Value-Aware Real-Time Scheduling for Intelligent Transportation Systems

Hoeseok Yang[1], Hokeun Kim[2], Choonghwan Lee[3], and Hyung-Chan An[4]

[1]Santa Clara University [2]Arizona State University [3]Moloco Inc. [4]Yonsei University

Time-Centric Reactive Software
October 2, 2025
Taipei, Taiwan

Outline

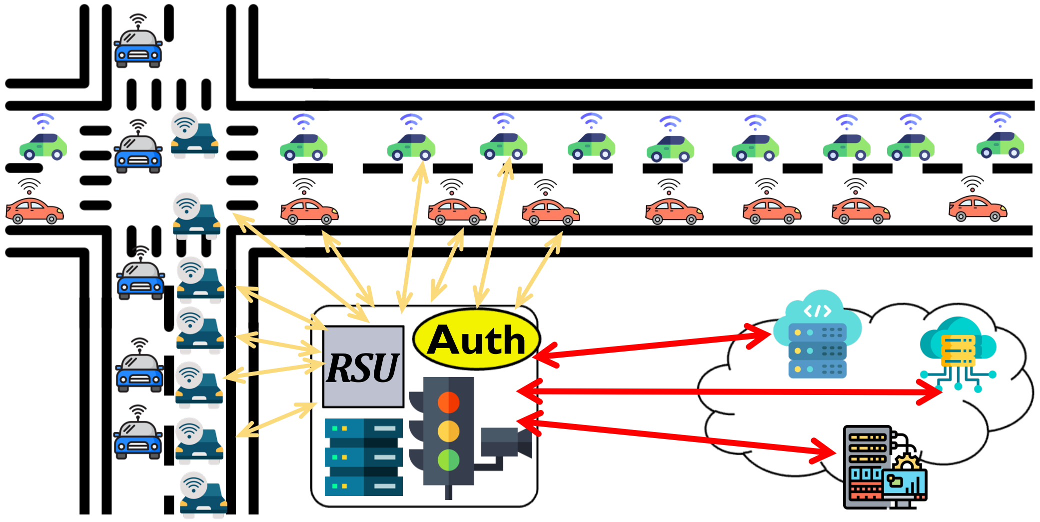
- Background and Motivation
- Previous Research
- Proposed Method
- Evaluation
- Summary and Conclusion

Outline

- Background and Motivation
  - Intelligent Transportation Systems (ITS)
  - Why Value-Aware Scheduling
- Previous Research
- Proposed Method
- Evaluation
- Summary and Conclusion

Intelligent Transportation Systems (ITS)

- Controls traffic through information exchange between
  - Road Side Units (RSUs) and
  - Connected Vehicles (CVs)

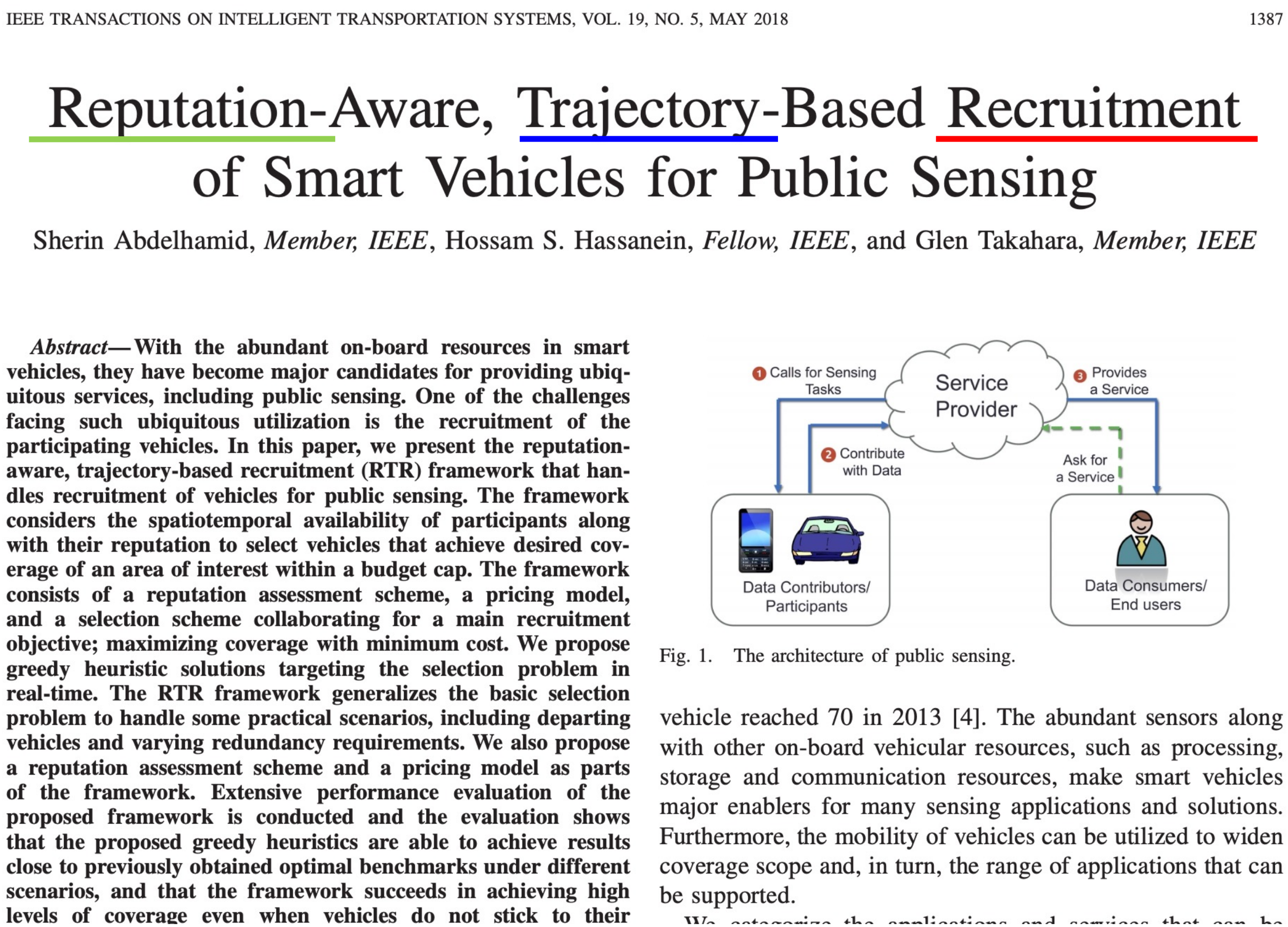


- Typical real-time decisions made by an ITS include:
  - Adaptive traffic-light timing
  - Emergency vehicle priority
  - Eco-driving guidance
  - Safe headway and speed advice

- This system must satisfy real-time constraints while considering that not all CVs can be included.

