International Journal of Advanced Information Science and Technology (IJAIST) ISSN: 2319:2682
Vol.5, No.4, April 2016 DOI:10.15693/ijaist.2016.v5.i4.9-18

What is aminimum ofUnpredictable Workload Pattern over allElastic Scaling in CloudComputing?

Ravi (Ravinder) Prakash G
Senior Professor Research,
BMS Institute of Technology
Dodaballapur Road,
Avalahalli, Yelahanka,
Bangalore – 560 064

Kiran M
Research Scholar
School of Computing &
Information Technology
REVA University, Yelahanka,
Bengaluru - 560064

***Abstract*: Measurability is a concept in elastic scaling that is based on two assumptions: (1) every cloud service provider is cautious, i.e., does not exclude any cloud consumer's Unpredictable Workloadresource pooling pattern choice from consideration, and (2) every cloud service provider respects the cloud consumer's Unpredictable Workloadresource pooling pattern preferences, i.e., deems one cloud consumer's Unpredictable Workloadresource pooling pattern choice to be infinitely more likely than another whenever it premises the cloud consumer to prefer the one to the other. In this paper we provide a new approach for measurability, by assuming that cloud service providers have asymmetric Unpredictable Workloadresource pooling pattern about the cloud consumer's Unpredictable Workloadutilities. We show that, if the uncertainty of each cloud service provider about the cloud consumer's Unpredictable Workloadutilities vanishes gradually in some regular manner, then the Unpredictable Workloadresource pooling pattern choices it can measurably make under common conjecture in measurability are all actually measureable in the original elastic scaling with no uncertainty about the cloud consumer's utilities.**

***Index terms:* Cloud service provider, cloud consumer, Unpredictable Workload, asymmetric, resource pooling pattern, utilities, elastic scaling, behavioral, measurably**

1. INTRODUCTION

Elastic scalingdeals with the ways the cloud service providers may reason about its cloud consumers before making a decision. More precisely, in elastic scalingcloud service providers base its Unpredictable Workloadresource pooling pattern choices on the conjectures about the cloud consumers' behavior, which in turn depend on its conjectures about the cloud consumers' conjectures about other cloud consumers' behavior, and so on [1][7][9] [21]. A major goal of elastic scalingin this work is to study such conjecture hierarchies, to impose reasonable conditions on these, and to investigate its resource pooling pattern behavioral implications.

A central idea in elastic scaling is common conjecture in measurability, stating that a cloud service providerpremises that its cloud consumers choose measurably, and so on. In our view, one of its most natural refinements is the concept of measurability. Measurabilityis based on the following two conditions: The first states thatcloud service providers are cautious [2][8][10] [22], meaning that they do not exclude any cloud consumers' Unpredictable Workloadresource pooling pattern choice from consideration. The second condition states that whenever premise that a Unpredictable Workloadresource pooling pattern choice$a$is better than another Unpredictable Workloadresource pooling pattern choice$b$ for acloud consumer, then the probability assign to$b$must be at most α times the probability assign to$a$. Under α-measurability there is common conjecture in the event that every cloud service provider is cautious and satisfies theα-actual trembling condition. AUnpredictable Workloadresource pooling pattern choice is called actually measureable if it can be chosen under α-measurability for every $\alpha > 0$[3] [11] [15] [20].

2. RESEARCH CLARIFICATION

The usual interpretation ofmeasurability assumes that cloud consumer makes mistakes, but that deem more costly mistakes much less likely than less costly mistakes. In this paper we offer a rather different approach for measurability. Instead of assuming premisecloud consumer to make mistakes, we rather suppose that have uncertainty about its utility function, while believing that it chooses measurably. We thus consider aelastic scalingwith asymmetric Unpredictable Workloadresource pooling pattern. Our result states that, if we let uncertainty about the cloud consumer's utility go to zero in some regular manner, then every Unpredictable Workloadresource pooling pattern choice that can measurably be made under common conjecture in measurability in theelastic scalingwith asymmetric Unpredictable Workloadresource pooling pattern, will be actually measureable in the original elastic scaling, in which there is no uncertainty about the cloud consumer's utilities.

In the elastic scalingwith asymmetric Unpredictable Workloadresource pooling pattern, we impose some regularity conditions on the cloud service providers' conjectures about the cloud consumer's utility functions which can be summarized as follows: First, for every outcome in the elastic scaling, the conjecture that cloud service provider$i$has about cloud service provider$j$'s utility from this outcome, is always normally distributed with its mean at the "original" utility in the original elastic scaling. As a consequence, cloud service provider$i$deems any utility function possible for cloud service provider$j$, and hence every resource pooling pattern choice for cloud service provider$j$can be optimal for some utility function deemed possible by i. Together with the condition that $i$premises in j's measurability, this actually makes sure that cloud service provider$i$deems every Unpredictable Workloadresource pooling pattern choice possible for cloud service provider$j$, thus mimicking the cautiousness condition described above. Secondly,i's conjecture about j's utility function should be independent from its conjecture about j's conjecture hierarchy. This makes intuitive sense since j's conjecture hierarchy is analytic property of this cloud service provider, whereas its utility function is notanalytic property [4][12][16] [23]. Therefore there is no obvious reason to expect any correlation between these two characteristics. Thirdly, i's conjecture about j's utilities from different outcomes in the elastic scalingshould be independent from each other. Possibly some of these conditions can be relaxed for the proof of our result, but we leave this issue for future research.

The paper is organized as follows: In Section 3 we introduce our elastic scalingmodel[5][13][17] [24]for elastic scaling with asymmetric Unpredictable Workloadresource pooling pattern, we formalize the idea of common conjecture in measurability for these elastic scaling, and show that common conjecture in measurability is always possible (Descriptive Study I). In Section 4 we introduce our elastic scalingmodel for elastic scalingwithsymmetric Unpredictable Workloadresource pooling pattern, and present the concept of measurability for these elastic scaling (Prescriptive Study). In Section 5 we state our result, establishing the connection between common conjecture in measurability in the elastic scalingwith asymmetric Unpredictable Workloadresource pooling patternin the presence of small uncertainty about the cloud consumer's utility function, and measurability in the original elastic scaling (Descriptive Study II). In Section 6 we provide some concluding remarks. All proofs are collected in Section 7.

3. DESCRIPTIVE STUDY I

3.1 Elastic Model

Throughout this paper we restrict attention to elastic scaling operations with two sets of cloud service provider. Let $\delta = (C_i, w_i)_{i\epsilon I}$be a finite, Unpredictable Workload where$I = \{1, 2\}$is the set of cloud service providers,C_i is the finite set of Unpredictable Workloadresource pooling pattern choices of cloud service provider$i$, $w_i$is cloud service provider$i$'s utility function. The function $w_i$assigns to every pair of Unpredictable Workloadresource pooling pattern choice$(c_1, c_2) \epsilon C_1 \times C_2$a utility $w_i(c_1, c_2) \epsilon F$.

In a elastic scalingwith asymmetric Unpredictable Workloadresource pooling pattern,cloud service providers do not only uncertainty about the cloud consumer's Unpredictable Workloadresource pooling pattern choices; they also have uncertainty about the cloud consumer's utility function. Hence a conjecture hierarchy should not only specify what the cloud service provider premises about the cloud consumer's Unpredictable Workloadresource pooling pattern choice but also what it premises about the cloud consumer's utility function. Not only this, it should also specify what the cloud service provider premises about the cloud consumer's conjecture about its own Unpredictable Workloadresource pooling pattern choice and utility function, and so on. A possible way of modelingsuch conjecture hierarchies is by means of the following necessary and sufficient condition.

Necessary and sufficient condition3.1(*elastic scalingmodel*). A finite elastic scalingmodel for $\delta$with asymmetric Unpredictable Workloadresource pooling pattern is a tuple$M = (S_i, v_i, K_i)_{i\epsilon I}$where (1) $S_i$is the set of Unpredictable Workload types for cloud service provider$i$. (2)$v_i: S_i \rightarrow \theta(C_j \times S_j)$is the conjecture assignment taking only finitely many different probability distributions on $\theta(C_j \times S_j)$and (3) $k_i$is the utility assignment that assigns to every $s_i \epsilon S_i$a utility function$k_i(s_i): C_1 \times C_2 \rightarrow F$. By$\theta(P)$we denote the set of probability distributions on$P$.Therefore, in aelastic scalingmodel, each Unpredictable Workload type$s_i$has a conjecture about cloud service provider$j$'s resource pooling pattern choice-Unpredictable Workload type combinations. And hence, inparticular, it has a conjecture about j's resource pooling pattern choice. But, as cloud service provider$j$'s Unpredictable Workload type also specifies its utility function and its conjecture about i's resource pooling pattern choice, cloud service provider$i$ also has some conjecture about cloud service provider$j$'s utility function, and about cloud service provider$j$'s conjecture about its own resource pooling pattern choice, and so on. In this way one can derive a complete conjecture hierarchy for every given Unpredictable Workload type.

