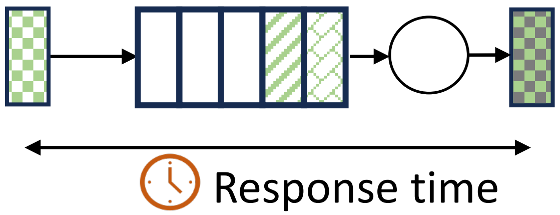
SkipPredict: When to Invest in Predictions for Scheduling

Rana Shahout and Michael Mitzenmacher

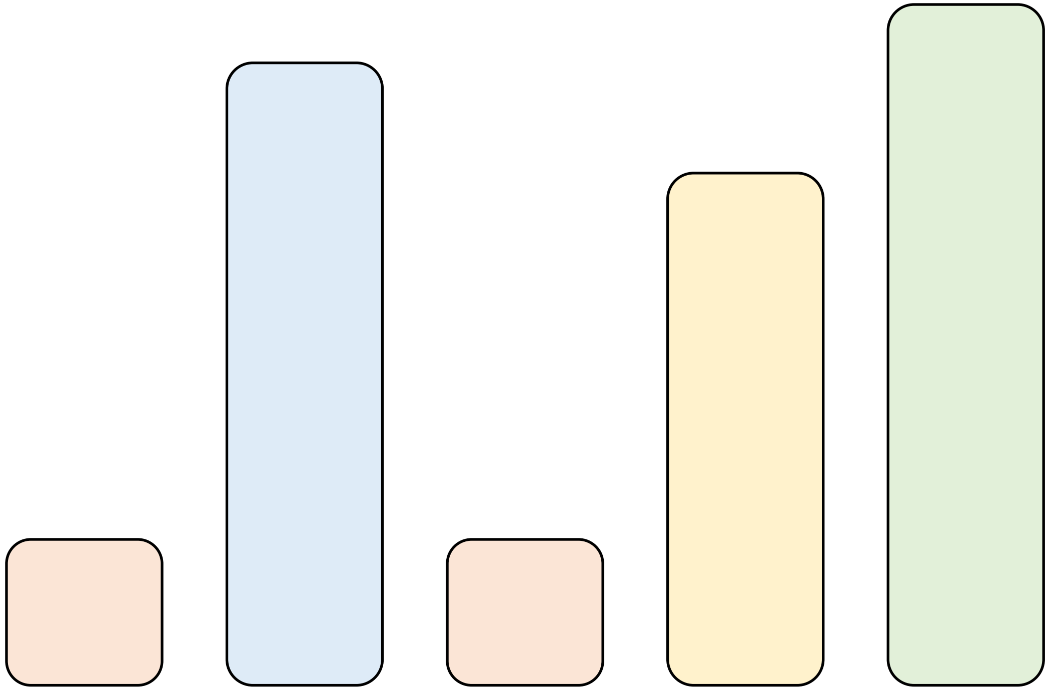
Harvard University

Goal: minimize response time for M/G/1 queueing systems



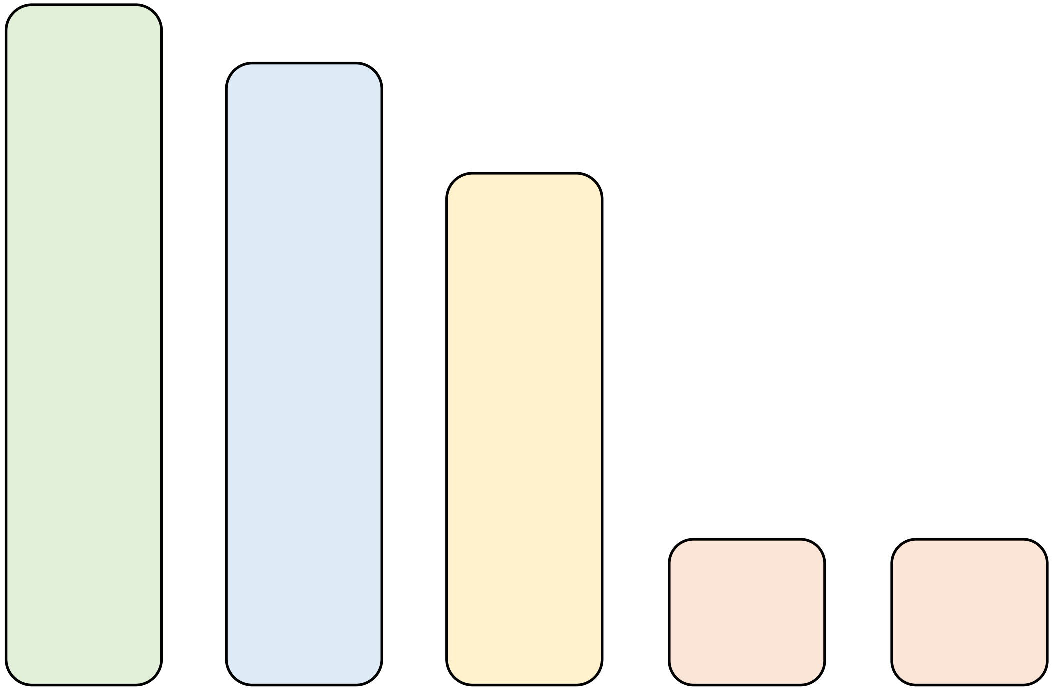
<u>No knowledge of job times</u>:

FCFS



<u>Exact knowledge of job times</u>:

Shortest Remaining Processing Time



How to Use Job Time Predictions in Scheduling?

- [Mitzenmacher 2019] examines a queue setting where jobs have predicted service times, deriving closed-form formulas for Shortest Predicted Remaining Processing Time (SPRPT) and other size-based policies.
- [Mitzenmacher 2021] studies the same setting with only a "1-bit” prediction, categorizing jobs as either short or long.

- But existing works (and learning-augmented algorithms in general) assume predictions come without cost...

Motivating Questions

- When does the use of predictions, including their computation, justify their costs?
- In scheduling, where we have multiple tasks, should all jobs be treated uniformly by computing predictions for each one?

Two Models

External Cost Model

- Predictions are generated by some external method without impacting job service times but incurring a cost.
- $cost = f(response\ time, prediction\ cost)$

