Ferry, K. Agrawal, C. Gill and C. Lu, Global EDF Scheduling for Parallel Real-Time Tasks, RTS, 2015.
6. J. Li, S. Ahuja, K. Agrawal, C. Gill and C. Lu, Mixed-Criticality Federated Scheduling for Real-Time Parallel Tasks (in submission).

Kunal Agrawal, Christopher Gill, Chenyang Lu, {kunal, cdgill, lu}@wustl.edu

Washington University in St. Louis (Grant #: 1337218)
Project Dates: September 2013 – August 2017