

Green Virtual Networks for Cloud Computing

Ruay-Shiung Chang

Dept. of Computer Science and Information Engineering
National Dong Hwa University
Hualien, Taiwan, R. O. C.
rschang@mail.ndhu.edu.tw

Chia-Ming Wu

Dept. of Computer Science and Information Engineering
National Dong Hwa University
Hualien, Taiwan, R. O. C.
cmwu@mail.ndhu.edu.tw

Abstract—Information and communication technology (ICT) profoundly impacts on environment because of its large amount of CO₂ emissions. In the past years, the research field of “green” and low power consumption networking infrastructures is of great importance for both service/network providers and equipment manufacturers. An emerging technology called cloud computing can increase the utilization and efficiency of hardware equipments. It can potentially reduce the global CO₂ emission. In this paper, we propose a virtual network architecture for cloud computing. Our virtual network can provide communication functions for virtual resources in cloud computing. We design an energy aware routing algorithm for our virtual router. We also design an efficient method for setting up the virtual network. The objective is to build a “green” virtual network in cloud computing.

Keywords—cloud computing; virtual network; virtual router; energy aware routing

I. INTRODUCTION

Information and communication technology (ICT) has a profound impact on economy and environment. According to some research reports, in the United State, energy consumed by the ICT equipments is roughly 8% of the total energy consumption, and it will increase 50% within a decade [8, 9, 17]. In addition to energy consumption, ICT equipments have a large amount of CO₂ emissions. According to Gartner, which is the world's leading information technology research and advisory company, estimation, the ICT equipment is responsible for 2% of global CO₂ emissions in 2007 [11].

In the past years, the research field of “green” and low power consumption networking infrastructures is very important for both service/network providers and equipment manufacturers. Despite ecological issues, the interest of the low power research has critical economical needs. Since the network's electrical requirements show a continuous growing trend over the past years, the energy costs of service/network providers have significantly increased [4, 7, 8, 9, 17].

Cloud computing is a paradigm that changes the behavior in the consumption and delivery of information technology (IT) services. Users use web service interfaces to demand resources and pay only for the resources that they actually consume [1, 2]. In addition to providing on-demand resources, clouds can deploy a custom-made environment for a given application. With the dynamical provision of resources, cloud computing can increase the utilization of resources and hence reduces the

number of IT hardware equipments. It can potentially reduce the global CO₂ emission. Furthermore, without purchasing, operating, maintaining, and periodically upgrading local computing infrastructures, cloud computing can lower the cost of IT services for an enterprise.

Virtualization is one of the main techniques of cloud computing. Virtualization makes abstractions of the resources, both hardware and software, of a computer system to simulate multiple execution environments. Virtualization can also reduce the number of IT hardware equipments [3, 5, 6].

Cloud computing shares data and computations through scalable networks. The role of the Internet is important in cloud computing where computing resources are distributed over a wide area network. The operator of the distributed cloud computing has to purchase bandwidth from different network providers for communicating between the servers deployed in various geographic locations. To provision the customized environment for a given application or service, in addition to the computing resources, the operator has to satisfy the network QoS. However, it is difficult to satisfy both computing and communicating requirements under different network performances and reliabilities [12, 16].

Network virtualization can offer an abstraction of the network for simplifying resource provisions in cloud computing. The operator of cloud computing deploys the services over the virtual networks without regard for the underlying infrastructure networks. Each service can own a customized virtual network. Thus, the network QoS of the service can be satisfied. Multiple virtual networks can run simultaneously over a single physical infrastructure without interfering with each other [3, 5, 6]. Path virtualization accelerates the packet forwarding process in the network since it can combine multiple channels into a routing path. It can also potentially reduce the energy used in routing.

In this paper, we propose a virtual network architecture for cloud computing. Our virtual network can provide communications for virtual resources in cloud computing. For those who need network resources, we can also provide the resources in an on-demand basis. We add some power saving mechanisms into the virtual network components such as routers and switches. We also design a power aware routing algorithm for cloud computing. The algorithm decreases the power consumption of the data transmission. We also design an energy aware network provision algorithm. Our algorithm is efficient and our design can make the cloud more “green”.

