Performance Evaluation of Scheduling Algorithms for Database Services with Soft and Hard SLAs

DataCloud-SC11
November 14, 2011

Hyun Jin Moon, *Yun Chi, Hakan Hacigumus*

NEC Laboratories America

Cupertino, USA

Empowered by Innovation NEC

Context: Who we are

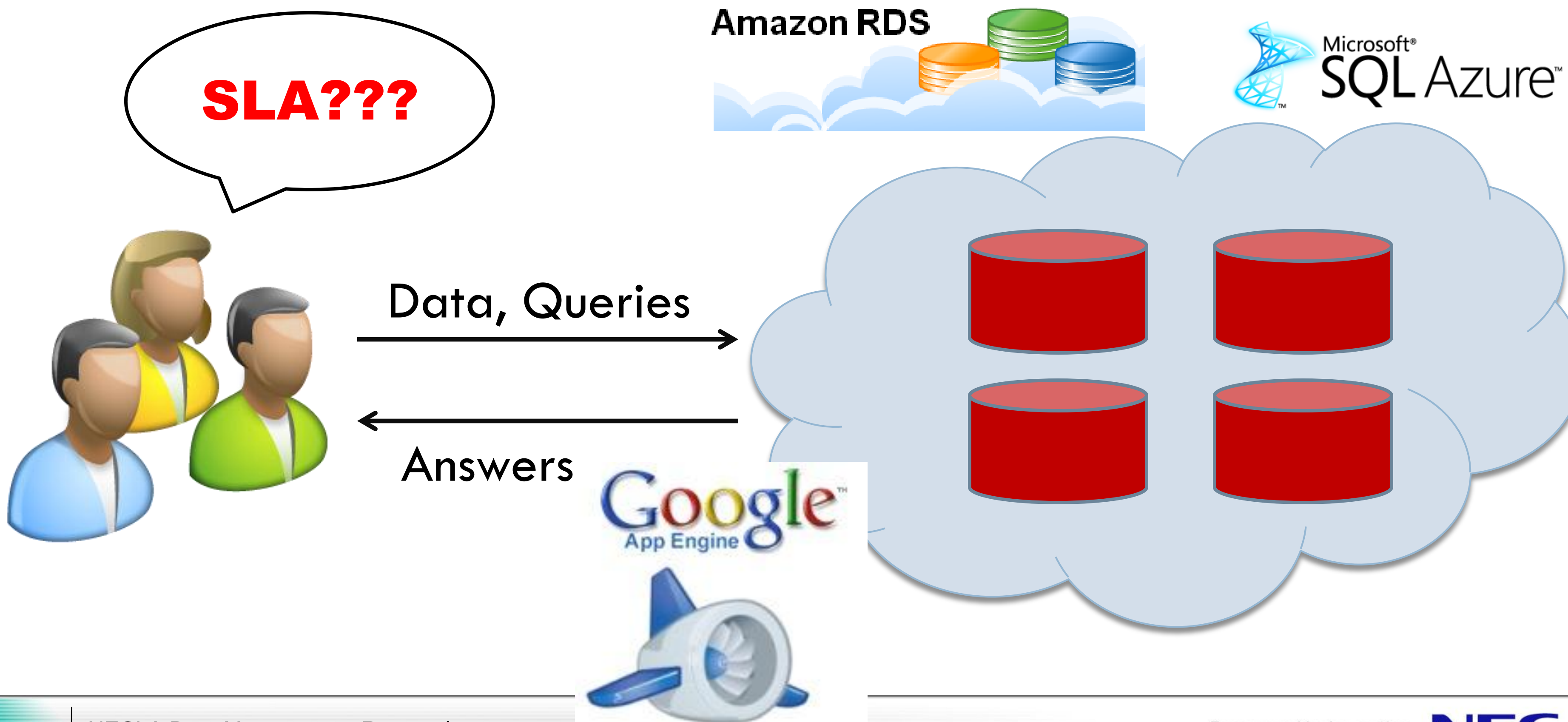
- NEC Labs Data Management Research Group
  - Focus: to build CloudDB platform
    - Microsharding: SQL on elastic Key-Value stores (e.g., HBase)
    - Maestro: resource and workload management
    - COSMOS: seamless mobility by CloudDB
  - Research overview in SIGMOD Record 2011 Sep issue

Context: Who we are

- NEC Labs Data Management Research Group
  - Focus: to build CloudDB platform
    - Microsharding: SQL on elastic Key-Value stores (e.g., HBase)
    - **Maestro: resource and workload management**
    - COSMOS: seamless mobility by CloudDB
  - Research overview in SIGMOD Record 2011 Sep issue

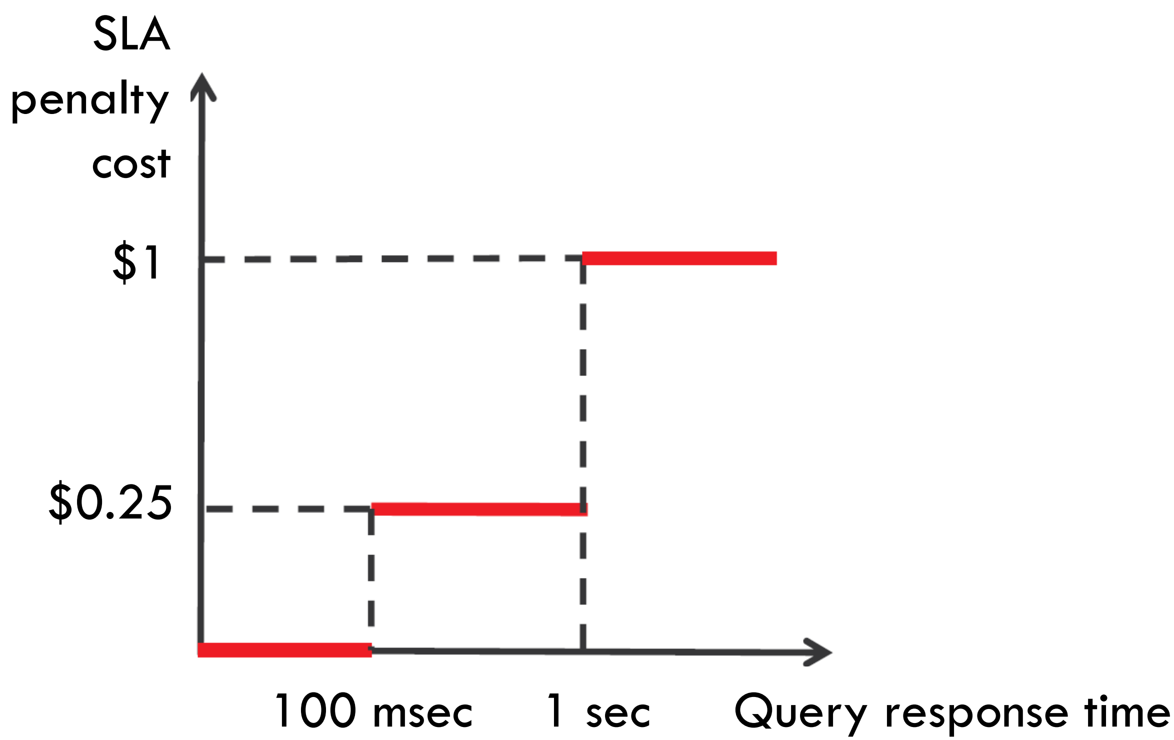
Data Service Offering

- Data management is not easy
  - Data migration, replication, consistency, elasticity, etc.
- Data service to the rescue!



