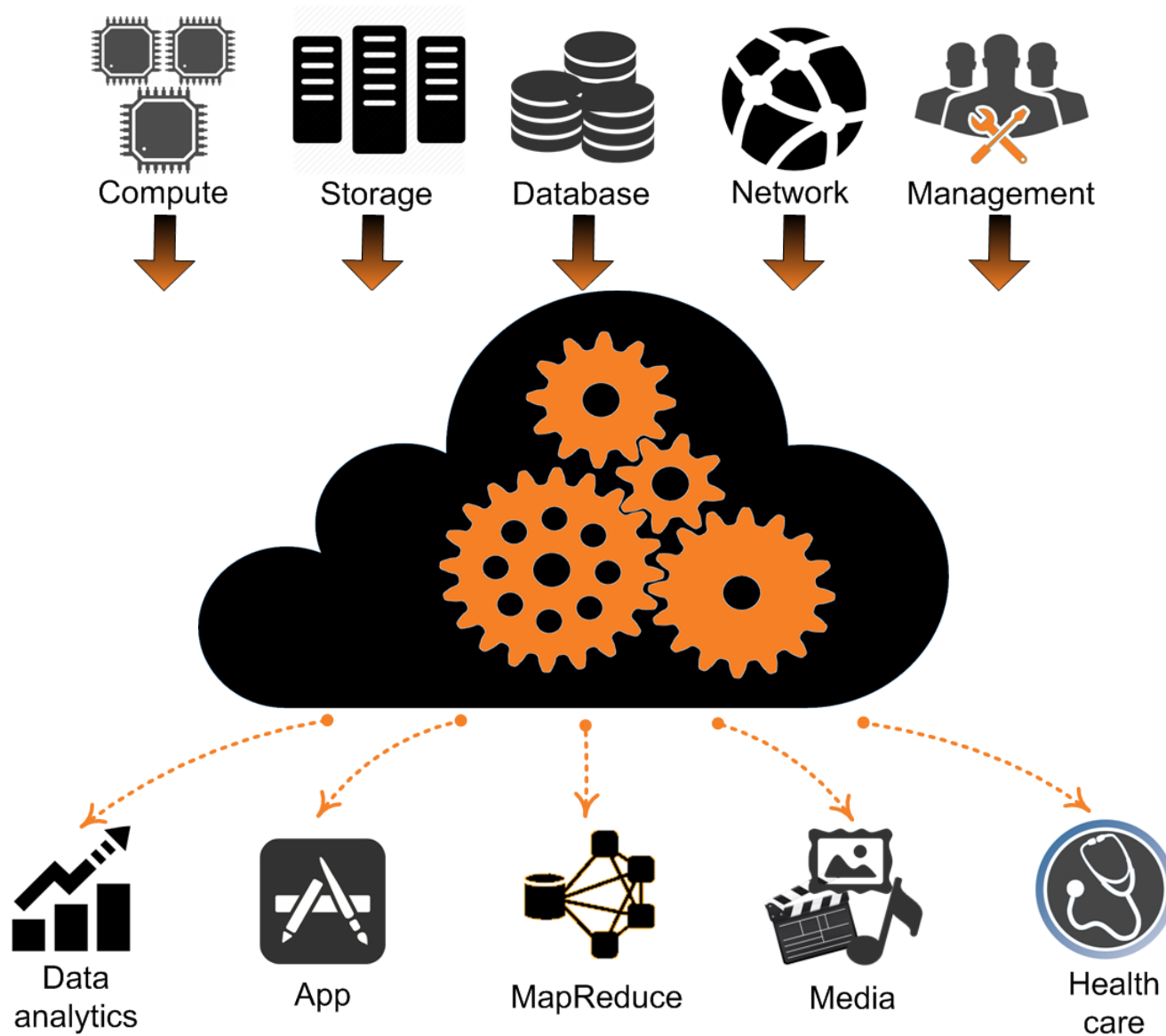


SCALING UP COMPUTATION: CLOUD ECONOMICS & AUTOMATION

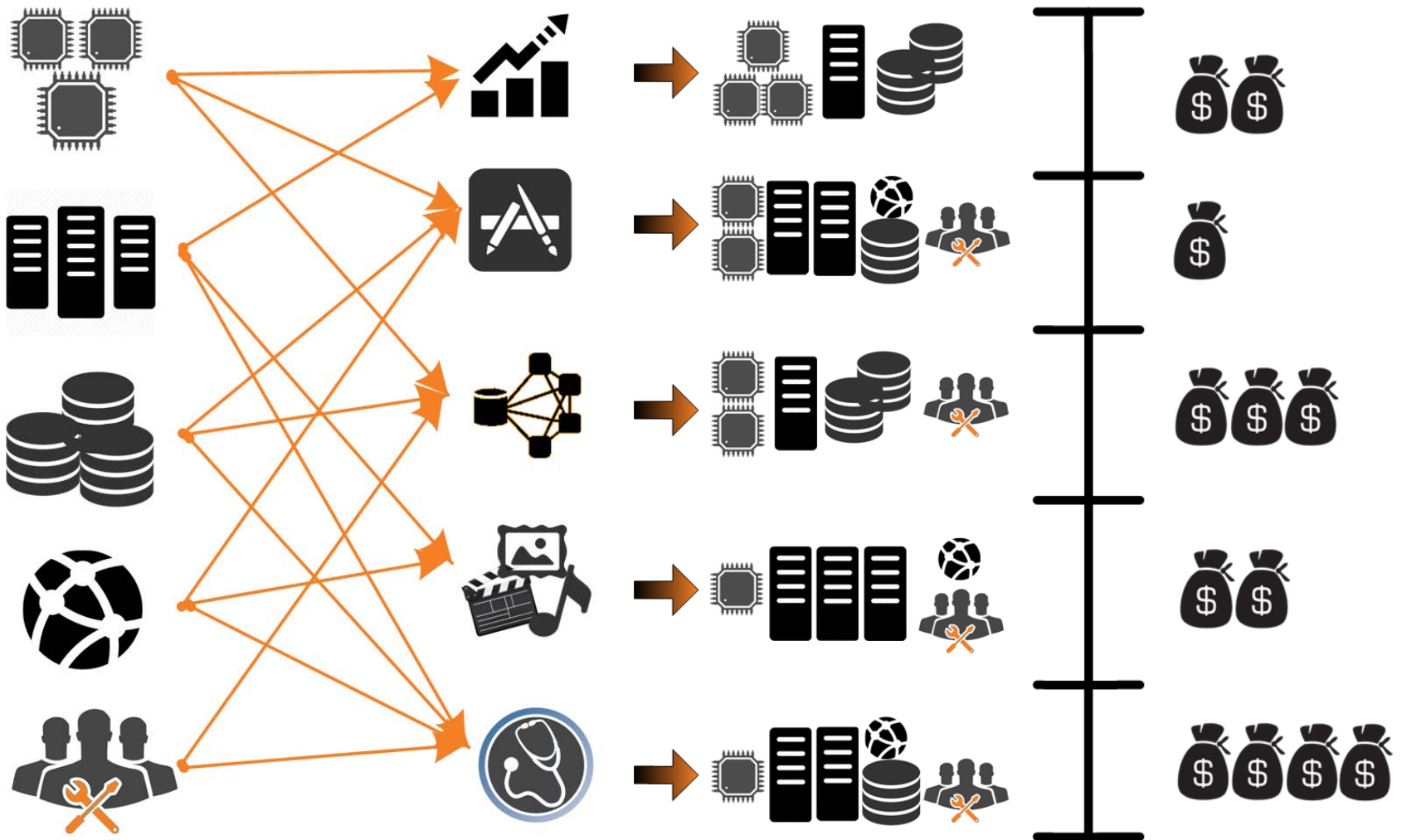
Chee Wei Tan



Cloud Computing



Cloud Resource Allocation and Pricing



Cloud Pricing

Usage-based cloud pricing






Auction-based cloud pricing



**Amazon's Elastic Compute Cloud (EC2)
spot instances**

Spot Instance

Step 1: Choose the instance type

 Amazon Linux Free tier eligible	<input type="checkbox"/>	Memory optimized	r3.large	2	15	1 x 32 (SSD)	-	Moderate
	<input type="checkbox"/>	Memory optimized	r3.xlarge	4	30.5	1 x 80 (SSD)	Yes	Moderate
 Red Hat Free tier eligible	<input type="checkbox"/>	Memory optimized	r3.2xlarge	8	61	1 x 160 (SSD)	Yes	High
	<input type="checkbox"/>	Memory optimized	r3.4xlarge	16	122	1 x 320 (SSD)	Yes	High
 SUSE Linux Free tier eligible	<input type="checkbox"/>	Memory optimized	r3.8xlarge	32	244	2 x 320 (SSD)	-	10 Gigabit

