Cloudy with a Spot of Opportunity: Analysis of Spot-Priced VMs for Practical Job Scheduling

Vedsar Kushwaha and Yogesh Simmhan
Supercomputer Education and Research Centre
Indian Institute of Science, Bangalore, India 560012
Email: vedsarkushwaha@ssl.serc.iisc.in, simmhan@serc.iisc.in

***Abstract*—Public Clouds offer elastic computing resources on-demand using a pay as you go model. While this has opened up access to computing infrastructure, the costs for accessing Cloud resources can be a barrier to adoption in emerging markets. Spot-priced virtual machines (VMs) are offered at deep discounts for the same compute capability as fixed price on-demand VMs. But they can be reclaimed by the Cloud provider at any time, affecting reliability. This paper characterises the behaviour of spot-priced VM from Amazon Web Service for the Asia-Pacific and US East Regions, and analyses their practical impact on running jobs on spot VMs. Our simulation study using jobs of diverse sizes evaluates the trade-offs between cost savings over fixed price VMs and job resilience. Our results show that in most cases, for the workloads studied, we can achieve an effective bottom-line cost savings of 80% using spot VMs, with over 95% reliability.**

I. Introduction

Cloud computing has made utility computing a reality, offering on-demand computing, storage, platform services and Software as a Service (SaaS), using a pay as you go model. Besides helping enterprises outsource their compute infrastructure, this has also democratised access to computational resources – startups can ramp up their compute usage without up-front cost, and scientists in the long tail of computing can periodically access high end resources, elastically [1]. Businesses in emerging markets have also benefited from the lower total cost of ownership (TCO) and ease of accessibility of globally distributed Cloud infrastructure and services [2].

At the same time, the pricing model of Clouds, specifically of Infrastructure as a Service (IaaS) providers, does not offer any discounts to emerging markets compared to their counterparts in developed nations. In fact, customers using virtual machines (VMs) in data centres present in emerging regions end up paying a higher price for such VMs compared to identical VMs in US or European data centres. For e.g., an m3.large VM with 2 virtual CPUs from Amazon Web Services (AWS) [1] costs US$0.140 in the US East Coast Region and US$0.154 in the European Union Region, while the same VM costs US$0.190 from an Amazon data centre in South America, and US$0.196 from their data centre in Asia-Pacific (Singapore). Such a price difference is due to factors such as cost of electricity, infrastructure, personnel, and taxation. Given that customers in emerging markets may prefer geographically proximate data centres, to ensure lower latency and also for compliance with local privacy laws, the premium pricing in emerging nations is a deterrence to Cloud adoption.

Cloud service providers have tried to minimise the operational cost of their unsold Cloud capacity through spot markets. AWS, which introduced spot VM instances in December 2009, remains the most popular of these spot market Cloud providers [3] though others exist [2]. Spot VMs work in a pseudo-auction model. They are often offered at a deep discount compared to fixed price on-demand VMs, despite offering equivalent performance and being available worldwide. As a result, they can significantly reduce the cost of using Cloud resources in emerging markets. However, with the reduction in price comes an additional risk, one of reliability. Spot Cloud providers like Amazon can reclaim spot instances at any time from the users without warning, causing them to lose unsaved data in the VM and disrupting service availability; nominally, the last partially used spot VM hour is not billed. This reclamation action is a function of the spot price set by Amazon and the price at which the user bids for the VM.

Due to this perceived lack of reliability, and also limited literature on spot VMs, spot markets have not gained the kind of traction that fixed price on-demand VMs have. There is some literature on modeling AWS Spot Prices [4], [5], [6] and even using them for Hadoop and SaaS applications [5], [7]. But none take a practical look at characterising the pricing behaviour of spot VMs, and its impact on reliably running jobs. Our earlier work has studied optimal scheduling of jobs on spot VMs to meet deadlines [8]. As we show in this current paper, using Amazon's spot VMs offers highly favourable trade-offs between cost and reliability, with limited need for sophisticated scheduling algorithms or price models. This makes them an attractive opportunity for emerging markets to leverage.

We make the following contributions in this paper: (1) We provide a cost analysis of Amazon's Spot VMs (§ III), specifically comparing the Asia-Pacific (Singapore) region against the US East Coast region, and also performing a study across time, with data from both 2014 and 2012. (2) Further, we perform a simulation study, using real spot price data, on running jobs of diverse compute requirements on such spot VMs, and analyse their savings–reliability trade-offs using metrics we propose (§ IV). Such a characterisation of spot pricing for jobs helps users in SMEs and emerging markets to effectively leverage its full potential.

II. Related Work

Significant research has gone into optimising the cost of scheduling applications on public Clouds [9], [1], [10]. In [11], the authors attempt to automate the match-making between the

[1] Amazon EC2 Pricing, http://aws.amazon.com/ec2/pricing/

[2] ComputeNext Cloud Brokerage https://www.computenext.com/

compute requirements of a job and Cloud resources. They use the availability of different VM sizes, and the elasticity offered in acquiring and releasing resources, to make scheduling decisions that are cost-efficient and meet a job's soft deadline. They use a generalisable job abstraction using workflows, and model Cloud system behaviour such as wait time for acquiring VMs. Researchers have also considered streaming jobs and modeled performance variation on Clouds due to multi-tenancy [12]. Jobs using a mix of on-demand VMs and reserved VMs have also been investigated [13], the latter costing marginally lesser than on-demand VMs but acquired in bulk for extended periods of time. The goal there is to decide the optimal number of reserved and on-demand instances that should be provisioned to minimise cost while satisfying a workload's response time tolerance. Despite such studies, these are applicable only to on-demand VMs that can be retained by the user as long as they pay the fixed per-hour price, and does not consider the uncertainity of spot-priced VMs, whose dynamic price modeling by the Cloud provider is often opaque. Our work is directed at understanding the easy use of spot VM instances for running jobs, and using simulations to bound the job's reliability while helping reduce the cost of running the job.

Researchers have examined Amazon's spot prices [3] with the goal of developing prediction models. An early work [14] attempts to reverse-engineer the pricing algorithm using data from US West region in 2009-2010. They posit that Amazon's spot price does not follow a market-based auction-model predicated on supply and client's demand, but is based on a constantly changing internal reserve price. Within periods, or *epochs*, the reserve price is constant but it varies across epochs. Other research has gone into using Markov Chain models to capture spot price variations [4], [5]. [6] examines cloud computing pricing dynamics across Amazon EC2 regions to discern the opportunity for arbitrage, and test for the influence of latency as a pricing wedge in the observed pricing dynamics. A detailed statistical analysis of spot prices is provided by [15], and it proposes a Gaussian mixture model to capture the spot pricing. They also considers the Asia-Pacific data centre, besides US and EU ones. Our work is in a similar vein, in trying to understand the spot price behaviour. However, rather than reflexively trying to predict the changes in pricing or accurately model it, we go on to examine how even a limited understanding of the pricing can translate into effectively running jobs on spot VMs. Our job simulation study estimates the practical impact of spot pricing on a job's reliability, and the savings from using spot over on-demand VMs.

