A Feasible Solution to Provide Cloud Computing over Optical

Citation:
Mina Taheri, and Nirwan Ansari. "A feasible solution to provide cloud computing over optical networks." *Network, IEEE, vol. 27, no. 6, pp. 31-35, 2013.*

URL:
http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=6678924

A Feasible Solution to Provide Cloud Computing over Optical Networks

Mina Taheri and Nirwan Ansari, *Fellow, IEEE*
Advanced Networking Laboratory
Department of Electrical and Computer Engineering
New Jersey Institute of Technology
Newark, NJ 07102, USA
Email: (mt243, nirwan.ansari)@njit.edu

Abstract

Cloud computing provides a whole new way to store, access, and exchange information, and introduces new service patterns and business opportunities for network providers and enterprises. Owing to the expensive intercommunication among data centers, the difficulty of managing diverse backups, and the cost of operating several data centers, the cloud resources are mostly limited to centralized architectures. However, achieving low latency and guaranteeing end to end performance are two main challenges in today's cloud computing. Meanwhile, PON-based broadband access networks have been widely considered as one of the most promising solutions for the future telecom-access platform that supports residential, enterprise, and mobile backhaul services. The latest research has achieved terabit symmetric capacity which can support up to 800 ONUs at gigabit rates. Therefore, deploying cloud services based on PON access networks provides flexibility in moving services closer to customers so as to ensure low latency and guarantee quality of experience (QoE) for the customers. In this paper, we propose, by incorporating PON OLT and ONU physical resources, an Infrastructure as a Service (IaaS) architecture which enables the delivery of cloud services to end users over PON access networks.

Index Terms

Cloud computing; Optical networks; Infrastructure as a Service; Cloud resources.

I. Introduction

Passive Optical Network (PON) has been constructed as a point-to-multipoint architecture which connects several geographically distributed Optical Network Units (ONUs) to an Optical Line Terminal (OLT) located at a central office via installed fiber. Instead of deploying several central offices, PON cuts down the equipment expenses to just one. The number of connected ONUs varies from 16 to 256. Various passive optical networks (PONs) [1] , including currently deployed Ethernet PON (EPON) and Gigabit PON (GPON) as well as next generation Wavelength Division Multiplexing (WDM) PON and Orthogonal Frequency Division Multiplexing (OFDM) PON have been introduced. PON's ability to provide broadband and secure access for end users attracts most of the service providers to consider it as an option for the backhaul network, which connects the backbone network to the edge subnetworks consisting of wired LANs, wireless access networks, cellular mobile base stations, and base station control points.

Cloud computing refers to the use of computing resources including hardware and software on third-party facilities instead of local computers. Cloud computing relies on resource sharing among servers in order to provide computing and storage services to a large number of end users. The current implementation of cloud computing mostly relies on large and centralized data centers. A long distance between end users and cloud servers in such architectures

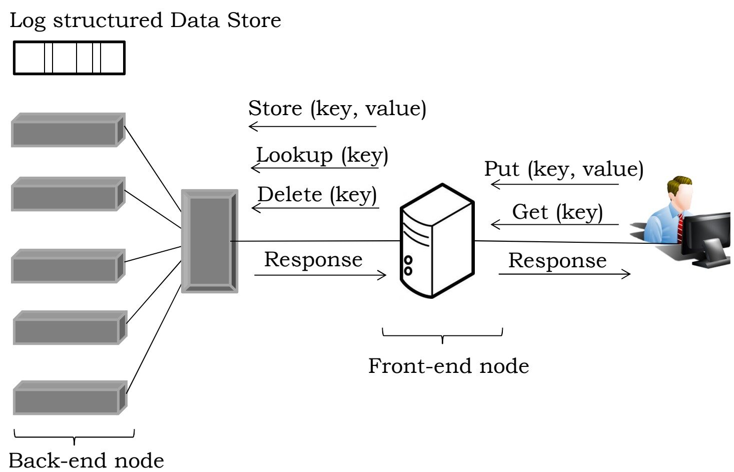

Fig. 1. FAWN-KV Architecture

impose considerable delay to the system. Yet, centralized architectures are not flexible to move the resources closer to customers. Owing to costly heat loss, temperature control, and need for resource over provisioning, distributed clouds are emerging as the eminent choice for cloud computing [2].