Note that each Unpredictable Workload type$s_i$can be identified with a pair$\big(k_i(s_i), v_i(s_i)\big)$where$k_i(s_i)$is its utility function and $v_i(s_i)$is its conjecture hierarchy. Since we required the conjecture assignment to take only finitely many different probability distributions, the elastic scalingmodel contains only finitely many different conjecture hierarchies.

3.2 Limitations on the Elastic Model

Our goal will be to model the situation where the cloud service providers have uncertainty about the cloud consumer's utility function, but where this uncertainty "vanishes in the limit". In order to formalize this we need to impose additional limitations on the elastic scaling-model.

Recall that every Unpredictable Workload type$s_i$can be identified with a pair $\big(k_i(s_i), v_i(s_i)\big)$, where $k_i(s_i)$is s_i'sutility function and $v_i(s_i)$is its conjecture hierarchy. Denote by $K_i$the set of all possible utility functions, and by $V_i$the set of all conjecture hierarchies in the elastic scalingmodel$M = (S_i, k_i, v_i)_{i\epsilon I}$.The first condition we impose is that $S_i = K_i \times V_i$, that is, for every possible utility function we can think of, and every conjecture hierarchy in the model, there exists a Unpredictable Workload type in the model with exactly this combination of utility function and conjecture hierarchy. Therefore in a sense we assume that the Unpredictable Workload typeis rich enough.

Secondly, we assume that s_i'sconjecture about j's utilityfrom $(c_1, c_2)$is statistically independent from its conjecture j's utility from$(\acute{c}_1, \acute{c}_2)$whenever $(c_1, c_2) \neq (\acute{c}_1, \acute{c}_2)$and that this conjecture is also statistically independent from its conjecture about j's conjecture hierarchy.

Finally we assume that s_i'sconjectures about j's utilities from the various outcomes in the elastic scaling are all induced by a unique normal distribution. More formally,s_i'sconjecture about j's utility from $(c_1, c_2)$is given by a normal distribution with its mean at $w_j(c_1, c_2)$– the "true" utility of cloud service provider$j$ in the original elastic scaling. Therefore, all these conjectures are distributed identically around the mean. By collecting all these conditions we arrive at the following necessary and sufficient condition.

Necessary and sufficient condition 3.2*(σ–regular elastic scalingmodel)*. Let Dbe the normal distribution on $F$with mean 0 and variance $\sigma^2 > 0$. Then a elastic scalingmodel$M = (S_i, k_i, v_i)_{i\epsilon I}$is σ–regular if for both cloud service providers i, (1) $S_i = K_i \times V_i$, (2) for every Unpredictable Workload type$s_i \epsilon S_i$, its conjecture about j's utility from $(c_1, c_2)$is statistically independent from its conjecture about j's utility from $(\acute{c}_1, \acute{c}_2)$whenever $(c_1, c_2) \neq (\acute{c}_1, \acute{c}_2)$and its conjecture about j's utilities is statisticallyindependent from its conjecture about j's conjecture hierarchy, and (3) for every Unpredictable Workload type$s_i \epsilon S_i$,and every resource pooling pattern choice-pair (c_1, c_2), the conjecture of $s_i$about j's utility from$(c_1, c_2)$is given by D, upto a shift of the mean to $w_j(c_1, c_2)$.

3.3 σ-Measurability

In this subsection we will define common conjecture in measurability inside aelastic scalingmodel with asymmetric

Unpredictable Workloadresource pooling pattern. In addition, if we require the elastic scaling-model to be σ-regular for a given normal distribution with mean 0 and variance σ^2, then we obtain the concept of σ-measurability. We first need some more notations. For given Unpredictable Workload type$s_i$and Unpredictable Workloadresource pooling pattern choicec_i, let $k_i(s_i)(c_i)$be the expected utility for Unpredictable Workload type$s_i$from choosing c_i, given its conjecture $v_i(s_i)$about the cloud consumer's Unpredictable Workloadresource pooling pattern choice, and given its utility function $k_i(s_i)$.

Necessary and sufficient condition 3.3 *(MeasureableUnpredictable Workloadresource pooling choice).* A Unpredictable Workloadresource pooling pattern choice$c_i$is measureable for $s_i$if $k_i(s_i)(c_i) \geq k_i(s_i)(\acute{c}_i)$for all $\acute{c}_i \in C_i$.

We will now define common conjecture in measurability. In words it says that a cloud service provider premises that its cloud consumer makes measureableUnpredictable Workloadresource pooling pattern choices, and premises that its cloud consumer premises that it makes measureableUnpredictable Workloadresource pooling pattern choices, and so on [25].

Formally, for every $\widehat{S_i} \subseteq S_i$, let

$$(C_i \times \widehat{S_i})^{quant} = \{(c_i, s_i) \epsilon C_i \times \widehat{S_i} : c_i \text{is measureable for } s_i\}.$$

Necessary and sufficient condition 3.4 *(Common conjecture in Measurability).* For cloud service providers$i$ we define subsets of Unpredictable Workload types$S_i^1, S_i^2, ...$in a recursive way as follows:

$$S_i^1 := \left\{s_i \in S_i : v_i(s_i)\left[\left(C_j \times S_j\right)^{quant}\right] = 1\right\},$$

$$S_i^2 := \left\{s_i \in S_i : v_i(s_i)\left[\left(C_j \times S_j^1\right)^{quant}\right] = 1\right\},$$

.

.

.

$$S_i^l := \left\{s_i \in S_i : v_i(s_i)\left[\left(C_j \times S_j^{l-1}\right)^{quant}\right] = 1\right\},$$

.

.

.

Unpredictable Workload type$s_i$expresses common conjecture in measurability if $s_i \in \cap_{l \in \mathbb{N}} S_i^l$.AUnpredictable Workload typeσ–measureable if it expresses commonconjecture in measurability with a σ–regular elastic scalingmodel.

Necessary and sufficient condition 3.5 *(σ–measureableUnpredictable Workload type).* Let $M = (S_i, v_i, k_i)_{i \epsilon I}$be a σ–regular elastic scalingmodel. Every Unpredictable Workload type$s_i \in S_i$that expresses common conjecture in measurability is calledσ–measureable.

Now we show that σ–measureableUnpredictable Workload types always exist.

Proposition 3.1 (σ–measureableUnpredictable Workload types always exist): Consider a finite Unpredictable Workload$\delta = (C_i, w_i)_{i \epsilon I}$, and some $\sigma > 0$. Then there is a σ–regular elastic scalingmodel$M = (S_i, v_i, k_i)_{i \epsilon I}$for $\delta$where all Unpredictable Workload types are σ–measureable. The proof can be found in Section 7.

3.4 Limit Measurability

In this subsection we focus on those Unpredictable Workloadresource pooling pattern choices, which can measurably be made under common conjecture in measurability when the uncertainty about the cloud consumer's utility vanishes. This will lead to the concept of limit measurability. We first need an additional necessary and sufficient condition.

Necessary and sufficient condition 3.6 *(Constant Unpredictable Workload typeand utility assignments).* A Unpredictable Workload sequence of elastic scalingmodels$((S_i^n, v_i^n, k_i^n)_{i \epsilon I})_{n \epsilon \mathbb{N}}$has constant Unpredictable Workload type and utility assignments if $S_i^n = S_i^m$and $k_i^n = k_i^m$for all $n$and m, and for cloud service providers$i$.We are now ready to say the concept of limit measureableUnpredictable Workloadresource pooling pattern choice.

Necessary and sufficient condition 3.7 *(Limit measureable resource pooling pattern choice).* Consider a finite Unpredictable Workload$\delta = (C_i, w_i)_{i \epsilon I}$with cloud service providers. A Unpredictable Workloadresource pooling pattern choice$c_i$is limit measureable if there is a Unpredictable Workload sequence$(\sigma_n)_{n \epsilon \mathbb{N}} \to 0$,and a Unpredictable Workload sequence$(M^n)_{n \epsilon \mathbb{N}}$of σ_n–regular elastic scalingmodels with constant Unpredictable Workload type and utility assignments, such that in every M^nthere is a σ_n-measureableUnpredictable Workload type$s_i^n$with utility function w_i, for which Unpredictable Workloadresource pooling pattern choice$c_i$is optimal.

