Energy-efficient bandwidth reservation for bulk data transfers in dedicated wired networks

Anne-Cécile Orgerie, Laurent Lefèvre, Isabelle Guérin Lassous

Energy-Efficient Bandwidth Reservation for Bulk Data Transfers in Dedicated Wired Networks

Anne-Cécile Orgerie, Laurent Lefèvre and Isabelle Guérin-Lassous

ENS de Lyon - INRIA RESO - Université Claude Bernard Lyon 1 - LIP (UMR CNRS, INRIA, ENS, UCBL)

École Normale Supérieure - 46, allée d'Italie 69364 LYON Cedex 07 - FRANCE

Annececile.Orgerie@ens-lyon.fr, Laurent.Lefevre@inria.fr, Isabelle.Gurein-Lassous@ens-lyon.fr

Abstract

The ever increasing number of Internet connected end-hosts call for high performance end-to-end networks leading to an increase in the energy consumed by the networks. Our work deals with the energy consumption issue in dedicated network with bandwidth provisionning and in-advance reservations of network equipments and bandwidth for Bulk Data transfers.

First, we propose an end-to-end energy cost model of such networks which described the energy consumed by a transfer for all the crossed equipments. This model is then used to develop a new energy-aware framework adapted to Bulk Data Transfers over dedicated networks. This framework enables switching off unused network portions during certain periods of time to save energy. This framework is also endowed with prediction algorithms to avoid useless switching off and with adaptive scheduling management to optimize the energy used by the transfers.

1 Introduction

The ever increasing number of Internet connected end-hosts call for high performance end-to-end networks. One particular network traffic is on the rise: the traffic generated by emerging applications, such as e-Science collaborative applications [42], that require transferring Tbytes of data on a near daily basis. Those applications tolerate delivery delays while requiring to be delivered before strict deadlines. These transfers are designated as Bulk Data Transfers and are described as moldable or flexible transfer jobs.

From science, to engineering and business domains, large-scale applications generate huge amounts of data which must be distributed over wide geographical areas. More and more, high-speed dedicated links are set up to support this huge traffic and all the associated remote management tasks. Research networks can be examples of such dedicated networks using bandwidth reservation [51], like UltraScience Net[1],

[1] UltraScience Net: `http://www.csm.ornl.gov/ultranet/`

OSCARS (On-demand Secure Circuits and Advance Reservation System)[2] and DRAGON (Dynamic Resource Allocation via GMPLS Optical Networks)[3].

In order to ensure quality of service and to respect deadline and bandwidth constraints for this type of traffic, bandwidth reservation can be used over the networks dedicated to these applications. Such a tight coordination mechanism leads to see the network as a service.

The growth of network users and applications leads to an increase in the topology complexity, an increase in the number of core equipments to ensure performance and reliability, and consequently, an increase in the energy consumed by the networks [4].

To guarantee great reactivity and availability, these dedicated networks are always fully powered on and running at full capacity even if there is no traffic. Thus, by taking advantage of the fully controlled nature of these networks, we show that great energy savings are feasible without impacting the quality of service they deliver.

The first step is to understand how the energy is used by the equipments on which these dedicated networks rely. The energy consumption of each equipment is included in the global energy cost model that we design and which takes into account the network topology and the traffic passing through the modeled network. This is a generic energy cost model that works for every kind of networks.

We have then used this energy cost model to develop a new energy-aware framework adapted to Bulk Data Transfers over dedicated networks. This framework enables switching off unused network portions during certain periods of time to save energy. This framework is also endowed with prediction algorithms to avoid useless switching off and with adaptive scheduling management to optimize the energy used by the transfers.

Our contributions include:

- an end-to-end energy cost model that considers the topology and the traffic to estimate the energy consumed by every networks;
- a network model which is adapted to Advance Bandwidth Reservations (ABR) for Bulk Data Transfer (BDT).
- a new complete and energy-efficient BDT framework including scheduling algorithms which provide an adaptive and predictive management of the ABRs.

This paper is organized as follows: Section 2 presents the related works in both Bulk Data Transfer and green networking domains. Section 3 describes the used models: the end-to-end energy model and the underlying network model we have designed, and states the problem formulation. The management algorithms of our new BDT framework are detailed in Section 4. Section 5 gives the conclusion and opens on future works.

[2]OSCARS: `http://www.es.net/oscars`
[3]DRAGON: `http://dragon.maxgigapop.net`

2 Related Works

2.1 Bulk Data Transfer and Bandwidth Provisionning

Typical Bulk Data Transfer applications include peer-to-peer protocols [60, 34] and Content Delivery Network facilities [16] with media servers that require timely transfer of large amounts of data among these different servers [56]. They can even be used on the Internet with water-filling techniques [41] which are compared in terms of performance and cost to the courier service *FedEx*.

BDT applications are numerous and present different characteristics, thus numerous BDT frameworks have been developped to solve their particular issues. Our work is focused on bandwidth provisionning for BDT.

2.1.1 Bandwidth allocation and routing algorithms

To provision bandwidth for BDT, the two main problems are to allocate bandwidth in time and in space. These two issues are solved respectively by bandwidth scheduling and path computation algorithms. Two basic provisionning modes are commonly distinguished [55, 39, 28]:

(1) on-demand mode: a connection request is made when needed, and it is then accepted or denied depending on the current bandwidth availability, arriving requests are queued until their bandwidth allocation; and

(2) in-advance mode: a connection request is granted for future time-slots based on bandwidth allocation schedules, arriving requests are scheduled in the future as soon as they arrive (system of agendas).

Two approaches can be taken to provision bandwidth over the network [49];

(1) the centralized approach: a centralized server maintains all resource availability information, and all reservation requests may retrieve availability information from the centralized server; and

(2) the distributed approach: each node maintains its resource availability information, reservation scheduling is done in a collaborative way.

The main problem with the centralized approach is the scalability issue [71], while with the distributed approach, it is the request processing time. The network management system is often called *bandwidth broker* [13, 71] which is in charge of the reservation requests admission control.

Many problems related to bandwidth allocation and path computation are NP-complete. In [44], Lin *et al.* describes two basic scheduling problems: fixed path with variable bandwidth and variable path with variable bandwith with a view to minimize the transfer end time of a given data size. They prove that these both problems are NP-complete and they propose greedy heuristic algorithms to solve them. In [43], they consider two other problems dealing with multiple data transfers. The bandwidth allocation and path computation algorithms are mainly inspired from Dijkstra and Bellman-Ford algorithms [55, 43, 39].

2.1.2 Network protocols

Usual network protocols are not adapted to dedicated networks since they are designed to work in a best-effort mode with congestion, failures and resource competition. Dedicated networks have different characteristics: high-speed, reliability and high-delay bandwidth product among others. Since a long time, it is well-known that TCP is inefficient in this kind of environment [38].

New protocols using UDP for data transfer and TCP for control like reliable blast UDP [33] or SABUL [25] have been developped. Other emerging solutions include a better adaptivity to maximize the data rate according to the receiver's capacity and to maximize the goodput by minimizing synchronous, latency-bound communication (Adaptive UDP [22], FRTP: Fixed Rate Transport Protocol [72]). These protocols are implemented on top of UDP as application-level processes.

Furthermore, specific reservation protocols have been developped since a long time like RSVP (ReSer-Vation Protocol) [70] where resources are reserved across a network for integrated services in QoS-oriented networks. RSVP also allows protocols to be designed and used on top of it to complete its functionnalities [56].

2.1.3 Advance reservation algorithms

On-demand mode can be seen as a special case of in-avance mode [39]. Thus focusing on advance reservations do not restrict our scope. The idea of making advance reservations of network resources is not recent [52].

The main problem for advance reservations in network environements is the unpredictability of the routing behavior. However, with the emergence of the MPLS (Multi-Protocol Label Switching) [54] standard with traffic engineering and explicit routing features, it becomes possible to disconnect the reservation management from the network layer, thus leading to an easier inter-operability for the ABR management systems. While MPLS insures the support of on-demand capabilities at layer 3, GMPLS (Generalized MPLS) provides the same functions at layer 2 and 1.

Different BDT scheduling techniques can be used: online scheduling where requests are processing as soon as they arrive or periodic batch scheduling where are scheduled with certain periodicity [42]. Different time models can also be used: continuous time models [39, 42] and discrete models [13, 51] with fixed time slots (slices) during which the resource allocations are similar.

Several other issues related to ABRs have been explored: fault tolerance [14], rerouting strategies [15], load-balancing strategies [68], time-shift reservations [49], etc.

Several visions, positions and propositions on bandwidth reservations for BDT in dedicated networks: on-demand vs. in-advance modes, online vs. periodic batch scheduling, centralized vs. decentralized reservation managements, continous time vs. time slot models. However, for the moment, none of the proposed solutions take network energy consumption as a mjor issue which should influence the design

of each algorithm related to the network management, from scheduling to routing. Yet, in 2007, at Michiharu Nakamura from Hitachi estimates that by 2015, routers will consume 9% of Japan's electricity[4]. Moreover, the electric bill grows as the traffic grows: exponentially due to the increase of required infrastructure.

2.2 Green wired networking

In [29], Gupta & Singh have shown that transmitting data through wired networks takes more energy in bits per Joule than transmitting data through wireless networks. Energy is indeed one of the main concern of wireless networks while, for now, it is not the case for wired networks since they are not battery constrained. However, the energy issue is becoming more and more present in wired networks because of the need to maintain network connectivity at all times [21].

The ever increasing demand in energy can yet be greatly reduced. Studies have indeed shown since few years that network links, and especially edge links, are lightly utilized [21, 48]. This fact has lead researchers to propose several approaches to take advantage from the link under-utilisation in order to save energy.

We have classified these approaches in several categories: the **optimization** approach that is based on improvements of the hardware components, such as routers and Network Interface Controllers (NICs) for example; the **shutwodn** approach that takes advantage of the idle periods to switch off the network components, such as switches and router's ports for example; the **slowdown** approach that puts in low power modes the network components during under-utilization periods; and the **coordination** approach that advocates a network-wide power management and global solutions including energy efficient routing for example.

