Understanding and Exploiting Amazon EC2 Spot Instances

Carey Williamson
Department of Computer Science
University of Calgary

Introduction and Motivation

- Within Amazon’s Elastic Compute Cloud (EC2) offerings, clients have a choice between services:
  - On-demand instances: reserve for given price and duration
  - Spot instances: cheaper price; unpredictable duration

- Challenges:
  - Spot instance prices fluctuate a lot!
  - Duration of instance availability unknown
  - Bid failures and revocations adversely impact QoS
  - How to “optimize” cost and performance in EC2 markets?

Example: On-Demand Prices in Amazon EC2

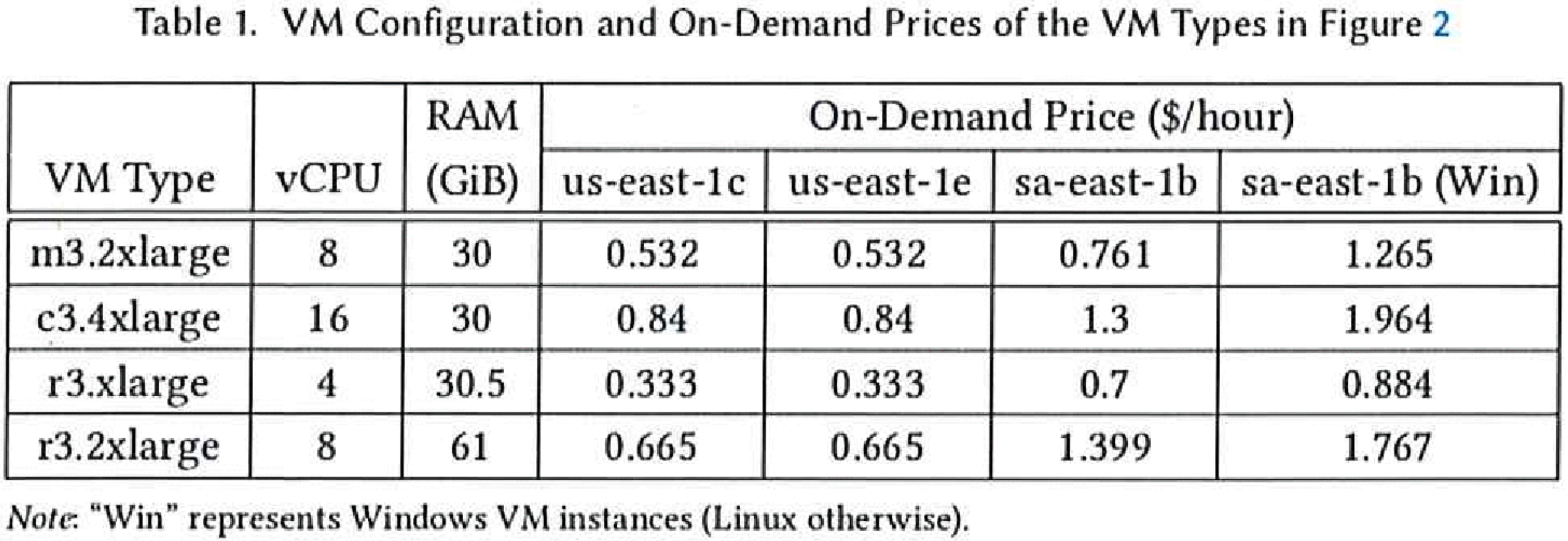
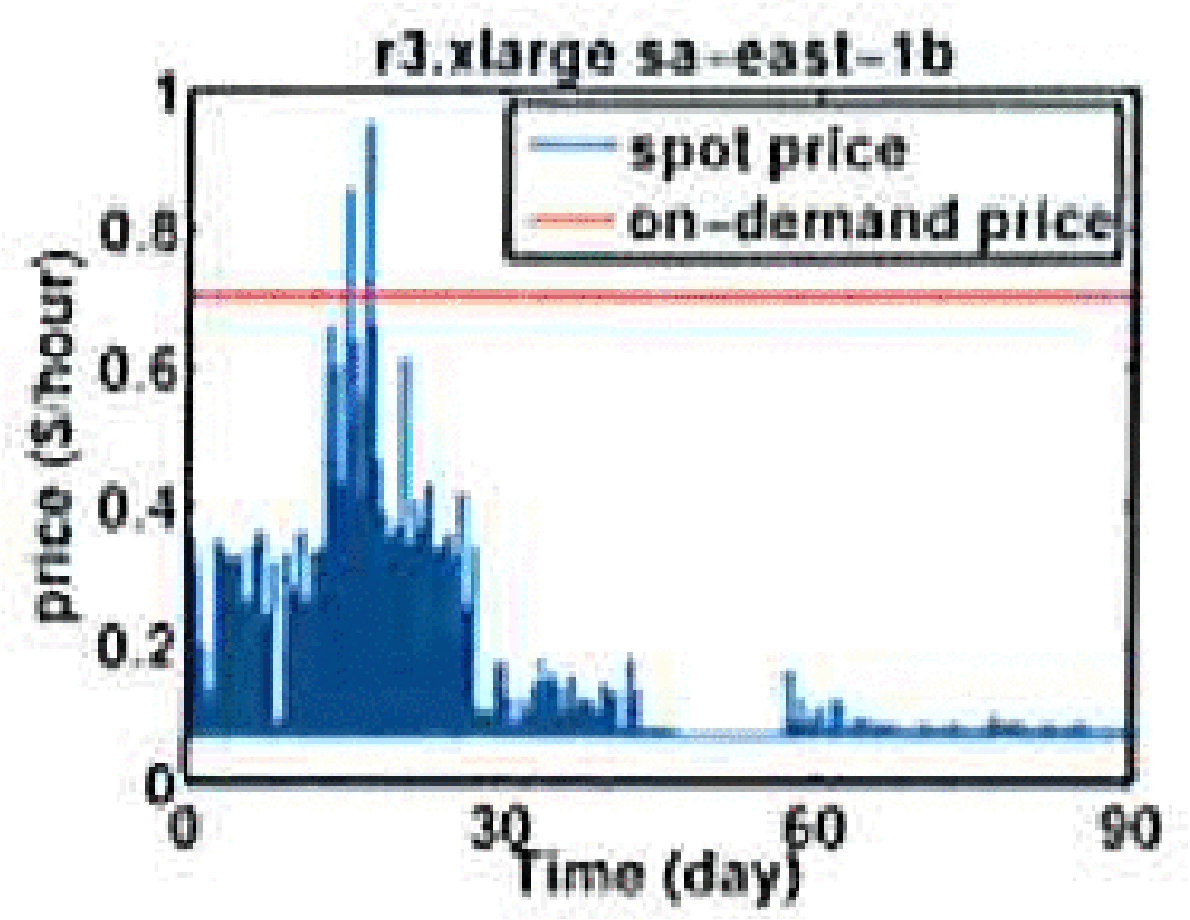