The rest of the paper is organized as follows. Section 2 surveys the related works. Section 3 introduces the virtual network. Section 4 explains the energy aware design. Finally, conclusions are given in Section 5.

II. RELATED WORKS

In [13], Zhu et al. present a virtual network named Cabernet (Connectively Architecture for Better Network Service). They use three-layer, service layer, connectivity layer, and infrastructure layer, network architecture to lower the barrier for deploying new services. The connectivity provider uses virtual network to satisfy the network requirements of the service provider. In the virtual network environment, the service provider can run “intradomain” protocols without regard for the many underlying infrastructure networks. Each service provider employing a dedicated wide-area virtual network can easily deploy new services. Cabernet provides not only computing resources, but also the whole virtual network to service providers.

He et al. present dynamically adaptive virtual networks for a customized Internet (DaVinci) [14]. Each of their proposed virtual network can run customized protocols to optimize network performance. DaVinci can use multiple paths between two end hosts to forward packets in order to satisfy the QoS requirements. For facilitating to run their virtual network, an edge virtual node is established to encapsulate packets. Users may use various methods, such as establishing a tunnel to a virtual node, configuring a Web browser to use a virtual node as a proxy, or DNS redirection, to connect to a virtual network.

Bavier et al. present a virtual network infrastructure called VINI [5] to facilitate network researchers to evaluate their network protocols and services. VINI supports simultaneous experiments with arbitrary network topologies on a shared physical infrastructure to provide researchers flexibility in designing their experiments. Their paper illustrates and tackles many important design issues in network virtualization.

In [15], Restrepo et al. present an energy profile aware routing scheme to decrease the energy consumption of the data transmission in the Internet. They take the energy consumption of equipments and traffic loads into account when the routing decision is made. They define several equations to minimize the overall energy consumption of an already dimensioned network. The energy profile of a network node is scalable. Before making a routing decision, the router has to know all energy profiles of nodes in every possible routing path. However, in reality, the energy profile will be changed constantly.

III. VIRTUAL NETWORK

The process of network virtualization is shown in Figure 1. Firstly, the network components such as the link, the switch, and router have to be virtualized. Tunneling is a technique to implement virtual links. And a soft router is used to implement the virtual router. Virtual routers can be run on commodity hardware such as server-class PCs. When users make a demand, a customized virtual network will be provided. The virtual network provision is a critical step, since it needs to satisfy the

requirements of users. Furthermore, it affects the utilization of the physical network.

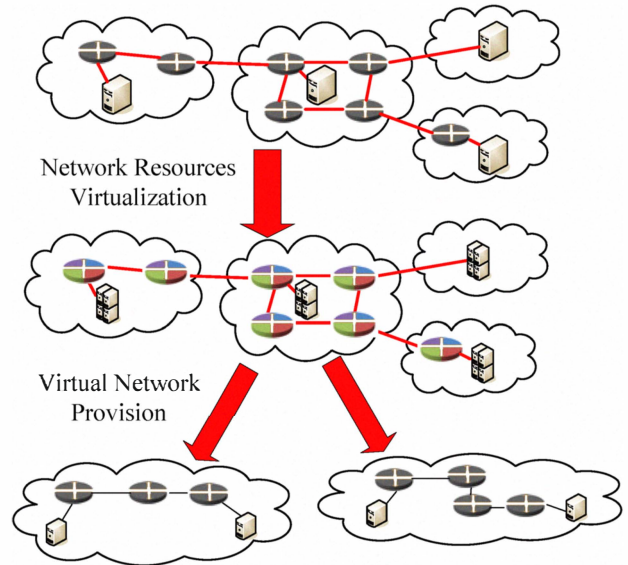


Figure 1. The process of network virtualization.