2.2.1 Energy consumption

Before beeing able to save energy with new technologies and mechanisms, researchers and network designers need to know how energy is consumed in network equipments. This preliminary analysis is the key to understand how energy can be saved and to design energy models of network equipments that will be used to validate new hardware components and new algorithms.

Several models have been proposed for the different network components [66, 3] and for the whole Internet [5, 4]. Based on real energy measurements, they allow researchers to validate their new frameworks and algorithms.

[4]Nature Photonics Technology Conference report available at `http://www.natureasia.com/en/events/photonics/2007_photon_conf_report.eps`

2.2.2 The *optimization* approach: hardware improvements

The first way to reduce the energy consumption of a component is to increase its energy efficiency. That's why network equipment manufacturers are proposing more and more *green* routers and switches [3] for example.

D-Link [5], Cisco [6], Netgear[7] are among the manufacturers proposing new *green* functionnalities in there products such as the power transmission adaptation to the link length, the power adaptation to the load, power off buttons, more energy efficient power supply, etc.

These new products come with green initiatives (GreenTouch [8], GreenStar Network [9], ECR initiative [10], etc.) and study groups (such as the IEEE 802.3 Energy Efficient Ethernet Study Group [11]) which aim to standardize and to enforce new regulations in terms of energy consumption for network equipments.

2.2.3 The *shutdown* approach: sleeping

Network links, and especially edge links, are lightly utilized [21, 48]. So, researchers have proposed to switch off (sleeping mode) the network equipments when they are not used [29, 20]. This technique raises several problems: connectivity loss, long re-synchronization time, to be always switching on and off can be more energy consuming than doing nothing.

New mechanisms have been designed to settle these issues: proxying techniques to keep the connectivity alive [47], and new mechanism to quickly re-synchronized both ends of a link [30] for example.

The shutdown approach takes advantage of idle periods in network traffic. Energy savings can also be made during low-demand period with the slowdown approach.

2.2.4 The *slowdown* approach: rate adaptation

The authors of [26] have shown that there is a negligible difference in power consumption whether an Ethernet link is idle of fully utilized. This is mainly due to the fact that when there is not data to transmit on the link, idle bit patters are still continuously transmitted in order to keep both NICs synchronized.

However, Gigabit Ethernet (1000BASE-T) specifications include backward compatibility with previous specifications, and thus, are also designed to operate at 10Mb/s, 100Mb/s and 1Gb/s. Moreover, when NICs and switches operate at lower data rates, they consume less energy [26].

This observation has lead several research teams to propose methods to dynamically adjust link data rates to the load [57, 9] based on the same principle as Dynamic Voltage Frequency Scaling (DVFS)

[5] http://dlinkgreen.com/energyefficiency.asp
[6] http://www.cisco.com/en/US/products/ps10195/index.html
[7] http://www.netgear.com/NETGEARGreen/GreenProducts/GreenRoutersGateways.aspx
[8] http://www.greentouch.org/
[9] http://www.greenstarnetwork.com/
[10] http://www.ecrinitiative.org/
[11] http://grouper.ieee.org/groups/802/3/eee_study/index.html

techniques for CPUs. This technique can only be used with mechanisms to quickly switch the data rate of an Ethernet link [11].

2.2.5 The *coordination* approach: network-wide management and global solutions

The optimization, shutdown and slowdown approaches are focused on specific network components. However, energy efficiency improvements have also to be made at wider scales. For example, routing algorithms [17, 53] and network protocols [37, 10] can be improved to save energy.

Coordinated power management schemes benefit from previously cited techniques, such has on/off and adapting rate techniques, and take decisions at a wider scale, and that implies that they make greater energy savings. For example, in low-demand scenarios with network redundancy, entire network paths can be switched off, and the traffic is routed on other paths [59, 58].

3 Models

3.1 End-to-End Energy Cost Model: ECOFEN

Energy savings are only possible at the cost of some formalization and modeling! Indeed, a deep understanding of where the electricity is wasted is required to develop energy-aware frameworks capable of taking the right decisions to optimize the energy consumption without impacting the system's performances.

Networks consist of several facilities: routers, switches, bridges, repeaters, hubs, firewalls, wired links, coaxial cables, optical fibers, twisted pair wires, antennas, wireless transmitters, network interface cards, access points, etc.

Each of these equipments have its own energy consumption scheme with different parameters influencing this consumption: type of equipment, traffic, load, number of connected equipments, number of switched on interfaces (ports), used protocols (that can add some processing time if high level operations are required such as flow identification, packet inspecting, etc.), energy saving modes that are used on the equipment, etc.

As seen in Section 2, several energy models have been made to express the electric consumption of particular equipments (routers [66], switches [3, 35], etc.) or particular networks (backbone networks [19], optical networks [64, 65], etc.). Nevertheless, the model presented here is the first that can be applied to any network with any networking equipments and any traffic. Contrary to other propositions, this model is an end-to-end model which gives the consumption of end-to-end communications including the energy consumption of each crossed equipment taking care of other comunications sharing these same equipments.

The energy consumption E of an equipment depends on the power consumption P of the equipment which varies over time t. For a given time period with a length equals to T, the energy is given by:

$$E(T) = \int_0^T P(t)\ dt \tag{1}$$

The energy consumption functions (per equipment) are linked to traffic with a weight more or less high and thus these functions vary over time. For example, the consumption of a router depends on the number of ports that are switched on and this value vary over time depending on the traffic.

Up to now, we have only considered the energy consumption as the factor to optimize. What about the money cost? The paid price D depends on the energy consumed E and also on the time it has been consumed:

$$D(E) = \sum_i NbKWh(t_i) \times Price(t_i) \tag{2}$$

where t_i represents a time period with a fixed energy cost $Price(t_i)$ and $NbKWh(t_i)$ is the number of KWh consumed during that time period. Indeed, the energy price vary over time (during slack periods such as nights, the electricity is cheaper). So, an optimization of the energy consumption does not necessarily leads to an optimization of the billing cost [50]. As our goal is to save energy first, we have choose to optimize the energy consumption without taking into account the electricity price.

In our model, we only consider as consuming equipments the facilities that are plugged and that consume electicity. That is to say that the links are not considered as consuming equipments. However, their "cost" is reflected in the equipments they link. Indeed, if a router, for example, embeds some energy saving features, the power used for the transmissions is related to the lenght of its output wired link. It requires more power with long links. A similar approach has been developped since a long time for wireless networks [36]: the transmitting power is adjusted according to the distance between the sender and its receiver to save energy [63]. Currently, by default, this energy saving feature is not applied in the routers: they use full power transmissions on all of their ports.

For a given equipment (router, repeater, ethernet card), the energy consumed by a transfer is given by:

$$E = E_{boot} + E_{work} + E_{halt} \tag{3}$$

where E_{boot} and E_{halt} can be equal to zero if we don't need to boot and to halt the equipment (due to transfer aggregation for example or if it stays always powered on). Network equipment manufacturers and designers are well placed to influence the energy consumed by the booting and halting periods: they can work on the hardware components to shorten the booting and halting durations. However, protocol

designers have also a role to play in the quest for energy efficiency (which is a Holy Grail [32]). Indeed, protocol re-synchronisation between linked nodes needs to be reduced to make the on/off technique usable [2, 57]. The optimization of E_{boot} and E_{halt} is out of the scope of this paper.

Each equipment has a fixed energy cost (E_{idle}) which corresponds to the energy consumed when it does nothing (no transfer), and a variable cost which depends on the traffic. The energy E_{work} consumed during the transfer includes this two costs and depends on:

- BD the bandwidth used by the transfer,
- L the length in time of the transfer
- the cross traffic on this equipment: we do not want to count several times the consumption of a router for example if several transfers are using it at the same time (we want to divide it by the number of transfer for the fixed costs)
- the type of equipment (router, NIC, ...)

The fixed part consists only of the latter parameter, the three other ones are variable and are linked to time. These interactions between network usage and energy consumption are the subject of several power cost models. Classical models include models with link rate switching which have a strong dependance to the link rate and a slight dependance to transmission rate as shown in Figure 1 by the line labelled *Adaptive Link Rate*. This graph does not show the energy and the time required to switch from one link rate to another one.

Another classical but less realistic model is the proportional power cost model. This model is not realistic for current network equipments. However, some studies affirms that efforts should be made to reach this model [6] in order to obtain a more fare model.

Figure 1 presents the power consumption of a particular equipment during a transfer. Yet, for a given equipment, let say a router for example, we also divide this consumption in two parts:

- the power consumed by the ports during transfers, which is in fact the variable part of the energy consumption;
- and the power consumed by the router itself when there is no transfers (the idle consumption of the router) which represents the fixed part.

