Non-parametric Bootstrap Detection of Availability Service Level Objective Violations in Cloud Storage

Maurizio Naldi
Dpt. of Civil Engineering and Computer Science
University of Rome Tor Vergata
00133 Rome, Italy
Email: maurizio.naldi@uniroma2.it

***Abstract*—Service quality commitments in cloud service provisioning are typically described in Service Level Agreements (SLA). Service availability is always a major parameter to be included in such SLAs. and the cloud provider is bounded to guarantee a minimum availability value, for which current cloud monitoring systems employ a naive estimator. In this paper a new estimation method is proposed for service availability, which is based on the bootstrap technique and employs a non-parametric statistical hypothesis test. Through Monte Carlo simulation, the method is shown to be much more accurate than the naive one under three stochastic models for the durations of operating and outage periods, exhibiting a Type I error probability lower than 1% in most cases, while the naive estimator yields error probabilities around 40%.**

I. Introduction

Since its inception, cloud storage has progressively gained adoption in a variety of contexts [1], [2] among all types of customers (business as well as individuals [3], [4]), with a large push on competitive pricing schemes [5]. Among companies a relevant factor is the promise of a strong reduction in costs, or at least a switch from CAPEX to OPEX [6].

Though cloud storage is provided free of charge by some providers for the less demanding customers, who just need some limited amount of storage, its quality of service is instead described by a set of Service Level Objectives, embodied in a Service Level Agreement (SLA), in all other cases. Among the parameters included in any SLA, availability, i.e. the capability of providing the service without interruptions, is always a key requirement [7], [8], which is particularly critical in scientific applications [9] but is also used as a parameter to rank cloud providers and switch from one provider to another on the basis of their performance [10]. Many efforts have been devoted to assess and/or improve the availability of a cloud provider (and its datacenters) on the basis of their architecture and components [11]–[15].

However, the actual availability may fall behind expectations [16], so that the cloud provider may violate the availability SLA and be subject to penalty. Though the cloud provider may devise its penalty policy to minimize the damage to its profit [17], it is of paramount importance that the availability is evaluated correctly, so that both parties are treated fairly and the cloud provider is not unduly damaged.

In this paper we claim that the current methods to evaluate the availability of cloud storage systems exhibit a poor statistical accuracy and propose a non-parametric statistical hypothesis test, based on the boostrap technique, to improve the statistical accuracy of availability evaluation. After defining availability for a cloud storage system in Section II, the new method is proposed in Section III. In Section IV we show that:

- the naive estimator yields a Type I error probability around 40% independently of the underlying model and of the average outage duration;
- the bootstrap estimator that we have proposed yields a Type I error probability at least one order of magnitude lower than the naive estimator;
- the observation interval must be chosen long enough as to avoid ending with zero or too few outages during the observation interval;
- the error performance of the bootstrap estimator worsens as the average outage duration grows (due to the diminishing number of outage events).

II. Availability in Service Level Agreements

As noted in the Introduction, the availability is one of the key parameters included in any Service Level Agreement. In this section we report a brief review of its measuring methods and its setting in SLAs.

Though several definitions of availability may be provided [18], [19], we stick to a simple one, which may be applied quite straightforwardly to cloud storage. A thorough analysis of the several parameters that may be included into an SLA for availability is contained in [20]. Here we consider the service to be either available or not. Definitions that include consideration of the delay in providing the data may be brought back into this framework by considering a threshold on the delay, so that the service is considered to be available if the data are delivered within that threshold and unavailable otherwise. The availability is therefore defined as the fraction of time the cloud is operating. Since the processes governing the operations of the cloud are stochastic, we can define the availability by introducing the two variables S (duration of an uptime, or operating, period) and D (duration of an outage, or downtime period). The availability is then

$$A = \frac{\mathbb{E}[S]}{\mathbb{E}[S] + \mathbb{E}[D]}, \tag{1}$$

where $\mathbb{E}[\cdot]$ represents the expected value. An alternative definition involves the two variables MTTF (Mean Time To Failure) and MTTR (Mean Time To Repair), so that

$$A = \frac{\text{MTTF}}{\text{MTTF} + \text{MTTR}}. \tag{2}$$

The two definitions are equivalent if the "Mean" in Equation (2) is taken over the population. If it is instead taken over a sample, Equation (2) represents an operational definition, so that the quantity measured by this definition is actually a random variable itself.