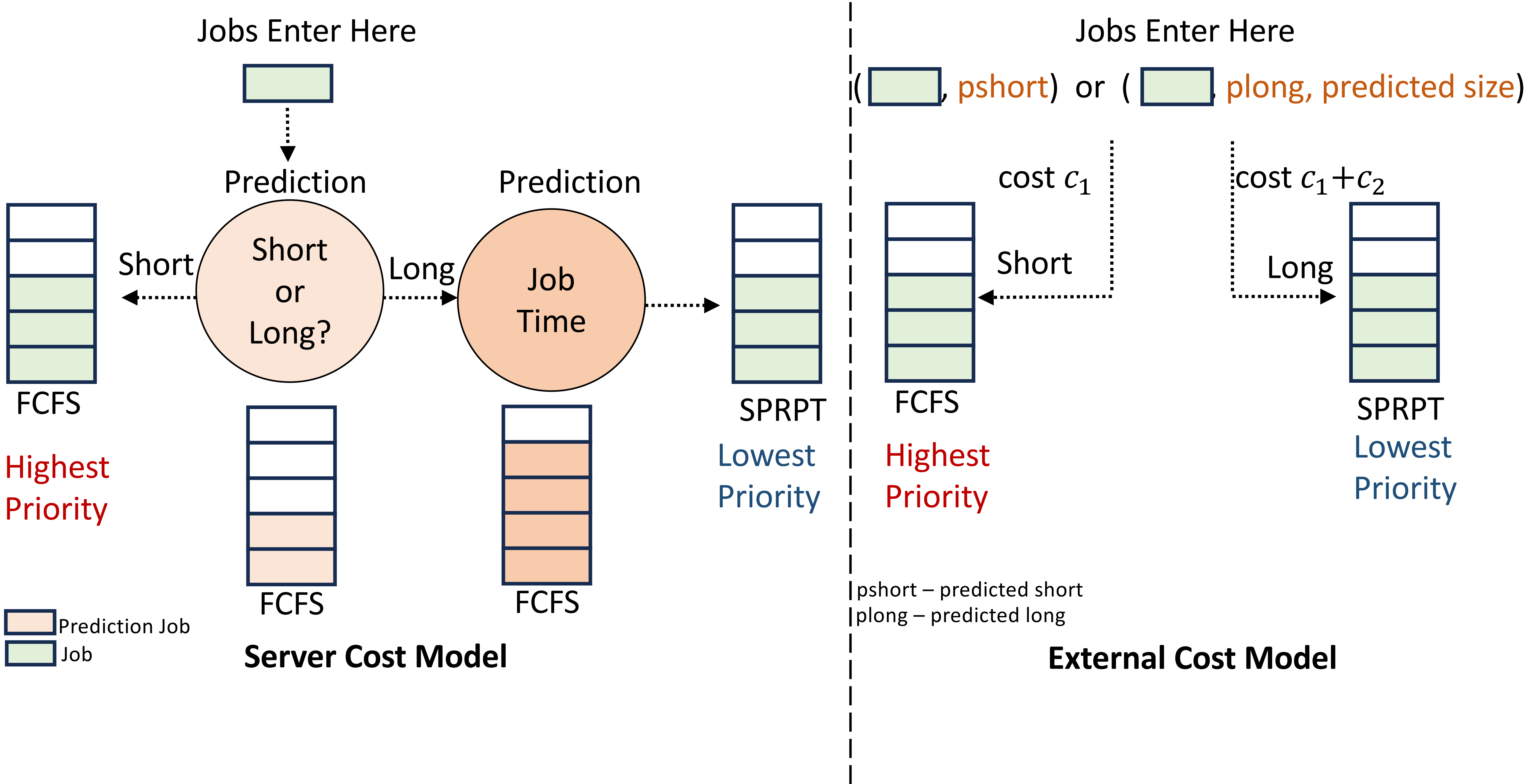
Server Cost Model

- Jobs and predictions run on same server
- $cost = response\ time$

SkipPredict Overview

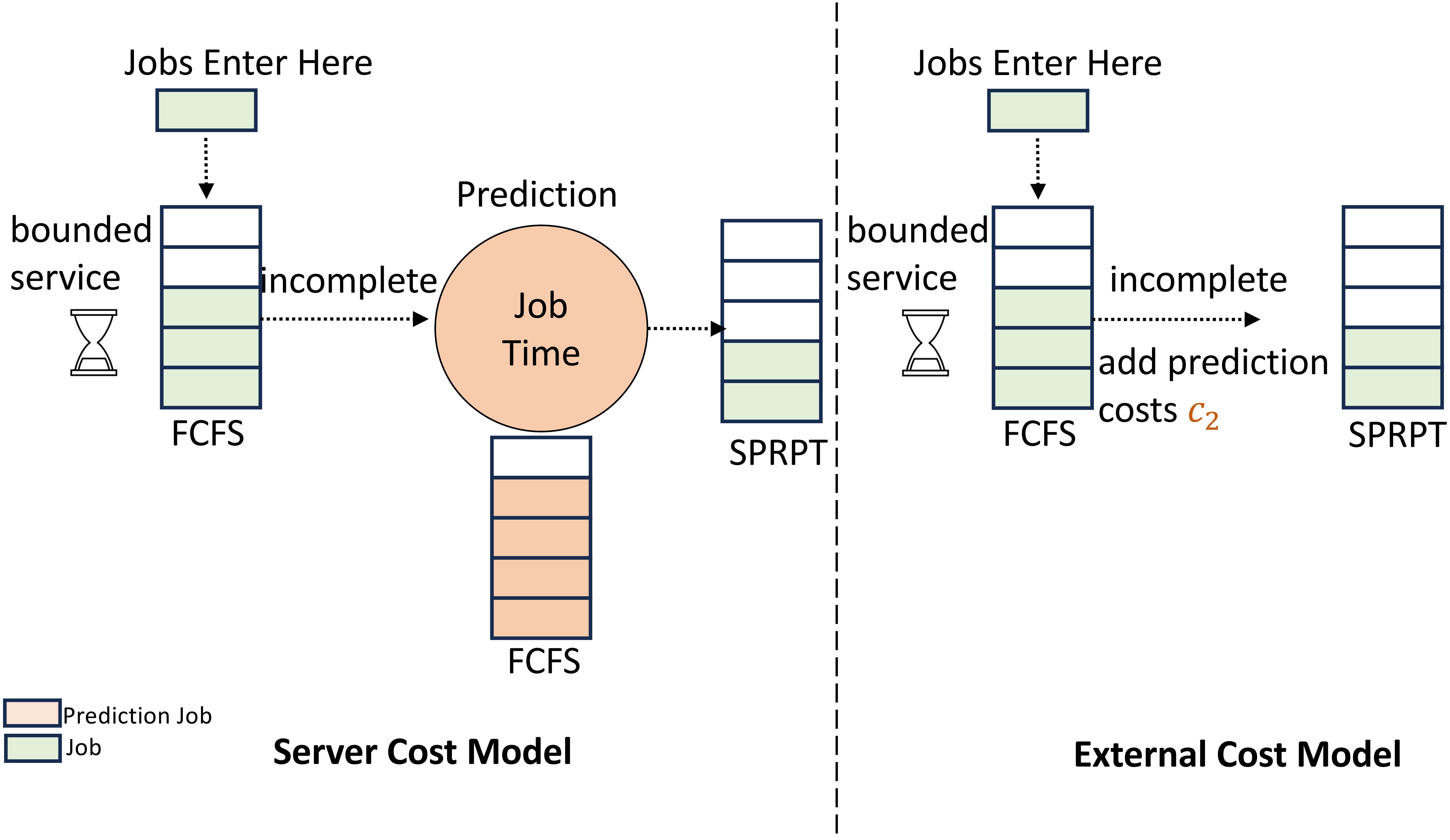
- First classify jobs as short or long using "cheap predictions" (1 bit). Then use "expensive predictions" (for job size estimates) only for long jobs (where it is worthwhile).

- Jobs below threshold T (short) are prioritized and scheduled by FCFS.
- Jobs above T (long) receive further size prediction and are scheduled by SPRPT.


Jobs Enter Here
Prediction
Short or Long?
Short
Long
Prediction
Job Time
FCFS
Highest Priority
FCFS
FCFS
SPRPT
Lowest Priority
Prediction Job
Job
Server Cost Model
Jobs Enter Here
(🟩, pshort) or (🟩, plong, predicted size)
cost c_1
cost c_1+c_2
Short
Long
FCFS
Highest Priority
SPRPT
Lowest Priority
pshort – predicted short
plong – predicted long
External Cost Model