From this energy model for a given equipment, it results that the total energy used by a transfer from Node A to Node B on the example desplayed on Figure 2 through Router 1 and 2, if there is no cross traffic, is:

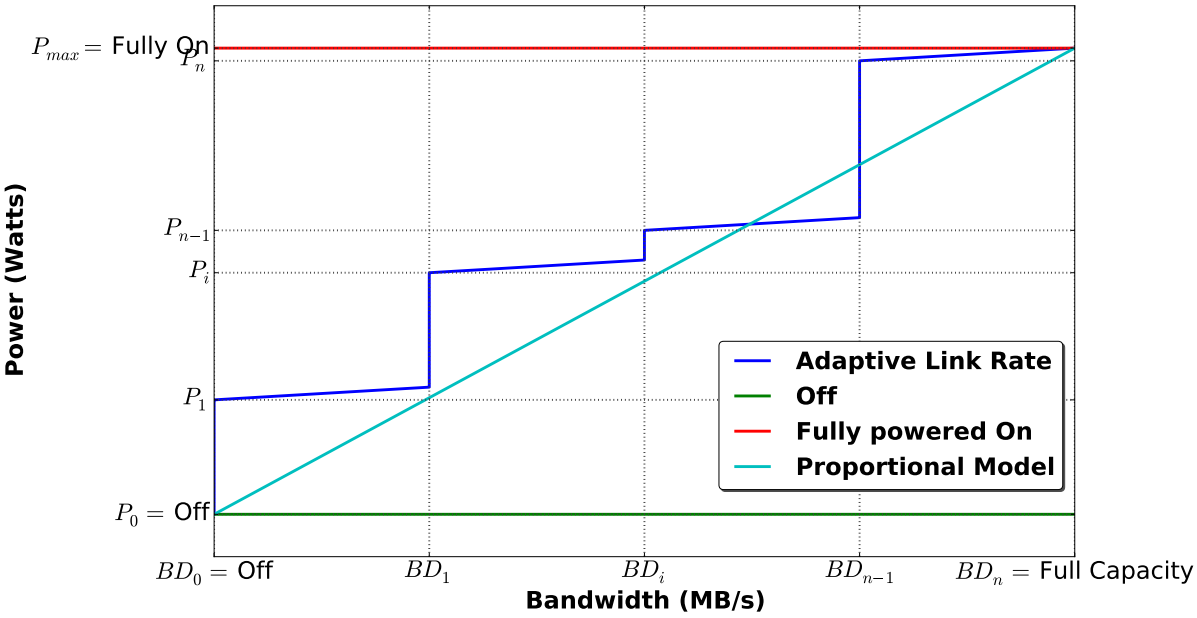

Figure 1: Models of power cost as a function of bandwidth

$$\begin{aligned} E_{tranfer} &= E_{EthernetCard}(NodeA, BD, L) + E_{Router}(Router1) \\ &+ E_{Port}(In, Router1, BD, L) + E_{Port}(Out, Router1, BD, L) \\ &+ E_{Port}(In, Router2, BD, L) + E_{Port}(Out, Router2, BD, L) \\ &+ E_{Router}(Router2) + E_{EthernetCard}(NodeB, BD, L) \end{aligned} \quad (4)$$

The functions E_{router} and E_{port} are different for each router. The energy consumed by an ethernet card ($E_{EthernetCard}$) depends on:

- its model (capacity, manufacturer, hosting node, etc.), this represents the fixed cost;
- BD the bandwidth used by the transfer;
- and L the length in time of the transfer.

In the same way, the energy consumed by a router (E_{router}) for a given transfer depends only on the router type (size, manufacturer), it is a fixed cost. But, if several transfers are using the same routers (cross traffic for example), this cost is devided among the transfers depending on their duration. The energy consumed by a router port during a transfer (E_{port}) depends on:

- the router's model;
- if the traffic is coming in or out;
- BD the bandwidth used by the transfer;

- and L the length in time of the transfer.

E_{port} is small compared to E_{router}.

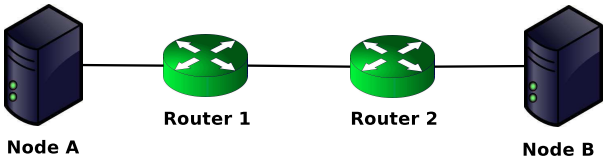

Figure 2: Scenario example

In the following, our model considers that the power function P_{work} is a piecewise affine function for the ports as presented on Figure 1 by the line labelled *Adaptive Link Rate*. This power function presents a strong dependance to the equipment state (ie. transmission rate in the case of a port) and a small dependance to the traffic. Using the notations of Figure 1, the power function of a port P_{work} can be written as a function of the bandwidth BD:

$$P_{work} = \begin{cases} P_0 & \text{if } BD = 0 \\ \alpha_1 BD + P_1 & \text{if } BD \in]0; BD_1] \\ \vdots & \\ \alpha_i BD + (P_i - \alpha_i BD_{i-1}) & \text{if } BD \in]BD_{i-1}; BD_i] \\ \vdots & \\ \alpha_n BD + (P_n - \alpha_n BD_{n-1}) & \text{if } BD \in]BD_{n-1}; BD_n] \end{cases} \quad (5)$$

where the α_i are the slopes ($\alpha_i > 0$) of the different linear portions (power levels) delimited by the BD_i (the different transmission rate levels) and the P_i define the start power of each different level.

We have called this model **ECOFEN** (Energy Consumption mOdel For End-to-end Networks). It gives the energy consumption of a given network (topology, type of equipments, routing protocol) for a given traffic (bandwidth utilization). It is a generic model that can be used for any kind of network and traffic.

ECOFEN assumes that the power function of each equipment (P_{work}) is known. So, the α_i, BD_i and P_i are known for each equipment which power consumption depends on the bandwidth and the fixed costs are also known for all the equipments.

With this information, all the above energy consumption for a given data transfer are computable. This requires a preliminar calibration campaign with wattmeters for each type of equipment in order to plot the power profiles (power functions depending on the usage like in the examples of Figure 1) of the utilized network equipments. In the future, such basic values should be included in the manufacturer specifications for each network equipment.

3.2 Network Model

In this paper, we have been focused on a special kind of networks which seems to be more energy-aware. Indeed, the coordination among the network elements induced by the reservation process works for common energy and performance-driven goals rather than selfish per-user goals.

The targeted applications [13] include large-scale distributed applications, data intensive peer-to-peer computing, cloud computing with virtual machine and data migrations, processing of large-scale data like the CERN-LHC experiments for example, media streaming services [34], etc. Advance reservations are especially really useful for applications that require strong network quality-of-service (QoS) as it is the case in Content Delivery Networks (CDN) [16] for example. The respect of the QoS requirements implies a tight coordination among the network elements.

In a first step, we will assume that we have a dedicated network where we can control and implement management features in each router and network equipment. Interoperability issues will be discussed later.

The network itself is represented as a directed graph $G = (V, E)$.

3.2.1 Reservation model

First, some basic notions should be defined.

Reservation: the user (end host) submits a data transfer reservation which correspond to a data volume (10 GB for example) and a deadline (in two hours for example). These basic information requirements are the only ones required for simple data transfer requests. These transfers are **malleable**, they are flexible enough to use any transmission rate, to have variable transmission rates over the time, or to be splitted in several parts.

Additional features can be specified such as maximal and minimal bandwidths (for video streaming for example or if the receiver is limited by its storage capacities), transfer profiles (step functions that express variable bandwidth requirement over time). These transfers are called **rigid** in contrast with malleable transfers.

Agenda: each network equipment (router, switch, bridges, repeaters, hubs, transmitters) has two agendas per port (per outgoing link) for the both ways (in and out). An agenda stores all the future reservations concerning its one-way link. This information is sometimes called the book-ahead interval [13]. Figure 3 presents an example of such an agenda.

Furthermore, each network equipment has also an agenda stating the on and off periods and the switching stages. This global agenda is in fact the combination of all the per-port agendas of the equipment: when no port is used for a certain amount of time (not to small), the network equipment can be switched off. Usage prediction algorithms are used to avoid to switch off if the equipment is going to be useful in a near future. These prediction algorithms will be described later.

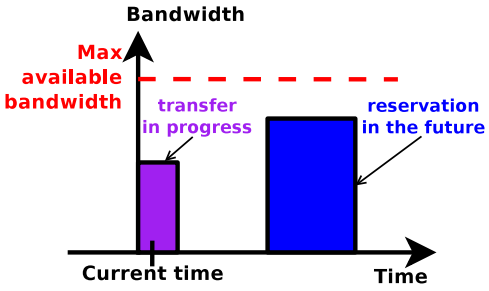

Figure 3: Simplified agenda

This model uses a continuous time model and not a discrete model with fixed time slots during which the resource allocations are similar [13, 51]. Indeed, as explained in [39], the storage of agendas is more flexible and less space using with the continuous time model. Thus, to store the per-port agendas, we use the time-bandwidth list structure used in several previous works [55, 45, 43]. Each port maintains its reservation status using a **time-bandwidth list** (TB list) which is formed by $(t[i], b[i])$ tuples, where $t[i]$ is a time and $b[i]$ is a bandwidth. These tuples are sorted in increasing order of $t[i]$. Thus $b[i]$ denotes the available bandwidth of the concerned port during the time period $[t[i], t[i+1]]$. If $(t[i], b[i])$ is the last tuple, then it means that a bandwidth of $b[i]$ is available from $t[i]$ to ∞. Each $t[i]$ is called an **event** in the agenda.

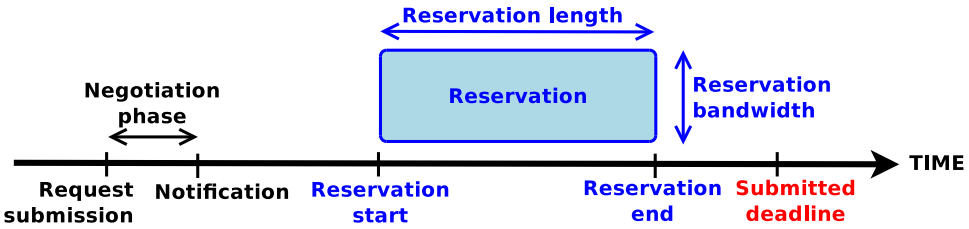

Figure 4: Reservation process

Figure 4 presents a typical sequence of events for the reservation process:

1. a user submits a reservation request (specifying at least the data volume and the required deadline) to the network management system.
2. the advance reservation environement launchs the negotiation phase including admission control, reservation scheduling and optimization policies.
3. the notification is sent to the user whether its request is accepted or not and when it is scheduled.
4. the reservation starts at the scheduled start time and ends at the scheduled end which occurs before the user-submitted deadline.

A reservation is not always a box as represented in Figure 4 with a fix bandwith during the whole transfer. A reservation can have variable rates during the transfer, especially malleable reservations and reservations with profiles.

The network equipments crossed during a transfer from node A to node B are called the **path** to go from A to B (set of vertices). During the reservation, the whole path taken by the transfered data is reserved, each crossed network equipment is reserved for this transfer, but not necessarily fully reserved. This mechanism guarantees a high level of QoS and gives the insurance that the deadline is well respected. This mechanism assumes that the network elements are, at least roughly, synchronized.