The idea of carrier cloud (http://www.slideshare.net/alcatellucentcloud/alcatellucentcloudthe carriercloudwhitepaper) allows customers to access service provider data center compute and storage resources through service provider WAN access networks. Service providers can allocate the cloud resources to customers in an optimum fashion by employing network management software. By combining the network and the cloud, the carrier cloud can improve performance, security, and availability in comparison to the existing cloud.

A PON-based broadband access network has been recently considered as one of the most inspiring options for the future telecomm-access platform to support residential, enterprise and mobile backhaul services. Owing to the vast deployment of PONs by major telecommunications companies, the strong capability of PON to provide high bandwidth with lower operating expenses has been proved [3]. Therefore, one can imagine that deploying cloud services based on PON access networks provides the flexibility in moving services closer to customers so as to ensure low latency and guarantee quality of experience (QoE) for the customers.

A Storage Area Network (SAN) is a high speed and special purpose network which provides large databases with fast data access by connecting a set of storage devices with their servers. To avoid server damage and recover the data after disasters like earthquakes, fire, and power outages, the storage sites should be separated far apart to ensure business continuity so that the probability of having all sites affected is rather low. SAN extension brings some challenges to the access networks. Extending SAN over a passive optical network (S-PON) has addressed some of these challenges such as scalability problems, bandwidth bottlenecks and cost issues [4]. Designing S-PON architecture based on the P2MP PON infrastructure addresses scalability and cost issues. Moreover, to tackle the bandwidth bottlenecks, three transmission technologies, Time Division Multiple Access (TDMA), Sub Carrier Multiple Access (SCMA), and Wavelength Division Multiple Access (WDMA) have been proposed to carry storage signals with gigabit-level transmission. Therefore, delivering SAN traffic to the long-haul at the rate of 2.5 Gb/s is becoming viable through S-PON technology.

One of the architectures that integrates clouds with an access network is presented in [5]. In order to access any service in a hybrid wireless-optical broadband access network (WOBAN), the request needs to pass all the way from Wireless Mesh Network (WMN) and

PON to some servers in the application service provider's domain. By integrating cloud components with wireless nodes, Cloud over WOBAN (CoW) could offload traffic from gateways, reduce delay and increase offered load of the network by locally serving the request.

In order to make the best usage of optic-fiber access, we investigate the feasibility of implementing cloud services over PON. Computing capability, storage capability and interconnection capability are the basic factors in evaluating the cloud computing performance. Although the limited amount of processing and storage capacity of ONUs and OLTs cannot meet cloud expectations, managing PON resources to coordinate distributed services appears to be a promising technology for cloud services.

Anderson *et al.* [6] proposed the Fast Array of Wimpy Node (FAWN) architecture to build a cost effective cluster for data intensive computing. The proposed architecture uses a large number of nodes with low processing capability and storage as the core of a large system. Using wimpy processors not only reduces I/O-induced idle cycles, but also executes more instructions per joule than their faster corresponding processors.

Owing to the recent worldwide deployment of PON as an energy efficient access network, providing cloud computing over PON in a low cost and efficient mode is desirable. In this paper, we propose a novel infrastructure to enable a PON system to provision not only as an access network but also as a cloud backhaul. Considering an ONU in PON as a wimpy node in FAWN, we can provide efficient, fast and cost-effective access to a large amount of random-access data in the optical access networks.

The remainder of the paper is organized as follows. Section II investigates the proper required memory for providing storage over PON. Section III overviews previously proposed FAWN architecture. Section IV outlines different layers of the cloud computing model. Available PON resources for cloud computing are elaborated in Section V. In Section VI, the feasibility of implementing FAWN on PON access networks is discussed. Finally, we conclude our paper in Section VII.

II. Comparison of different memory storage

In order to build high performance and energy-efficient cluster systems, FAWN combines low power CPUs with small amounts of local flash storage [6]. RAMCloud [7] architecture builds a software platform that aggregates the memory of a large number of commodity servers to host all the application data in a datacenter. Since all the information is kept in DRAM, RAMCloud can provide durable and available storage with 100 to 1000 times lower access latency and greater throughput than disk-based storages. The most obvious drawbacks of RAMCloud are high cost per bit and high energy use per bit. For both metrics, RAMCloud storage will be 50 to 100 times worse than a pure disk-based system and 5 to 10 times worse than a storage system based on flash memory. A RAMCloud system also requires more floor space in a data center than a system based on disk or flash memory. Thus, if an application must store a large amount of data inexpensively and has a relatively low access rate, RAMCloud is not the best solution.