A. Virtual Network Architecture

The architecture of our virtual network is shown in Figure 2. We use a hierarchical structure to build the virtual network for facilitating management. In Figure 2, the virtual resource uses virtual links to connect to the virtual bridge. The virtual bridge can increase flexibility and transferring speed in the virtual network. The virtual bridge provides the interface and forwards packets for the virtual resources that connect to it. The virtual bridge decreases the loading of the virtual router. The virtual router forwards packets to the destination. The routing algorithm run on the virtual router can be customized. Furthermore, the virtual router can also run the routing protocol defined by the user. The resource monitor sends the physical network resource information to the virtual network controller for deciding resource provision.

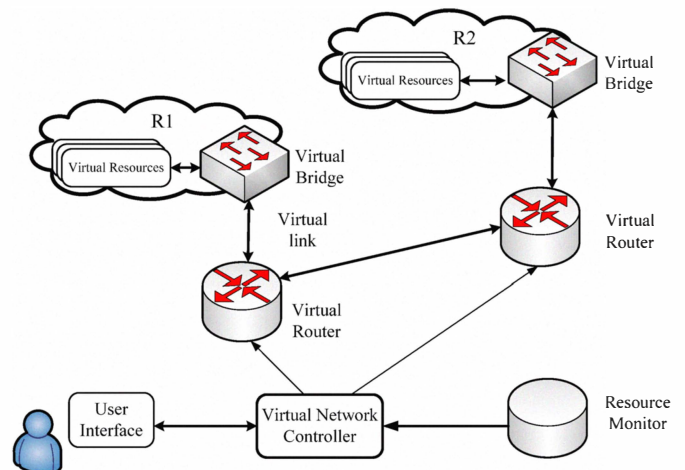


Figure 2. The architecture of our virtual network.

The virtual network controller manages the virtual network. Our virtual network has two objectives. The first is to provide communications for the virtual resources in the cloud. While the operator of cloud computing deploys the computing resources for a service or user, we have to provide the network communication for the deployment. In this case, the virtual network controller has to send the routing information to the related virtual routers. Another objective is to provide the virtual network to the user that needs a customized network to run his routing algorithm and protocol. The user uses the user interface to demand his requirements such as the number nodes, the network topology, and bandwidths. The virtual network controller efficiently provides and satisfies the requirements of the user.

B. Virtual Network Implementation

In this subsection, we describe the implementation techniques of the virtual network components. We explain the implementation decision we made when building our virtual network for cloud computing.

C. Virtual Links

Virtual links must be simple, fast, flexible, and isolated. Several techniques, such as time-division multiplexing (TDM), Multi-Protocol Label Switching (MPLS), and Tunneling, can be used to implement the virtual link. TDM is limited by the physical substrate. MPLS needs the label switch router to forward data. Tunneling uses encapsulation to provide virtual links. In tunneling, each packet is encapsulated within the packet format of the transit network. Thus, tunneling provides a simple method to construct the virtual link.

Generic Routing Encapsulation (GRE) is a tunneling protocol with fixed encapsulation overhead. It is suitable for virtual link implementation. However, in cloud computing, multiple virtual hosts may run on a computer. To allow multiplex virtual hosts on the same network using overlapping address space, the virtual link has to support non-IP packet transferring. Therefore, we use Ethernet tunnel based on GRE to implement virtual links between the virtual host and the virtual bridge. Those links between bridges and routers are implemented by IP tunnels.

A virtual link must be isolated from links that belong to other virtual networks. Some applications running on virtual hosts may overuse the network bandwidth. Two methods are proposed to solve this problem. In [3], the authors terminate the tunnel in the root context. This method allows the infrastructure administrator to restrict users' bandwidths. Another method is to run customized traffic-management protocol in each virtual network. This solution is more flexible. However, this method cannot solve the problem that some virtual networks maliciously try to acquire more bandwidths and harm the performance of other virtual networks. Our network substrate providers are all universities. Most our users are researchers in universities. We use the latter method to provide the isolation of the virtual link.

D. Virtual Bridge

The role of the virtual bridge in our virtual network resembles a default gateway. It is responsible for managing the virtual hosts connecting to it. Its major function is to receive and forwards packet. Entering packets are decapsulated to check their destinations. If the destination node of a packet is connected to the bridge, it is forwarded directly. Otherwise, it is transferred to the virtual router using IP encapsulation. The architecture of the virtual bridge consists of the decapsulation, encapsulation, forwarding engine, and forwarding table. We will modify codes from a Linux Open Source to implement the virtual bridge.