3.2.2 Protocol Model

Advance bandwidth reservations require a transport protocol adapted to their specific network characteristics: with high bandwidth, often high latency, and free from congestion. In that case, TCP is not appropriate and performant enough in particular because of its slow start and congestion window mechanisms. The used protocol should be:

- reliable
- rapid (few overhead due to the protocol)
- which can work with a fix bandwidth (allocate a fix bandwidth in an Ethernet card)
- few ACKs to increase efficiency (not a problem in dedicated networks).

Protocols which are more aggressive than TCP such as XCP (eXplicit Control Protocol) for example can be used [40] as well as protocols using UDP for data transfer and TCP for control as seen in Section 2.

We assume that each link is bi-directional and symmetric (same performances in both ways) and the routing protocol used is also symmetric: the path used from a node A to a node B and the reversed path from the node B back to the node A are symmetric. This can be done by using the explicit routing functionality of MPLS [54] for example.

3.2.3 Router Model

For the sake of clarity, each network equipment, apart from the end-host ethernet cards and from the links, is called a router. Firewalls, transmitters, switches, etc. are considered as particular routers with different functionnality. This language simplification has no impact on the generality of the model. It is just to make the explanations more clear. So, the global network is formed by three categoriess of components: routers, links and end host. Each category of component has its own energy consumption function depending on its own parameters as stated in Section 3.1. Each router stores also its energy cost function depending on its own characteristics and on its utilization (so on its agendas).

As stated in previous subsections, each router needs to store an agenda for each of its ports. So, the routers require extended memory and computing capacities to maintain these agenda up-to-date. The reservations are thus stored in a decentralized manner in each concerned router.

Routers should also have the capacity to use energy-efficient techniques such as ALR (Adaptive Link Rate) [9] to adapt the transmission rate to the usage and thus, to decrease the energy consumption when not working at full capacity.

3.3 Problem Formulation

Up to now, we have proposed:

- an end-to-end energy cost model for general networks and
- a network model for bulk data transfers with advance bandwidth reservations.

For the sake of clarity, each network equipment, apart from the end-host ethernet cards and from the links, is called a router. Firewalls, transmitters, switches, etc. are considered as particular routers with different functionnality. This language simplification has no impact on the generality of the model. It is just to make the explanations more clear. So, the global network is formed by three categoriess of components: routers, links and end host. Each category of component has its own energy consumption function depending on its own parameters as stated in Section 3.1.

The general architecture of our network model as viewed by the end user is presented on Figure 5.

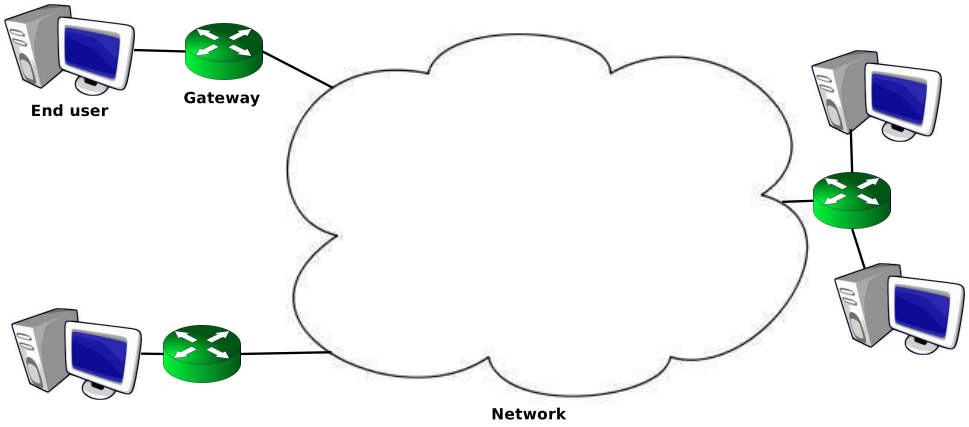

Figure 5: General architecture as seen per the end users

The scenario, we are considering here is the following: a large number of files have to be transferred from multiple senders to multiples receivers. Each transmission corresponds to a single file and a single pair of nodes (sender-receiver) which are called end users. Each end user is directly connected to a **gateway** (also called bandwidth broker in such networks [71, 13] when there is a centralized management). End users submit reservation requests to gateways in order to reserve bandwith for a given time to transfer a file to their receivers. Only the BDT service provider knows the network, its topology and its state (traffic, energy consumption, on or off).

In such a context, the objective is to find a good trade-off between performance (number of granted reservations) and energy consumption of the considered network.

4 In-Advance Reservation for Bulk Data Transfers

To achieve energy-efficiency, our network management infrastructure combines several techniques:

- unused network components are put in sleep mode;
- energy optimization of the reservation scheduling by reservation aggregation;
- minimization of the control messages required by the infrastructure;
- usage of DTN to manage the infrastructure;
- network usage prediction to avoid too frequent on/off cycles.

The following section will describe the usage of these techniques and how they are combined to form the global energy-efficient network management solution that we propose. In this section, we assume that the routing protocol gives only one path between each pair of network nodes (the routing is unique and symmetric).

4.1 Energy Optimization through Reservation Scheduling and Aggregation

Assuming that the complete energy cost functions are available for all the network nodes, the key problem is to use this information in order to schedule the bulk data transfers in an energy-efficient way.

On the basic scenario presented in Figure 2, the node A wants to send data to the node B. He submits a reservation with at least, a certain data volume V to be transfered and a deadline d. The reservation request is sent to a gateway (or bandwidth broker) as explained on Figure 5.

To take an energy-efficient placement decision, the scheduler should know the agenda of all the network equipments of the path (routers and ports) between A and B. The following subsections explain how this collect is done. Here, we are going to detail the scheduling algorithm.

After the agenda collect and fusion, the **availability agenda** of the path is obtained. It contains all the residual bandwidth of the path. It is stored as a normal agenda with a time-bandwidth list.

Then, the reservation is scheduled at the less energy consuming place. A **place** is a time period during which the reservation can be put without collapsing with others and without passing the deadline. An continuous time model is used, but all the seconds are not tested as possible reservation start times. Indeed, to be more energy-efficient, the reservation should be aggregated with other ones if possible. Thus, each event is tested to be a reservation start time or end time.

The energy consumption of each possible place is computed using ECOFEN; that is to say, using the energy cost function of each network component of the path (routers, ethernet cards, ports, etc.) as explained in Section 3.1. The energy consumption computation also includes the cost of new switching on and off of network resources (ports, routers, etc.) if these resources are required and were off. If putting the reservation at that place just delays the switching off or moves the switching on forward, this is not taken into account.

Algorithm 1 describes this scheduling process.

Algorithm 1 Scheduling algorithm

If the availability agenda of the path is empty **Then**
  Put the reservation in the middle of the remaining period before the deadline, if possible. Otherwise, put it now ($+\epsilon$ for the request processing time).
Else
  If there is no event before the deadline **Then**
    Put the reservation in the middle of the remaining period before the deadline if possible. Otherwise, put it as soon as possible.
  Else
    ForEach event in the availability agenda of the path and while it occurs before deadline **Do**
      Try to place the reservation after and before the event.
      Memorize the possible places (no collision with other reservations and end before deadline).
    If there is no possible place **Then**
      If the reservation can be put before the deadline **Then**
        Put the reservation now ($+\epsilon$ for the request processing time).
      Else
        If some events were not possible because of the deadline constraint **Then**
          If the reservation can be put now (some bandwidth is available) without respecting the deadline **Then**
            Propose this solution to the user.
          Else
            ForEach of these remaining events while no solution has been found **Do**
              Try to place the reservation after the event without respecting the deadline.
              Store the soonest possible place (no collision with other reservations) to propose it to the user.
    Else
      ForEach possible place **Do**
        Estimate the energy consumption of the transfer using the energy cost functions of each equipment.
      If there is one less energy consuming solution **Then**
        Take that place!
      Else
        Take the soonest place among the less energy consuming ones.

When the algorithm tries to place a reservation, it uses as much bandwidth as it can at each time to minimize the length of the reservation and thus, the energy consumed by this reservation.

In the worst case, this algorithm has a complexity in $O(n^2)$ with n the number of events in the availability agenda of the path.

On a given network equipment, a new reservation can take place at the same time that another one already scheduled if the network equipment has not reached its full bandwidth: Figure 6 shows an agenda with such a case. However, two reservations cannot overlap: the sum of their bandwidth can never exceed the bandwidth capacity of the network equipment.

For example, on Figure 7, no transmission from A to B can have a rate greater than 1 Gb/s as it is the minimum of the max available bandwidth on the path from A to B. Moreover, if a reservation

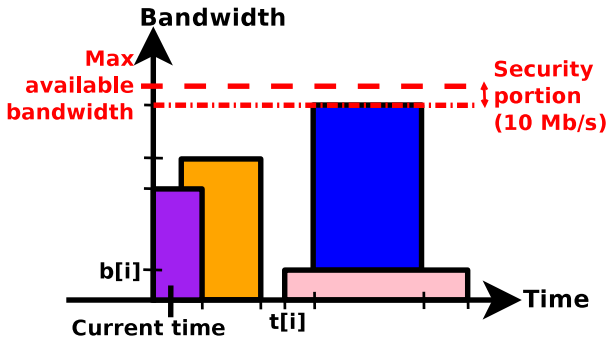

Figure 6: Agenda example

R using 9.5 Gb/s of bandwidth has already be done by other end users (not presented on the figure), a reservation from A to B can still use 500 Mb/s during the reservation R. In fact, it will use only 490 Mb/s because, we always keep a free bandwidth portion of 10 Mb/s on each link for management messages and for the ACKs. This portion is called the **security portion**.

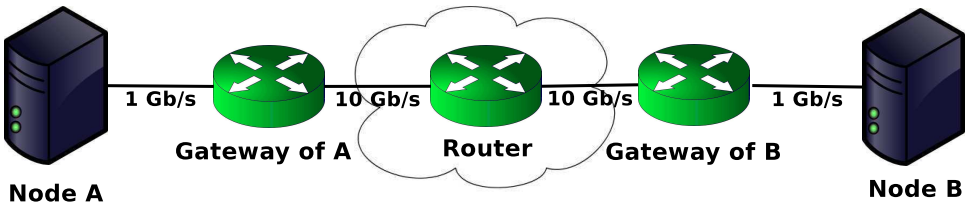

Figure 7: Simple network example

On Algorithm 1, if the reservation is in the case where *there is no possible place*, the algorithm puts it as soon as possible. Indeed, if the reservation can be scheduled before the deadline without being cut in several parts, then it can be put now (with a small delay due to request processing time). In fact, as no place has been found next to an already granted reservation, and as the reservation can be scheduled before the deadline, the bigest remaining free part of the bandwidth starts now.