In practice, the availability in present monitoring systems as proposed in the literature is measured by the single instance value

$$\hat{A} = \frac{\sum_{i=1}^{N} s_i}{\sum_{i=1}^{N} s_i + \sum_{i=1}^{N} d_i}, \tag{3}$$

where N is the number of outages occurring during the measurement period, and the s_i's and d_i's are the durations respectively of the operating periods and the outage periods. This is the approach taken, e.g., in [21], [22]. This definition is also that implied, e.g., in the SLA definition proposed by Amazon, for which the Monthly Uptime Percentage is calculated by subtracting from 100% the percentage of minutes during the month in which any of the Included Products and Services was in the state of Region Unavailable (see https://aws.amazon.com/ec2/sla/). In the following, we refer to the estimator expressed by Equation (3) as the naive estimator.

We assume that the cloud monitoring system can probe the cloud with a frequency as high as desired so as to identify the onset of an outage with very high precision. For the time being, we do not consider the problems related to the presence of the network between the cloud customer (or the cloud monitoring system) and the cloud itself: as shown in [23]–[25], network effects may hide the true availability of cloud services. We assume that the measurement has been somewhat refined by taking into account the network contribution. We also consider the measurement to take place over an observation horizon T, which in the following will be set as either a month or a year.

An extensive survey of availability targets as set in SLAs is shown in Table I (excerpted from [26]). We see that the majority of cloud providers claim an ambitious 100% availability. This is equivalent to say that the cloud is always available. Even if shutdown periods due to preventive maintenance are not counted as contributing to the total downtime (a typical assumption), this claim is hard to believe: no physical system can exhibit a total absence of failures. As a matter of fact, contrary to their claims, outages are reported even for those cloud providers claiming a 100% availability. For example, if we consider the first provider in Table I declaring uninterrupted operations (CloudSigma), the company itself declares two connectivity failures in May 2018 alone (as reported on their website status.cloudsigma.com).

III. A BOOTSTRAP-BASED TEST

In Section II, we have defined the availability and introduced the naive estimator. In this section, we propose an alternative estimator based on the bootstrap technique and develop a related SLA compliance test. In the following, we indicate by X the bootstrap-based availability estimate over a single period of observation.

Cloud provider	Availability [%]
Amazon Web Service	99.95
AT&T Synaptic	99.9
CloudSigma	100
ElasticHosts	100
FlexiScale	100
GoGrid	100
JoyentCloud	100
layeredtech	100
Locaweb	99.9
Opsource	100
Rackspace	100
ReliaCloud	100
RSAWEB Cloud servers	ND
SliceHost	ND
Storm On demand	100
Terremark vCloud express	100
VPSNET	100

TABLE I
SERVICE LEVEL AGREEMENT COMMITMENTS FOR AVAILABILITY

In our case we want to test whether the actual availability is equal to the target one. We identify the latter as A_0, setting a value for it, e.g., $A_0 = 0.99$. We formulate the test as a statistical hypothesis test, where the null hypothesis H_0 is that the availability equals the target one, i.e.

$$H_0 \triangleq A \geq A_0. \tag{4}$$

If the null hypothesis is true, the provider is considered as compliant with SLA commitments. We wish to compare the null hypothesis versus the alternative hypothesis

$$H_1 \triangleq A < A_0. \tag{5}$$

If the alternative hypothesis is true, the provider is considered as non compliant.

Of course, we do not know the actual availability A but just its estimate X, which is a random variable. IT is quite straightforward to assume that such an estimate depends on the actual availability. We also assume that it belongs to the location family, i.e. distributions arising from different values of the availability are just shifted versions of the same function, with the shift being represented by the availability itself:

$$f_X(x|A) = f_X(x - A). \tag{6}$$

We recall that the location family includes the large location-scale family to which well known distributions belong, such as the normal, uniform, logistic, Student's t, or the Generalized Extreme Value one.

If we set a confidence level c (e.g., $c = 0.05$), we reject the null hypothesis (hence, declare the cloud provider non-compliant) when $X < t$, with the threshold t set by the following equation, since we use a one-sided test (we are concerned about the actual availability being lower than the target one)

$$\mathbb{P}[X < t|H_0] = c. \tag{7}$$

or, equivalently, t is the c-order percentile of X under the hypothesis H_0:

$$t = X_{(c)}|H_0, \tag{8}$$

Applying the test could be straightforward if we knew the distribution of X, which we unfortunately do not know. We can resort to the bootstrap technique, described first by Efron and Tibshirani in [27], which allows to compute the accuracy of an estimate when it cannot be derived from a model. In bootstrap, we apply random sampling with replacement from a single sample of our variable to obtain a number of bootstrap replicas, which can then be employed to compute, e.g., the standard error of the estimate.