Why and How ITS Recruits Only a Subset of Vehicles

- Why not recruit *all* vehicles?
  - Budget limits
  - Data quality and trust
  - Redundancy
  - Real-time constraints
- Typical recruitment criteria
  - Availability
    - Trajectories
  - Reputation
    - Past behavior, accuracy



IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, VOL. 19, NO. 5, MAY 2018 1387

Reputation-Aware, Trajectory-Based Recruitment of Smart Vehicles for Public Sensing

Sherin Abdelhamid, *Member, IEEE*, Hossam S. Hassanein, *Fellow, IEEE*, and Glen Takahara, *Member, IEEE*

***Abstract*— With the abundant on-board resources in smart vehicles, they have become major candidates for providing ubiquitous services, including public sensing. One of the challenges facing such ubiquitous utilization is the recruitment of the participating vehicles. In this paper, we present the reputation-aware, trajectory-based recruitment (RTR) framework that handles recruitment of vehicles for public sensing. The framework considers the spatiotemporal availability of participants along with their reputation to select vehicles that achieve desired coverage of an area of interest within a budget cap. The framework consists of a reputation assessment scheme, a pricing model, and a selection scheme collaborating for a main recruitment objective; maximizing coverage with minimum cost. We propose greedy heuristic solutions targeting the selection problem in real-time. The RTR framework generalizes the basic selection problem to handle some practical scenarios, including departing vehicles and varying redundancy requirements. We also propose a reputation assessment scheme and a pricing model as parts of the framework. Extensive performance evaluation of the proposed framework is conducted and the evaluation shows that the proposed greedy heuristics are able to achieve results close to previously obtained optimal benchmarks under different scenarios, and that the framework succeeds in achieving high levels of coverage even when vehicles do not stick to their**

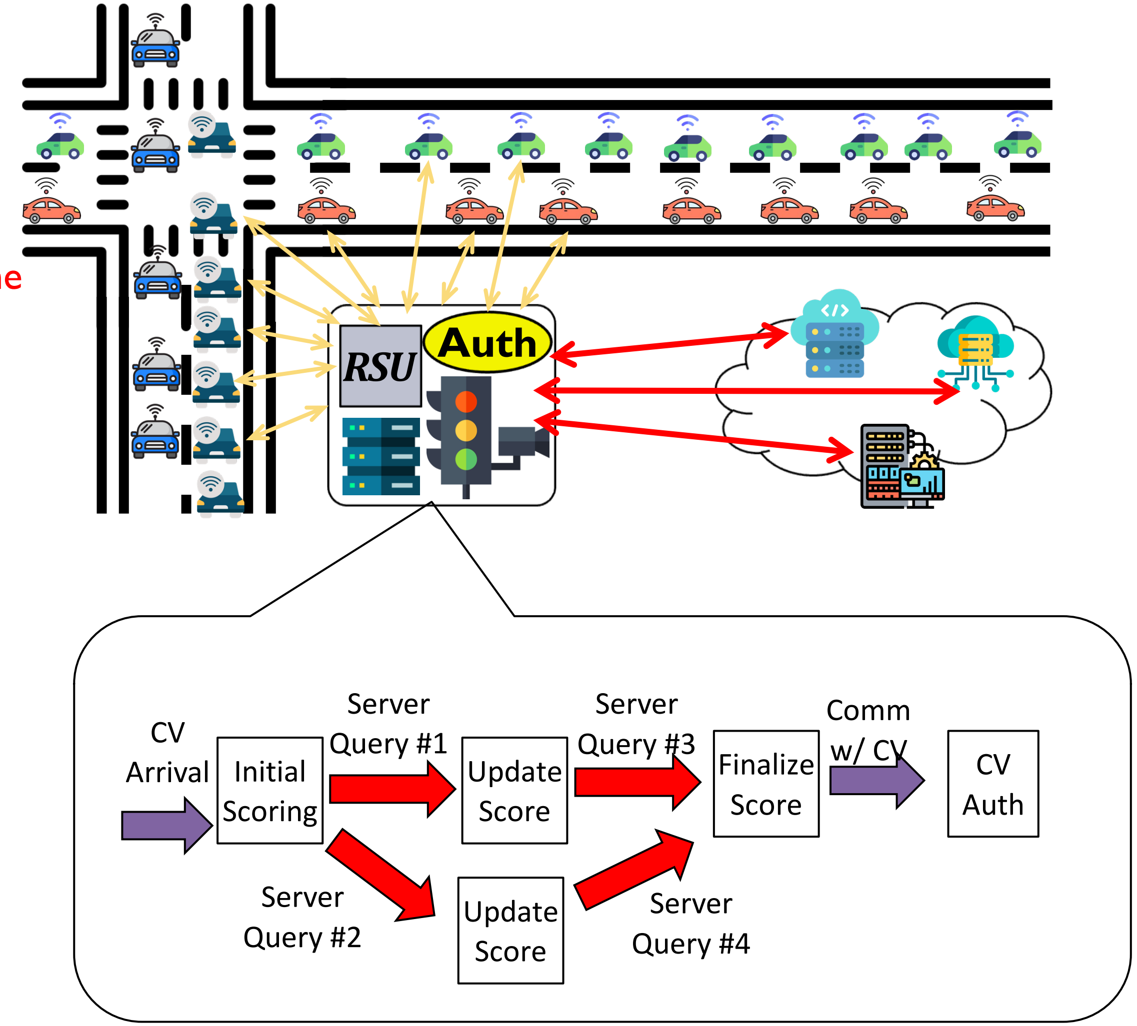
1 Calls for Sensing Tasks
Service Provider
3 Provides a Service
2 Contribute with Data
Ask for a Service
Data Contributors/ Participants
Data Consumers/ End users

Fig. 1. The architecture of public sensing.

vehicle reached 70 in 2013 [4]. The abundant sensors along with other on-board vehicular resources, such as processing, storage and communication resources, make smart vehicles major enablers for many sensing applications and solutions. Furthermore, the mobility of vehicles can be utilized to widen coverage scope and, in turn, the range of applications that can be supported.

Recruitment via *Value-Aware Scheduling*

- Recruitment is done by authentication
  - which takes places in steps and require lots of DB lookup
  - **Task Graph**
- We propose to improve the quality of decisions by
  - prioritizing high-reputation vehicles → scheduled within the deadline
  - Aggregated high scores raises overall decision quality.
- Not traditional hard real-time scheduling
  - Not all CVs are scheduled within the deadline
  - No contribution if deadline is missed
- Not traditional soft real-time scheduling
  - Soft real-time tolerates occasional deadline misses
  - This work
    - only tasks that meet the deadline are recruited
    - those with missed deadline are excluded
- The score value is only known at run time



Value-Aware Scheduling in Real-Time Bidding (BTB)

- Advertisers send a continuous stream of real-time bid requests
  - Just like CVs in ITS
- Serving every request is not an economically wise decision
  - Running the bidding algorithm is not free
    - GoogleCloud and AWS charge a lot!
    - Profit grows by focusing resources on high-revenue bid opportunities and safely ignoring low-value ones