If the reservation cannot been put before the deadline, the algorithm puts it as soon as possible. Indeed, in that case, it means that there are some places after events which were not considered because of the deadline constraint (these events can be before the deadline). The list of these remaining events is not empty because otherwise, it means that we are in the first case: the availability agenda path is empty and that case has already been treated. So, as this list of these remainings events is not empty, the earliest possible start time is now or one of these events because of the meaning of these events: bandwidth release or occupancy. However, the earliest possible start time is not the first bandwidth realising event since a reservation put after can take all the bandwidth.

The reservation uses at any time as much bandwidth as they can without collapsing with other reservations and exceeding the max available bandwidth l minus the security portion. Yet, the reservations might use different paths (if they have different destinations), so they have different max available

bandwidth, and thus they have different and even varying transmission rates.

4.2 Request Processing and Agenda Collect: Usage of DTN

When a gateway receives a reservation request, the first operation to execute is admission control. The validity of the request is checked. Then, each request requires to collect the agendas of all the equipments (ports and routers) along the network path between the source and the destination.

In fact, the agenda of a link is stored twice: once in each port linked by this link. This mechanism allows the two concerned routers to have the right energy cost function without having to ask to its neighbors. Indeed, both sending and receiving data is energy consuming, thus both should be taken into account to compute the energy consumption.

In order to do this agenda collect, all the agendas of the path will be sent to the gateway of the receiver. The sender gateway will send a particular management message along the path containing the required agendas it owns (agenda of the receiving port linked to the sender and agenda of the transmitting port linked to the next hop on the path). Then, each router of the path will add to this message its own agenda and the agenda of the transmitting port linked to the next hop. Each router sends this growing message to the next hop until it reaches the receiver gateway. The receiver gateway adds the agenda of its transmitting port linked to the receiver and ask the receiver its own agenda (containing on, off and transition periods beteen on and off). In the same way, this message also contains all the energy cost functions of crossed network equipments (for each port and router).

Thus, the receiver gateway ends up with all the the required agendas. The availability agenda of the path is not computed on the fly by each router because the agendas are required to compute the energy consumptions (the energy cost functions depend on the network equipment usage).

However, this process works only if all the ports and routers are on when the agenda collect is done. Indeed, when they are not used, the network equipments (individual ports or entire routers) are put into sleep mode. To solve this issue, DTN (Disruption-Tolerant Networking) [24] technologies are used. Indeed, DTN are perfectly fitted for this type of scenario where parts of the network are not always available without any guaranty of end-to-end connectivity at any time.

The idea is to add a kind of TTL (time-to-live) in seconds to each end-user request: when the TTL expired, if the request has not reached the receiver gateway and is not come back, then all the sleeping nodes of the path are awaken and the agenda collect is performed. While the TTL is not expired, the agenda collect message moves forward along the path until meeting a sleeping node. Then, as long as the TTL is not expired, the message waits in the previous node for the sleeping node to wake up, and when it wakes up, the message is sent to it and continues its way. Thus, hop by hop, the agenda collect message moves towards the receiver gateway.

However, using DTN requires to be able to wake up a node from a neighboring one. Several techniques

can be used to solve this issue :

1) using techniques inspired from sensor networks: each node is periodically awaken. The problem is to find a period which satisfies both the energy and the performance issues. This is not an on-demand waking service.

2) using wireless network cards with satellite connections on each node. This card remains always on (or is awaken frequently) and can awake the whole node at the receipt of a special packet. This solution requires an operational wireless network.

3) using a low power dedicated network (like a control network) which is a spanning tree of the considered dedicated network. Each node of the actual network keeps a special low-power interface powered on and this interface is able to wake up the whole node and is linked to the control network. This solution involves that all the nodes are duplicated but not all the links. This solution is similar to the previous one, but with a wired control network instead of a wireless one.

All of these techniques have an impact on the energy consumption and must be studied over different scenarios to determine the best one. The TTL is determined by the user. A TTL equal to 0 means that all the path should be awaken, if it is not already the case, and that the request should be processed as soon as possible in order to answer quickly.

An option frequently taken in the litterature [62, 61, 7, 18] is to always keep a path between any two nodes and just switch off some links. However, these solutions avoid the problem: all the nodes remain always on even if idle, and it does not provide a mechanism to wake up a node. For the papers raising this issue, the main used technique is the periodical wake-up [46, 12, 8]. A wake-on-arrival technique is described in [29]: routers are ware up automatically when detecting an incoming traffic on their ports. But, although it is a highly desirable technique to save energy, it is not yet a realistic solution since interfaces and routers today does not provide this kind of hardware support. In fact, to detect a traffic on all the ports, it requires to have an always on component on each port.

After a wake-up, a synchronization mechanism is required for the router to be fully working again [67].

4.3 Agenda fusion

When all the agendas have been collected by the receiver gateway, they should be merged to make the path availability agenda. This agenda fusion process (Algorithm 2) is described on Figure 8.

Algorithm 2 Agenda fusion

Sort in increasing order the list t of all the events from all the agenda.
ForEach event $t[i]$ in this ordered list **Do**
  ForEach agenda **Do**
    Compute the residual bandwidth at $t[i]$.
  Take the minimum of the residual bandwidths. It's $b[i]$.

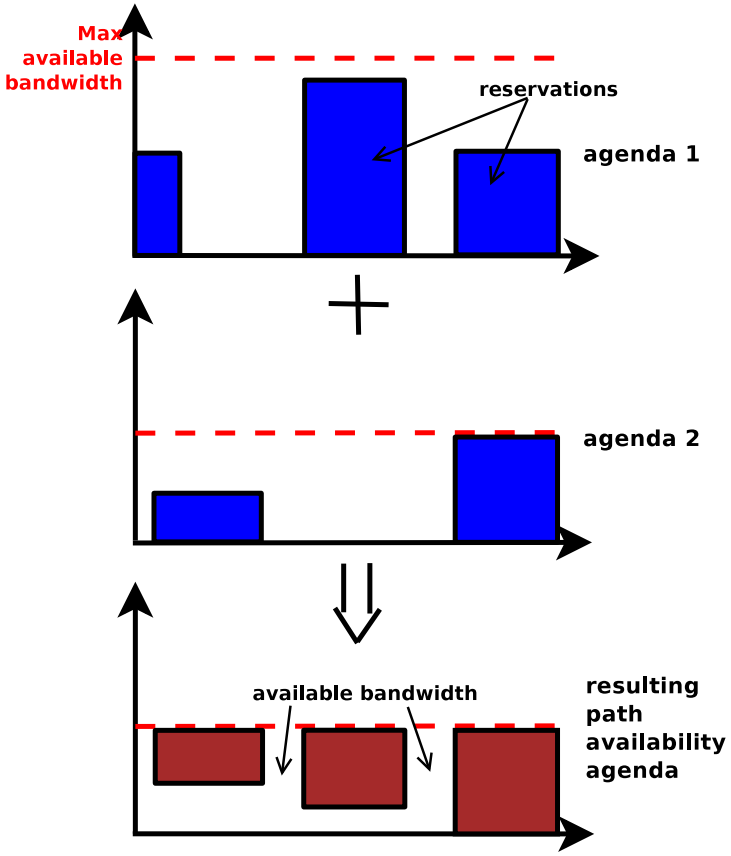

Figure 8: Agenda fusion

This algorithm complexity is $O(n \ \log n + n \times m)$ where n is the total number of event (length of the list t) and m is the number of merged agendas. Indeed, the sorting complexity is $O(n \ \log n)$ and then, there is a for loop of size m in a for loop of size n.

4.4 Reservation granting

The receiver gateway has been chosen to schedule the reservation request because in such a way, only one end-to-end message round-trip is required to grant a reservation. Indeed, the first end-to-end message aggregates all the agendas and energy cost functions from the sender to the receiver. Then, the agendas are merged as explained in the next subsection, and the reservation is scheduled as explained in the previous subsection. Next, an end-to-end message is sent from the BDT receiver to the BDT sender with the reservation scheduling in order to update all the agendas of the path if the reservation can be granted. The sender gateway is in charge of notifying the sender of the acceptance and scheduling of its reservation if the request has been accepted. Otherwise, the sender gateway proposes another solution to the end-user with a less restrictive deadline. If the user accepts this solution, the sender gateway sends the message to update wll the agendas of the path.

To sum up, to grant a reservation, the network management infrastructure works as follows:

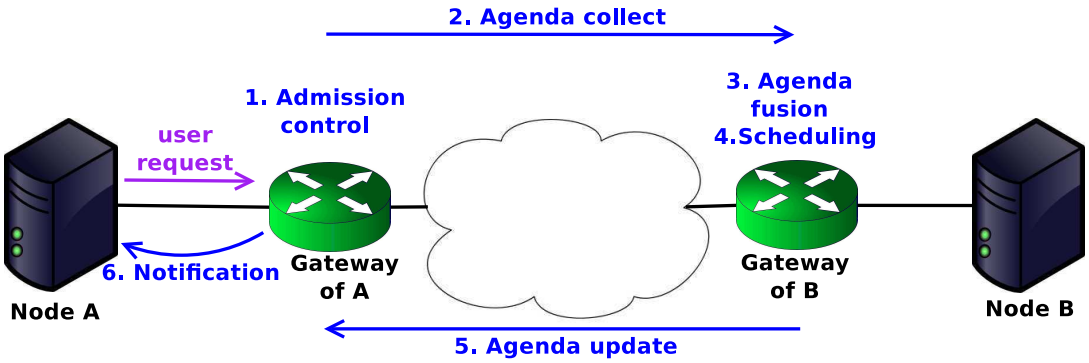

Figure 9: Steps to grant a reservation

- First step: *admission control*. The gateway checks the request validity.
- Second step: *agenda collect*. The receiver node's gateway collects all the agendas of the path.
- Third step: *agenda fusion*. The receiver node's gateway makes the available agenda of the path.
- Fourth step: *reservation scheduling*. The reservation is scheduled in an energy-efficient way. If the transfer is not possible before the deadline, the soonest possible deadline is computed.
- Fifth step: *agenda update*. This transfer is updated in all the concerned agendas of the path (propagation).
- Sixth step: *notification*. The user is informed of the reservation scheduling if the request was possible or an other solution with a different deadline is proposed to the user

These steps are synthesized in Figure 9. However, a locking concept is required to insure that different reservation requests will not interfere with each other.