Step 2: Configure the instance details

– Number of instances & bid price

Number of instances ⓘ

Purchasing option ⓘ ☒ Request Spot Instances

Current price ⓘ

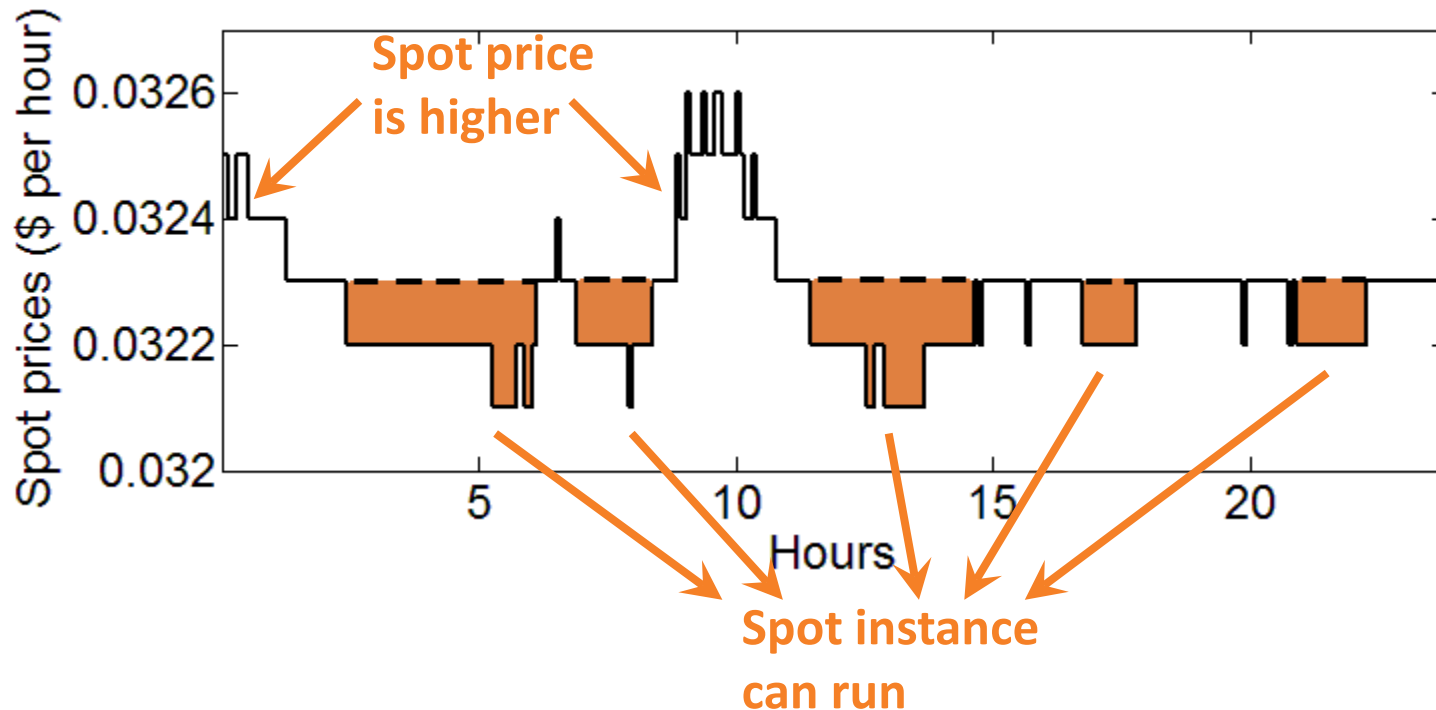
us-east-1a	0.1582
us-east-1b	0.1587
us-east-1d	0.1821
us-east-1e	0.1856

Maximum price ⓘ \$

**Name the price
YOURSELF!**

Spot Pricing

Spot price history for an r3.xlarge instance in the US Eastern region on September 09, 2014



Our Questions

- Question #1

How might the cloud provider set the price?

- Question #2

What prices should users bid?

Our Questions

- Question #1

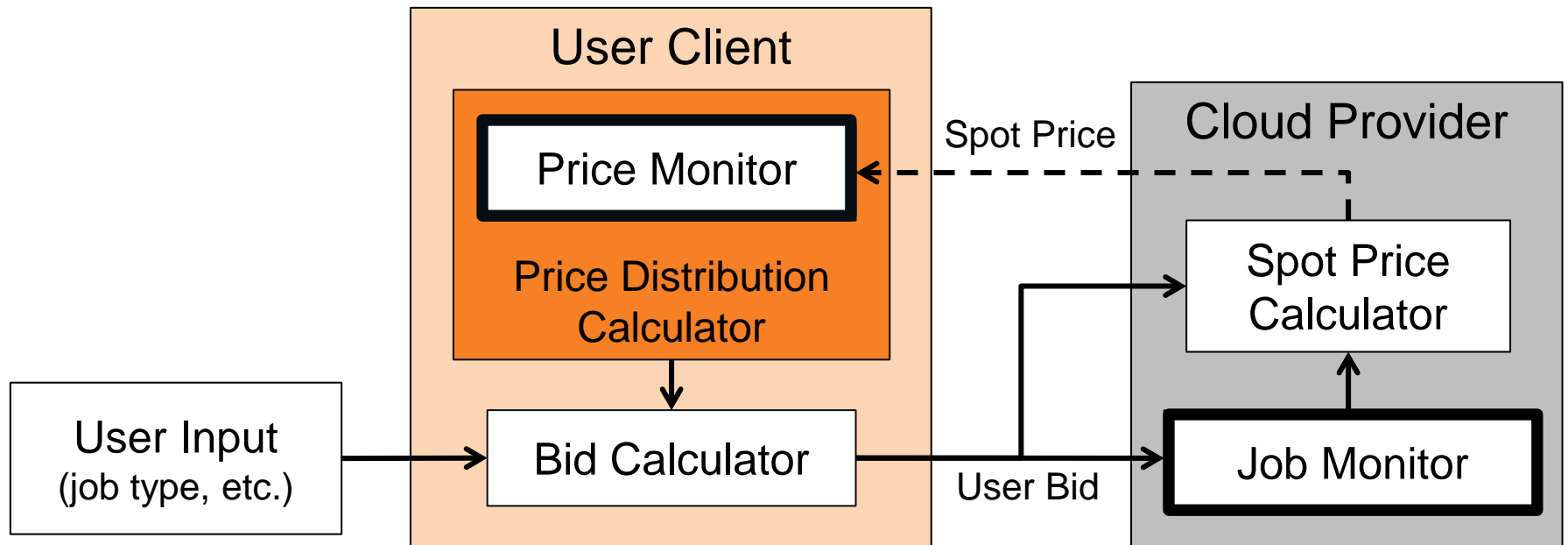
How might the cloud provider set the price?

- Question #2

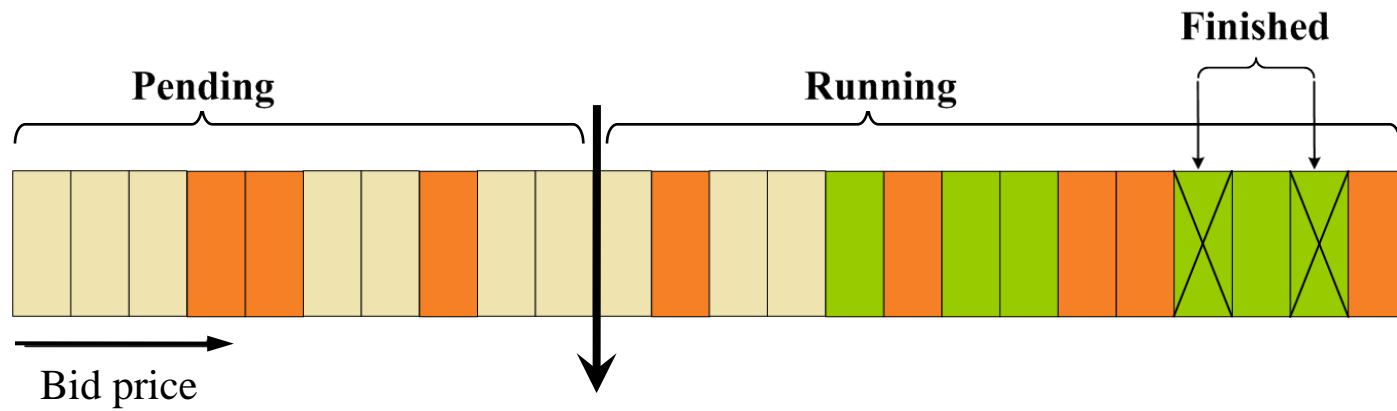
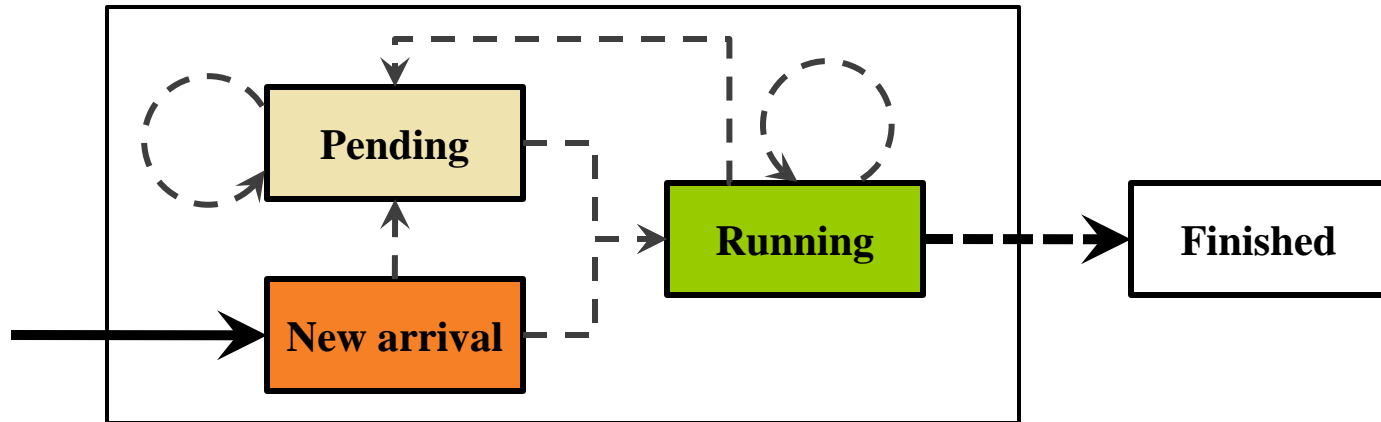
What prices should users bid?

