What keeps your network up at night?

A link sleeping study of Switch LAN

Romain Jacob
ETH Zürich

Switch Network WG

June 27, 2024

What do you think consumes more energy?

Data Centers

Telco Networks

What do you think consumes more energy?

	Data Centers	or	Telco Networks	
In 2022	240-340 TWh		260-360 TWh	

https://www.iea.org/energy-system/buildings/data-centres-and-data-transmission-networks

What do you think consumes more energy?

Data Centers or **Telco Networks**

	Data Centers		Telco Networks	
In 2022	240-340	TWh	260-360	TWh
In 2015	200	TWh	220	TWh
Change of	+20-70%	in energy	+18-64%	in energy

https://www.iea.org/energy-system/buildings/data-centres-and-data-transmission-networks

What do you think consumes more energy?

Data Centers or Telco Networks

	Data Centers		Telco Networks	
In 2022	240-340	TWh	260-360	TWh
In 2015	200	TWh	220	TWh
Change of	+20-70%	in energy	+18-64%	in energy
	+340%	in workload	+600%	in traffic

https://www.iea.org/energy-system/buildings/data-centres-and-data-transmission-networks

Energy efficiency improved a lot

	Data Centers		Telco Networks	
Change in energy is much smaller	+20-70%	in energy	+18-64%	in energy
than in work done.	+340%	in workload	+600%	in traffic

Energy efficiency improved a lot but not enough!

Change in energy is positive!	Data Centers	Telco Networks
	+20-70% in energy	+18-64% in energy

"With great power comes great responsibility"

- It is easy to keep increasing network capacity
- It is much harder to keep increasing energy efficiency

“With great power comes great responsibility”

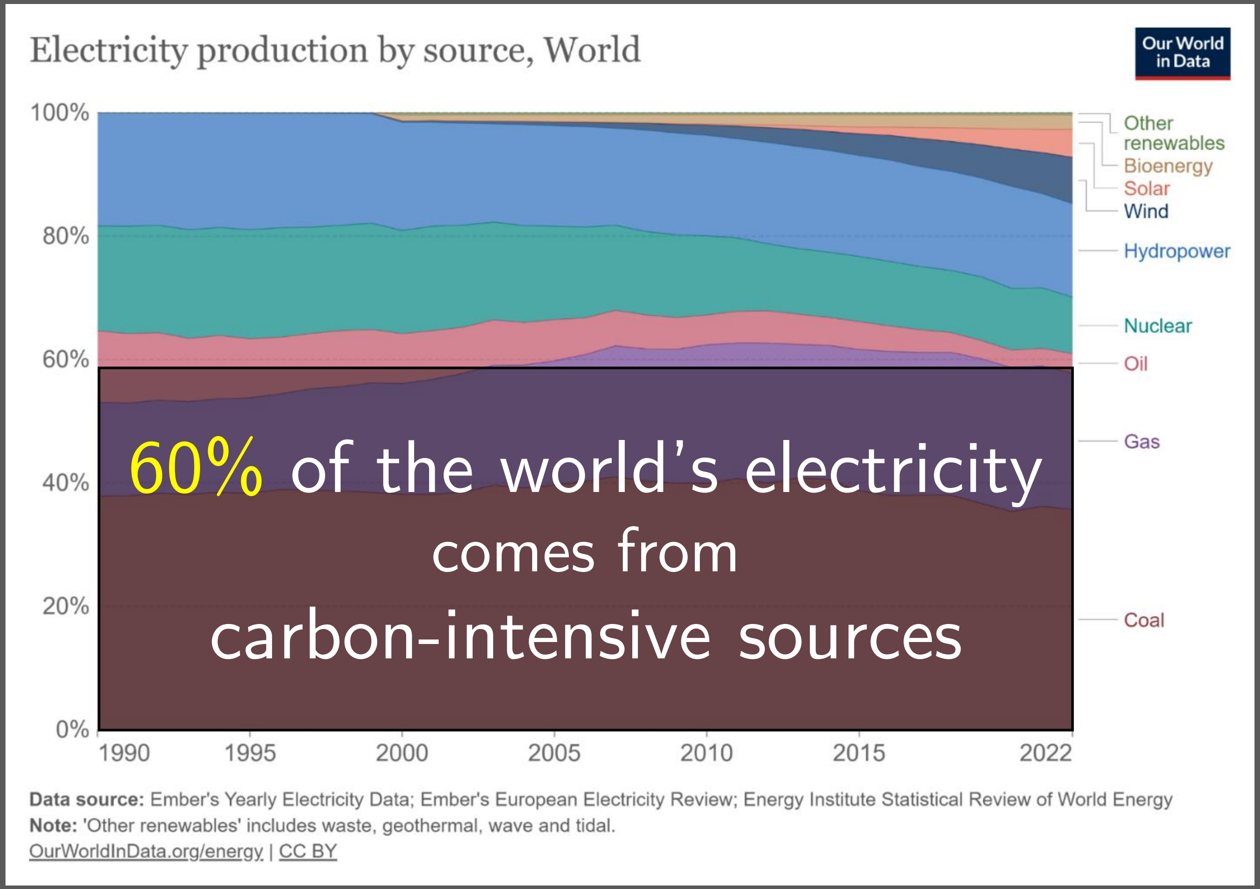
- It is easy to keep increasing network capacity
- It is much harder to keep increasing energy efficiency

▶ Total energy usage is likely to keep increasing.

"With great power comes great responsibility" and carbon footprint.

- It is easy to keep increasing network capacity
- It is much harder to keep increasing energy efficiency

- Total energy usage is likely to keep increasing.
- Producing electricity emits carbon.



https://ourworldindata.org/grapher/electricity-prod-source-stacked

SIGCOMM 2003

Greening of the Internet

Maruti Gupta
Department of Computer Science
Portland State University
Portland, OR 97207
mgupta@cs.pdx.edu

Suresh Singh
Department of Computer Science
Portland State University
Portland, OR 97207
singh@cs.pdx.edu

ABSTRACT

In this paper we examine the somewhat controversial subject of energy consumption of networking devices in the Internet, motivated by data collected by the U.S. Department of Commerce. We discuss the impact on network protocols of saving energy by putting network interfaces and other router & switch components to sleep. Using sample packet traces, we first show that it is indeed reasonable to do this and then we discuss the changes that may need to be made to current Internet protocols to support a more aggressive strategy for sleeping. Since this is a position paper, we do not present results but rather suggest interesting directions for core networking research. The impact of saving energy is huge, particularly in the developing world where energy is a precious resource whose scarcity hinders widespread Internet deployment.

Categories and Subject Descriptors

C.2.1 [**Network Architecture & Measurement**]: [Network Topology]; C.2.2 [**Network Protocols**]: [Routing Protocols]; C.2.6 [**Internetworking**]: [Routers, Standards]

General Terms

Algorithms, Measurement, Economics

Keywords

Energy, Internet, Protocols

1. INTRODUCTION

Recently, an opinion has been expressed in various quarters (see [5, 12]) that the energy consumption of the Internet is "too high" and that since this energy consumption can only grow as the Internet expands, this is a cause for concern. One may disagree, as we do, with the qualitative statement that the energy consumption of the Internet is too high, because it is a small fraction of the overall energy

Device	*Approximate Number Deployed*	*Total AEC* TW-h
Hubs	93.5 Million	1.6 TW-h
LAN Switch	95,000	3.2 TW-h
WAN Switch	50,000	0.15 TW-h
Router	3,257	1.1 TW-h
Total		6.05 TW-h

Table 1: Breakdown of energy draw of various networking devices (TW-h refers to Tera-Watt hours and AEC to Annual Electricity Consumption).

consumption. However, the absolute numbers do indicate a need to be more energy efficient. We use the analysis presented by these observers as a starting point to discuss an exciting new direction for future core networking research. We believe that if energy can be conserved by careful engineering then there is no reason why we should not do so as this has implications not only for reducing energy needs in the U.S. but also on speeding up Internet deployment and access in the developing world where energy is very scarce.

Table 1 [14] summarizes the energy consumption by Internet devices in the U.S. as of the year 2000. These values are copied from Tables 5-59 (Hub), 5-61 (LAN switch), 5-62 (WAN switch), and 5-64 (Router) of [14]. The data is broken up based on network device type, which is useful in analyzing where and how energy savings can be garnered. In order to arrive at the various energy numbers in the table, the authors took into account the percentage of different types of devices deployed (e.g., number of CISCO 2500 type routers, number of 7505s, etc) and then used the average energy consumption values of these devices to arrive at the final numbers shown in the table[1]. Two energy values missing from the table are the energy cost of *cooling* the equipment and that of UPS (Uninterruptable Power Supplies) equipment[2]. The future expectation is that the energy consumption of networking devices will increase by 1 TW-h by 2005 [14].

Expressed as a percentage of total U.S. energy expenditure in the year 2000, the energy drawn by the devices in Table 1 accounts for approximately 0.07% of the total. Given that this is almost negligible in comparison to other energy

[1]Note that the energy draw varies based on load and the values used in this study are based on observed average values.

[2]According to [14], air conditioning in data centers containing routing equipment costs approximately 20 – 60 Watts/ft^2.

The Internet core consumes more Joules per Bytes than wireless LANs.

SIGCOMM 2003

Greening of the Internet

Maruti Gupta
Department of Computer Science
Portland State University
Portland, OR 97207
mgupta@cs.pdx.edu

Suresh Singh
Department of Computer Science
Portland State University
Portland, OR 97207
singh@cs.pdx.edu

ABSTRACT

In this paper we examine the somewhat controversial subject of energy consumption of networking devices in the Internet, motivated by data collected by the U.S. Department of Commerce. We discuss the impact on network protocols of saving energy by putting network interfaces and other router & switch components to sleep. Using sample packet traces, we first show that it is indeed reasonable to do this and then we discuss the changes that may need to be made to current Internet protocols to support a more aggressive strategy for sleeping. Since this is a position paper, we do not present results but rather suggest interesting directions for core networking research. The impact of saving energy is huge, particularly in the developing world where energy is a precious resource whose scarcity hinders widespread Internet deployment.

Categories and Subject Descriptors

C.2.1 [**Network Architecture & Measurement**]: [Network Topology]; C.2.2 [**Network Protocols**]: [Routing Protocols]; C.2.6 [**Internetworking**]: [Routers, Standards]

General Terms

Algorithms, Measurement, Economics

Keywords

Energy, Internet, Protocols

1. INTRODUCTION

Recently, an opinion has been expressed in various quarters (see [5, 12]) that the energy consumption of the Internet is "too high" and that since this energy consumption can only grow as the Internet expands, this is a cause for concern. One may disagree, as we do, with the qualitative statement that the energy consumption of the Internet is too high, because it is a small fraction of the overall energy

Device	*Approximate Number Deployed*	*Total AEC* TW-h
Hubs	93.5 Million	1.6 TW-h
LAN Switch	95,000	3.2 TW-h
WAN Switch	50,000	0.15 TW-h
Router	3,257	1.1 TW-h
Total		6.05 TW-h

Table 1: Breakdown of energy draw of various networking devices (TW-h refers to Tera-Watt hours and AEC to Annual Electricity Consumption).

consumption. However, the absolute numbers do indicate a need to be more energy efficient. We use the analysis presented by these observers as a starting point to discuss an exciting new direction for future core networking research. We believe that if energy can be conserved by careful engineering then there is no reason why we should not do so as this has implications not only for reducing energy needs in the U.S. but also on speeding up Internet deployment and access in the developing world where energy is very scarce.

Table 1 [14] summarizes the energy consumption by Internet devices in the U.S. as of the year 2000. These values are copied from Tables 5-59 (Hub), 5-61 (LAN switch), 5-62 (WAN switch), and 5-64 (Router) of [14]. The data is broken up based on network device type, which is useful in analyzing where and how energy savings can be garnered. In order to arrive at the various energy numbers in the table, the authors took into account the percentage of different types of devices deployed (e.g., number of CISCO 2500 type routers, number of 7505s, etc) and then used the average energy consumption values of these devices to arrive at the final numbers shown in the table[1]. Two energy values missing from the table are the energy cost of *cooling* the equipment and that of UPS (Uninterruptable Power Supplies) equipment[2]. The future expectation is that the energy consumption of networking devices will increase by 1 TW-h by 2005 [14].

Expressed as a percentage of total U.S. energy expenditure in the year 2000, the energy drawn by the devices in Table 1 accounts for approximately 0.07% of the total. Given that this is almost negligible in comparison to other energy

[1]Note that the energy draw varies based on load and the values used in this study are based on observed average values.
[2]According to [14], air conditioning in data centers containing routing equipment costs approximately 20 – 60 Watts/ft^2.