E. Virtual Router

In the last few years, many researches focus on Software Router (SR) designs [4, 5, 6]. Linux Software Router provides a convenient way to build the virtual router in cloud computing. Common criticisms of SR mainly focus on the data plane. However, SR provides high flexibility and modularity in cloud computing. Nowadays the performance of a general purpose computer is significantly improved. The performance gap between SR and a dedicated router might not be so large. We will implement the virtual router based on Linux SR. The generic architecture of the virtual router is shown in Figure 3. Each virtual router is composed of the routing table, transmission daemon, and the virtual interface. The transmission daemon runs the customized routing algorithm. Virtual interfaces share the physical interface. In the computer with the multi-core CPU, the instruction parallelism profoundly affects the performance of the virtual router. We modify the related components of Linux SR to improve the performance of our virtual router.

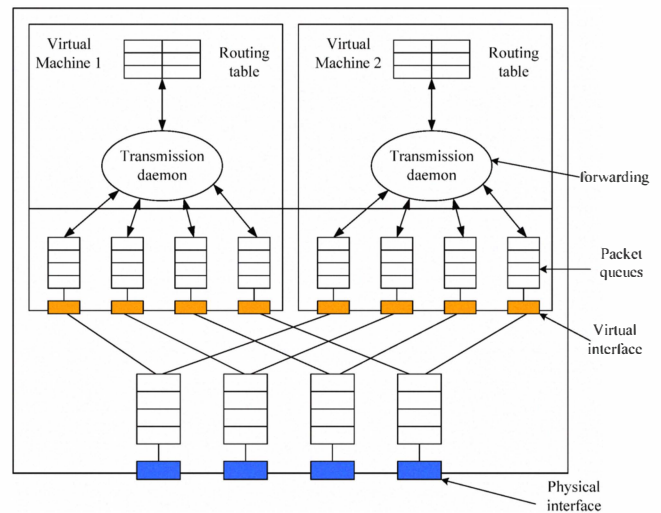


Figure 3. The architecture of the virtual router.

F. Virtual Network Provision

The user interface shown in Figure 2 is a graphic user interface (GUI). Users can demand their virtual network through this interface. The functions include *file*, *configuration*,

and *actions*. The function *file* allows users to manage their files. Subfunctions under *file* include *new*, *open*, *save*, and *close*. Users can use *configuration* to setup their network specification and task workflow specification. Basically, *bandwidth*, *router*, *bridge*, and *task* are its subfunctions. When users finish their proper setup, they can use *actions* function to run their experiment execution. All parameters through the user interface are sent to the virtual network controller.

Functions of the virtual network controller are network specification, task workflow specification, and experiments execution. The network specification creates the virtual network according to the specification of the network demanded by the user. The user can use drag-and-drop method to create the virtual network. An example of the virtual network is shown in Figure 4. The user needs three routers, two bridges, and five virtual machines.

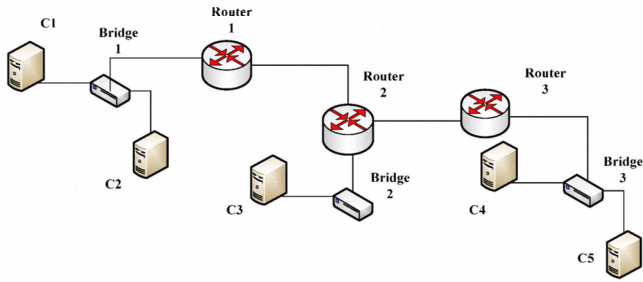


Figure 4. An example of the virtual network demanded by the user.

After drawing the virtual network, the user has to setup the specifications of these network components. The virtual network controller uses the provision algorithm described in Subsection 4.C. to create the virtual network. While the network has been constructed, the user needs to do task workflow specification. The task workflow can be presented by an acyclic graph shown in Figure 5. Each circle in Figure 5 represents a task while the arrow line stands for the execution dependence. The user needs to indicate which tasks should be executed by the virtual hosts and specifies the order of execution.