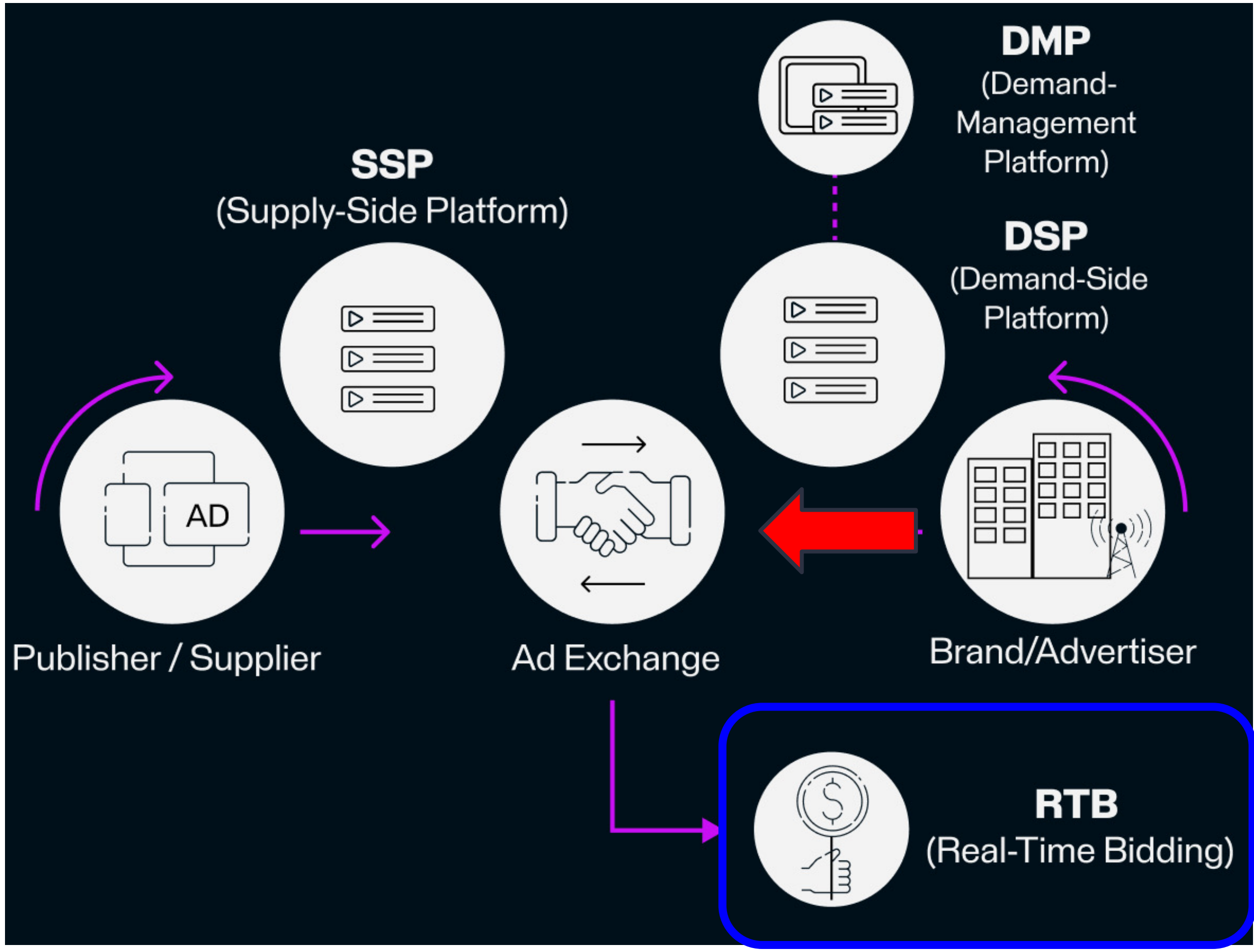


Image credit: Perion

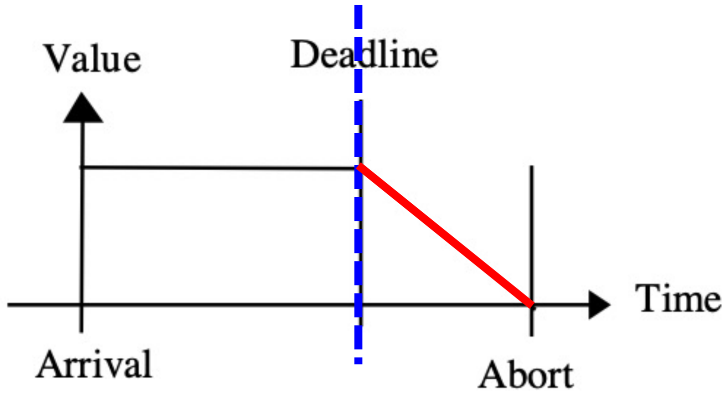
Outline

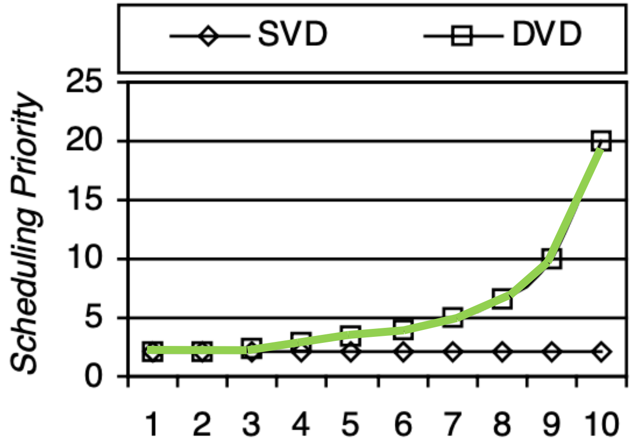
- Background and Motivation
- Previous Research
  - Dynamic Value Density
  - Value-Maximizing in Animation Rendering
  - Mixed-Criticality Scheduling
- Proposed Method
- Evaluation
- Summary and Conclusion

Dynamic Value Density (DVD)

Saud Ahmed Aldarmi and Alan Burns. "*Dynamic value-density for scheduling real-time systems.*" Proceedings of 11th Euromicro Conference on Real-Time Systems. Euromicro RTS'99. IEEE, 1999.

- Soft Real-Time
  - Value decays gradually after its deadline
- Priority := (Dynamic Value Density) = **(Value)/(Remaining Time)**
  - As opposed to (static density) = (Value) / (WCET)
  - Higher value → Higher priority
  - As remaining time shrinks → Priority rises
- Key effects
  - Imminent deadlines boost priority, avoiding wasted CPU time on tasks that would otherwise miss and yield no value

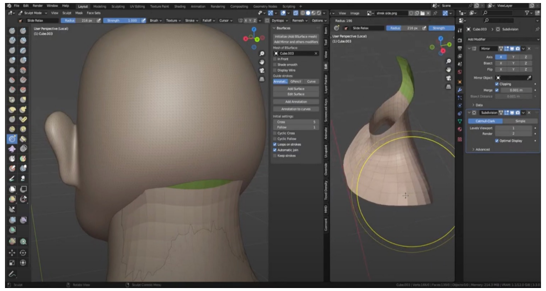


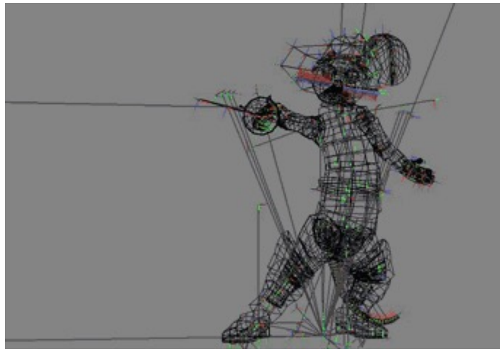


Animation Rendering-Cluster Scheduling Problem

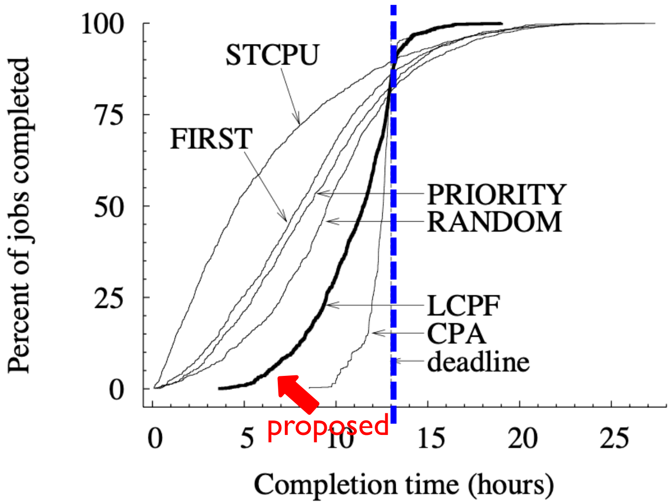
Anderson, Eric, et al. "*Value-maximizing deadline scheduling and its application to animation rendering*." Proceedings of the seventeenth annual ACM symposium on Parallelism in algorithms and architectures. 2005.