The Internet core consumes more Joules per Bytes than wireless LANs.

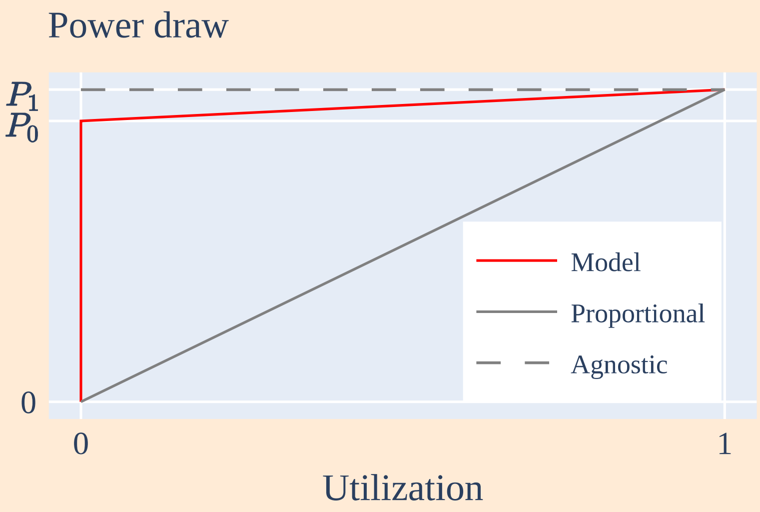
2x and 24x more…

depending on your hypotheses

1 Network devices are always “on.”

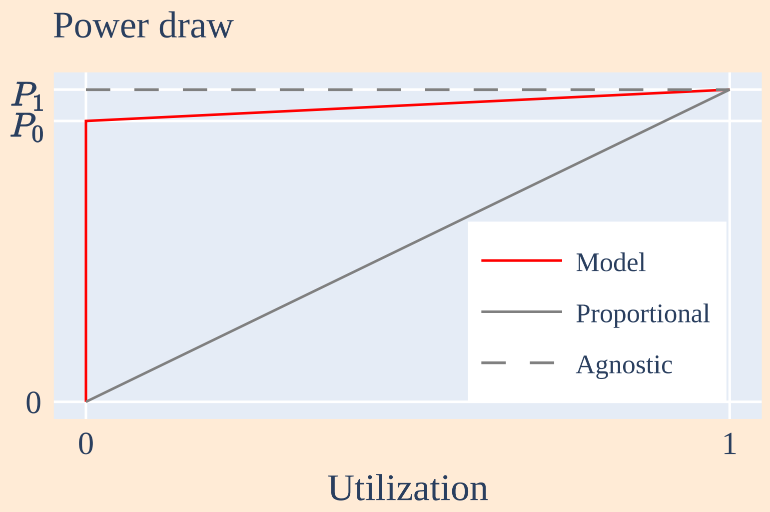
1 Network devices are always “on.”

2 Network devices’ energy consumption is mainly independent of traffic load.



1 Network devices are always "on."

2 Network devices' energy consumption is mainly independent of traffic load.



3 Network devices are under-utilized.

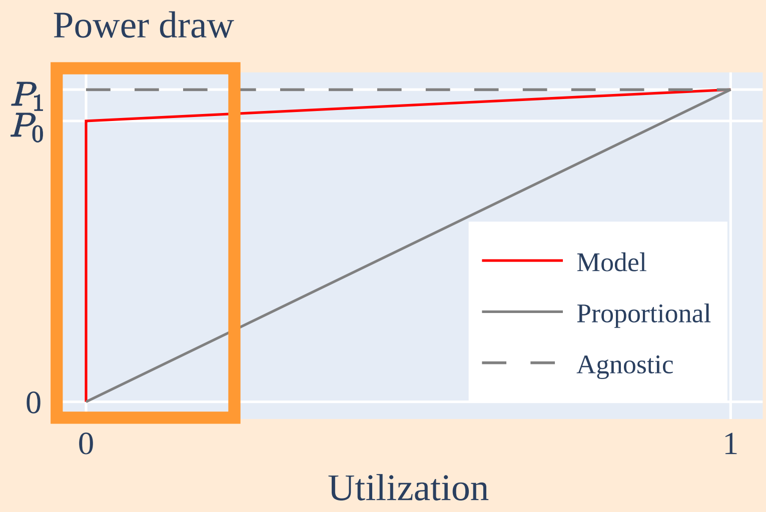
ISP overprovision networks to support

- Peak traffic
- Fault tolerance

1 Network devices are always "on."

2 Network devices' energy consumption is mainly independent of traffic load.

3 Network devices are under-utilized.

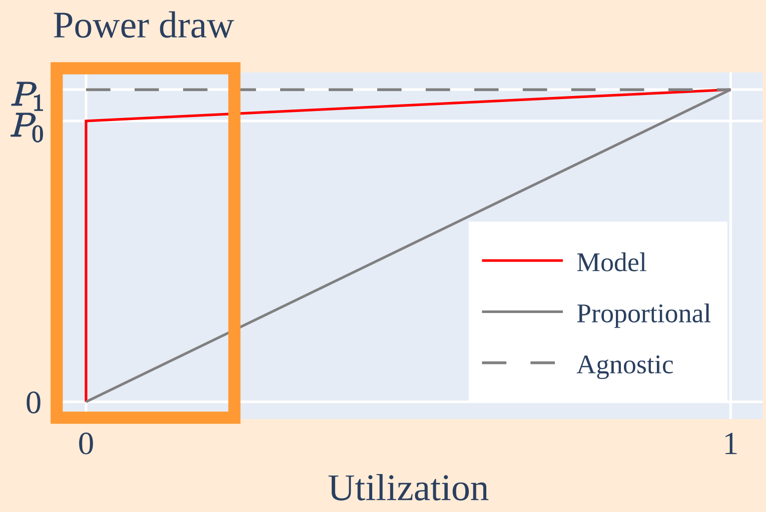


ISP overprovision networks to support

- Peak traffic
- Fault tolerance

You may wonder

Is that really true?



ISP overprovision networks to support

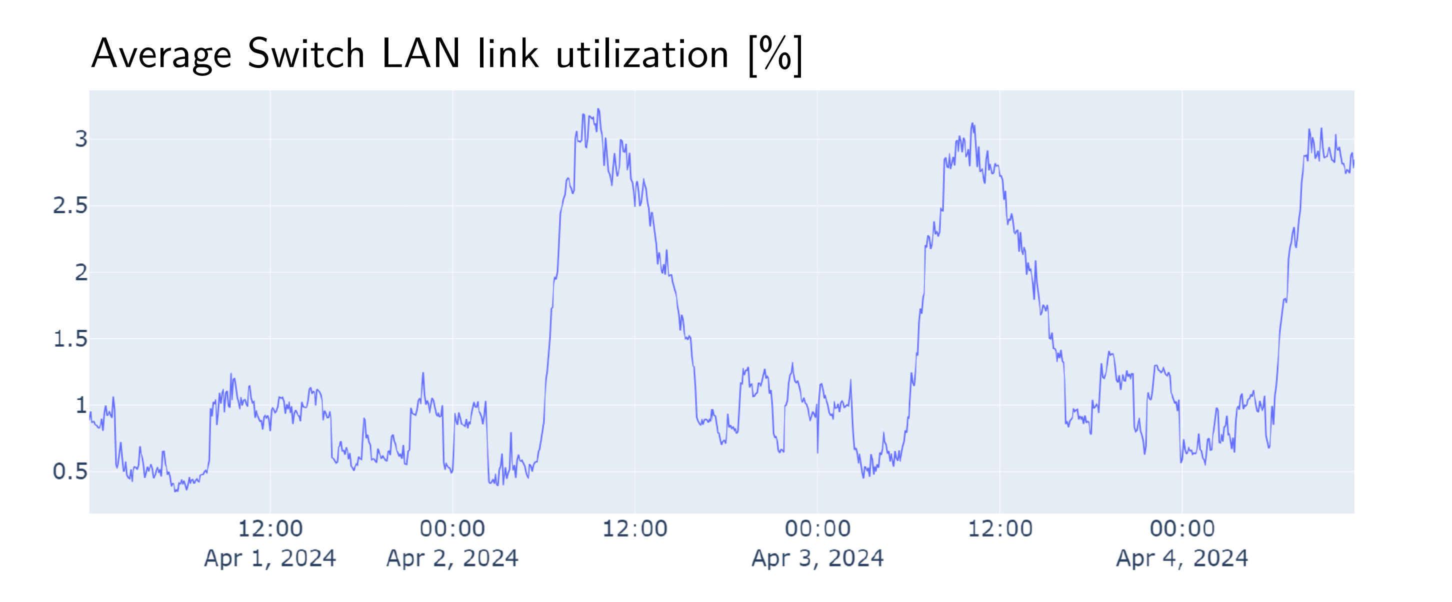
- Peak traffic
- Fault tolerance

What do it think the average link load on the Switch LAN network is?

What do it think the average link load on the Switch LAN network is?

2.1%

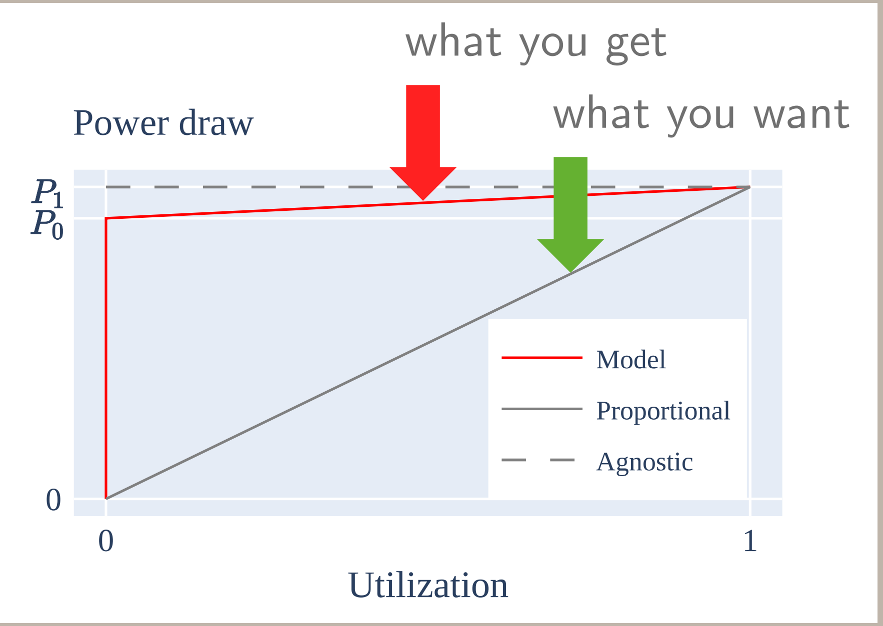
over 2.5 months of data, internal links only



1 Network devices
are always "on."

2 Network devices' energy consumption
is mainly independent of traffic load.

3 Network devices
are under-utilized.



ISP overprovision
networks to support

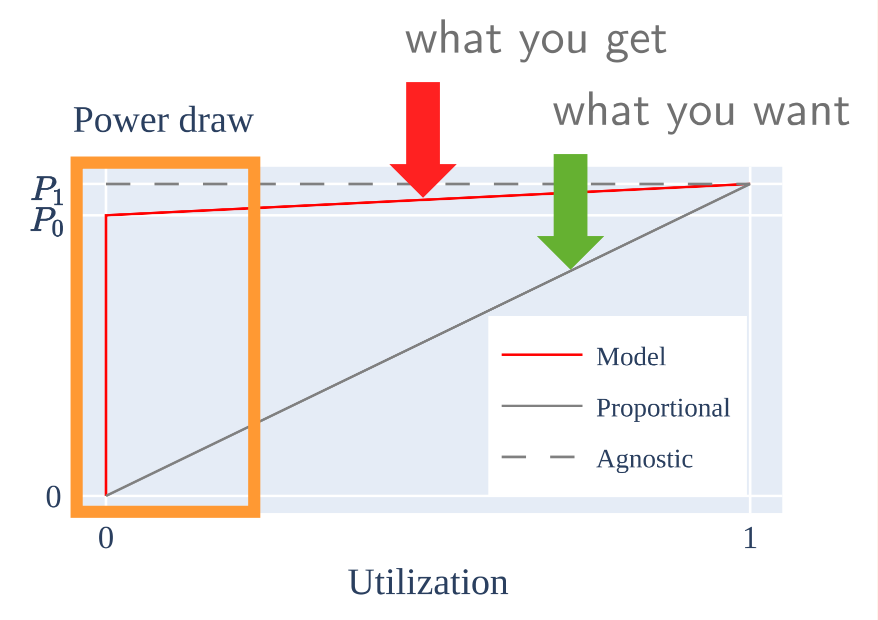
- Peak traffic
- Fault tolerance

There two ways to improve energy efficiency

- Run more often at high utilization

  “Buffer-and-Burst”

