# Allocation of Virtual Infrastructures on Multiple IaaS Providers with Survivability and Reliability Requirements

Anderson S. Raugust, Felipe R. de Souza, Maurício A. Pillon, Charles C. Miers, Guilherme P. Koslovski
*Graduate Program in Applied Computing – Santa Catarina State University – Joinville – Brazil*
*anderson.raugust@edu.udesc.br, dcc6frs@joinville.udesc.br, {mauricio.pillon, charles.miers, guilherme.koslovski}@udesc.br*

***Abstract*—The cloud computing paradigm consolidated the on-demand provisioning of virtual resources. However, the diversity of services, prices, data centers, and geographical footprints, have turned the clouds into a complex and heterogeneous environment. There are several IaaS providers differentiated by the provisioning costs and availability figures. Due to management complexity, the survivability and reliability aspects are often disregarded by tenants, eventually resulting in heavy losses due to unavailability of services that are hosted on Virtual Infrastructures (VIs). We present an alternative to improve VIs survivability and reliability, which considers the use of replicas and the spreading of virtual resources atop providers, regions, and zones. Replicas are used to achieve a user-defined reliability level while the controlled spreading of VI components decrease the probability of full outages. In addition, our proposal performs a cost-effective allocation. We formulate the VI allocation with survivability and reliability requirements as a Mixed Integer Program (MIP). Following, binary constraints are relaxed to obtain a Linear Program (LP). The LP solution is given as input for the simulated annealing technique, composing the Reliable and Survivable Virtual Infrastructure Allocation (RS-VIA) mechanism. Simulation results with different reliability requests indicate an increasing in survivability without inflating costs.**

***Index Terms*—Virtual infrastructures; allocation; survivable; IaaS; multi-provider; reliable**

## 1. Introduction

The Infrastructure-as-a-Service (IaaS) cloud providers offer VIs following the pay-as-you-go model, in which tenants are charged for the Virtual Machines (VMs) flavors and network requirements [1]. Several services are hosted by VIs and an eventual unavailability can affect users on different parts of the world. Although providers comply with strict administrative actions in their Data Centers (DCs) and divulge Service Level Agreement (SLA) availability figures (*e.g.*, $99.95\%$), their efforts may not be enough for critical applications. Indeed, the outage of VIs can induce financial loss to several companies. For instance, when Amazon EC2 had a 20-hours outage, VIs hosting services like Netflix, Instagram, and Pinterest were impacted (*e.g.*, unavailable services, and slow access) affecting millions of users [2]. Recently, another 4-hours outage, affected services like Github, Trello, Giphy, Medium, and Slack [3]. In these cases, cloud tenants just receive credits to re-launch their VIs.

For VIs hosting critical applications, SLAs based only on the up-time availability metrics are insufficient. Although the VI-hosted service may be available, the delivered information may be inaccurate. Indeed, reliability and survivability are essential goals as unplanned DC outages are fairly common, specially in the network control and management plans [4]. Reliability accounts for the probability that a VI is operating properly, while VI survivability indicates the ability to remain operational in the occurrence of cloud provider outages. Both are more precise metrics when compared to up-time availability. An intuitive technique to increase reliability is the use of VM replicas ready to take over the operation in case of failure [5]. However, replicating each VM of the VI twice, at least, the provisioning cost could be prohibitive for most tenants. One approach to decrease the cost is to reduce the spectrum of replicas to only critical instances, those vital for running the application [6]. Consequently, the temporary unavailability of non-critical VMs is tolerable. Complementary, an approach to increase VI survivability is the spreading of VMs atop multiple resources, decreasing the probability of total failure [7], [8]. Although intuitive, both approaches require the analysis of data from multiple providers, regions, and zones.

In this context, we present an alternative to improve VI reliability and survivability based on replicas (for critical components) and controlled virtual resources spreading atop providers. Both techniques are agnostic to hosted applications requirements, and to internal high availability mechanisms. Among the management tasks performed for provisioning a VI, our proposal acts as a cloud broker on allocation of IaaS providers, guided by the tenant's perspective. In short, we make three main contributions in this paper: *(i)* We formulate the VI allocation in multi-cloud providers as a MIP. Our formulation considers the survivability, reliability, and cost aspects. Moreover, the MIP comprises regular and critical VMs as well as data transfer requirements. *(ii)* MIP constraints are relaxed to obtain a LP, and the simulated annealing technique is applied to find an acceptable solution, composing the RS-VIA. *(iii)* Simulation results based on SLA data from public cloud providers are analyzed demonstrating the proposal's applicability.

This paper is organized as follows. Sec. 2 outlines the motivation and challenges on VI allocation atop multiple IaaS providers and formulates the problem. Sec. 3 details the proposed MIP, while the proposed heuristic is described in Sec. 4. Simulation results are presented in Sec. 5, and related work is reviewed in Sec. 6. Final considerations and future works are discussed in Sec. 7.

## 2. Motivation and Problem Formulation

A VI is a combination of VMs and network resources provisioned following the tenant's requirements [9]. On a single cloud provider scenario, a tenant selects the target provider and submits a VI request indicating the VMs configuration and the SLA specification. The cloud provider relies on online algorithms to allocate physical servers and links for hosting the VI request [10]–[13]. Usually, the provider aims to maximizing profit, decreasing cost, and increasing Quality-of-Service (QoS) [10], [14].

There are two technical barriers on tenant's perspective at this point. First, and foremost, the tenants often lack of technical knowledge to select the appropriated provider and to manage the VI. Second, the cloud-internal allocation is a provider-oriented algorithm. In this sense, the present work plays the role of a cloud broker using public providers data to assist tenants on survivable and reliable VI provisioning.