4. PRESCRIPTIVE STUDY

4.1 Elastic Model

Let $\delta = (C_i, w_i)_{i \epsilon I}$be a finite, Unpredictable Workload with cloud service providers. In aelastic scalingwithsymmetric Unpredictable Workloadresource pooling patterncloud service providers do not have uncertainty about the cloud consumer's utility function. Therefore a conjecture hierarchy only needs to specify what a cloud service provider premises about the cloud consumer's Unpredictable Workloadresource pooling pattern choice, what it premises about the cloud consumer's conjecture about its own Unpredictable Workloadresource pooling pattern choice, and so on. Therefore the elastic scalingmodel will be simpler compared to the case of asymmetric Unpredictable Workloadresource pooling pattern.

Necessary and sufficient condition 4.1 *(elastic scalingmodel).* A elastic scalingmodel forδwithsymmetric Unpredictable Workloadresource pooling patternis a tuple $M = (\Omega_i, \rho_i)_{i \epsilon I}$where (1)$\Omega_i$is the finite set of Unpredictable Workload types for cloud service provider$i$, and (2) $\rho_i : \Omega_i \to \theta(C_j \times \Omega_j)$is the conjecture assignment.

Therefore, in aelastic scalingmodel, each Unpredictable Workload type$\tau_i$has a conjecture about cloud service provider$j$'s Unpredictable Workloadresource pooling pattern choice-Unpredictable Workload type combinations. And hence, in particular, it has a conjecture about j's Unpredictable Workloadresource pooling pattern choice. But, as cloud service provider$j$'s Unpredictable Workload type also specifies its conjecture about cloud service provider$i$'s Unpredictable Workloadresource pooling pattern choice, cloud service provider$i$also has some conjecture about cloud service provider$j$'s conjecture about its own Unpredictable Workloadresource pooling pattern choice, and so on. In this way one can derive a complete conjecture hierarchy for every given Unpredictable Workload type.

For given Unpredictable Workload type$\tau_i$and Unpredictable Workloadresource pooling pattern choice$c_i$we define $w_i(c_i,\tau_i)$as the expected utility for Unpredictable Workload type$\tau_i$from choosing$c_i$given its conjecture $\rho_i(\tau_i)$about its cloud consumer's Unpredictable Workloadresource pooling pattern choice (and *given* its "fixed" utility function w_i). Unpredictable Workload type$\tau_i$is said to *prefer*Unpredictable Workloadresource pooling pattern choicec_i to Unpredictable Workloadresource pooling pattern choice$\acute{c}_i$ when $w_i(c_i,\tau_i) > w_i(\acute{c}_i,\tau_i)$. We say that a Unpredictable Workload typeτ_i*considers possible* some cloud consumer's Unpredictable Workload typeτ_i if $\rho_i(\tau_i)(c_j,\tau_j) > 0$for some $c_j \in C_j$.Now we introduce the key condition in measurability, which is the α–actual trembling condition. Intuitively it says that (1) a cloud service provider should deem possible all cloud consumer's Unpredictable Workloadresource pooling pattern choices, and (2) if a cloud service provider premises Unpredictable Workloadresource pooling pattern choice$a$is better than Unpredictable Workloadresource pooling pattern choice$b$for the other cloud service provider, then it should deem Unpredictable Workloadresource pooling pattern choice$a$much more likely than Unpredictable Workloadresource pooling pattern choice$b$.

Necessary and sufficient condition 4.2 *(α-actual trembling condition)*: Let $\alpha > 0$. A Unpredictable Workload type$\tau_i$satisfies the α-actual trembling condition if (1) for eachτ_j that$\tau_i$deems possible, $\rho_i(\tau_i)(c_j,\tau_j) > 0$for all $c_j \in C_j$,and (2) for every τ_j that $\tau_i$deems possible, wheneverτ_j *prefers*c_j to $\acute{c}_j$, then$\rho_i(\tau_i)(\acute{c}_j,\tau_j) \leq \alpha \cdot \rho_i(\tau_i)(c_j,\tau_j)$.

Therefore, the first condition says that whenever$\tau_i$deems some Unpredictable Workload typeτ_j possible,$\tau_i$also assumes every Unpredictable Workloadresource pooling pattern choice is possible forτ_j.Measurability is based on the event that the Unpredictable Workload types should not only satisfy the α-actual trembling condition themselves, but also express common conjecture in the event that Unpredictable Workload types satisfy the α-actual trembling condition.

Necessary and sufficient condition 4.3 *(α-actually measureableUnpredictable Workload type)*. A Unpredictable Workload type$\tau_i$is α-actually measureable if:$\tau_i$satisfies the α-actual trembling condition,$\tau_i$only deems possible cloud consumer's Unpredictable Workload typesτ_j which satisfy the α-actual trembling condition, $\tau_i$only deems possible cloud consumer's Unpredictable Workload typesτ_j which only deem possible cloud service provider$i$'s Unpredictable Workload typesτ_i'which satisfy the α-actual trembling condition, and so on.Actually measureableUnpredictable Workloadresource pooling pattern choices are those Unpredictable Workloadresource pooling pattern choices, which can measurably be made by α-actually measureableUnpredictable Workload typesfor all α.

Necessary and sufficient condition 4.4 *(Actually measureable resource pooling pattern choice)*. A Unpredictable Workloadresource pooling pattern choice$c_i$is α-actually measureable if there is a elastic scalingmodel and aα-actually measureableUnpredictable Workload type$\tau_i$within it for which$c_i$is optimal. AUnpredictable Workloadresource pooling pattern choice$c_i$is actually measureable if it is α-actually measureable for all$\alpha > 0$.

5. DESCRIPTIVE STUDY II

5.1 Statement of the result

For aUnpredictable Workloadwe analyzed two contexts, one with asymmetric Unpredictable Workloadresource pooling pattern and another withsymmetric Unpredictable Workloadresource pooling pattern. In the context with asymmetric Unpredictable Workloadresource pooling pattern, where cloud service providers have uncertainty about the cloud consumer's utility, we introduced the concept of a limit measureableUnpredictable Workloadresource pooling pattern choice. In the context withsymmetric Unpredictable Workloadresource pooling pattern, where cloud service providers have no uncertainty about the cloud consumer's utility, we discussed the concept of a actually measureableUnpredictable Workloadresource pooling pattern choice. In our result we connect these two concepts.

Proposition 5.1 *(Limit Measurability implies Measurability)*: Consider a finite Unpredictable Workloadwithcloud service providers. Every limit measureableUnpredictable Workloadresource pooling pattern choice for the context with asymmetric Unpredictable Workloadresource pooling pattern is a actually measureableUnpredictable Workloadresource pooling pattern choice for the context withsymmetric Unpredictable Workloadresource pooling pattern.

5.2 Illustration of the result

By means of an example we provide some intuition for our result. More precisely we show how a measureableUnpredictable Workload type in the context of asymmetric Unpredictable Workloadresource pooling pattern can be transformed into anactually measureableUnpredictable Workload type in the context of symmetric Unpredictable Workloadresource pooling pattern. Also we show that whenσgoes to zero then ϵgoes to zero as well.Let us start with the context of asymmetric Unpredictable Workload resource pooling pattern. Let Dbe the normal distribution with mean 0 and varianceσ^2.From the proof of Proposition 3.1 we know that there exists a regular elastic scalingmodel$M = (S_i,v_i,k_i)_{i\epsilon I}$where every Unpredictable Workload type is measureable and all the Unpredictable Workload types have the same conjecture hierarchy. Therefore,Unpredictable Workload types onlydiffer by their utility function. For each of the Unpredictable Workload types s_1 of cloud service provider 1 we denote by $\rho_1$the conjecture about cloud service provider 2's Unpredictable Workloadresource pooling pattern choice, and for each Unpredictable Workload type$s_1$let $\rho_1$be the conjecture about cloud service provider 1's Unpredictable Workloadresource pooling pattern choice. As we assume that all the Unpredictable Workload types have the same conjecture hierarchy, $\rho_1$and $\rho_2$are unique.

For both cloud service providers$i$ let $O_i$be the probability distribution on cloud service provider$i$'s utility functions generated by D. Since the elastic scaling-model is σ-regular every Unpredictable Workload types_j has the conjecture $O_i$about i's utility function. Let$K_i(c_i,\rho_i)$be the set of utility functions for cloud service provider$i$ such that the Unpredictable Workloadresource pooling pattern choice$c_i$is optimal under the conjecture $\rho_i$about the cloud consumer's Unpredictable Workloadresource pooling pattern choice. Since every Unpredictable Workload type$s_i$expresses common conjecture in measurability, the probability it assigns to acloud consumer's Unpredictable Workloadresource pooling pattern choicec_j is exactly the probability it assigns to the event that j's utility function is in $K_j(c_j, \rho_j)$which is $O_j\left(K_j(c_j, \rho_j)\right)$.