The sender gatway is also in charge of controlling that the sender respects the network configuration of its reservation (bandwidth, destination, duration). It checks that the SLA is well respected by both parts.

4.5 Prediction Models and management of sleeping network equipments

As outlined in [69], rate switching consumes time and energy. As in our framework, each equipment always know its utilisation, it can precisely anticipate and adjust its transmission rate to fit the demand without packet losses.The ALR buffer threshold policy described in [27] is useless here.

Switching on and off network equipments consume as well energy and time [31]. Our framework aims to switch off unused resources to save energy. So, these switching stages should be accurately studied to determine whether it is more energy efficient to switch off and switch on again or to let the resource idle for a given period of inactivity.

The simpliest distributed on/off algorithm is to switch off as soon as there is nothing to do. Yet, it might be not the most energy efficient solution. Figure 10 presents the two possible cases for a given period of inactivity:

1. the upper graph presents the case where the resource is switched off, stays off for a while and then is switched on again.
2. the lower graph shows the case where the resource stays idle for the whole time.

The two colored areas represent the energy consumption for the two cases. We define T_s the switching threshold such as these two consumptions are equal for a given resource. This means that if the period of inactivity is greater than T_s, the most energy efficient scenario is to switch off the resource and to switch it on again at the end. Otherwise, it is more energy efficient to let the resource idle.

Thus, for a given network resource, the formal definition of T_s is as follows:

$$T_s = \frac{E_{ON \to OFF} + E_{OFF \to ON} - P_{OFF}(\delta_{ON \to OFF} + \delta_{OFF \to ON})}{P_{idle} - P_{OFF}} \tag{6}$$

where P_{idle} is the idle consumption of the resource (in Watts), P_{OFF} the power consumption when the resource is off (in Watts), $\delta_{ON \to OFF}$ the duration of the resource shutdown (in seconds), $\delta_{OFF \to ON}$ the duration of the resource boot, $E_{ON \to OFF}$ the energy consumed to switch off the resource (in Joules) and $E_{OFF \to ON}$ the energy consumed to switch on the resource (in Joules).

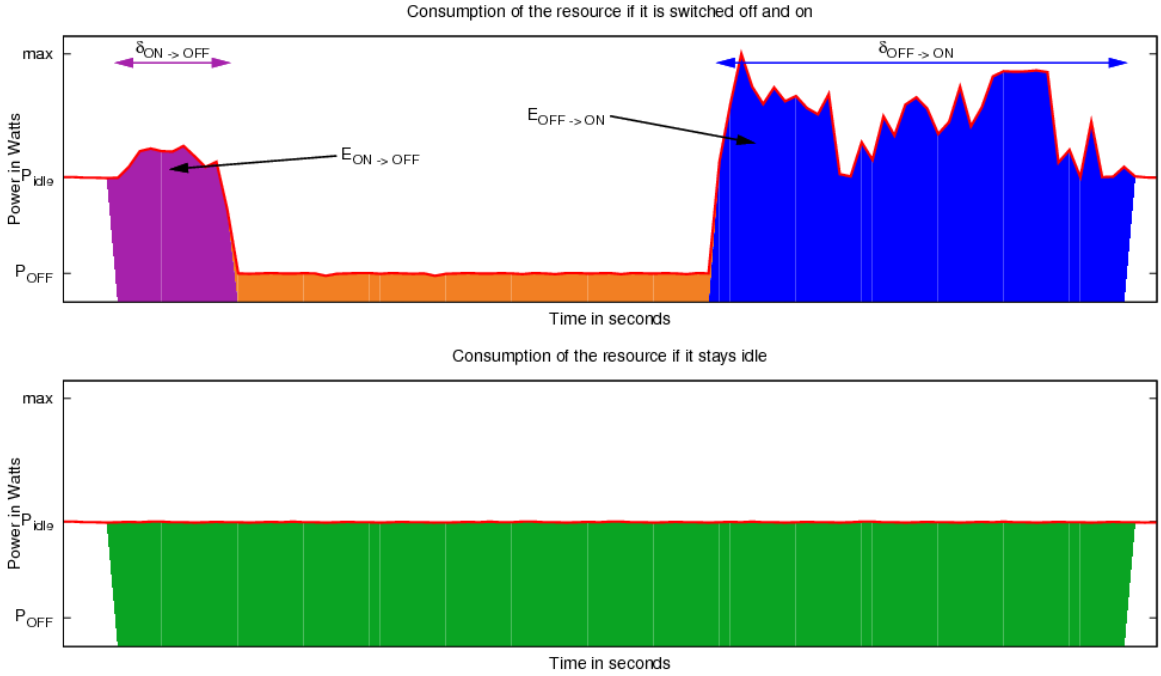

Figure 10: T_s

Then, the key idea is that at the end of a transfer between two nodes, if one port is idle for more than T_s seconds, the port is switched off and if all the port of a router are switched off, then the router itself

is switched off. In addition, to avoid unnecessary on/off cycles, prediction algorithms are used to predict the next utilization of a link. So, the algorithms at the end of a reservation is presented in Algorithm 3.

Algorithm 3 At the end of a reservation

```
ForEach port used by this transfer Do
  If there is a reservation in the port's agenda starting in less than T_s seconds Then
    Let the port powered on (at lower transmitting rate).
  Else
    Predict the next utilization of this port.
    If the predicted usage is in less than T_s seconds Then
      Let the port powered on (at lower transmitting rate).
    Else
      Switch off the port (sleeping mode).
      If this port was the last powered on port Then
        Switch off the router (sleeping mode).
```

This algorithm is in fact distributed and executed at the end of a transfer by each port independently of one another. The prediction algorithms rely on recent history (past agenda) of the port. They are based on average values of past inactivity period durations and feedbacks which are average values of differences between past predictions and the past corresponding events in the agenda.

4.6 Adaptivity

Up to now, we have presented algorithms which work on a static way: once a decision has been taken, it cannot be changed. Our adaptivity functionality is dynamic and allows reservations to be moved after their registration in the path agendas while guaranteeing the same QoS and respecting the user deadline. Indeed, off-line algorithms can lead to optimal solutions in terms of energy conservation because all the reservation requests are known since the beginning. Yet, it is not the case with on-line algorithms such as our.

An example of the working of this adaptivity functionality is presented on Figure 11. The reservation $R4$ has just been inscribed into the two agendas and thus, the reservation $R3$ can be put just after if it is more energy efficient than the former solution.

This step is realized by the receiver's gateway after sending the request notification to the sender's gateway and during the agenda update of all the agendas of the path. Only the reservations between the same two nodes are considered since receiver's gateway has no access to the agendas for another couple of sender-receiver.

4.7 Discussion

The proposed network management optimizes the energy consumption of the overall architecture at any time. However, we have not yet studied the energy optimization of transfers themselves.

Indeed, we have assumed that at any time, the most energy efficient behavior is to use as much bandwidth as possible (from source to destination). Yet, we have not proved that this algorithm leads

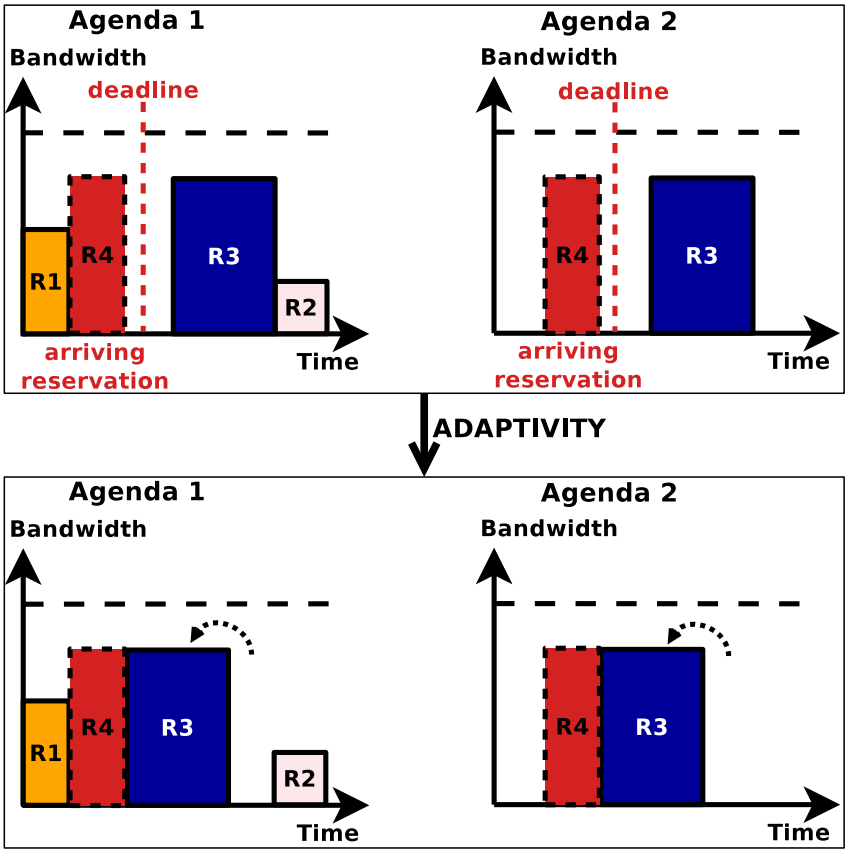

Figure 11: Adaptivity

to the minimum energy consumption.