Concerning to the tenant's perspective, QoS and cost-related goals are recurring aspects [15]. The former is addressed by selecting VMs and services based on previously defined flavors, while for the latter, the pay-as-you-go model avoid over-provisioning costs that commonly happens in private and dedicated DCs. Survivability and reliability are QoS requirements that can impact on management and operational costs. Although cloud providers inform the up-time availability of IaaS services on the SLA establishment, nothing is accounted on reliability and correctness of the service hosted by VIs. There are cases where one hour of downtime can result in million-dollar losses [2]. However, tenants do not have direct access to the cloud DC and rely on providers services to minimize the impact of outages, or may have to implement application-level solutions [8], [16]. In this sense, we propose a cost-effective, survivable, and reliable allocation of a VI atop multiple IaaS providers. Table 1 summarizes the notation used along this paper.

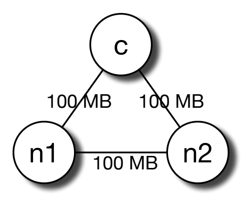
### 2.1. VI Requests and IaaS Providers

A tenant must identify the critical components, and the target reliability level on the SLA establishment, to request a reliable and survivable VI [5], [6]. In this sense, a VI request comprises two set of VMs, termed: regular, and critical. The failure of a regular VM is not severe for the VI-hosted service performance, while the failure of a critical one can fully interrupt the service. Formally, a VI request is represented by $VI(N, D, V, c)$, where $N$ is the set of VMs, $D \subset N$ represents the critical VMs, $V$ denotes the virtual links between VMs (each link has a data transfer request, $v_{nm}$). The target reliability is given by $c$ (*i.e.*, $99.995\%$). The VI request must be allocated atop a single or multiple cloud providers. Each IaaS provider is represented by $P(R, Z)$, in which servers are organized in regions ($R$), and zones ($Z$).

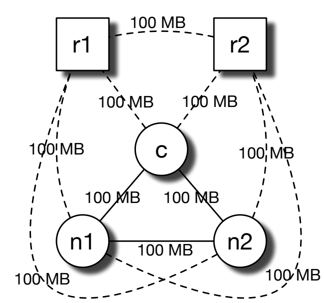
| Notation | Description |
|---|---|
| $P(R, Z)$ | IaaS provider comprising a set of regions ($R$), and zones ($Z$). |
| $j \in R_i$ | A region $j$ from the provider $i$. |
| $k \in Z_{ij}$ | A zone $k$ from the region $j$ and provider $i$. |
| $VI(N, D, V, c)$ | A VI composed of $N$ VMs, $D \subset N$ critical VMs, a traffic matrix ($V$), and the target reliability level ($c$). |
| $n \in N$ | A regular VM $n$. |
| $m \in D \subset N$ | A critical VM $m$. |
| $l_{nm} \in V$ | A virtual link between VMs $n$ and $m$. Each link requests a data volume to be transferred $v_{nm}$. |
| $B$ | Set of replicas for the worst-case failure scenario. |
| $M(i, j, k, c, s)$ | Number of replicas for supporting a reliability level $c$ with $s$ critical VMs on provider $i$, region $j$, and zone $k$. |
| $C(i, j, n)$ | Cost for hosting VM $n$ on provider $i$, region $j$. |
| $C_v(z, k)$ | Cost for data transfers between zones $z$ and $k$, accounted even for different providers. |
| $x_{nijk}$ | VM $n$ mapping on provider $i$, region $j$, and zone $k$ (binary). |
| $b_{nijk}$ | Replica $b$ mapping on provider $i$, region $j$, and zone $k$ (binary). |
| $xl_{nmzk}$ | Virtual links ($nm$) to zones ($z$ and $k$) mapping matrix (binary). |
| $bl_{nmzk}$ | Backup links ($nm$) to zones ($z$ and $k$) mapping matrix (binary). |
| $y_i^p$ | Number of VMs hosted by provider $i$ (integer). |
| $y_{ij}^r$ | Number of VMs hosted by provider $i$, and region $j$ (integer). |
| $y_{ijk}^z$ | Number of VMs hosted by provider $i$, regions $j$, and zone $k$. |

TABLE 1. NOTATION TO REPRESENT VI REQUESTS, IAAS PROVIDERS, AND THE FUNDAMENTAL SYSTEM DETAILS.

Fig. 1 and 2 resume all examples we used along this paper. A request for a reliability level $c = 99.995\%$, one critical VM, and two regular VMs ($n1$ and $n2$) is submitted (Fig. 1(a)), with the data transfer request for each VMs pair (100MB).

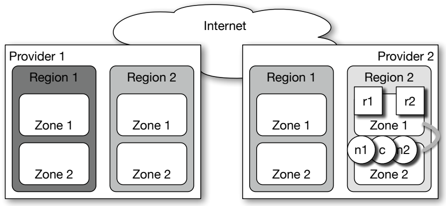



(a) VI request.

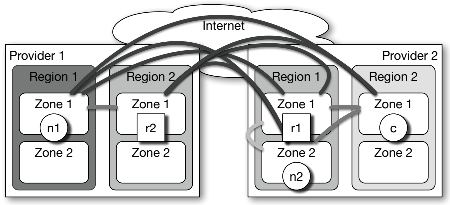



(b) VI and replicas.

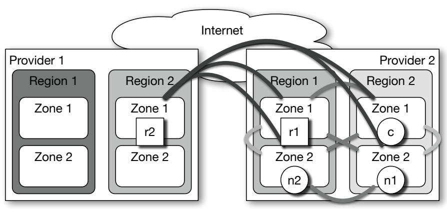
Figure 1. VI allocation with target reliability $c = 99.995\%$.




(a) Minimization of VI allocation cost.




(b) Spreading VI resources atop providers.




(c) Cost-effective, reliable, and survivable allocation.

Figure 2. VI allocation with target reliability $c = 99.995\%$ and 3 groups of failure (providers, regions, and zones). The lighter is the color, then lowest is the allocation cost.