L. Zheng, C. Joe-Wong, C. W. Tan, M. Chiang & X. Wang, How to bid the cloud? **ACM SIGCOMM**, 2015

Our Solution

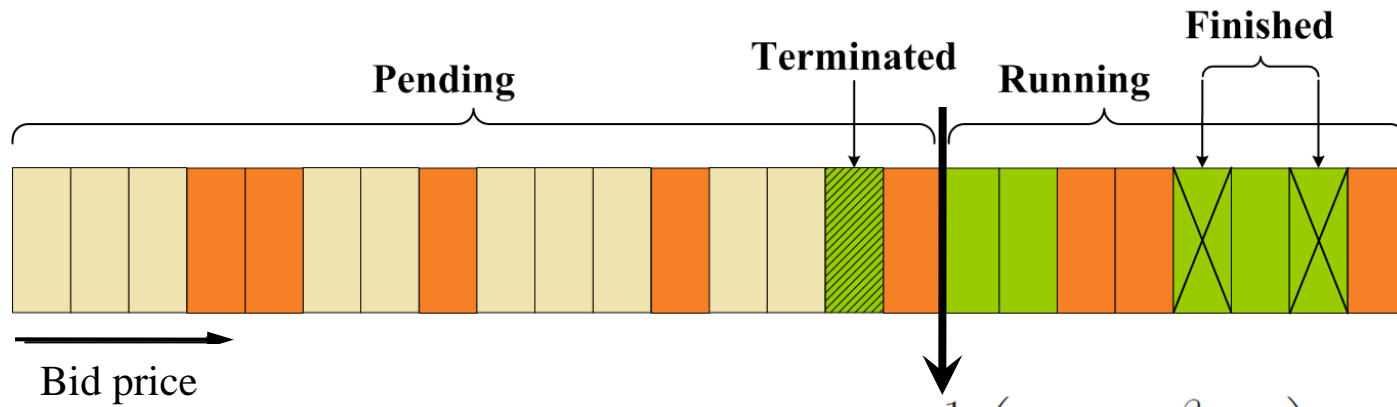
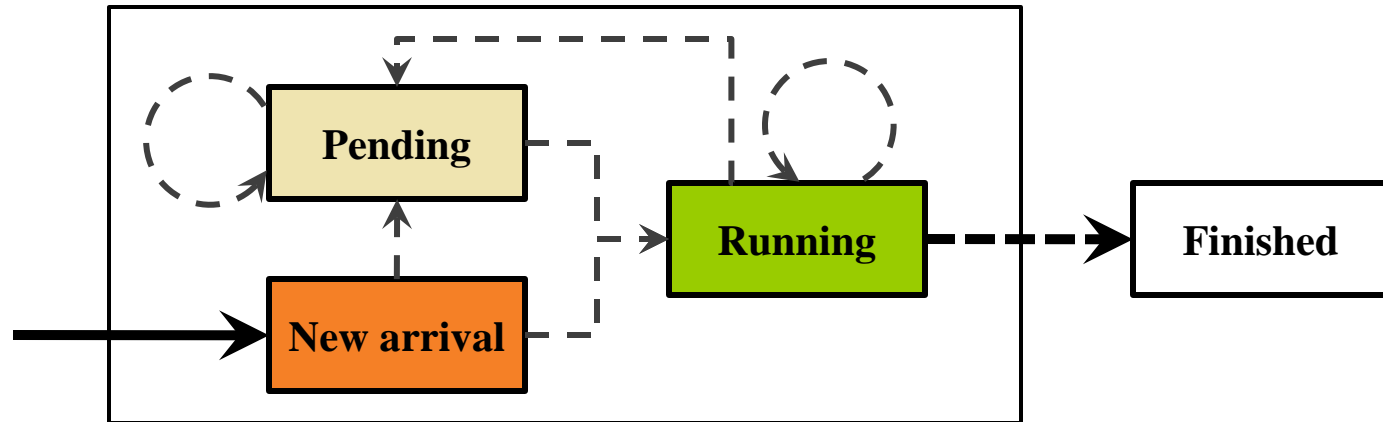


Cloud Provider Model



$$\pi^*(t) = h(\Lambda(t)) = \frac{1}{2} \left(\bar{\pi} - \frac{\beta}{1 + \frac{1}{\theta}\Lambda(t)} \right)$$

Cloud Provider Model



$$\pi^*(t) = h(\Lambda(t)) = \frac{1}{2} \left(\bar{\pi} - \frac{\beta}{1 + \frac{1}{\theta} \Lambda(t)} \right)$$

Bid Types

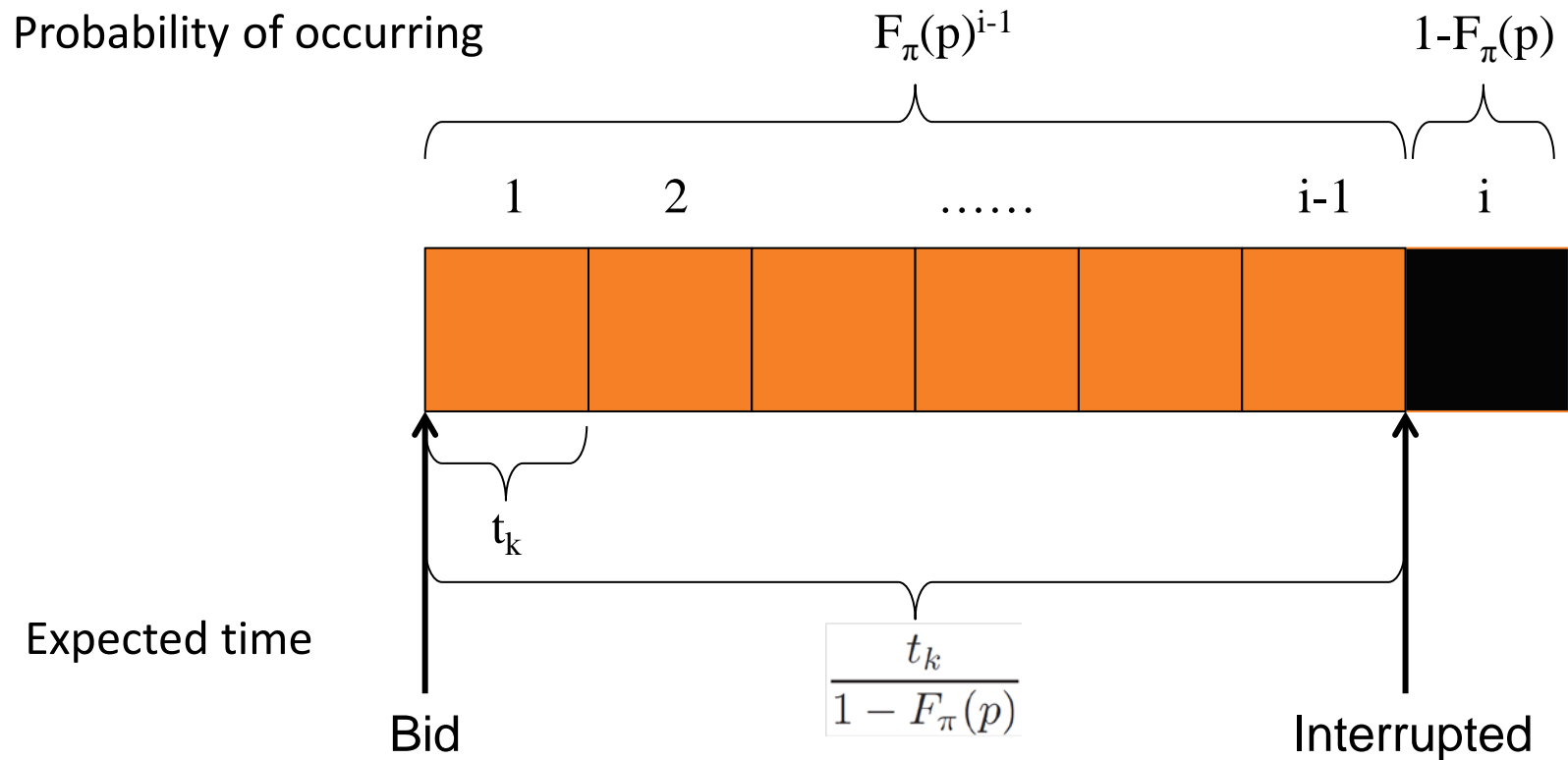
- One-time user bid

- Submitted once and then exit the system once they fall below the current spot price.
- Job interrupted without completing