- Value-Maximizing Scheduling (by HP)
  - Each "rendering job" is represented as a task graph
  - Common overnight deadline (a few hours)
    - Artists submit their jobs when leaving work and review the results the next morning
  - Tried to maximize total value earned over night
  - Used for Dreamworks' animation rendering



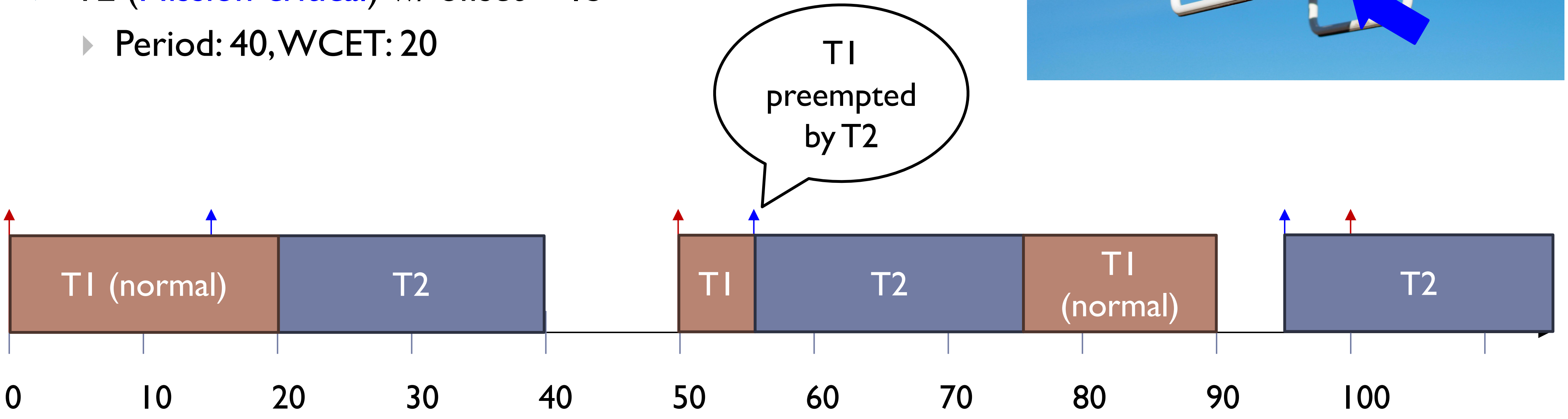


- They proved that this is NP-hard and proposed a two-phase method
  - near-optimal when each job's critical path is well below the global deadline.
- Experiments (DreamWorks Shrek 2 cluster)
  - On real 8-week traces from a 1,000-CPU cluster
  - w/ Shrek 2 rendering data



Mixed-Criticality Scheduling (Normal)

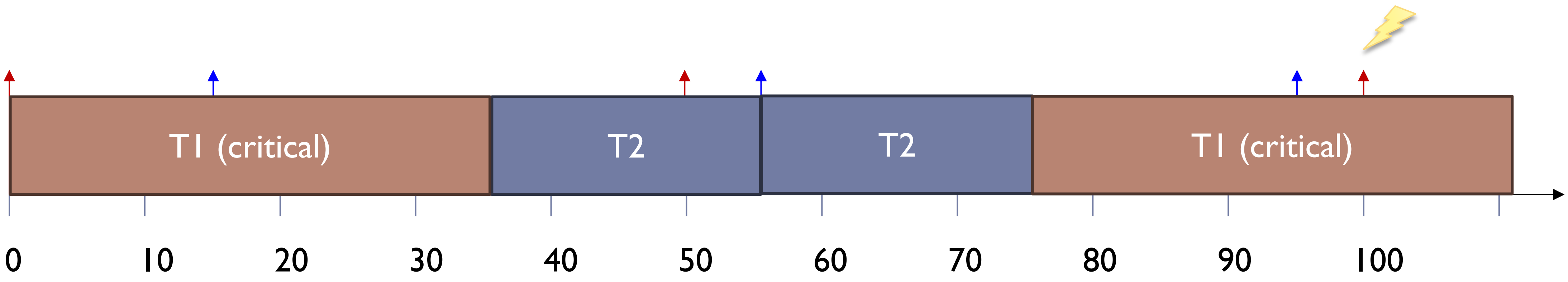
- Task Sets with dynamic importances
  - T1 (Safety-critical) w/ offset = 0
    - [Normal] Period: 50, WCET: 20
    - [Critical] Period: 50, WCET: 35
  - T2 (Mission-critical) w/ offset = 15
    - Period: 40, WCET: 20



Mixed-Criticality Scheduling (Critical)

- Task Sets with dynamic importances
  - T1 (Safety-critical) w/ offset = 0
    - [Normal Period: 50, WCET: 20
    - [Critical] Period: 50, WCET: **35**
  - T2 (Mission-critical) w/ offset = 15
    - Period: 40, WCET: 20

Deadline violation on the safe-critical task!

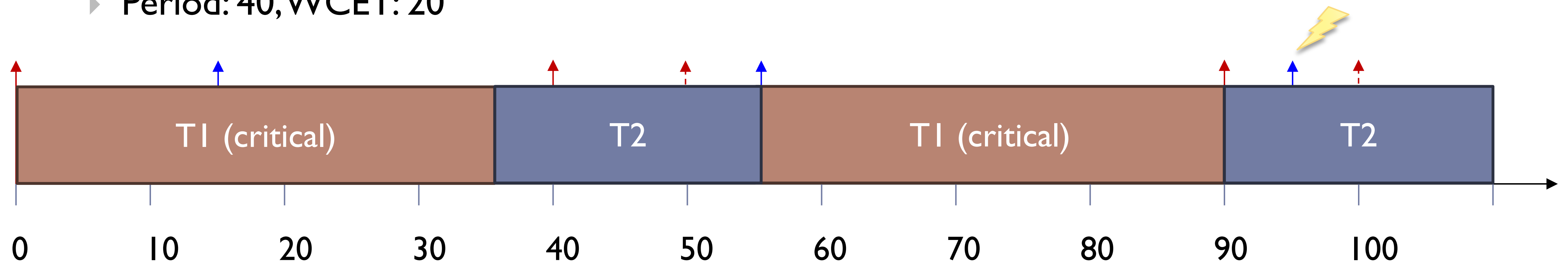


Mixed-Criticality Scheduling with EDF-VD (Critical)

Baruah, Sanjoy, et al. "*The preemptive uniprocessor scheduling of mixed-criticality implicit-deadline sporadic task systems.*" 2012 24th Euromicro Conference on Real-Time Systems. IEEE, 2012.

- Task Sets with dynamic importances
  - T1 (Safety-critical) w/ offset = 0
    - Normal: Period: 50, WCET: 20
    - Critical: Period: 50, WCET: **35** virtual deadline (50→40)
  - T2 (Mission-critical) w/ offset = 15
    - Period: 40, WCET: 20

Deadline violation on the mission-critical task!