Flash memories are not only much faster than disks and cheaper for the same amount of DRAM, but also consume less power than both. In the proposed architecture, a cluster-based key-value storage (FAWN-KV) is used in order for the wimpy nodes to act as the core of a large system. Andersen *et al.* [6] showed that a FAWN cluster can provide over 350 queries/Joule, thus presenting a great potential for I/O intensive workloads.

III. An overview of FAWN architecture

A. FAWN Data Storage

As depicted in Fig.1, the FAWN system consists of a large number of back-end nodes and some front-end nodes. In the key-value store data structure, data consists of two parts. The

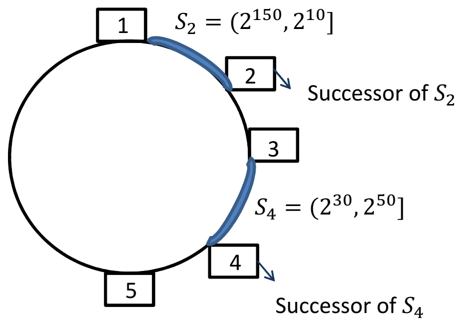

Fig. 2. Consistent Hashing

first part is a string which is considered as the key (e.g., a person's name), and the second part is the actual data which represents the value in the key-value relationship (e.g., an Email address). FAWN-KV is significantly designed based on a log structure per back-end node datastore, which is referred to as FAWN-DS. These nodes are organized into a ring, and keys are mapped to these nodes by using consistent hashing [8] as follows. Every key and every node has a 160-bit identifier. A node's identifier is generated by hashing the node's IP address, and a key's identifier is allocated by hashing the key. Nodes and keys are ordered based on their identifiers and placed on a circle module 2^{160}. A key will find its position in the ring based on its identifier and walk clockwise to meet the first node whose position is larger than the key's position. This node is called the "successor" node of this particular key. Therefore, each key is mapped to its successor. In doing so, a range of keys could be mapped to every node, and values for the key range are stored in the node's flash storage. Fig. 2 illustrates the assignment of the key ranges to their successors. As depicted in the figure, the successor of the keys belongs to range S_2 in node 2, and similarly node 4 is the owner of keys in the range of S_4. Using this identifier circle for storing data in optical grid was previously proposed in [9].

By using a standard Put and Get interface, shown in Fig.1, each user sends its requests to a front-end node. The front-end node, based on the key, sends the request to the specific back-end node. The back-end node responds to the request by using its FAWN-DS, which is well designed particularly for wimpy nodes with a limited amount of DRAM. Writing to a flash storage is sequential while reading from the flash storage needs a single random access. This log-structure data storage supports Lookup, Store, and Delete functions. An in-memory (DRAM) Hash Index is used to map 160-bit keys to a stored value in the storage.

B. Replication and Consistency

In order to provide consistency and reliability, FAWN-KV uses chain replication [10]. In the chain replication procedure, data are stored in three consequent nodes. The front-end node sends a Put request for a particular key to its "successor node" which becomes a head of the chain. The head node stores the value corresponding to this key in its data storage and forward it to the next node in the ring. Likewise, the second node stores and forwards the request to the third node which is the tail node. The tail node stores the value and sends the response to the front-end node. To ensure reliability, the nodes in the chain keep the Put request until they receive the acknowledgement from the user side. This operation is simple as nodes write the request sequentially in log storage, and requests are sent in-order across the chain.

