DENS: Data Center Energy-Efficient Network-Aware Scheduling

Dzmitry Kliazovich and Pascal Bouvry
University of Luxembourg
6 rue Coudenhove Kalergi, Luxembourg
dzmitry.kliazovich@uni.lu, pascal.bouvry@uni.lu

Samee Ullah Khan
North Dakota State University
Fargo, ND 58108-6050
samee.khan@ndsu.edu

Abstract— **In modern data centers, energy consumption accounts for a considerably large slice of operational expenses. The state of the art in data center energy optimization is focusing only on job distribution between computing servers based on workload or thermal profiles. This paper underlines the role of communication fabric in data center energy consumption and presents a scheduling approach that combines energy efficiency and network awareness, termed DENS. The DENS methodology balances the energy consumption of a data center, individual job performance, and traffic demands. The proposed approach optimizes the tradeoff between job consolidation (to minimize the amount of computing servers) and distribution of traffic patterns (to avoid hotspots in the data center network).**

Keywords-network-aware scheduling, energy-efficient, data center, cloud computing, congestion

I. INTRODUCTION

Data centers are becoming increasingly popular for the provisioning of computing resources. The cost and operational expenses of data centers have skyrocketed with the increase in computing capacity [24].

Energy consumption is a growing concern for data centers operators. It is becoming one of the main entries on a data center operational expenses (OPEX) bill [1, 2]. The Gartner Group estimates energy consumptions to account for up to 10% of the current OPEX, and this estimate is projected to rise to 50% in the next few years [3].

The slice of roughly 40% is related to the energy consumed by information technology (IT) equipment [24], which includes energy consumed by the computing servers as well as data center network hardware used for interconnection. In fact, about one-third of the total IT energy is consumed by communication links, switching, and aggregation elements, while the remaining two-thirds are allocated to computing servers [6]. Other systems contributing to the data center energy consumption are cooling and power distribution systems that account for 45% and 15% of total energy consumption, respectively.

The first data center energy saving solutions [25] operated on a distributed basis and focused on making the data center hardware energy efficient. There are two popular techniques for power savings in computing systems. The Dynamic Voltage and Frequency Scaling (DVFS) technology, adjusts hardware power consumption according to the applied computing load and the Dynamic Power Management (DPM), achieves most of energy savings by powering down devices at runtime. To make DPM scheme efficient, a scheduler must consolidate data center jobs on a minimum set of computing resources to maximize the amount of unloaded servers that can be powered down (or put to sleep) [5]. Because the average data center workload often stays around 30%, the portion of unloaded servers can be as high as 70% [4].

Most of the existing approaches for job scheduling in data centers focus exclusively on the job distribution between computing servers [7] targeting energy-efficient [8] or thermal-aware scheduling [9]. To the best of our knowledge, only a few approaches have considered data center network and traffic characteristics for developing energy-efficient data center schedulers [10-12].

Ref. [10] identifies the problem associated with existing multi-path routing protocols in typical fat tree network topologies. Two large traffic flows may be assigned to share the same path if their hash values collide leaving other paths under-loaded. The problem is solved with the introduction of a complex central scheduler that performs flow differentiation and analysis of flow traffic demands across the data center network. Traffic-aware virtual machine placement is proposed in [12]. Relying on the knowledge about network topology, virtual machines are placed to optimize traffic flows inside a data center network. The approach presented in [11], also allows job migration control during runtime with a specifically designed network-aware scheduler. The migration scheduler is aware of the migration delays and bandwidth resources required. As we may see, most of the existing solutions leave the networking aspect unaccounted for in an energy-efficient optimization setting.

This paper presents a data center scheduling methodology that combines energy efficiency and network awareness. The methodology is termed DENS, which is an acronym for **d**ata center **e**nergy-efficient **n**etwork-aware **s**cheduling. The DENS methodology aims to achieve the balance between individual job performances, job QoS requirements, traffic demands, and energy consumed by the data center. Data intensive jobs require low computational load, but produce heavy data streams directed out of the data center as well as to the neighboring nodes. Such data intensive jobs are typically

produced by popular video sharing or geographical information services. The scheduling approach presented in this paper is designed to avoid hotspots within a data center while minimizing the number of computing servers required for job execution. In the proposed methodology, the network awareness is achieved with the introduction of feedback channels from the main network switches. Moreover, the proposed methodology reduces computational and memory overhead compared to previous approaches, such as flow differentiation, which makes the proposed methodology easy to implement and port to existing data center schedulers.

The rest of the paper is organized as follows. Section II summarizes the background knowledge on a typical data center architecture, energy consumption models, and data center network congestion. Section III presents the core of the scheduling approach and defines the necessary components of the proposed methodology. In Section IV, we will present and discuss experimental results. Finally, Section V will conclude the paper and outline directions for future work on the topic.