There has been work on developing frameworks and job scheduling algorithms for spot-priced VMs. [16] uses an economics-based approach to develop scheduling policies when there is resource uncertainty. They investigate profit-aware job admission control and scheduling over resources that have an uncertainity in their availability and pricing in the future. [7] examines SaaS running on IaaS spot instances, and how the SaaS provider can charge its customer for executing its services and paying them a penalty for failing to meet service level agreements. They proposes a spot VM bidding scheme and VM allocation policy designed to optimise the average revenue earned per time unit. They offer both complex and simple heuristics for the scheduling, and offer simulation results based on AWS spot price data from the US East region. Our own prior work [8] uses check-pointing and migration strategies to increase the reliability of jobs running on spot VMs, using existing price prediction models. Others [5] have also leveraged the robustness of the Hadoop MapReduce framework to mitigate the impact of running on less reliable, but cheaper, spot instances. These research involve non-trivial job analysis, spot prediction models and scheduling strategies to run on spot instances, many of which are not translatable to practice. In this paper, we instead examine when "good enough" is enough, and show that even simple price analysis can offer insight on scheduling approaches on spot VMs, and provide adequate cost-benefit trade-offs to many applications.

At a more abstract level, research has explored how Cloud pricing models and markets can be developed and used by service IaaS providers and Cloud brokerages [17]. These are intended to help improve resource utilisation, reduce VM pricing, enhance quality of service for customers, and maximise the profit for Cloud service providers [18]. While such literature offers formal models for Cloud vendors, there is little evidence to show that sophisticated pricing and market-based models have been adopted in public Clouds at large scale, partly because the system design and assumptions imposed by such research may not hold in reality, and also because Cloud providers and users often prefer simplicity in practice.

III. Analysis of Spot Pricing

A. Dataset and Virtual Machine Description

Amazon Web Service (AWS) is the largest provider of spot VMs on public Clouds globally, in addition to on-demand VMs offered as part of their IaaS. For our analysis, we consider four different VM types that Amazon recommends for general purpose computing: *m1.small, m1.medium, m1.large,* and *m1.xlarge* [3] [4]. We abbreviate these as **Small, Medium, Large** and **XLarge**, respectively, in the rest of the paper. We also consider two different AWS regions (or data centres), one in Singapore for Asia-Pacific (*ap-southeast-1a*) and the other in Northern Virginia for US East Coast (*us-east-1a*) [5]. We abbreviate these as **AP-SE** and **US-E**, respectively. AP-SE is important for geo-location with Asian markets. All four VM sizes are available in both these regions, with identical performance but different on-demand prices (Table I)[6].

TABLE I. Performance and on-Demand Price for VM Types

VM Type	Performance				†On-Demand Price [US Cents]	
	Virtual CPUs (vCPU)	Elastic Compute Units (ECU)	Memory [GiB]	Storage [GB]	AP-SE	US-E
m1.small	1	1	1.7	1 × 160	5.8	4.4
m1.medium	1	2	3.75	1 × 410	11.7	8.7
m1.large	2	4	7.5	2 × 410	23.3	17.5
m1.xlarge	4	8	15	4 × 410	46.7	35.0

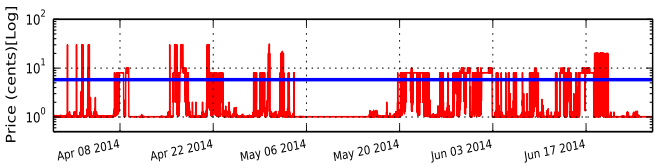
† *As on July 15, 2014*

[3] Amazon EC2 Instances, http://aws.amazon.com/ec2/instance-types/

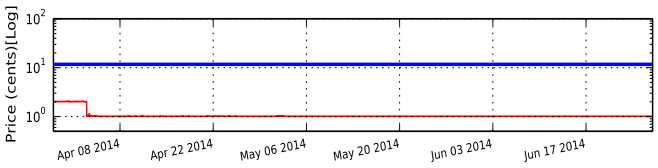
[4] AWS recently introduced *m3.** instance types as an upgrade from *m1.**. We use *m1.** in this paper to allow analysis across time: 2012 and 2014. We expect *m3.** to have similar pricing behaviour, and our results to carry forward.

[5] AWS Regions and Availability Zones, http://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-regions-availability-zones.html

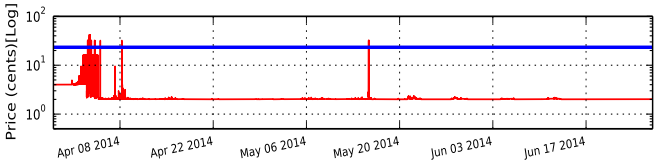
[6] On-demand VM prices change too, but over the period of months than hours. For simplicity, we use the static on-demand VM prices at the time of writing (July 15, 2014). So the on-demand VM price in 2012 would be different from this. However, the on-demand prices have consistently fallen over time. So the comparison we make between spot VM prices in 2012 with on-demand VM prices in 2014 is actually favourable to on-demand VMs.



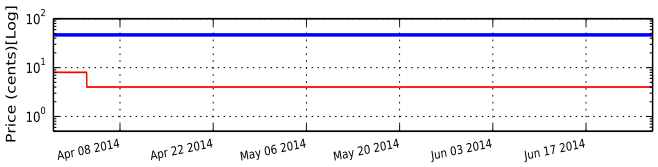
(a) Small VM



(b) Medium VM

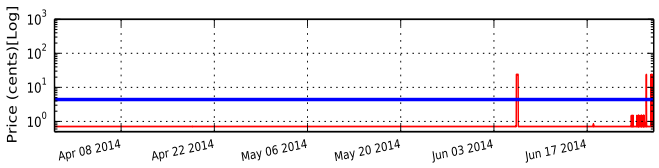


(c) Large VM

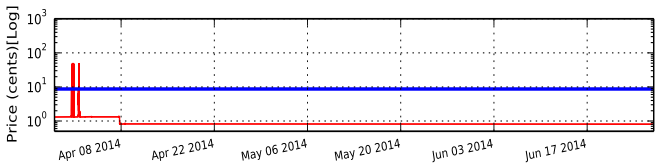


(d) XLarge VM

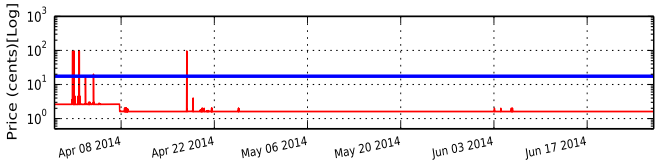
Fig. 1. AWS Spot prices (log scale) observed in *Apr–Jun 2014* in *AP-SE*. For reference, blue horizontal line shows fixed on-demand price for that VM.