## 2.2. Probability of Failure

There is usually a sequence of events that results in failures. Initially, a fault activation causes an error that is propagated to a failure [17]. The failure of a subsystem can cause a fault in other system that interact with it, following the propagation chain. Such failures may happen in servers and network resources (*e.g.,* switches, and routers). Logs and data on DC failures are useful to understand and reduce the probability of future events [4]. However, raw data is privately accounted and confidentially kept. Cloud tenants are just aware of availability figures specified during the SLA establishment. Precise information on Mean Time Between Failures (MTBF), Mean Time To Repair (MTTR), and Mean Time Between Outages (MTBO) are usually not shared. Limited by the confidentially barrier on MTBF, MTTR, and MTBO figures, a tenant must rely on approximations to improve the VI configuration, specifically for identifying the number of VM replicas. When not available, the probability of failures and the reliability numbers can be inferred based on previous outages. For instance, the MTBF can be roughly deduced for the last 30 days as $\frac{720-\sum \text{duration of outage}}{\#\text{outages}}$, for a one-hour window. The probability of failures ($p$) is given by $\frac{1}{\text{MTBF}}$. Finally, the reliability is given as $1-p$.

Widespread measures of IaaS providers availability and outages (30 days period) are accounted by external services, such as CloudHarmony (https://cloudharmony.com/). For instance, on April 2017, CloudHarmony identified an availability of $99.997\%$ for *ap-northeast-2* region of Amazon EC2 provider, and $99.809\%$ for *ams-e* region of ElasticHosts provider. The latter had a higher number of outages over the analyzed period. In this way, the reliability is approximately defined as $97.495\%$ and $99.861\%$ for ElasticHosts and Amazon, respectively. It is worthwhile to highlight that $p$ is an approximation. Any mechanism capable of offering a more precise probability can be applied. Moreover, the probability represents an independent failure (crash) which may affect a single resource (*e.g.*, a server) or a group of resources (*e.g.*, zone, region). In this sense, it is evidenced that the spreading of VMs and replicas across different failure groups is beneficial to decrease the probability of total failure, consequently increasing the VI survivability [7], [8].

## 2.3. Defining Replicas for Critical VMs

The use of replicas is a promising approach for fullfilling the reliability gap between the providers and the VI requirement [6], [16]. Initially, the critical VMs ($D \subset N$) and the target reliability ($c$) for VI are requested. Afterwards, this information is combined with the providers probabilities of failure (zones) to apply the Opportunistic Redundancy Pooling (ORP) technique [5]. ORP uses an incomplete regularized beta function, $I_{1-p} = (n, k+1)$, where $n$ is the number of critical VMs ($|D|$), $k+1$ is the number of required replicas, and $1-p$ is the zone reliability level. Therefore, the number of replicas is the smallest number that guarantees $c$. Formally, the number of replicas required for achieving $c$ with $D$ critical VMs is per zone calculated and represented by $M$.

Fig. 3 exemplifies the composition of $M$. Using ORP, a range of critical nodes supported by $k$ replicas is identified. Given the number of critical VMs ($x$-axis), the number of replicas ($y$-axis) is computed for providers with distinct failure figures. Based on April/2017 data collected by CloudHarmony, the VPS.NET (Atlanta) region has a low probability of failure ($p = 0.001$), and consequently, with a target reliability $c$ is $99.995\%$, only $4$ replicas are need to support between 74 and 241 critical VMs. $M$ is indexed by zone, target reliability $c$, and the number of critical VMs.

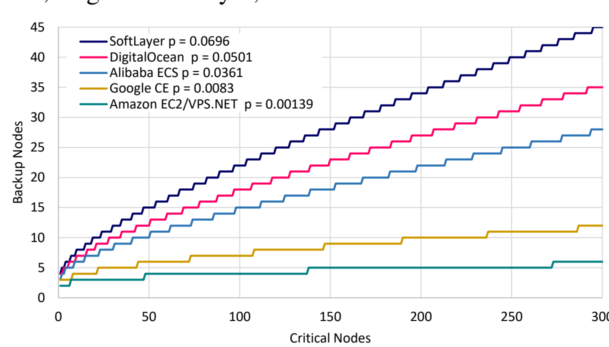



Figure 3. Number of replicas required to support $c = 99.995\%$ for providers with different $p$.

Fig. 1(b) exemplifies the extension of a VI request (from Fig. 1(a)) adding replicas and links. For this example, 2 replicas are arbitrarily added ($r1$ and $r2$) to achieve the target reliability $c$ atop both providers. The dashed lines represent the new virtual links required to delivery connectivity in the occurrence of a failure.

## 2.4. Allocating IaaS providers to host VIs

The VI requests are individually analyzed by the mechanism characterizing an on-line allocation problem [10]. The mapping of VMs to zones is given by $\mathcal{M} : N \mapsto Z$. The internal provider allocation policy is out of scope on this paper. We argue providers selection is a tenant's choice, consequently performed considering the tenant's perspective, while intra DC allocation algorithms [5], [10]–[13] are arbitrarily defined by the provider. In addition, the broker service can be executed any time for accomplishing with new probability of failures. However, reallocation and migration mechanisms are not discussed and indicated as future work.

With regard to the tenant's perspective, the allocation goal is the cost-effective selection of providers guided by survivability and reliability requirements. The first dimension aims at minimizing the VMs (regular, critical and replicas) and networking (data transfer between VMs) provisioning costs, while the second aims at minimizing the impact of providers failures on the VI [7], [8].

Three examples of VI allocation are presented on Fig. 2. For differentiating prices between providers and regions (as commonly performed by public clouds), a color scale is used. The lighter is the color, the low-cost is the allocation. The same approach is applied for lines on virtual links. Initially, Fig. 2(a) exemplifies an allocation decreasing the VI provisioning cost. All resources are placed on two zones from a single region. Thus, besides of decreasing the VMs provisioning cost, the allocation softened the communication costs as data transfers inside a zone are not charged (white color). A survivable-only solution is presented by Fig. 2(b). VMs are spread atop 5 zones, 4 regions, and 2 providers, ignoring the provisioning cost. Indeed, data transfer between providers must be performed. Based on this allocation, the probability of a total VI failure is minimized for all failure groups (providers, regions, and zones).