Lets take an example: node A wants to send 2 GB of data to node B and node A and B are directly linked by a 1 GB/s link. Our algorithm will schedule the transfer and set the bandwidth at 1 GB/s (minus the security portion). If we assume that the security portion is negligible, it takes 0.2 seconds to transmit 2 GB of data at 1 GB/s. Thus, this transfer will consume:

$$E_{transfer} = E_{EthernetCard}(NodeA, 1GB/s, 0.2s) + E_{EthernetCard}(NodeB, 1GB/s, 0.2s)$$

If we denote $P_{EthernetCard}(NodeA, 1GB/s)$ the power consumed by node A when it transmits data at 100 MB/s, we have:

$$E_{transfer} = P_{EthernetCard}(NodeA, 1GB/s) \times 0.2 + \quad P_{EthernetCard}(NodeB, 1GB/s) \times 0.2$$

However, another solution could be to adjust the Ethernet card to work at 100 MB/s and thus, it does not use the full capacity and it takes more time. In that case, the transfer consumes:

$$\begin{aligned} E'_{transfer} &= E_{EthernetCard}(NodeA, 100MB/s, 2s) + E_{EthernetCard}(NodeB, 100MB/s, 2s) \\ &= P_{EthernetCard}(NodeA, 100MB/s) \times 2 + P_{EthernetCard}(NodeB, 100MB/s) \times 2 \end{aligned}$$

If we assume that the NICs are identical and thus have the same power consumption $P_{EthernetCard}(100MB/s)$ and $P_{EthernetCard}(1GB/s)$ depending on the rate. Then the second solution uses less energy to transfer the data if and only if:

$$P_{EthernetCard}(1GB/s) > 10 \times P_{EthernetCard}(100MB/s)$$

If we use the figures provided in [69] for a NIC, we have $P_{EthernetCard}(100MB/s) = 0.4$ Watts, and $P_{EthernetCard}(1GB/s) = 3.6$ Watts. In that case, our scenario is the most energy efficient with a consumption equal to 0.72 Joules (and 0.8 Joules for the second scenario). However, here, we only considered the energy used to transfer data and not the overall energy of the infrastructure during a certain period of time. Thus, these two energy consumption do not represent the same period of time (0.2 and 2 seconds). To compare them over an identical time period, we should add to $E_{transfer}$ the cost of staying off during 1.8 seconds.

We have not taken into account the energy required to switch on the NICs at the beginning and to switch them off at the end of the transfer since these energy costs are identical in both scenarios.

This remark shows that our algorithm should be compared with other solutions and that the optimal solution is hard to find even in scenarios with fixed routing. This situation is the result of the non-proportionnality between energy and usage: cost functions are linear by steps and not just linear.

5 Conclusion and Future Works

Scientists increasingly rely on the network for high-speed data transfers, result disseminations and collaborations. Networks are thus becoming the critical component. In 2007, to distribute the entire collection of Hubble telescope data (about 120 terabytes) to various research institutions, scientists chose to copy these data on hard disks and to send these hard disks via mail. It was faster than using the network [23]. To solve this issue, dedicated networks are built to transfer large amount of scientific data, like for example for the LHC (Large Hadron Collider) which produces 15 million gigabytes of data every year [1].

Bandwidth provisioning has been made feasible for network operators since several years thanks to protocols such as MultiProtocol Label Switching (MPLS) [54] and Reservation Protocol (RSVP) [70]. However, for end users with no network traffic knowledge, this task is impossible without collaboration with the other nodes.

On the other hand, as networks become increasingly essential, their electric consumption reaches unattended peaks [4]. Up to now, the main concern to design network equipments and protocols was only performance and thus, energy consumption was not taken into account. With the not affordable growth of network electricity demand, it is high time to consider energy as a main priority for network design.

The first step to obtain green wired networks is to understand how the energy is consumed in the network by separating the consumption of each network component and by analyzing the link between energy consumption and usage. This essential analysis is the basis to develop energy-aware framework for network management.

Following these steps, we have proposed:

- an end-to-end energy cost model that considers the topology and the traffic to estimate the energy consumed by every networks;
- a network model which is adapted to Advance Bandwidth Reservations (ABR) for Bulk Data Transfer (BDT).
- a new complete and energy-efficient BDT framework including scheduling algorithms which provide an adaptive and predictive management of the ABR.

Our future works will be focusing on the validation of this framework and its adaptation to other sdenarios including not-static routing algorithms to take advantage from multipath routing to consume less energy.

References

[1] http://lcg.web.cern.ch/lcg/public/default.htm.

[2] H. Anand, C. Reardon, R. Subramaniyan, and A. George. Ethernet Adaptive Link Rate (ALR): Analysis of a MAC Handshake Protocol. In *31st IEEE Conference on Local Computer Networks*, pages 533–534, Nov. 2006.

[3] G. Ananthanarayanan and R. Katz. Greening The Switch. Technical report, September 2008.

[4] M. Baldi and Y. Ofek. Time For A "Greener" Internet. Dresden, Germany, June 2009. Best Paper Award.

[5] J. Baliga, K. Hinton, and R. Tucker. Energy Consumption Of The Internet. In *COIN-ACOFT 2007: Joint International Conference On Optical Internet, 2007 And The 2007 32nd Australian Conference On Optical Fibre Technology.*, pages 1–3, Melbourne, Australia, June 2007.

[6] L.A. Barroso and U. Holzle. The Case for Energy-Proportional Computing. *Computer*, 40(12):33 –37, Dec. 2007.

[7] B. Bathula and J. Elmirghani. Energy Efficient Optical Burst Switched (OBS) Networks. In *GreenComm: 2nd International Workshop On Green Communications (IEEE GLOBECOM Workshop)*, pages 1–6, Dec. 2009.

[8] B. Bathula and J. Elmirghani. Green networks: Energy efficient design for optical networks. In *IFIP International Conference on Wireless and Optical Communications Networks (WOCN '09)*, pages 1 –5, 28–30 2009.

[9] M. Bennett, K. Christensen, and B. Nordman. Improving The Energy Efficiency Of Ethernet: Adaptative Link Rate Proposal, July 2006.

[10] J. Blackburn and K. Christensen. A Simulation Study Of A New Green BitTorrent. Dresden, Germany, June 2009.

[11] F. Blanquicet and K. Christensen. An Initial Performance Evaluation of Rapid PHY Selection (RPS) for Energy Efficient Ethernet. In *32nd IEEE Conference on Local Computer Networks (LCN 2007)*, pages 223 –225, 2007.

[12] F. Blanquicet and K. Christensen. PAUSE Power Cycle: A New Backwards Compatible Method To Reduce Energy Use Of Ethernet Switches. Technical report, April 2008.

[13] L.-O. Burchard. Networks with Advance Reservations: Applications, Architecture, and Performance . *Journal of Network and Systems Management*, 13(4):429–449, 2005.

[14] L.-O. Burchard and M. Droste-Franke. Fault Tolerance in Networks with an Advance Reservation Service. In *11th International Workshop on Quality of Service (IWQoS 2003)*, pages 215–228, 2003.

[15] L.-O. Burchard, B. Linnert, and J. Schneider. Rerouting strategies for networks with advance reservations. In *First International IEEE Conference on e-Science and Grid Computing (E-Science 2005)*, July 2005.

[16] J.W. Byers, J. Considine, M. Mitzenmacher, and S. Rost. Informed content delivery across adaptive overlay networks. *IEEE/ACM Transactions on Networking*, 12(5):767 – 780, oct. 2004.

[17] J. Chabarek, J. Sommers, P. Barford, C. Estan, D. Tsiang, and S. Wright. Power Awareness In Network Design And Routing. In *INFOCOM 2008. The 27th Conference On Computer Communications. IEEE*, pages 457–465, April 2008.

[18] L. Chiaraviglio, D. Ciullo, E. Leonardi, and M. Mellia. How Much Can The Internet Be Greened? In *GreenComm: 2nd International Workshop On Green Communications (IEEE GLOBECOM Workshop)*, pages 1–6. IEEE Computer Society, 2009.

[19] L. Chiaraviglio, M. Mellia, and F. Ner. Energy-Aware Backbone Networks: A Case Study. Dresden, Germany, June 2009.

[20] L. Chiaraviglio, M. Mellia, and F. Neri. Energy-Aware Networks: Reducing Power Consumption By Switching Off Network Elements. Technical report, FEDERICA-Phosphorus tutorial and workshop (TNC2008), 2008.

[21] K. Christensen, C. Gunaratne, B. Nordman, and A. George. The next frontier for communications networks: power management. *Computer Communications*, 27(18):1758–1770, 2004.

[22] B. Eckart, X. He, and Q. Wu. Performance Adaptive UDP For High-Speed Bulk Data Transfer Over Dedicated Links. pages 1–10, april 2008.

[23] C. Farivar. Google's Next-Gen of Sneakernet. [online] http://www.wired.com/science/discoveries/news/2007/03/73007, 2007.

[24] S. Farrell, V. Cahill, D. Geraghty, I. Humphreys, and P. McDonald. When TCP Breaks: Delay- and Disruption- Tolerant Networking. *IEEE Internet Computing*, 10(4):72–78, Aug. 2006.

[25] Y. Gu and R. Grossman. SABUL: A Transport Protocol for Grid Computing. *Journal of Grid Computing*, 1(4):377–386, 2003.

[26] C. Gunaratne, K. Christensen, and B. Nordman. Managing Energy Consumption Costs In Desktop PCs And LAN Switches With Proxying, Split TCP Connections, And Scaling Of Link Speed. *International Journal Of Network Management*, 15(5):297–310, 2005.

[27] C. Gunaratne, K. Christensen, and S. Suen. Ethernet Adaptative Link Rate (ALR): Analysis Of A Buffer Threshold Policy. In *GLOBECOM'06: IEEE Global Telecommunications Conference*, pages 1–6, San Francisco, USA, November 2006.

[28] C. Guok, J. Lee, and K. Berket. Improving the bulk data transfer experience. *International Journal of Internet Protocol Technology*, 3(1):46–53, 2008.