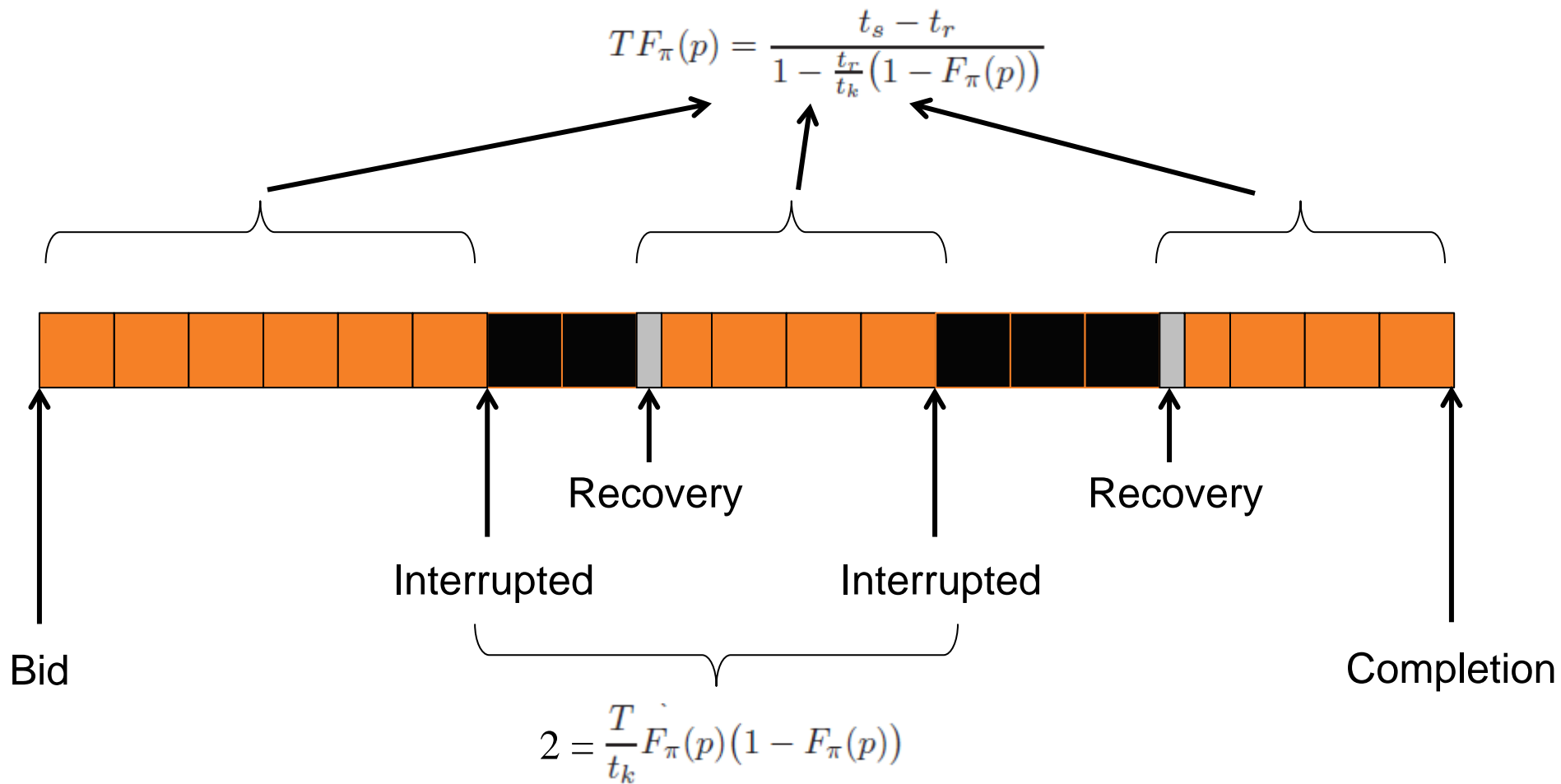
- Persistent user bid

- Resubmitted in each time period until the job finishes or is manually terminated by the user.
- Longer waiting and completion time.

Placing One-time Bids



Placing Persistent Bids



Bidding MapReduce Jobs

**Master
node**

$$\frac{t_k}{1 - F_{\pi}^m(p_m)}$$

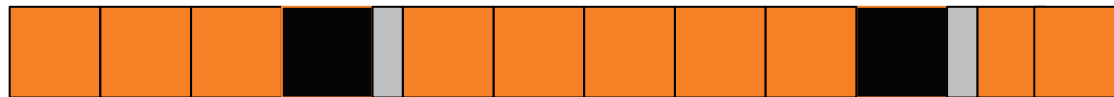


V

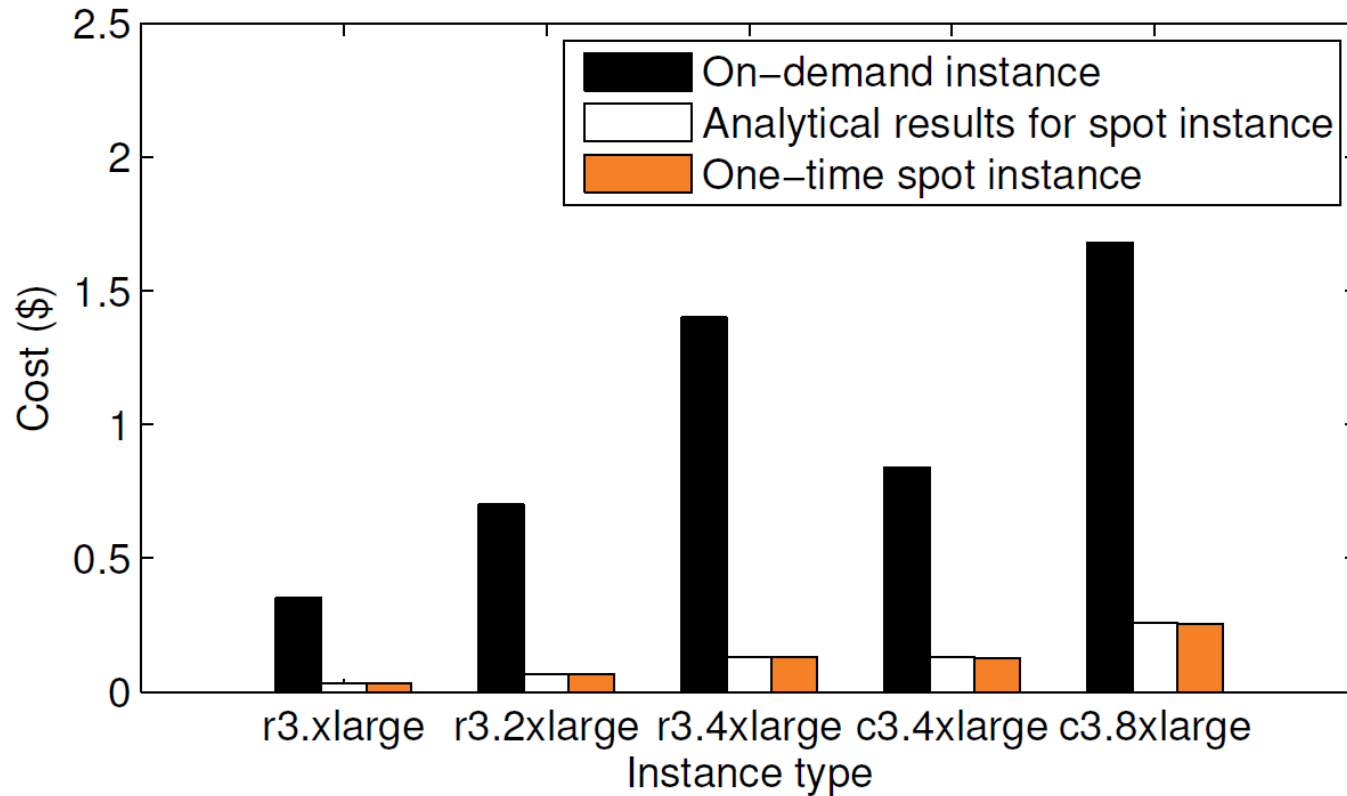
$$\frac{1}{F_{\pi}^v(p_v)} \left(\frac{t_s + t_o - Mt_r}{1 - \frac{t_r}{t_k} (1 - F_{\pi}^v(p_v))} - \frac{(M-1)t_k}{1 - F_{\pi}^v(p_v)} \right)$$