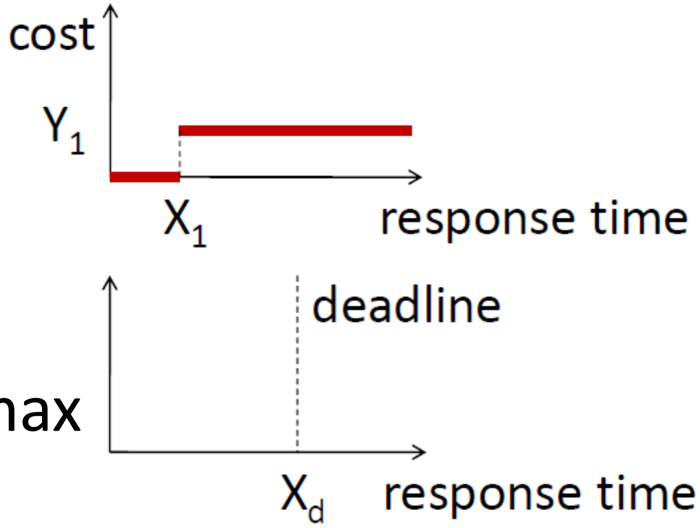
SLA Cost

- A general function on response time and SLA penalty

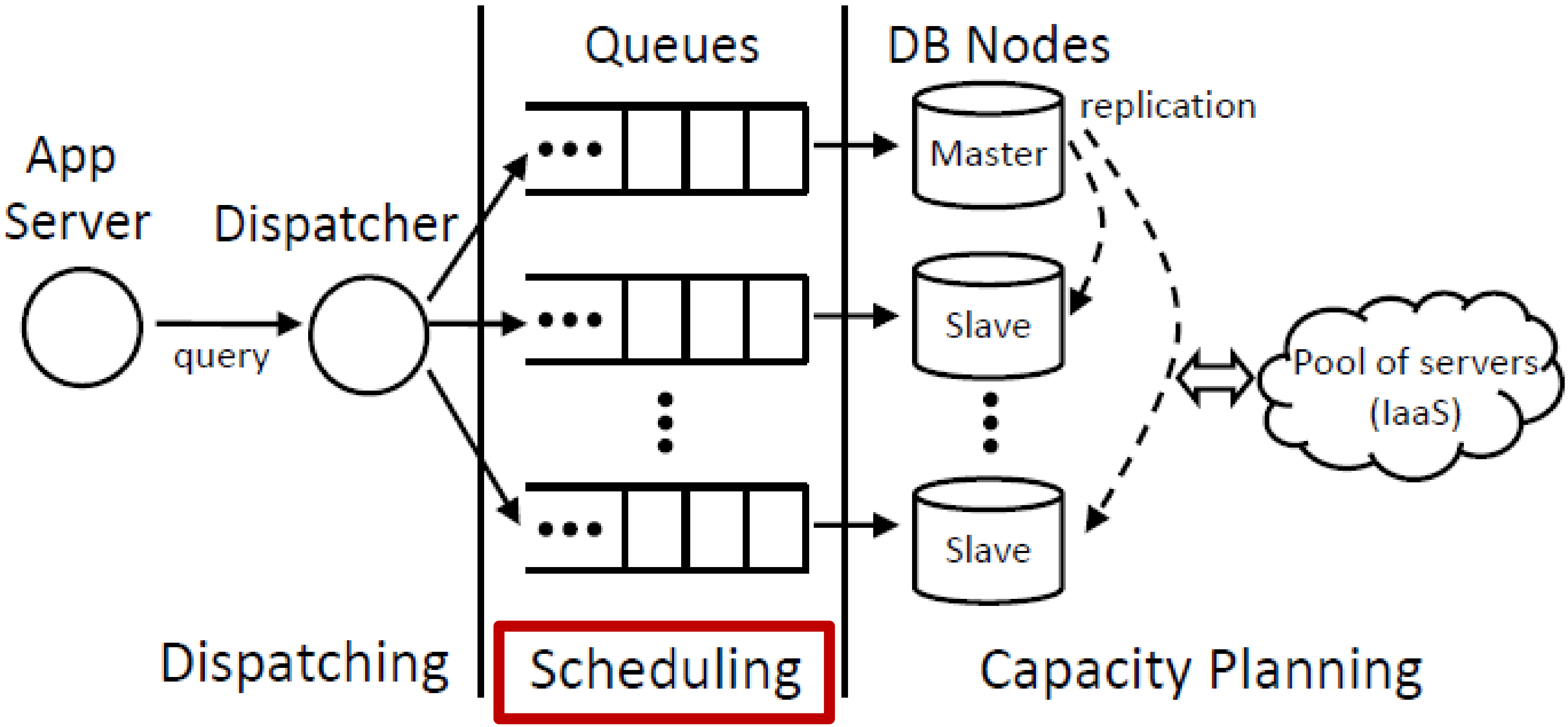


Soft and Hard SLAs

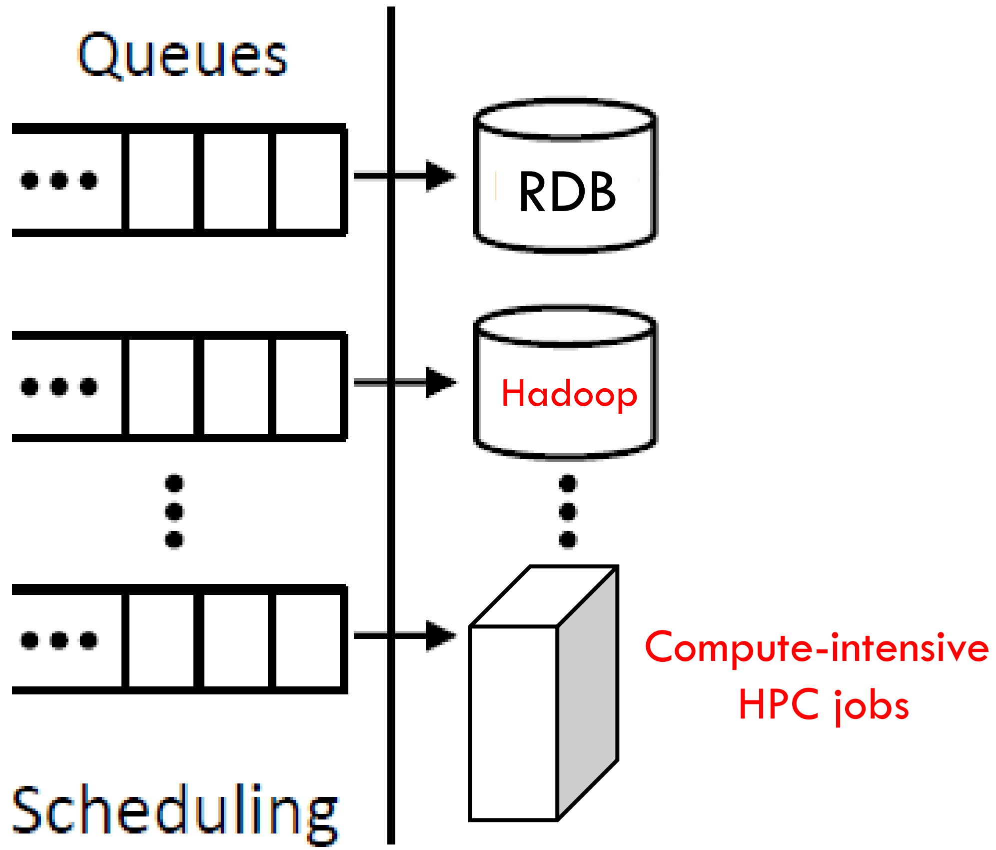
- *Soft* SLA
  - The SLA from the previous slide
- *Hard* SLA (i.e., Deadline)
  - A response time deadline and the max violation percentage
- Both SLAs in action!
  - Soft SLA: customer-facing performance-penalty agreement for all jobs
  - Hard SLA: performance goal set within the service provider for all/subset of jobs