Since Dhas full support, it follows that all these probabilities are positive. Now we turn to the context of symmetricUnpredictable Workloadresource pooling pattern. We construct aelastic scalingmodel with a single Unpredictable Workload type$\tau_1$for cloud service provider 1 and a single Unpredictable Workload type$\tau_2$for cloud service provider 2. Let the conjecture of $\tau_1$about the cloud service provider 2's Unpredictable Workloadresource pooling pattern choice be given by the $\rho_1$constructed above, and similarly for the conjecture of τ_2. Therefore, the conjecture about the cloud consumer's Unpredictable Workloadresource pooling pattern choice has not changed by moving from the context with asymmetric Unpredictable Workloadresource pooling pattern to the context with symmetric Unpredictable Workloadresource pooling pattern.

6. CONCLUDING REMARKS

We premise that measurability is a very natural concept in elastic scaling, but it has not yet received the attention it deserves. In this paper we have established a new approach for measurability from the viewpoint of elastic scaling withasymmetric Unpredictable Workloadresource pooling pattern. In elastic scalingwith asymmetric Unpredictable Workloadresource pooling pattern we define aUnpredictable Workloadresource pooling pattern choice as limit measureable if it can measurably be made under common conjecture of measurability when the uncertainty vanishes gradually in some regular way. We show the existence of such Unpredictable Workloadresource pooling pattern choices. We then prove that each limit measureableUnpredictable Workloadresource pooling pattern choice in the elastic scalingwith asymmetric Unpredictable Workloadresource pooling pattern is actually measureable for the context with symmetric Unpredictable Workloadresource pooling pattern.

7. PROOFS

7.1 Existence of MeasureableUnpredictable Workload types

We prove Proposition 3.1, which guarantees the existence of σ-measureableUnpredictable Workload types. Consider a finite Unpredictable Workload $M = (C_i, w_i)_{i\epsilon I}$and, some $\sigma > 0$. Let D be the normal distribution with mean 0 and variance σ^2. In fact we will construct a σ-regular elastic scalingmodel where all Unpredictable Workload types of cloud service provider 1 have the same conjecture $\rho_2$about cloud service provider 2's Unpredictable Workloadresource pooling pattern choice and all Unpredictable Workload types of cloud service provider 2 have the same conjecture $\rho_1$about cloud service provider 1's Unpredictable Workloadresource pooling pattern choice. We construct $\rho_1$and $\rho_2$by means of the fixed key of some correspondence.

For every conjecture $\rho_j \in \theta(C_j)$and every utility function w_i, we define

$$C_i(\rho_j, w_i) := \{C_i \in C_i : w_i(c_j, \rho_j) \geq w_i(\acute{c}_i, \rho_j) \text{for all } \acute{c}_i\}.$$

We also define $O_i$as the probability distribution on the set of utility functions of cloud service provider$i$ induced by D. For every $\rho_j \in \theta(C_j)$we define

$$G_i(\rho_j) := \{\rho_i \in \theta(C_j) : \rho_i = \int_{w_i \in K_i} \phi_i(x_i)\ dO_i,$$

where$\phi_i(x_i) \in \left(C_i(\rho_j, x_i)\right)$for every $x_i \in K_i\}$.

Here $K_i$denotes the set of all possible utility functions for cloud service provider$i$. Therefore every $\rho_i \in G_i(\rho_j)$is obtained by taking for every utility function x_ia randomization over optimal Unpredictable Workloadresource pooling pattern choices against $\rho_j$and then taking the expected randomization with respect to O_i. Now we define a correspondence $G$from $\theta(C_1) \times \theta(C_2)$to $\theta(C_1) \times \theta(C_2)$by

$$G(\rho_1, \rho_2) := G_1(\rho_2) \times G_2(\rho_1).$$

Now we use fixed keyposition to prove that Ghas a fixed key. Clearly $G$is upper hemi-continuous and compact valued. We show that G is convex valued. For this it is sufficient to show that $G_1$and G_2 are convex valued. For a given ρ_2, take $\rho_1^{'}$, $\rho_1^{''}$ in (ρ_2). We show that $\psi\rho_1^{'} + (1-\psi)\rho_1^{''}$ is also in $G_1(\rho_2)$. By definition

$$\rho_1^{'} = \int_{x_1} \phi_1^{'}(x_1)\, dO_1 \text{and } \rho_1^{''} = \int_{x_1} \phi_1^{''}(x_1)\ dO_1$$

where$\phi_1^{'}(x_1)$,$\phi_1^{''}(x_1) \epsilon\ \theta\left(C_1(\rho_2, x_1)\right)$for every$x_1$. Therefore we have

$$\psi\rho_1^{'} + (1-\psi)\rho_1^{''} = \int_{x_1} (\psi\phi_1^{'}(x_1) + (1-\psi)\phi_1^{''}(x_1))\, dO_1$$

where$\psi\phi_1^{'}(x_1) + (1-\psi)\phi_1^{''}(x_1) \epsilon\ \theta\left(C_1(\rho_2, x_1)\right)$ for every x_1. Hence by definition $\psi\rho_1^{'} + (1-\psi)\rho_1^{''} \epsilon\ G_1(\rho_2)$.This implies that $G_1$is convex valued. The same applies to $G_2$and hence we can conclude that $G$is convex valued. Now using fixed keyposition$G$has a fixed key(ρ_1^*, ρ_1^*).

Since $\rho_1^* \epsilon\ G_1(\rho_2^*)$it follows that

$$\rho_1^* = \int_{x_1} \phi_1^*(x_1)\ dO_1$$

where$\phi_1^*(x_1) \epsilon\ \theta\left(C_1(\rho_2^*, x_1)\right)$for every x_1. Similarly

$$\rho_2^* = \int_{x_2} \phi_2^*(x_2)\ dO_2$$

where$\phi_2^*(x_2) \epsilon\ \theta\left(C_2(\rho_1^*, x_2)\right)$for every x_2.

We will now construct aelastic scalingmodel$M = (S_i, v_i, k_i)_{i\epsilon I}$. For both cloud service providers i, define

$$S_i = \{s_i{}^{x_i} : x_i \in K_i\}.$$

Let the utility assignment $k_i$be given by

$$k_i(s_i{}^{x_i}) = x_i$$

for every $s_i{}^{x_i} \in S_i$. In order to define the conjecture assignment $v_i$we first define for every Unpredictable Workload type$s_i{}^{x_i}$a density function $v_i^{\sim}(s_i{}^{x_i})$on$C_j \times S_j$as follows:$v_i^{\sim}(s_i{}^{x_i})(c_j, s_j{}^{x_j}) := \phi_j^*(x_j)(c_j)$,where$\phi_j^*(x_j)(c_j)$is the probability that probability distribution $\phi_j^*(x_j)$assigns to c_j. For every Unpredictable Workload type$s_i{}^{x_i}$let $v_i(s_i{}^{x_i}) \in \theta(C_j \times S_j)$ be the probability distribution induced by density function $v_i^{\sim}(s_i{}^{x_i})(c_j, s_j{}^{x_j})$and the probability distribution $Q_j$on K_j. That is, for every set of Unpredictable Workload types $H \subseteq S_j$given by

$$H := \{s_j{}^{x_j} : x_j \epsilon\ G\}$$

We have that

$$v_i(s_i{}^{x_i})(\{c_j\} \times H) := \int_{x_j \in G} v_i^{\sim}(s_i{}^{x_i})(c_j, s_j{}^{x_j})\ dO_j$$

It follows that the conjecture of Unpredictable Workload type$s_i{}^{x_i}$about cloud service provider$j$'s resource pooling pattern choice is given by ρ_j^*. Namely, the probability that Unpredictable Workload type$s_i{}^{x_i}$assigns to Unpredictable Workloadresource pooling pattern choicec_j is equal to

$$v_i(s_i{}^{x_i})(\{c_j\}\times K_j) = \int_{x_j \in K_j} v_i^{\sim}(s_i{}^{x_i})(c_j, s_j{}^{x_j})\ dO_j$$

$$= \int_{x_j \in K_j} \phi_j^*(x_j)(c_j)\ dO_j$$

$$= \rho_j^*(c_j).$$

Therefore all Unpredictable Workload types of cloud service provider$i$ have the same conjecture ρ_j^*about cloud service provider$j$'s Unpredictable Workloadresource pooling pattern choice. This completes the construction of the elastic scalingmodel. It follows directly from the construction that the elastic scalingmodelis σ-regular.We now show that every Unpredictable Workload type in this model expresses common conjecture in measurability. For this it is sufficient to show that every Unpredictable Workload type$s_i{}^{x_i}$premises in the cloud consumer's measurability. Therefore, we must show for the cloud service providers i and every $s_i{}^{x_i} \in S_i$that$v_i^{\sim}(s_i{}^{x_i})[(C_j \times S_j)^{quant}] = 1$. In order to prove, we show that $v_i^{\sim}(s_i{}^{x_i})(c_j, s_j{}^{x_j}) > 0$ only if $c_j$is measureable for $s_j{}^{x_j}$.