  Time-shifting



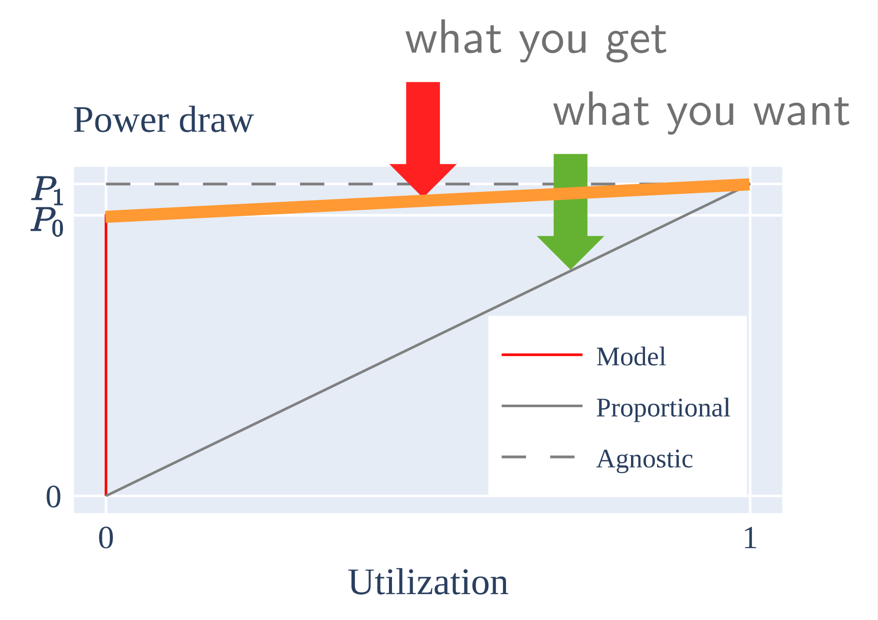
There two ways to improve energy efficiency

- Run more often at high utilization

  "Buffer-and-Burst"

  Time-shifting

- Take low-utilization power down



The basic idea is to turn off "stuff" whenever possible.

What can we possibly turn off?

- Ports
- Line cards
- Entire device...

The basic idea is to turn off “stuff” whenever possible.

What can we possibly turn off?

- Ports
- Line cards
- Entire device...

- Memory banks
- Power supplies
- LEDs ... etc.

The basic idea is to turn off “stuff” whenever possible.

What can we possibly turn off?

- Ports
- Line cards
- Entire device…
- Memory banks
- Power supplies
- LEDs … etc.

It can be more subtle than on/off.

- Change a port rate from 100G to 10G
- Down-clock the ASIC
- Cache frequently used FIB entries

What keeps your network up at night?

A link sleeping study of Switch LAN

The basic idea is to turn off "stuff" whenever possible. That's nothing new.

Academia NSDI 2008

Reducing Network Energy Consumption via Sleeping and Rate-Adaptation

Sergiu Nedevschi[*†] Lucian Popa[*†] Gianluca Iannaccone [†]
Sylvia Ratnasamy [†] David Wetherall[‡§]

Abstract

We present the design and evaluation of two forms of power management schemes that reduce the energy consumption of networks. The first is based on putting network components to sleep during idle times, reducing energy consumed in the absence of packets. The second is based on adapting the rate of network operation to the offered workload, reducing the energy consumed when actively processing packets.

For real-world traffic workloads and topologies and using power constants drawn from existing network equipment, we show that even simple schemes for sleeping or rate-adaptation can offer substantial savings. For instance, our practical algorithms stand to halve energy consumption for lightly utilized networks (10-20%). We show that these savings approach the maximum achievable by any algorithms using the same power management primitives. Moreover this energy can be saved without noticeably increasing loss and with a small and controlled increase in latency (<10ms). Finally, we show that both sleeping and rate adaptation are valuable depending (primarily) on the power profile of network equipment and the utilization of the network itself.

1 Introduction

In this paper, we consider power management for networks from a perspective that has recently begun to receive attention: the conservation of energy for operating and environmental reasons. Energy consumption in network exchanges is rising as higher capacity network equipment becomes more power-hungry and requires greater amounts of cooling. Combined with rising energy costs, this has made the cost of powering network exchanges a substantial and growing fraction of the total cost of ownership – up to half by some estimates[23]. Various studies now estimate the power usage of the US network infrastructure at between 5 and 24 TWh/year[25, 26], or $0.5-2.4B/year at a rate of $0.10/KWh, depending on what is included. Public via standards such as EnergyStar. In fact, EnergyStar standard proposals for 2009 discuss slower operation of network links to conserve energy when idle. A new IEEE 802.3az Task Force was launched in early 2007 to focus on this issue for Ethernet [15].

Fortunately, there is an opportunity for substantial reductions in the energy consumption of existing networks due to two factors. First, networks are provisioned for worst-case or busy-hour load, and this load typically exceeds their long-term utilization by a wide margin. For example, measurements reveal backbone utilizations under 30% [16] and up to hour-long idle times at access points in enterprise wireless networks [17]. Second, the energy consumption of network equipment remains substantial even when the network is idle. The implication of these factors is that *most* of the energy consumed in networks is wasted.

Our work is an initial exploration of how overall network energy consumption might be reduced without adversely affecting network performance. This will require two steps. First, network equipment ranging from routers to switches and NICs will need power management primitives at the hardware level. By analogy, power management in computers has evolved around hardware support for *sleep* and *performance* states. The former (*e.g.,*C-states in Intel processors) reduce idle consumption by powering off sub-components to different extents, while the latter (*e.g.,*SpeedStep, P-states in Intel processors) tradeoff performance for power via operating frequency. Second, network protocols will need to make use of the hardware primitives to best effect. Again, by analogy with computers, power management preferences control how the system switches between the available states to save energy with minimal impact on users.

Of these two steps, our focus is on the network protocols. Admittedly, these protocols build on hardware support for power management that is in its infancy for networking equipment. Yet the necessary support

RIPE

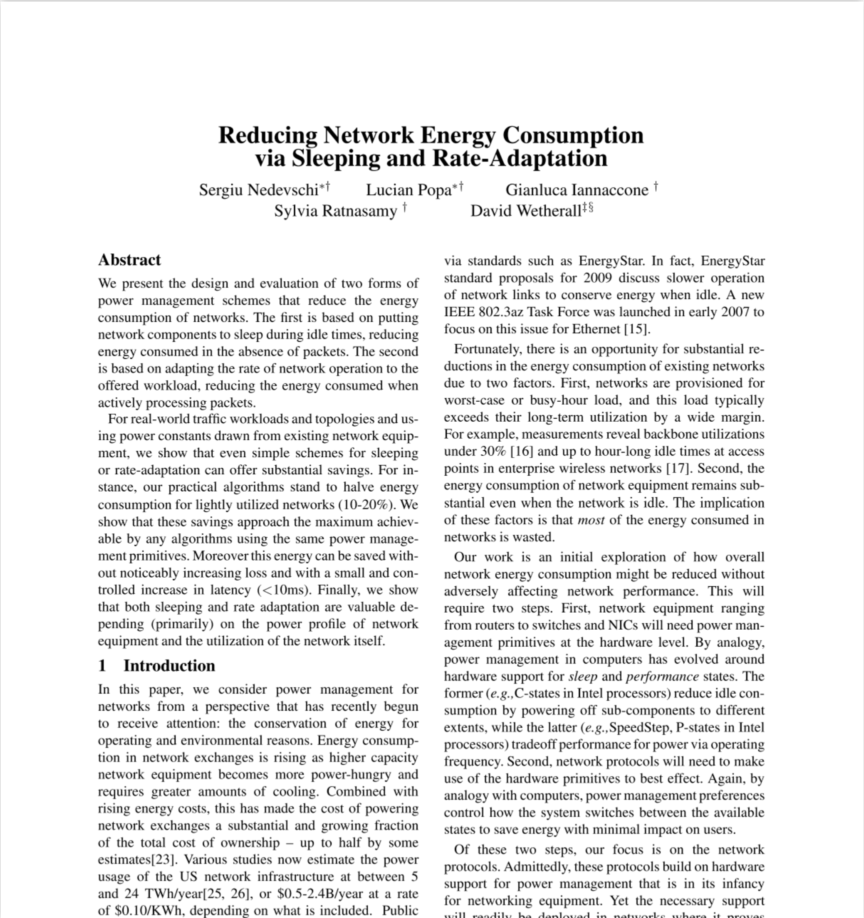
Techniques to reduce network power consumption

Peter Ehiwe, May 2023 @RIPE86

The theory says we can save tens of energy % in ISP networks.

Academia

NSDI 2008



Reducing Network Energy Consumption via Sleeping and Rate-Adaptation

Sergiu Nedevschi*† Lucian Popa*† Gianluca Iannaccone †
Sylvia Ratnasamy † David Wetherall‡§

Abstract

We present the design and evaluation of two forms of power management schemes that reduce the energy consumption of networks. The first is based on putting network components to sleep during idle times, reducing energy consumed in the absence of packets. The second is based on adapting the rate of network operation to the offered workload, reducing the energy consumed when actively processing packets.

For real-world traffic workloads and topologies and using power constants drawn from existing network equipment, we show that even simple schemes for sleeping or rate-adaptation can offer substantial savings. For instance, our practical algorithms stand to halve energy consumption for lightly utilized networks (10-20%). We show that these savings approach the maximum achievable by any algorithms using the same power management primitives. Moreover this energy can be saved without noticeably increasing loss and with a small and controlled increase in latency (<10ms). Finally, we show that both sleeping and rate adaptation are valuable depending (primarily) on the power profile of network equipment and the utilization of the network itself.

1 Introduction

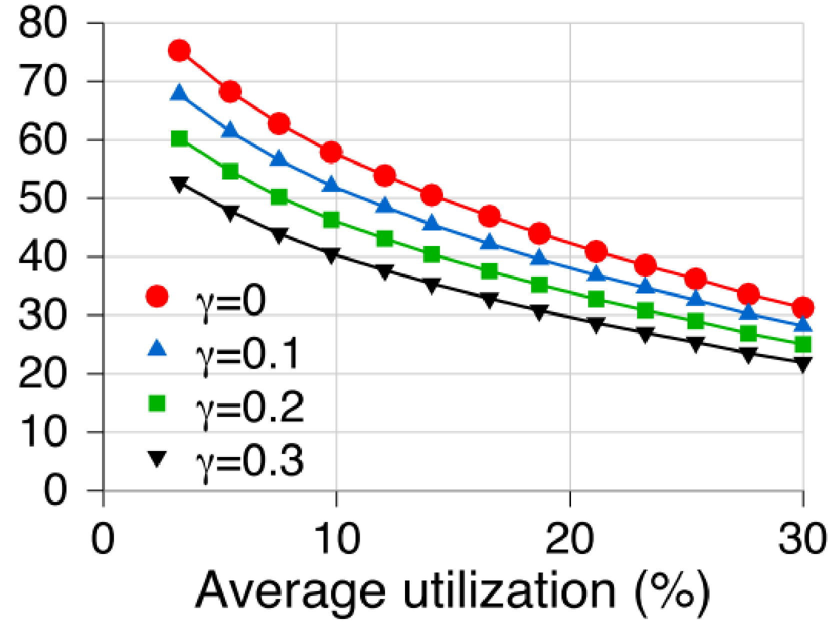
In this paper, we consider power management for networks from a perspective that has recently begun to receive attention: the conservation of energy for operating and environmental reasons. Energy consumption in network exchanges is rising as higher capacity network equipment becomes more power-hungry and requires greater amounts of cooling. Combined with rising energy costs, this has made the cost of powering network exchanges a substantial and growing fraction of the total cost of ownership – up to half by some estimates[23]. Various studies now estimate the power usage of the US network infrastructure at between 5 and 24 TWh/year[25, 26], or $0.5-2.4B/year at a rate of $0.10/KWh, depending on what is included. Public

via standards such as EnergyStar. In fact, EnergyStar standard proposals for 2009 discuss slower operation of network links to conserve energy when idle. A new IEEE 802.3az Task Force was launched in early 2007 to focus on this issue for Ethernet [15].