Our Reference Architecture



Applicability of our work



Taxonomy of Scheduling Algorithms

	Cost-unaware	Cost-aware
Deadline-aware		
Deadline-unaware	FCFS SJF (Shortest Job First)	

Taxonomy of Scheduling Algorithms

	Cost-unaware	Cost-aware
Deadline-aware	EDF (Earliest Deadline First) AED [Haritsa91]	
Deadline-unaware	FCFS SJF (Shortest Job First)	

Taxonomy of Scheduling Algorithms

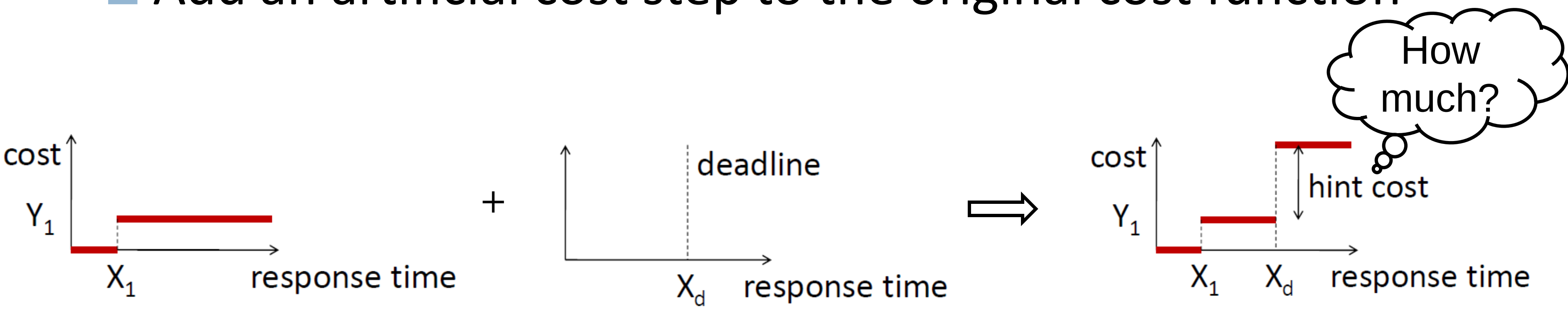
	Cost-unaware	Cost-aware
Deadline-aware	EDF (Earliest Deadline First) AED [Haritsa91]	
Deadline-unaware	FCFS SJF (Shortest Job First)	BEValue2 [Jensen85] FirstReward [Irwin04] iCBS [Chi11]

Taxonomy of Scheduling Algorithms

	Cost-unaware	Cost-aware
Deadline-aware	EDF (Earliest Deadline First) AED [Haritsa91]	**iCBS-DH**
Deadline-unaware	FCFS SJF (Shortest Job First)	BEValue2 [Jensen85] FirstReward [Irwin04] iCBS [Chi11]

iCBS-DH

- Cost- and deadline-aware scheduling
  - Option 1: invent a new scheduling algorithm
  - Option 2: leverage an existing algorithm
- Extend iCBS into **iCBS-DH**
  - Make it deadline-aware as well
  - Add an artificial cost step to the original cost function



(Soft SLA) Cost Function   (Hard SLA) Deadline   Cost Function with Deadline Hint

Experiment Setup (1/5)

- Server, database
  - Intel Xeon 2.4GHz, Two single-core CPUs, 16GB memory
  - MySQL 5.5, InnoDB 1.1.3, 1GB bufferpool
- Dataset, query
  - TPC-W 1GB scale data
  - 6 query templates chosen from the TPC-W workload
  - Open system workload, Poisson arrival, 85% load

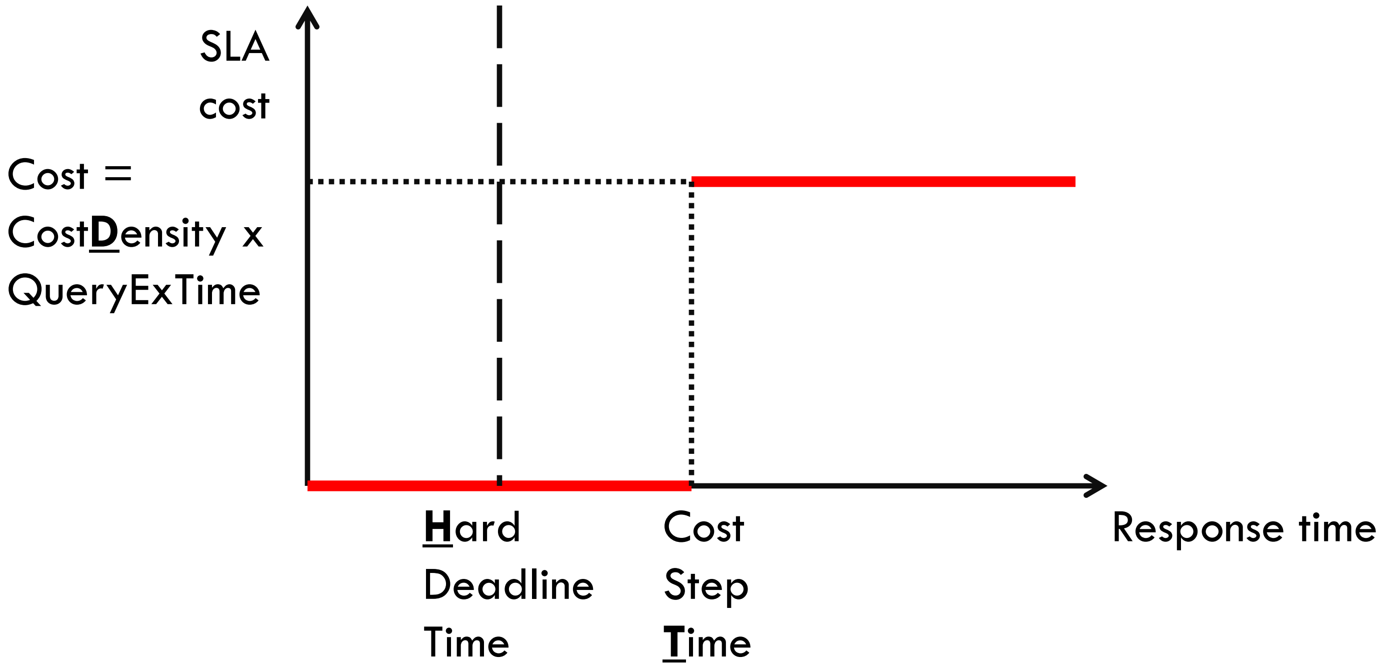
Experiment Setup (2/5)

- Runs
  - 5 seconds per run (>10K queries finished)
  - Each data point: the average of five repeated runs
- Execution time estimate
  - SJF, FirstReward, BEValue2, iCBS, iCBS-DH need it
  - Estimate from history: Mean+StandardDeviation

Experiment Setup (3/5)

NEC Laboratories America
Relentless passion for innovation

- SLA design for experiments
  - We need both soft and hard SLAs
  - Three parameters are used to create varying SLAs
    - DTH: CostDensity, CostStepTime, HardDeadineTime