Suppose that $v_i^{\sim}(s_i{}^{x_i})(c_j, s_j{}^{x_j}) > 0$. Since $v_i^{\sim}(s_i{}^{x_i})(c_j, s_j{}^{x_j}) := \phi_j^*(x_j)(c_j)$, it follows that $\phi_j^*(x_j)(c_j) > 0$. As by definition $\phi_j^*(x_j) \in \theta\left(C_j(\rho_i^*, x_j)\right)$it follows that $c_j \in C_j(\rho_i^*, x_j)$. Remember that the conjecture of Unpredictable Workload type$s_j{}^{x_j}$about cloud service provider$i$'s Unpredictable Workloadresource pooling pattern choice is exactly ρ_i^*. Since $c_j \in C_j(\rho_i^*, x_j)$it follows that $c_j$is measureable for Unpredictable Workload type$s_j{}^{x_j}$. Therefore we have shown that $v_i^{\sim}(s_i{}^{x_i})(c_j, s_j{}^{x_j}) > 0$only if $c_j$is measureable for $s_j{}^{x_j}$. This implies that Unpredictable Workload type$s_i{}^{x_i}$premises in the cloud consumer's measurability. Since this holds for every Unpredictable Workload type in the model it follows that every Unpredictable Workload type in the elastic scalingmodel expresses common conjecture in measurability. Therefore every Unpredictable Workload type in the model is σ-measureable because the model is σ-regular. This completes the proof.

7.2 Corollaries

In this subsection we state some technical corollaries, which we need for the proof of the result.

Corollary 7.1. *If* P, Q *and* R*are data valued, independent random variables then*$\Pr(P \geq \max\{Q,R\}) \geq \Pr(P \geq Q) \cdot \Pr(P \geq R)$.

Proof. Let $g_Q$and $g_R$be the probability density functions of the random variables $Q$and R.

Now,

$$\Pr(P \geq \max\{Q,R\})$$

$$= \int_q \int_r \Pr(P \geq \max\{q,r\})\, dg_Q(q)\ dg_R(r)$$

$$\geq \int_q \int_r \Pr(P \geq \max\{q,r\}) \cdot \Pr(P \geq \min\{q,r\})\ dg_Q(q)\ dg_R(r)$$

$$= \int_q \int_r \Pr(P \geq q) \cdot \Pr(P \geq r)\, dg_Q(q)\ dg_R(r)$$

$$= \int_q \Pr(P \geq q)\ dg_Q(q) \cdot \int_r \Pr(P \geq r)\ dg_R(r)$$

$$= \Pr(P \geq Q) \cdot \Pr(P \geq R).$$

Note that the first and third equality follow from the fact that $Q$and R are independent, and the inequality holds because $\Pr(P \geq \min\{q,r\}) \leq 1$.

Corollary7.2.*Let* P*be a random variable with* $H(P)$*=*γ. *Then for any number*$t > 0$,

$$(|P-\gamma| \geq t) \leq \frac{\operatorname{Var}(P)}{t^2}$$

Corollary 7.3. *For every*$n \in \mathbb{N}$,*let* $P_n^1, P_n^2, \dots P_n^m$*be independent random variables with* $H(P_n^i)$*=*γ^i*for all* n*and* i, $\gamma^1 > \gamma^2 > \cdots > \gamma^m$,*and* $\lim_{n\to\infty} \operatorname{Var}(P_n^i) = 0$*for all* i. *Then,*

$$\lim_{n\to\infty} \Pr(P_n^1 \geq P_n^2 \geq \cdots \geq P_n^m) = 1.$$

Proof. For a given n,

$$\Pr(P_n^1 \geq P_n^2 \geq \cdots \geq P_n^m) \geq 1 - \Pr(P_n^i \geq P_n^j \text{ for some } i < j).$$

For fixed $i < j$we have,

$$\Pr(P_n^i < P_n^j) = \Pr(P_n^j - P_n^i > 0)$$

$$= \Pr\left((P_n^j - P_n^i) - (\gamma^j - \gamma^i) > \gamma^i - \gamma^j\right)$$

$$\leq \Pr\left(\left|(P_n^j - P_n^i) - (\gamma^j - \gamma^i)\right| > \gamma^i - \gamma^j\right)$$

$$\leq \frac{\operatorname{Var}(P_n^j - P_n^i)}{(\gamma^i - \gamma^j)^2}$$

$$= \frac{\operatorname{Var}(P_n^j) + \operatorname{Var}(P_n^i)}{(\gamma^i - \gamma^j)^2}$$

The last equality follows from the fact that $P_n^j$and P_n^i are independent. Now, note that $\lim_{n\to\infty} \operatorname{Var}(P_n^i) = 0$and $\lim_{n\to\infty} \operatorname{Var}(P_n^j) = 0$, which implies $\lim_{n\to\infty} \operatorname{Var}(P_n^i < P_n^j) = 0$. Then, from above it follows that

$$\lim_{n\to\infty} \Pr(P_n^1 \geq P_n^2 \geq \cdots \geq P_n^m) = 1.$$

Consider a Unpredictable Workload sequence$(D_n)_{n\in\mathbb{N}}$of normal distributions with mean 0 and varianceσ_n^2such that $\sigma_n \to 0$as$n \to \infty$. The density function $g_n$of D_n is given by

$$g_n(p) = \frac{2}{\sigma_n\sqrt{2\pi}} \mathrm{e}^{-(p^2/2\sigma_n^2)} \text{for all } p\,.$$

We show that for large $n$the right tail of $D_n$becomes arbitrarily steep everywhere.

Corollary 7.4. *Consider a Unpredictable Workload sequence*$(D_n)_{n\in\mathbb{N}}$*of normal distributions with mean* 0 *and variance* σ_n^2, *such that* $\sigma_n \to 0$*as* $n \to \infty$. *Let* g_n*be the density functions of these distributions. Then for all*$c > 0$ *and*$\alpha > 0$*there is* $N \in \mathbb{N}$*such that* $g_n(p+c)/g_n(p) \leq \alpha$*for all*$n \geq N$*and all* $p > 0$.

Proof. Take $c > 0$ and $\alpha > 0$. Then

$$\frac{g_n(p+c)}{g_n(p)} = \frac{\mathrm{e}^{-((p+c)^2/2\sigma_n^2)}}{\mathrm{e}^{-(p^2/2\sigma_n^2)}} = \mathrm{e}^{-(1/2\sigma_n^2)((p+c)^2-p^2)}$$
$$= \mathrm{e}^{-(1/2\sigma_n^2)(2cp+c^2)} \leq \mathrm{e}^{-(c^2/2\sigma_n^2)}$$

Now as $c > 0$ is fixed and $\sigma_n \to 0$as $n \to \infty$, we can find Nlarge enough such that $\mathrm{e}^{-(c^2/2\sigma_n^2)} \leq \alpha$for $n \geq N$.

Corollary 7.5. *Consider aUnpredictable Workload sequence$(D_n)_{n\in\mathbb{N}}$of normally distributed random variables such that $H(P_n) = 0$for all n, and $var(P_n) \to 0$as $n \to \infty$. Let $g_n$be the density functions of these random variables. Then, for every$0 < p < q$it holds that*

$$\lim_{n\to\infty} \frac{\Pr(P_n \geq q)}{\Pr(P_n \geq p)} = 0 \,.$$

Proof. Fix $0 < p < q$and fix a$\alpha > 0$. Then, by corollary 7.4 there is an Nsuch that $g_n(r + (q-p))/g_n(r) \leq \alpha$for all $n \geq N$and all $r > 0$. Take some $n \geq N$. Then,

$$\Pr(P_n \geq q) = \int_q^\infty g_n(r)\, dr = \int_p^\infty g_n(r + (q-p))\, dr$$
$$\leq \alpha \cdot \int_p^\infty g_n(r)\, dr = \alpha \cdot \Pr(P_n \geq p).$$

This implies that

$$\lim_{n\to\infty} \frac{\Pr(P_n \geq q)}{\Pr(P_n \geq p)} = 0 \,.$$

7.3 Proof of the result

We finally prove our main proposition, which is Proposition 5.1. We proceed by three steps.