II. BACKGROUND

A. Data Center Topology

Three-tier trees of hosts and switches form the most widely used data center architecture [26]. It (see Fig. 1) consists of the *core* tier at the root of the tree, the *aggregation* tier that is responsible for routing, and the *access* tier that holds the pool of computing servers (or hosts). Early data centers used two-tier architectures with no aggregation tier. However, such data centers, depending on the type of switches used and per-host bandwidth requirements, could typically support not more than 5,000 hosts. Given the pool of servers in today's data centers that are of the order of 100,000 hosts [13] and the requirement to keep layer-2 switches in the access network, a three-tiered design becomes the most appropriate option.

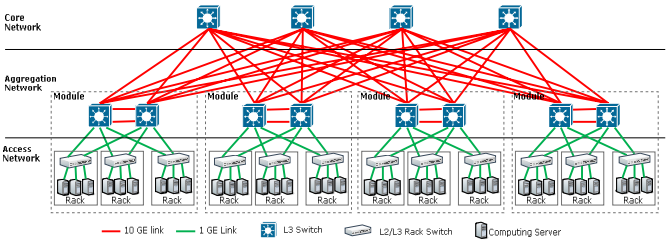

Figure 1. Three-tier data center architecture.

Although 10 Gigabit Ethernet (GE) transceivers are commercially available, in a three-tiered architecture the computing servers (grouped in racks) are interconnected using 1 GE links. This is due to the fact that 10 GE transceivers: (a) are too expensive and (b) probably offer more capacity than needed for connecting computing servers. In current data centers, rack connectivity is achieved with inexpensive Top-of-Rack (ToR) switches. A typical ToR switch shares two 10 GE uplinks with 48 GE links that interconnect computing servers within a rack. The difference between the downlink and the uplink capacities of a switch defines its oversubscription ratio, which in the aforementioned case is equal to 48/20 = 2.4:1. Therefore, under full load, only 416 Mb/s will remain available to each of the individual servers out of their 1 GE links.

At the higher layers of hierarchy, the racks are arranged in modules (see Fig. 1) with a pair of aggregation switches servicing the module connectivity. Typical oversubscription ratios for these aggregation switches are around 1.5:1, which further reduces the available bandwidth for the individual computing servers to 277 Mb/s.

The bandwidth between the core and aggregation networks is distributed using a multi-path routing technology, such as the Equal Cost Multi-Path (ECMP) routing [14]. The ECMP technique performs a per-flow load balancing, which differentiates the flows by computing a hash function on the incoming packet headers. For a three-tiered architecture, the maximum number of allowable ECMP paths bounds the total number of core switches to eight. Such a bound also limits the deliverable bandwidth to the aggregation switches. This limitation will be waved with the (commercial) availability of 100 GE links, standardized in June 2010 [15].

Designing data center topologies is an extremely important research topic. Fat-tree successors are constantly proposed for large-scale data centers [16, 17]. However, the fact that not even a single data center has been built (to this date) based on such proposals, we constrict the scope of this paper to the three-tiered architecture. Nevertheless, we must note that all of the findings of this research will remain valid for any or all types of data center topologies.

B. Energy Models

Computing servers account for a major portion of data center energy consumption. The power consumption of a computing server is proportional to the CPU utilization. An idle server consumes around two-thirds of its peak-load consumption to keep memory, disks, and I/O resources running [18]. The remaining one-third changes almost linearly with the increase in the level of CPU load.

There are two main approaches for reducing energy consumption in computing servers: (a) DVFS [27] and (b) DPM [28]. The DVFS scheme adjusts the CPU power (consequently the performance level) according to the offered load. The aforementioned is based on the fact that power in a chip decreases proportionally to $V^2 \cdot f$, where V is a voltage, and f is the operating frequency. The scope of the DVFS optimization is limited to CPUs. Therefore, computing server components, such as buses, memory, and disks remain functioning at the original operating frequency. On the contrary, the DPM scheme can power down computing servers (that includes all components), which makes such a technique very energy efficient. However, if there occurs a need to power up (powered down) the server, a considerable amount of energy must be consumed compared to the DVFS scheme.

Switches form the basis of the interconnection fabric that delivers job requests to the computing servers for execution. Energy consumption of a switch depends on the: (a) type of switch, (b) number of ports, (c) port transmission rates, and (d) employed cabling solutions. The energy consumed by a switch can be generalized by the following [29]:

$$P_{switch} = P_{chassis} + n_{linecards} \cdot P_{linecard} + \sum_{i=0}^{R} n_{ports} \cdot P_r, \quad (1)$$

where $P_{chassis}$ is the power consumed by the switch base hardware, $P_{linecard}$ is the power consumed by an active linecard, and P_r corresponds to the power consumed by an active port (transmitter) running at the rate r. In Eq. (1), only the last component, P_r, scales with a switch's transmission rate. This fact limits the benefits of any rate adaptive scheme as the combined consumption of switch transceivers accounts for just 3-15% of switch's total energy consumption [29]. Both $P_{chassis}$ and $P_{linecard}$ do not scale with the transmission rate and can only be avoided when the switch hardware is powered down (given that there is no traffic to be handled by the switch).