The focus of our work is to identify an intermediate approach, as given by Fig. 2(c). The allocation still used 2 providers and 5 zones, but reduced the number of regions to 3, motivated by the allocation cost: region 1 from provider 1 was ignored due to the high price. In addition, the number of virtual links communicating over the Internet was reduced. Finally, it is noted the reliability level $c$ was achieved for all scenarios by adding the replicas.

# 3. Exact MIP for Allocating Survivable and Reliable VIs

## 3.1. Variables and Objective

Four variables are used to identify which providers must host a given VI request (*e.g.*, Fig. 1(b)). Initially, $x_{nijk}$, a binary variable, indicates the mapping of regular and critical VMs ($n \in N$) on provider $i$, region $j$, and zone $k$. For applying the same rationale to replicas, the set $B$ must be defined. However, the exact number of replicas depends on which providers, regions and zones will be selected to host the critical VMs, and such information is unknown in advance. On the survivability perspective, $B$ represents the worst-case scenario where the zone selected for hosting critical VMs has the highest probability of failure. However, the model aims at minimizing the number of replicas need for guaranteeing the requested reliability level. The allocation of a replica is indicated by the binary variable $b_{nijk}$ ($n \in B$).

Thus, for data transfer between VMs, two variables are used to define the allocation of virtual links, $xl$ and $bl$. The former represents the allocation of a virtual link $l_{mn}$ between VMs $n$ and $m$, while the second follows the rationale to replicas. The source $n$ of a $l_{nm}$ link is mapped to the corresponding zone that is hosting $n$, while the target $m$ is mapped to destination zone. For regular and critical VMs, $l_{nm}$ are known in advance, while for connectivity to replicas, they are quantified on-the-fly. In this sense, all possible connections between $N$ (regular and critical VMs) and $B$ (replicas) are analyzed by $bl$. Moreover, the connectivity between replicas ($B \times B$) is also accounted. However, just those need (according to $b$) are effectively allocated.

**3.1.1. VI Allocation Cost.** IaaS providers apply different cost models for VMs, usually differentiated by regions. In this sense, function $C(i, j, n)$ returns the cost for hosting a VM $n$ on provider $i$, region $j$, and Eq. (1) and (2) account the costs for hosting all VMs and the dynamically defined replicas, respectively.

$$C_{vm}(VI) = \sum_{n \in N} \sum_{i \in P} \sum_{j \in R_i} \sum_{k \in Z_{ij}} x_{nijk} \times C(i, j, n) \quad (1)$$

$$C_{vmb}(VI) = \sum_{w \in B} \sum_{i \in P} \sum_{j \in R_i} \sum_{k \in Z_{ij}} b_{wijk} \times C(i, j, w) \quad (2)$$

The costs for data transfer between VMs are given by Eq. (3) (regular and critical) and Eq. (4) (replicas). As commonly applied by public cloud providers, the data transfer cost is differentiated for zones, regions, and providers. This information is abstracted by $C_v(z, k)$, with informs the per MB price for transferring data between zones $z$ and $k$ (even between different providers).

$$C_{net}(VI) = \sum_{l_{nm} \in V} \sum_{i_s \in P} \sum_{j_s \in R_{i_s}} \sum_{z \in Z_{i_s j_s}} \sum_{i_t \in P} \sum_{j_t \in R_{i_t}} \sum_{k \in Z_{i_t j_t}} xl_{nmzk} \times v_{nm} \times C_v(z, k) \quad (3)$$

$$C_{netb}(VI) = \sum_{l_{nm} \in N \times B} \sum_{i_s \in P} \sum_{j_s \in R_{i_s}} \sum_{z \in Z_{i_s j_s}} \sum_{i_t \in P} \sum_{j_t \in R_{i_t}} \sum_{k \in Z_{i_t j_t}} (bl_{nmzk} \times v_{nm} \times C_v(z, k)) + \sum_{l_{nm} \in B \times B} \sum_{i_s \in P} \sum_{j_s \in R_{i_s}} \sum_{z \in Z_{i_s j_s}} \sum_{i_t \in P} \sum_{j_t \in R_{i_t}} \sum_{k \in Z_{i_t j_t}} (bl_{nmzk} \times v_{nm} \times C_v(z, k)) \quad (4)$$

Finally, the total cost for allocating a VI is given by Eq. (5). The weight's vector $\alpha$ is used to denote the importance level of each component.

$$C_{total}(VI) = \alpha_{vm}C_{vm}(VI) + \alpha_{vmb}C_{vmb}(VI) + \alpha_{net}C_{net}(VI) + \alpha_{netb}C_{netb}(VI) \quad (5)$$

**3.1.2. Impact of IaaS Providers Failures.** An intuitive approach to decrease the impact of failures on VI-hosted applications is the spreading of virtual resources atop multiple domains of failures [7], [8], [16]. In our context, a domain of failure is a provider, region, or zone. A zone represents the smallest unit, consequently with the highest probability of failure. The remaining domains aggregate zones (or regions) and soften the probability. In short, in tenant's perspective, the larger the spreading of virtual resources, the lower the probability that a failure can cause an outage on VI-hosted service. Formally, three integer variables are used to represent the use of providers, regions and zones, $y_i^p, y_{ij}^r$ e $y_{ijk}^z$, respectively. Eqs. (6)-(8) account the number of VMs hosted by providers, regions, and zones.

$$y_i^p = \sum_{j\in R_i}\sum_{k\in Z_{ij}}\left(\sum_{n\in N} x_{nijk} + \sum_{w\in B} b_{wijk}\right); \forall i \in P \quad (6)$$

$$y_{ij}^r = \sum_{k\in Z_{ij}}\left(\sum_{n\in N} x_{nijk} + \sum_{w\in B} b_{wijk}\right); \forall i \in P; \forall j \in R_i \quad (7)$$

$$y_{ijk}^z = \sum_{n\in N} x_{nijk} + \sum_{w\in B} b_{wijk}; \forall i \in P; \forall j \in R_i; \forall k \in Z_{ij} \quad (8)$$