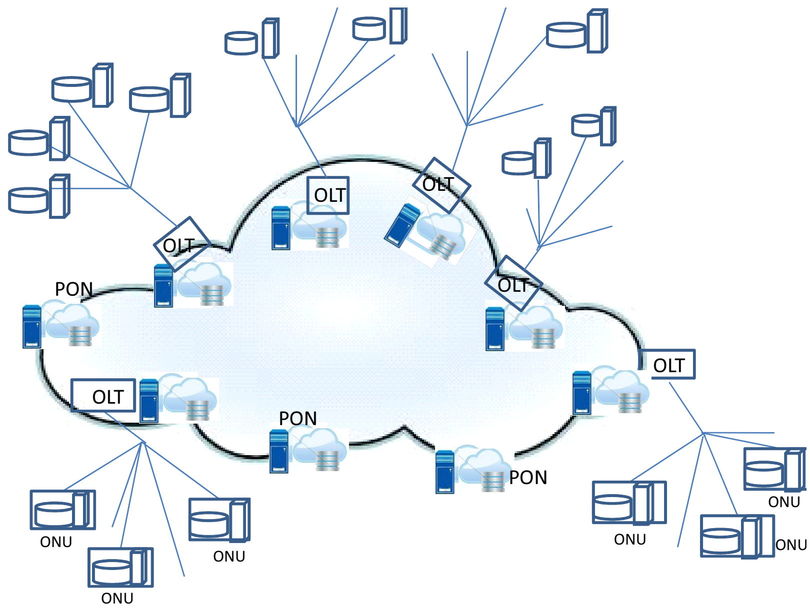

Fig. 3. Viable PON Resources for Cloud Computing

IV. CLOUD COMPUTING LAYERS

Cloud generally can be categorized into three different layers based on the service they provide: Infrastructure as a Service(IaaS), Platform as a Service(PaaS), and Software as a Service(SaaS) [11]. Similar to the seven layers of the OSI model in networking, each layer of the cloud computing model is conceptually related to the previous layers.

IaaS, which is also referred to Hardware as a Service (HaaS), provisions hardware, storage, virtual machines, servers, and networking components; it connects all of the resources to deliver software applications. Therefore, the IaaS service provider is responsible for hosting, configuring, and maintaining the equipment. IaaS customers can create or remove virtual machines and network them instead of purchasing servers or hosted services. Customers are charged according to the consumed resources. Amazon EC2 (Elastic Compute Cloud) and Amazon S3 (Simple Storage Service) are two examples of IaaS vendors.

Platform as a Service (PaaS) delivers a computing platform and solution stack as a service. PaaS ensures the providers that different environments get resources (hardware, operating systems, storage and network capacity) properly. PaaS basically provisions a means to rent different resources over the internet. Customers, who pays for platform services, do not need to manage an operating system. They just create their own applications within a programming language which is hosted by the platform services. Google's App Engine is an example of PaaS by which clients can build web applications and deploy them on Google servers.

SaaS provides on-demand applications over the internet by employing multitenant architecture and complex caching mechanisms. These applications may include email, Customer Relationship Management (CRM), and other office productivity applications. Some type of services like email is provided to the customers for free while enterprise services have to pay monthly or by usage. For instance, Salesforce is an industry leader which provides CRM services.

From the above discussion, it is inferred that IaaS plays a key role in the cloud computing environment. The primary step of providing cloud services is to check and manage the available resources. Deploying cloud services on access networks like PONs first requires the construction of an infrastructure.

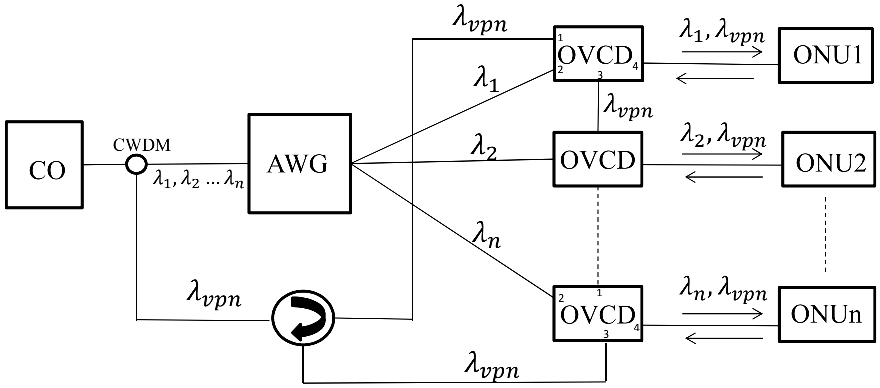

Fig. 4. ONU to ONU Communication

V. PON RESOURCES FOR CLOUD COMPUTING

Luo *et al.* [12] briefly discussed the suitability of PON for cloud computing. A PON OLT chassis contains several OLT cards. Each OLT card consists of several OLT ports. Every OLT port is connected to a set of ONUs. The number of ONUs connected to an OLT varies from 16 to 256. ONUs can lie at different distances from OLT. A typical PON can support distances of up to 20 km while a distribution distance over 60 km can be achieved through Long-Reach PONs. Each OLT port with its connected ONUs is considered as a PON system. PON OLT manages the resources of these distributed ONUs.