Obviously, not all of the switches can dynamically be put to sleep. Each core switch consumes a considerable amount of energy to service large switching capacity. Because of their location within the communication fabric and proper ECMP forwarding functionality, it is advisable to keep the core network switches running continuously at their maximum transmission rates. On the contrary, the aggregation switches service modules, which can be powered down when the module racks are inactive. The fact that on average most of the data centers are utilized around 30% of their compute capacity [4], it makes perfect sense to power down unused aggregation servers. However, such an operation must be performed carefully by considering possible fluctuations in job arrival rates. Typically, it is enough to keep a few computing servers running idle on top of the necessary computing servers as a buffer to account for possible data center load fluctuation [18].

C. Data Center Network Congestion

Utilizing a communication fabric in data centers entails the concept of running multiple types of traffic (LAN, SAN, or IPC) on a single Ethernet-based medium [30]. On one side, the Ethernet technology is cheap, easy to deploy, and relatively simple to manage, on the other side, the Ethernet hardware is less powerful and provisions for small buffering capacity. A typical buffer size in an Ethernet network is in the order of 100s of KB. However, a typical buffer size of an Internet router is in the order of 100s of MB [19]. Small buffers and the mix of high-bandwidth traffic are the main reasons for network congestion.

Any of the data center switches may become congested either in the uplink direction or the downlink direction or both. In the downlink direction, the congestion occurs when individual ingress link capacities overcome individual egress link capacities. In the uplink direction, the mismatch in bandwidth is primarily due to the bandwidth oversubscription ratio, which occurs when the combined capacity of server ports overcomes a switch's aggregate uplink capacity.

Congestion (or hotspots) may severely affect the ability of a data center network to transport data. Currently, the Data Center Bridging Task Group (IEEE 802.1) [31] is specifying layer-2 solutions for congestion control, termed IEEE 802.1Qau specifications. The IEEE 802.1Qau specifications introduce a feedback loop between data center switches for signaling congestion. Such a feedback allows overloaded switches to backpressure heavy senders with the congestion notification signal. Such a technique may avoid congestion-related losses and keep the data center network utilization high. However, it does not address the root of the problem as it is much more efficient to assign data-intensive jobs to different computing servers in the way that jobs avoid sharing common communication paths. To benefit from such spatial separation in the three-tiered architecture (see Fig. 1), the jobs must be distributed among the computing servers in proportion to the job communication requirements. Data-intensive jobs, like ones generated by video sharing applications, produce a constant bit-stream directed to the end-user as well as communicate with other jobs running in the data center. However, such a methodology contradicts the objectives of energy-efficient scheduling, which tries to concentrate all of the active workloads on a minimum set of servers and involve minimum number of communication resources. This tradeoff between energy-efficiency, data center network congestion, and performance of individual jobs is resolved using a unified scheduling metric presented in the subsequent section.

III. The DENS Methodology

The DENS methodology minimizes the total energy consumption of a data center by selecting the best-fit computing resources for job execution based on the load level and communication potential of data center components. The communicational potential is defined as the amount of end-to-end bandwidth provided to individual servers or group of servers by the data center architecture. Contrary to traditional scheduling solutions [7] that model data centers as a homogeneous pool of computing servers, the DENS methodology develops a hierarchical model consistent with the state of the art data center topologies. For a three-tier data center, the DENS metric M is defined as a weighted combination of server-level f_s, rack-level f_r, and module-level f_m functions:

$$M = \alpha \cdot f_s + \beta \cdot f_r + \gamma \cdot f_m, \quad (2)$$

where α, β, and γ are weighted coefficients that define the impact of the corresponding components (servers, racks, and/or modules) on the metric behavior. Higher α values favor the selection of overloaded servers in under-utilized racks. Higher β values will prioritize computationally loaded racks with low network traffic activity. Higher γ values favor selection of loaded modules. The γ parameter is an important design variable for job consolidation in data centers. Taking into account that $\alpha + \beta + \gamma$ must equal unity, the values of $\alpha = 0.7$, $\beta = 0.2$, and $\gamma = 0.1$ are selected experimentally (see Section IV for details) to provide a good balance in the evaluated three-tier data center topology.

The factor related to the choice of computing servers combines the server load $L_s(l)$ and its communication potential $Q_r(q)$ that corresponds to the fair share of the uplink resources on the ToR switch. This relationship is given as:

$$f_s(l,q) = L_s(l) \cdot \frac{Q_r(q)^{\varphi}}{\delta_r}, \quad (3)$$

where $L_s(l)$ is a factor depending on the load of the individual servers l, $Q_r(q)$ defines the load at the rack uplink by analyzing the congestion level in the switch's outgoing queue q, δ_r is a bandwidth over provisioning factor at the rack

switch, and φ is a coefficient defining the proportion between $L_s(l)$ and $Q_r(q)$ in the metric. Given that both $L_s(l)$ and $Q_r(q)$ must be within the range $[0,1]$ higher φ values will decrease the importance of the traffic-related component $Q_r(q)$. Similar to the case of computing servers, which was encapsulated in Eq. (3), the factors affecting racks and modules can be formulated as:

$$f_r(l,q) = L_r(l) \cdot \frac{Q_m(q)^{\varphi}}{\delta_m} = \frac{Q_m(q)^{\varphi}}{\delta_m} \cdot \frac{1}{n}\sum_{i=1}^{n} L_s(l), \quad (4)$$

$$f_m(l) = L_m(l) = \frac{1}{k}\sum_{j=0}^{k} L_r(l), \quad (5)$$

where $L_r(l)$ is a rack load obtained as a normalized sum of all individual server loads in the rack, $L_m(l)$ is a module load obtained as a normalized sum of all of the rack loads in this module, n and k are the number of servers in a rack and the number of racks in a module respectively, $Q_m(q)$ is proportional to the traffic load at the module ingress switches, and δ_m stands for the bandwidth overprovisioning factor at the module switches. It should be noted that the module-level factor f_m includes only a load-related component l. This is due to the fact that all the modules are connected to the same core switches and share the same bandwidth using ECMP multi-path balancing technology.

The fact that an idle server consumes energy that is almost two-thirds of its peak consumption [18], suggests that an energy-efficient scheduler must consolidate data center jobs on a minimum possible set of computing servers. On the other hand, keeping servers constantly running at peak loads may decrease hardware reliability and consequently affect the job execution deadlines [20]. To address the aforementioned issues, we define the DENS load factor as a sum of two sigmoid functions:

$$L_s(l) = \frac{1}{1+e^{-10\left(l-\frac{1}{2}\right)}} - \frac{1}{1+e^{-\frac{10}{\varepsilon}\left(l-\left(1-\frac{\varepsilon}{2}\right)\right)}}. \quad (6)$$

The first component in Eq. (6) defines the shape of the main sigmoid, while the second component servers as a penalizing function aimed at the convergence towards the maximum server load value (see Fig. 2). The parameter ε defines the size and the incline of this falling slope. The server load l is within the range $[0,1]$. For the tasks having deterministic computing load, l the server load can be computed as the sum of computing loads of all of the running tasks. Alternatively, for the tasks with predefined completion deadline, the server load l can be expressed as the minimum amount of computational resource required from the server to complete all the tasks right-in-time.

Being assigned into racks, the servers share the ToR switch uplink channels for their communication demands. However, defining a portion of this bandwidth used by a given server or a flow at the gigabit speeds during runtime is a computationally expensive task. To circumvent the aforementioned undesirable characteristic, both Eqs. (3) and (4) include a component, which is dependent on the occupancy level of the outgoing queue $Q(q)$ at the switch and scales with the bandwidth over provisioning factor δ.

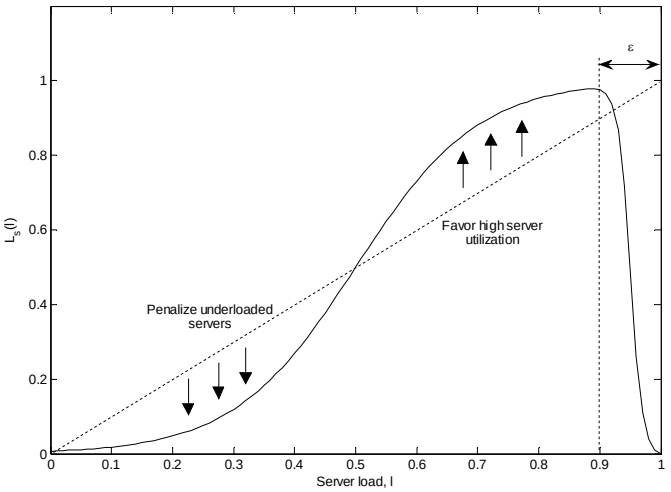

Figure 2. Computing server selection by DENS metric.

Instead of relying on the absolute size of the queue, the occupancy level q is scaled with the total size of the queue Q_{max} within the range $[0,1]$. The range corresponds to none and full buffer occupancy. By relying on buffer occupancy, the DENS metric reacts to the growing congestion in racks or modules rather than transmission rate variations. To satisfy the aforementioned behavior, $Q(q)$ is defined using inverse Weibull cumulative distribution function:

$$Q(q) = e^{-\left(\frac{2q}{Q_{max}}\right)^2}. \quad (7)$$

The obtained function, illustrated in Fig. 3, favors empty queues and penalizes fully loaded queues. Being scaled with the bandwidth over provisioning factor δ in Eq. (3) and Eq. (4) it favors the symmetry in the combined uplink and downlink bandwidth capacities for switches when congestion level is low. However, as congestion grows and buffers overflow, the bandwidth mismatch becomes irrelevant and immeasurable. The Eq. (7) is inspired by the Random Early Detection (RED) [22] and Backward Congestion Notification (BCN) [23] technologies.