Table 1. VM Configuration and On-Demand Prices of the VM Types in Figure 2

VM Type	vCPU	RAM (GiB)	On-Demand Price ($/hour)			
			us-east-1c	us-east-1e	sa-east-1b	sa-east-1b (Win)
m3.2xlarge	8	30	0.532	0.532	0.761	1.265
c3.4xlarge	16	30	0.84	0.84	1.3	1.964
r3.xlarge	4	30.5	0.333	0.333	0.7	0.884
r3.2xlarge	8	61	0.665	0.665	1.399	1.767

Note: "Win" represents Windows VM instances (Linux otherwise).

An Example of Spot Instance Dynamics



More Examples of Spot Instance Dynamics

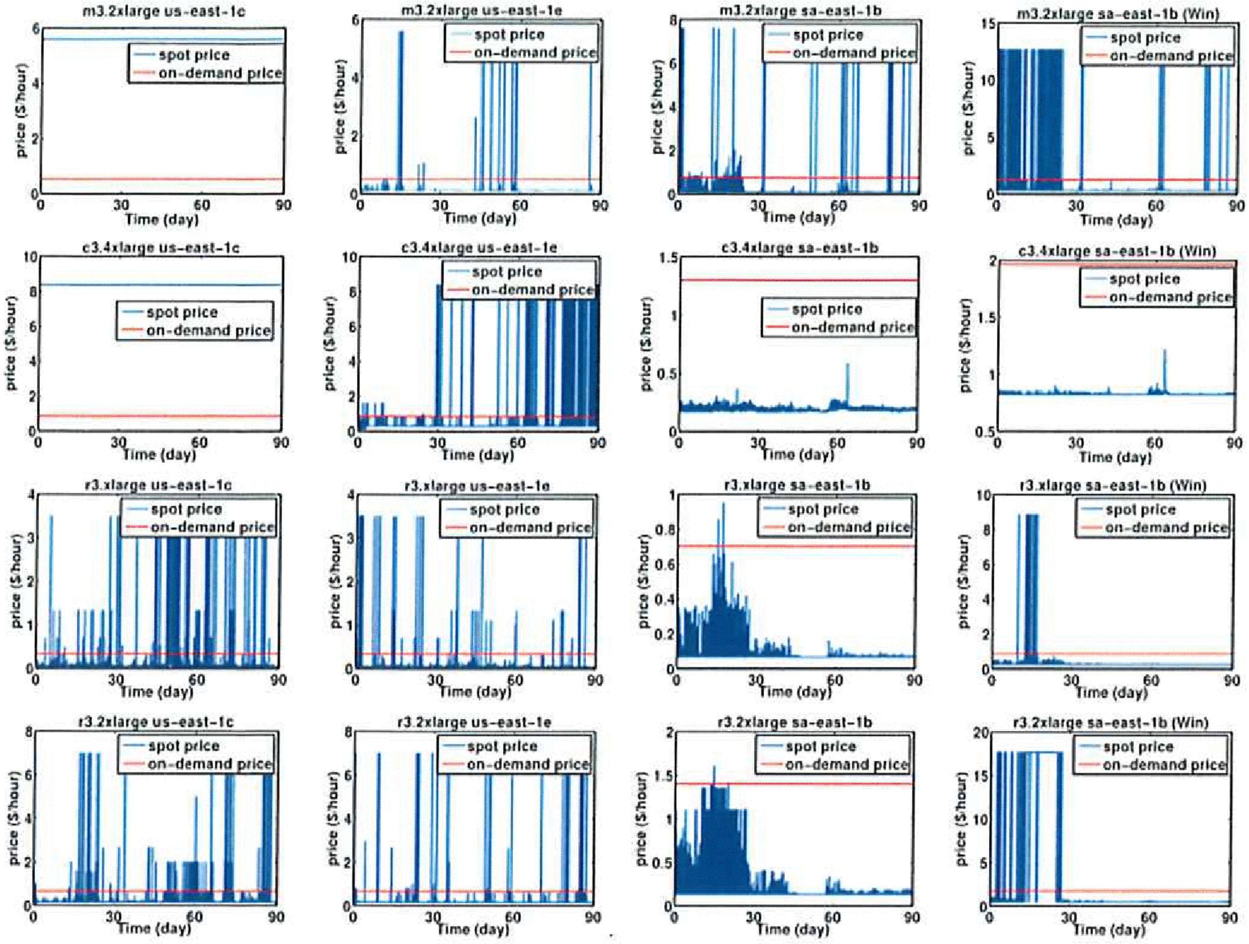


Fig. 2. Sample spot price time series collected during the 90-day period (03/12/2017 to 06/10/2017), chosen due to their very different properties. We show spot price traces from two EC2 global regions, us-east-1 (US East) and sa-east-1 (South America). Within each region, we pick different availability zones, denoted by the tail characters c, e, and b. For traces from the same availability zone, we use "Win" to represent Windows VMs (Linux otherwise). For each availability zone, we select four VM types: m3.2xlarge (general purpose), c3.4xlarge (compute optimized), and r3.xlarge and r3.2xlarge (memory optimized).