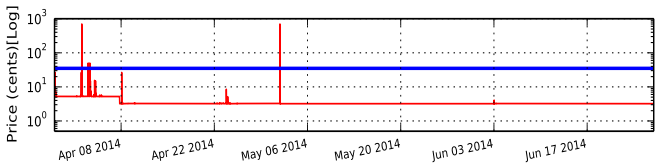
(a) Small VM



(b) Medium VM



(c) Large VM



(d) XLarge VM

Fig. 2. AWS Spot prices (log scale) observed in *Apr–Jun 2014* in *US-E*. For reference, blue horizontal line shows fixed on-demand price for that VM.

We collect spot price data [3] for two different time periods that are separated by 15 months: *Aug–Dec, 2012* for US-E, and *Apr–Jun, 2014* for AP-SE and US-E, for these four VM types using AWS's Command Line Interface [7]. We abbreviate these time periods as **2012** and **2014**. AWS reports the spot price only when it changes. We discretise this into uniform-spaced spot prices at 1 min intervals for our analysis. If prices change within one minute, we consider the maximum price within that interval; this happens fewer than 25 times in the 1.918 $million$ spot price intervals we consider overall. Figs. 1, 2, and 3 show the uniform-spaced discretised spot prices for the four VM types in AP-SE 2014, US-E 2014 and US-E 2012 [8].

B. Variation in Spot Prices

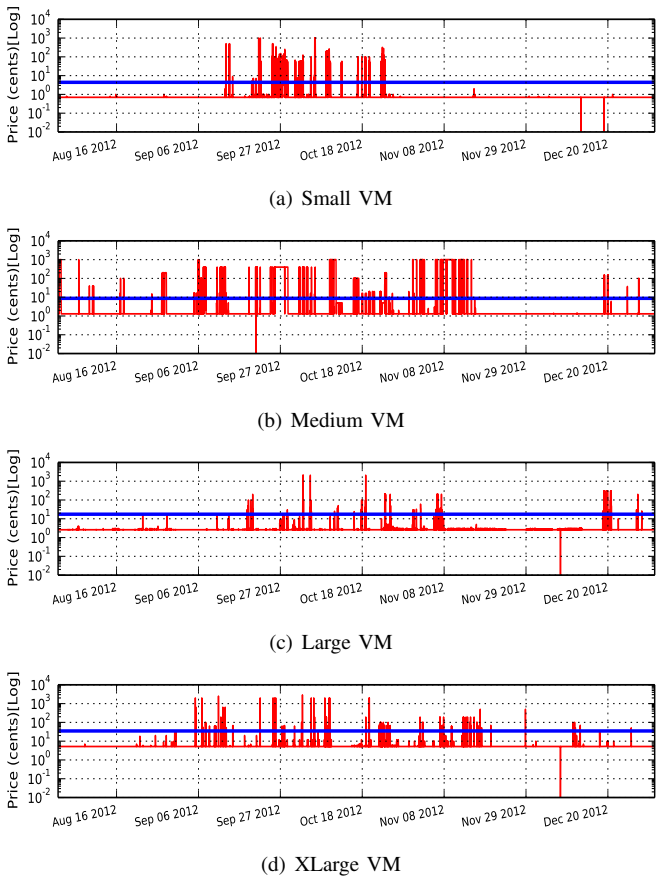
Figs. 1, 2 and 3 show the observed spot prices in US Cents (¢) per VM-hour over time. The blue solid line in each plot indicates fixed on-demand price for this VM, also provided in Table I. We observe that the spot price is often below the on-demand price, but there are sharp spikes when the spot price rises to much more than the on-demand price, sometimes peaking to US$10 per VM hour (Fig. 3(d)). These spikes are intermittent, and may indicate AWS trying to flush spot VM users due to internal demand [19].

[7]AWS only provides the past 3 months of spot price data. Since the accuracy of non-AWS provided historic data is unknown, we use AWS data that we directly collect: in 2014 for AP-SE, and in 2012 and 2014 for US-E.

[8]Pricing datasets used in this paper are available at http://dream-lab.serc.iisc.in/data/

Unlike upward price spikes, we notice that the minimum spot price for each VM type does not go below a threshold value for a region. As shown in Fig. 4, which plots the number of virtual CPUs (vCPUs) per VM along X Axis against its minimum observed spot price, this threshold spot price value is proportional to the number of vCPUs for AP-SE and US-E in 2014. Note that both small and medium VMs have 1 vCPU though they have 1 and 2 ECUs respectively, hence the constant minimum price for Small and Medium. This lower bound may correlate with the operating cost for AWS to run that VM type (e.g. power, cooling, personnel, taxes, etc.) at that data centre. US-E in 2012 (Fig. 3) has a few outliers, where the spot price has dropped briefly to US$0.0001. Hence its minimum observed price appears flat and close to US$0.00.

The prices across regions appear to be uncorrelated. For e.g., the Pearson's correlation coefficient (ρ) between spot prices for AP-SE and US-E in 2014 for each VM type are $\rho_{Small} = -0.023$, $\rho_{Medium} = 0.242$, $\rho_{Large} = 0.052$, $\rho_{XLarge} = 0.093$. While this lack of correlation may provide price arbitrage opportunities across regions [6], the price in the US-E is often smaller than the price at AP-SE. So applications may be better off bidding for instances in US-E if geo-location, network proximity or legal policies are not a constraint. For e.g., in 2014, the spot prices of US-E VMs were smaller than AP-SE VMs during these fraction of times: $\Delta_{Small} = 99.39\%$, $\Delta_{Medium} = 94.34\%$, $\Delta_{Large} = 95.91\%$, and $\Delta_{XLarge} = 94.56\%$. Note that the fixed prices of on-demand VMs in US-E are also cheaper their AP-SE counterparts (Table I).



(a) Small VM

(b) Medium VM

(c) Large VM

(d) XLarge VM

Fig. 3. AWS Spot prices (log scale) observed in *Aug–Dec 2012* in *US-E*. For reference, blue horizontal line shows fixed on-demand price for that VM.

Interestingly, despite no consistent price trends across regions, we do notice a distinctive step-down pattern in the spot prices for Medium, Large and XLarge VMs in AP-SE and US-E in Apr 2014. The price drops sharply by 50% for all VM types in AP-SE, followed by a similar drop for these VM types in US-E a few days later. This may indicate that a Cloud fabric upgrade or pricing algorithm upgrade is being rolled out in one data centre first, followed by other data centres. For e.g., AWS dropped their on-demand prices on Apr 1, 2014[9].

C. Spot Price Probability Distribution

The probability density function (PDF) for the discretised spot prices is calculated for the four VM types in AP-SE 2014, and US-E 2014 and 2012. These show the normalised frequency of occurrence of a particular spot price within the time periods considered in 2014 or 2012. For brevity, we only show the plots for medium VMs in Figs. 5(a), 5(b), and 5(c); plots for other VMs sizes are comparable.

