Green Cloud Architecture: Low Power Routers for an Energy-Aware Data Transport

Fatoumata B. Kasse and Bamba Gueye
Université Cheikh Anta Diop de Dakar
Dakar, Senegal

Halima Elbiaze
Université du Québec à Montréal
Montreal, Canada

***Abstract*—So far, mitigating the energy consumed by data centers has received much attention in order to increase the energy-efficiency in clouds. Nevertheless, the energy consumed by data transport represents a significant percentage according to the overall consumption of the cloud. Hence, by exploiting network and router consolidation we design and evaluate a Green Cloud Architecture (GCA), where we can either shut down, or make in sleeping mode virtual routers; or migrate virtual routers towards another physical router according to energy-awareness. Our green architecture significantly reduces the number of powered ON routers, and thus the power consumption during data transport by up to 41%.**

I. Introduction

Cloud computing is often compared to the geographic distributed computing architectures - the so-called Grid - that appeared as a trend in supercomputing and distributed computing [1]. The main goals of both architectures are to reduce computing costs, and increase flexibility, and reliability. In contrast to grid technologies, hardware and platforms can be virtualized in cloud computing. Likewise, each user has unique access to its individual virtualized environment.

The current trend is to build data centers in geographic area with access to cheap power, or have cold temperature like arctic regions, and thus, the geographic distance between users and cloud will be lengthened. Due to the effect of Internet applications, that are obviously based for instance on Web, peer-to-peer and web-based video-on-demand services, the amount of data that will be transfered over the Internet, either from users towards cloud or from cloud to users, as well according to existent cloud service models, will increase significantly [2]. Since Internet traffic grows, the required equipment to route this traffic should follow this trend, and thus a growth in power consumption of the equipment is unavoidable.

Therefore, the power consumed during transport and switching represents a significant percentage of total energy consumption in cloud computing [2], [3], [4]. Quite often the concern of cloud providers is business, and thus the energy consumed in transport and switching is not considered. Recently, in order to take into account the issues described previously, the *ETSI* (European Telecommunications Standards Institute) standards group for network functions virtualization has been created by seven of the world's leading operators [5].

In this paper, we focus on power saving strategies during transport and switching in cloud computing. Firstly, we propose a *Green Cloud Architecture (GCA)* where router virtualization is the mainstay to transport the data. Therefore, physical routers can create multiple virtual routers, as well we design mechanisms to acquire and control network resources. Since the power consumed by a router depends on the number of activate ports [2], by enabling line cards or virtual routers instances to be dormant, we can reduce the power consumed by routers as well as the overall electricity consumption due to communications. Secondly, we evaluate our GCA by designing an energy-aware resource allocation algorithm which maps the virtual routers on top of physical routers and seeks to minimize the power consumed during transport.

The remainder of the paper is organized as follows. In Section II, we survey the different studies related to energy-efficiency. Section III presents an overview of existing technologies on network and device virtualization. Next, Section IV illustrates our green network architecture which enables power savings during transport. Section V evaluates the proposed Virtual Network Embedding (VNE) heuristic into heterogeneous substrate network. Finally, Section VI concludes the paper and outlines our future work.

II. Related work

By considering three different cloud service models like software, storage, and processing, Baliga *et al.* [4] investigate the energy consumed in switching and transmission, likewise data processing, and data storage according to a public and private cloud. The observed trend is that private cloud is more efficient than public cloud. Berl *et al.* [2] suggest to optimize or to redevelop network protocols in order to achieve more energy-efficient cloud.

Several resource allocation and discovery approaches in network virtualization environment have been surveyed in [6], [7]. Recent studies like [8] showed that when the virtual nodes *(Vnodes)* and virtual links *(VL)* are embedded jointly this allows a better mapping with lower cost and less elapsed time compared to the two stage embedding approach [9]. A prior work considered an embedding which enables path splitting and link migration [10]. Nevertheless, for a better embedding as suggested by [8], their approach [10] should correlate their link migration with the previous node-mapping.