Spot Instance Terminology

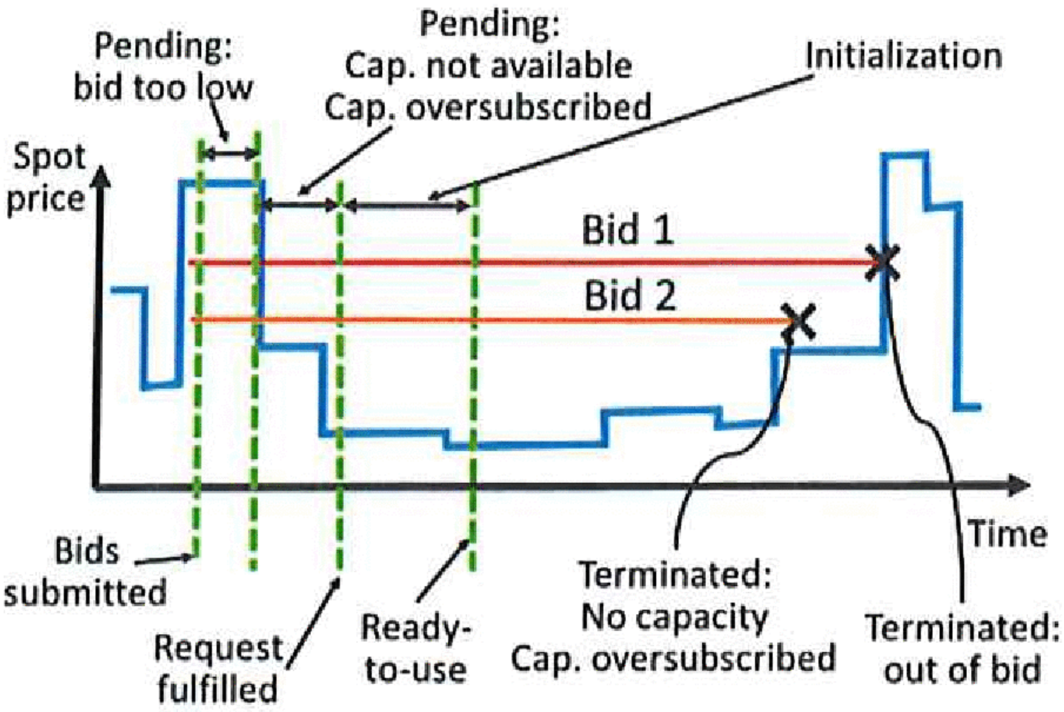


Fig. 1. Illustration of a spot instance operation using two hypothetical bids (Bid 1 and Bid 2).

This Paper

- Research Questions:
  - After bidding, how long does it take to get an instance?
  - What is the expected lifetime of an instance?
  - What is the average cost incurred for an instance?
  - What is the probability of instance revocations?
  - How can spot instance features be exploited to devise efficient and cost-effective procurement algorithms?
- Two main contributions:
  - Empirical study of Amazon EC2 instances (2015 and 2017)
  - Two case studies to evaluate effectiveness of models

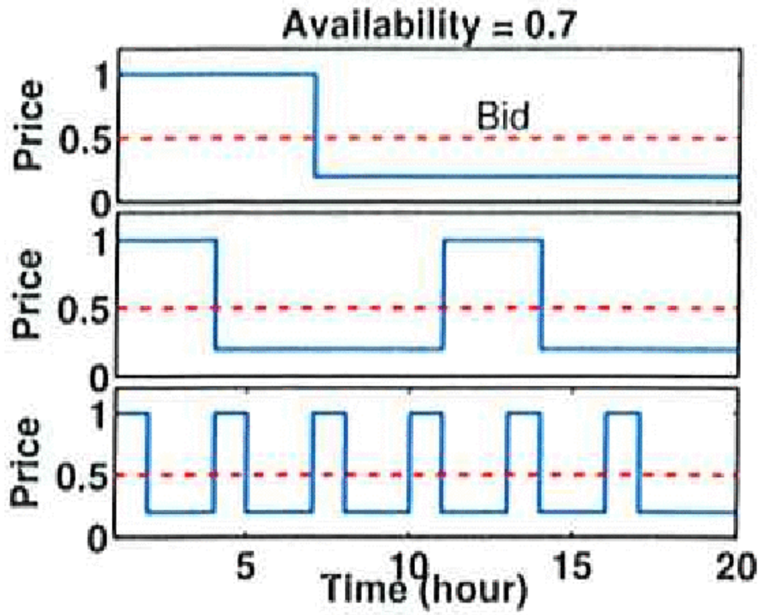
C. Wang, Q. Liang, and B. Urgaonkar, "An Empirical Analysis of Amazon EC2 Spot Instance Features Affecting Cost-Effective Resource Procurement", ACM ToMPECS, Vol. 3, No. 2, Article 6, pp. 1-24, March 2018. (Extended journal version of earlier ICPE 2017 paper)

Empirical Analysis (preview)

- Feature 1: Lifetime of spot instances
  - Exhibits sufficient temporal dependence for prediction
- Feature 2: Average cost during lifetime
  - Exhibits sufficient temporal dependence for prediction
- Feature 3: Simultaneous revocations
  - Some spatial dependence; weaker temporal dependence
- Feature 4: Time to start a spot instance
  - No obvious structural properties that can be exploited

Figure 3

- Simple CDF-based approaches from the literature are not sufficient for this problem , since they do not distinguish the following three (synthetic) scenarios



- Need to explicitly consider “lifetimes” of instances

Figure 4

- Illustration of “lifetime” for a successful bid
- Illustration of “average cost” for a successful bid

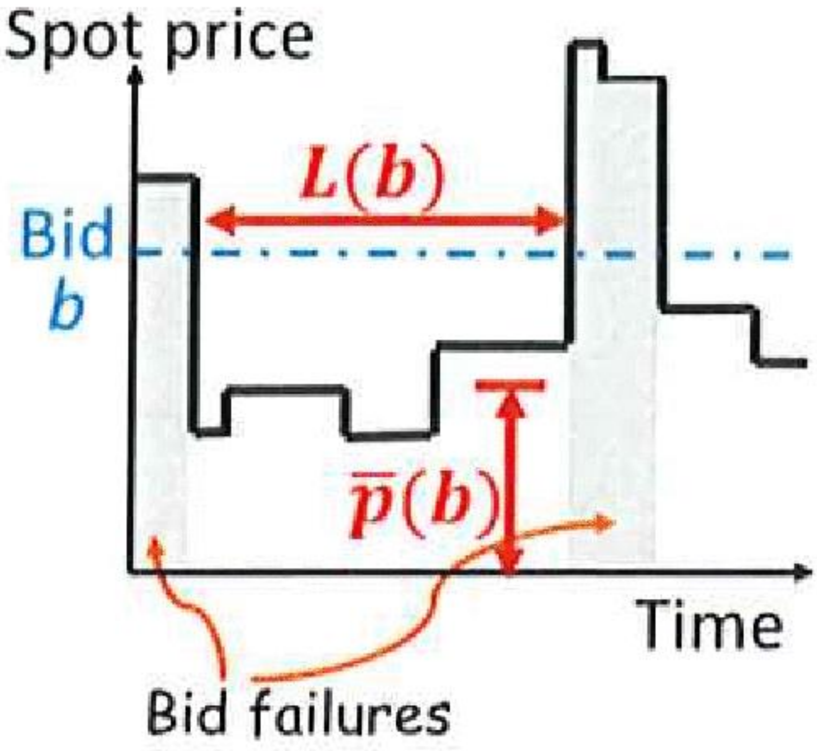


Figure 5

- Some markets may be strongly correlated (right), while others may only be weakly correlated (left)
- Correlated markets might have correlated bid failures (i.e., multiple spot instances revoked at similar time)
- Results are conditional upon bid values though!