We notice from Figs. 5(b), and 5(c) that the PDF of spot prices for US-E has changed between 2012 and 2014, with the range of probable values narrowing down from $10^{-2} - 10^{+3}$ to $10^{-1} - 10^{+2}$. This may be a seasonal characteristic within a year, or a changing price pattern across years.

Interestingly, a single spot price typically appears a majority of the time for a region and VM, during a time period.

[9]AWS Price Reduction, 26 Mar 2014. http://aws.amazon.com/blogs/aws/aws-price-reduction-42-ec2-s3-rds-elasticache-and-elastic-mapreduce/

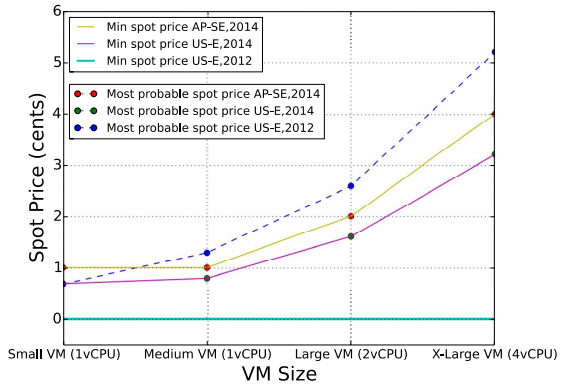


Fig. 4. The *minimum observed spot price* for each VM type, across regions and years, are in solid line. The *most probable spot price*, for the same VM types, region and year, are in dotted line with marker. Note that both these plots coincide exactly, except for US-E 2012.

In fact, such a price occurs $> 80\%$ of the time in most cases, except for AP-SE Small and Large in 2014, when it occurs $> 40\%$ of the time. Table II shows the most probable price and its frequency for different VMs, regions and periods. This indicates that spot prices often tend toward a particular probable value. In fact, when we overlay the most probable spot price on top of the minimum observed spot price in Fig. 4, these two values coincide (except for US-E 2012 that has a few outliers in minimum observed cost). This suggests that Amazon may often offer spot prices at near operational cost.

TABLE II. MOST PROBABLE SPOT PRICES AND THEIR PROBABILITY

Region	Time Period	VM Type	Spot Price [US Cents]	max(Pr)
AP-SE	2014	Small	1.01	0.41
AP-SE	2014	Medium	1.01	0.87
AP-SE	2014	Large	2.01	0.48
AP-SE	2014	XLarge	4.01	0.94
US-E	2014	Small	0.71	0.99
US-E	2014	Medium	0.81	0.88
US-E	2014	Large	1.61	0.88
US-E	2014	XLarge	3.21	0.80
US-E	2012	Small	0.70	0.95
US-E	2012	Medium	1.30	0.87
US-E	2012	Large	2.60	0.86
US-E	2012	XLarge	5.20	0.96

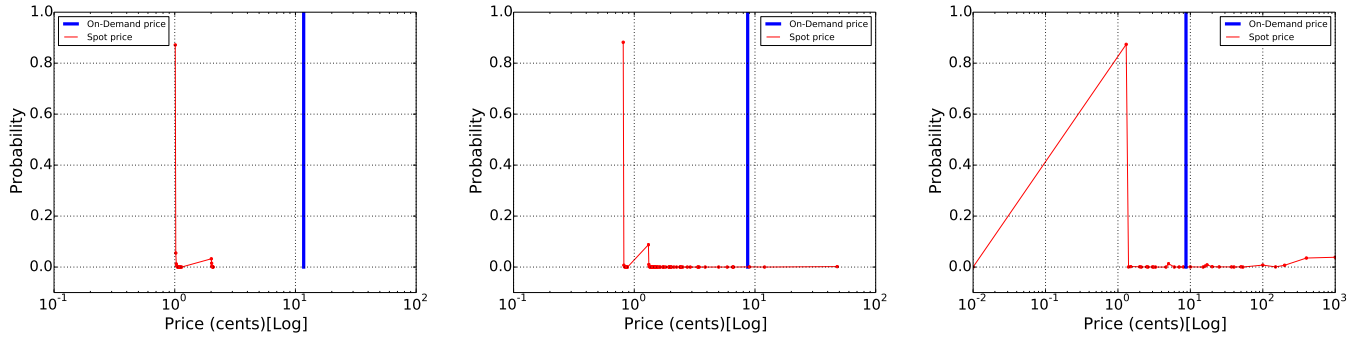
NOTE: Numbers in red highlight $< 80\%$ probability for most probable spot price

When we compare the most probable spot price against the equivalent fixed price on-demand VM in a region in Fig. 6, we notice that the former is at least $5\times$ cheaper than the on-demand price. As we increase the VM size, the price advantage between most the probable spot price and the fixed on-demand price increases to almost $12\times$ for XLarge VM in AP-SE in 2014. Larger spot VMs thus offer an enhanced price benefit.

D. Rate of Change in Spot Prices

To estimate the dynamism of spot price changes, we count the frequency of spot price changes with in a single day, both upward and downward. We also count the magnitude of price decreases and increases each day. Say $\mathcal{D}_j$ is the j^{th} day in a given time period, τ_j^{start} and τ_j^{end} are the timestamp for the start (midnight) and end of day D_j, $\mathcal{S}^\nu(t)$ is the spot price at timestamp t for VM type ν. The frequency of total price changes in the day $\mathcal{D}_j$, and the *frequency* of price increases and decreases in the day, for a VM ν are given by:

$$\mathcal{F}^\nu_{tot}(D_j) = \sum_{m=\tau_j^{start}}^{\tau_j^{end}} 1 \mid \mathcal{S}^\nu(m) \neq \mathcal{S}^\nu(m+1)$$



(a) AP-SE, 2014 (b) US-E, 2014 (c) US-E, 2012

Fig. 5. Probability distribution of spot prices (log scale) for *Medium VMs*, in different regions and time periods. For reference, blue vertical line shows fixed on-demand price for medium VM in each region.