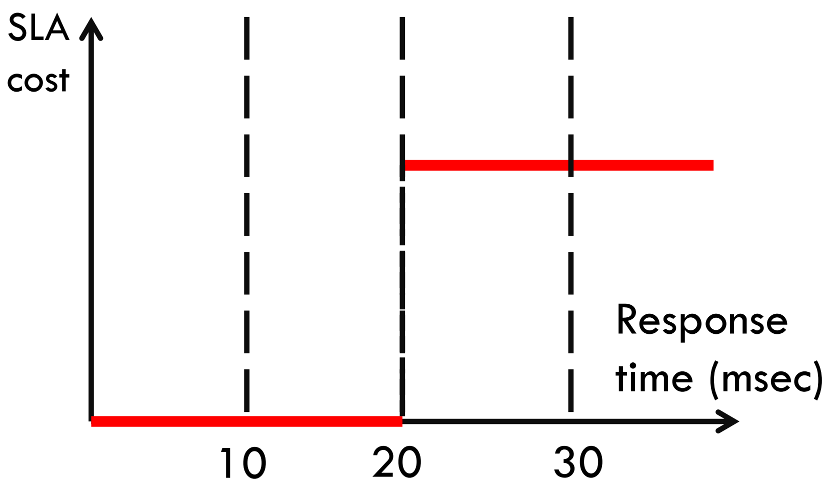
Experiment Setup (4/5)

- SLA design with DTH code
  - Cost**D**ensity - CostStep**T**ime – **H**ardDeadlineTime
  - E.g., DTH=113

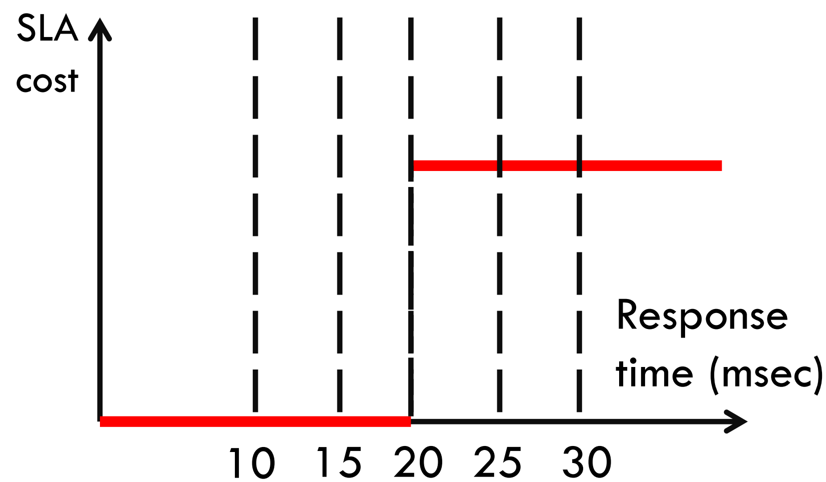
Query Type	ExTime (msec)	CostDensity			CostStepTime (msec)				HardDeadlineTime (msec)				
		1	2	3	1	2	3	4	1	2	3	4	5
Q1	0.23	3	1	5	20	10	30	20-40	10	20	30	10	30
Q2	0.23	3	2	4	20	15	25	20-40	10	20	30	15	25
Q3	0.30	3	3	3	20	20	20	20-40	10	20	30	20	20
Q4	0.41	3	4	2	20	25	15	20-40	10	20	30	25	15
Q5	0.54	3	5	1	20	30	10	20-40	10	20	30	30	10

Experiment Setup (5/5)

- Varying hard deadline in the following slides
  - Fixed CostDensity and CostStepTime (varied in the paper)
  - DTH: 11x (i.e., 111 through 115)

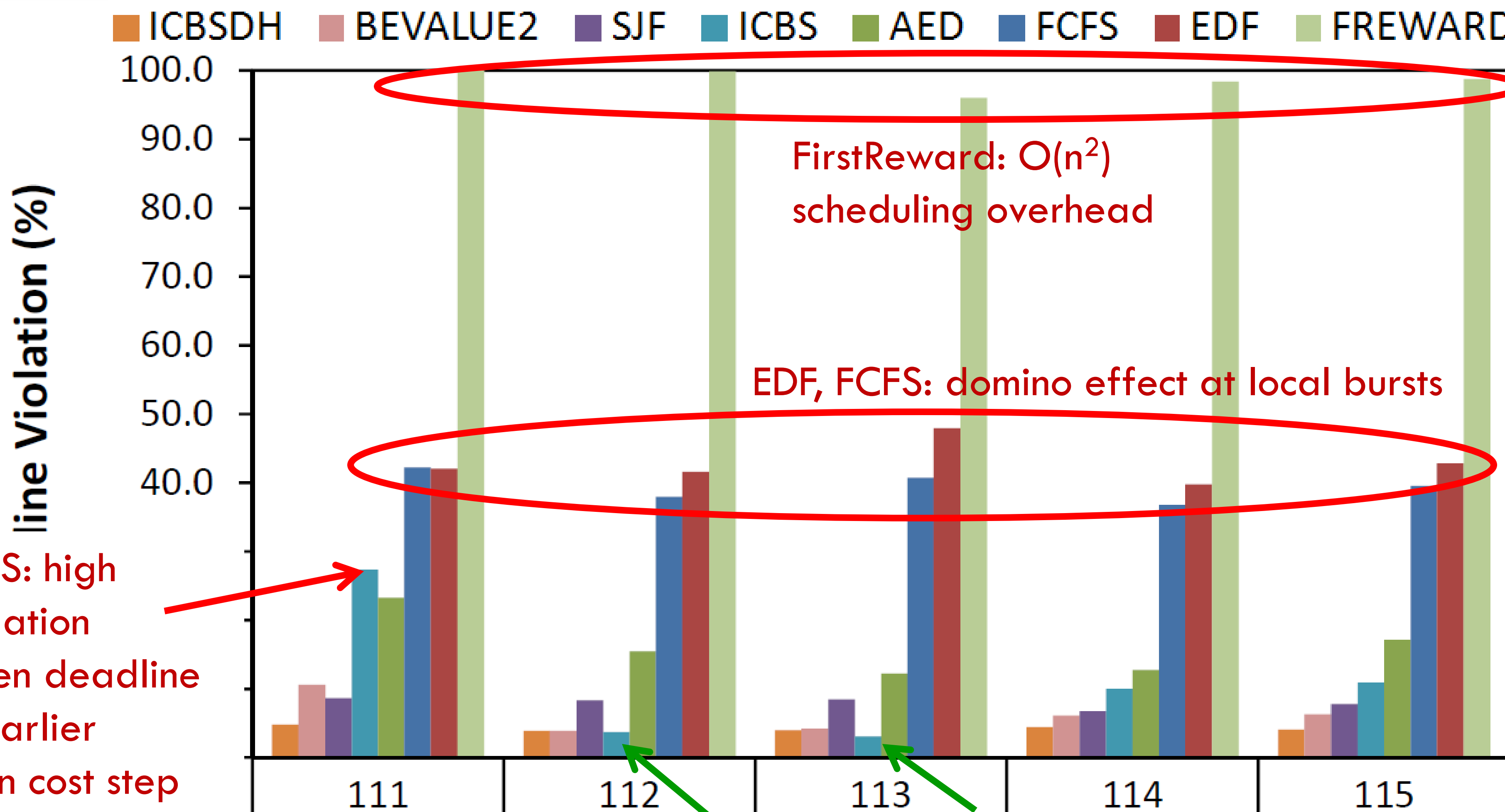


DTH=111 112 113
(all queries)



DTH=114: Q1 Q2 Q3 Q4 Q5
DTH=115: Q5 Q4 Q3 Q2 Q1
(Q1: shortest, Q5: longest)