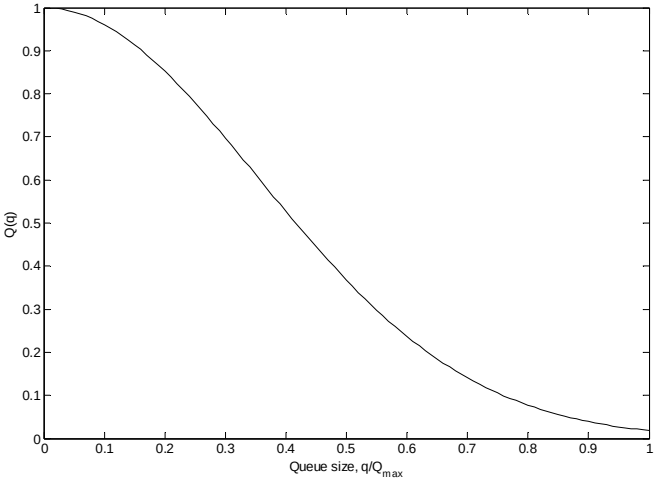

Figure 3. Queue selection by DENS metric.

Fig. 4 presents the combined $f_s(l,q)$ as defined in Eq. (3). The obtained bell-shaped function favors selection of servers with the load level above average located in racks with the minimum or no congestion.

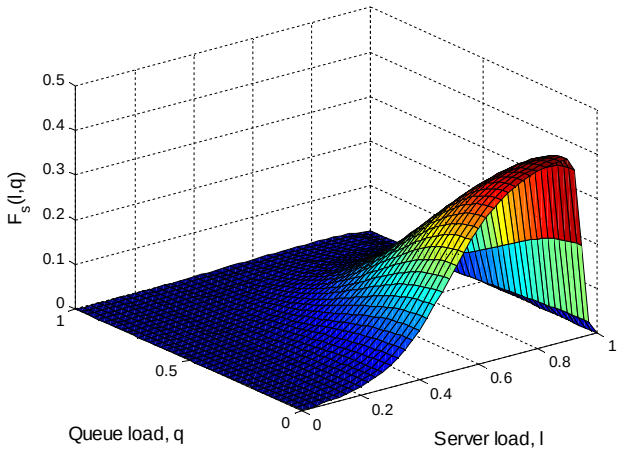

Figure 4. Server selection by DENS metric according to its load and communicational potential.

IV. Performance Evaluation

For performance evaluation purposes, the proposed DENS methodology was implemented in the GreenCloud simulator [21, 32]. The GreenCloud is a cloud computing simulator designed to capture data center communication processes at the packet level. It implements a set of energy-efficient optimization techniques, such as DVFS [27] and DPM [28], and offers tools to monitor energy consumption in data center servers, switches, and other components.

A three-tier tree data center topology comprised of 1536 servers arranged into 32 racks each holding 48 servers, served by 4 core and 8 aggregation switches (see Fig. 1), was used in all simulation experiments. We used 1 GE links for interconnecting servers in the inside racks while 10 GE links were used to form a fat-tree topology interconnecting access, aggregation, and core switches. The propagation delay on all of the links was set to 10 ns.

The workload generation events are exponentially distributed in time to mimic typical process of user arrival. As soon as a scheduling decision is taken for a newly arrived workload it is sent over the data center network to the selected server for execution. The size of the workload is equal to 15 KB. Being fragmented, it occupies 10 Ethernet packets. During execution, the workloads produce a constant bitrate stream of 1 Mb/s directed out of the data center. Such a stream is designed to mimic the behavior of the most common video sharing applications. To add uncertainties, during the execution, each workload communicates with another randomly chosen workload by sending a 75 KB message internally. The message of the same size is also sent out of the data center at the moment of task completion as an external communication.

The average load of the data center is kept at 30% that is distributed among the servers using one of the three evaluated schedulers: (a) DENS scheduler proposed in Section III of this paper, (b) Green scheduler performing the best-effort workload consolidation on a minimum set of servers, and (c) a round-robin scheduler which distributes the workloads equally.

The servers left by the schedulers idle are powered down using DPM technique to reduce power consumption. A similar technique is applied to the unused network switches in aggregation and access networks. The core network switches remain always operational at the full rate due to their crucial importance in communications.

Fig. 5 presents the server load distribution for all three of the evaluated schedulers. Fig. 6 reports a combined uplink load at the corresponding rack switches. The Green scheduler consolidates the workload leaving the most (around 1016 on average) servers idle in the evaluated data center. These servers are then powered down. However, the load of the loaded servers (left part of the chart) is kept close to the maximum and no consideration of network congestion levels and communication delays is performed. As a consequence, a number of workloads scheduled by the Green scheduler produces a combined load exceeding ToR switch forwarding capacity and causes network congestion. The round-robin scheduler follows a completely opposite policy. It distributes computing and communicational loads equally among servers and switches; thereby the network traffic is balanced and no server is overloaded. However, the drawback is that no server or network switch is left idle for powering down, making the round-robin scheduler as the least energy-efficient.

The DENS methodology achieves the workload consolidation for power efficiency while preventing computing servers and network switches from overloading. In fact, the average load of an operating server is around 0.9 and the average load of the rack switch uplink is around 0.95. Such load levels ensure that no additional delays in job communications are caused by network congestion. However, this advantage comes at a price of a slight increase in the number of running servers. On average, DENS scheduler left 956 servers as opposed to 1016 servers left idle by the Green scheduler.