In step 1, we show how aσ-regular elastic scalingmodel$M$withasymmetric Unpredictable Workloadresource pooling pattern can be transformed into a elastic scalingmodel$M'$with symmetric Unpredictable Workloadresource pooling pattern. More precisely, we transform every Unpredictable Workload type$s_i$in Minto a Unpredictable Workload type$\tau_i(s_i)$inM'which has the same conjecture about the cloud consumer's Unpredictable Workloadresource pooling pattern choice as s_i.In step 2, we take aUnpredictable Workloadresource pooling pattern choicec_i^*that is limit measureable. Therefore we can find a Unpredictable Workload sequence$(D_n)_{n\in\mathbb{N}}$of normal distributions with mean 0 and variance σ_n^2, with $\sigma_n^2 \to 0$as $n \to \infty$, and a Unpredictable Workload sequence$(M^n)_{n\in\mathbb{N}}$of σ_n-regular elastic scalingmodels with constant Unpredictable Workloadtype and utility assignments, such that in every M^nthere is a σ_n-measureableUnpredictable Workload type$s_i^n$with utility function $w_i$for which resource poolingUnpredictable Workloadpattern choice$c_i^*$is optimal. We show that the Unpredictable Workload type$s_i^n$is transformed into aUnpredictable Workload type$\tau_i(s_i^n)$which is α_n-actually measureable for some α_n. Since, for all n, $c_i^*$is measureable for $t_i^n$and $\tau_i(s_i^n)$has the same conjecture about the cloud consumer's Unpredictable Workloadresource pooling pattern choice and the same utility function as s_i^n, it follows that $c_i^*$is measureable for $\tau_i(s_i^n)$for all n. As $\tau_i(s_i^n)$is α_n-actually measureable for every n, it follows that $c_i^*$is α_n-actually measureable for all n.In step 3, we prove that $\lim_{n\to\infty} \alpha_n = 0$. Hence, $c_i^*$is α-actually measureable for every $\alpha > 0$ and therefore actually measureable.

Step 1. Take some $\sigma > 0$. Let $M = (S_i, v_i, k_i)_{i\epsilon I}$be a σ-regular elastic scalingmodel for $\delta$with asymmetric Unpredictable Workloadresource pooling pattern. Now we transform this elastic scalingmodel$M$into aelastic scalingmodel$M' = (\Omega_i, \rho_i)_{i\epsilon I}$with symmetric Unpredictable Workloadresource pooling pattern. Using the fact that $M$is σ-regular we can write

$$v_i(s_i) \,\epsilon\, \theta\big(C_j \times k_j \times V_j\big).$$

Now take$\Omega_i = V_i$and $\Omega_j = V_j$. Clearly, $\Omega_i$and Ω_j are finite sets as $V_i$and V_j are finite. For every $s_i \,\epsilon\, S_i$define the Unpredictable Workload type$\tau_i(s_i)$by

$$\rho_i\big(\tau_i(s_i)\big) := \max g_{C_j \times V_j} v_i(s_i).$$

Therefore,

$$\rho_i\big(\tau_i(s_i)\big)\big(C_j \times V_j\big) = v_i(s_i)\big(K_j \times \{(C_j \times V_j)\}\big)$$

for all $\big(C_j \times V_j\big)$. Hence,

$$\rho_i\big(\tau_i(s_i)\big) \,\epsilon\, \theta\big(C_j \times V_j\big) = \theta\big(C_j \times \Omega_j\big)$$

By construction $\tau_i(s_i)$has the same conjecture about j's Unpredictable Workloadresource pooling pattern choice as s_i. This completes the construction of the elastic scaling$M' = (\Omega_i, \rho_i)_{i\epsilon I}$.

Step 2. Take aUnpredictable Workloadresource pooling pattern choicec_i^*that is limit measureable. Hence, there exists a Unpredictable Workload sequence$(D_n)_{n\in\mathbb{N}}$of normal distributions with mean 0 and variance σ_n^2, with $\sigma_n^2 \to 0$as $n \to \infty$, and a Unpredictable Workload sequence$(M^n)_{n\in\mathbb{N}}$of σ_n-regular elastic scalingmodels with constant Unpredictable Workloadtype and utility assignments, such that in every M^nthere is a σ_n-measureableUnpredictable Workload type$s_i^n$with utility function $w_i$for which Unpredictable Workloadresource pooling pattern choice$c_i^*$is optimal. Let the constant Unpredictable Workload type in the Unpredictable Workload sequence$(M^n)_{n\in\mathbb{N}}$of elastic scalingmodels be $S_i$and S_j, and the constant utility assignments be $k_i$and k_j.Fix an n. Then, within the elastic scalingmodel$M^n = (S_i, v_i^n, k_i)_{i\epsilon I}$there is an σ_n-measureableUnpredictable Workload type$s_i^n \,\epsilon\, S_i$with utility function $w_i$for which $c_i^*$is optimal. Since Unpredictable Workload types_i^nonly deems possible j's Unpredictable Workload types which are σ_n-measureable, and only deems possible j's Unpredictable Workload types which only deem possible i's Unpredictable Workload types which are σ_n-measureable and so on. We may assume without loss of generality that all the Unpredictable Workload types in $M^n$are σ_n-measureable. Let $M'^n = (\Omega_i^n, \rho_i^n)_{i\epsilon I}$be the corresponding elastic scalingmodel with symmetric Unpredictable Workloadresource pooling pattern, as constructed in step 1.For every $\tau_i \,\epsilon\, \Omega_i^n$, we define a number $\alpha_n(\tau_i)$ as follows: Let $\mathrm{Poss}(\tau_i)$be theset of Unpredictable Workload types in $\Omega_j$that$\Omega_i$deems possible. For a given Unpredictable Workload type$\tau_j \,\epsilon\, \mathrm{Poss}(\tau_i)$, suppose that $\tau_j$prefers Unpredictable Workloadresource pooling pattern choice$c_j^1$to c_j^2,$c_j^2$to c_j^3, and so on. Therefore, we obtain an ordering $(c_j^1, c_j^2, c_j^3, \ldots c_j^m)ofj$'s Unpredictable Workloadresource pooling pattern choices.

Then define

$$\alpha_n\big(\tau_i, \tau_j\big) = \max_{t \,\epsilon\, \{2,3,\ldots,m\}} \frac{\rho_i^n(\tau_i)(c_j^t, \tau_j)}{\rho_i^n(\tau_i)(c_j^{t-1}, \tau_j)}$$

Next we define

$$\alpha_{i,n} = \max_{\tau_i \epsilon\, \Omega_i^n, \tau_j \,\epsilon\, \mathrm{Poss}\,(\tau_i)} \alpha_n\big(\tau_i, \tau_j\big)$$

Finally let

$$\alpha_n = \max\{\alpha_{i,n}, \alpha_{j,n}\}.$$

Note that by construction every Unpredictable Workload type in M'^n satisfies the α_n-actual trembling condition; hence every Unpredictable Workload type in M'^n is α_n-actually measureable. In particular $\tau_i(s_i^n)$ is α_n-actually measureable [19] [26].

Step 3. Now we show that $\lim_{n\to\infty} \alpha_n = 0$. It is sufficient to show that

$$\lim_{n\to\infty} \frac{\rho_i^n(\tau_i)(c_j^t, \tau_j)}{\rho_i^n(\tau_i)(c_j^{t-1}, \tau_j)} = 0 \qquad (1)$$

for every $\tau_i \in \Omega_i^n$ and every $\tau_j \in \mathrm{Poss}(\tau_i)$ and every t. As before, cloud service provider j's Unpredictable Workloadresource pooling pattern choices are ordered $c_j^1, \dots c_j^m$ such that τ_j prefers Unpredictable Workloadresource pooling pattern choice c_j^1 to c_j^2, c_j^2 to c_j^3, and so on. We assume, without loss of generality, that all resource pooling pattern preferences are strict.Fix some $\tau_i \in \Omega_i^n$ and $\tau_j \in \mathrm{Poss}(\tau_i)$. Suppose that $\tau_i = \tau_i(s_i)$ for some $s_i \in S_i$ and that $\tau_j = \tau_j(s_j)$ for some $s_j \in S_j$. Let $\phi_j \in \theta(C_i)$ be τ_i'sconjecture about i's Unpredictable Workloadresource pooling pattern choice [28]. As before, let K_j be the set of utility functions for cloud service provider j. For every $t \in \{1, \dots, m\}$, let $P^t: K_j \to F$ be given by

$$P^t(k_j) := k_j(c_j^t, \phi_j) = \sum_{c_i \in C_i} \phi_j(c_i) \cdot k_j(c_j^t, c_i)$$

for every $k_j \in K_j$. Therefore, $P^t(k_j)$ denotes the expected utility for cloud service provider j induced by Unpredictable Workloadresource pooling pattern choice c_j^t, under the conjecture ϕ_j and the utility function k_j. Note that P^t is a random variable, as cloud service provider i holds a probability distribution on K_j, induced by D. The probability distribution of P^t depends on n, and is denoted by $\omega_n^t(P^t)$. Note that P^t has a normal distribution with mean

$$H(P^t) = w_j(c_j^t, \phi_j),$$

and variance

$$Var^n(P^t) = \textstyle\sum_{c_i \in C_i} (\phi_j(c_i))^2 \cdot \sigma_n^2 (2)$$