In our case, if we generate B bootstrap replicas of the sequence of running times and outages that has been actually observed (which provided the value a for the availability, as measured through Equation (3)), we obtain the empirical cumulative distribution function $\hat{F}_X(x)$, which we can use as a bootstrap-based estimation of the distribution $F_X(x)|A = a$. The sought-after distribution under the hypothesis H_0 would then be related to the bootstrap-based one by

$$F_X(x)|H_0 \simeq \hat{F}_X(x - a + A_0)|A = a \tag{9}$$

The test condition expressed by Equation (7) can therefore be expressed as follows

$$\begin{aligned}\mathbb{P}[X < t|A = A_0] &= \mathbb{P}[X - a + A_0 < t|A = a] \\ &= \mathbb{P}[X < t + a - A_0|A = a] = c\end{aligned} \tag{10}$$

Since we can estimate the percentile $\hat{b} = X_{(c)}|A = a$ by bootstrap, and $b = t + a - A_0$ by Equation (10), we have an estimate of the critical threshold $t \simeq \hat{b} - a + A_0$. We reject therefore the null hypothesis if the observed value is

$$a < t \simeq b - a + A_0 \rightarrow a < \frac{b + A_0}{2}. \tag{11}$$

The critical threshold on the observed availability is therefore the arithmetic average of the target availability and the c-percentile of the bootstrap estimator.

IV. Method validation

In order to test the accuracy of our estimation procedure, we have applied it to a set of cases where we are able to exactly compute the availability. In this section we describe those cases and report the results of a battery of Monte Carlo simulation tests.

Our test procedure consists of setting a case where the statistics of operating and outage periods are well defined, so that we can simulate the case through a Monte Carlo approach and compare the simulation results against the true availability. In the simulation we apply both the bootstrap-based test that we have described in Section III and a single instance test. The latter consists in measuring the availability over the observation horizon as per Equation (3), i.e., the naive estimator, and compare it against the threshold A_0: we declare the provider non-compliant if we have $\hat{A} < A_0$.

The simulation procedure is described as Algorithm 1.

Algorithm 1 Simulation of compliance checks

```
for i = 1 to Number of simulation runs do
    Reset time counter
    Reset Overall outage time
    while Time counter <Simulation horizon do
        Generate random duration of operating period
        Generate random duration of outage
        Add to overall outage time
        Update time counter
    end while
    Compute availability for simulation instance
    Generate bootstrap replicas
    Compute bootstrap percentile
    Compute critical threshold
    Apply bootstrap test as per Inequality (11)
    Apply the single instance test
end for
```

As test cases we have considered the following three models:

- Markov;
- Poisson-Pareto;
- Pareto-Lognormal.

All three models have been proposed in the literature, as detailed in the following.

It is to be stressed that the use of models, though proposed in the literature, is due to them providing a ground truth to evaluate the bootstrap-based technique. Their use allows us to simulate a system with a controlled availability. We would not be able to assess the performance of the method by relying on experimental data alone, since we would not be able to associate the true availability value. It is also to be noted that the bootstrap method is non-parametric, since it does not rely on specific assumption about the distribution of uptimes and downtimes.

All the simulations reported in the following have been carried out for two values of target availability $A = 0.99, 0.999$, a confidence level $c = 0.05$, a number of boostrap replicas $B = 1000$, and 10000 simulation runs. The simulation design parameters have been set by first setting the mean outage duration and then computing the related mean operating period duration through the equation

$$\mathbb{E}[S] = \mathbb{E}[D]\frac{A_0}{1 - A_0}. \tag{12}$$

As a figure of merit for both tests we consider the Type I error probability, i.e. the probability of rejecting the null hypothesis when it is instead true. In all the cases we have examined, we have simulated, we have set the durations of operating and outage periods according to the target availability, so that the null hypothesis is always true, and the Type I error probability becomes equal to the probability of getting

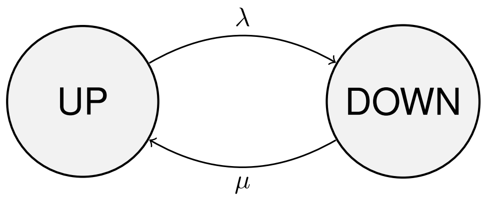

Fig. 1. Alternation of states in a Markov model

a no-compliance statement. For the time being, we do not consider Type II errors.