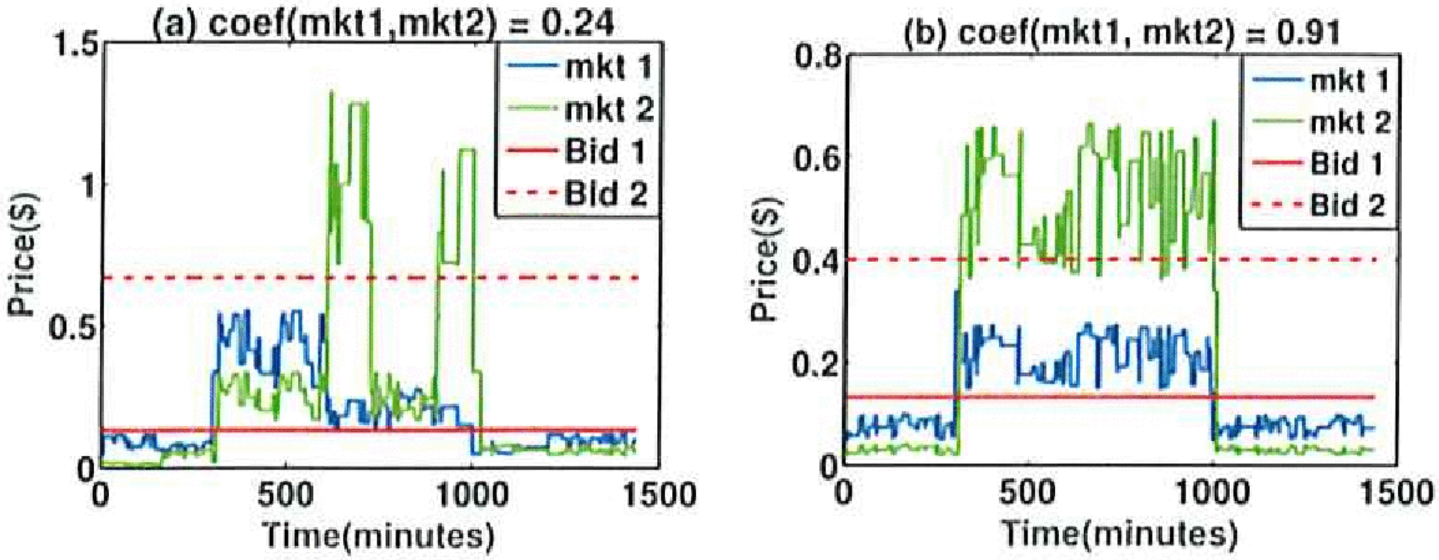


Figure 6

- Empirical correlation results
- Observations:
  - — Stronger spatial relationship within regions than across
  - — Detected by both metrics
  - — Proposed approach detects some spatial corr across regions
  - — Weak spatial correlation observed for instance types

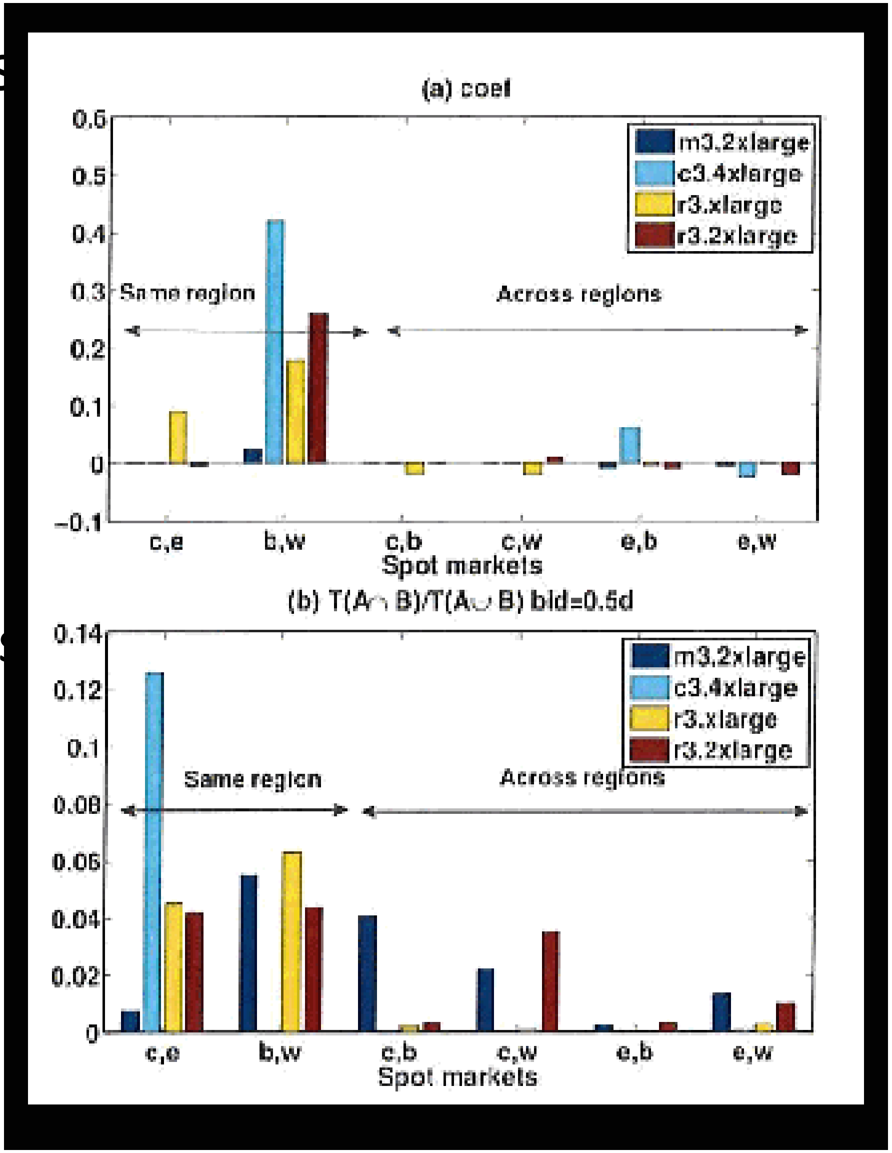


Figure 7

- Some temporal correlation in start up times
- No obvious correlation structure based on bids