To explore the uplink load in detail, we measured the traffic statistics at the most loaded switch ToR switch (the leftmost in Fig. 6). Fig. 7 presents a combined ToR switch uplink load evolution, while Fig. 8 presents the uplink queue evolution at the same switch for the first 15 seconds of simulation time. Under the Green scheduler, the link is constantly overloaded and the queue remains almost constantly full, which causes multiple congestion losses. All queues were limited to 1000 Ethernet packets in our simulations. Under the DNS scheduler, the buffer occupancy is mostly below the half of its size with an average of 213 packets, displayed with a dashed line in Fig. 8. At certain instances of time the queue even remains empty having no packets to send. This fact results in a slightly reduced uplink utilization level of 0.95.

Table I compares the impact of different scheduling policies on the level of data center energy consumption. The data is collected for an average data center load of 30%. The most energy inefficient is a round-robin scheduler. It does not allow any of the servers or network switches to be powered down for the whole duration of data center operation.

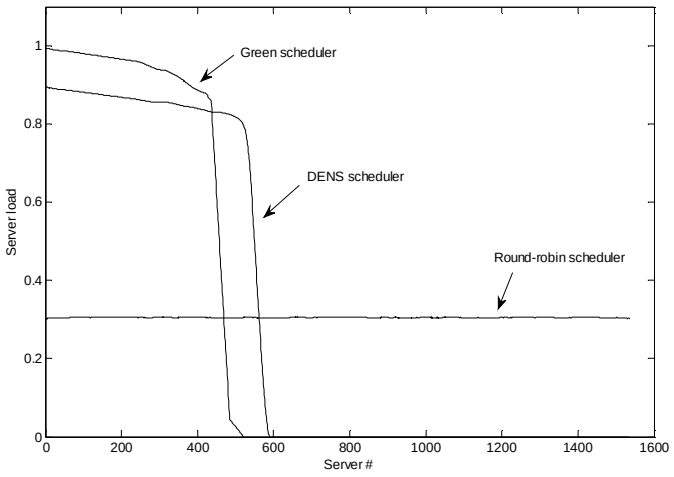

Figure 5. Server workload distribution performed by DENS, Green, and round-robin schedulers.

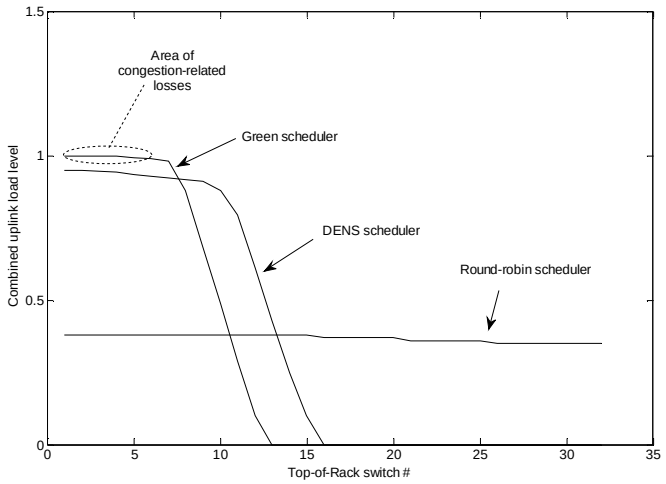

Figure 6. Combined uplink traffic load at the rack switches.

The Green scheduler is the most efficient. It releases around two-thirds of servers and network switches, which considerably reduces the energy consumption levels. With the Green scheduler, the data center energy consumption is slashed in half compared to when a round-robin scheduler is utilized. The DENS methodology when compared to the Green scheduler adds around: (a) 4% to the total data center consumption, (b) 3% in servers' energy consumption, and (c) 1% in switches' energy consumption. This slight increase in energy consumption is justified by the need of additional computing and communicational resources, detected by DENS methodology, and required for keeping the quality of job execution at the desired level. In contrast to the Green scheduler, DENS methodology uses network awareness to detect congestion hotspots in the data center network and adjust its job consolidation methodology accordingly. It becomes particularly relevant for data intensive jobs which are constrained more by the availability of communication resources rather than by the available computing capacities.

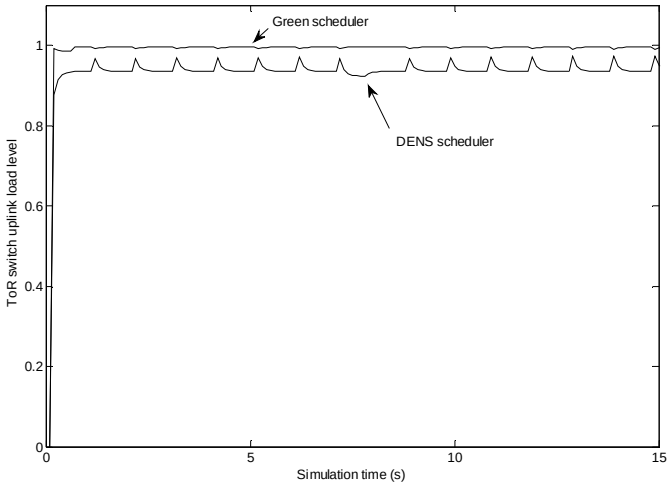

Figure 7. ToR switch uplink load.