A. The Markov model

We consider the case where the alternation of states for the cloud is governed by a Markov process, as considered in [28], where the durations of both states follow an exponential distribution. In this case, the cloud switches from the UP state (running) to the DOWN state (outage) at the rate λ, while it reverts to the UP state at the rate μ. The resulting availability is

$$A = \frac{\text{MTTF}}{\text{MTTF} + \text{MTTR}} = \frac{1/\lambda}{1/\lambda + 1/\mu} = \frac{\mu}{\lambda + \mu}. \tag{13}$$

Setting the simulation parameters is quite straightforward, since, after setting the target availability A_0, we simulate the operating periods and the outage periods by drawing pseudorandom numbers from two exponential distributions. The exponential distribution describing the outage period has average value

$$\mathbb{E}[D] = 1/\mu = \text{MTTR}, \tag{14}$$

while that describing the operating period has average value

$$\mathbb{E}[S] = 1/\lambda = \frac{1}{\mu}\frac{A_0}{1 - A_0} = \text{MTTR}\frac{A_0}{1 - A_0}. \tag{15}$$

In Fig. 1 we show the simulation results when the target availability is $A_0 = 0.99$ and we issue the compliance statement over a month. We see that the naive estimator is quite close to flipping an unbiased coin, exhibiting a negligible dispersion (its minimum value is 0.4341 and its maximum values is 0.4848). On the other hand, the bootstrap-based estimator yields an error that is at least 16 times smaller (in many cases even two order of magnitudes smaller) and appears to increase with the average outage duration.

If the cloud provider boasts a higher availability, the problem of correctly estimating it conflicts with the length of the observation period. In fact, the probability of having no outages during a month is 83.5% when the availability is 0.999 and the average outage duration is 4 hours, so that most observation periods will end without any outage to report. In those cases a naive 100% availability estimate would be reported. We can examine the performances of the two estimators in Fig. 3. The curves pertaining to the observation period of 1 month (indicated in the legend as 1m for short) tell us that the naive estimator keeps on its Type I error probability close to 50%, while the performance of the bootstrap-based estimator gets worse as the average outage duration lengthens: as the average outage duration exceeds 4 hours, the Type I error probability of the bootstrap estimator gets practically identical to that of the naive estimator. This is not the fault of the estimator, but rather the consequence of too short an observation period. If we extend the period over which we issue a compliance/no-compliance statement to 1 year (indicated as 1y in the legend), the things turn straight again, as can be seen in Fig. 3: the bootstrap estimator receives as input a number of outage data large enough to get an error probability lower than 1% in most cases, while the performance of the naive estimator remain close to the flipping coin status.

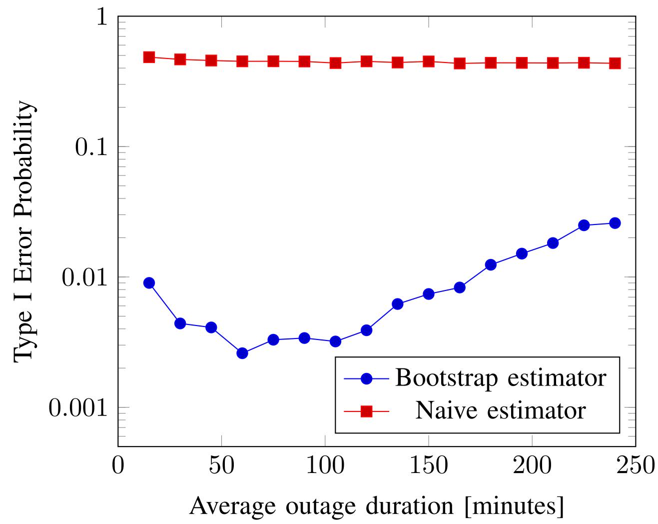

Fig. 2. Type I error probability for the Markov model with 0.99 availability

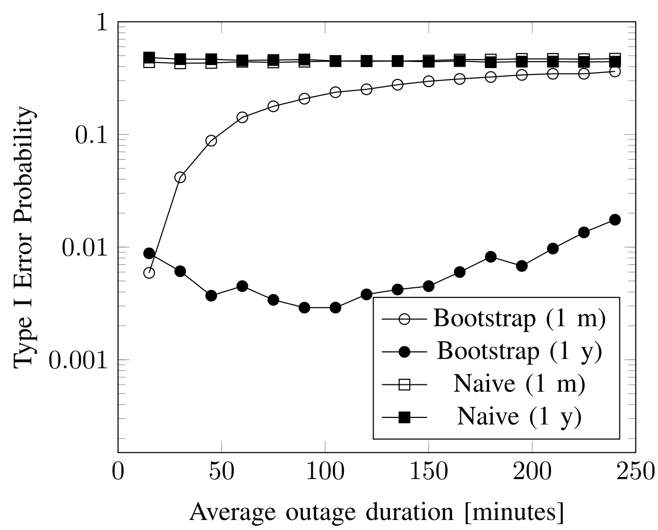

Fig. 3. Type I error probability for the Markov model with 0.999 availability

B. The Poisson-Pareto model

We consider now a different model, which has instead been derived specifically for cloud services. In this case the distribution of the number of outages during the observation interval is described by a Poisson model (so that the duration of the operating period follows an exponential distribution), while the duration of outages follows the Generalized Pareto distribution (GP), hence the Poisson-Pareto name. The cumulative distribution function of the duration of outages is

$$\mathbb{P}[X < x] = F_{\xi,\beta}(x) = \begin{cases} 1 - (1 + \xi x/\beta)^{-1/\xi} & \text{if } \xi \neq 0 \\ 1 - e^{-x/\beta} & \text{if } \xi = 0 \end{cases}, \tag{16}$$

where β is the scale parameter and ξ is the shape parameter.