Fortunately, there is an opportunity for substantial reductions in the energy consumption of existing networks due to two factors. First, networks are provisioned for worst-case or busy-hour load, and this load typically exceeds their long-term utilization by a wide margin. For example, measurements reveal backbone utilizations under 30% [16] and up to hour-long idle times at access points in enterprise wireless networks [17]. Second, the energy consumption of network equipment remains substantial even when the network is idle. The implication of these factors is that *most* of the energy consumed in networks is wasted.

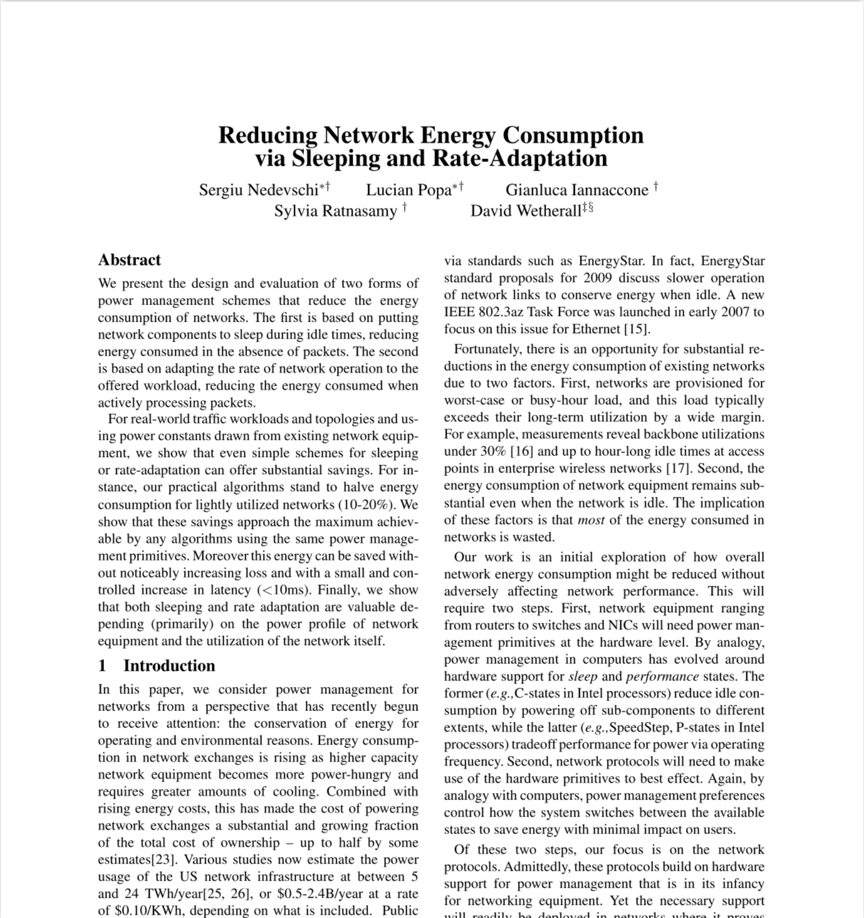
Our work is an initial exploration of how overall network energy consumption might be reduced without adversely affecting network performance. This will require two steps. First, network equipment ranging from routers to switches and NICs will need power management primitives at the hardware level. By analogy, power management in computers has evolved around hardware support for *sleep* and *performance* states. The former (*e.g.*,C-states in Intel processors) reduce idle consumption by powering off sub-components to different extents, while the latter (*e.g.*,SpeedStep, P-states in Intel processors) tradeoff performance for power via operating frequency. Second, network protocols will need to make use of the hardware primitives to best effect. Again, by analogy with computers, power management preferences control how the system switches between the available states to save energy with minimal impact on users.

Of these two steps, our focus is on the network protocols. Admittedly, these protocols build on hardware support for power management that is in its infancy for networking equipment. Yet the necessary support



The theory says we can save tens of energy % in ISP networks.

Academia NSDI 2008



Reducing Network Energy Consumption via Sleeping and Rate-Adaptation

Sergiu Nedevschi*† Lucian Popa*† Gianluca Iannaccone † Sylvia Ratnasamy † David Wetherall‡§

Abstract

We present the design and evaluation of two forms of power management schemes that reduce the energy consumption of networks. The first is based on putting network components to sleep during idle times, reducing energy consumed in the absence of packets. The second is based on adapting the rate of network operation to the offered workload, reducing the energy consumed when actively processing packets.

For real-world traffic workloads and topologies and using power constants drawn from existing network equipment, we show that even simple schemes for sleeping or rate-adaptation can offer substantial savings. For instance, our practical algorithms stand to halve energy consumption for lightly utilized networks (10-20%). We show that these savings approach the maximum achievable by any algorithms using the same power management primitives. Moreover this energy can be saved without noticeably increasing loss and with a small and controlled increase in latency (<10ms). Finally, we show that both sleeping and rate adaptation are valuable depending (primarily) on the power profile of network equipment and the utilization of the network itself.

1 Introduction

In this paper, we consider power management for networks from a perspective that has recently begun to receive attention: the conservation of energy for operating and environmental reasons. Energy consumption in network exchanges is rising as higher capacity network equipment becomes more power-hungry and requires greater amounts of cooling. Combined with rising energy costs, this has made the cost of powering network exchanges a substantial and growing fraction of the total cost of ownership – up to half by some estimates[23]. Various studies now estimate the power usage of the US network infrastructure at between 5 and 24 TWh/year[25, 26], or $0.5-2.4B/year at a rate of $0.10/KWh, depending on what is included. Public

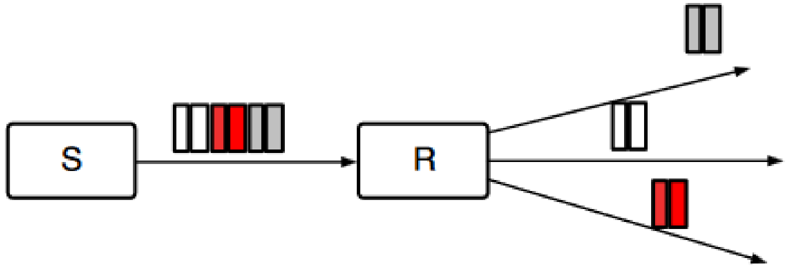
via standards such as EnergyStar. In fact, EnergyStar standard proposals for 2009 discuss slower operation of network links to conserve energy when idle. A new IEEE 802.3az Task Force was launched in early 2007 to focus on this issue for Ethernet [15].

Fortunately, there is an opportunity for substantial reductions in the energy consumption of existing networks due to two factors. First, networks are provisioned for worst-case or busy-hour load, and this load typically exceeds their long-term utilization by a wide margin. For example, measurements reveal backbone utilizations under 30% [16] and up to hour-long idle times at access points in enterprise wireless networks [17]. Second, the energy consumption of network equipment remains substantial even when the network is idle. The implication of these factors is that *most* of the energy consumed in networks is wasted.

Our work is an initial exploration of how overall network energy consumption might be reduced without adversely affecting network performance. This will require two steps. First, network equipment ranging from routers to switches and NICs will need power management primitives at the hardware level. By analogy, power management in computers has evolved around hardware support for *sleep* and *performance* states. The former (*e.g.*,C-states in Intel processors) reduce idle consumption by powering off sub-components to different extents, while the latter (*e.g.*,SpeedStep, P-states in Intel processors) tradeoff performance for power via operating frequency. Second, network protocols will need to make use of the hardware primitives to best effect. Again, by analogy with computers, power management preferences control how the system switches between the available states to save energy with minimal impact on users.

Of these two steps, our focus is on the network protocols. Admittedly, these protocols build on hardware support for power management that is in its infancy for networking equipment. Yet the necessary support

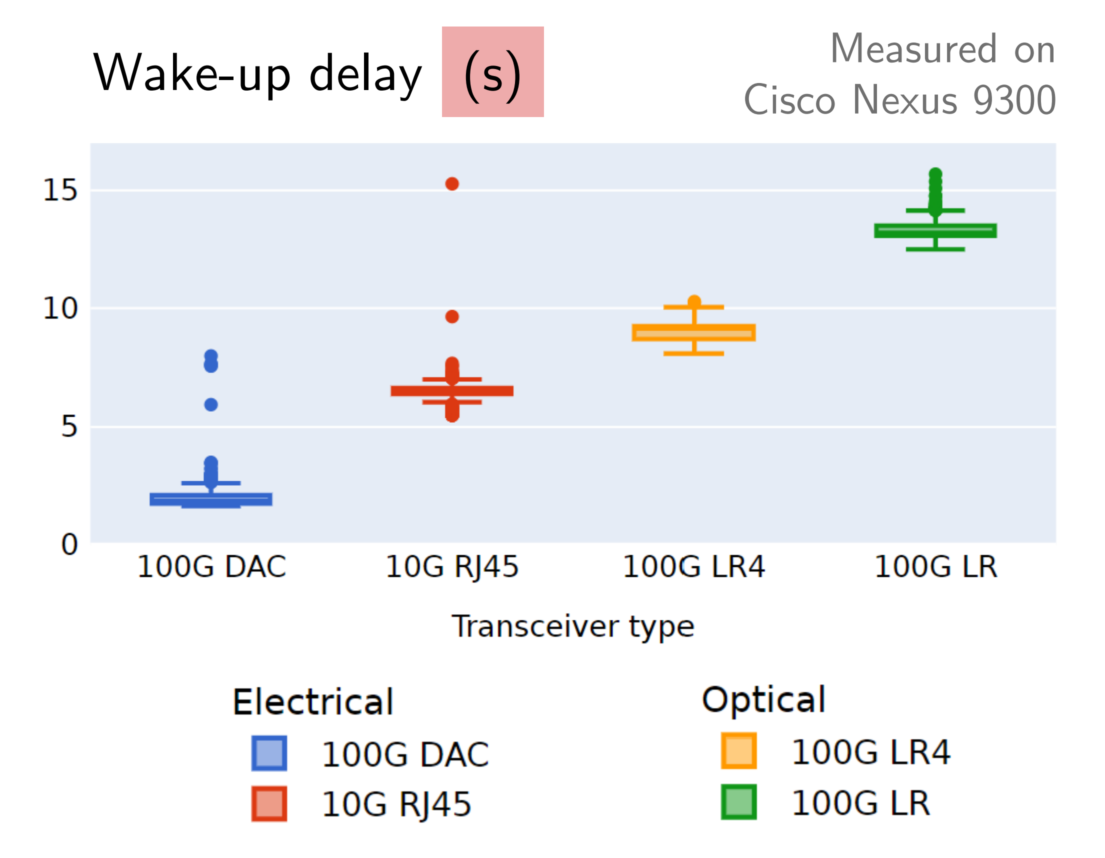
How? Buffer-and-Burst



Assuming

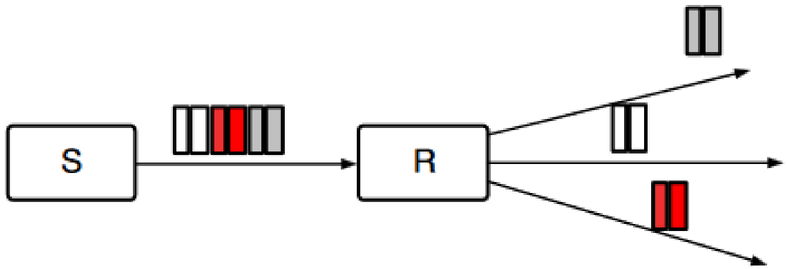
- Wake-up delay $1ms$
- Buffering time $10ms$