**Slave
nodes**



One-time Bids

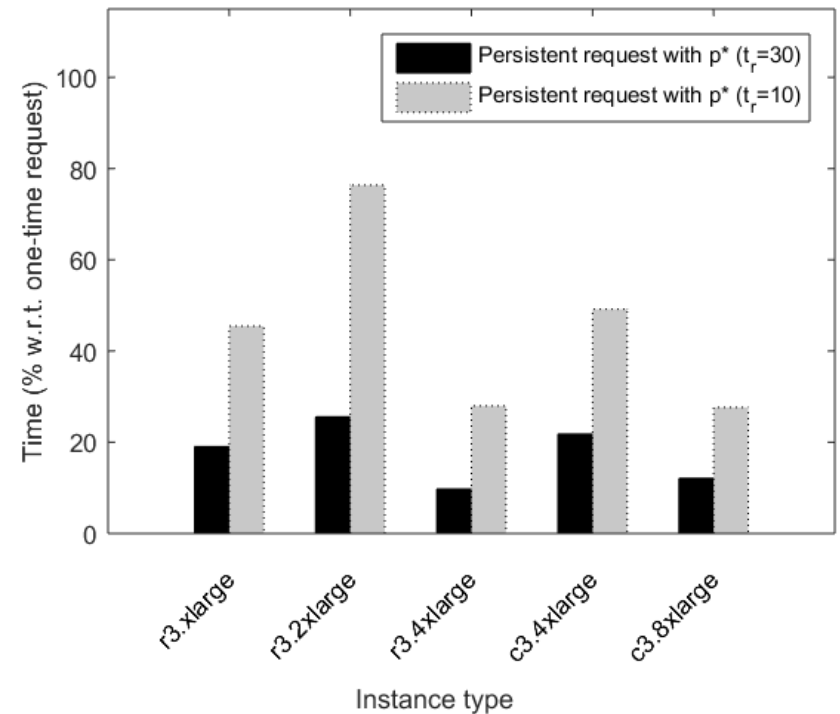
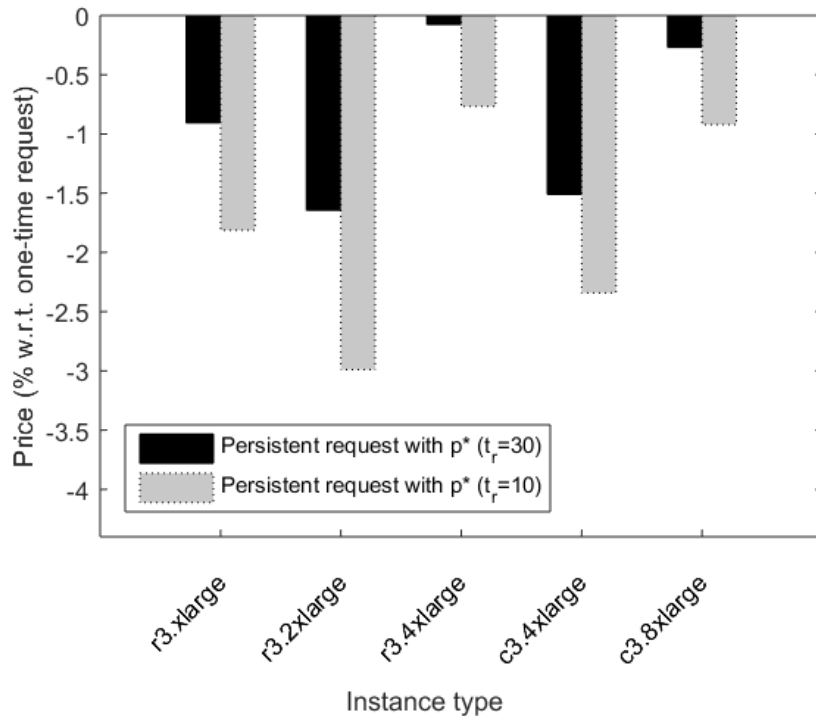


User costs are reduced by up to 91%, without any interruptions.

Persistent Bids

bid price (time) of persistent bids – bid price (time) of one-time bids

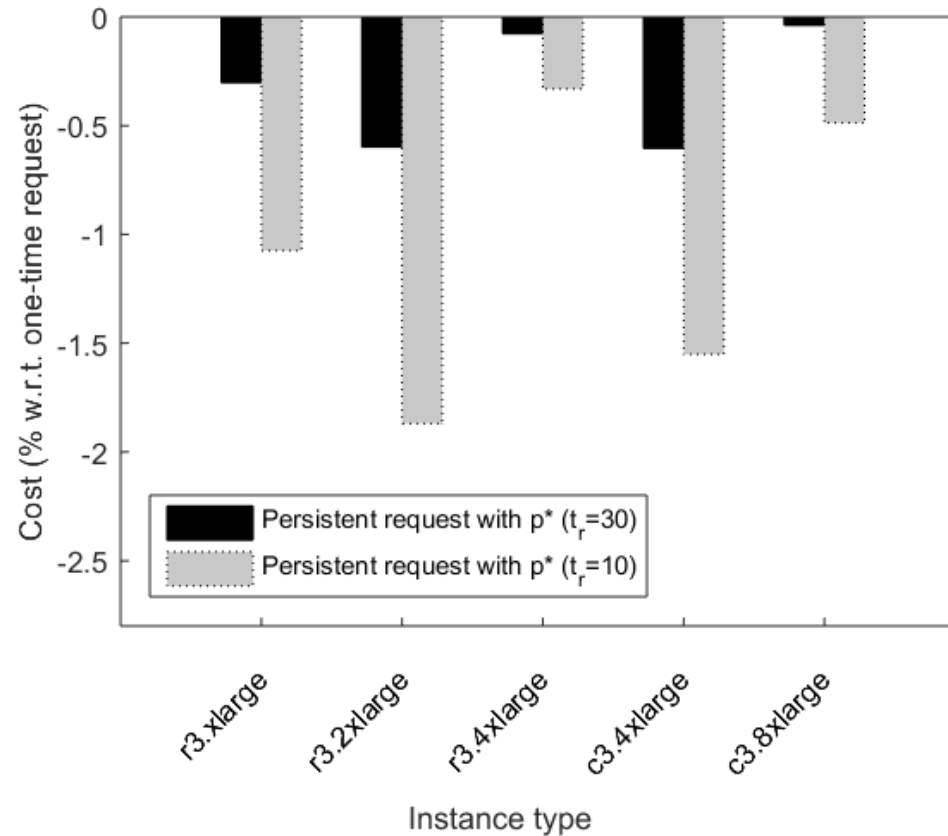
bid price (time) of one-time bids



A lower optimal bid price.

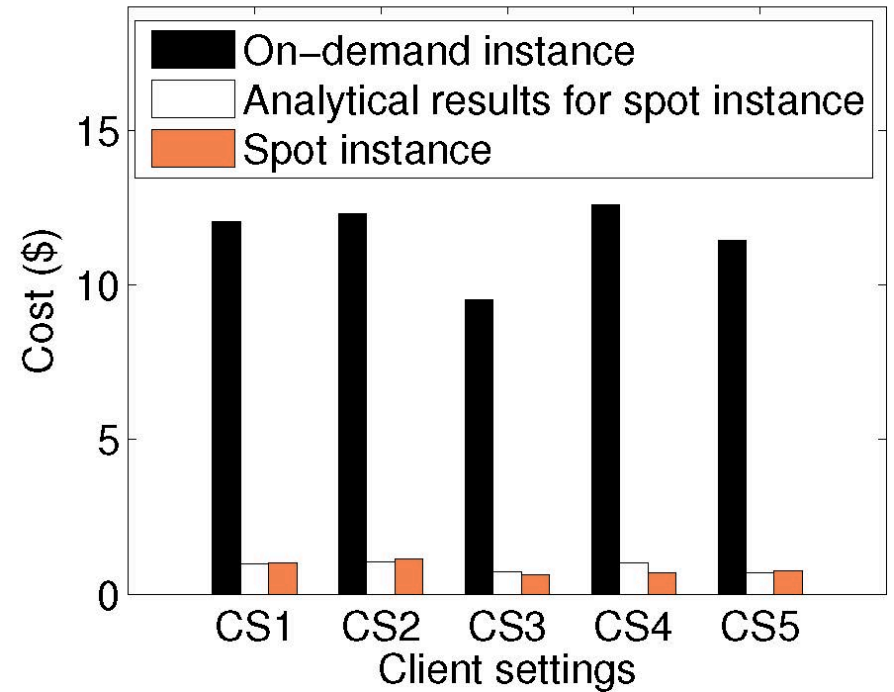
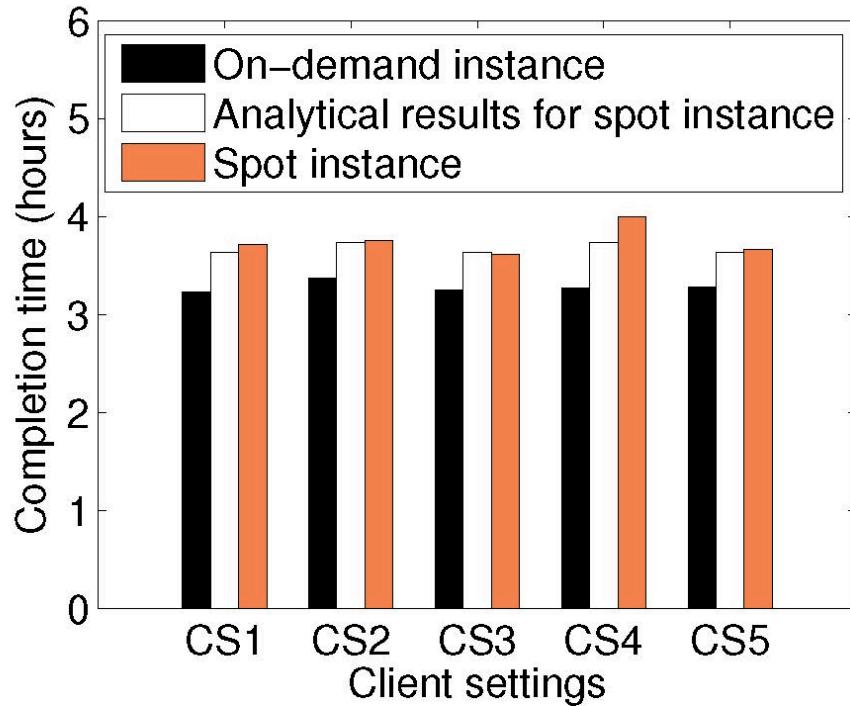
A longer completion time.