The processing, storage, and interconnection capability of a system are the three major factors in affecting the cloud computing performance. Considering the Dhrystone benchmark, the CPU performance of each component is measured by Dhrystone million instructions per second (DMIPS). A single OLT card can provide up to 3065 DMIPS processing capability and up to 256 MB storage capacity. Every ONU in a PON system can also be equipped with up to 1260 DMIPS CPU and 128 MB memory capacity. Owing to different PON technologies, the interconnection speed between OLT and ONUs could be 1, 1.25, 2.5, or 10 Gb/s. Candidate architectures for next-generation PON are evaluated in [13].

As depicted in Fig. 3, each PON possesses a noticeable amount of storage and processing resource in a cloud. Since a large number of PON structures is deployed, a significant amount of cloud computing resource can be realized in the access networks. In a chassis configuration of 16 cards, 8 ports, and 256 split per PON, the associated ONUs can provide a processing capability of up to $41,287,680$ DMIPS and a storage capability of up to $4,194,304$ MB. Considering 2.5 Gb/s as the interconnection speed, the feasibility of providing cloud computing over a PON system is readily attainable. Thus, a network of distributed weak servers which are connected by gigabit per second fiber optics can act as a single large scale server. Accordingly, researchers at Berkeley claimed that Network of Workstations (NOW) can provide better cost-performance for parallel applications than a massively parallel processing architecture (MPP) [14].

VI. FAWN OVER PON

Since the requested services from the users side are mostly local, deploying an infrastructure-based integration of cloud with an access network is promising. As discussed in Section V, each PON consists of some ONUs with limited resources. Considering a chassis, we have a large number of these constrained ONUs which can act like the wimpy nodes (front-end nodes) in FAWN (Fig. 1). By attaching a small amount of low-cost flash memory to each ONU, FAWN could be implemented in PONs to provide fast and energy efficient processing of random read-intensive workloads. OLT in this scheme plays the same role as the front-end node in FAWN.

In order to apply chain replication in PON, ONU to ONU communication is required. By using low cost optical passive components and OFDMA technology, Deng *et al.* [15] proposed a WDM-PON architecture to support intra-PON and inter-PON ONU to ONU networking. In the designed WDM-PON architecture, Fig. 4, a specific wavelength (λ_{vpn}) is assigned to transfer Virtual Private Network (VPN) traffic. In order to combine and separate the optical VPN signals and up/downstream traffic, a Coarse Wave Division Multiplexing (CWDM) filter is used.

An Optical VPN Circulator Device (OVCD) is used to lead the traffic to the right port. Based on the input port, traffic in OVCD is emitted to a related output port. Downstream traffic and VPN traffic, which enter port 1 and port 2, respectively, are exiting from port 4. The entered VPN traffic from port 4 is sent to port 3, and upstream traffic from port 4 is transmitted to port 2. Upon receiving VPN and control data, the ONU finds out the subcarrier on which the traffic is transmitted. Then, it regenerates an OFDMA signal and sends the VPN traffic to the next ONU (Fig. 4). After passing all the ONUs, VPN traffic can go to other WDM-PON systems through Central Office (CO). By doing so, a fault tolerance configuration model is achieved.

As depicted in Fig. 5, users send their Put and Get requests to the system through their connected ONUs. Each ONU forwards the arriving requests to the OLT (front-end node). Afterwards, based on the request's key, the OLT sends the Put request to the ONU (back-end node) responsible for that specific key. This ONU becomes the head of the replication chain. Using ONU to ONU communication, data are stored in three consequent ONUs in the chain. After storing data, the tail ONU responds to the OLT. For the Get request, the OLT directly sends the request to the tail node of the chain that responds to the request. The OLT collects the responses of the users' requests and returns them to the ONUs. Owing to the high-speed interconnections between the OLT and the ONUs, the imposed communication delay is negligible. To balance the stored data and make the system heterogeneous, each ONU can provision multiple virtual ONUs which are placed in multiple positions. In this case, every virtual ONU has a virtual ID. Basically, each physical ONU is responsible for multiple different key ranges. When a particular key is mapped to its successor, the value associated to the key will be stored in the flash memory of the successor's physical ONU.