Practice



Theory

How? Buffer-and-Burst



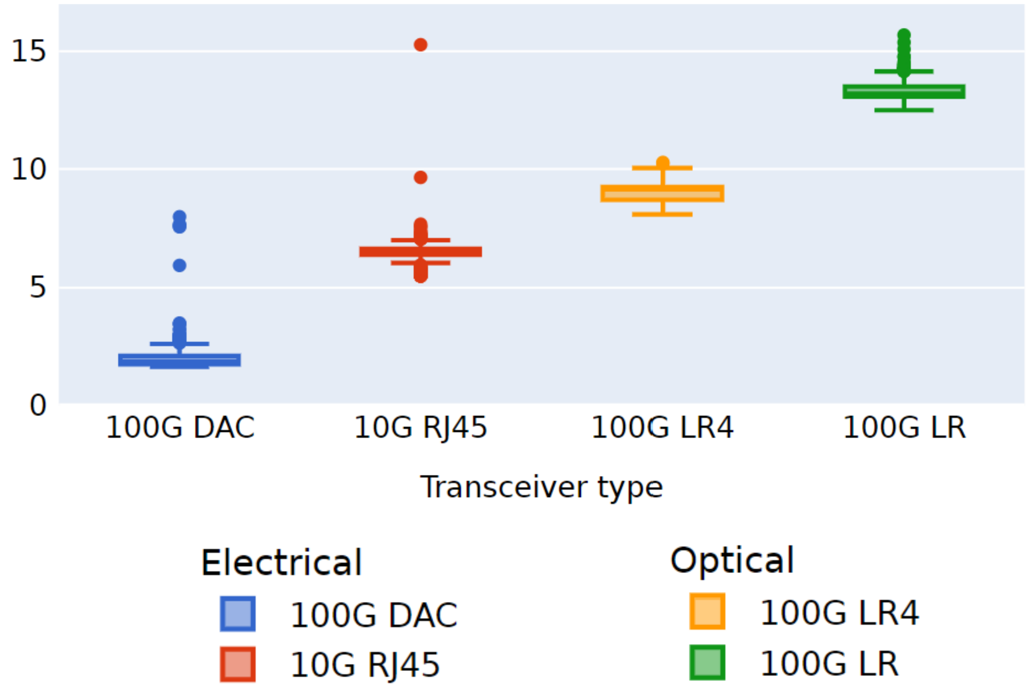
Assuming

- Wake-up delay $1ms$
- Buffering time $10ms$

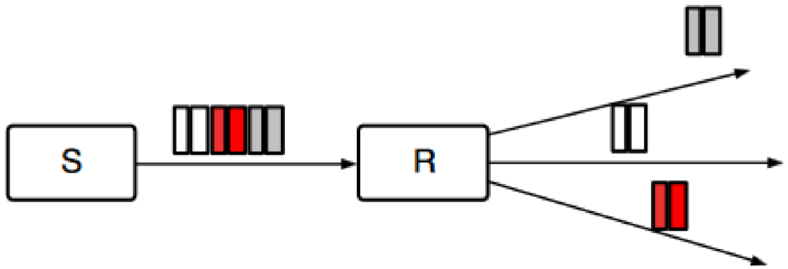
In practice, transcievers are 1000x slower to start than required for savings via buffering.

Wake-up delay (s)

Measured on Cisco Nexus 9300



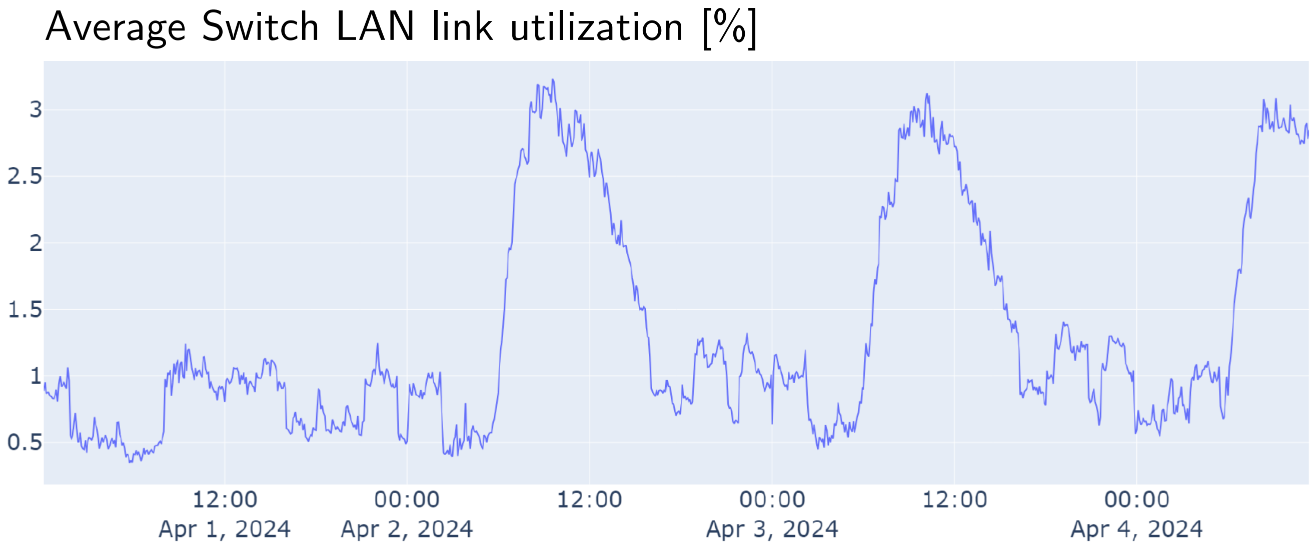
How? Buffer-and-Burst



Assuming

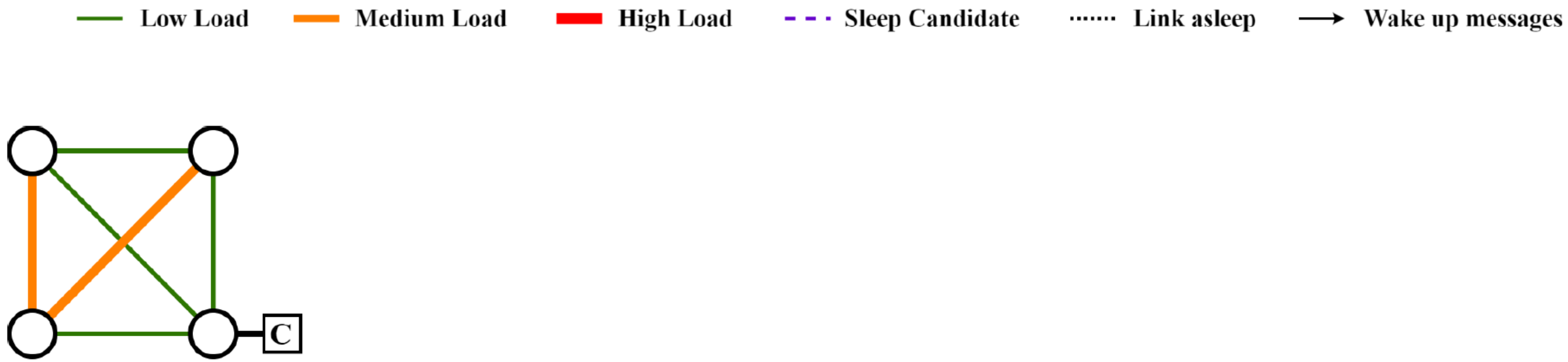
- Wake-up delay $1ms$
- Buffering time $10ms$

We can still “sleep” at longer timescales.



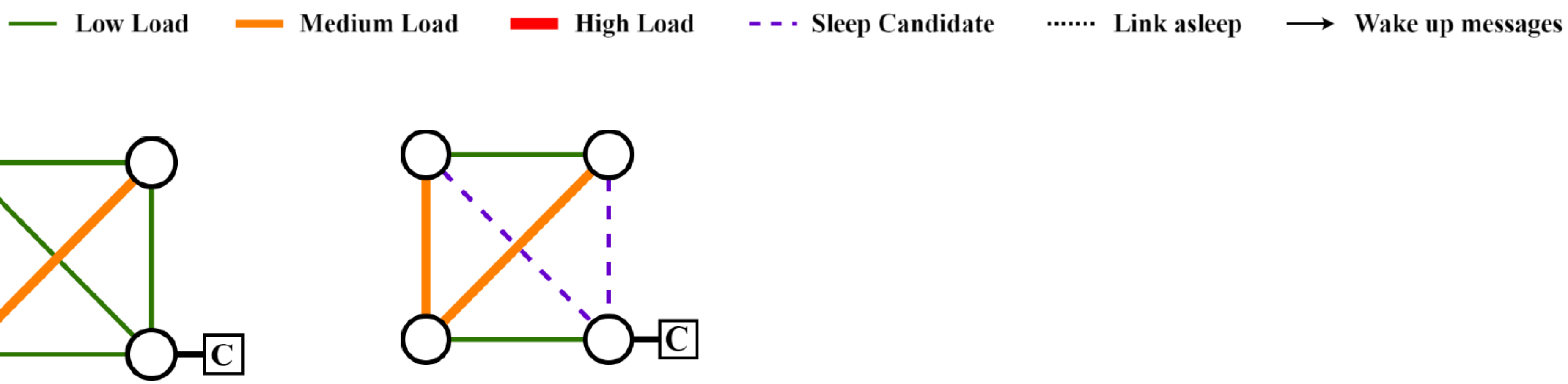
Ultimately, it is very similar to a traditional TE problem.

We can still "sleep" at longer timescales. How?



Collect Link Loads

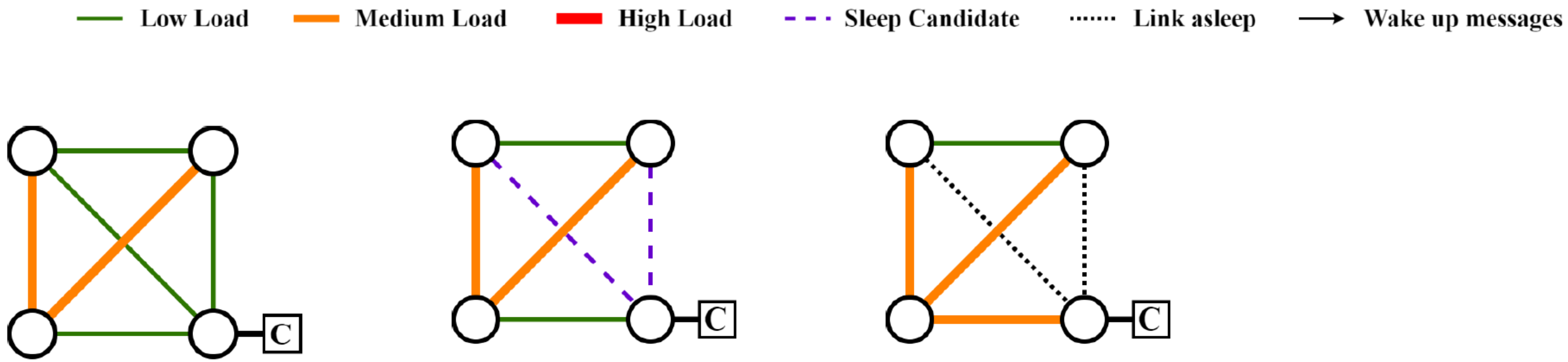
We can still "sleep" at longer timescales. How?



Collect
Link Loads

Select links
to turn off

We can still "sleep" at longer timescales. How?

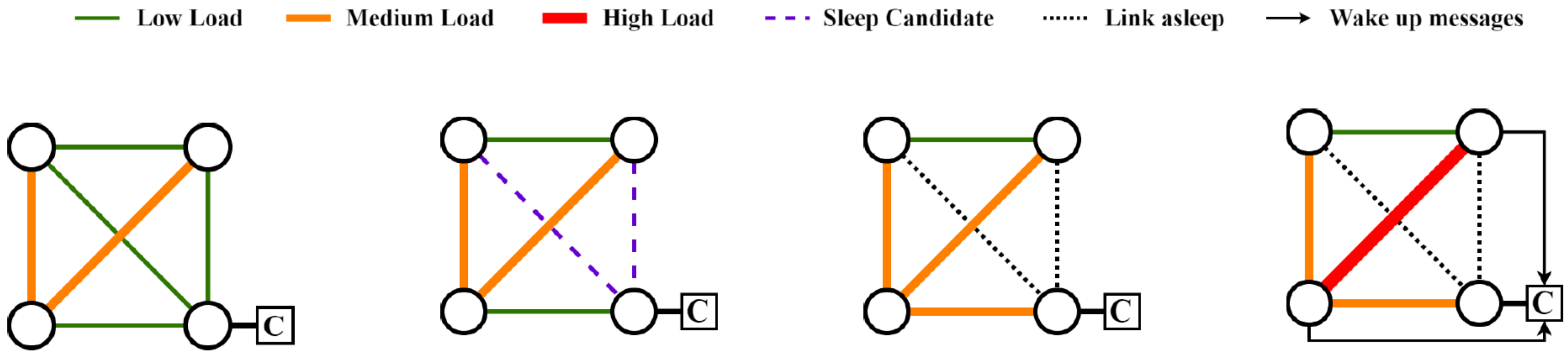


Collect Link Loads

Select links to turn off

Turn links off

We can still "sleep" at longer timescales. How?