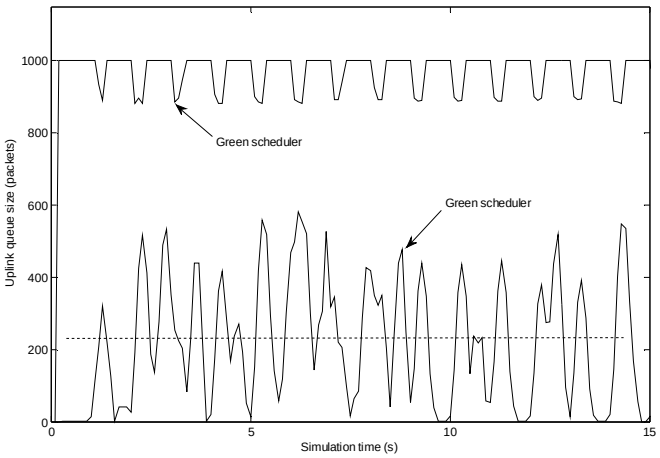

Figure 8. ToR switch uplink buffer occupancy.

TABLE I. DATA CENTER ENERGY CONSUMPTION

Parameter	Power Consumption (kW·h)		
	Round Robin scheduler	***Green Scheduler***	***DENS scheduler***
Data center	417.5K	203.3K (48%)	212.1K (50%)
Servers	353.7K	161.8K (45%)	168.2K (47%)
Network switches	63.8K	41.5K (65%)	43.9K (68%)

V. CONCLUSIONS

This paper underlines the role of communication fabric in data center energy consumption and presents a methodology, termed DENS, that combines energy-efficient scheduling with network awareness. The DENS methodology balances the energy consumption of a data center, individual job performance, and traffic demands. The proposed approach optimizes the tradeoff between job consolidation (to minimize the amount of computing servers) and distribution of traffic patterns (to avoid hotspots in the data center network).DENS methodology is particularly relevant in data centers running data-intensive jobs which require low computational load, but produce heavy data streams directed to the end-users.

The simulation results obtained for a three-tier data center architecture underline DENS operation details and its ability to maintain the required level of QoS for the end-user at the expense of the minor increase in energy consumption. Future work will focus on the implementation and testing of DENS methodology in realistic setups using testbeds.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the funding from Luxembourg FNR in the framework of GreenIT project (C09/IS/05) and a research fellowship provided by the European Research Consortium for Informatics and Mathematics (ERCIM).

REFERENCES

[1] X. Fan, W.-D. Weber, and L. A. Barroso, "Power Provisioning for a Warehouse-sized Computer," In Proceedings of the ACM International Symposium on Computer Architecture, San Diego, CA, June 2007.

[2] R. Raghavendra, P. Ranganathan, V. Talwar, Z. Wang, and X. Zhu, "No "Power" Struggles: Coordinated Multi-level Power Management for the Data Center," APLOS 2008.

[3] Gartner Group, available at: http://www.gartner.com/

[4] J. Liu, F. Zhao, X. Liu, and W. He, "Challenges Towards Elastic Power Management in Internet Data Centers", in Proceedings of the 2nd International Workshop on Cyber-Physical Systems (WCPS 2009), in conjunction with ICDCS 2009., Montreal, Quebec, Canada, June 2009.

[5] Bo Li, Jianxin Li, Jinpeng Huai, Tianyu Wo, Qin Li, and Liang Zhong, "EnaCloud: An Energy-Saving Application Live Placement Approach for Cloud Computing Environments," IEEE International Conference on Cloud Computing, Bangalore, India, 2009.

[6] Li Shang, Li-Shiuan Peh, and Niraj K. Jha, "Dynamic Voltage Scaling with Links for Power Optimization of Interconnection Networks," Proceedings of the 9th International Symposium on High-Performance Computer Architecture table of contents, 2003.

[7] Ying Song, Hui Wang, Yaqiong Li, Binquan Feng, and Yuzhong Sun, "Multi-Tiered On-Demand Resource Scheduling for VM-Based Data Center," IEEE/ACM International Symposium on Cluster Computing and the Grid (CCGRID), pp. 148 – 155, May 2009.

[8] A. Beloglazov, and R. Buyya, "Energy Efficient Resource Management in Virtualized Cloud Data Centers," IEEE/ACM International Conference on Cluster, Cloud and Grid Computing (CCGrid), pp. 826 – 831, May 2010.

[9] Qin Tang, S. K. S. Gupta, and G. Varsamopoulos, "Energy-Efficient Thermal-Aware Task Scheduling for Homogeneous High-Performance Computing Data Centers: A Cyber-Physical Approach," IEEE Transactions on Parallel and Distributed Systems, vol.19, no. 11, pp. 1458 – 1472, November 2008.