The Poisson-Pareto model has been derived in [16] on the basis of a dataset of customer-reported outages for five major cloud providers (Google, Amazon, Rackspace, Salesforce, Windows Azure). The sources of data were Cloutage (cloutage.org), founded by the Open Security Foundation in April 2010 but now discontinued, and the International Working Group on Cloud Computing Resiliency (IWGCR, hosted on http://iwgcr.org/), a working group with a mission to monitor and analyze cloud computing resiliency.

The Poisson-Pareto model has been employed in [29] to assess the sustainability of refunds linked to insurance contracts for cloud services, as well as in [25] to assess the contribution of the network to the overall unavailability.

After setting the durations of the operating and outage periods according to the target availability, as described at the beginning of Section IV, we must set the parameters of the distributions governing the durations of the operating and outage periods. As to the exponential distribution for the operating period, we use Equation (15). Instead, for the Pareto distribution, we first set the shape parameter $\xi = 0.4$ on the basis of the data reported in [16] (it was the value reported for Google), and then set the scale parameter as

$$\beta = (1 - \xi)\mathbb{E}[D]. \tag{17}$$

As we have done for the Markov model, we evaluate the performance of both availability estimators by computing the Type I error probability. In Fig. 4 we report the results for the target availability $A_0 = 0.99$ over a month. We observe a similar behaviour as in the case of the Markov model. The naive estimator is quite off the mark, with an average error probability of 0.3794 and minimal dispersion around that value, i.e, practically independent of the average outage duration. Instead the bootstrap estimator exhibits a growing trend with the average outage duration, but guarantees an error probability not larger than 2% in the range examined, roughly twenty times lower than what is achieved by the naive estimator.

If we assume a larger availability figure, i.e. 0.999, we reach the same conclusions as in the case of the Markov model. Examining Fig. 5, we see that a larger availability compels us to issue compliance/no-compliance statements over a longer period, otherwise the possible absence of outage events over a short period (such as a month) would distort our estimate. Turning to a longer observation period strongly reduces the error probability for the bootstrap estimator (the largest figure

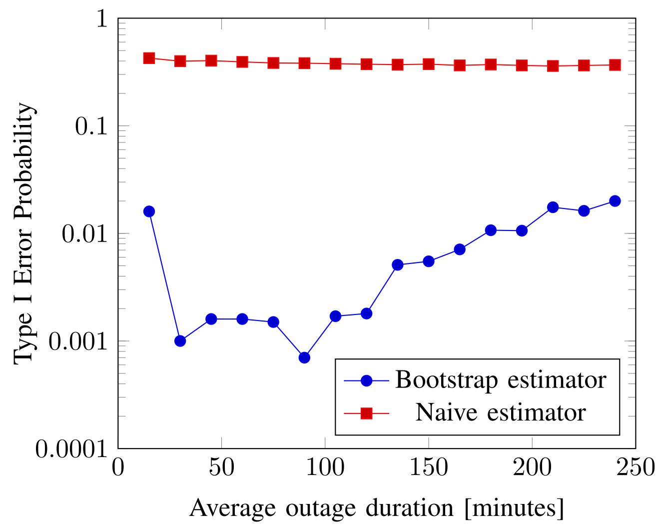

Fig. 4. Type I error probability for the Poisson-Pareto model with 0.99 availability

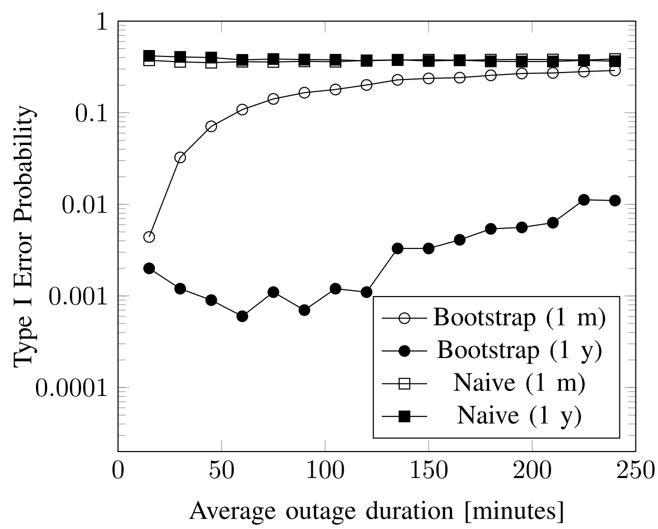

Fig. 5. Type I error probability for the Poisson-Pareto model with 0.999 availability

for 1 year window is now 1.1%), while it has a negligible effect on the performance of the naive estimator.