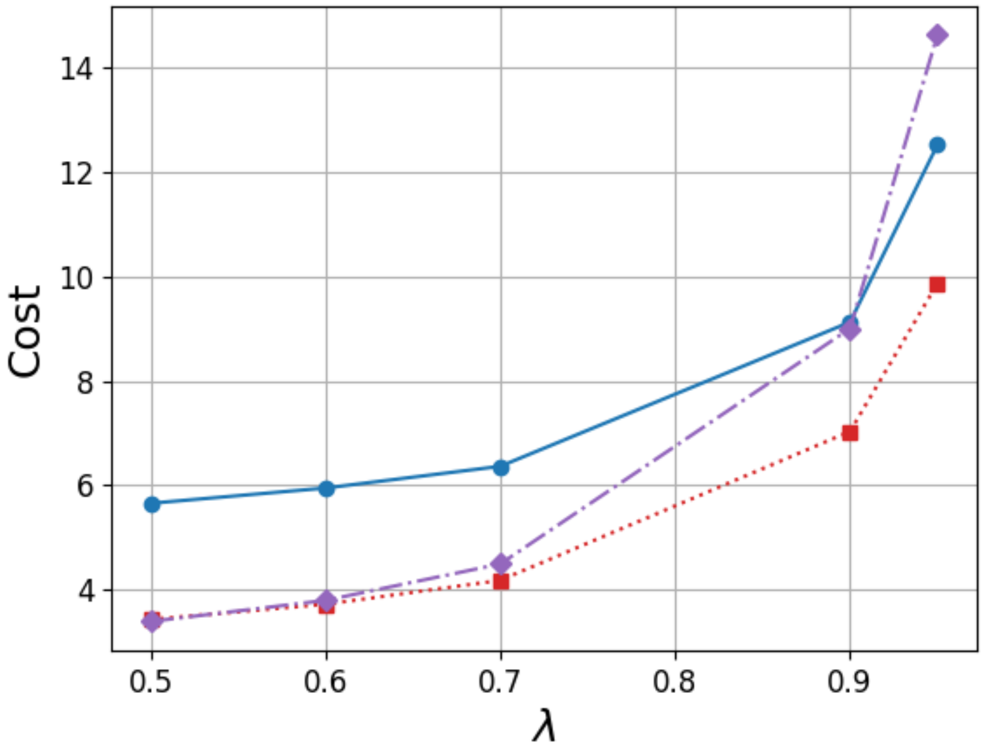
DelayPredict Overview

- Goal: Avoid short/long predictions for every job, but obtain a similar benefit.
- DelayPredict initially assumes all jobs are short and runs them for time up to L.
- If the job has service time > L, it is long. After L service preempt it and predict it as with SkipPredict.