For spreading the VMs and replicas atop failure groups, three integer (positive) variables (Eqs. (9)-(11)) are applied for compositing the minimization (min. $I(VI)$, Eq. (15)). All three variables maximize the distribution atop failure groups (providers, regions, and zones) respecting the number of VMs and replicas (Eqs. (12)-(14)). The weight's vector $\beta$ differentiates the importance of each component.

$$I^p \geq y_i^p; \forall i \in P \quad (9)$$

$$I^r \geq y_{ij}^r; \forall i \in P; \forall j \in R_i \quad (10)$$

$$I^z \geq y_{ijk}^z; \forall i \in P; \forall j \in R_i; \forall k \in Z_{ij} \quad (11)$$

$$I^p \leq |N| + |B|; \forall i \in P \quad (12)$$

$$I^r \leq |N| + |B|; \forall i \in P; \forall j \in R_i \quad (13)$$

$$I^z \leq |N| + |B|; \forall i \in P; \forall j \in R_i; \forall k \in Z_{ij} \quad (14)$$

$$I(VI) = \beta_p I^p + \beta_r I^r + \beta_z I^z \quad (15)$$

**3.1.3. Objective Function.** The minimization of Eq. (16) results on lowest allocation cost and decreases the impact caused by a failure. The first term is normalized by the cost for hosting on the costly zone ($C_{max}(VI)$), while the second term is normalized by the number of VMs and replicas.

$$min: \frac{C_{total}(VI)}{C_{max}(VI)} + \frac{I(VI)}{|N| + |B|} \quad (16)$$

## 3.2. Constraints

For guaranteeing the SLA QoS, a set of capacity, data transfer, meta and binary constraints must be satisfied.

$$\sum_{i\in P}\sum_{j\in R_i}\sum_{k\in Z_{ij}} x_{nijk} = 1; \forall n \in N \quad (17)$$

$$\sum_{i\in P}\sum_{j\in R_i}\sum_{k\in Z_{ij}} b_{nijk} \leq 1; \forall n \in B \quad (18)$$

$$\sum_{w\in B}\sum_{i\in P}\sum_{j\in R_i}\sum_{k\in Z_{ij}} b_{wijk} \geq min(M) \quad (19)$$

$$\sum_{w\in B}\sum_{i\in P}\sum_{j\in R_i}\sum_{k\in Z_{ij}} b_{wijk} \leq |B| \quad (20)$$

$$\sum_{q\in Z_{st}} xl_{nmkq} + \sum_{q\in Z_{st}} xl_{nmzq} = x_{nijk} + x_{mijk}$$
$$i \in P, j \in R_i, k \in Z_{ij}, s \in P, t \in R_s, l_{nm} \in V \quad (21)$$

$$\sum_{q\in Z_{st}} bl_{nmkq} + \sum_{q\in Z_{st}} bl_{nmzq} = x_{nijk} + b_{mijk}$$
$$i \in P, j \in R_i, k \in Z_{ij}, s \in P, t \in R_s, l_{nm} \in N \times B \quad (22)$$

$$\sum_{q\in Z_{st}} bl_{nmkq} + \sum_{q\in Z_{st}} bl_{nmzq} = b_{nijk} + b_{mijk}$$
$$i \in P, j \in R_i, k \in Z_{ij}, s \in P, t \in R_s, l_{nm} \in B \times B \quad (23)$$

$$\sum_{k\in Z_{ij}}\sum_{q\in Z_{st}} xl_{nmkq} = 1$$
$$i \in P, j \in R_i, s \in P, t \in R_s, l_{nm} \in V \quad (24)$$

$$\sum_{k\in Z_{ij}}\sum_{q\in Z_{st}} bl_{nmkq} \leq 1$$
$$i \in P, j \in R_i, s \in P, t \in R_s, l_{nm} \in N \times B \quad (25)$$

$$\sum_{k\in Z_{ij}}\sum_{q\in Z_{st}} bl_{nmkq} \leq 1$$
$$i \in P, j \in R_i, s \in P, t \in R_s, l_{nm} \in B \times B \quad (26)$$

Constraints (17) and (18) indicate that VMs and replicas, respectively, must be allocated at most one time. The minimum number of replicas indicated by ORP is guaranteed by Eq. (19), while the upper-bound limit is the allocation on the zone with highest failure (Eq. (20)). Constraints (21)-(23) ensure that virtual links $V$ are hosted by zones hosting source and destination [18]. Finally, Eqs. (24)-(26) guarantee that virtual links are hosted at most one time.

# 4. Allocation Mechanism

Solving a MIP is known to be computationally infeasible. Thus, we relax the binary constraints obtaining a LP. Latter, the approximated result is interpreted and used as input for a simulated annealing technique. The combination of LP with both heuristics compose RS-VIA.

## 4.1. Relaxing Binary Variables

For obtaining an LP, the binary constraints of variables $x$, $b$, $xl$, and $bl$ are relaxed ($\geq 0$, $\leq 1$, $\in \mathbb{R}$). Previous work applied deterministic and random rounding techniques to interpret the LP results [19]. Although efficient for physical resources allocation for hosting virtual networks, the techniques are not suitable for multiple providers selection (Sec. 5.3.2). In this work, we propose the use of Simulated Annealing (SA) for interpreting the LP (Alg. 1).

```
Input: $VI, x, b, T, \alpha$
Output: $\mathcal{M}$; VI mapping
1  $T = C_{max}(VI)$
2  $obj_{best} = T$
3  $sol = \emptyset$
4  while $T \geq 1$ do
5      shuffle(N)
6      shuffle(B)
7      for $n \in N \cup B$ do
8          for $i \in P, j \in R_i, k \in Z_{ij}$ do
9              if $n \in N$ then
10                 $p_k = x_{nijk} \times \sum_{l_{nm} \in V} v_{nm}; m \in \mathcal{M}$
11             else
12                 $p_k = b_{nijk} \times \sum_{bl_{nm} \in V} v_{nm}; m \in \mathcal{M}$
13             end
14         end
15         cand = $\emptyset$
16         for $i \in P, j \in R_i, k \in Z_{ij}$ do
17             s = $\lceil \frac{p_k}{min(p)} \rceil$
18             for range(1, s) do
19                 cand.add($w$)
20             end
21         end
22         c = rand(cand)
23         $sol \leftarrow$ [c, z]
24     end
25     $obj$ = Eq. 16
26     if $obj \leq obj_{best}$ then
27         $obj_{best} = obj$
28         $\mathcal{M}$ = sol
29     $T = T \times (1 - \alpha)$
30 end
31 return $\mathcal{M}$
```