Su *et al.* [9] devised an embedding (*EA-VNE*), based on CPU and bandwidth constraints, that does not support path splitting. The heterogeneity issues between nodes and links

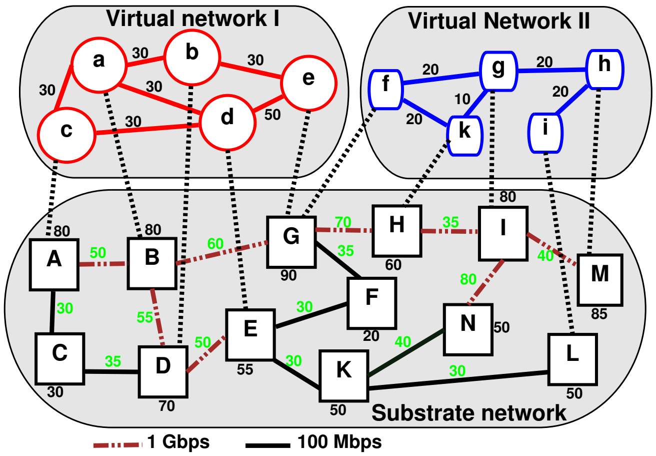


Fig. 1. Assignment of VN onto a substrate network

in terms of power consumption and bandwidth capacity for serving VN requests is considered in [11]. By considering path splitting, the authors of [11] provide a better mapping compared to [9].

III. Network/Device virtualization

By creating several virtual machines (VMs) on a physical node (substrate node), we reduce the amount of used hardware, in addition, we enhance the utilization of resources. In fact, node consolidation enables to move VMs running on multiple under-utilized nodes onto a single node, or to a minimal number of physical nodes, and hence, the remaining nodes can be set to power-saving modes.

For instance, Fig. 1 shows a network virtualization environment where two VNs topologies are embedding onto a shared substrate network (physical infrastructure). According to both VNs, the numerical values beside the link express the bandwidth requested by the VLs of these two topologies. In contrast, with respect to the substrate network, the values beside physical nodes and links represent the stresses (*i.e.*, available resource) of those nodes and links at a given time t.

A. Router consolidation

Cloud data centers can reduce the energy consumed through router consolidation [12], whereby different workload can share the same physical node using virtualization, and unused nodes can be switched off [3]. In fact, router virtualization techniques [12] allow a single physical router to create multiple virtual routers (VR), and each VR maintains multiple routing tables to serve traffic from multiple networks due to the fact that there is a separate router instances for each VR. A virtualized router called VROOM (Virtual ROuter On the Move) has been proposed in [12]. By leveraging the separation of the control plane and the data plane proposed in current routers, a given VR can be migrated from one physical router to another without disrupting the data traffic it carries.

The seamless traffic is due to the fact that the control plane of a given VR is migrated from one physical router to another, and then this control plane clones the data-plane state at the new location while continuing to update the state at the old location. The reason why no data packets loss are noticed is that VROOM forwards packet using the data plane that is loaded in the new location and the old one, hence, the asynchronous migration of link connectivity is supported. Once all links belonging to a VR that is moved are migrated, the old data plane, and outgoing links can be safely removed. Note that before the migration process, a tunnel for redirecting routing messages is established between both physical routers. For more details about the migration mechanisms we suggest to the reader to refer to [12].

IV. Green Cloud Architecture (GCA) overview

A. GCA introduction

Fig. 2 illustrates a middleware where router virtualization is the mainstay of our green cloud architecture. The overall structure of GCA is formed by: *(i)* a standard Internet Service Provider network logically split into access network, metro network and core network; *(ii)* two common three-tiered data center networks formed by the core, aggregation, and access layers [13]. The distribution of virtualized routers [12] and non-virtualized routers is performed randomly in Fig. 2. As might be expected, both types of routers can share the same topology and work together seamlessly. Clouds enable virtualized datacenters and applications that are offered as services. The proposed architecture can dispatch users traffic between the primary data center and the secondary data center with respect to the amount of traffic monitored within a given data center, or geographic proximity of cloud's customers. Therefore, we promote a high availability of applications and data access as well as performance scalability with less energy consumed by considering virtualized routers.