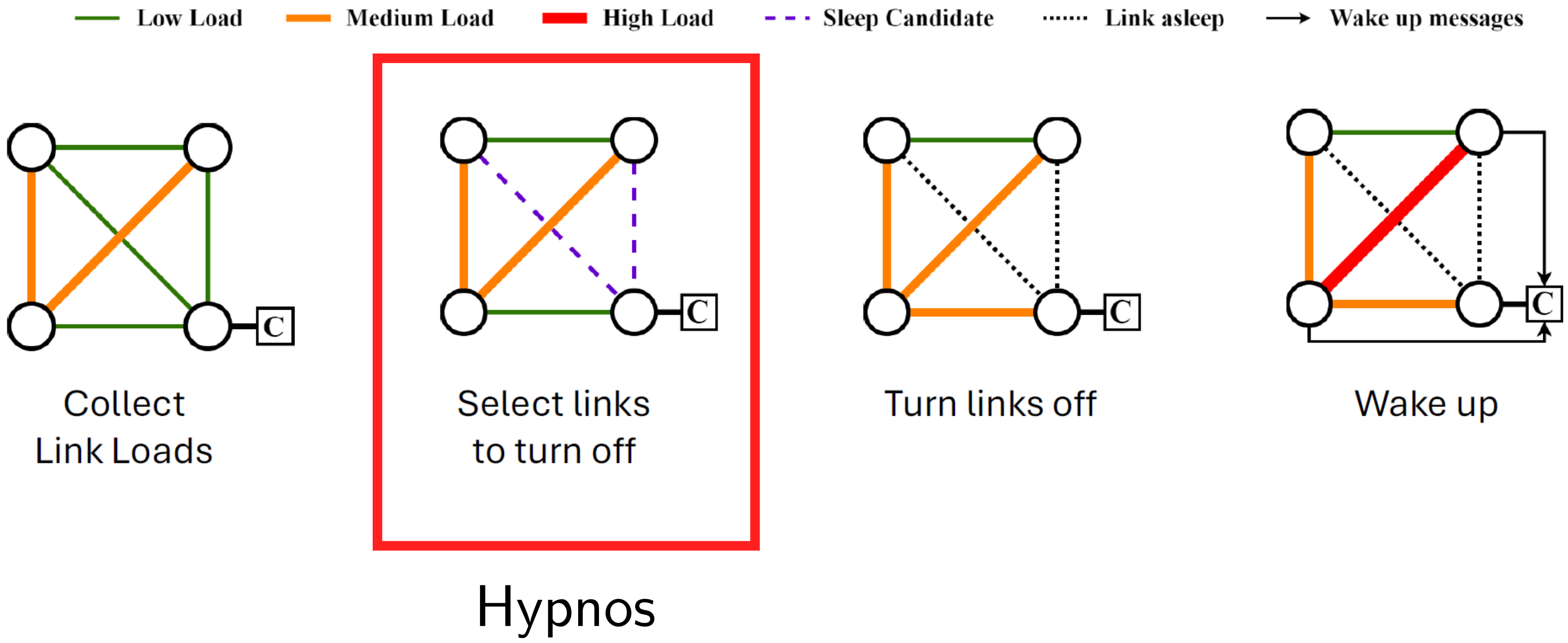
Collect Link Loads

Select links to turn off

Turn links off

Wake up

The hard bit is selecting the links to turn off.



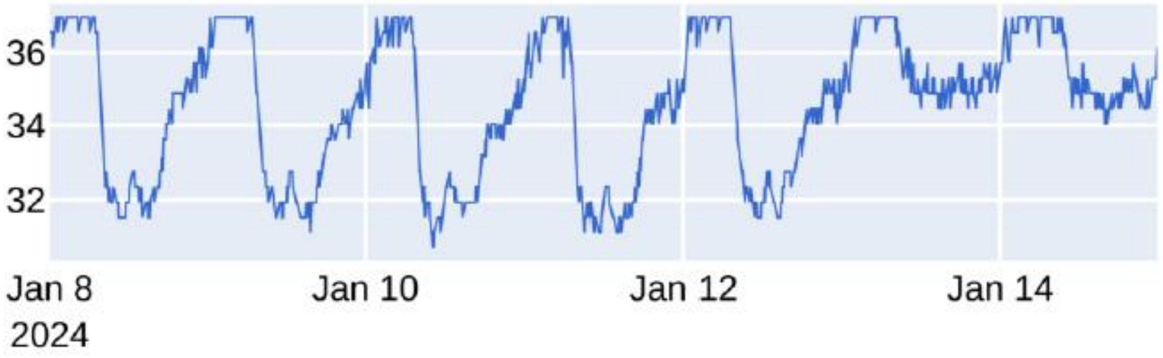
Hypnos selects sleeping links with four simple heuristics.

Hypnos – Greek god of sleep

- Prioritize low-utilization links
- Cap the total amount of rerouted traffic
- Check for local bottlenecks
- Check for global connectivity

Hypnos turns 1/3 of the links off without inducing congestion.

Number of links turned off [%]

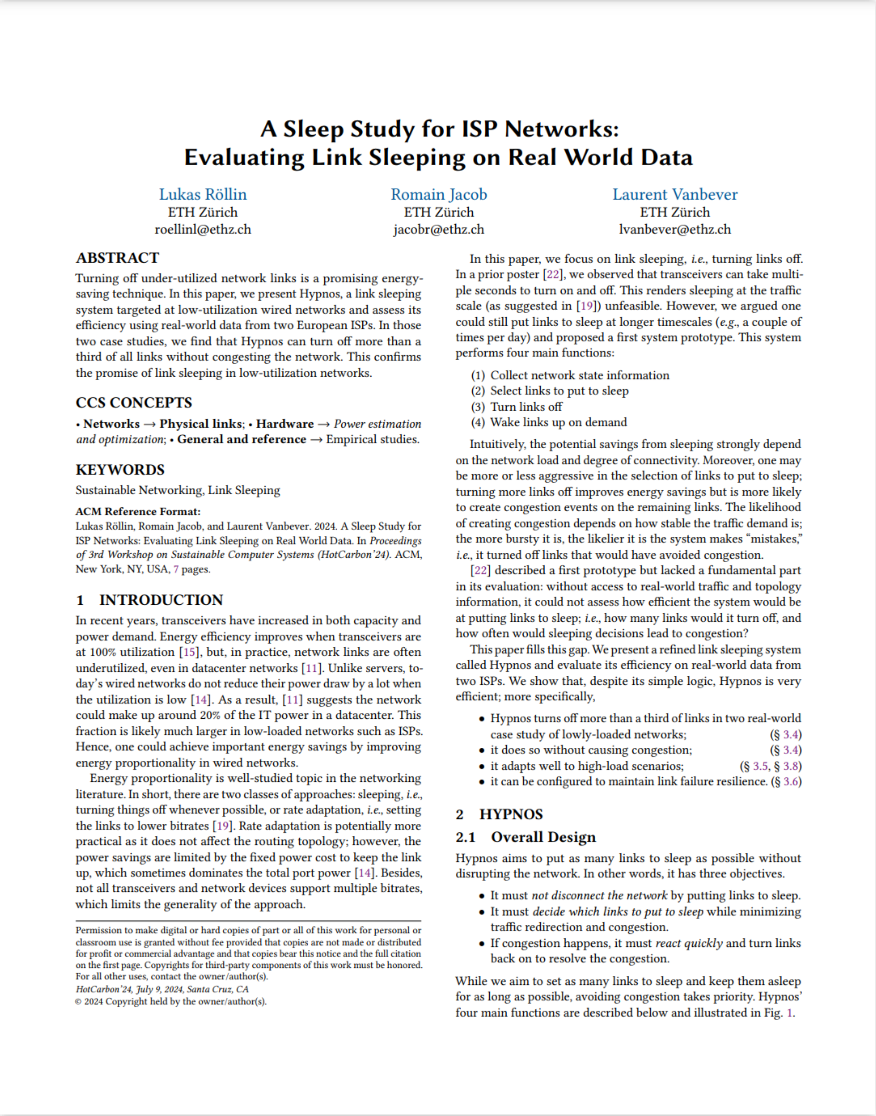


Simulation results

One week sample

Check the full paper for details.

- To be presented at HotCarbon'24
- Happening on July 9th
- Remote attendance is free!
  but registration is required



A Sleep Study for ISP Networks: Evaluating Link Sleeping on Real World Data

Lukas Röllin
ETH Zürich
roellinl@ethz.ch

Romain Jacob
ETH Zürich
jacobr@ethz.ch

Laurent Vanbever
ETH Zürich
lvanbever@ethz.ch

ABSTRACT

Turning off under-utilized network links is a promising energy-saving technique. In this paper, we present Hypnos, a link sleeping system targeted at low-utilization wired networks and assess its efficiency using real-world data from two European ISPs. In those two case studies, we find that Hypnos can turn off more than a third of all links without congesting the network. This confirms the promise of link sleeping in low-utilization networks.

CCS CONCEPTS

• **Networks** → **Physical links**; • **Hardware** → *Power estimation and optimization*; • **General and reference** → Empirical studies.

KEYWORDS

Sustainable Networking, Link Sleeping

ACM Reference Format:

Lukas Röllin, Romain Jacob, and Laurent Vanbever. 2024. A Sleep Study for ISP Networks: Evaluating Link Sleeping on Real World Data. In *Proceedings of 3rd Workshop on Sustainable Computer Systems (HotCarbon'24).* ACM, New York, NY, USA, 7 pages.

1 INTRODUCTION

In recent years, transceivers have increased in both capacity and power demand. Energy efficiency improves when transceivers are at 100% utilization [15], but, in practice, network links are often underutilized, even in datacenter networks [11]. Unlike servers, today's wired networks do not reduce their power draw by a lot when the utilization is low [14]. As a result, [11] suggests the network could make up around 20% of the IT power in a datacenter. This fraction is likely much larger in low-loaded networks such as ISPs. Hence, one could achieve important energy savings by improving energy proportionality in wired networks.

Energy proportionality is well-studied topic in the networking literature. In short, there are two classes of approaches: sleeping, *i.e.*, turning things off whenever possible, or rate adaptation, *i.e.*, setting the links to lower bitrates [19]. Rate adaptation is potentially more practical as it does not affect the routing topology; however, the power savings are limited by the fixed power cost to keep the link up, which sometimes dominates the total port power [14]. Besides, not all transceivers and network devices support multiple bitrates, which limits the generality of the approach.

In this paper, we focus on link sleeping, *i.e.*, turning links off. In a prior poster [22], we observed that transceivers can take multiple seconds to turn on and off. This renders sleeping at the traffic scale (as suggested in [19]) unfeasible. However, we argued one could still put links to sleep at longer timescales (*e.g.*, a couple of times per day) and proposed a first system prototype. This system performs four main functions:

(1) Collect network state information
(2) Select links to put to sleep
(3) Turn links off
(4) Wake links up on demand

Intuitively, the potential savings from sleeping strongly depend on the network load and degree of connectivity. Moreover, one may be more or less aggressive in the selection of links to put to sleep; turning more links off improves energy savings but is more likely to create congestion events on the remaining links. The likelihood of creating congestion depends on how stable the traffic demand is; the more bursty it is, the likelier it is the system makes "mistakes," *i.e.*, it turned off links that would have avoided congestion.

[22] described a first prototype but lacked a fundamental part in its evaluation: without access to real-world traffic and topology information, it could not assess how efficient the system would be at putting links to sleep; *i.e.*, how many links would it turn off, and how often would sleeping decisions lead to congestion?