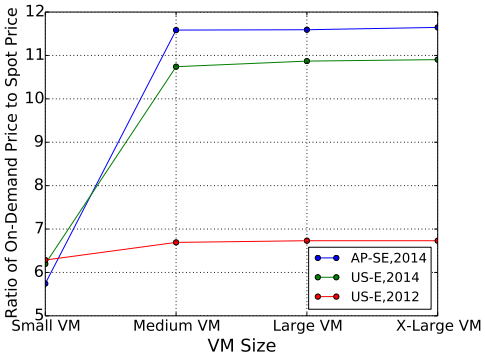


Fig. 6. Ratio of fixed on-demand price to the most probable spot price. The most probable spot price is $5-12\times$ cheaper than the on-demand price.

$$\mathcal{F}^{\nu}_{inc}(D_j) = \sum_{m=\tau_j^{start}}^{\tau_j^{end}} 1 \mid \mathcal{S}^{\nu}(m) < \mathcal{S}^{\nu}(m+1)$$

$$\mathcal{F}^{\nu}_{dec}(D_j) = \sum_{m=\tau_j^{start}}^{\tau_j^{end}} 1 \mid \mathcal{S}^{\nu}(m) > \mathcal{S}^{\nu}(m+1)$$

where m is in minute increments. Similarly, the net *magnitude* of price changes, $\mathcal{M}^{\nu}_{tot}(\mathcal{D}_j)$, in a day $\mathcal{D}_j$ is given by:

$$\frac{\sum_{m=\tau_j^{start}}^{\tau_j^{end}} \mathcal{S}^{\nu}(m+1) - \mathcal{S}^{\nu}(m) \mid \mathcal{S}^{\nu}(m) \neq \mathcal{S}^{\nu}(m+1)}{\mathcal{F}^{\nu}_{tot}(D_j)}$$

Likewise defined for the magnitude of price increases and decreases, $\mathcal{M}^{\nu}_{inc}(\mathcal{D}_j)$ and $\mathcal{M}^{\nu}_{dec}(\mathcal{D}_j)$, within a day $\mathcal{D}_j$.

We plot the frequency and magnitude of spot price changes per day for medium VM in AP-SE in 2014 in Fig. 7. For brevity, we omit plots for other VM sizes, regions and time periods, but the results are similar. Surprisingly, despite the price changes appearing to be intermittent in Figs. 1–3, we see that the number of times a price changes in the positive and negative direction within each day is almost identical. While the number of changes per day varies (e.g. Jun, 2014 has no changes but early Apr, 2014 has > 50 changes per day in in Fig. 7(a)), these are symmetric in terms of frequency. Furthermore, the magnitude of positive and negative changes within a day are themselves similar, which means that despite prices changes within a day, the net price change at the end of a day tends to zero, as seen in Fig. 7(b).

Table III shows the correlation between the number and magnitude of spot price increases and decreases within a day, across VMs, regions and time periods. But for a few exceptions in red, we see that $\rho > 0.950$ for both frequency and magnitude, strongly indicating that net change in either direction is conserved within a 24 hr period. In fact, we see a similar conservation (though slightly weaker; not shown) within a 12 hr period too. This suggests that Amazon's spot pricing is incremental/symmetric in nature, and that prices that go up tend to come down, and vice versa, and are highly conserved within a single day. As a result, this periodicity can be exploited in designing spot VM bidding strategies.

TABLE III. CORRELATION BETWEEN THE FREQUENCY/MAGNITUDE OF PRICES INCREASES AND DECREASES WITHIN A 24 HOUR PERIOD

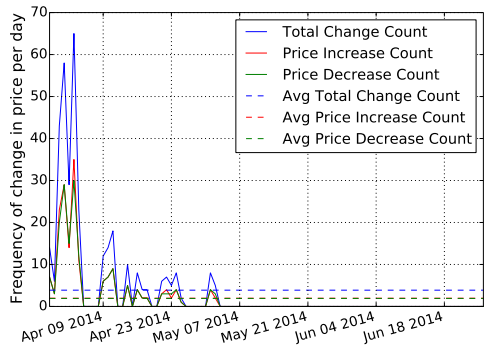
Region	Time Period	Small	Medium	Large	XLarge
Correlation between Freq. of Increase and Freq. of Decrease					
AP-SE	2014	0.990	0.995	0.998	0.703
US-E	2014	0.981	0.999	0.992	0.991
US-E	2012	0.984	0.985	0.996	0.992
Correlation between Mag. of Increase and Mag. of Decrease					
AP-SE	2014	0.991	0.648	0.999	-0.009
US-E	2014	0.862	0.999	0.995	0.994
US-E	2012	0.973	0.989	0.989	0.976

NOTE: Numbers in red highlight < 0.950 correlation

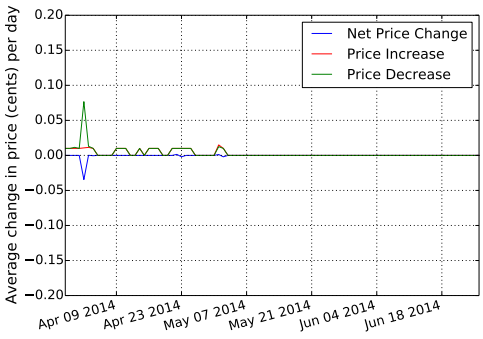
IV. ANALYSIS OF JOBS ON SPOT VMS

In this section, we translate our observations on spot prices into their corresponding impact when running jobs on spot VMs through a simulation study. We make several simplifying assumptions that help generalise our job analysis. Individual jobs run exclusively on a spot VM, each with a resource requirement (*job size*) specified in terms of *ECU core-minutes*. Elastic Compute Unit (ECU) is a normalised unit of compute capability reported for VMs by Amazon. Small, Medium, Large and XLarge VMs of the *m1* class have 1, 2, 4, and 8 ECUs, respectively. We assume these jobs are CPU bound, and can fit in the memory, storage and network capacity available on any of the VM types listed in Table I. We consider diverse job sizes: $10, 30, 60, 240, 480$, and 1440 core-mins. The wall clock duration of each job is defined as $\delta = \frac{Job\ Size}{ECU\ of\ VM}$. E.g., a 30 core-min job will take 15 mins to complete on a Medium VM, while a 1440 core-min job will take 3 hrs on an XLarge VM. Unless a spot VM is reclaimed, both spot and on-demand VMs of the same size take the same duration for a job.

To run jobs, we bid for spot VMs at prices that are fractional values of the fixed on-demand prices of the same



(a) Frequency of Spot Price Changes per day. Horizontal dashed lines show averages across the entire period.



(b) Magnitude of Spot Price Changes Per Day

Fig. 7. Frequency and Magnitude of change in spot prices – increases, decreases and total changes – per day for *Medium VMs* in *AP-SE, 2014*.

VM type in that region. We consider bid prices at 70%, 80%, 90%, 100%, and 110% of the on-demand price. Amazon assigns spot VMs if the bid price is greater than the current spot price, and these VMs are retained only until the bid price remains above the spot price, i.e., if the spot price increases above the bid price, it is an *out-of-bid event*, the VM reclaimed, and the job on it is terminated with all progress lost. For simplicity, we do not change the bid price once defined.

A job is charged only at the spot price, even if the bid price is greater. Billing is in hourly increments, and charges accrue immediately at the VM hour boundary (or partial hour, if the job completes within the hour). The spot price when the VM is acquired is used as the billing rate for the following VM hour. Each subsequent hour uses the current spot price at the start of that VM hour. However, if an out-of-bid event happens at anytime in-between, the partial hour used is not billed. Any whole hours used before the last partial hour, though, is charged even as the job has failed.