B. GCA objective

We aim to identify and manage cloud traffic by scheduling the Internet traffic through their destinations (*e.g.*, based on the targeted cloud services, location of data centers). Therefore, one can assign the corresponding flows to a given Line Card (*LC*) with respect to a fixed router which supports several VRs instances. Indeed, each LC is composed of a hypervisor (*i.e.* LC controller) that creates, manages, and releases VRs. Since VRs are created and managed by the hypervisor, which is hosted by LCs, obviously they have their own routing tables, forwarding tables, and local buffer memory. Each VR runs independently and it is possible to migrate, or shut down one VR without affecting the others.

For instance, Fig. 2 illustrates a load balancer located at the aggregation level whereof its role is to forward traffic towards another data center. We depicted a scenario where three users request a cloud service. For instance, "Application A", "Application B", and "Application C" can be seen as a virtual network (VN) request that should be mapped into the network. Following an optimal content routing, user's applications are redirected on both data centers by taking different paths within the network. We envisage to identify the cloud traffic with the use of the IP Header *DS* (Differentiated Service) field

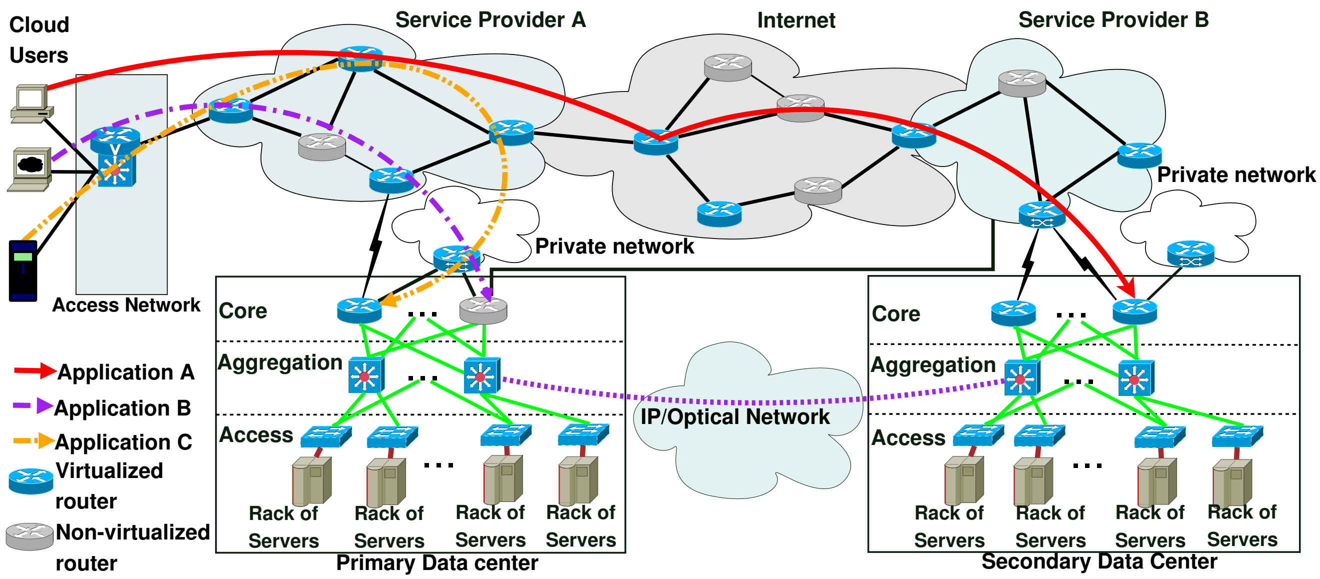


Fig. 2. Green Cloud Architecture (GCA)

as proposed in [14]. Obviously, the cloud network traffic is marked at the edge of the network by setting the DS field of the packets according to their DS value [14]. Since cloud traffic is recognized by taking into account the embedded information in the DS field, it can be isolated from other usual Internet traffic. Thereafter, a virtual routing and forwarding are performed by physical routers along the path.