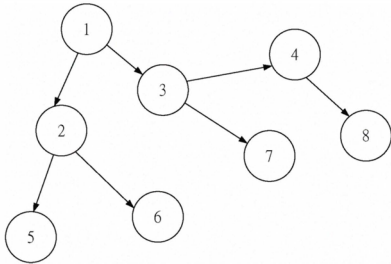


Figure 5. An example of the task workflow.

While doing experiment execution, two methods can be used to dispatch commands to virtual machines. The first is using shared folders. This method is supported by some virtual machines. The second is using the remote invocation. Each virtual machine needs to run the Java remote management interface in this method. How to communicate between the

controller and the virtual machine is a major issue in this method. We use the shared folder method for its simplicity.

IV. ENERGY AWARE DESIGN

A. Energy Model

The power consumption of an application in cloud computing can be defined as [10]:

$$P = P_{computations} + P_{communications} \quad (1)$$

where $P_{computations}$ is the power consumed by computing and $P_{communications}$ is the power consumed by communicating.

If this application is assigned to N virtual nodes running on M physical nodes, Equation 1 becomes

$$P_{computations} = \sum_{i=1}^N P_{computations}^i(w^i) + \sum_{j=1}^M P_{node}^j \quad (2)$$

where $P_{computations}^i(w^i)$ is the power consumed by i virtual node with w^i workload and P_{node}^j is the static power consumed by j physical node. The static power consumption of the physical node includes the power consumptions of the power supply, fans, lights, and peripherals.

$P_{communications}$ is the total power consumed by transferring data in the application and it is defined as the following equation.

$$P_{communications} = \sum_{i=0}^C P_{path}^i(F_i) \quad (3)$$

where C is the number of communications in the application. $P_{path}^i(F_i)$ is the power consumption of communication i , and F_i is the data flow of the communication.

A communication path is through several links and routers. If a path passes K links, $P_{path}^i(F_i)$ can be defined as:

$$P_{path}^i(F_i) = \sum_{j=1}^K P_{link}^j(F_i) + \sum_{j=1}^{K-1} P_{router}^j(F_i) \quad (4)$$

where $P_{link}^j(F_i)$ is the total power consumption through the link j and $P_{router}^j(F_i)$ is the power consumption of router j .

We define the bit power P_{bit} metric as the power consumed when one bit of data is transported through a communication path.

$$P_{bit} = \sum_{j=1}^K P_{link}^j + \sum_{j=1}^{K-1} P_{router}^j \quad (5)$$

The power consumption is the requirement of a system at a discrete point in time. The energy consumption corresponds to the system operational cost. The energy consumption can be derived from the equation of the power consumption and is computed by

$$E = \int_1^2 P \cdot dt \quad (6)$$

Applying this to Equation 2, 3, 4, 5, we can get the energy equations illustrated in the following.

$$E_{computations} = \int_1^2 P_{computations} \cdot dt \quad (7)$$

$$E_{communications} = \int_1^2 P_{communications} \cdot dt \quad (8)$$

$$E_{path}(F_i) = \int_1^2 P_{path}(F)_i \cdot dt \quad (9)$$

$$E_{bit} = \int_1^2 P_{bit} \cdot dt \quad (10)$$

To decrease the energy consumed by an application, we have to minimize both Equations 7 and 8 when deploying demanded resources to the application. Three types of services are provided by cloud computing. They are Software as a Service (SaaS), Platform as a Service (PaaS), and Infrastructure as a Service (IaaS). For satisfying the requirement of users, the resource provision system of cloud computing needs to frequently send the image to the virtual host through the network. Thus, in cloud computing, the energy consumption on communication is larger than that of other distributed computing.

Many methods focus on reducing energy consumption of the computing. Using dynamic voltage and frequency scaling on a computer can decrease the energy consumed by the CPU. The main idea is to supply lower working voltage or frequency to the CPU while the workload is not heavy. In [4], the authors use several working modes of the CPU to lower the computing energy. Another common method is the sleeping mode. A computer is turned into the sleeping mode while idle.