In order to avoid congestion caused by intensive calling of one or a few keys, the OLT can maintain a small, high speed query cache which helps reducing the outgoing load to a single ONU.

The Lookup operation in each ONU works as follows: based on the available key, a 4-byte pointer to the location of stored data in Data Log and a valid bit will be extracted from Hash Index, and the corresponding data will be retrieved from the Data Log. During the Store process, an entry data will be appended to the log, the corresponding pointer of the hash table entry will be updated to the new offset, and the valid bit will be set to true. Delete makes the corresponding hash entry of the key invalid. Deleting the entries is similar to Store with the difference of writing a *Delete* entry to the end of the data log and setting the valid bit to false. The previous entry of this key will be invalidated, and space will be reclaimed upon performing garbage collection in a log-structure file system.

Fig. 5 illustrates the procedure of serving a user over PON cloud. The user sends the Put or Get request to the OLT through ONU1. Based on the key, the OLT sends a Put request to its successor ONU (ONU4). Considering the replication concept, the value of the key is stored in three ONUs. Therefore, the Store operation of the Put request from ONU1 is performed in ONU4, ONU5, and ONU6 sequentially while the Lookup operation of a Get request is done in ONU6. ONU6 sends the *Response* message back to the OLT, and the message is sent from

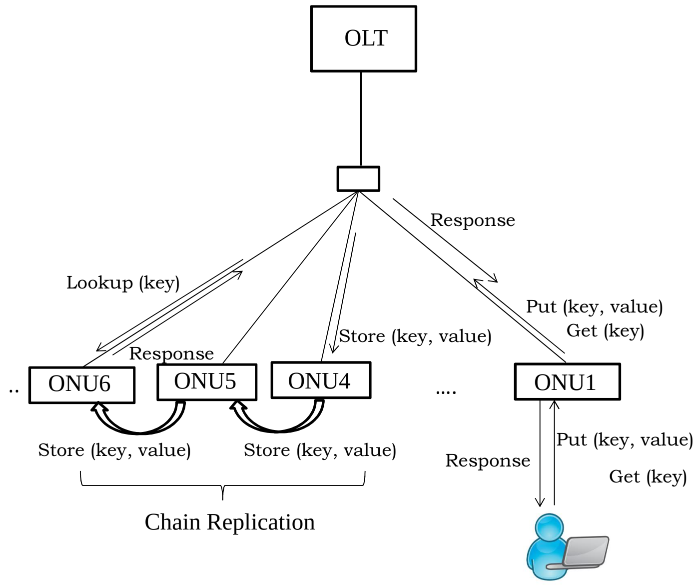

Fig. 5. Serving a Cloud Request over PON

the OLT to the user through ONU1.

Although the proposed chain replication is illustrated in WDM PON, it can potentially be deployed in TDM PON. Instead of sending the Put request to the head ONU, the OLT sends data to the three consecutive ONUs in TDM PON. Data is stored in all the three nodes and the third node responds to the OLT. The only remained issue is the increase of traffic load from the OLT to the ONUs. Storing data in one, two, or three nodes for providing consistency in TDM PON can be adaptively changed based on the real time traffic load.

Since the FAWN architecture has been previously demonstrated, the proposed scheme similarly provides a viable approach to efficiently run I/O-bound data intensive and computationally simple high query rate applications at a low cost.

VII. Conclusion

In this paper, the feasibility of providing cloud services in optical networks and particularly in WDM-PON has been discussed. A large number of already deployed computing and storage resources in PONs along with a high interconnection speed make PON systems a suitable backhaul to cloud services. By considering available resources in PON, we have proposed to deploy a substantiated architecture for managing PON resources in an efficient manner. Locally serving the clients in the access networks not only decreases the response time and increases the performance, but also offloads the traffic from links toward the data centers and frees up capacity for other traffic. We have suggested an infrastructure based architecture to take the first step of implementing cloud computing in PONs.

References

[1] N. Ansari and J. Zhang, *Media Access Control and Resource Allocation for Next Generation Passive Optical Networks*. Springer, 2013.

[2] P. T. Endo, A. V. de Almeida Palhares, N. N. Pereira, G. E. Goncalves, D. Sadok, J. Kelner, B. Melander, and J. Mangs, “Resource allocation for distributed cloud: Concepts and research challenge,” *IEEE Network*, vol. 25, no. 4, pp. 42–46, 2011.