**Algorithm 1:** RS-VIA: simulated annealing.

## 4.2. Heuristic Algorithm

The SA algorithm (Alg. 1) receives as input the VI request, two parameters ($T$ and $\alpha$) for controlling the SA execution, and the LP results given by relaxed variables $x$ and $b$. While the annealing criteria holds ($T$, lines 4 and 29), the SA shuffles the VMs sets $N$ and $B$ for composing an initial solution (lines 5 and 6). For each VM a candidate is chosen based on LP values. The set of candidates, termed *cand*, is composed of all possible candidates previously identified by the LP ($x > 0, b > 0$). Rather of composing *cand* only based on LP [19], RS-VIA accounts the networking impact analyzing the previous mapping ($\mathcal{M}$) on lines 10 and 12. Preference is given to candidates with high $p_k$ as the network cost may be reduced (line 17). After placing all VMs, the objetive function is accounted (line 25) and stored if improves the previous one (lines 26 to 28). Latter, a suitable solution or an empty mapping ($\mathcal{M}$) is returned.

# 5. Evaluation and Analysis

As proof-of-concept, a cloud broker was implemented in Java v1.8 using the IBM CPLEX optimizer (v12.6.1.0)[1]. The simulation was executed on a desktop using processor AMD Phenom II X4 (4 cores), 4GB RAM, running GNU/LinuxUbuntu 14.04.

## 5.1. Metrics

For representing the tenant's perspective, seven metrics were selected. *(i) Regular and critical VMs costs*. *(ii) Cost of replicas*. *(iii) Network cost* (Eq. (3)). *(iv) Network cost between replicas* (Eq. (4)). *(v)-(vii) Number of zones, regions and providers* used for hosting VMs regular, critical and replicas. The cost metrics are normalized by the maximum cost, while the failure groups (zones, regions, and providers) are represented as the ratio regarding the total group size.

1. IBM CPLEX Optimizer: https://www.ibm.com/software/commerce/optimization/cplex-optimizer/

## 5.2. Simulation Scenarios

**5.2.1. Parameters.** The probability of failure for each zone was extracted from CloudHarmony, specifically the data from August 2017. Network prices are uniformly selected on three ranges: *(i)* between zones on the same region: [\$0.01, \$0.05]; *(ii)* inter zones on the same provider: [\$0.1, \$0.5]; and *(iii)* over the Internet: [\$1.5, \$2.0]. Data transfer between VMs placed on the same zone are not charged.

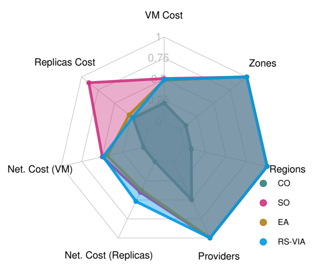
We choose the popular $m3.large$ instance from Amazon EC2 [20] to compose the VI request. To define the cost function $C(i, r, n)$, a similar configuration was selected for each provider. Finally, VMs were organized in a full-mesh topology, and data transfer requests were defined as 500 MB per month. The allocation policies are divided into 2 target reliability, 99.95% and 99.995%, resulting on different configurations of $M$ set (as discussed in Sec. 2.3). For parameterizing the objective function, each element of $\alpha$ was configured with 0.25, while 0.33 was used for $\beta$.

**5.2.2. Exact Allocation and RS-VIA.** This simulation analyzes two public cloud providers (Amazon EC2 and Google Computing Engine) totalizing a geographical footprint of 17 regions and 24 zones. A VI composed of 5 regular and 5 critical VMs is requested to be allocated with 4 approaches. The Cost-Only (CO) has as objective the minimization of allocation cost (Fig. 2(a)), while the Survivable-Only (SO) considers the maximum spreading of virtual resources as main goal, without concerning on the final cost of the VI allocation (Fig. 2(b)). The Exact Allocation (EA) combines the goals of cost and survivability (Fig. 2(c)). Finally, the RS-VIA aims to provide an approximate solution to EA.

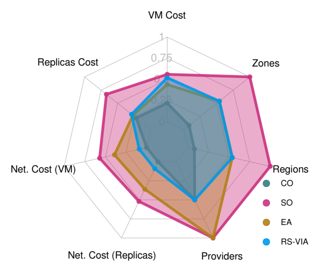
**5.2.3. RS-VIA Allocation.** The scenario analyzes the applicability of the RS-VIA on real cloud data. The simulation comprises 31 public cloud providers with a geographical footprint of 48 regions and 133 zones. This scenario compares 3 VI requests composed of 50 VMs, with distinct configurations on number of regular and critical VMs ($40 - 10$, $25 - 25$, $10 - 40$). As the literature lacks on algorithms to allocate reliable and survivable VIs atop multiple providers (Sec. 6), we perform a comparison with a random selection of candidates. Each approach is executed 10 times and the reported results are mean values with standard deviation.

## 5.3. Simulation Results

**5.3.1. Exact MIPs and RS-VIA.** Simulation results for this scenario are presented on Fig. 4. As expected, the CO approach has the smallest footprint, prioritizing the concentration of VMs on regions with lowest prices. Moreover, the replicas are hosted by the same zones on both scenarios (the lowest price). In turn, the SO model resulted on the opposite behaviour of CO: it allocates more providers, regions and zones increasing the data transfer costs. The VI spreading is performed without concern to provisioning costs. All providers, regions, and zones are candidates to spread the VMs, and certainly expensive zones are selected. Referring to Fig. 2(b) to exemplify the cost increasing incurred from the SO approach, the more expensive group, Region 1 from Provider 1, hosts a VM. The same is observed in the simulation results.