We use the uniform-spaced spot prices at 1 min intervals available for the four VM types in AP-SE in 2014, and US-E in 2012 and 2014 for our simulation study. Each job size is "started" on every VM type in each region and time period, at each minute interval. Thus, for a 30 core-min job, we simulate its run $(90\ days \times 24\ hours \times 60\ mins - 29\ mins) = 129,571$ times for a Small VM in AP-SE during Apr-Jun 2014. We do likewise for the 6 job sizes, 4 VM types and 3 regions/periods, for a total of about 9.31 $million$ simulated job runs.

A. Definitions

1) Successful and Failed Jobs: A job is successful if the spot price of the VM on which the job is running remains less than the bid price of that VM, during the entire duration of job. If the spot price of the VM becomes greater than the bid price at any point during the job's duration, the job fails.

2) Reliability: The Reliability (or success rate), $\mathcal{R}$ for a job is defined as:

$$\mathcal{R} = \frac{Number\ of\ Successful\ Jobs}{Total\ Number\ of\ Jobs\ Attempted}$$

The $Failure\ Rate$ for a job is $(1 - \mathcal{R})$.

3) Savings: The relative savings for a Job $\mathcal{J}_\delta$, of wall clock duration δ when running on VM of size ν, started at time τ_i:

$$\mathcal{P}^\nu(\mathcal{J}_\delta, \tau_i) = \sum_{h=0}^{\delta} \mathcal{O}^\nu - \mathcal{S}^\nu(\tau_i + h)$$

where h is in hourly increments, $\mathcal{O}^\nu$ is the fixed price of on-demand VM of size ν, and $\mathcal{S}^\nu(t)$ is the spot price of VM of size ν at timestamp t. Note that savings is defined only for successful jobs, and it accumulates by the hour.

The cumulative normalised savings for the above job, started at each minute boundary of the spot price time period $\langle \tau^{begin}, \tau^{end} \rangle$ (e.g. $\langle$1 Apr 2014, 30 Jun 2014 $\rangle$), with a reliability of $\mathcal{R}$ is:

$$\mathcal{P}^\nu\%(\mathcal{J}_\delta, \tau^{begin}, \tau^{end}) = \frac{\sum_{m=\tau^{begin}}^{\tau^{end}-\delta+1} \mathcal{P}^\nu(\mathcal{J}_\delta, m)}{(\mathcal{O}^\nu \times \delta) \times (\mathcal{R} \times n)}$$

where $n = \big((\tau^{end} - \delta + 1) - \tau^{begin})\big)$ is the number of jobs simulated in time period $\langle \tau^{begin}, \tau^{end} \rangle$, and m is in minute increments. $(\mathcal{O}^\nu \times \delta)$ is the job's cost on fixed-price on-demand VMs, and $(\mathcal{R} \times n)$ gives the number of successful jobs.

4) Loss: The relative loss for a Job $\mathcal{J}_\delta$, of wall clock duration δ when running on VM of size ν, started at time τ_i and with an out-of-bid event occurring at the hour $\widehat{\delta}$ from the start time is given by:

$$\mathcal{L}^\nu(\mathcal{J}_\delta, \tau_i, \widehat{\delta}) = \sum_{h=0}^{\widehat{\delta}-1} \mathcal{S}^\nu(\tau_i + h)$$

where h is in hourly increments, and $\widehat{\delta} < \delta$. Note that loss is defined only for failed jobs, and it accumulates by the hour.

Similarly, the cumulative normalised loss for the above job, started at each minute boundary of the spot price time period $\langle \tau^{begin}, \tau^{end} \rangle$, with a reliability of $\mathcal{R}$ is:

$$\mathcal{L}^\nu\%(\mathcal{J}_\delta, \tau^{begin}, \tau^{end}) = \frac{\sum_{m=\tau^{begin}}^{\tau^{end}-\delta+1} \mathcal{L}^\nu(\mathcal{J}_\delta, m, \widehat{\delta})}{(\mathcal{O}^\nu \times \delta) \times ((1 - \mathcal{R}) \times n)}$$

where $\widehat{\delta}$ is a function of the spot price when each job is simulated, and $((1 - \mathcal{R}) \times n)$ gives the number of failed jobs.

5) Effective Savings: The reliability weighted savings, or effective savings, for a job with reliability $\mathcal{R}$, normalised savings $\mathcal{P}\%$ and normalised loss $\mathcal{L}\%$ is given by $\mathcal{R} \times \mathcal{P}\% - (1 - \mathcal{R}) \times \mathcal{L}\%$.

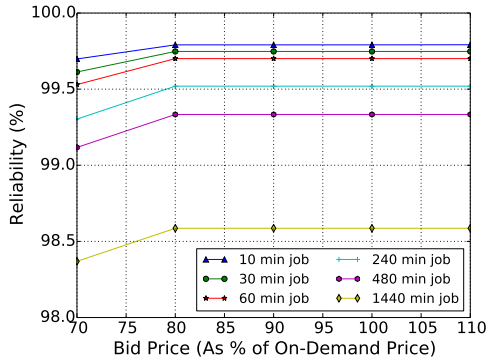


Fig. 8. Reliability of jobs on *US-E 2014, Medium*, as the bid price, as a fraction of on-demand price, increases.

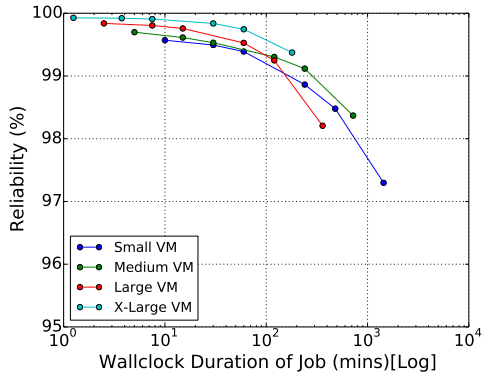


Fig. 10. Reliability of jobs running on various VMs in *US-E, 2014*, with varying wallclock duration of jobs, when bidding at *70% of on-demand price*

B. Reliability of Jobs

Since spot VMs have a perceived lack of robustness, first we discuss the reliability of jobs on spot VMs before their potential cost benefits. Fig. 8 shows a representative plot of how the reliability of various sized jobs change as the bid price in increased from 70% to 110% of the on-demand price, for Medium VM in US-E in 2014. Some observations are that *the reliability is relatively high across the board, at above* 98%, and there is barely an improvement of 1% as the bid price goes from 70% to 110%.

We focus on bid prices at 70% of on-demand price, to characterise the lower end of reliability among our parameter space. Fig. 9 shows the reliability vary across regions/periods for different VM sizes, as the job size varies along the X Axis, using a 70% bid price. We see that a common trend is that as the job size increases, the reliability decreases. This is understandable, since the larger the job, longer its run duration and greater the chance of an out-of-bid event. We also see that the reliability of jobs running on a VM size changes across region and over time, as seen by the different slopes.