Result 1: Varying Deadlines, Violation

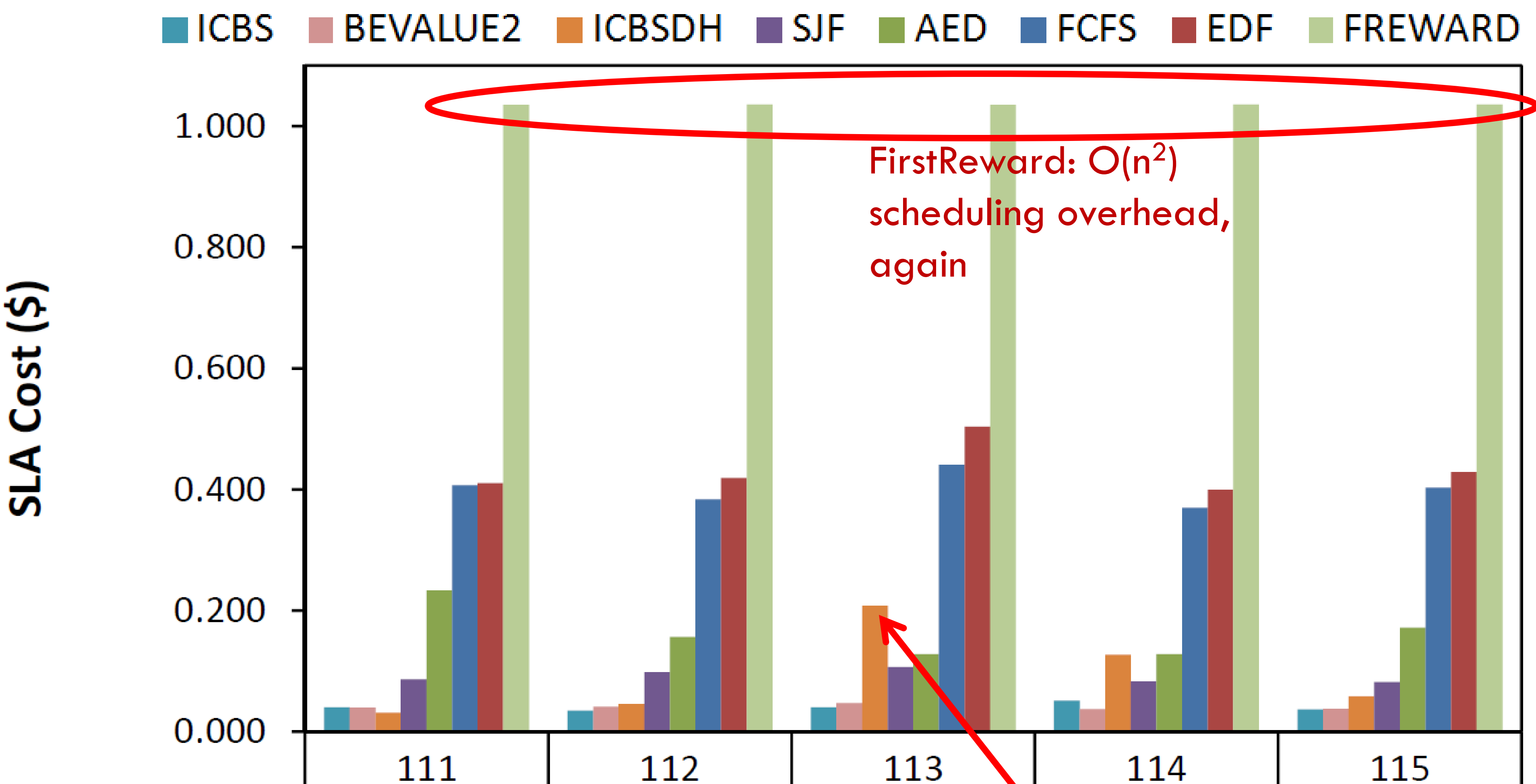


iCBS: high violation when deadline is earlier than cost step

iCBS-DH, BEValue2, SJF, iCBS perform well.
iCBS-DH performs the best, reliably.

iCBS low violation when deadline is same as or later than cost step

Result 1: Varying Deadlines, Cost

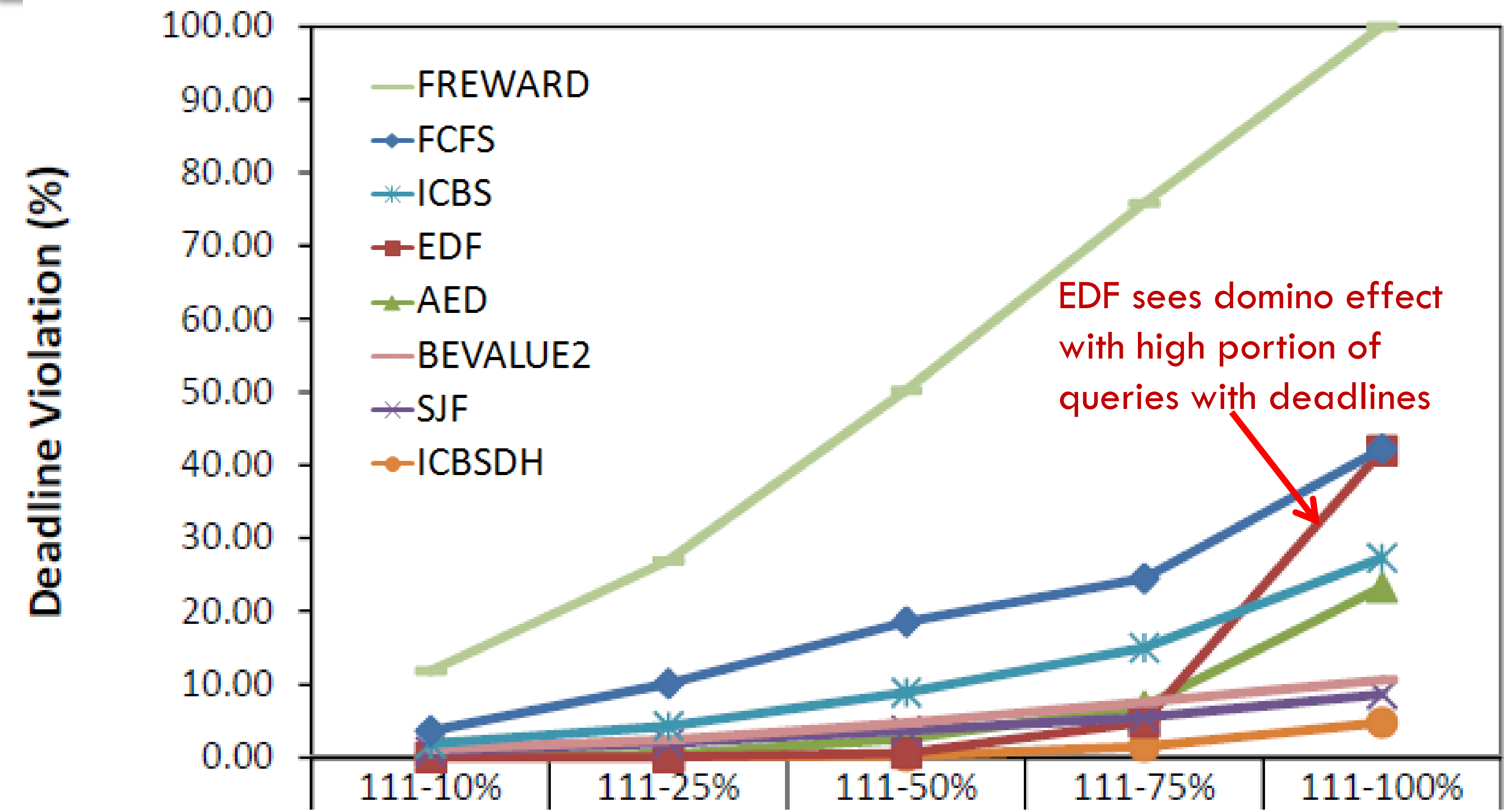


iCBS, BEValue2, iCBS-DH, SJF perform well.
iCBS performs the best, reliably.