Persistent Bids



The overall costs are further reduced.

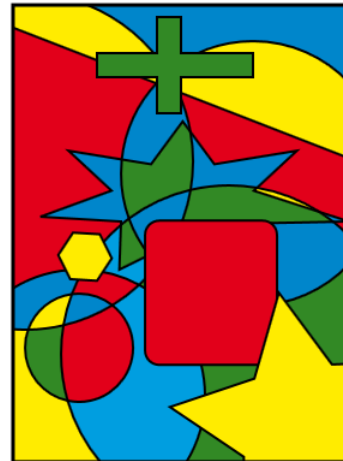
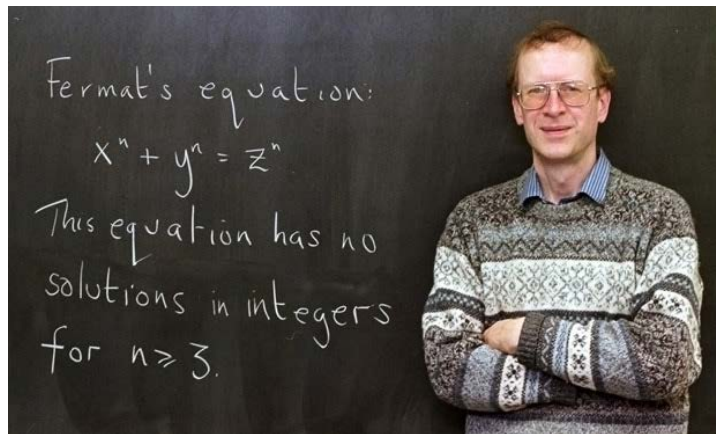
MapReduce Jobs



The cost is reduced by up to 92.6% with just a 14.9% completion time increase.

Scientific-Computing-As-A-Service

- Scientific computing is the holy grail of cloud computing
- Major recent breakthroughs in scientific discovery
 - Four Color Theorem
 - Fermat's Last Theorem



How to make bucks selling Theorems?!

Introduction

- Proving an information inequality is a crucial step in establishing the converse results in coding theorems.
- An information inequality involving many random variables is difficult to be proved manually.
- [Yeung 1997] developed a framework that uses linear programming for verifying linear information inequalities.
- Under this framework, our paper considers a few other problems that can be solved by using Lagrange duality and convex approximation.
- S. Ho, C. W. Tan and R. W. Yeung, Proving and disproving information inequalities, IEEE ISIT 2014



To Prove:

- ITIP("H(U) <= H(R)", "I(U;X) = 0"; "H(U|RX) = 0")
- True. The inequality follows from
- $$\begin{aligned} H(U) + H(R) &= (-H(U,X) + H(U,R,X)) + (H(R) + H(X) - H(R;X)) + \\ &\quad \{-H(U) - H(X) + H(U;X)\} + \{H(R;X) - H(U;R;X)\} \\ &\geq 0; \end{aligned}$$
- where (\cdot) is non-negative as it is either conditional entropy or conditional mutual information. All $\{\cdot\}$ are equal to 0 due to the given constraints. Equality holds iff all (\cdot) are equal to 0.

To Disprove:

- ITIP("I(A;B|CD) + I(B;D|AC) <= I(A;B|D) + I(B;D|A) + H(A) + I(B;D|C)")
- Not provable by ITIP.
- It can be disproved by a probability distribution satisfying all the following Shannon's information measures equal to zero:

$H(A|B,C,D); H(C|A,B,D), H(D|A,B,C), I(A;B|C), I(A;B|D), I(A;C|D), I(A;D), I(B;C|A), I(B;D|A), I(B;D|C), I(C;D|A).$

From the above output from ITIP, we can deduce the following counterexample. Let $X; Y$ and Z be three independent binary random variables with entropy equal to 1.

Let $(A, B, C, D) = (X \oplus Y, X, Y \oplus Z, Z).$

$I(A;B|D) + I(B;D|A) + H(A) + I(B;D|C) - I(A;B|CD) - I(B;D|CA)$

$= -1 < 0.$

Scale Up Computation

- Automation of Information Theory Prover
 - Computational algorithms (optimization based)
 - Cloud-computing (scale-up no. of random variables)
- Distributed data storage and security applications
 - Secured Storage Code
 - Privacy of data by information leakage to eavesdroppers during repair
 - Hybrid repair (gap with functional repair), when
 - Storage Network Topology Optimization
 - All prior work assumes a complete connectivity topology for storage network, however practical networks have different communication capacities and diverse (sparse) network topology
 - Information-theoretic Security and Network Security
 - Application-specific equality constraints in LP most interesting (Markov chain, security)

Scientific-Computing-As-A-Service



Welcome to ITTPc

Submit a new problem:

The first way: upload the file with the [problem format](#).

1. Please choose your file:

No file chosen

2. Upload the file to the cloud:

☐ Shortest Proof

The second way: input the problem in the box.

Please input the objective function:

$I(A; B \mid F) + H(A, B, C, D \mid F) - 2H(A) + I(A; B \mid F) - H(A) + I(B; C \mid F) - H(A) + I(A; D \mid F) - H(A) \leq 0$

Please input constraints:

$H(A, B, C, D) = 4H(A)$
 $H(A) = H(B)$
 $H(B) = H(C)$
 $H(C) = H(D)$

☐ Shortest Proof

Check previous results:

Input problem ID to obtain results for submitted problems.

Scientific-Computing-As-A-Service

https://cerg1.ugc.edu.hk/ x 130.211.240.38/html/resul x

← → ↻ 130.211.240.38/html/result.php

Result:

Go back to home page [Home](#)

Problem 1446550623 is solved as follows:

Time:

Started at:
2015-11-3 11:36:58
Finished at:
2015-11-3 11:36:58

Linear Program:

[Download](#)

Dual Linear Program:

[Download](#)

Solution:

[Download](#)

Disproof:

[Download](#)

Objective function: $I(A;B|F)+H(A,B,C,D|F)-2H(A)+I(A;B|F)-H(A)+I(B;C|F)-H(A)+I(A;D|F)-H(A)\leq 0$

Constraints:
 $H(A,B,C,D)=4H(A)$
 $H(A)=H(B)$
 $H(B)=H(C)$
 $H(C)=H(D)$

Not provable by ITIP.
It can be disproved by a probability distribution satisfying all the following Shannon's information measures equal to zero:
 $H(B,F,C,D)-H(A,B,F,C,D),$
 $H(A,F,C,D)-H(A,B,F,C,D),$
 $H(A,B,C,D)-H(A,B,F,C,D),$
 $H(A,B,F,D)-H(A,B,F,C,D),$

7:37 PM
11/3/2015

Scientific-Computing-As-A-Service

	IEEE TRANSACTIONS ON
INFORMATION THEORY	
A Journal Devoted to the Theoretical and Experimental Aspects of Information Transmission, Processing, and Utilization	
JULY 1999	VOLUME 45 NUMBER 5 IETTAW (ISSN 0018-9448)
PAPERS	
<i>U. Wachsmann, R. F. H. Fischer, and J. B. Huber</i>	Multilevel Codes: Theoretical Concepts and Practical Design Rules 1361
<i>A. Barg, E. Kruskal, and H. C. A. van Tilborg</i>	On the Complexity of Minimum Distance Decoding of Long Linear Codes 1392
<i>E. Fishler, O. Amrani, and Y. Be'ery</i>	Geometrical and Performance Analysis of GMD and Chase Decoding Algorithms 1406
<i>C. Rong, T. Helleseth, and J. Lahtonen</i>	On Algebraic Decoding of the \mathbb{Z}_4 -Linear Calderbank-McGuire Code 1423
<i>A. R. Calderbank, G. D. Forney, Jr., and A. Vardy</i>	Minimal Tail-Biting Trellises: The Golay Code and More 1435
<i>V. Tarokh, H. Jafarkhani, and A. R. Calderbank</i>	Space-Time Block Codes from Orthogonal Designs 1456
<i>G. Caire, G. Taricco, and E. Biglieri</i>	Optimum Power Control Over Fading Channels 1468
<i>H. Jafarkhani and V. Tarokh</i>	Design of Successively Refinable Trellis-Coded Quantizers 1490
<i>G. I. Shamir and N. Merhav</i>	Low-Complexity Sequential Lossless Coding for Piecewise-Stationary Memoryless Sources 1498
<i>T. Berger and R. Zamir</i>	A Semi-Continuous Version of the Berger-Young Problem 1520
<i>A. Kato and K. Zeger</i>	On the Capacity of Two-Dimensional Run-Length Constrained Channels 1527
<i>G. Ritter</i>	Efficient Estimation of Neural Weights by Polynomial Approximation 1541
<i>M. A. Gibson and J. Bruck</i>	Efficient Digital-to-Analog Encoding 1551
<i>C. Herley and P. W. Wong</i>	Minimum Rate Sampling and Reconstruction of Signals with Arbitrary Frequency Support 1555
<i>G. Le Breuer, J.-F. Bercher, and G. Demoment</i>	A New Look at Entropy for Solving Linear Inverse Problems 1565
<i>W. Luo and A. Ephremides</i>	Stability of N Interacting Queues in Random-Access Systems 1579
<i>A. Yan and W.-B. Gong</i>	Time-Driven Fluid Simulation for High-Speed Networks 1588

Conclusion

- Model for cloud provider's setting of the spot prices.
- Bidding strategies: Tradeoff between prices and times
 - One-time bids: bidding higher prices to avoid interruptions .
 - Persistent bids: bidding lower prices to save money.
- Application to the MapReduce jobs.
- Temporal correlations, risk-awareness, task dependence, collective user behavior, etc.
- Scaling up Scientific Computing as a Service

Thank you!