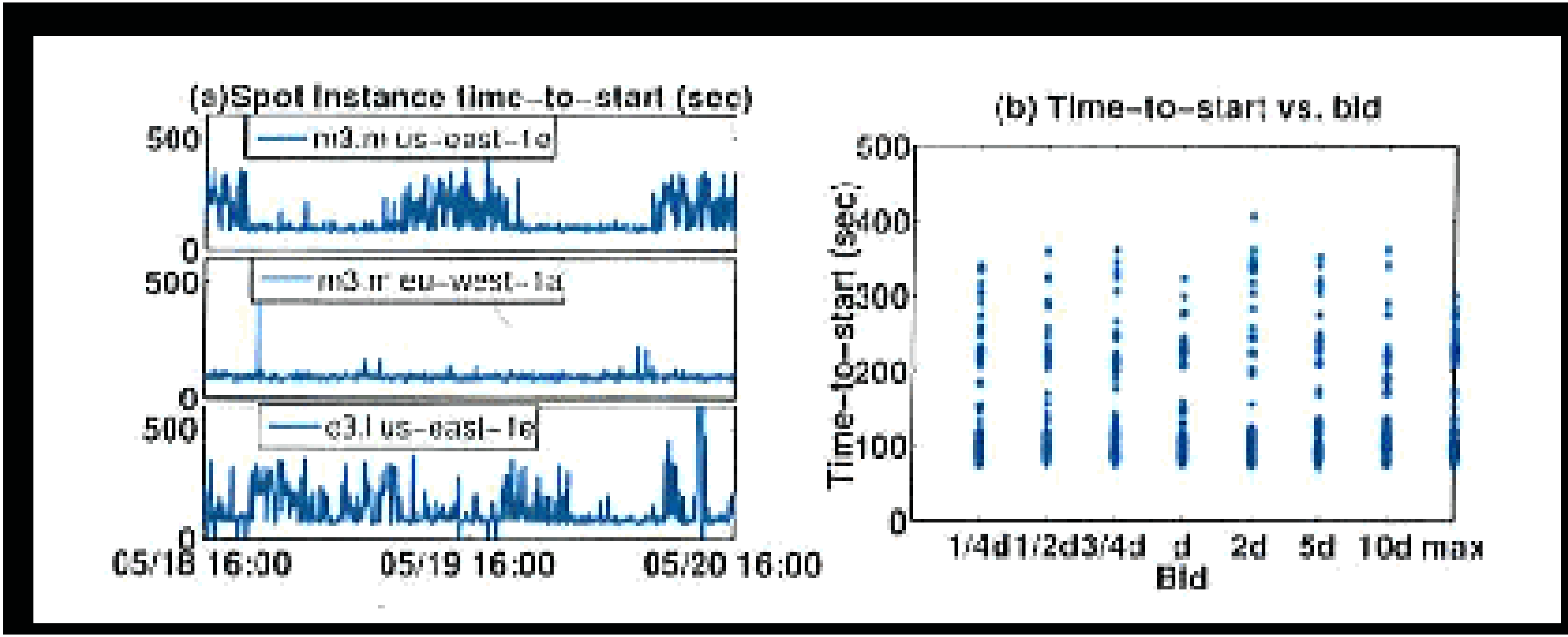
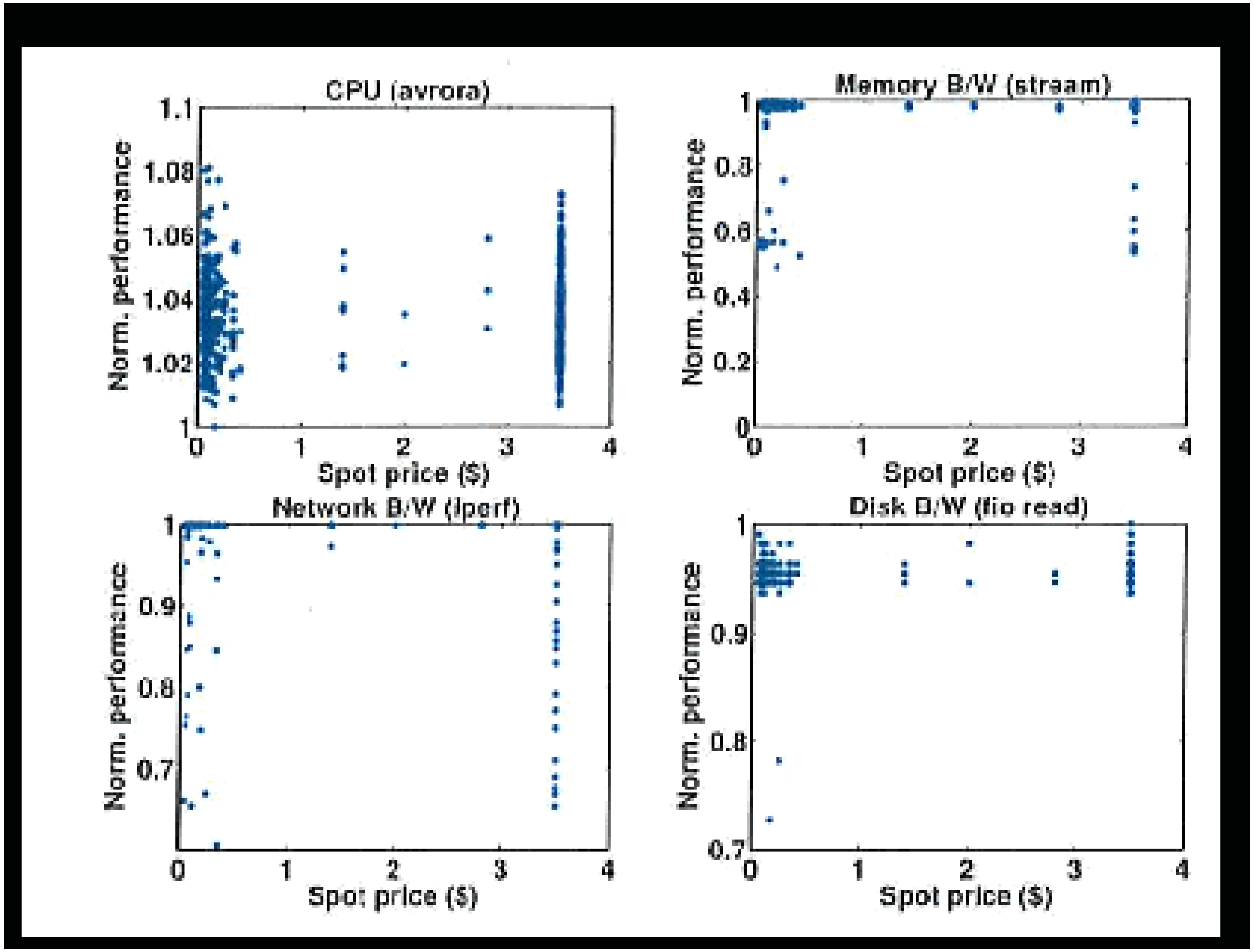