iCBS-DH has high cost when cost step is earlier than deadline

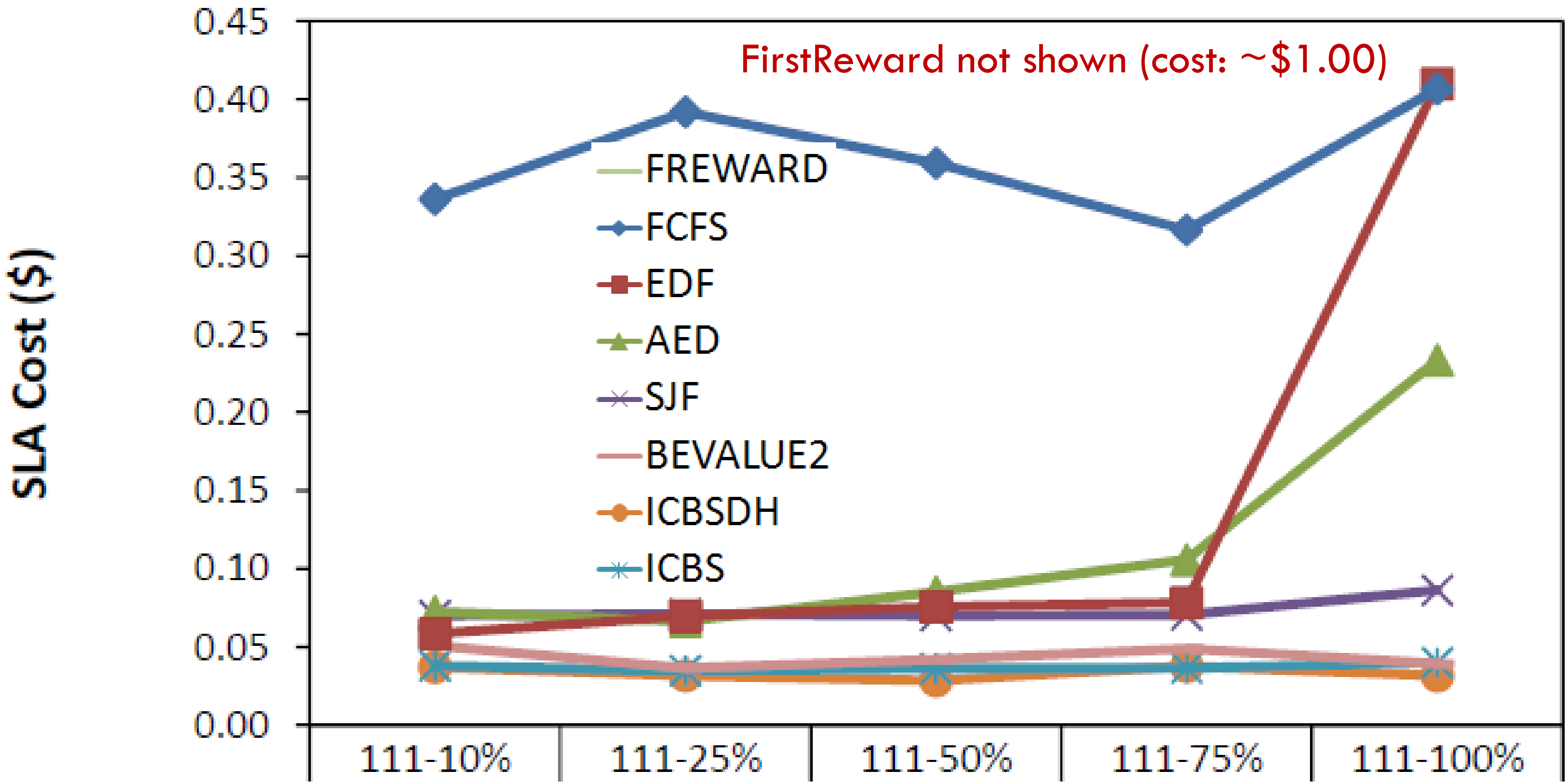
Result 2: Varying portion of queries that have deadlines, Violation

NEC Laboratories America
Relentless passion for innovation



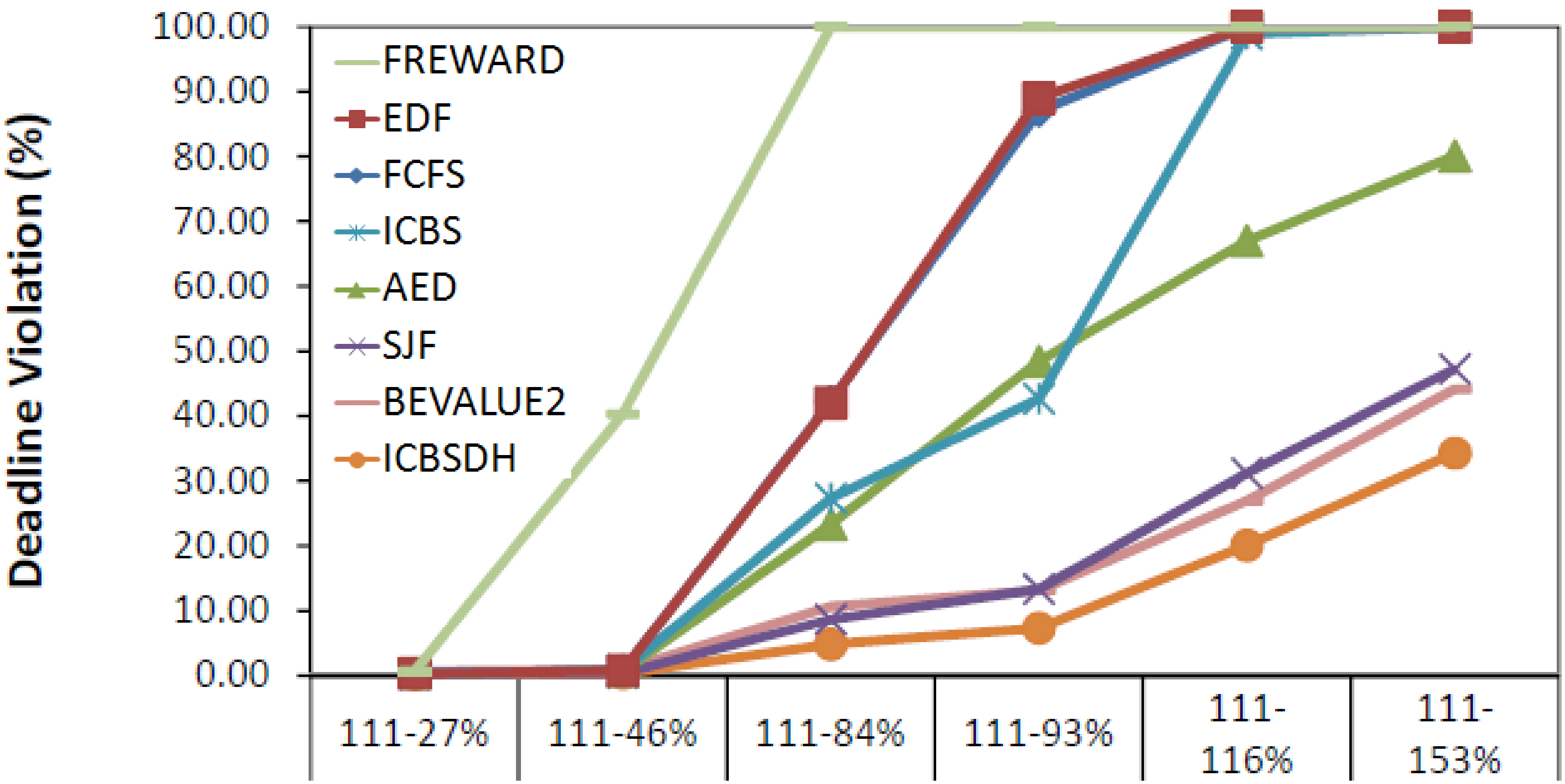
iCBS-DH, SJF, BEValue2 perform well. iCBS-DH perform the best, reliably.