By considering GCA, we are able to put some or all VRs belonging to a particular physical router either into low-energy sleep states, or shut down VRs according to traffic matrices; or migrate VRs by taking into account the power prices in different geographic locations, or migrate VRs to shift load away from congested physical links. Seeing that VRs are isolated, we can restart one VR without affecting other services on the physical router. Also, available resources can be reallocated as needed to each VR leading to more power-aware router. As consequence, we enable network consolidation, similar to server or data center consolidation, by reducing electrical processing in intermediate hops.

Of course, it is mandatory to define appropriate sleeping schedules with respect to LC. VRs instances within an area that forward traffic from cloud users towards clouds' services can probably sleep for long interval during periods of relative inactivity (*e.g.*, at nigh, week end, geographical con- straints), or short interval with respect to traffic behaviour. We should avoid packet loss, and a huge number of awake/sleep attempts. In order to reduce the number of attempts to switch ON/OFF routers, we consider the node stress metric that enables our power-aware algorithm to instantiate VN request on physical nodes that have the highest stress (Section IV-D).

C. Analytical considerations of GCA

A VNE deals with an efficient mapping of VNs onto physical network resources. The algorithm should find a set of Vnodes $\mathbb{N}^V$ onto a physical set of nodes $\mathbb{N}^S$, and a set of virtual links $\mathbb{E}^V$ onto a set of physical links $\mathbb{E}^S$. In other words, the substrate network is a graph $\mathbb{G}^S = (\mathbb{N}^S, \mathbb{E}^S)$, and the given VN to be embedded is a graph $\mathbb{G}^V = (\mathbb{N}^V, \mathbb{E}^V)$. Therefore, each substrate node (physical router) $n_i^S \in \mathbb{N}^S$ has an associated *Energy Proportionality Index* (EPI) value and a *Normalized power* capacity. EPI exhibits the potential correlation between the power consumed by a router and its load, and the *Normalized power* enables to estimate the per-bit energy consumption during the transmission and switching.

Nodes and links having maximum stress will have the highest priority during the embedding process. Indeed, the link stress ($\mathcal{LS}$) of a link $i \in \mathbb{E}^S$ represents the link utilization rates, whereas the node stress ($\mathcal{NS}$) of a given node $i \in \mathbb{N}^S$ gives its overall energy consumption. Note that the computation of both metrics are described in [11]. Consequently, according to energy-efficiency concern and the quality of service requested by the VN topology, a VL can be split up and embedded onto several substrate links according to fixed constraints. In this sense, path splitting enables to achieve power savings by harnessing substrate nodes and links that tend towards a high energy proportionality to handle VN. The possibility that VLs can be split up over multiple physical paths in favour of reducing energy consumption as well as maximizing the traffic sent through the links, while minimizing the usage of total resources of the substrate network.

D. Energy-aware resource allocation algorithm

After the computation of node and link stresses, a ranking of all nodes following their stress is performed. Basically, the node $i \in \mathbb{N}^S$ having the maximum stress will be the first potential node amongst the set of candidates for hosting a Vnode. The path cost, of each potential path between the source candidate $s \in v.Candidates$ towards all other

Algorithm 1 Energy-aware embedding algorithm (Upon i'th VN arrival)

Inputs:
$\mathbb{G}^S = (\mathbb{N}^S, \mathbb{E}^S)$: substrate topology;
$\mathbb{G}_i^V = (\mathbb{N}_i^V, \mathbb{E}_i^V)$: VNet topology;
Output: VNet$_{Embed}$ (Embedded Virtual Network)
1 Rank Vnodes $v \in \mathbb{N}_i^V$ according to their number of candidates $n \in \mathbb{N}^S$
2 Firstly assign nodes $v \in \mathbb{N}_i^V$ that have fewer substrate candidates nodes
3 **foreach** Node $v \in \mathbb{N}_i^V$ **do**
4 **foreach** Link $k \in \mathbb{E}_i^V$ connected to v **do**
5 LinkedVNode=GetLinkDestination(k)
6 **foreach** $SourceCandidate\ s$ in $v.Candidates$ **do**
7 CostNRG(s,d) = 0
8 **foreach** $DestCandidate\ d$ in *LinkedVNode.Candidates* **do**
9 PathCost(s,d) = CostNRG(s, d)
10 **end for**
11 CostNRG(s,d) = CostNRG(s,d) + $\frac{\sum_{e \in L(s,d)} \mathcal{LS}(t_i^-,e) \times \mathcal{NS}(t_i^-,s)}{Count(v.Candidates)}$
12 **end for**
13 **end for**
14 v.Embed = s | Path-cost is minimized with respect to power savings
15 **end for**