[29] M. Gupta and S. Singh. Greening Of The Internet. In *SIGCOMM '03: Proceedings Of The 2003 Conference On Applications, Technologies, Architectures, And Protocols For Computer Communications*, pages 19–26, Karlsruhe, Germany, August 2003. ACM.

[30] M. Gupta and S. Singh. Dynamic Ethernet Link Shutdown For Energy Conservation On Ethernet Links. In *Communications, 2007. ICC '07. IEEE International Conference On*, pages 6156–6161, Glasgow, Scotland, June 2007.

[31] M. Gupta and S. Singh. Using Low-Power Modes For Energy Conservation In Ethernet LANs. In *INFOCOM 2007. 26th IEEE International Conference On Computer Communications. IEEE*, pages 2451–2455, Anchorage, Alaska, USA, May 2007.

[32] S. Harizopoulos, M. Shah, J. Meza, and P. Ranganathan. Energy Efficiency: The New Holy Grail of Data Management Systems Research. In *Fourth Biennial Conference on Innovative Data Systems Research (CIDR)*, Jan. 2009.

[33] E. He, J. Leigh, O. Yu, and T. Defanti. Reliable Blast UDP : Predictable High Performance Bulk Data Transfer. pages 317 – 324, 2002.

[34] M. Hefeeda, A. Habib, D. Xu, B. Bhargava, and B. Botev. CollectCast: A peer-to-peer service for media streaming. *Multimedia Systems*, 11(1):68–81, 2005.

[35] H. Hlavacs, G. Da Costa, and J.-M. Pierson. Energy Consumption of Residential and Professional Switches. In *International Conference on Computational Science and Engineering (CSE '09)*, volume 1, pages 240 –246, Aug. 2009.

[36] T.-C. Hou and V. Li. Transmission range control in multihop packet radio networks. *IEEE Transactions on Communications*, 34(1):38 – 44, jan 1986.

[37] L. Irish and K. Christensen. A "Green TCP/IP" to Reduce Electricity Consumed by Computers. In *Proceedings of IEEE Southeastcon*, pages 302–305, 1998.

[38] V. Jacobson, R. Braden, and D. Borman. TCP Extensions for High Performance. RFC 1323, 1992.

[39] E.-S. Jung, Y. Li, S. Ranka, and S. Sahni. An Evaluation of In-Advance Bandwidth Scheduling Algorithms for Connection-Oriented Networks. In *International Symposium on Parallel Architectures, Algorithms, and Networks (I-SPAN 2008)*, pages 133–138, May 2008.

[40] D. Katabi, M. Handley, and C. Rohrs. Congestion control for high bandwidth-delay product networks. In *SIGCOMM '02: Proceedings of the 2002 conference on Applications, technologies, architectures, and protocols for computer communications*, pages 89–102, New York, NY, USA, 2002. ACM.

[41] N. Laoutaris, G. Smaragdakis, P. Rodriguez, and R. Sundaram. Delay tolerant bulk data transfers on the Internet. In *SIGMETRICS '09: Proceedings of the eleventh international joint conference on Measurement and modeling of computer systems*, pages 229–238, New York, NY, USA, 2009. ACM.

[42] Y. Li, S. Ranka, and S. Sahni. In-advance path reservation for file transfers In e-Science applications. In *IEEE Symposium on Computers and Communications (ISCC 2009)*, pages 176 –181, 5-8 2009.

[43] Y. Lin and Q. Wu. On Design Of Bandwidth Scheduling Algorithms For Multiple Data Transfers In Dedicated Networks. In *ANCS '08: Proceedings Of The 4th ACM/IEEE Symposium On Architectures For Networking And Communications Systems*, pages 151–160, New York, NY, USA, 2008. ACM.

[44] Y. Lin and Q. Wu. Path Computation With Variable Bandwidth For Bulk Data Transfer In High-Performance Networks. In *High-Speed Networks Workshop (HSN 2009)*. IEEE, 2009.

[45] Y. Lin, Q. Wu, N. Rao, and M. Zhu. On Design Of Scheduling Algorithms For Advance Bandwidth Reservation In Dedicated Networks. pages 1 –6, april 2008.

[46] S. Nedevschi, L. Popa, G. Iannaccone, S. Ratnasamy, and D. Wetherall. Reducing Network Energy Consumption Via Sleeping And Rate-Adaptation. pages 323–336, San Francisco, USA, April 2008.

[47] B. Nordman and K. Christensen. Proxying: The Next Step In Reducing IT Energy Use. *Computer*, 43(1):91–93, 2010.

[48] A. Odlyzko. Data Networks are Lightly Utilized, and will Stay that Way. *Review of Network Economics*, 2, 2003.

[49] A. Patel, Y. Zhu, Q. She, and J. Jue. Routing and Scheduling for Time-Shift Advance Reservation. In *18th Internatonal Conference on Computer Communications and Networks (ICCCN 2009)*, pages 1–6, Aug. 2009.

[50] A. Qureshi, R. Weber, H. Balakrishnan, J. Guttag, and B. Maggs. Cutting The Electric Bill For Internet-Scale Systems. volume 39, pages 123–134, Barcelona, Spain, August 2009.

[51] K. Rajah, S. Ranka, and Ye Xia. Advance Reservations and Scheduling for Bulk Transfers in Research Networks. *IEEE Transactions on Parallel and Distributed Systems*, 20(11):1682–1697, Nov. 2009.

[52] W. Reinhardt. Advance Reservation of Network Resources for Multimedia Applications. In *Second International Workshop on Multimedia (IWAKA 1994)*, pages 23–33, London, UK, 1994. Springer-Verlag.

[53] J. Restrepo, C. Gruber, and C. Machuca. Energy Profile Aware Routing. In *IEEE International Conference on Communications (ICC Workshops 2009)*, pages 1–5, June 2009.

[54] E. Rosen, A. Viswanathan, and R. Callon. Multiprotocol Label Switching Architecture. RFC 3031, 2001.

[55] S. Sahni, N. Rao, S. Ranka, Y. Li, E.-S. Jung, and N. Kamath. Bandwidth Scheduling and Path Computation Algorithms for Connection-Oriented Networks. In *Sixth International Conference on Networking (ICN 2007)*, April 2007.

[56] A. Schill, S. Kühn, and F. Breiter. Design and evaluation of an advance reservation protocol on top of RSVP. In *Fourth International Conference on Broadband Communications (BC '98)*, pages 23–40, London, UK, UK, 1998. Chapman & Hall, Ltd.

[57] L. Shang, L.-S. Peh, and N. Jha. Dynamic Voltage Scaling with Links for Power Optimization of Interconnection Networks. In *9th International Symposium on High-Performance Computer Architecture (HPCA03)*, page 91, Washington, DC, USA, 2003. IEEE Computer Society.

[58] L. Shang, L.-S. Peh, and N. Jha. PowerHerd: a distributed scheme for dynamically satisfying peak-power constraints in interconnection networks. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 25(1):92–110, Jan. 2006.

[59] G. Shen and R. Tucker. Energy-Minimized Design for IP Over WDM Networks. *Journal of Optical Communications and Networking*, 1(1):176–186, 2009.

[60] R. Sherwood, R. Braud, and B. Bhattacharjee. Slurpie: a cooperative bulk data transfer protocol. In *INFOCOM 2004: twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, volume 2, pages 941 – 951, 7-11 2004.

[61] A. Silvestri, A. Valenti, S. Pompei, F. Matera, A. Cianfrani, and A. Coiro. Energy saving in optical transport networks exploiting transmission properties and wavelength path optimization. *Optical Switching and Networking*, 2010.

[62] V. Soteriou and L.-S. Peh. Dynamic Power Management For Power Optimization Of Interconnection Networks Using On/Off Links. pages 15–20, August 2003.

[63] I. Stojmenovic and X. Lin. Power-aware localized routing in wireless networks. *IEEE Transactions on Parallel and Distributed Systems*, 12(11):1122–1133, Nov. 2001.

[64] R.. Tucker. Green Optical Communications - Part I: Energy Limitations in Transport. *IEEE Journal of Selected Topics in Quantum Electronics, Special Issue on Green Photonics*, 2010.

[65] R. Tucker. Green Optical Communications - Part II: Energy Limitations in Networks. *IEEE Journal of Selected Topics in Quantum Electronics, Special Issue on Green Photonics*, 2010.

[66] H.-S. Wang, L.-S. Peh, and S. Malik. A Power Model for Routers: Modeling Alpha 21364 and InfiniBand Routers. *Symposium on High-Performance Interconnects*, 0:21, 2002.

[67] S.-W. Wong, L. Valcarenghi, S.-H. Yen, D. Campelo, S. Yamashita, and L. Kazovsky. Sleep Mode for Energy Saving PONs: Advantages and Drawbacks. In *GreenComm: 2nd International Workshop On Green Communications (IEEE GLOBECOM Workshop)*, pages 1–6, 30 2009-dec. 4 2009.

[68] C. Xie, F. Xu, N. Ghani, E. Chaniotakis, C. Guok, and T. Lehman. Load-Balancing for Advance Reservation Connection Rerouting. *IEEE Communications Letters*, 14(6):1–3, june 2010.

[69] B. Zhang, K. Sabhanatarajan, A. Gordon-Ross, and A. George. Real-Time Performance Analysis Of Adaptive Link Rate. pages 282–288, Montreal, Canada, October 2008.

[70] L. Zhang, S. Deering, D. Estrin, S. Shenker, and D. Zappala. RSVP: A New Resource ReSerVation Protocol. *IEEE Network*, 7:8–18, 1993.

[71] Z.-L. Zhang, Z. Duan, and Y. Hou. On Scalable Design of Bandwidth Brokers. *IEICE transactions on communications*, 8(E84-B):2011–2025, 2001.

[72] X. Zheng, A. Mudambi, and M. Veeraraghavan. FRTP: Fixed Rate Transport Protocol - A modified version of SABUL for end-to-end circuits. In *First Workshop on Provisioning and Transport for Hybrid Networks (PATHNets, BroadNets workshop)*, Oct. 2004.