Comparison of Existing Value-Aware Scheduling

	Task Model	Soft vs. Hard	Pre-emptive?	Importance?	Scheduling Policy
EDF	Independent Task Set	Hard	Pre-emptive	N/A	Time-based
DVD	Independent Task Set	Soft	Pre-emptive	User-specified	Value-based
Animation Rendering	Task Graph	N/A	Non-preemptive	User-specified or determined by Algorithm	Value-based
EDF-VD (Mixed-Criticality)	Independent Task Set	Mixed	Pre-emptive	User-specified	Time-based

Outline

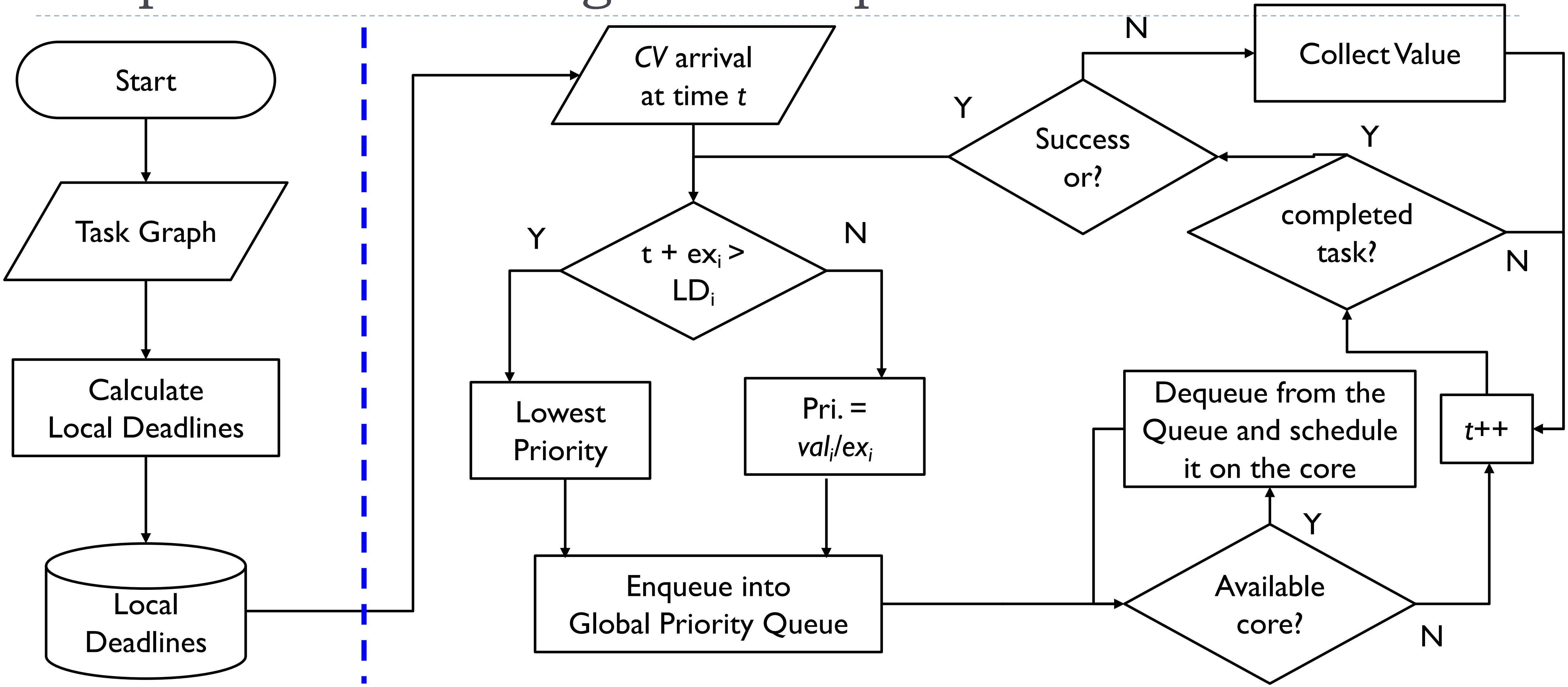
- Background and Motivation
- Previous Research
- Proposed Method
  - Task/Scheduling Model
  - Proposed Scheduling Policy
- Evaluation
- Summary and Conclusion

Task/Scheduling Model

- Recruitment (or Authentication) is modeled as a **task graph *G*** := <*N*, *E*>
  - A set of tasks (nodes) ***N*** and a set of data dependencies (edges) ***E***
    - Each $n \in N$ is associated with WCET: $\mathbf{ex_n}$ and Value: $\boldsymbol{val_n}$
    - Each $e=\langle n_s,n_d\rangle \in E$ is associated with communication delay $comm_e$ (applies after the completion of n_s)
  - Task graphs arrive in a sporadic manner
    - With a mean arrival rate of **1/λ**

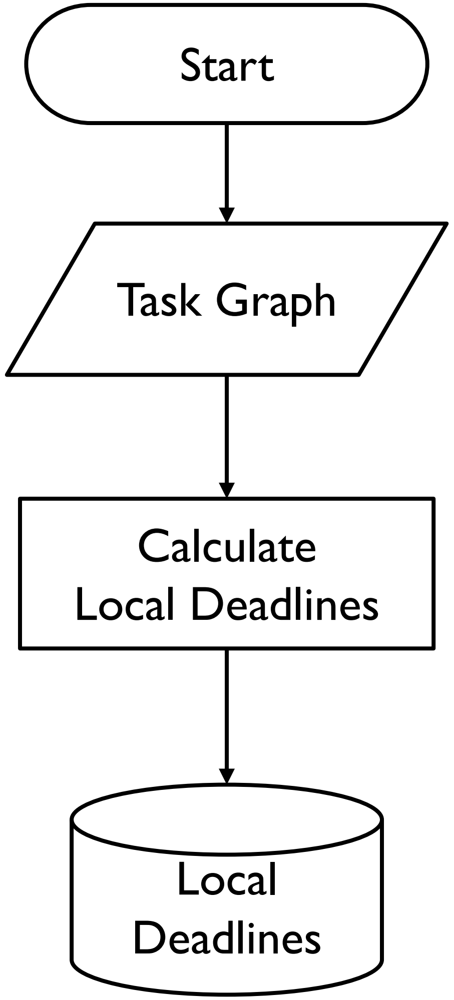
- Scheduling
  - ***Global* scheduling** policy for multi-core (as opposed to *portioned* scheduling)
  - Each node is scheduled in a **non-preemptive** manner