destinations $d \in LinkedVNode.Candidates$, is estimated by taking into consideration line from 3 to 12 (Algorithm 1): (i) the node stress of each physical node that acts as potential source node with respect to the path that joins a fixed Vnode denoted as destination; (ii) the link stress of the set of substrate links $L(s,d)$ that may form the path towards the potential substrate node that hosts the Vnode; (iii) the number of substrate nodes that met constraints for hosting a fixed Vnode in order to treat fairly the substrates nodes owning a different number of possible VLs. Thereby, the candidate node that allows a path with energy savings (i.e lowest CostNRG) is selected (line 14).

V. Performance evaluation

A. Simulation settings

Table I shows the average power consumed by the different components of two given routers such as the *Cisco GSR 12008* and the *Cisco 7507*. The LC consumptions illustrated in Table I are derived from [15]. According to Table I, the units of column labeled "*Power*" and "*NPower*" are expressed in Watts and Watts/Mbps respectively. The term "*Qty*" returns the number of each LC type installed in the router. Also, "*FE*" and "*GE*" means Fast Ethernet and Gigabit Ethernet respectively.

For the GSR 12008, its idle state consumes approximately 430 Watts, and the 7507 idle state consumes approximately 210 Watts [15]. Here, the idle state refers to the chassis power consumption of a given router. The values of both metrics EPI and $NPower$ are computed according to the formulae illustrated in [11] . The corresponding EPI of both routers are quite similar but an important gap exists according to the metric Normalized Power ($NPower$). The observed trend is that the GSR has a better energy proportionality compared to the 7507 router.

We implemented our power-aware resource allocation algorithm in Matlab. Our discrete event simulator considers the same parameters depicted in [10] during the extensive simulation experiments. In this respect, the substrate network is a 100-node 354-link random topology generated by the *GT-ITM* tool. Physical nodes are chosen randomly as a GSR or a 7507 router. We assume that VN topology requests arrive in a Poisson process with an average rate $\lambda = 5$ VN requests per time unit. These requests are gathered during a fixed time window equal to 10 time units and processed at next time window as in [10]. In this setup, when the resource constraints of the Vns are satisfied they have an exponential service time with an average of $\mu = 10$ time units. Otherwise, the failed requests for instantiation of VNs will be enqueued and rescheduled at an appropriate time.

Besides, the number of Vnodes is uniformly distributed from 2 to 10, and each pair of Vnodes is randomly connected with probability 0.5 (*i.e.*, for n-node VN, we have $n(n-1)/4$ links). In contrast to previous works [6] we set up an admission control mechanism. We run our simulation for 500 time units, which corresponds to about 2500 requests on average in one instance of simulation. Vnodes can request a capacity bandwidth equal either to 100 Mbps or 1Gbps.

B. Results

Fig. 3 shows the Cumulative Distribution Function (CDF) of the mean activated line cards per node for different VN arrival rate (λ). Note that λ varies from 5 to 40, as well during this experiment the time window is fixed to 10 time units. Again, Fig. 3 illustrates the efficiency of our energy-aware resource allocation algorithm. Indeed, our VNE heuristic reduce the resources usage by turning on the smallest number of LCs with respect to the network load. For $\lambda = 5$, on average 20% of nodes have zero activated LC which means an important

(a) Cisco GSR 12008

Line card (LC) type	Power	Qty	EPI (in %)	NPower
4 Port GE	92	2	43.12	0.058
4 Port OC-12/POS	72	1		
1 Port OC-48/POS	70	1		

(b) Cisco 7507

Line card (LC) type	Power	Qty	EPI (in %)	NPower
1 Port FE	26	3	42.77	0.085
4 Port GE	30	1		
1 Port 1.544 Mbps DS1	49	1		

TABLE I
ROUTER POWER CONSUMPTION SUMMARY.