Figure 8

- Empirical measurements of "effective capacity"
- Relatively small variation for any of the metrics



Empirical Analysis (review)

- Feature 1: Lifetime of spot instances
  - Exhibits sufficient temporal dependence for prediction
  - Using 7-14 days of recent past history gives best results
- Feature 2: Average cost during lifetime
  - Exhibits sufficient temporal dependence for prediction
  - Using 7-14 days of recent past history gives best results
- Feature 3: Simultaneous revocations
  - Some spatial dependence; weaker temporal dependence
  - Suggests splitting jobs across regions and instance types
- Feature 4: Time to start a spot instance
  - No obvious structural properties that can be exploited

Case Studies (preview)

- Application 1: Latency-sensitive data processing
  - In-memory key-value store (memcached)
- Application 2: Delay-tolerant batch computing
  - Primary spot instances with backup on-demand instances
- For each application, two types of evaluation:
  - Trace-driven simulation (90 days of spot instance pricing)
  - System prototyping (24-hour EC2 expt, real-time control)

Figure 9

- Wikipedia workload characteristics
- Clear diurnal patterns
- Time-varying working set size

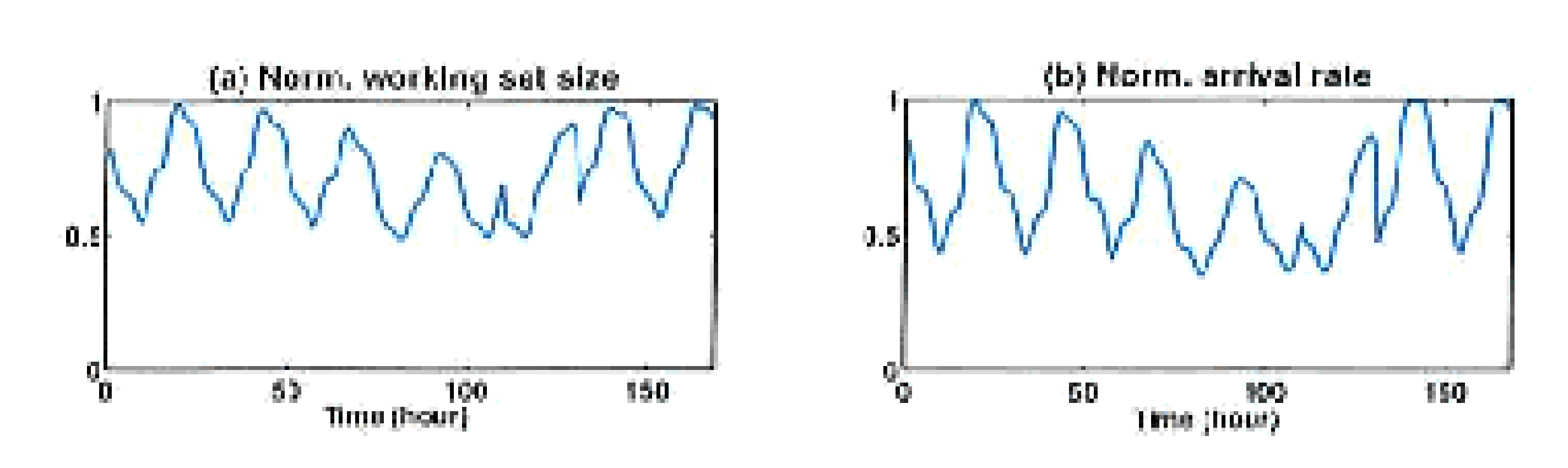


Figure 10

- Trace-driven simulation evaluation of approaches
- Data loss performance for Application 1