Several researches on reducing communication energy have been proposed. In [15], they propose energy aware routing algorithms to decrease communication energy while data are routed to their destinations. This method needs the routing protocol to support energy profiles delivery. Each router in this method uses the energy information carried by control plane protocol to calculate energy efficient traffic paths. The traffic load also affects the communication energy. Thus, in [15], the authors combine energy profiles and traffic loads for routing decision in order to reduce communication energy.

When a virtual host is deployed to an application in cloud computing, our virtual networks provide the communication mechanism for data transmissions. In this case, since the computing resources are deployed, we minimize the communication energy shown in Equation 3 to decrease energy consumption by the application. Another service of our virtual network is the network provision. We provide a whole network including communication nodes, routers, and the network topology according to the user demand. In such situation, both $E_{computations}$ and $E_{communications}$ will be considered to decrease energy consumed by the provided network.

B. Minimizing Communication Energy

The communication energy is consumed by the network components including physical links, switches, routers, and servers, while transferring data. In cloud computing, a virtual link consists of several physical links in the substrate network.

Figure 4 shows the relationship between the virtual network and the substrate network. In Figure 6(a), the substrate network has 12 nodes and 18 links. A virtual network provided by the substrate network shown in Figure 6(a) is shown in Figure 6(b). The virtual network has five nodes, B, D, F, I, and K. The virtual links are denoted by the red line in Figure 6(b). The virtual link between nodes I and B can be constructed by several routes in the substrate network. These routes are (j, i) or (r, o, a). Usually, the shortest route is selected to forward the packet. However, the shortest route may not be the energy efficiency route. We use the energy profile of the network to decide the routing.

The energy profiles of two components, the link and the router, have to be obtained. The energy profile of the link includes all network equipments (e.g. layer 2 switches, racks, shelves, fiber convertors) along this link. The energy profiles are delivered by the routing protocol. We use the routing protocol such as OSPF to create the routing table in the router. The traffic load information is sent by the virtual network controller shown in Figure 2. We also take the traffic load information into account to decrease the energy consumption.

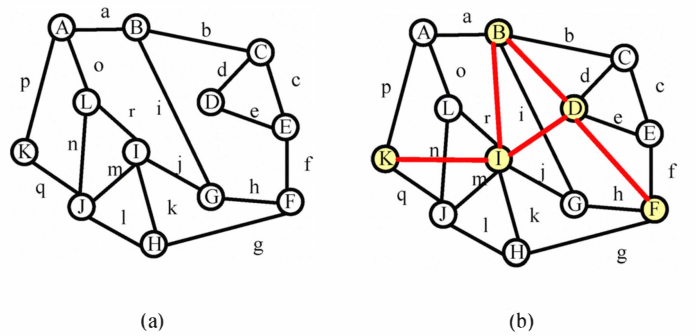


Figure 6. The substrate network and the virtual network.

Two steps energy consumption is used to calculate the energy efficiency routes. The first step is a single virtual link (e.g. the virtual link between nodes K and I shown in Figure 6(a)). In this step, all nodes in Figure 6(a) have to calculate the energy efficiency routes to other nodes. In the second step, we calculate the virtual route using the routing information from the virtual nodes. Thus, each virtual network has to run its routing protocol.

C. Minimizing Computation Energy

When providing a virtual network, both $E_{computations}$ and $E_{communications}$ can be minimized to decrease the energy consumption. While the user demands a virtual network, we search the available network resources to create the virtual network that meets the user's requirements, such as the topology, the number of nodes, and link bandwidth. In the searching process, we calculate the probable power consumption using the energy profiles of the resources and Equations 2 and 3. In Equation 3, we minimize the number of commodity hardware to decrease the static power consumption of the computation. However, minimizing the static power consumption of the computation may increase the energy consumed by the communication.

Figure 7 shows an example of the provision process. In Figure 7, nodes A, B, and G, are assigned to the virtual nodes 1, 2, and 5, respectively. To provide virtual node 4, we have to search available resources to calculate the probable energy consumption. If the number of available resources is R , the worst case complexity of the provision algorithm is $O(R!)$. We use the heuristic approach to find an approximate solution.