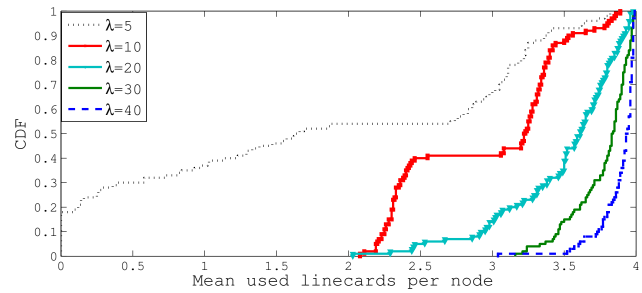
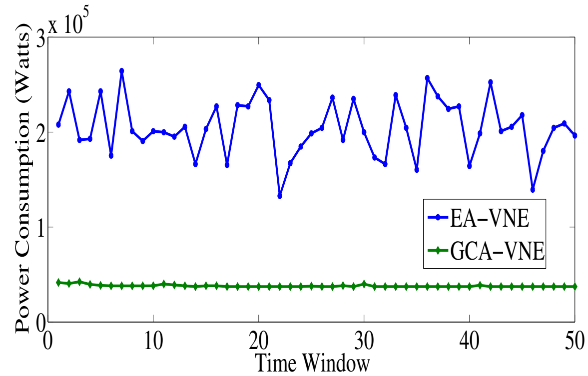


Fig. 3. CDF of activated line cards vs. load

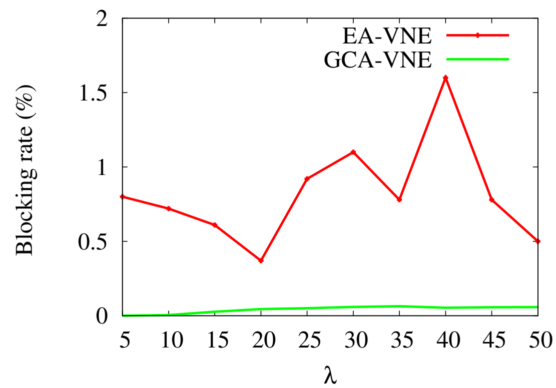
energy saving. On the other hand, the used LCs per node increase according to the rise of VN arrival rate.

In order to evaluate the performance of our algorithm, we implemented the *EA-VNE* algorithm [9]. We considered the same evaluation settings as used in [9], except the number of substrate nodes which is fixed at 100. Fig. 4(a) depicts the energy consumed by the overall network when the VNs are embedded according to an energy concern. A non-efficient network means a network where all networking devices are running according to their maximum energy. It is easy to verify that after 10 time windows the power consumption of GCA-VNE is in its steady state. The average energy consumption for GCA-VNE (resp. EA-VNE) is roughly equal to $37,000$ watts (resp. $196,000$ watts), whereas the energy consumed by a non-efficient network is always equal on average to $63,000$ watts (resp. $300,000$ watts). Indeed, GCA-VNE (resp. EA-VNE) can save up to 41% (resp. 35%) power cost with respect to a non-efficient network. This gain is due to the fact that GCA-VNE leverages path splitting.

Fig. 4(b) illustrates the percentage of rejected VN according to the average rate λ that varies from 5 to 50. Note that, the VN requests are gathered during a fixed time window 10, and processed at next time window. The percentage of rejected VN in EA-VNE is higher than GCA-VNE. When $\lambda = 5$ all arrival VNs are accepted for GCA-VNE approach in contrast to EA-VNE where 0.8% of virtual networks are rejected. Note that for λ values up to 40 the blocking rate is roughly in its steady phase with respect to GCA-VNE.