This paper fills this gap. We present a refined link sleeping system called Hypnos and evaluate its efficiency on real-world data from two ISPs. We show that, despite its simple logic, Hypnos is very efficient; more specifically,

- Hypnos turns off more than a third of links in two real-world case study of lowly-loaded networks; (§ 3.4)
- it does so without causing congestion; (§ 3.4)
- it adapts well to high-load scenarios; (§ 3.5, § 3.8)
- it can be configured to maintain link failure resilience. (§ 3.6)

2 HYPNOS

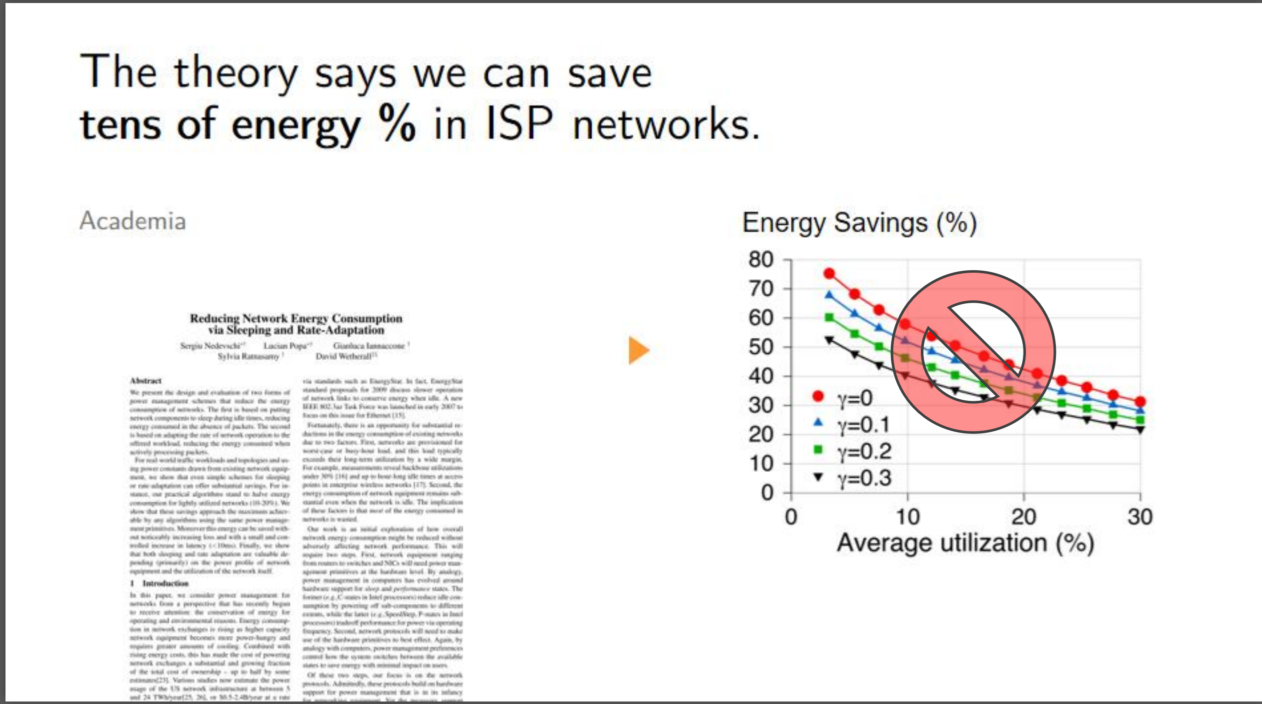
2.1 Overall Design

Hypnos aims to put as many links to sleep as possible without disrupting the network. In other words, it has three objectives.

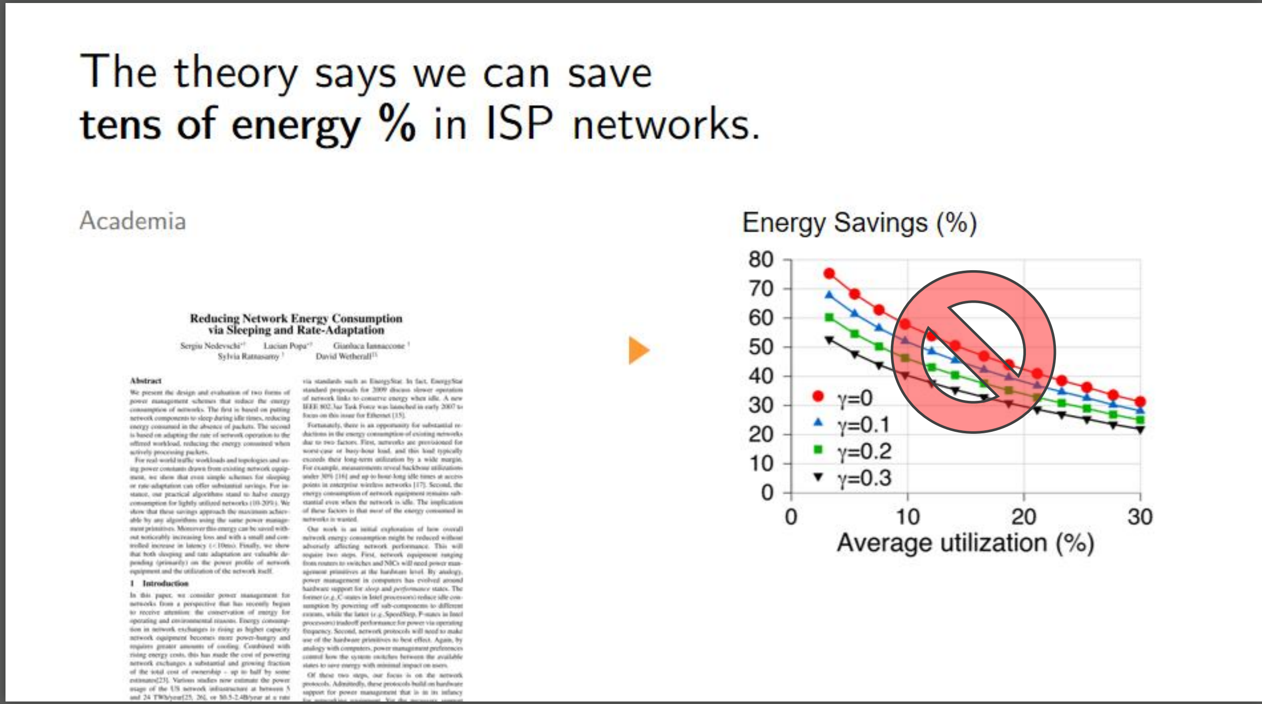
- It must *not disconnect the network* by putting links to sleep.
- It must *decide which links to put to sleep* while minimizing traffic redirection and congestion.
- If congestion happens, it must *react quickly* and turn links back on to resolve the congestion.

While we aim to set as many links to sleep and keep them asleep for as long as possible, avoiding congestion takes priority. Hypnos' four main functions are described below and illustrated in Fig. 1.

How much energy can we really save?



How much energy can we really save?



With Hypnos?

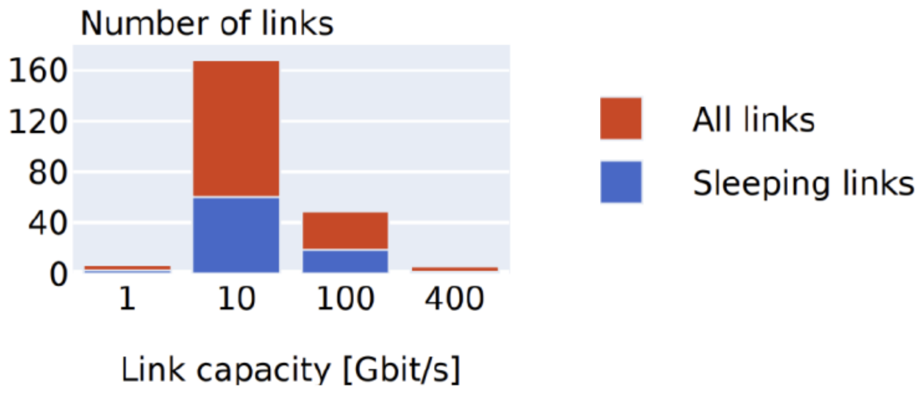
Transciever power numbers

Datasheet values, LR models

Capacity	1G	10G	100G	400G
Power	1W	1W	4W	10.5W

times

Average number of links off, per type



Simulation predicts 35% savings

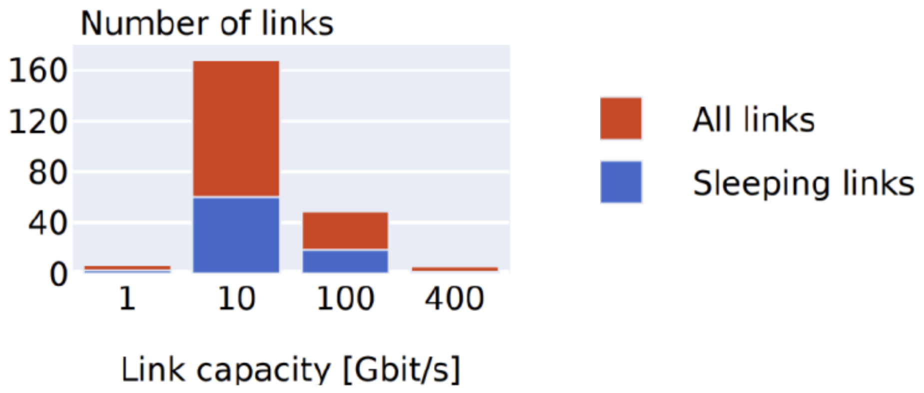
Transciever power numbers

Datasheet values, LR models

Capacity	1G	10G	100G	400G
Power	1W	1W	4W	10.5W

times

Average number of links off, per type



equals

35% savings

Simulation predicts 35% savings on transceiver power!

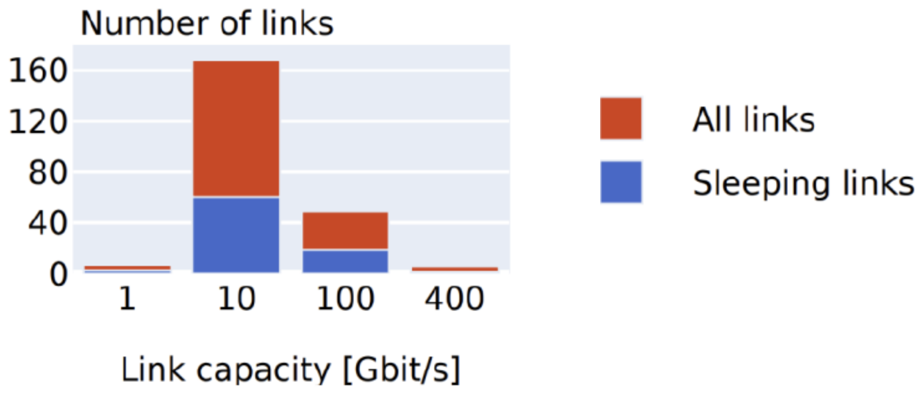
Transciever power numbers

Datasheet values, LR models

Capacity	1G	10G	100G	400G
Power	1W	1W	4W	10.5W

times

Average number of links off, per type



equals

35% savings on transceiver power

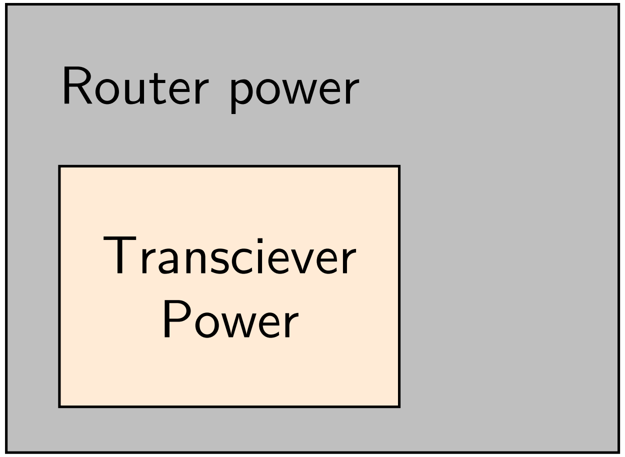
that is

~300 out of 850W

How big is the transceiver power relative to the total?

How big is the transceiver power relative to the total? It depends...

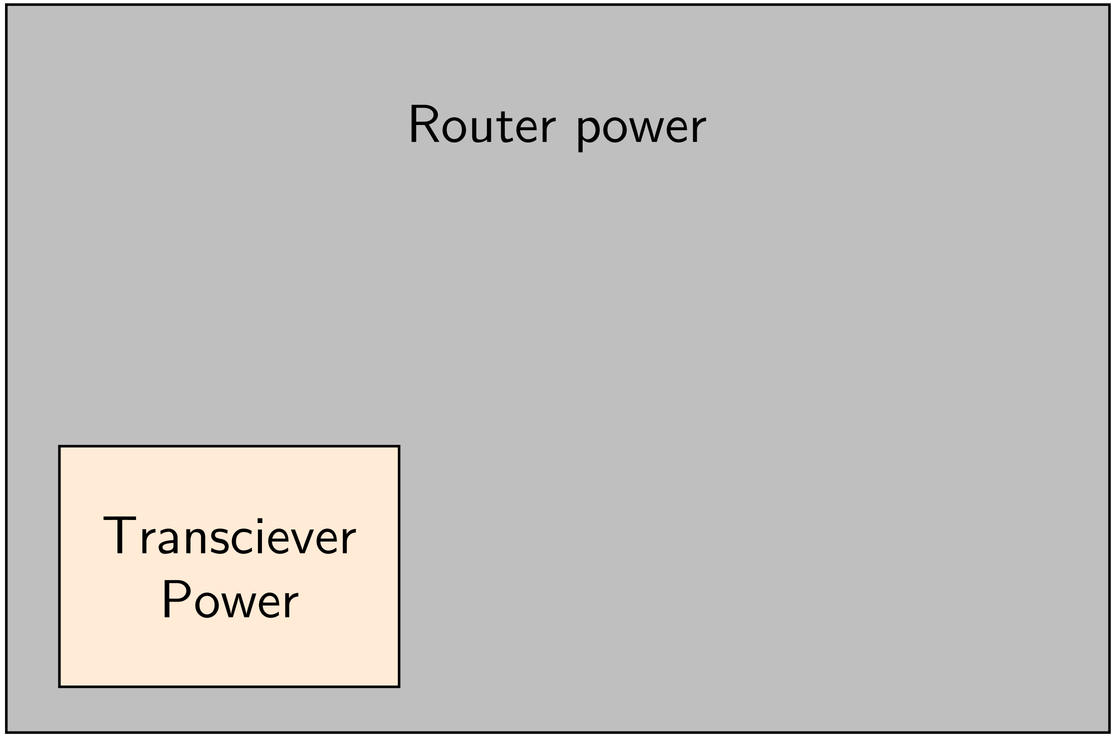
Quite big



Area ~ Power footprint

How big is the transceiver power relative to the total? It depends...