Result 2: Varying portion of queries that have deadlines, Cost



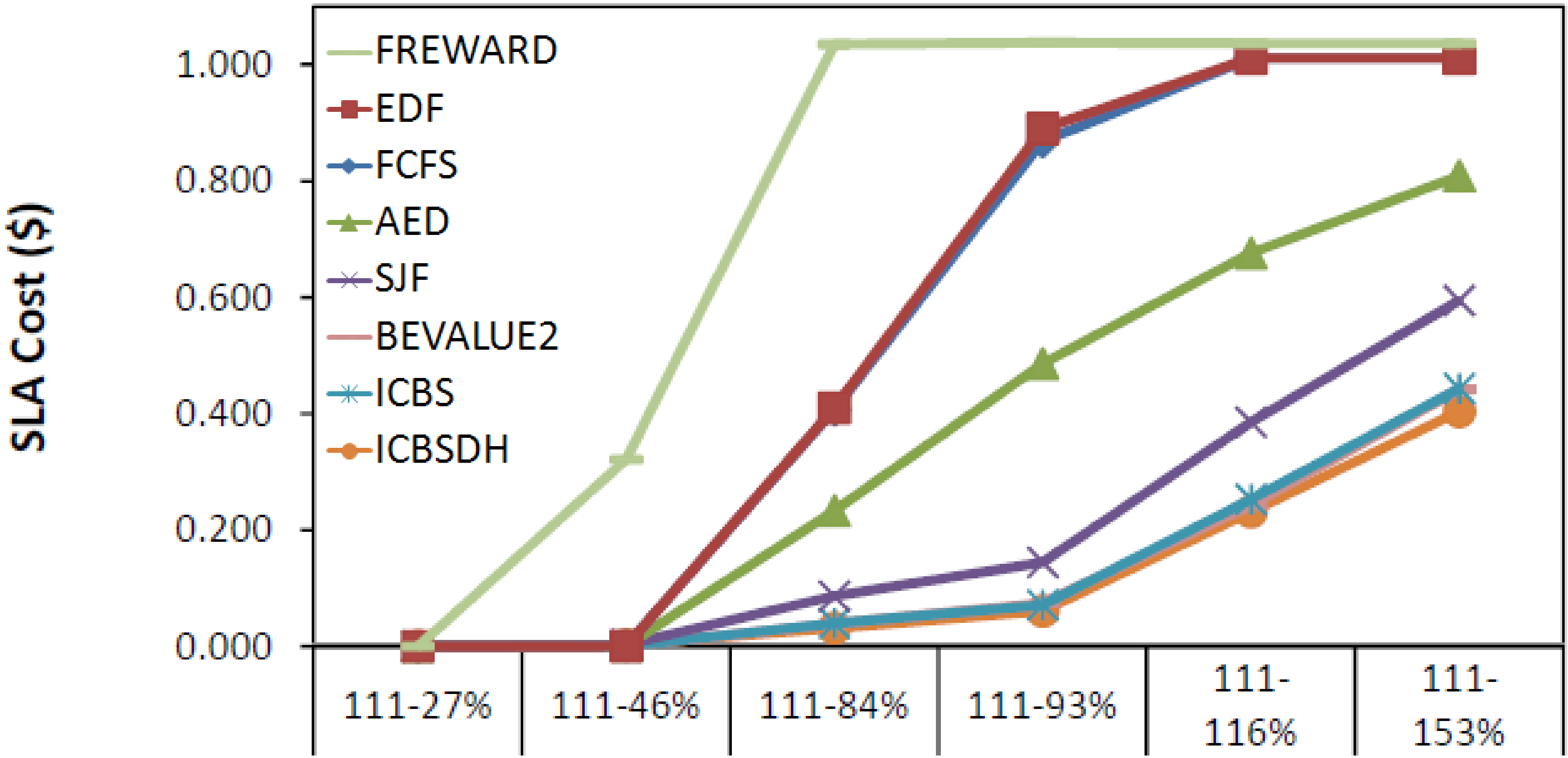
iCBS, iCBS-DH, BEValue2 perform well. iCBS-DH perform the best, reliably.

Result 3: Varying load, Violation



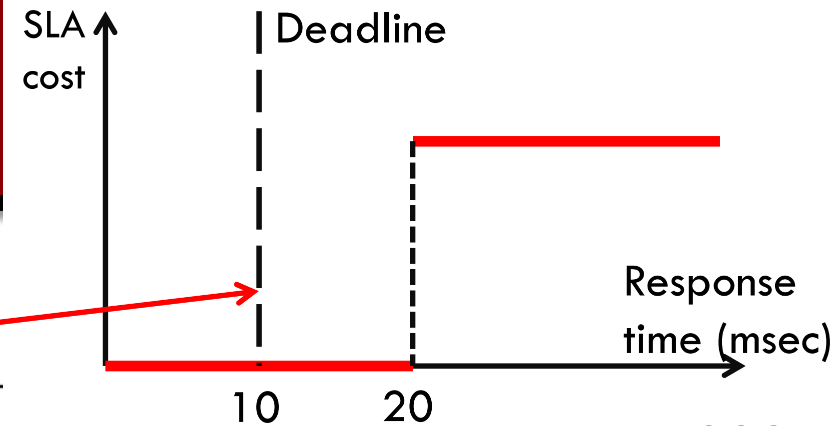
iCBS-DH, BEValue2, SJF perform well under overload.