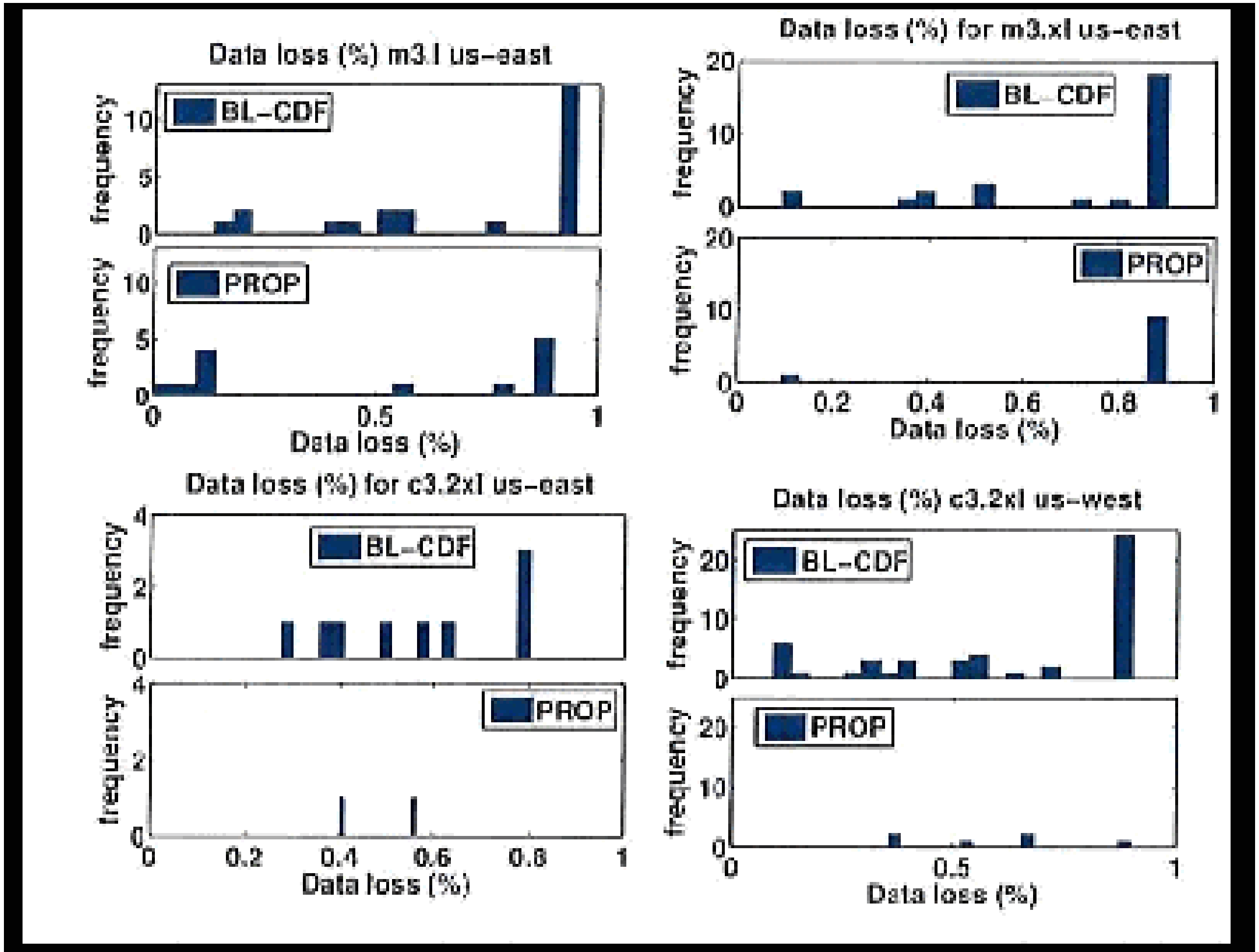


Figure 11

- Experimental evaluation/comparison of approaches
- Application 1: memcached

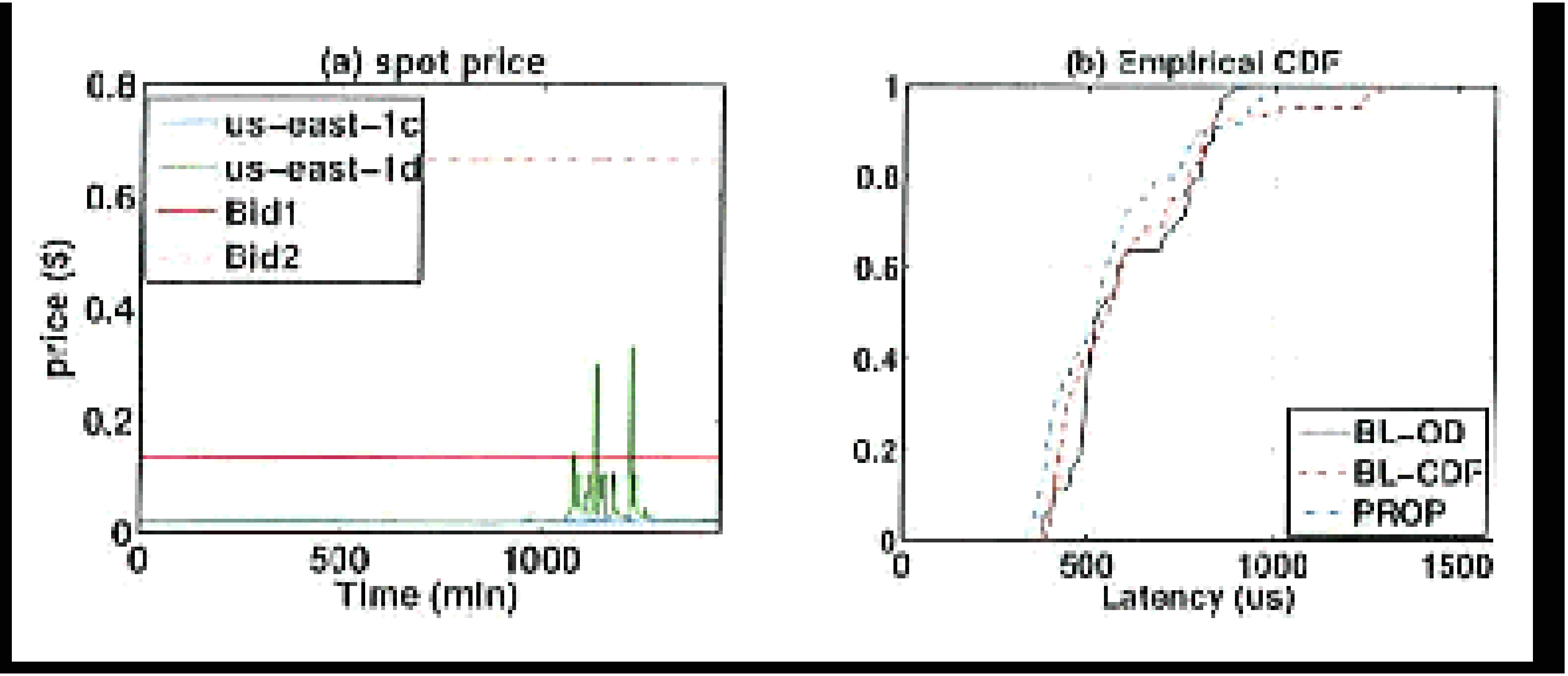


Figure 12

- Trace-driven simulation evaluation of approaches
- Application 2: batch computing