The provision problem can be reduced to the subgraph mapping problem. To design our algorithm, we define two graphs.

Definition 1. A substrate network (SN) is a weighted digraph $SN(R, E)$, where R is a set of resource nodes, $R = \{ (r_i, r_j) \mid r_i \in R, r_j \in R \}$ is a set of links. For each $e_{ij} = (r_i, r_j)$ in E , there is a weight w_{ij} which represents the energy profile of link e_{ij} .

Definition 2. A virtual network (VN) is a weighted digraph $VN(V, L)$, where V is a set of virtual nodes, $L = \{ (v_i, v_j) \mid v_i \in V, v_j \in V \}$ is a set of virtual links. For each $l_{ij} = (v_i, v_j)$ in L , there is a weight p_{ij} which represents the energy profile of virtual link l_{ij} .

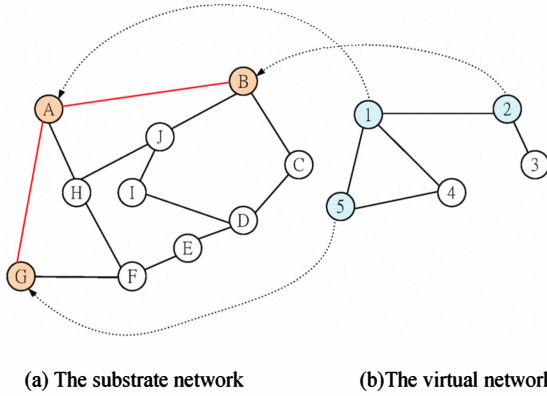


Figure 7. An example of the provision process.

A virtual resource is regarded as a resource node in Definition 1. We sort all available resources according to their communications energies and push the result to the unassigned stack in increasing order. The smallest is placed at the top of the stack. We pop the stack to assign a virtual resource in the SN to a virtual node in VN. After assigning, both related communication and computation energy are calculated. Such assignment has to satisfy the bandwidth requirement. A virtual resource is assigned to all unassigned virtual node. We select the result that has the minimal energy consumption. After a virtual resource has been assigned to a virtual node, we pop another virtual resource to continue assigning. The assigning process is repeated until all virtual nodes have been assigned. Our provision algorithm is shown as the following.

Provision ($SN(R, E)$, $VN(V, L)$, Bandwidth requirement):

- Step 1. Set initial value ∞ to $v_i, \forall v_i \in V$. Set $Min = \infty$.
- Step 2. Sort all available $r_i \in R$ according to Σe_{ij} . Store the result to the unassigned stack in increasing order. The smallest is placed at the top of the stack.
- Step 3. Pop a node r from the unassigned stack.

- Step 4. Select an unassigned $v \in V$.
- Step 5. Establish all related $l \in L$ in the VN for this v .
- Step 6. Calculate all $p_{ij}, \Sigma p_{ij}$, and bandwidths of related links.
- Step 7. If the bandwidth of related links cannot satisfy the requirement, goto Step 11.
- Step 8. If $\Sigma p_{ij} < Min$, $Min = \Sigma p_{ij}$ and assign this r to this v .
- Step 9. If all unassigned v have been selected, go to step 11.
- Step 10. Select another unassigned $v \in V$, go to step 5.
- Step 11. If there is unassigned $v \in V$, go to step 4

V. CONCLUSIONS

In this paper, we focus on building the virtual network in cloud computing. To facilitate network management, our virtual network is constructed using an hierarchical structure. We also virtualize the network components such as links, bridges, and router. Our virtual network can provide the communication between virtual hosts with flexibility. Furthermore, the virtual network provided by us can run the customized routing protocol.

To decrease energy consumption, we present a routing algorithm with energy awareness. We estimate the energy consumed by the network component to decide the packet forwarding route. In this algorithm, we can minimize the energy consumption of the communication. We also design an energy aware provision algorithm to decrease the energy consumed by the virtual network.