(a) $c = 99.95\%$.




(b) $c = 99.995\%$.

Figure 4. Results for VI allocations with different target reliability $c$.

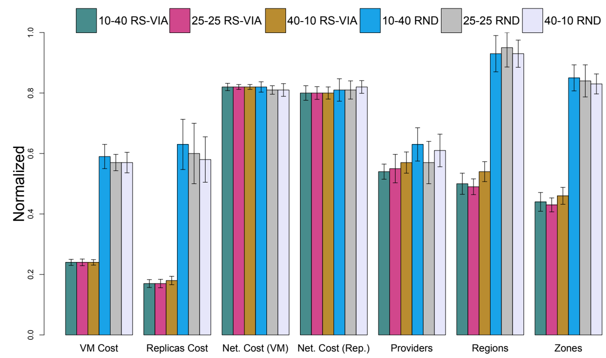
The EA benefits from the pros of both approaches. Although the cost is increased, the number of regions and zones hosting the VI provides a survivability unreachable by the CO approach. Although efficient, the application of EA on real cloud scenarios is unpractical due to the combinatorial explosion. In this sense, EA is only applicable with small data sets. It is worthwhile to highlight that the results with RS-VIA are close to EA in Fig. 4(a), differing in Fig. 4(b) as a single provider is selected, and consequently the data transfer cost is decreased. This behaviour is explained by Eq. (16): an approximated solution can be reached with a distinct tradeoff. Finally, RS-VIA stayed close to the objective function reached by the MIP (0.68 and 0.64, respectively).

**5.3.2. Simulation Results with RS-VIA.** Fig. 5 shows the different behaviors of the allocation with RS-VIA and the random selection (RND). The random approach spreads the resources inflating the VM costs, however, the networking costs remain close to RS-VIA, being statistically equivalents. It is worthwhile to mention that the allocations with RS-VIA reduced the VMs (regular, critical and replicas) costs, achieving a cost-effective solution. Moreover, RS-VIA maintained the spreading based on the exact model, providing a survivable VI.

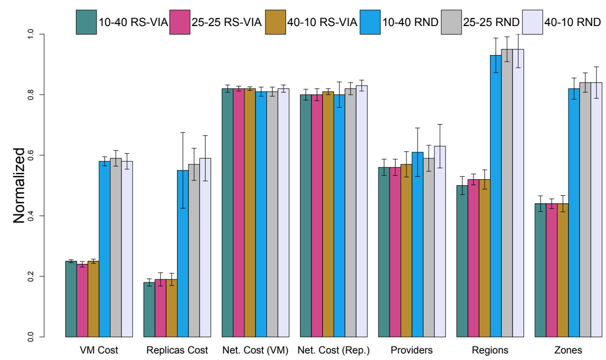
## 6. Related Work

The specialized literature comprises the allocation of physical resources to host VIs, and techniques to improve virtual resources survivability and reliability.

**Allocating physical resources for hosting VIs.** Houidi *et al.* [13] proposed a MIP and a set of heuristics to solve the Virtual Network Embedding (VNE) problem focusing on




(a) $c = 99.95\%$.




(b) $c = 99.995\%$.

Figure 5. VI requests allocation with RS-VIA and randomly performed.

cost reduction and acceptance ratio increase. They propose the allocation atop multiple providers. We share a similar approach considering IaaS clouds, however, providers details are not required neither interoperability mechanisms.

A different perspective was analyzed by Caron *et al.* [21]: instead of considering multiple providers, the proposal aimed the simultaneously allocation atop a private cluster and a public cloud. An optimal allocation concerning the multiple allocation criteria was proposed. Ficco *et al.* [22] proposed a meta-heuristic scheme for managing elastic resources reallocation in cloud infrastructures. They aim to maintain a balance between the different interests of clients SLAs and the provider during the allocation, resizing, replication and migration processes. Both proposals can be jointly applied with our approach for improving the selection of a candidate cloud provider.

Regarding to the allocation of VIs into DCs, techniques to minimize the bandwidth consumption combined with privacy support were proposed in [18]. We are aligned with this proposal considering the virtual link modeling. In [12], a tree-based heuristic was proposed to speedup the VI allocation. The heuristic tends to group virtual resources increasing the impact of an eventual failure. Although not aiming a survivable allocation, a controlled spreading of virtual resources atop a cloud DC was applied in [11], however full knowledge and control on cloud DC is required.

Summing up, the literature on VI embedding into DC, or similar scenarios (VNE), comprises multiple proposals with distinct goals [10]. Concerning to the multiple provider approaches, the previously proposed techniques rely on interoperability data and/or sharing of private provider's data, while the present proposal is based on public information and can be applied for any IaaS provider. Moreover, the present proposal is agnostic to private allocation mechanism.

**Techniques for provisioning survivable VIs**. The survivable provisioning of VIs was proposed in [6]. Similarly to the present work, the mechanism relied on ORP for defining the number of replicas. However, the allocation was conducted on a controlled DC with where the mechanism has full knowledge on probability of failures and MTBF. The allocation was performed in two steps, first defining the number of replicas and later applying an allocation heuristic, which can lead to a suboptimal solution. Groups of failures and cost-effective allocation were not considered. Our approach advance the field by jointly defining the replicas and spreading VMs on multiple providers. In short, on a single step, the exact survivable, reliable and cost-effective allocation is accounted, as discussed in Sec. 5.3.1.

The ORP technique was also applied for VNE [5]. A small set of replicas was defined for backing up multiple tenants. We share a different view on the present work considering a non-cooperative scenario as usually observed on public providers. Indeed, the SLA is individually performed with each tenant defining the target reliability. In addition, Bodik *et al.* [7] improved the fault tolerance on DC without increasing the bandwidth load, while Cavalcanti *et al.* [8] investigated the tradeoff between DC fragmentation and survivable provisioning. It is worthwhile to highlight that the present proposal combined cost-effective with survivable and reliable VI allocation on multiple cloud providers, filling a research gap with concerns to the tenant's perspective.