(a) Overall network power consumption



(b) Blocked Virtual Networks

Fig. 4. Comparison between GCA-VNE and EA-VNE.

VI. Conclusion

The virtualization of physical routers involved in the transport of data contributes to reducing the energy consumed by routers as well as the overall energy consumed in transport. Therefore, we formulated a power-aware VNE algorithm that considers path splitting and the support of heterogeneous virtual and physical networks. Furthermore, GCA-VNE promotes the assignment of links and nodes simultaneously. The obtained results illustrate that our resource allocation algorithm (CGA-VNE) can save up to 41% of energy cost compared to an unaware-energy network. Ongoing work aims to provide a simulation platform which models and simulates GCA with respect to end-to-end Cloud users environment. As future work we plan to take into consideration node and link migration, as well to deal with potential substrate link failures.

References

[1] I. Foster, Y. Zhao, I. Raicu, and S. Lu, "Cloud Computing and Grid Computing 360-Degree Compared," in *IEEE Grid Computing Environments Workshop*, November 2008, pp. 1–10.

[2] A. P. Bianzino, C. Chaudet, D. Rossi, and J.-L. Rougier, "A survey of green networking research," *IEEE Communications Surveys and Tutorials*, vol. 14, no. 1, pp. 3–20, 2012.

[3] R. Bolla, R. Bruschi, F. Davoli, and F. Cucchietti, "Energy efficiency in the future Internet: A survey of existing approaches and trends in energy-aware fixed network infrastructures," *Proc. IEEE Communications Surveys & Tutorials*, vol. 13, no. 2, pp. 223–244, 2011.

[4] J. Baliga, R. W. A. Ayre, K. Hinton, and R. S. Tucker, "Green Cloud Computing: Balancing Energy in Processing, Storage and Transport," *Proc. IEEE*, vol. 99, no. 1, pp. 149–167, January 2011.

[5] ETSI, "Leading operators create ETSI standards group for network functions virtualization," 22 January 2013 - http://www.etsi.org/news-events/news/644-2013-01-isg-nfv-created.

[6] A. Belbekkouche, M. Hasan, and A. Karmouch, "Resource discovery and allocation in network virtualization," *IEEE Communications Surveys & Tutorials*, no. 99, pp. 1–15, June 2012.

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

[8] J. Nogueira, M. Melo, J. Carapinha, and S. Sargento, "Virtual network mapping into heterogeneous substrate networks," *IEEE Symposium on Computers and Communication*, pp. 438–444, June 2011.

[9] S. Su, Z. Zhang, X. Cheng, Y. Wang, Y. Luo, and J. Wang, "Energy-aware virtual network embedding through consolidation," in *INFOCOM Workshops*, 2012, pp. 127–132.

[10] M. Yu, Y. Yi, J. Rexford, and M. Chiang, "Rethinking virtual network embedding: substrate support for path splitting and migration," *SIGCOMM Comput. Commun. Rev.*, vol. 38, no. 2, pp. 17–29, Mar. 2008.

[11] F. B. Kasse, B. Gueye, and H. Elbiaze, "Leveraging network virtualization for energy-efficient cloud: Future directions," in *Proc. IEEE LCN*, Edmonton, Canada, September 2014, pp. 494–497.

[12] Y. Wang, E. Keller, B. Biskeborn, J. van der Merwe, and J. Rexford, "Virtual routers on the move: live router migration as a network-management primitive," in *Proc. SIGCOMM*, Seattle, 2008, pp. 231–242.

[13] Q. Zhang, L. Cheng, and R. Boutaba, "Cloud computing: state-of-the-art and research challenges," *Journal of Internet Services and Applications*, vol. 1, no. 1, pp. 7–18, May 2010.

[14] M. Mohit, "A comprehensive solution to cloud traffic tribulations," in *International Journal on Web Service Computing*, vol. 1, no. 2, December 2010, pp. 1–13.

[15] J. Chabarek, J. Sommers, P. Barford, C. Estan, D. Tsiang, and S. Wright, "Power awareness in network design and routing," in *Proc. INFOCOM*, April 2008, pp. 457–465.