If we now compare the actual error figures obtained under the Markov and the Poisson-Pareto model (rather than just the trend), we see that there are no appreciable differences. The bootstrap estimator appears to be rather robust with respect to the underlying model (we remark that the bootstrap estimator is non parametric and does not use any assumption about the actual stochastic process governing the alternation of operating and outage periods).

C. The Pareto-Lognormal model

We finally consider a third model, which has been derived from a measurement campaign conducted on an enterprise cloud system with 3 datacenters over an 18 months period, reporting 331 outage events. The model was proposed in [30] on the basis of Anderson-Darling goodness-of-fit tests.

In this model the best-fit distribution for the duration of operating periods is Pareto, while the outage periods appear to be best described by a log-normal model. We have then

$$\mathbb{P}[S < x] = \begin{cases} 1 - \left(\frac{h}{x}\right)^{\alpha} & \text{if } x > h \\ 0 & \text{if } x \leq h \end{cases} \tag{18}$$

for the operating period duration, with α as the shape factor and h as the minimum value, and

$$f_D(x) = \frac{1}{x\sigma\sqrt{2\pi}} e^{-\frac{(\ln x - \mu)^2}{2\sigma^2}}, \tag{19}$$

for the outage duration.

Though the availability as estimated on their enterprise cloud is not reported in [30], we can recover the values MTTF=2349 minutes and MTTR=227 minutes. The former value is simply obtained as a rough estimate by dividing the number of outages by the length of the observation period. The latter value is instead the expected value of the lognormal distribution, when the location parameter is $\mu = 4.58$ and the scale parameter is $\sigma = 1.3$, as indicated in [30]. Those figures lead to 91.1% availability.

Since we wish to carry out a hypothesis test adopting different availability target figures, but want to retain the same statistical properties as estimated by that measurement campaign, we perform our simulation experiment by keeping the same Pareto shape factor and the same lognormal coefficient of variation (i.e. the ratio of the standard deviation to the expected value) as indicated in [30]. We set then $\alpha = 4.94$ for the Pareto model and $\sigma = 1.3$ for the lognormal one, since the coefficient of variation is simply $\mathrm{CV} = \sqrt{e^{\sigma^2} - 1}$.

After setting those parameters, the remaining distribution parameters can easily be derived. In fact, for each value of the average outage duration (i.e., each value of MTTR), we obtain the location parameter of the lognormal distribution as

$$\mu = \ln \mathrm{MTTR} - \frac{\sigma^2}{2}. \tag{20}$$

We can finally recover the minimum value h of the Pareto distribution by first deriving the average duration of the operating period $\mathrm{MTTF} = \mathbf{E}[S]$ through Equation (12) and then obtaining

$$h = \frac{\alpha - 1}{\alpha} \mathrm{MTTF}. \tag{21}$$

As we have done with the previous models, we report the Type I error probability. The curves for the case $A_0 = 0.99$ are shown in Fig. 6. We see again that the naive estimator exhibits poor performance, though quite independent of the average outage duration. The error probability of the bootstrap estimator is instead growing with the average outage duration, but it is nevertheless at least one order of magnitude lower than the naive estimator, and 5.05% at most.

If the target availability requirement is tighter, we see the same phenomenon observed for the other models (see Fig. 7): the error probability of the bootstrap estimator grows heavily when the observation interval is too short (1 month) and becomes fast indistinguishable from what is achieved with the naive estimator. If we get back to a long enough observation interval, the bootstrap estimator regains its advantage by at least one order of magnitude. The error probability achieved in the best cases is anyway a bit larger than what we have obtained under the Markov and the Poisson-Pareto models.

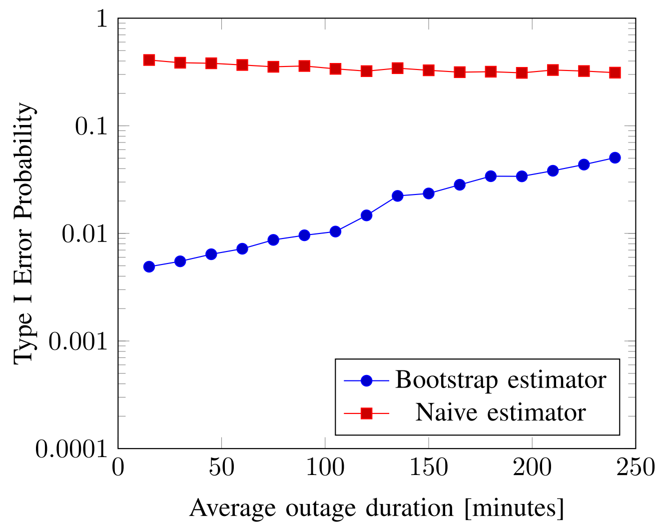

Fig. 6. Type I error probability for the Pareto-Lognormal model with 0.99 availability