- L. Zheng, C. Joe-Wong, C. W. Tan, M. Chiang & X. Wang, How to bid the cloud? **ACM SIGCOMM**, 2015
- S. Ho, C. W. Tan and R. W. Yeung, Proving and disproving information inequalities, IEEE ISIT 2014

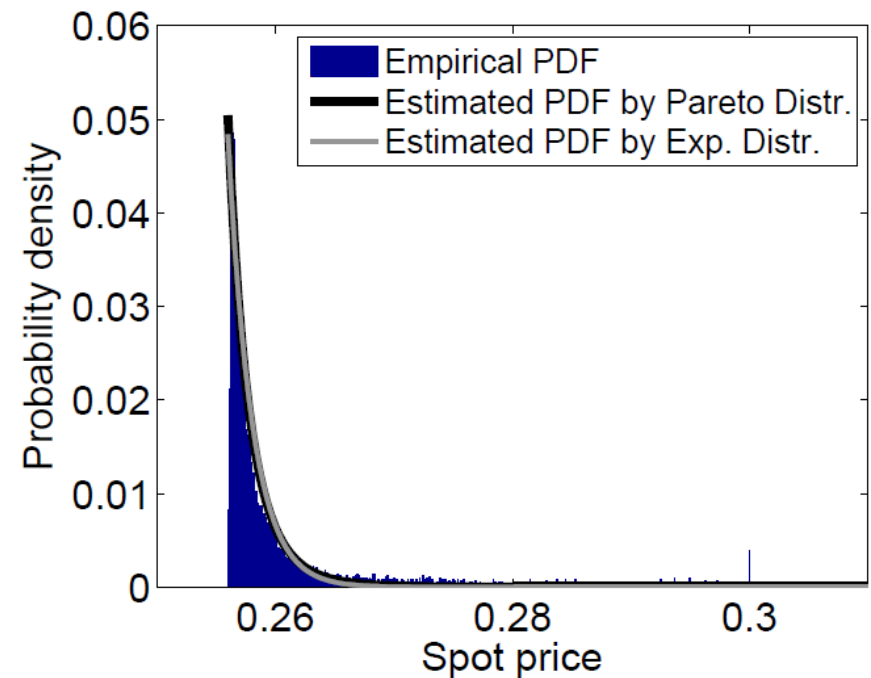
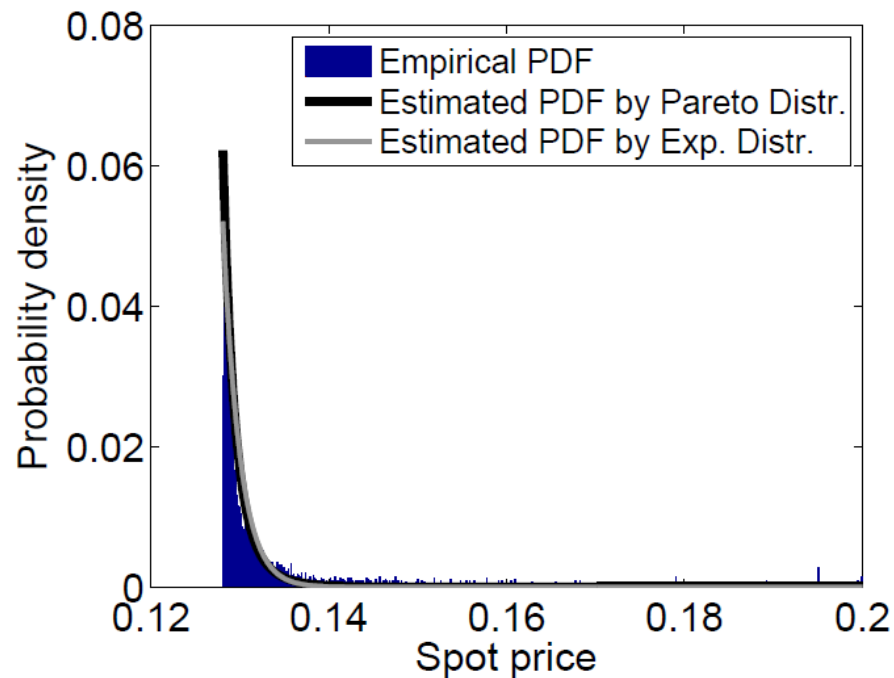
Backup Slides

Cloud provider revenue maximization.

$$\begin{aligned} & \underset{\pi(t)}{\text{maximize}} && \beta \log \left(1 + L(t) \frac{\bar{\pi} - \pi(t)}{\bar{\pi} - \underline{\pi}} \right) \\ & && + \pi(t) L(t) \frac{\bar{\pi} - \pi(t)}{\bar{\pi} - \underline{\pi}} \\ & \text{subject to} && \underline{\pi} \leq \pi(t) \leq \bar{\pi}. \end{aligned}$$

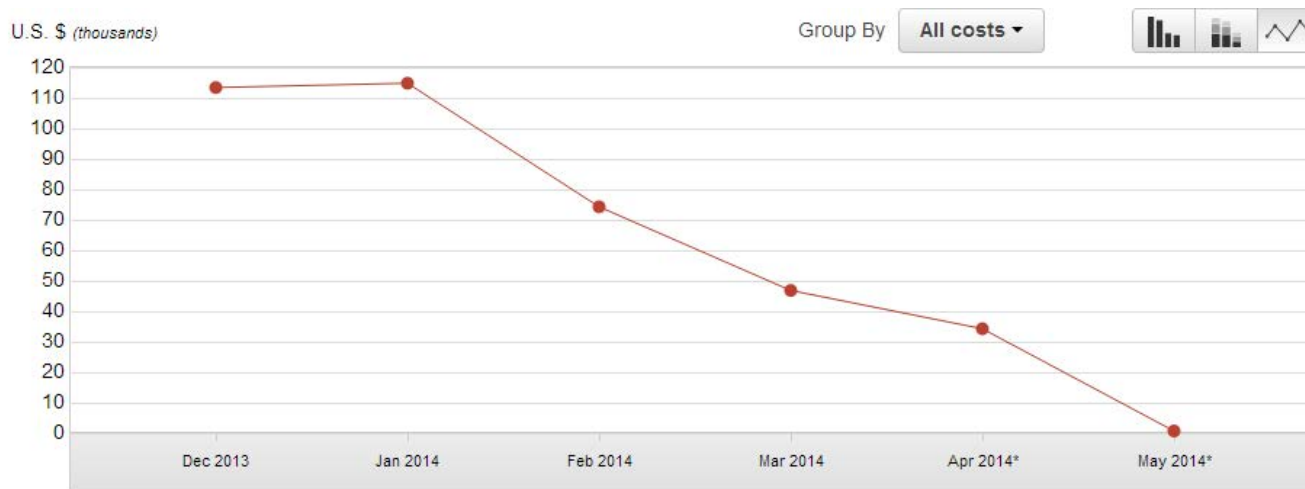
$$\Rightarrow \pi^*(t) = h(\Lambda(t)) = \frac{1}{2} \left(\bar{\pi} - \frac{\beta}{1 + \frac{1}{\theta} \Lambda(t)} \right)$$

Validation from Historical Spot Prices



Real-life Spot Instance Example

Mozilla Amazon EC2 usage got cheaper by using spot instance with fixed-price bidding in 2013-2014.



September: cost-effective m3.xlarge on-demand instances.

October: m3.xlarge spot instances with unexpected interruptions.

December: upgrading to use cheaper c3.xlarge on-demand instances.

February: a mix of c3.xlarge on-demand and m3.xlarge spot instances.

March: the majority of workload switched to spot instances.

April: Amazon further drops its spot prices.

Source: <http://taras.glek.net/blog/2014/05/09/how-amazon-ec2-got-15x-cheaper-in-6-months/>

EC2 Instance Types

Balanced



Memory-optimized



Compute-optimized



	m3			r3			c3		
	vCPU	Memory	Storage	vCPU	Memory	Storage	vCPU	Memory	Storage
.xlarge	4	15	1x32	4	30.5	1x80	4	7.5	2x40
.2xlarge	8	30	2x80	8	61	1x160	8	15	2x80
.4xlarge	----	----	----	16	122	1x320	16	30	2x160
.8xlarge	----	----	----	----	----	----	32	60	2x320

Single-Instance One-time Bids

Optimal bid prices for one-time bids that run for one hour.