In particular, it follows that $\lim_n \to_\infty Var^n(P^t) = 0$,as $\lim_n \to_\infty \sigma_n^2$. Since, by assumption, τ_j strictly prefers c_j^1 to c_j^2, strictly prefers c_j^2 to c_j^3, and so on, we have that $H(P^1) > H(P^2) > \cdots > H(P^m)$.Let ω^n be the probability distribution of the random set of data value $(P^1, \dots, P^m)$ [6][14][18]. Recall that all Unpredictable Workload types in M^n are σ_n-measureable,which implies that all Unpredictable Workload types in M^n express common conjecture in measurability. As such, Unpredictable Workload types $s_i \in S_i$ (which generates τ_i) expresses common conjecture in measurability [27]. In particular, s_i only assigns positive probability to those Unpredictable Workloadresource pooling pattern choice-Unpredictable Workload type combinations (c_j, s_j) where c_j is optimal for t_j. Now, as $\tau_i = \tau_i(s_i)$ and $\tau_j = \tau_j(s_j)$, we have that $\rho_i^n(\tau_i)(c_j^t, \tau_j)$ is the probability that c_j^t is optimal for s_j, and that is $\omega^n(P^t \geq P^l \text{ for all } l)$. Then,

$$\frac{\rho_i^n(\tau_i)(c_j^t, \tau_j)}{\rho_i^n(\tau_i)(c_j^{t-1}, \tau_j)} = \frac{\omega^n(P^t \geq P^l \text{ for all } l)}{\omega^n(P^{t-1} \geq P^l \text{ for all } l)} \qquad (3)$$

Hence, in order to prove (1), we must show that

$$\lim_{n\to\infty} \frac{\omega^n(P^t \geq P^l \text{ for all } l)}{\omega^n(P^{t-1} \geq P^l \text{ for all } l)} = 0$$

for all $t \in \{2, \dots, m\}$,. We distinguish two cases.

Case 1. First we consider the case where $t = 2$. Then we have,

$$\frac{\omega^n(P^t \geq P^l \text{ for all } l)}{\omega^n(P^{t-1} \geq P^l \text{ for all } l)} \leq \frac{\omega^n(P^2 \geq P^1)}{\omega^n(P^1 \geq P^2 \geq P^3 \geq \cdots \geq P^m)}$$

Recall that $H(P^1) > H(P^2) > \cdots > H(P^m)$. But then, by Corollary 7.3, $\omega^n(P^2 \geq P^1) \to 0$ and $\omega^n(P^1 \geq P^2 \geq P^3 \geq \cdots \geq P^m) \to 1$, and hence

$$\frac{\omega^n(P^2 \geq P^1)}{\omega^n(P^1 \geq P^2 \geq P^3 \geq \cdots \geq P^m)} \to 0\,,$$

which implies that

$$\frac{\omega^n(P^t \geq P^l \text{ for all } l)}{\omega^n(P^{t-1} \geq P^l \text{ for all } l)} \to 0\,,$$

as $n \to \infty$.

Case 2. Now we consider the case where $t > 2$.

Let P^{max} be the random variable given by $P^{max} := \max_{j \neq t, t-1} P_j$. We have

$$\frac{\omega^n(P^t \geq P^l \text{ for all } l)}{\omega^n(P^{t-1} \geq P^l \text{ for all } l)}$$

$$= \frac{\omega^n((P^t \geq P^{t-1}) \text{ and} (P^t \geq P^{max}))}{\omega^n((P^{t-1} \geq P^t) \text{ and} (P^{t-1} \geq P^{max}))}$$

$$\leq \frac{\omega^n(P^t \geq P^{max})}{\omega^n((P^{t-1} \geq P^t) \text{ and} (P^{t-1} \geq P^{max}))}$$

$$\leq (\text{by Corollary 7.1}) \frac{\omega^n(P^t \geq P^{max})}{\omega^n((P^{t-1} \geq P^t) \cdot \omega^n(P^{t-1} \geq P^{max}))}$$

$$= \frac{\omega^n(P^t \geq P^{max})}{\omega^n(P^{t-1} \geq P^{max})} \cdot \frac{1}{\omega^n(P^{t-1} \geq P^t)}$$

$$= \frac{\omega^n(P^t \geq P^{max})}{\omega^n(P^{t-1} \geq P^{max} - (H(P^{t-1}) - H(P^t)))} \cdot \frac{1}{\omega^n(P^{t-1} \geq P^t)}$$

where the last equality follows from the observation that $P^{t-1} - H(P^{t-1})$ and $P^t - H(P^t)$ have the same distribution.

Now, from Corollary 7.3 it follows that $\omega^n(P^{t-1} \geq P^t) \to 1$ as $n \to \infty$.

We show that

$$\frac{\omega^n(P^t \geq P^{max})}{\omega^n\left(P^{t-1} \geq P^{max} - \left(H(P^{t-1}) - H(P^t)\right)\right)} \to 0$$

as $n \to \infty$.

Let us define $c: H(P^{t-1}) - H(P^t)$. Therefore, we have to show that

$$\frac{\omega^n(P^t \geq P^{max})}{\omega^n(P^{t-1} \geq P^{max} - c)} \to 0 \qquad (4)$$

as $n \to \infty$. Note that $\omega^n(P^t \geq P^{max}) \leq \omega^n(P^t \geq P^1)$. We first show that there exists $N \in \mathbb{N}$ such that for all $n \in N$,

$$\omega^n(P^t \geq P^{max} - c) \geq \omega^n(P^t \geq P^1 - c/2) \qquad (5)$$

Now,

$$\omega^n(P^t \geq P^{max} - c)$$

$$= \omega^n(P^t \geq P^{max} - c \,|P^{max} = P^1) \cdot \omega^n(P^{max} = P^1)$$
$$+\omega^n(P^t \geq P^{max} - c \,|\, P^{max} \neq P^1) \cdot \omega^n(P^{max} \neq P^1)$$
$$\geq \omega^n(P^t \geq P^{max} - c|P^{max} = P^1) \cdot \omega^n(P^{max} = P^1)$$
$$= \omega^n(P^t \geq P^1 - c) \cdot \omega^n(P^{max} = P^1)$$

Therefore, to show (5) it is sufficient to show that there exists $N \in \mathbb{N}$such that for all $n \geq N$,

$$\omega^n(P^t \geq P^1 - c) \cdot \omega^n(P^{max} = P^1) \geq \omega^n(P^t \geq P^1 - c/2) \quad (6)$$

Using Corollary 7.3, $\omega^n(P^{max} = P^1) \to 1$ as $n \to \infty$. We have,

$$\frac{\omega^n(P^t \geq P^1 - c/2)}{\omega^n(P^t \geq P^1 - c)}$$
$$= \frac{\omega^n((P^t \geq P^1) - (H(P^t) - H(P^1)) \geq -c/2 - (H(P^t) - H(P^1)))}{\omega^n((P^t \geq P^1) - (H(P^t) - H(P^1)) \geq -c - (H(P^t) - H(P^1)))}$$

Note that $\omega^n((P^t - P^1) - (H(P^t) - H(P^1)))$has a normal distribution with mean 0 and where the variance of $\omega^n(P^t - P^1)$tends to 0 as $n \to \infty$. Moreover, $-c - (H(P^t) - H(P^1)) > 0$ as $H(P^t) - H(P^1) < H(P^t) - H(P^{t-1}) = -c$. Hence, using Corollary 7.5,

$$\frac{\omega^n((P^t \geq P^1) - (H(P^t) - H(P^1)) \geq -c/2 - (H(P^t) - H(P^1)))}{\omega^n((P^t \geq P^1) - (H(P^t) - H(P^1)) \geq -c - (H(P^t) - H(P^1)))}$$
$$\to 0$$

as$n \to \infty$. Then, we have,

$$\frac{\omega^n(P^t \geq P^1 - c/2)}{\omega^n(P^t \geq P^1 - c)} \to 0$$

Therefore, there exists $N \in \mathbb{N}$such that for all $n \geq N$,

$$\omega^n(P^{max} = P^1) \geq \frac{\omega^n(P^t \geq P^1 - c/2)}{\omega^n(P^t \geq P^1 - c)}$$