[3] Y. Luo, N. Yin, N. Ansari, and T. Wang, “Resource management for broadband access over TDM PONs,” *IEEE Network*, vol. 21, no. 5, pp. 20–27, 2007.

[4] S. Yin, Y. Luo, S. Rago, J. Yu, N. Ansari, and T. Wang, "Storage area network extension over Passive Optical Networks (S-PONs)," *IEEE Communications Magazine*, vol. 46, no. 1, pp. 44–52, 2008.

[5] A. Reaz, V. Ramamurthi, and M. Tornatore, "Cloud-over-WOBAN (CoW): an offloading-enabled access network design," in *IEEE International Conference on Communications (ICC)*, Univ. of California, Davis, CA, USA, Jun 2011, pp. 1–5.

[6] D. Andersen, J. Franklin, M. Kaminsky, A. Phanishayee, L. Tan, and V. Vasudevan, "Fawn: A Fast Array of Wimpy Nodes," in *ACM SIGOPS 22nd Symposium on Operating Systems Principles*. ACM, 2009, pp. 1–14.

[7] J. K. Ousterhout, P. Agrawal, D. Erickson, D. Kozyrakis, J. Leverich, D. Mazieres, S. Mitra, A. Narayanan, M. Rosenblum, S. M. Rumble, E. Stratmann, and R. Stutsman, "The case for ramclouds: Scalable high-performance storage entirely in dram," *ACM SIGOPS Operating Systems Review*, vol. 43, pp. 92–105, 2010.

[8] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, "Chord: A scalable peer-to-peer lookup service for internet applications." in *ACM SIGCOMM*, vol. 31, no. 4, 2001, pp. 149–160.

[9] L. Liu, X. Hong, J. Wu, and J. Lin, "Experimental investigation of a peer-to-peer-based architecture for emerging consumer grid applications," *Journal of Optical Communications and Networking*, vol. 1, no. 1, pp. 57–68, 2009.

[10] R. van Renesse and F. B. Schneider, "Chain replication for supporting high throughput and availability," in *6th conference on Symposium on Operating Systems Design and Implementation*, vol. 6, 2004.

[11] S. Bhardwaj, L. Jain, and S. Jain, "Cloud computing: A study of infrastructure as a service IaaS,," *Intl J. Eng. and Information Technology*, vol. 2, no. 1, pp. 60–63, 2010.

[12] Y. Luo, F. Effenberger, and M. Sui, "Cloud computing provisioning over passive optical networks network design," in *First IEEE International Conference on Communications in China (ICCC2012)*, Beijing, China, 2012.

[13] J. Zhang, N. Ansari, Y. Luo, F. Effenberger, and F. Ye, "Next-generation PONs: A performance investigation of candidate architectures for next-generation access stage 1," *IEEE Communications Magazine*, vol. 47, no. 8, pp. 49–57, 2009.

[14] T. E. Anderson, D. E. Culler, and D. A. Patterson, "A case for NoW (Networks of Workstations)," *IEEE Micro*, vol. 15, no. 1, pp. 54–64, 1995.

[15] L. Deng, Y. Zhao, X. Pang, X. Yu, D. Liu, and I. T. Monroy, "Intra and inter-PON ONU to ONU virtual private networking using ofdma in a ring topology," in *2011 IEEE International Topical Meeting on Microwave Photonics Conference*, Singapore, Oct. 2011, pp. 176–179.

Nirwan Ansari (S78-M83-SM94-F09) is Professor of Electrical and Computer Engineering at NJIT. His current research focuses on various aspects of broadband networks and multimedia communications. He co-authored two books and co-edited a reference book. He has contributed over 450 technical papers, over one third published in widely cited refereed journals/magazines. He was also granted over twenty US patents. He has served on the Editorial Board and Advisory Board of nine journals.

Mina Taheri Hosseinabadi Received the B.Sc degree in electrical engineering with focus on electronics and the M.Sc degree in electrical engineering with specialization on communication (Network) from Isfahan University of Technology (IUT) in 2007 and 2010, respectively. Currently, she is working toward her Ph.D degree in electrical engineering at New Jersey Institute of Technology (NJIT), Newark. Her research interests includes energy consumption, optical network architectures and cloud computing.