Instance type	On-demand price	One-time bid	
		Optimal price	Actual price
r3.xlarge	\$0.35	\$0.0374	\$0.033
r3.2xlarge	\$0.70	\$0.0795	\$0.066
r3.4xlarge	\$1.40	\$0.1430	\$0.130
c3.4xlarge	\$0.84	\$0.1669	\$0.128
c3.8xlarge	\$1.68	\$0.2903	\$0.256

Single-Instance One-time Bids

Optimal bid prices for one-time bids that run for one hour.

Instance type	On-demand price	One-time bid		
		Optimal price	Offline retrospective price	Actual price
r3.xlarge	\$0.35	\$0.0374	\$0.0324	\$0.033
r3.2xlarge	\$0.70	\$0.0795	\$0.0644	\$0.066
r3.4xlarge	\$1.40	\$0.1430	\$0.1304	\$0.130
c3.4xlarge	\$0.84	\$0.1669	\$0.1324	\$0.128
c3.8xlarge	\$1.68	\$0.2903	\$0.2640	\$0.256

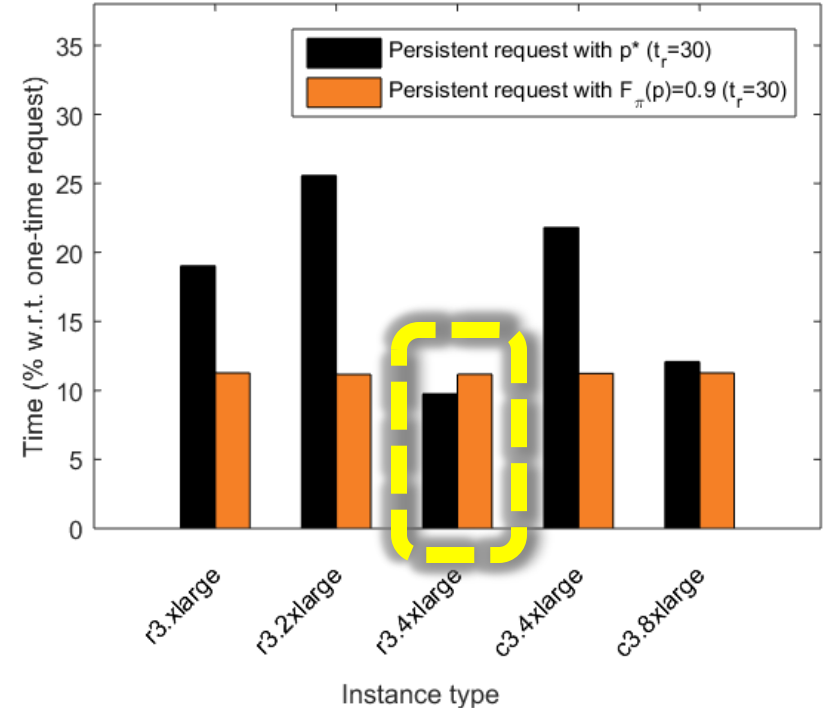
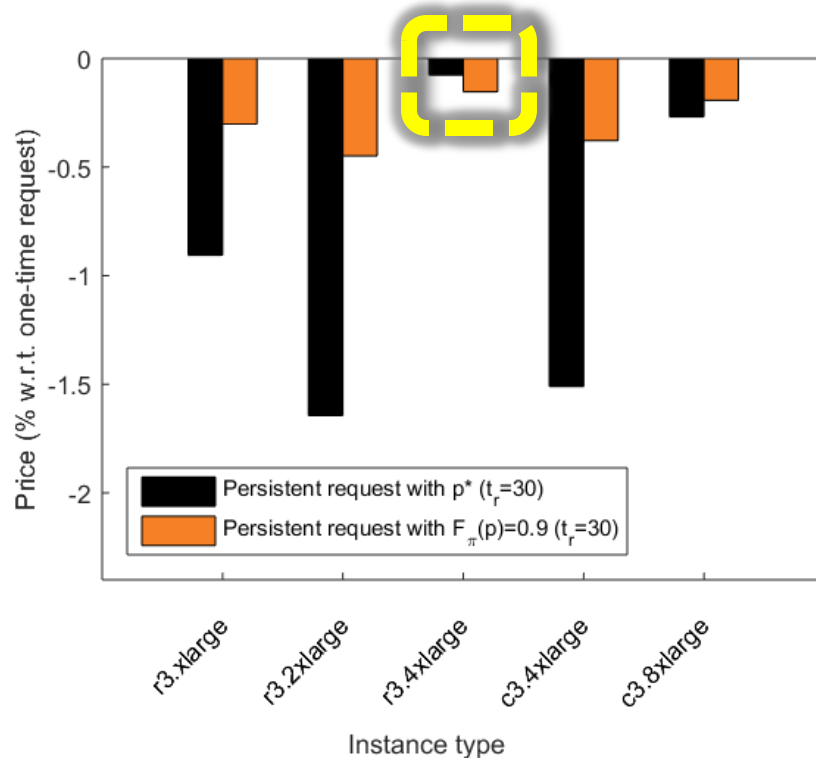
Single-Instance Persistent Bids

Optimal bid prices with different recovery times.

Instance type	On-demand price	Persistent bid	
		Optimal price ($t_r=10s$)	Optimal price ($t_r=30s$)
r3.xlarge	\$0.35	\$0.0332	\$0.0355
r3.2xlarge	\$0.70	\$0.0661	\$0.0711
r3.4xlarge	\$1.40	\$0.1327	\$0.1422
c3.4xlarge	\$0.84	\$0.1322	\$0.1413
c3.8xlarge	\$1.68	\$0.2648	\$0.2831

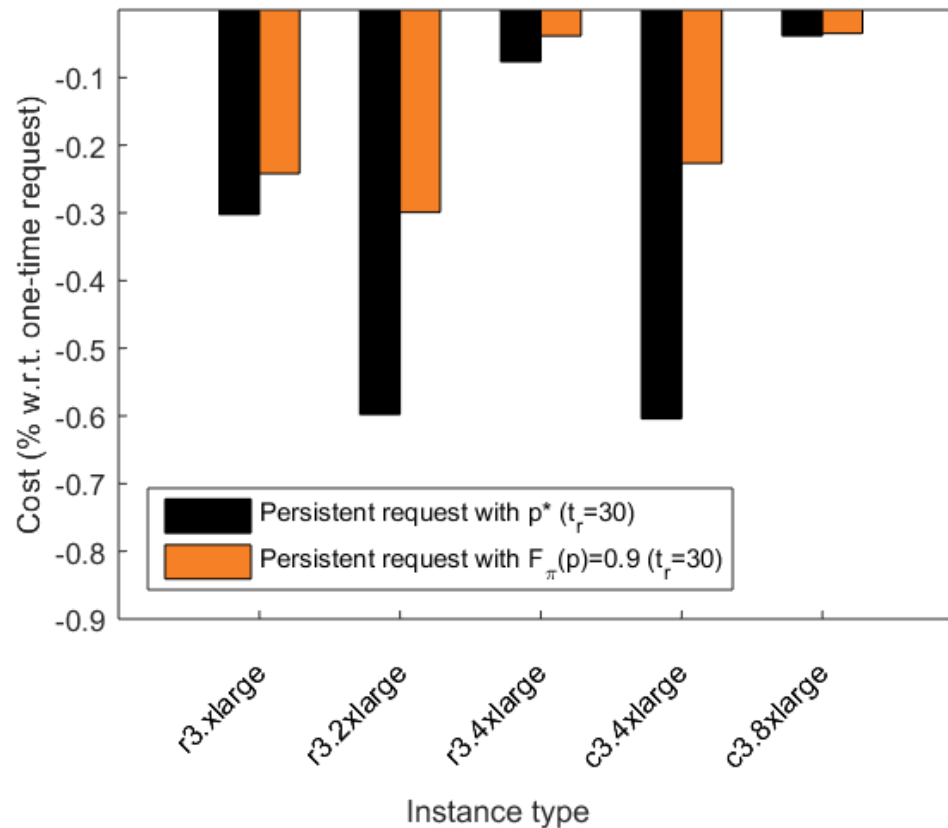
Longer recovery times yield higher bid prices.

Single-Instance Persistent Bids



Bidding at the 90th percentile price yields either higher bid prices and lower completion times or lower bid prices and longer completion times.

Single-Instance Persistent Bids



Our bid prices are optimal for minimizing users' costs.

MapReduce Jobs

Optimal bid prices and actual costs for a MapReduce job.

Client Setting	Master node			Slave nodes			
	Instance Type	Bid prices	Actual cost	Instance type	Bid prices	Node numbers	Actual cost
CS1	c3.xlarge	\$0.133	\$0.10	m3.2xlarge	\$0.070	5	\$0.90
CS2	m3.xlarge	\$0.101	\$0.13	m3.2xlarge	\$0.070	5	\$1.03
CS3	m3.xlarge	\$0.102	\$0.13	r3.2xlarge	\$0.071	3	\$0.51
CS4	r3.xlarge	\$0.042	\$0.13	m3.2xlarge	\$0.070	5	\$0.58
CS4	r3.xlarge	\$0.042	\$0.13	r3.2xlarge	\$0.071	3	\$0.64