[10] M. Al-Fares, S. Radhakrishnan, B. Raghavan, N. Huang, and A. Vahdat, "Hedera: Dynamic Flow Scheduling for Data Center Networks," in Proceedings of the 7th USENIX Symposium on Networked Systems Design and Implementation (NSDI '10), San Jose, CA, April 2010.

[11] A. Stage and T. Setzer, "Network-aware migration control and scheduling of differentiated virtual machine workloads," in Proceedings of the 2009 ICSE Workshop on Software Engineering Challenges of Cloud Computing, International Conference on Software Engineering. IEEE Computer Society, Washington, DC, May 2009.

[12] Xiaoqiao Meng, V. Pappas, and Li Zhang, "Improving the Scalability of Data Center Networks with Traffic-aware Virtual Machine Placement," IEEE INFOCOM, San Diego, California, March 2010.

[13] P. Mahadevan, P. Sharma, S. Banerjee, and P. Ranganathan, "Energy Aware Network Operations," IEEE INFOCOM Workshops, pp. 1 – 6, 2009.

[14] D. Thaler and C. Hopps, "Multipath issues in unicast and multicast nexthop selection," Internet Engineering Task Force Request for Comments 2991, November 2000.

[15] IEEE std 802.3ba-2010, "Media Access Control Parameters, Physical Layers and Management Parameters for 40 Gb/s and 100 Gb/s Operation," June 2010.

[16] Chuanxiong Guo, Haitao Wu, Kun Tan, Lei Shiy, Yongguang Zhang, and Songwu Luz, "DCell: A Scalable and Fault-Tolerant Network Structure for Data Centers," ACM SIGCOMM, Seattle, Washington, U.S.A., 2008.

[17] Chuanxiong Guo, Guohan Lu, Dan Li, Haitao Wu, Xuan Zhang, Yunfeng Shi1, Chen Tian, Yongguang Zhang1, and Songwu Lu, "BCube: A High Performance, Server-centric Network Architecture for Modular Data Centers," ACM SIGCOMM, Barcelona, Spain, 2009.

[18] G. Chen, W. He, J. Liu, S. Nath, L. Rigas, L. Xiao, and F. Zhao, "Energy-aware server provisioning and load dispatching for connection-intensive internet services," the 5th USENIX Symposium on Networked Systems Design and Implementation, Berkeley, CA, USA, 2008.

[19] M. Alizadeh, B. Atikoglu, A. Kabbani, A. Lakshmikantha, Rong Pan, B. Prabhakar, and M. Seaman, "Data center transport mechanisms: Congestion control theory and IEEE standardization," Annual Allerton Conference on Communication, Control, and Computing, September 2008.

[20] C. Kopparapu. "Load Balancing Servers, Firewalls, and Caches," John Wisely & Sons Inc., 2002.

[21] D. Kliazovich, P. Bouvry, Y. Audzevich, and S. U. Khan, "GreenCloud: A Packet-level Simulator of Energy-aware Cloud Computing Data Centers," IEEE Global Communications Conference (GLOBECOM), Miami, FL, USA, December 2010.

[22] S. Floyd and V. Jacobson, "Random Early Detection gateways for Congestion Avoidance," IEEE/ACM Transactions on Networking, vol, 1 no. 4, pp. 397 - 413, August 1993.

[23] D. Bergamasco, A. Baldini, V. Alaria, F. Bonomi, and R. Pan, "Methods and Devices for Backward Congestion Notification," US Patent 2007/0081454.

[24] R. Brown et al., "Report to congress on server and data center energy efficiency: Public law 109-431," Lawrence Berkeley National Laboratory, 2008.

[25] A. Berl, E. Gelenbe, M. Di Girolamo, G. Giuliani, H. De Meer, Minh Quan Dang, and K. Pentikousis, "Energy-Efficient Cloud Computing," The Computer Journal, vol. 53, no. 7, pp. 1045 – 1051, 2009.

[26] "Cisco Data Center Infrastructure 2.5 Design Guide," Cisco press, March 2010.

[27] J. Pouwelse, K. Langendoen, and H. Sips, "Energy priority scheduling for variable voltage processors," International Symposium on Low Power Electronics and Design, pp. 28 – 33, 2001.

[28] L. Benini, A. Bogliolo, and G. De Micheli, "A survey of design techniques for system-level dynamic power management," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 8, no. 3, pp. 299 – 316, June 2000.

[29] P. Mahadevan, P. Sharma, S. Banerjee, and P. Ranganathan, "A Power Benchmarking Framework for Network Devices," in Proceedings of the 8th international IFIP-TC 6 Networking Conference, Aachen, Germany, May 11 - 15, 2009.

[30] S. Garrison, V. Oliva, G. Lee, and R. Hays, "Ethernet aliance: Data Center Bridging," November 2008.

[31] IEEE 802.1 Data Center Bridging Task Group, available at: http://www.ieee802.org/1/pages/dcbridges.html

[32] D. Kliazovich, P. Bouvry, Samee U. Khan, "GreenCloud: A Packet-level Simulator of Energy-aware Cloud Computing Data Centers," Journal of Supercomputing, special issue on Green Networks, 2011.