As cloud computing becomes more wide-spread, energy efficiency will become more important. The green virtual network architecture proposed in this paper is just a beginning. More research needs to be done to make it more efficient and to measure the actual performances.

Acknowledgements: This research is supported partially by NSC under contract numbers 96-2221-E-259-018-MY3.

REFERENCE

- [1]. Amazon Elastic Compute Cloud (Amazon EC2), <http://aws.amazon.com/ec2>, 2008.
- [2]. Amazon Simple Storage Service (Amazon S3), <http://aws.amazon.com/s3>, 2008.
- [3]. S. Bhatia, M. Motiwala, W. Muhlbauer, Y. Mundada, V. Valancius, A. Bavier, N. Feamster, L. Peterson, and J. Rexford, "Trellis: A platform for building flexible, fast virtual networks on commodity hardware," *Proceedings of Workshop on Real Overlays and Distributed Systems (ROADS)*, December 2008
- [4]. R. Bolla, R. Bruschi, and A. Ranieri, "Performance and Power Consumption Modeling for Green COST Software Router," *Proceedings of COMSNET*, 2009, pp. 1-8
- [5]. A. Bavier, N. Feamster, M. Huang, L. Peterson, and J. Rexford, "In VINI Veritas: Realistic and Controlled Network Experimentation," *Proceedings of SIGCOMM'06*, 2006, pp. 3-14.
- [6]. B. C. B. Chan, J. C. F. Lau, and J. C. S. Lui, "OPERA: An open-source extensible router architecture for adding new network services and protocols," *Transactions on Journal of Systems and Software*, vol. 78, no. 1, 2005, pp.24-36.
- [7]. J. Chabarek, J. Sommers, et al., "Power awareness in network design and routing," *proceedings of INFOCOM*, Phoenix, Arizona, U.S.A, Apr. 15-17, 2008.

- [8]. L. Ceuppens, A. Sardella, and D. Kharitonov, "Power Saving Strategies and Technologies in Network Equipment Opportunities and Challenges, Risk and Rewards," *Proceedings of Applications and the Internet 2008*, 2008, pp. 381-384
- [9]. M. Etoh, T. Ohya, and Y. Nakayama, "Energy Consumption Issues on Mobile Network Systems," *Proceedings of Applications and the Internet 2008*, 2008, pp. 365-368.
- [10]. X. Feng, R. Ge, and K. W. Cameron, "Power and Energy Profiling of Scientific Applications on Distributed Systems," *Proceeding of the 19th IEEE International Parallel and Distributed Processing Symposium*, 2005, pp.34-43.
- [11]. Gartner, "Gartner Estimates ICT Industry Accounts for 2 Percent of Global CO2 Emissions," <http://www.gartner.com/it/page.jsp?id=503867>.
- [12]. T. Miyamoto, M. Hayashi, and H. Tanaka, "Customizing Network Functions for High Performance Cloud Computing," *Proceedings of Eighth IEEE International Symposium on Network Computing and Applications*, 2009, pp. 130-133.
- [13]. Y. Zhu, R Z. S., S. Rangarajan, and J. Rexford, "Cabernet: Connectivity Architecture for Better Network Services," *Proceedings of ReArch '08*, Dec. 9, 2008, Madrid, SPAIN
- [14]. J. He, Z. S. Rui, Ying Li, C. Y. Lee, J. Rexford, and M. Chiang, "DaVinci: Dynamically Adaptive Virtual Networks for a Customized Internet," *Proceedings of CoNext*, December 2008
- [15]. J. C. C., Restrepo, C. G. Gruber, and C. M. Machuca, "Energy Profile Aware Routing," *Proceedings of IEEE International Conference on Communications Workshops 2009*, pp. 1-5.
- [16]. M. A. Vouk, "Cloud Computing-Issues, Research and Implementation," *Proceedings of the ITI 2008 30th Int. Conf. on Information Technology Interfaces*, June 2008, Cavtat, Croatia, pp. 31-40.
- [17]. T. T. Ye, "Analysis of power consumption on switch fabrics in network routers," *proceedings of the 39th Design Automation Conference*, New Orleans, U.S.A, June 10-14, 2002.