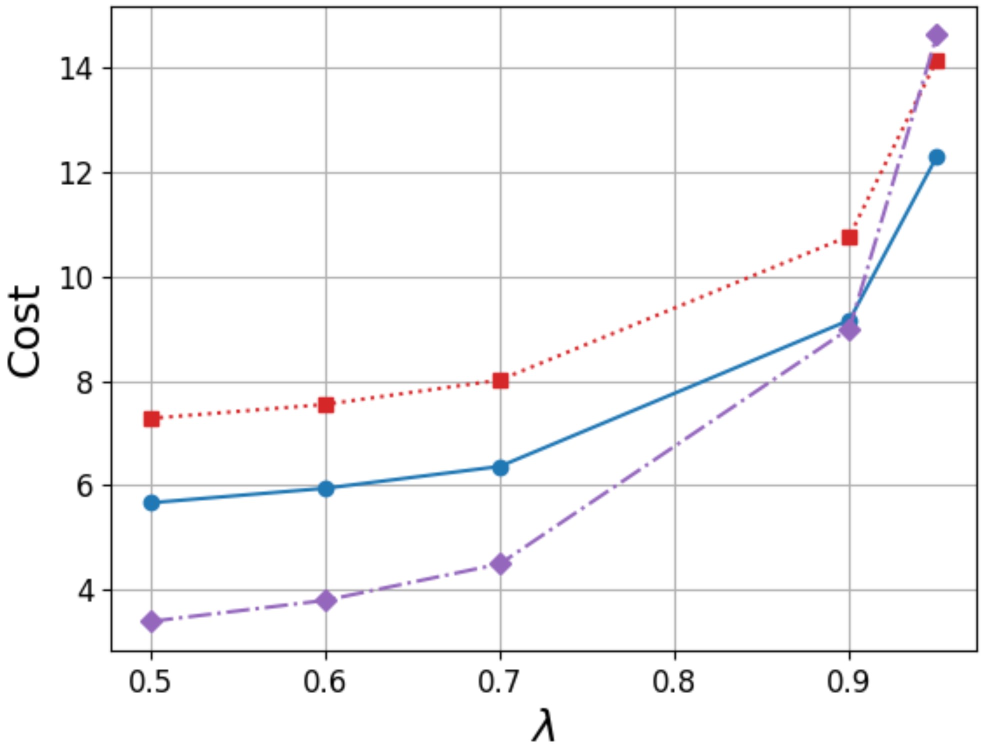

Jobs Enter Here
bounded
service
FCFS
incomplete
Prediction
Job
Time
FCFS
SPRPT
Prediction Job
Job
Server Cost Model
Jobs Enter Here
bounded
service
FCFS
incomplete
add prediction
costs c_2
SPRPT
External Cost Model

- In the paper, we provide closed-form equations for SkipPredict and DelayPredict, along with proofs using the SOAP

SkipPredict leads with a large cost gap; with a small gap, DelayPredict performs best



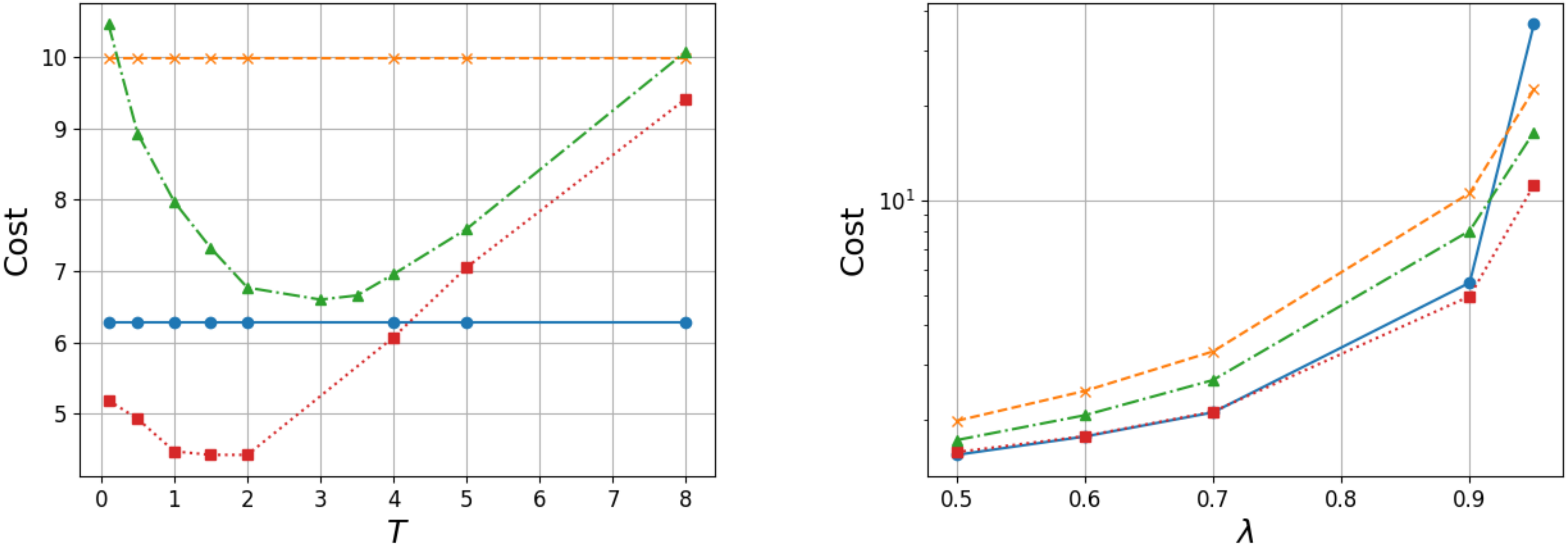
Large cost gap, $c_1 = 0.5, c_2 = 4$



Small cost gap, $c_1 = 3.5, c_2 = 4$

● SPRPT 🟥 SkipPredict ◆ DelayPredict λ *arrival rate*

SkipPredict’s advantage increases with larger cost gaps


Cost
T
Cost
10^1
λ