Result 3: Varying load, Cost

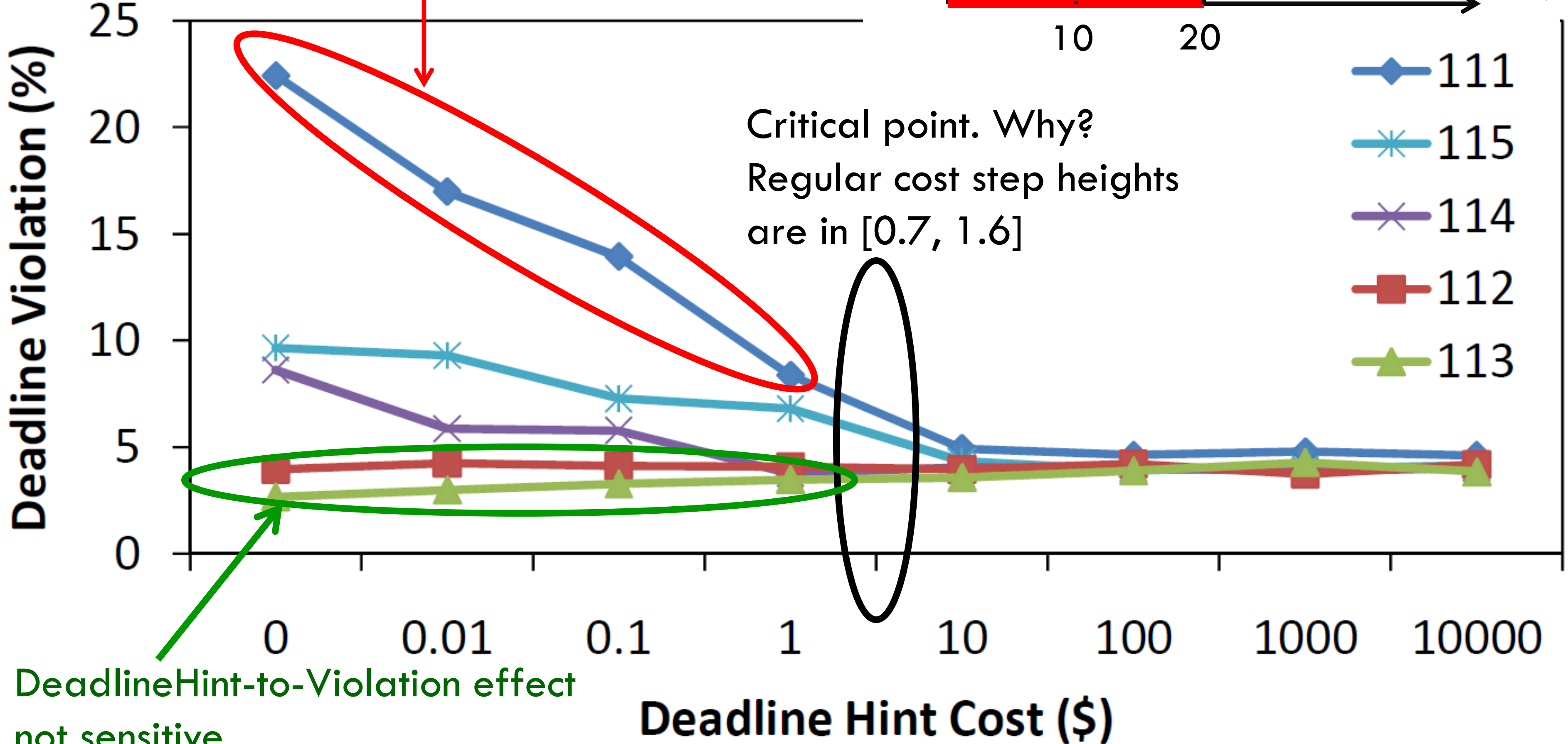


iCBS, iCBS-DH, BEValue2 perform well under overload.

Result 4: Deadline Hint

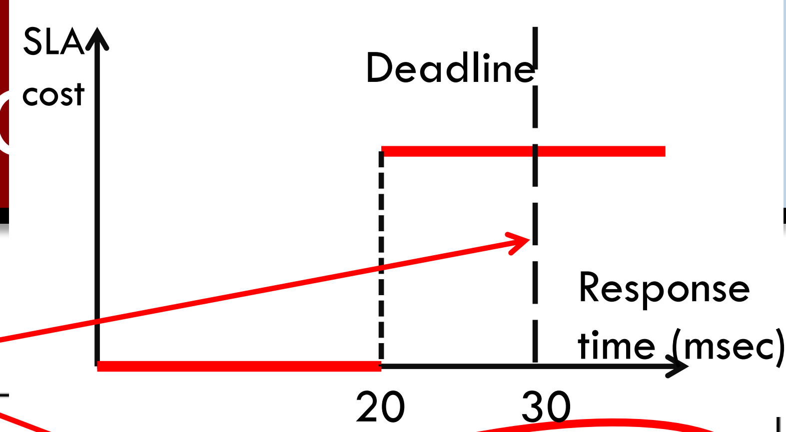


DeadlineHint-to-Violation effect most sensitive, when deadline is earlier than cost step.

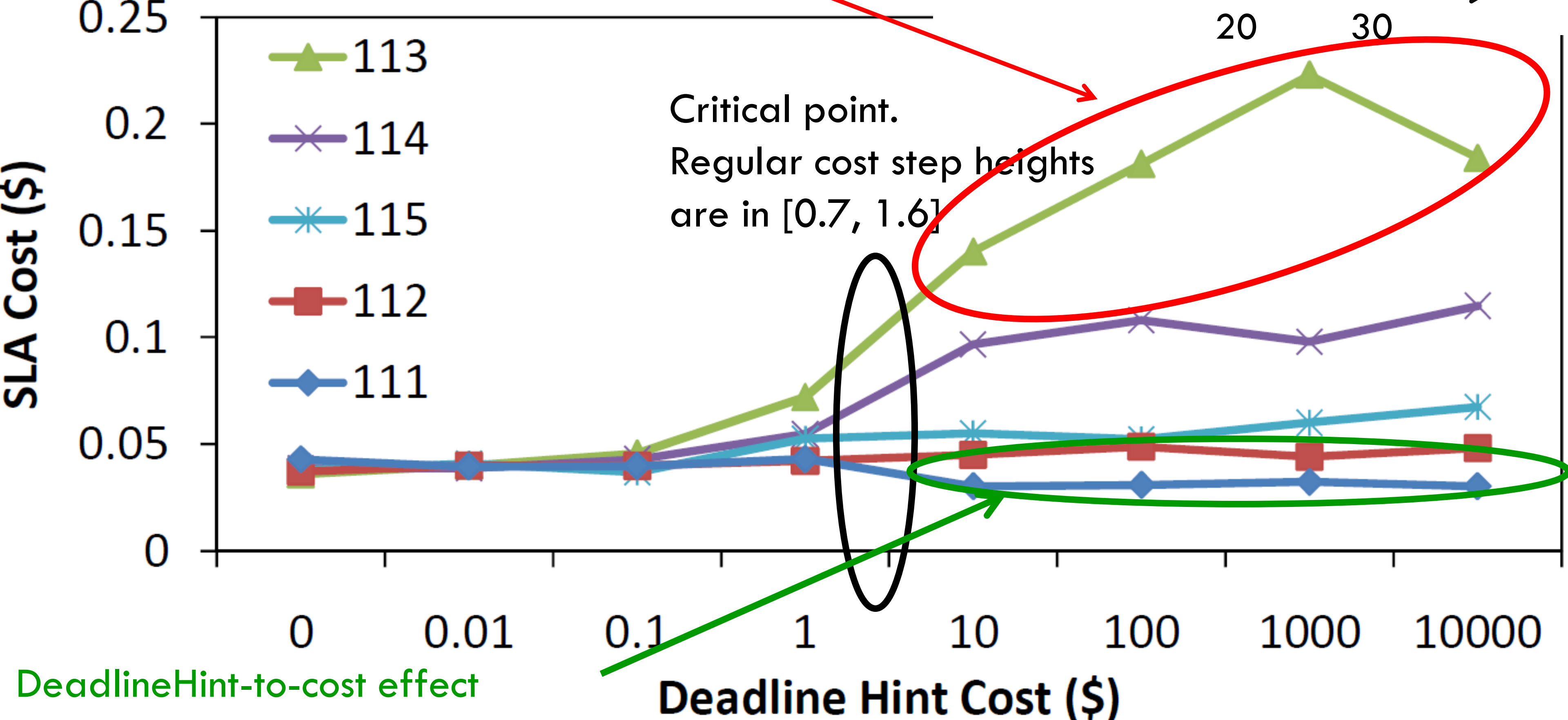


DeadlineHint-to-Violation effect not sensitive,
when deadline is later than cost step.

Result 4: Deadline Hint, C



DeadlineHint-to-Cost effect most sensitive, when cost step is earlier than deadline.



DeadlineHint-to-cost effect not sensitive, when cost is later than deadline.

Summary

	Cost-unaware	Cost-aware
Deadline-aware	Vulnerable to overload EDF AED [Haritsa91]	iCBS-DH: lowest deadline violation, low SLA cost
Deadline-unaware	FCFS SJF	BEValue2 [Jensen85] FirstReward [Irwin04] -O(n^2) iCBS [Chi11]: lowest SLA cost

Thank you!

- Any question or comments?