Longer job durations can increase the chance of an out-of-bid event. We derive a stronger correlation between the wallclock time that a job runs for on a VM and the reliability of jobs on that VM. Fig. 10 illustrates this for US-E in 2014, as a complement of the Fig. 9(b). Here, the plots are more closely grouped across VM sizes, indicating $\sim 1\%$ drop in reliability for every 200 $mins$ increase in a job's duration.

C. Savings and Loss of Jobs

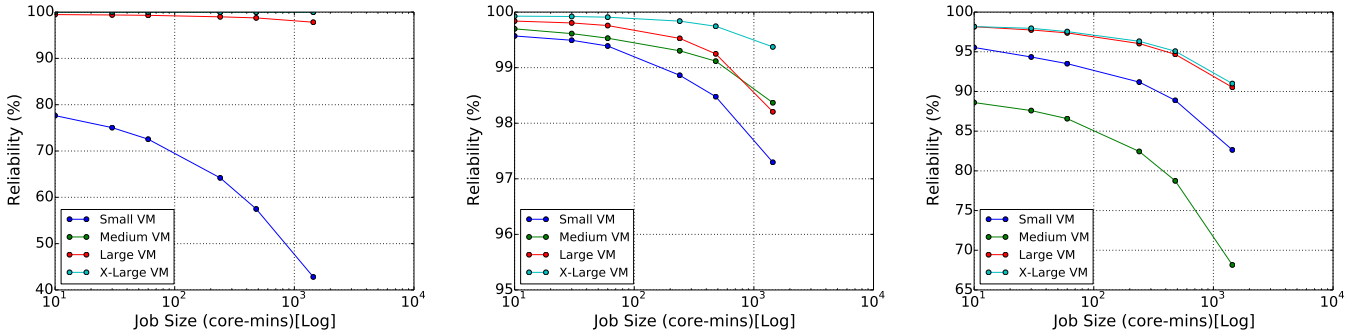
The normalised savings for a job is the fractional benefit that can be gained from running it successfully on spot VMs, relative to the cost on an equivalent fixed-price on-demand VMs. For jobs that fail due to out-of-bid events, there is a cost paid for whole VM hours used without accomplishing the job. We calculate the normalised loss for a failed job as the loss from running it on spot VMs, relative to running it successfully on a fixed-price on-demand VM. Fig. 11 plots these values when bidding at 70% of on-demand price. Across all regions, *we gain a savings of over* 80% *by using spot VMs* over on-demand VMs. This corresponds to the ratio between most probable spot price and on-demand price being $5\times$ or more in Fig. 6.

Notice that as the job sizes increase, we do not see a tangible change in the savings % gained. This is understandable, since the normalised savings comes from the difference between the spot and on-demand price (§ IV-A3). As spot price does not change often, the gains remain constant. We also see that AP-SE and US-E Small VMs offer a smaller savings, at 80% compared to 90% for the other VMs. From Table II, US-E Small VM's spot price at US$0.0071, in 99% of the time, while Medium is at US$0.0081, in 88% of the time. So a Small spot priced VM with half the compute power of a Medium costs almost as much most of the time. The on-demand prices of a Small is however half as much as an on-demand Medium VM. *As a result, small spot-priced VMs offer lower savings.*

The loss that is suffered on account of failed jobs is also relatively small, and often limited to $< 5\%$, with the only exception being when using Small VMs on US-E in 2014, with large jobs; it rises to 7%. We do see that as the job size increases, the probability of loss also increases linearly. It grows from 0% for jobs that fit within one wallclock hour of a VM, by about 1% for every additional 400 $mins$ of job size. Note that this loss does not include the lost opportunity cost of failing to run the job. While not plotted, we report that there is negligible impact of the bid price increasing from $70\% - 100\%$, but we do see the average loss % grow sharply by $\sim 5\%$ when bidding over the on-demand price at 110%.

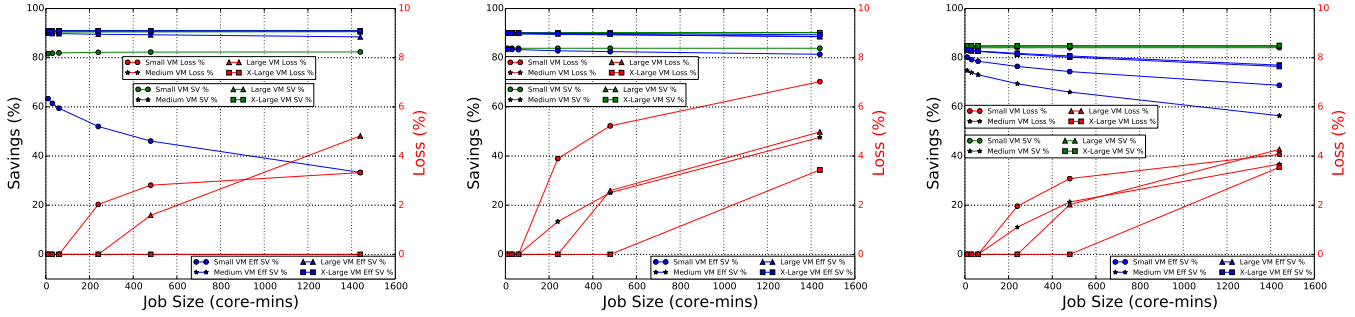
The effective savings offers a reliability weighted function over savings and loss. This is the bottom-line savings (ignoring the lost opportunity cost of failed jobs). Fig. 11 shows these in blue lines. In most cases, *the effective savings is fairly high at* 90%. However, US-E in 2012 shows lower effective savings across VMs as the job size increases, as does AP-SE in 2014 for Small VM. Notice that in Fig. 9, the reliability for these corresponding VMs drops with the job size. So we see the effect of a linear combination of decreasing reliability and increasing normalised loss in Fig. 11. *Note that Small VMs suffer a dual penalty*: jobs run for a longer duration on them, increasing the chance of an out-of-bid event, and the loss also accumulates over more hours for them.