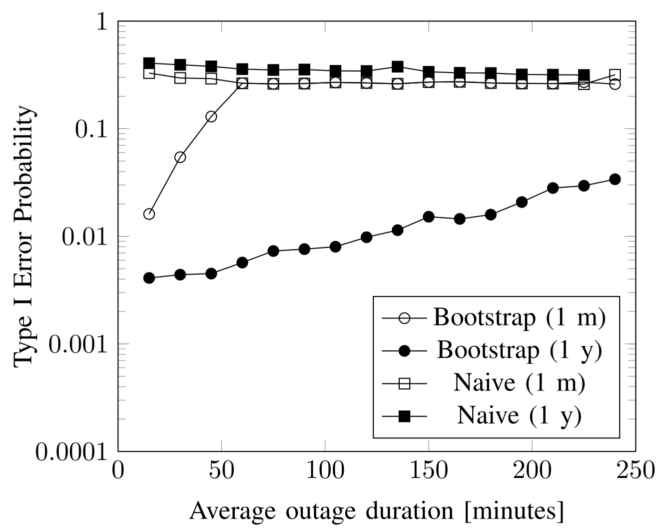

Fig. 7. Type I error probability for the Pareto-Lognormal model with 0.999 availability

V. Conclusion

A new method, based on the bootstrap technique, has been proposed to estimate the availability of a cloud storage service. The method is completely non-parametric, since it does not make any assumption about the stochastic characteristics of the service. The method can be employed in cloud monitoring systems to issue compliance/no-compliance statements concerning service availability through a statistical hypothesis test. While the naive method currently in use in monitoring systems errs roughly 40% of the time, the new bootstrap-based one exhibit a Type I error probability lower than 1% in most cases. The adoption of the bootstrap-based method in cloud monitoring system would allow a far more accurate representation of the actual service quality and avoid false charges of no-compliance in possible disputes between a cloud provider and its customers.

REFERENCES

[1] D. Connor, P. H. Corrigan, J. E. Bagley, and S. S. NOW, "Cloud storage: Adoption, practice and deployment," *An Outlook Report from Storage Strategies NOW*, 2011.

[2] J. Ju, J. Wu, J. Fu, Z. Lin, and J. Zhang, "A survey on cloud storage." *JCP*, vol. 6, no. 8, pp. 1764–1771, 2011.

[3] J. Wang, "Critical factors for personal cloud storage adoption in china," *Journal of Data and Information Science*, vol. 1, no. 2, pp. 60–74, 2016.

[4] C.-S. Wang and S.-L. Lin, "Why are people willing to pay for cloud storage service?" in *Computer and Information Science (ICIS), 2016 IEEE/ACIS 15th International Conference on*. IEEE, 2016, pp. 1–6.

[5] M. Naldi and L. Mastroeni, "Cloud storage pricing: A comparison of current practices," in *Proceedings of the 2013 International Workshop on Hot Topics in Cloud Services*, ser. HotTopiCS '13. New York, NY, USA: ACM, 2013, pp. 27–34.

[6] ——, "Economic decision criteria for the migration to cloud storage," *European Journal of Information Systems*, vol. 25, no. 1, pp. 16–28, 2016.

[7] M. Alhamad, T. Dillon, and E. Chang, "Conceptual SLA framework for cloud computing," in *Digital Ecosystems and Technologies (DEST), 2010 4th IEEE International Conference on*. IEEE, 2010, pp. 606–610.

[8] S. Mubeen, S. A. Asadollah, A. V. Papadopoulos, M. Ashjaei, H. Pei-Breivold, and M. Behnam, "Management of service level agreements for cloud services in iot: A systematic mapping study," *IEEE Access*, 2017.

[9] V. Ravindhren and S. Ravimaran, "Ccmacloud critical metric assessment framework for scientific computing," *Cluster Computing*, pp. 1–11, 2017.

[10] J. OLoughlin and L. Gillam, "A performance brokerage for heterogeneous clouds," *Future Generation Computer Systems*, 2017.

[11] R. Nachiappan, B. Javadi, R. N. Calheiros, and K. M. Matawie, "Cloud storage reliability for big data applications: A state of the art survey," *Journal of Network and Computer Applications*, vol. 97, pp. 35–47, 2017.