Proposed Scheduling in a Simplified Flowchart

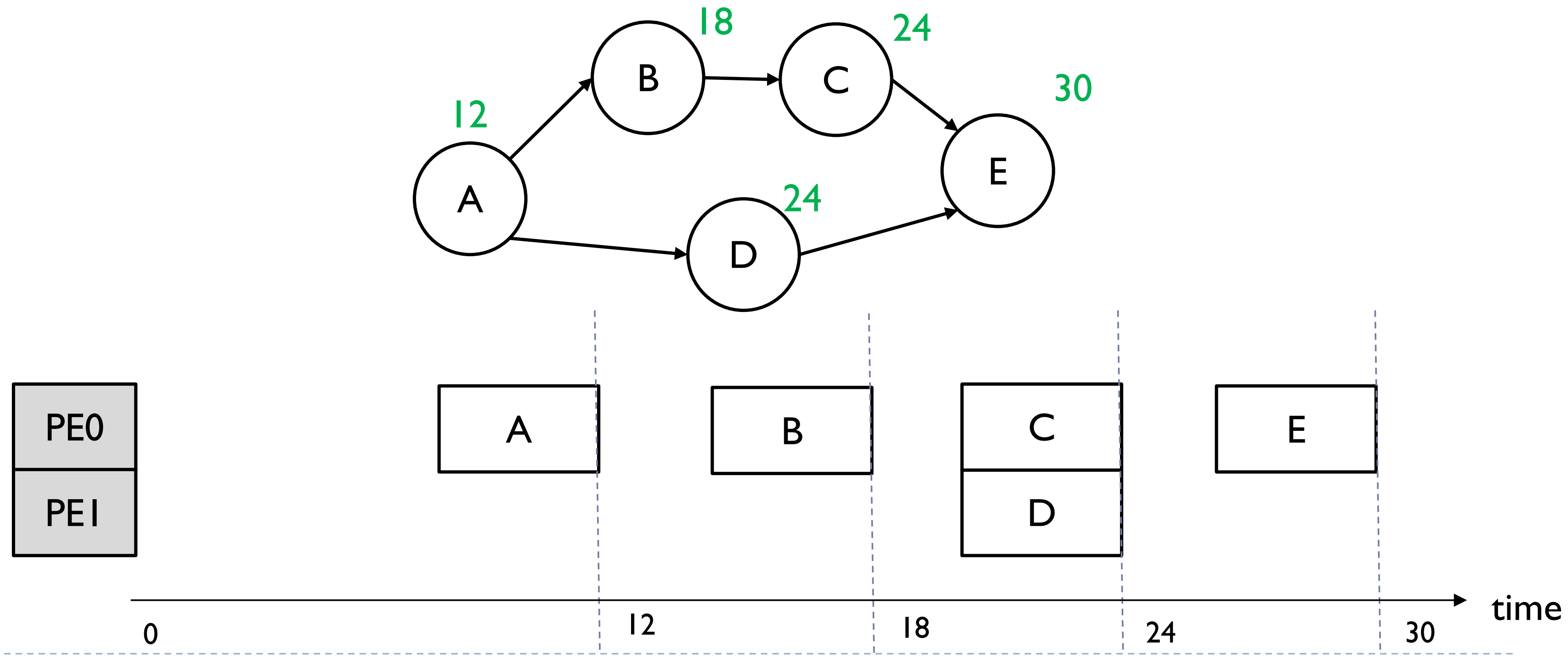


[**offline**: calculate local deadline] [**online**: value-aware priority assignment] [**online**: task graph scheduling]

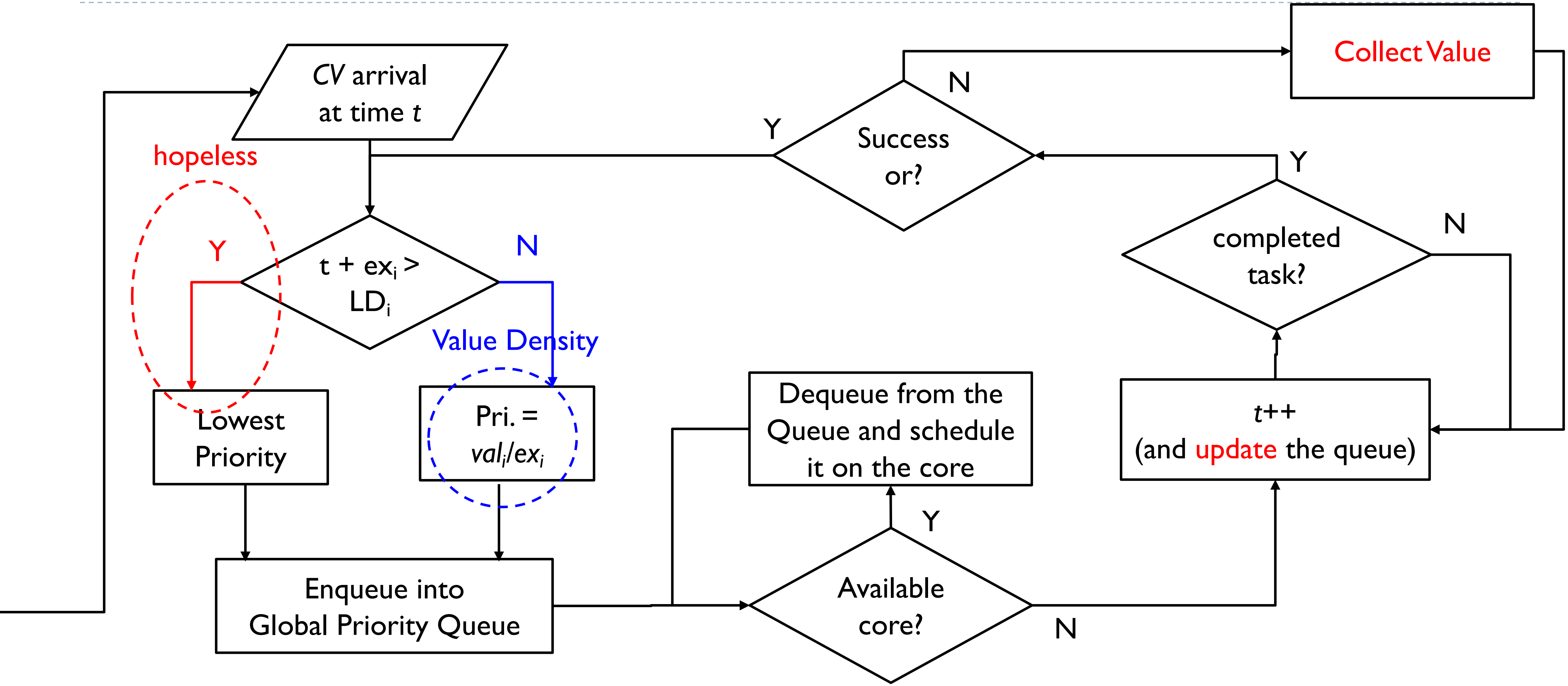
Offline: Calculating Local Deadlines



- First calculate the local deadline (LD) of each task node
  - Based on As-Late-As-Possible (ALAP) List Scheduling
  - E.g., Latency constraint: 30, ex = 4, comm=2



Online Time- and Value-Aware Task Graph Scheduling



Comparison with Existing Approaches

	Task Model	Soft vs. Hard	Pre-emptive?	Importance?	Scheduling Policy
EDF	Independent Task Set	Hard	Pre-emptive	N/A	Time-based
DVD	Independent Task Set	Soft	Pre-emptive	User-specified	Value-based
Animation Rendering	Task Graph	N/A	Non-preemptive	User-specified or Determined by Algorithm	Value-based
EDF-VD (Mixed-Criticality)	Independent Task Set	Mixed	Pre-emptive	User-specified	Time-based
Proposed	Task Graph	Mixed	Non-preemptive	Determined by Algorithm	Time/Value-based

Outline

- Background and Motivation
- Previous Research
- Proposed Method
- Evaluation
  - Simulator and Configurations
  - Results
- Summary and Conclusion