This proves (6), which as we have shown, implies (5).Now, by (5) we have

$$\frac{\omega^n(P^t \geq P^{max})}{\omega^n(P^t \geq P^1 - c)}$$
$$\leq \frac{\omega^n(P^t \geq P^1)}{\omega^n(P^t \geq P^1 - c/2)}$$
$$= \frac{\omega^n((P^t \geq P^1) - (H(P^t) - H(P^1)) \geq -(H(P^t) - H(P^1)))}{\omega^n((P^t \geq P^1) - (H(P^t) - H(P^1)) \geq -c/2 - (H(P^t) - H(P^1)))}$$
$$= \frac{\omega^n((P^t \geq P^1) - (H(P^t) - H(P^1)) \geq (H(P^t) - H(P^1)))}{\omega^n((P^t \geq P^1) - (H(P^t) - H(P^1)) \geq (H(P^t) - H(P^1)))}$$
$$\to 0$$

as$n$goes to infinity. Here the convergence follows from Corollary 7.5 as $(H(P^1) - H(P^t)) \geq -c/2$. Therefore, we have shown (4), which completes case 2. Hence, we have shown that (1) holds for all t. Therefore, $\lim_{n\to\infty} \alpha = 0$ and hence the proof is complete.

8. REFERENCES

[1] Kiran M., Saikat Mukherjee, Ravi Prakash G., Characterization of Randomized Shuffle and Sort Quantifiability in MapReduce Model, International Journal of Computer Applications, 51-58, Volume 79, No. 5, October 2013.

[2] Amresh Kumar, Kiran M., Saikat Mukherjee, Ravi Prakash G., Verification and Validation of MapReduce Program model for Parallel K-Means algorithm on Hadoop Cluster, International Journal of Computer Applications, 48-55, Volume 72, No. 8, June 2013.

[3] Barroso, L.A., Ho¨lzle, U.: The datacenter as a computer: an introduction to the design of warehouse-scale machines. Synth. Lect. Comput. Architect. 4, 1–45 (2009).

[4] Kiran M., Amresh Kumar, Saikat Mukherjee, Ravi Prakash G., Verification and Validation of MapReduce Program Model for Parallel Support Vector Machine Algorithm on Hadoop Cluster, International Journal of Computer Science Issues, 317-325, Vol. 10, Issue 3, No. 1, May 2013.

[5] Ravi Prakash G, Kiran M. Saikat Mukherjee, On Randomized Preference Limitation Protocol for Quantifiable Shuffle and Sort Behavioral Implications in MapReduce Programming Model, Parallel & Cloud Computing, Vol. 3, Issue 1, 1-14, January 2014.

[6] Fehling, C., Leymann, F., Mietzner, R., Schupeck, W.: A collection of patterns for cloud types, cloud service models, and cloud-based application architectures. Technical report, University of Stuttgart (2011)

[7] Ravi Prakash G, Kiran M, On The Least Economical MapReduce Sets for Summarization Expressions, International Journal of Computer Applications, 13-20, Volume 94, No.7, May 2014.

[8] Ravi (Ravinder) Prakash G, Kiran M., On Randomized Minimal MapReduce Sets for Filtering Expressions, International Journal of Computer Applications, Volume 98, No. 3, Pages 1-8, July 2014.

[9] Fehling, C., Leymann, F., Retter, R., Schumm, D., Schupeck, W.: An architectural pattern language of cloud-based applications. In: Proceedings of the 18th Conference on Pattern Languages of Programs (PLoP), Portland, (2011).

[10] Fehling, C., Leymann, F., Rutschlin, J., Schumm, D.: Pattern-based development and management of cloud applications. Future Internet 4, 110–141 (2012). (doi:10.3390/fi4010110)

[11] Ravi (Ravinder) Prakash G, Kiran M., How Minimal are MapReduce Arrangements for Binning Expressions. International Journal of Computer Applications Volume 99 (11): 7-14, August 2014.

[12] Ravi (Ravinder) Prakash G, Kiran M., Shuffling Expressions with MapReduce Arrangements and the Role of Binary Path Symmetry. International Journal of Computer Applications 102(16): 19-24, September 2014.

[13] Ravi (Ravinder) Prakash G, Kiran M; How Replicated Join Expressions Equal Map Phase or Reduce Phase in a MapReduce Structure? International Journal of Computer Applications, Volume 107 (12): 43-50, December 2014.

[14] Fehling, C., Ewald, T., Leymann, F., Pauly, M., Ru¨tschlin, J., Schumm, D.: Capturing cloud computing knowledge and experience in patterns. In: Proceedings of the 5th IEEE International Conference on Cloud Computing (CLOUD), Honolulu, (2012).

[15] Ravi (Ravinder) Prakash G, Kiran M., On Composite Join Expressions of Map-side with many Reduce Phase. International Journal of Computer Applications Volume 110(9): 37-44, January 2015.

[16] Dimitri P. Bertsekas, Convex Optimization Algorithms, Athena Scientific, Hardcover Edition ISBN: 1-886529-28-0, 978-1-886529-28-1, Publication: February, 2015, 576 pages.

[17] Ravi (Ravinder) Prakash G, Kiran M; How Reduce Side Join Part File Expressions Equal MapReduce Structure into Task Consequences, Performance? International Journal of Computer Applications, Volume 105(2):8-15, November 2014

[18] Ravi (Ravinder) Prakash G,Kiran M. "On the MapReduce Arrangements of Cartesian product Specific Expressions". International Journal of Computer Applications 112(9):34-41, February 2015.

[19] Ravi (Ravinder) Prakash G, Kiran M., On Job Chaining MapReduce Meta Expressions of Mapping and Reducing Entropy Densities. International Journal of Computer Applications 113(15): 20-27, March 2015.

[20] Ravi (Ravinder) Prakash G, Kiran M. "On Chain Folding Problems of Chain Mapper and Chain Reducer Meta Expressions". International Journal of Computer Applications 116(16): 35-42, April 2015.

[21] Ravi (Ravinder) Prakash G, KiranM."On Job Merging MapReduce Meta Expressions for Multiple Decomposition Mapping and Reducing". International Journal of Computer Applications 118 (13):14-21, May 2015.

[22] Ravi (Ravinder) Prakash G, Kiran M." Characterization of Randomized External Source Output Map Reduce Expressions". International Journal of Computer Applications 123(14):9-16, August 2015.

[23] Ravi (Ravinder) Prakash G, Kiran M., Does there Exist Pruning Decomposition for MapReduce Expressions Arrangements?. International Journal of Computer Applications 125(12): 41-48, September 2015.

[24] Ravi (Ravinder) Prakash G, Kiran M: Can one find External Source Input Expressions for which there exist Map Reduce Configurations? International Journal of Computer Applications 128(12): 14-21, October 2015.

[25] Ravi (Ravinder) Prakash G. and Kiran M., Is It True for Static Scaling Cloud Model there Exists a Centrally Asymmetric Static Workload Pattern? Communications on Applied Electronics 3(4): 39-48, November 2015.

[26] Ravi (Ravinder) Prakash. G and Kiran M., Given a Static Workload Cloud Computing Patterns does it have an Elastic Scaling? Communications on Applied Electronics 4(2): 17-26, January 2016.

[27] Ravi (Ravinder) Prakash G. and Kiran M., How can Periodic Workload Cloud Pattern benefit from Periodically Peaking Utilization?. International Journal of Applied Information Systems 10(5):27-36, February 2016.

[28] Ravi (Ravinder) Prakash G. and Kiran M., On Characterized Once-in-a-Lifetime Workload patterns for Periodic Static Scaling. International Journal of Applied Information Systems 10(6):33-43, March 2016.

Author Profile

Prof. Ravi (Ravinder) Prakash G. teaches Data Visualization, Networks, Kernel Methods for Pattern Analysis, Geometric Methods for Digital Image Analysis, Industrial Imaging, Statistical and Computational Inverse Problems, (Complementary Metal Oxide Semiconductor) CMOS Circuit Design, Layout, and Simulation., Convex Optimization.

Kiran M, Research Scholar, Reva University, Bangalore. He is actively engaged research in Distributed Computing, Cloud Computing