[12] B. Liu, X. Chang, Z. Han, K. Trivedi, and R. J. Rodríguez, "Model-based sensitivity analysis of iaas cloud availability," *Future Generation Computer Systems*, 2018.

[13] E. Ataie, R. Entezari-Maleki, L. Rashidi, K. S. Trivedi, D. Ardagna, and A. Movaghar, "Hierarchical stochastic models for performance, availability, and power consumption analysis of iaas clouds," *IEEE Transactions on Cloud Computing*, 2017.

[14] P. T. Endo, G. L. Santos, D. Rosendo, D. M. Gomes, A. Moreira, J. Kelner, D. Sadok, G. E. Gonçalves, and M. Mahloo, "Minimizing and managing cloud failures," *Computer*, vol. 50, no. 11, pp. 86–90, 2017.

[15] M. R. Mesbahi, A. M. Rahmani, and M. Hosseinzadeh, "Highly reliable architecture using the 80/20 rule in cloud computing datacenters," *Future Generation Computer Systems*, vol. 77, pp. 77–86, 2017.

[16] M. Naldi, "The availability of cloud-based services: Is it living up to its promise?" in *Design of Reliable Communication Networks (DRCN), 2013 9th International Conference on the*. IEEE, 2013, pp. 282–289.

[17] Y. Xiaoyong, T. Hongyan, L. Ying, J. Tong, L. Tiancheng, and W. Zhonghai, "A competitive penalty model for availability based cloud sla," in *Cloud computing (CLOUD), 2015 IEEE 8th International Conference on*. IEEE, 2015, pp. 964–970.

[18] M. M. Qiu, Y. Zhou, and C. Wang, "Systematic analysis of public cloud service level agreements and related business values," in *Services Computing (SCC), 2013 IEEE International Conference on*. IEEE, 2013, pp. 729–736.

[19] J. C. Mogul, R. Isaacs, and B. Welch, "Thinking about availability in large service infrastructures," in *Proceedings of the 16th Workshop on Hot Topics in Operating Systems*. ACM, 2017, pp. 12–17.

[20] Y. Xiaoyong, L. Ying, J. Tong, L. Tiancheng, and W. Zhonghai, "An analysis on availability commitment and penalty in cloud sla," in *Computer Software and Applications Conference (COMPSAC), 2015 IEEE 39th Annual*, vol. 2. IEEE, 2015, pp. 914–919.

[21] T. Labidi, A. Mtibaa, W. Gaaloul, S. Tata, and F. Gargouri, "Cloud SLA modeling and monitoring," in *Services Computing (SCC), 2017 IEEE International Conference on*. IEEE, 2017, pp. 338–345.

[22] S. Alboghdady, S. Winter, A. Taha, H. Zhang, and N. Suri, "C'mon: Monitoring the compliance of cloud services to contracted properties," in *Proceedings of the 12th International Conference on Availability, Reliability and Security*. ACM, 2017, p. 36.

[23] Z. Hu, L. Zhu, C. Ardi, E. Katz-Bassett, H. V. Madhyastha, J. Heidemann, and M. Yu, "The need for end-to-end evaluation of cloud availability," in *International Conference on Passive and Active Network Measurement*. Springer, 2014, pp. 119–130.

[24] M. Naldi, "Accuracy of third-party cloud availability estimation through icmp," in *Telecommunications and Signal Processing (TSP), 2016 39th International Conference on*. IEEE, 2016, pp. 40–43.

[25] ——, "ICMP-based third-party estimation of cloud availability," *International Journal of Advances in Telecommunications, Electrotechnics, Signals and Systems*, vol. 6, no. 1, pp. 11–18, 2017.

[26] E. Casalicchio and L. Silvestri, "Mechanisms for SLA provisioning in cloud-based service providers," *Computer Networks*, vol. 57, no. 3, pp. 795–810, 2013.

[27] B. Efron and R. J. Tibshirani, *An introduction to the bootstrap*. CRC press, 1994.

[28] L. Mastroeni and M. Naldi, "Network protection through insurance: Premium computation for the on-off service model," in *8th International Workshop on the Design of Reliable Communication Networks, DRCN 2011, Krakow, Poland, 10-12 October, 2011*, 2011, pp. 46–53.

[29] ——, "Insurance pricing and refund sustainability for cloud outages," in *Economics of Grids, Clouds, Systems, and Services*, C. Pham, J. Altmann, and J. Á. Bañares, Eds. Cham: Springer International Publishing, 2017, pp. 3–17.

[30] J. Dunne and D. Malone, "Obscured by the cloud: A resource allocation framework to model cloud outage events," *Journal of Systems and Software*, vol. 131, pp. 218–229, 2017.