Lastly, we look at a scatter plot of the normalised savings % vs. the reliability % for different VM types, across regions and time periods, in Fig. 12. The colors are grouped by the VM type, so VMs that cluster along the top right of the plot offer significant savings with high reliability, on an average across the jobs. We see that for AP-SE in 2014, all but small VMs consistently offer a $90\%+$ *savings over on-*



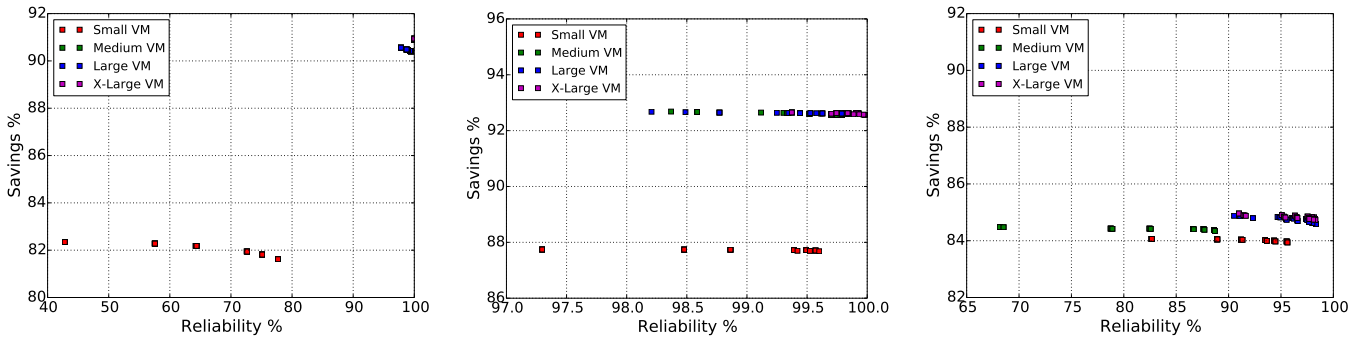
(a) AP-SE, 2014 (b) US-E, 2014 (c) US-E, 2012

Fig. 9. Reliability of various VMs with varying job sizes, when bidding at 70% of on-demand price. Note that the Y Axis scaling is different across plots.



(a) AP-SE, 2014 (b) US-E, 2014 (c) US-E, 2012

Fig. 11. Primary Y Axis shows Normalised Savings% (Green) and Effective Savings% (Blue), as Job Size increases on X Axis. Secondary Y Axis shows Normalised Loss% (Red) for the job sizes. We bid at 70% of on-demand price.



(a) AP-SE 2014 (b) US-E 2014 (c) US-E 2012

Fig. 12. Scatter plot of Reliability % vs. Average Savings % for different VMs, across regions and time periods. Each marker represents a job size, and colors are assigned by VM Size. Note that X Axis scales vary across plots.

demand with a 95%+ *reliability*. Medium, Large and XLarge are equally good in US-E in 2014, with even the Small offering $> 95\%$ reliability and $> 85\%$ savings. US-E in 2012, however, offers lower reliability for Small and Medium and a slightly diminished savings of 85% across the board.

V. Conclusion

In this paper, we have provided an analysis of AWS spot prices for the AP-SE and US-E regions. We see that prices across regions appear to be uncorrelated except when major pricing changes or software updates happen. There are seasonal and annual variations in the pricing pattern, but the probability that the spot price is at the minimum observed price is fairly high. Also, larger spot VM offer a better price advantage over on-demand VMs. It is interesting to note that the number and magnitude of price changes in the upward and downward directions are conserved within each day.

More so, we have mapped its impact, through a simulation study, on running diverse job sizes. Using meaningful metrics such as reliability and effective savings, our study offers key insights into practically using spot-priced VMs relative to on-demand VMs. The reliability of jobs is relatively high across the board, at above 98%, with barely an improvement as the bid price goes beyond 70% of on-demand VM price. We see effective savings of over 80% in most cases when using spot VMs, with 90% effective savings observed in 2014 data. Small VM offer lower savings due to several factors, and are less preferred. These results suggest that AWS spot instances are highly favourable for cost-conscious enterprises in emerging markets.

REFERENCES

[1] W. Lu, J. Jackson, and R. Barga, "Azureblast: A case study of developing science applications on the cloud," in *ACM International Symposium on High Performance Distributed Computing*, 2010.

[2] N. Kshetri, "Cloud computing in developing economies," *Computer*, vol. 43, 2010.

[3] A. W. Service, "Amazon ec2 spot instances," Jul 2014, http://aws.amazon.com/ec2/purchasing-options/spot-instances/.

[4] M. Zafer, Y. Song, and K.-W. Lee, "Optimal Bids for Spot VMs in A Cloud for Deadline Constrained Jobs," in *IEEE International Conference on Cloud Computing (CLOUD)*, Jun. 2012.

[5] N. Chohan, C. Castillo, M. Spreitzer, M. Steinder, A. Tantawi, and C. Krintz, "See spot run: Using spot instances for mapreduce workflows," in *USENIX conference on Hot topics in Cloud Computing*, 2010, pp. 7–7.

[6] H. K. Cheng, Z. Li, and A. Naranjo, "Cloud computing spot pricing dynamics: Latency and limits to arbitrage," Warrington College of Business Administration and Hough S Graduate of Business and University of Florida, Tech. Rep., 2012.

[7] M. Mazzucco and M. Dumas, "Achieving performance and availability guarantees with spot instances," in *High Performance Computing and Communications*, 2011.

[8] H.-Y. Chu and Y. Simmhan, "Cost-efficient and resilient job life-cycle management on hybrid clouds," in *IEEE International Parallel and Distributed Processing Symposium*, 2014.

[9] G. Lee, B.-G. Chun, and H. Katz, "Heterogeneity-aware resource allocation and scheduling in the cloud," in *USENIX Conference on Hot Topics in Cloud Computing*, 2011.

[10] D. Warneke and O. Kao, "Exploiting dynamic resource allocation for efficient parallel data processing in the cloud," *IEEE Transactions on Parallel and Distributed Systems*, vol. 22, no. 6, 2011.

[11] M. Mao and M. Humphrey, "Auto-scaling to minimize cost and meet application deadlines in cloud workflows," in *Supercomputing*, 2011.

[12] A. G. Kumbhare, Y. Simmhan, and V. K. Prasanna, "Plasticc: Predictive look-ahead scheduling for continuous dataflows on clouds," in *IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid)*, 2014.

[13] Y.-J. Hong, M. Thottethodi, and J. Xue, "Dynamic server provisioning to minimize cost in an iaas cloud," in *SIGMETRICS*, 2011.

[14] O. A. Ben-Yehuda, M. Ben-Yehuda, A. Schuster, and D. Tsafrir, "Deconstructing amazon ec2 spot instance pricing," *ACM Transactions on Economics and Computation*, vol. 1, 2013.

[15] B. Javadi, R. K. Thulasiram, and R. Buyya, "Statistical modeling of spot instance prices in public cloud environments," in *4th IEEE/ACM International Conference on Utility and Cloud Computing (UCC 2011)*, vol. 1, 2011.

[16] F. I. Popovici and J. Wilkes, "Profitable services in an uncertain world," in *Supercomputing*, 2005.

[17] F. Teng and F. Magoulès, "A new game theoretical resource allocation algorithm for cloud computing," in *International Conference on Advances in Grid and Pervasive Computing*, 2010.

[18] B. Sharma, R. K. Thulasiram, P. Thulasiraman, S. K. Garg, and R. Buyya, "Pricing cloud compute commodities: A novel financial economic model," in *Cloud and Grid Computing*, 2012.

[19] Dave@AWS, "Is amazon effectively killing the spot market for ec2 instances?" Sep 2011, https://forums.aws.amazon.com/message.jspa?messageID=281545.