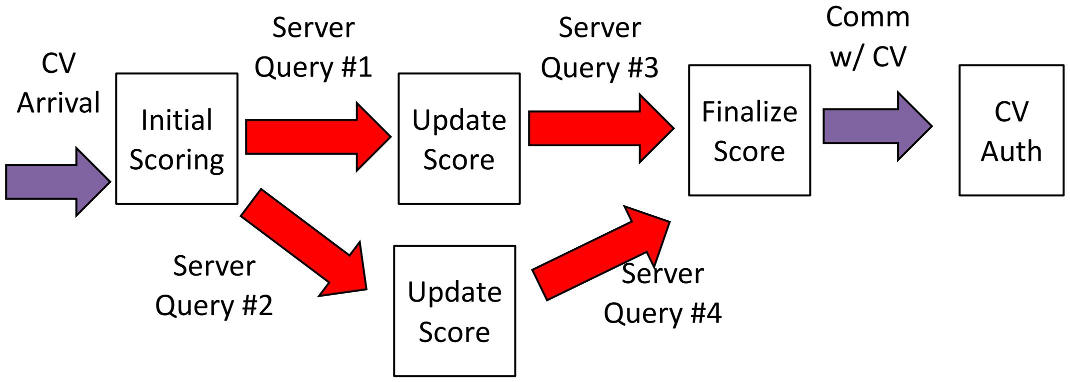
Simulator and Configurations

- In-house ITS simulator with the proposed scheduling policy is built on top of an open-source discrete event simulator (SimPy)

- Score-based Authentication Task Graph
  - With 5 task nodes
  - WCET of each node = 100ms
  - Communication delay = 100ms
  - Authentication deadline: 1s

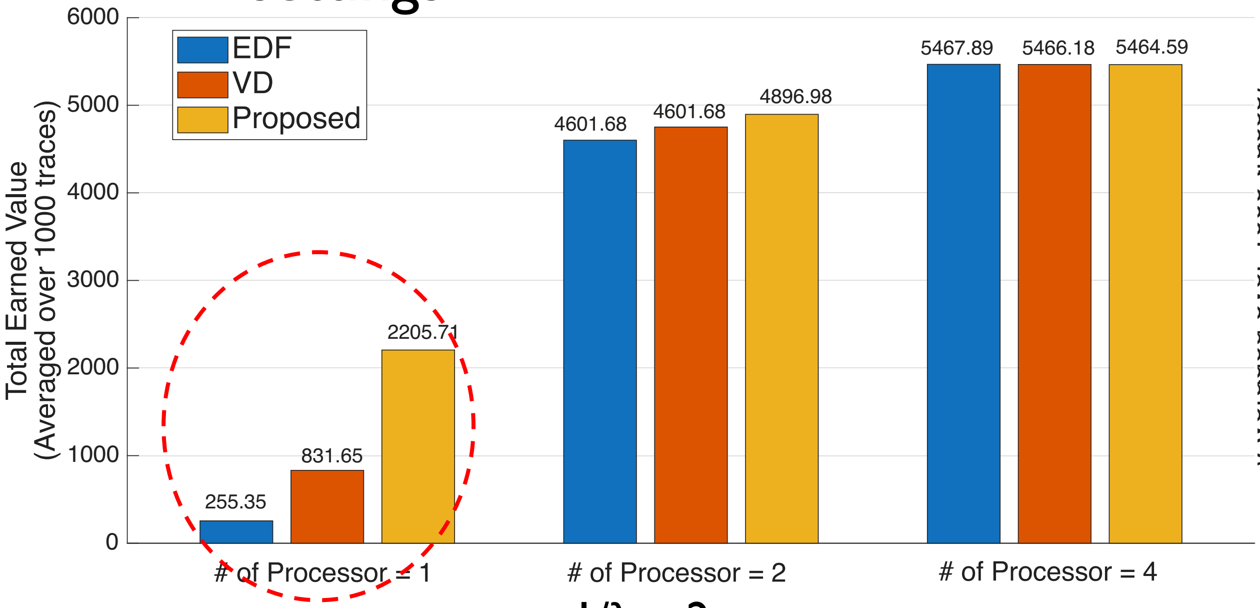
- CV arrival rate 1/λ varied from 1 to 3
- Values: randomly chosen between 1 and 10

- Global Non-Preemptive Scheduling on
  - Homogeneous multi-processor w/ 1, 2, and 4 cores



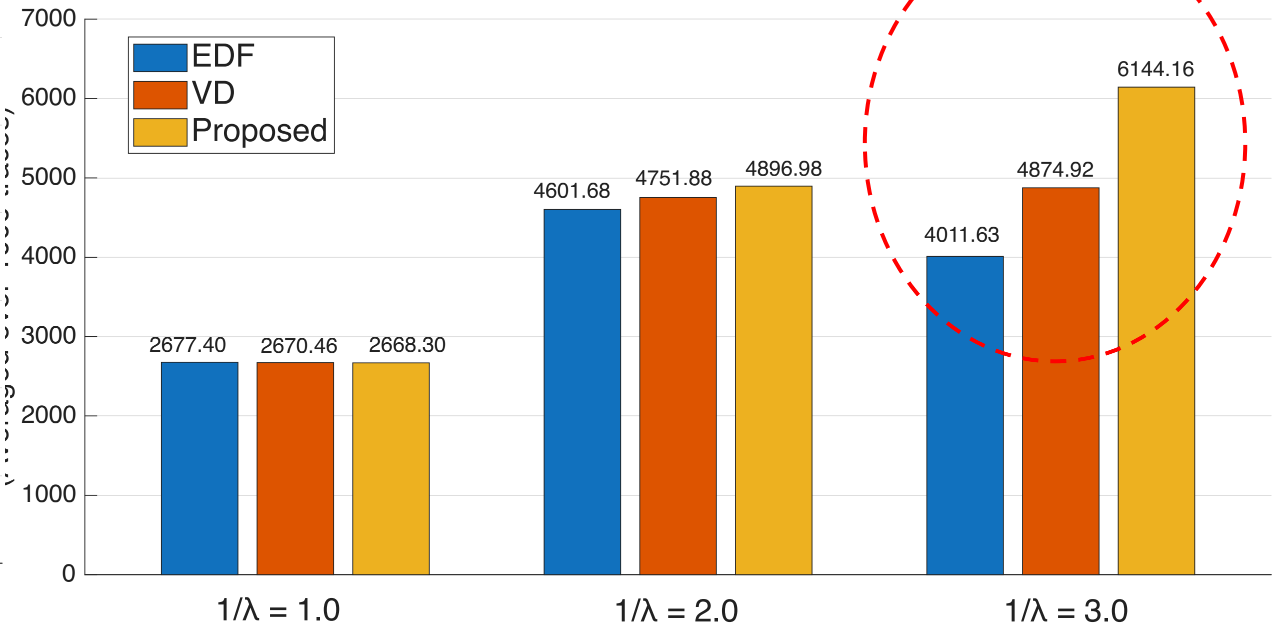
Experimental Results

- Experiments performed for 1000 randomly generated CV arrival traces (each being for 100s)
  - The proposed method proven to be effective in resource-constrained or busy settings



1/λ = 2

Smaller resource ⟷ Bigger resource



2 processors

Lower arrival rate ⟷ Higher arrival rate

Conclusion and Summary

- We proposed a **value-aware real-time scheduling technique for intelligent transportation systems**, where selective recruitment of vehicles is essential.
- In the proposed approach, both **value (e.g., reputation) and timing urgency are jointly considered** in scheduling decisions, thereby maximizing the aggregated value of the connected vehicles that meet their deadlines.
- Extensive discrete-event simulations demonstrated that the proposed technique is particularly **effective under high system load or limited resource conditions**.

Thank You!

(hoeseok.yang@scu.edu)

Traditional Real-Time Scheduling View Points

Hard Real-Time	Soft Real-Time
All tasks *must* meet deadlines	Occasional deadline misses *tolerated*

👉 Not all tasks meet their deadlines, but every task admitted (i.e., recruited) is required to meet its deadline.