## 7. Considerations & Future work

We presented an alternative to increase the VI survivability, ensuring the request reliability through replicas, without increase the cost of the VI allocation. In order to achieve that, we formulate a MIP to define the exact allocation of the VI atop multiples providers. Latter, a set of variables were relaxed obtaining a LP. The approximated results are used as input for decreasing the number of candidates on a SA algorithm, composing RS-VIA. The results shows our solution is effective in terms of reliability and survivability, without inflating the provisioning cost. The total cost remains as close as possible to the minimum for the requested VI, respecting the target reliability. Further work aims at performing the implementation as an open cloud service.

**Acknowledgments**. This research was supported by the UDESC, PROMOP / CAPES, and developed at LabP2D.

## References

[1] P. M. Mell and T. Grance, "SP 800-145. the NIST definition of cloud computing," NIST, Tech. Rep., 2011.

[2] A. Avram, "Amazon EC2 Outage Explained and Lessons Learned," InfoQ, Tech. Rep., 2011.

[3] Amazon EC2, "Amazon Web Services summary of the amazon s3 service disruption in the northern virginia (us-east-1) region," Amazon, Tech. Rep., 2017.

[4] R. Govindan, I. Minei, M. Kallahalla, B. Koley, and A. Vahdat, "Evolve or die: High-availability design principles drawn from googles network infrastructure," in Proc. of the SIGCOMM conference, ACM. Florianopolis, Brazil: ACM, 2016, pp. 58–72.

[5] W.-L. Yeow, C. Westphal, and U. C. Kozat, "Designing and embedding reliable virtual infrastructures," SIGCOMM Computer Communication Review, vol. 41, no. 2, pp. 57–64, 2011.

[6] G. Koslovski, W. L. Yeow, C. Westphal, T. T. Huu, J. Montagnat, and P. Vicat-Blanc, "Reliability support in virtual infrastructures," in 2nd Int. Conf. on Cloud Computing Technology and Science. Indianapolis, USA: IEEE, Nov 2010, pp. 49–58.

[7] P. Bodík, I. Menache, M. Chowdhury, P. Mani, D. A. Maltz, and I. Stoica, "Surviving failures in bandwidth-constrained datacenters," in SIGCOMM Conf. on Applications, Technologies, Architectures, and Protocols for Computer Comm. ACM, 2012, pp. 431–442.

[8] G. Cavalcanti, R. R. Obelheiro, and G. Koslovski, "Optimal resource allocation for survivable virtual infrastructures," in Design of Reliable Communication Networks. Ghent, Belgium: IEEE, 2014, pp. 1–8.

[9] B. Sotomayor, R. S. Montero, I. M. Llorente, and I. Foster, "Virtual infrastructure management in private and hybrid clouds," IEEE Internet computing, vol. 13, no. 5, pp. 14–22, 2009.

[10] A. Fischer, J. F. Botero, M. T. Beck, H. De Meer, and X. Hesselbach, "Virtual network embedding: A survey," IEEE Communications Surveys & Tutorials, vol. 15, no. 4, pp. 1888–1906, 2013.

[11] F. R. de Souza, C. C. Miers, A. Fiorese, and G. P. Koslovski, "QoS-Aware Virtual Infrastructures Allocation on SDN-based Clouds," in Proc. of the Int. Symp. on Cluster, Cloud and Grid Computing, ser. CCGRID. Madrid, Spain: IEEE, 2017.

[12] R. de Oliveira and G. P. Koslovski, "A tree-based algorithm for virtual infrastructure allocation with joint virtual machine and network requirements," International Journal of Network Management, vol. 27, no. 1, p. e1958, 2017.

[13] I. Houidi, W. Louati, W. B. Ameur, and D. Zeghlache, "Virtual network provisioning across multiple substrate networks," Computer Networks, vol. 55, no. 4, pp. 1011–1023, 2011.

[14] N. M. K. Chowdhury and R. Boutaba, "A survey of network virtualization," Computer Networks, vol. 54, no. 5, pp. 862–876, 2010.

[15] J. Rosenberg and A. Mateos, The Cloud at Your Service, 1st ed. Manning Publications Co., 2010.

[16] S. Rajagopalan, B. Cully, R. O'Connor, and A. Warfield, "Secondsite: Disaster tolerance as a service," SIGPLAN Not., vol. 47, no. 7, pp. 97–108, Mar. 2012.

[17] A. A. Ucla, A. Avizienis, J. claude Laprie, and B. Randell, Fundamental concepts of dependability. Newcastle, U.K.: University of Newcastle upon Tyne, Computing Science, 2001.

[18] L. R. Bays, R. R. Oliveira, L. S. Buriol, M. Barcellos, and L. P. Gaspary, "A toolset for efficient privacy-oriented virtual network embedding and its instantiation on SDN/OpenFlow-based substrates," Computer Communications, vol. 82, pp. 13–27, 2016.

[19] M. Chowdhury, M. Rahman, and R. Boutaba, "Vineyard: Virtual network embedding algorithms with coordinated node and link mapping," IEEE/ACM Transactions on Networking, vol. 20, no. 1, pp. 206–219, Feb 2012.

[20] V. Persico, P. Marchetta, A. Botta, and A. Pescapè, "Measuring network throughput in the cloud," Comput. Netw., vol. 93, no. P3, pp. 408–422, Dec. 2015.

[21] E. Caron and M. D. de Assunção, "Multi-criteria malleable task management for hybrid-cloud platforms," in 2016 2nd CloudTech. Marrakech, Morocco: IEEE, May 2016, pp. 326–333.

[22] M. Ficco, C. Esposito, F. Palmieri, and A. Castiglione, "A coral-reefs and game theory-based approach for optimizing elastic cloud resource allocation," Future Gen. Comp. Sys., vol. 78, pp. 343–352, 2018.