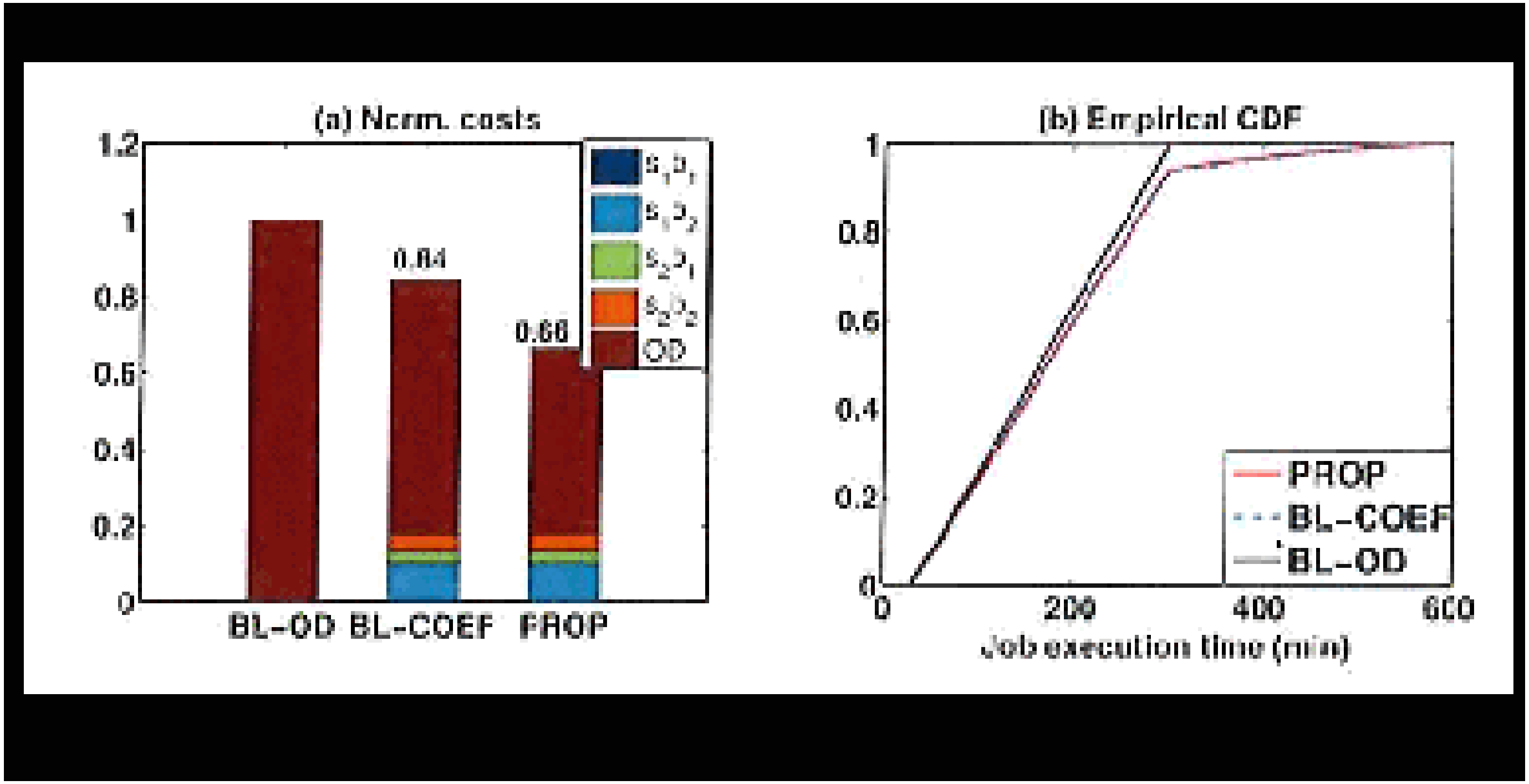
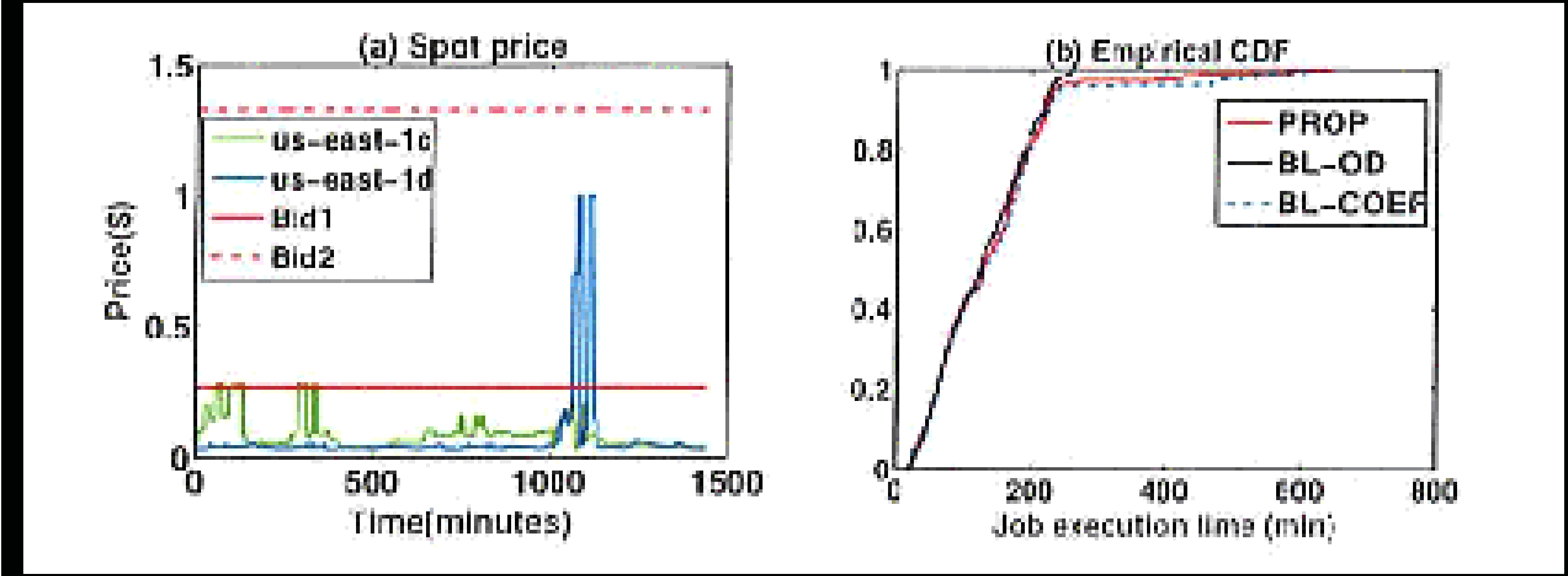


Figure 13

- Experimental evaluation/comparison of approaches
- Application 2: batch computing



Case Studies (review)

- Application 1: Latency-sensitive data processing
  - In-memory key-value store (memcached)
  - Comparable cost but much better performance
- Application 2: Delay-tolerant batch computing
  - Primary spot instances with backup on-demand instances
  - Comparable performance with 18% lower costs
- For each application, two types of evaluation:
  - Trace-driven simulation (90 days of spot instance pricing)
  - System prototyping (24-hour EC2 expt, real-time control)

Summary and Conclusions

- There are four key features of Amazon EC2 instances that need to be considered (i.e., spot instance lifetime; average price during lifetime; simultaneous revocations; time to start)
- Temporal and spatial dependence in some features can be exploited to improve predictions
- Need to consider impacts of (simultaneous) revocations
- Need to condition results based on bid price
- Two real-world case studies demonstrate the effectiveness of the proposed approach in practice
  - Can save 20-70% on costs versus on-demand instances
  - Can provide flexible tradeoffs between cost and performance