User-Specified Importance	Time-Based Importance
e.g., *Fixed*-Priority (FP)	e.g., *Rate*-Monotonic (RM), Earliest-*Deadline* First (EDF), …

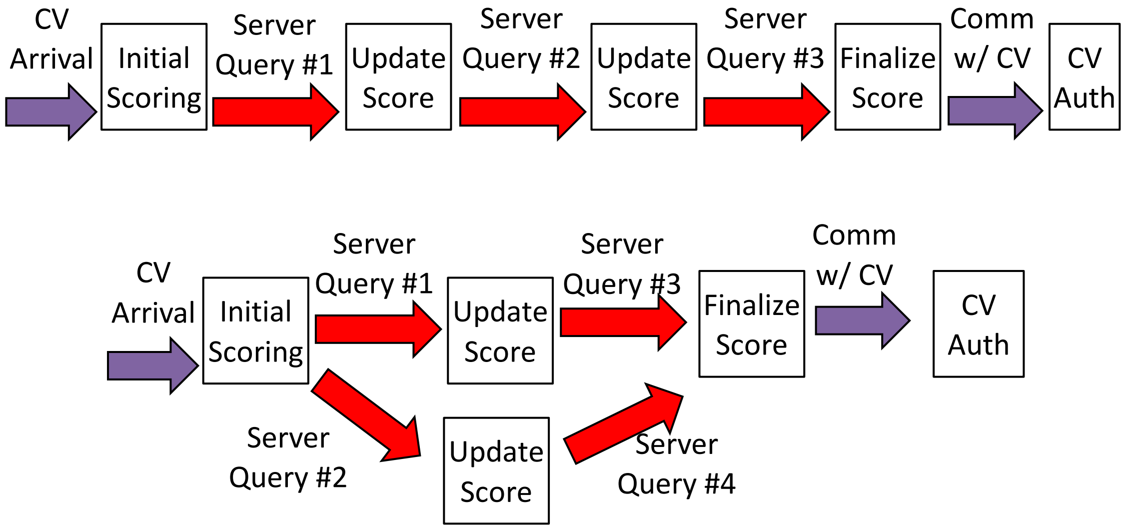
👉 Importance is determined by an algorithm, e.g., based on reputation or coverage

Static Priority	Dynamic Priority
e.g., FP, RM, …	e.g., EDF

👉 The "importance" is only known at run time.

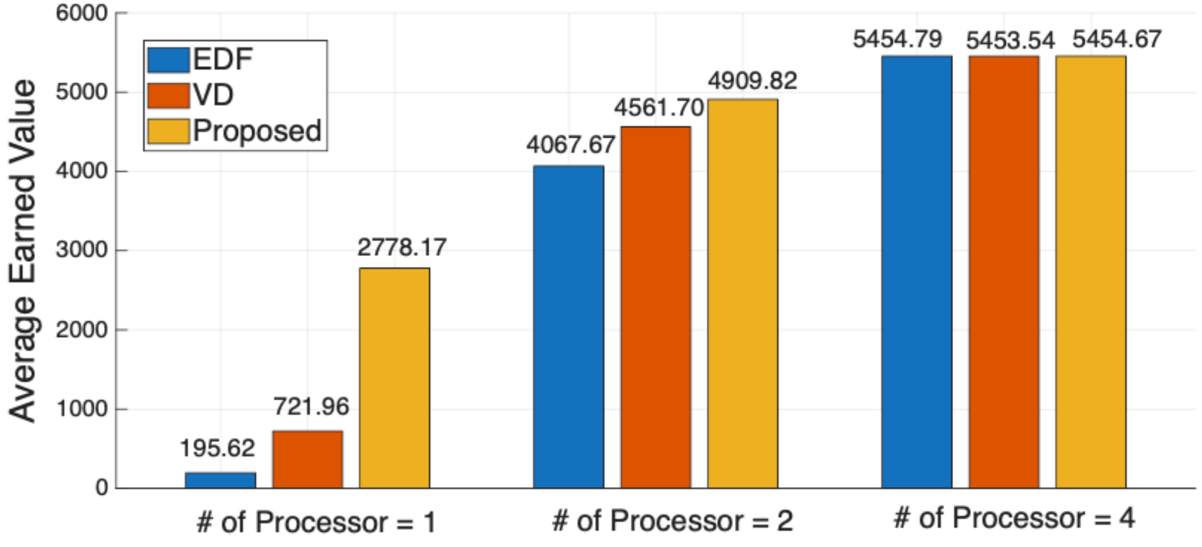
Simulator and Configurations

- In-house ITS simulator with the proposed scheduling policy is built on top of an open-source discrete event simulator (SimPy)
- Two different Score-based Authentication
  - Serial and Parallel
  - WCET of each node = 100ms
  - Communication delay = 100ms
- CV arrival rate 1/λ varied from 1s to 3s
- Global Non-Preemptive Scheduling on
  - Homogeneous multi-processor w/ 1, 2, and 4 cores

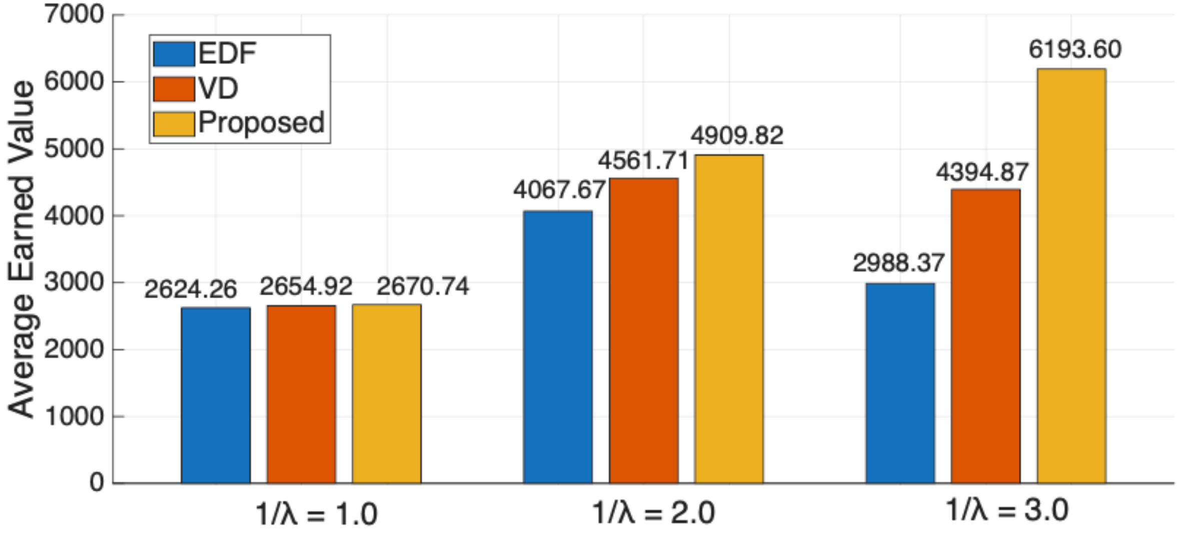


Experimental Results - Serial

- Experiments performed for 1000 randomly generated CV arrival traces
  - The proposed method proven to be effective in resource-constrained or busy settings



1/λ = 2



2 processors

Smaller resource ⟺ Bigger resource

Lower arrival rate ⟺ Higher arrival rate