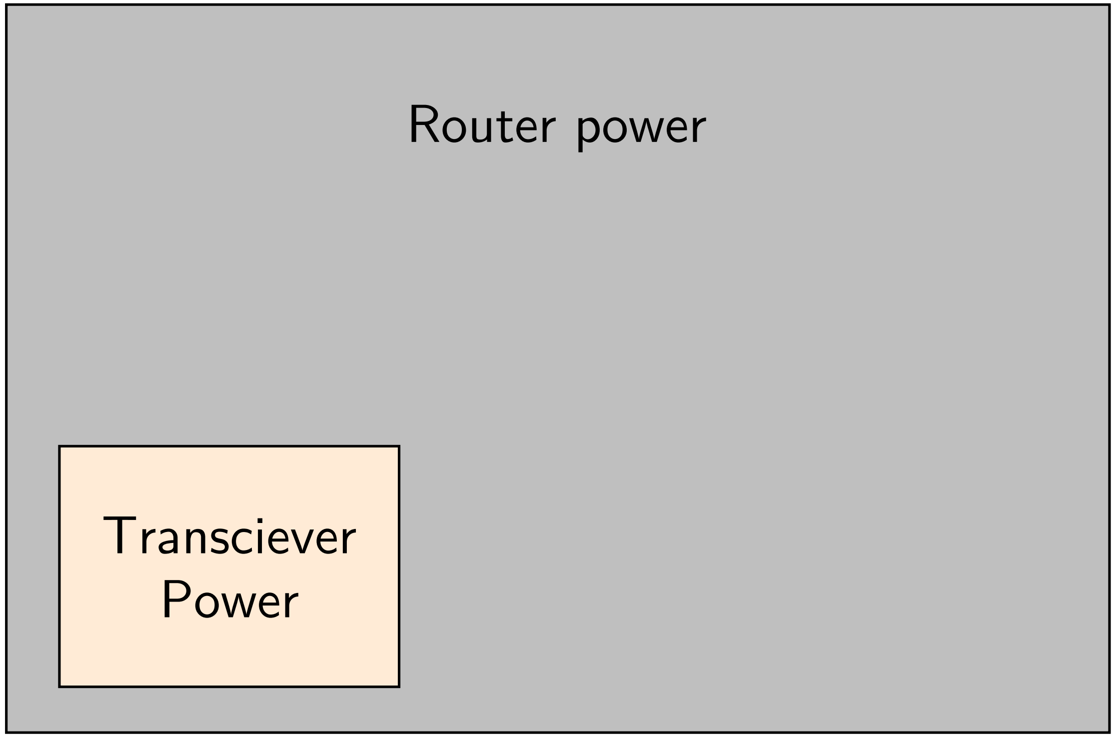
Not so big...



Area ~ Power footprint

How big is the transceiver power relative to the total? It depends...

Not so big...



Area ∼ Power footprint

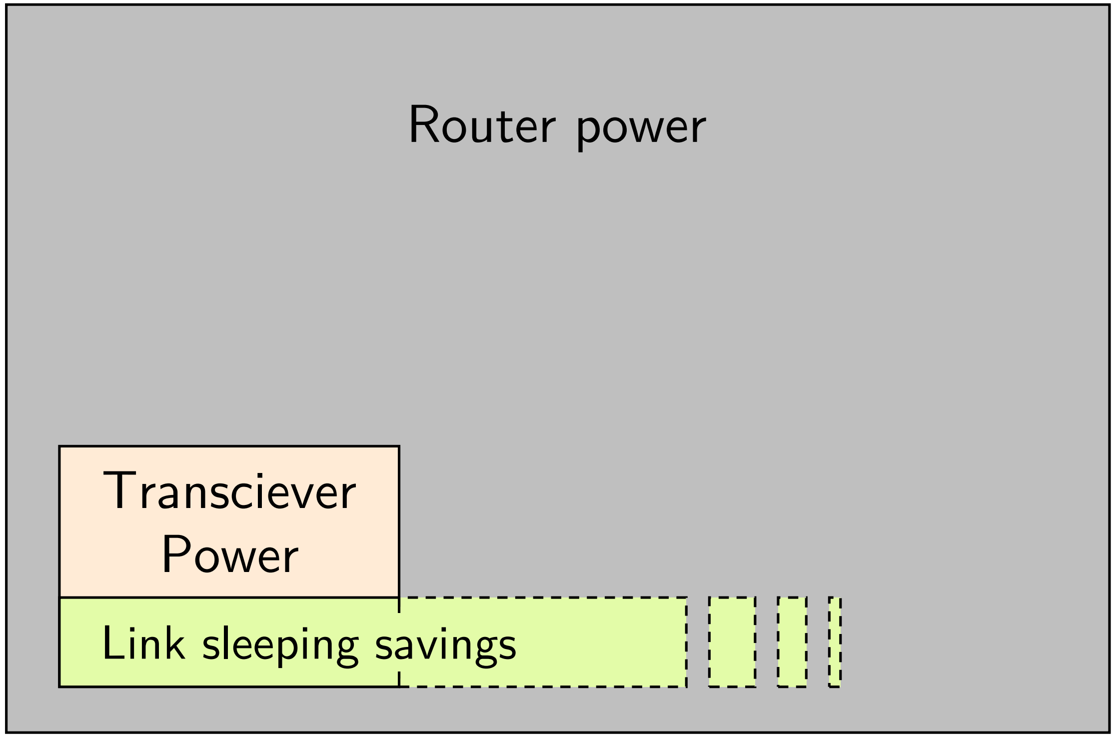
In practice

It is not clear how much power a router draws

- We do not know how large the grey box is.
- We do not know how large the orange box is either...

Link sleeping saves power on the router side too

but it is harder to estimate.



Area ~ Power footprint

In practice

Turning links off reduces power on the router side as well, but we do not know how much.

Quantifying the savings from link sleeping needs more work.

We need

1. **Power data**
   to understand better where power goes
2. **Power models**
   to predict the effects of changes
3. **Testing**
   to validate the effectiveness of solutions

Quantifying the savings from link sleeping needs more work.

We need

1. Power data
   to understand better where power goes
2. Power models
   to predict the effects of changes
3. Testing
   to validate the effectiveness of solutions

Vendors tell you very little about energy consumption.

- Datasheets talk about max/“typical” power
- Devices are never under full load

Vendors tell you very little about energy consumption.

- Datasheets talk about max/“typical” power
- Devices are never under full load

? How much power is drawn in practice?

We need to fix power data transparency!

Most PSUs measure the power they deliver.

but

- The data format is not standard.
- The data is not always available to the user.
- We do not know if the data is trustworthy...

Lots of IETF discussions about those issues right now

The only way to validate PSU data is to measure externally and compare!

Since Jan. 1 2024

Systematic collection of PSU readings from all Switch routers via SNMP

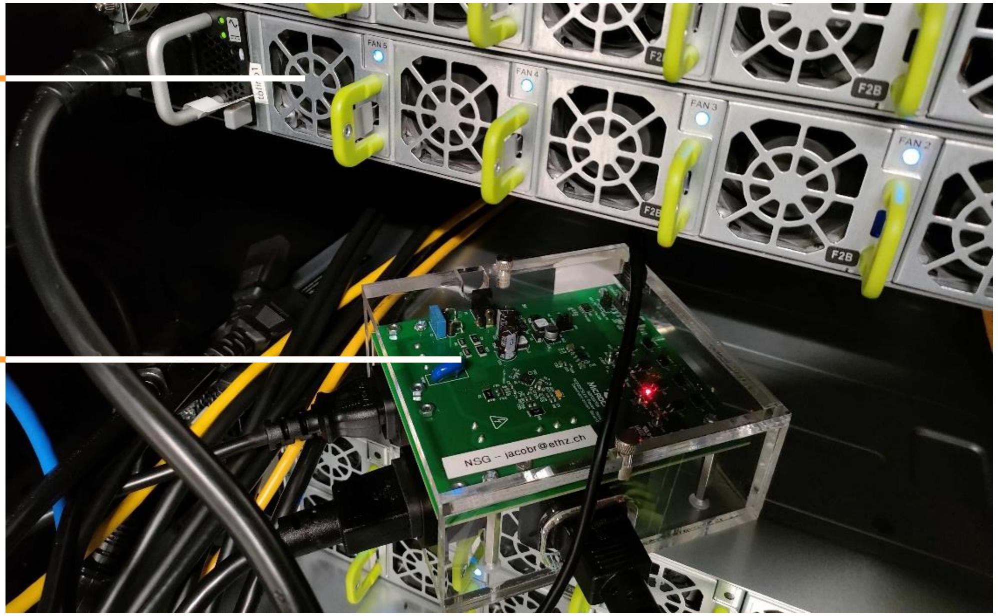
🙏 Many thanks to Simon Leinen!

In parallel

Profiling a various routers and switches

Router

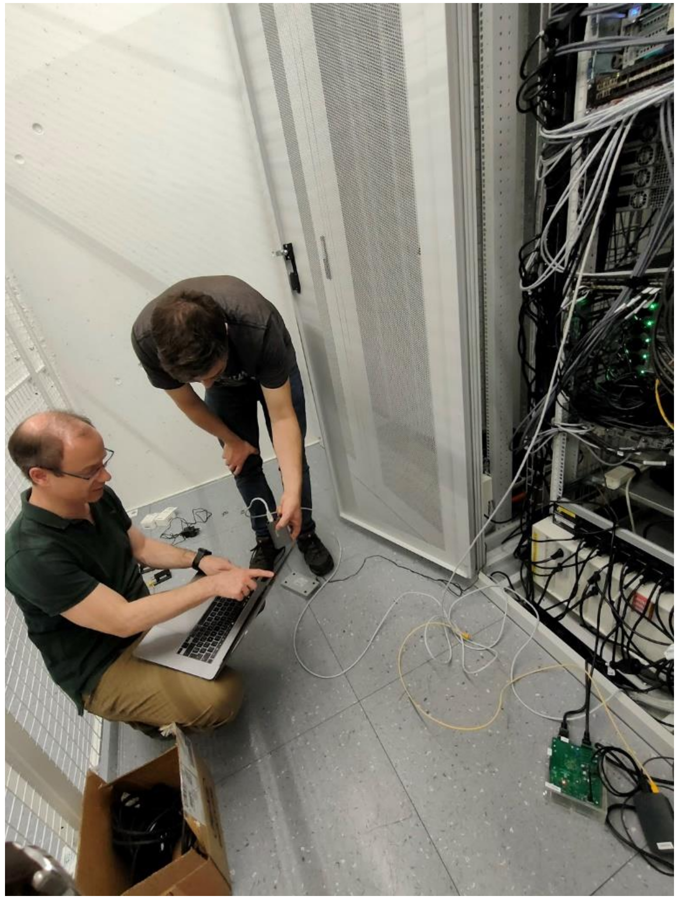
Power meter



The only way to validate PSU data is to measure externally and compare!

... Still WiP ...

Markus Wittmer
Andrea Tognola



In the Switch PoP at ETH

Our objective is to create a public database of power data: powerDB

suggestions for a better name are welcome...

The database contains

- Datasheet information
- PSU readings
- External measurements
- Power models
  More on that one in a second

Would you share your network's data?

We work on tools to make it easy 😉

Quantifying the savings from link sleeping needs more work.

We need

1 Power data
to understand better where power goes

2 Power models
to predict the effects of changes

3 Testing
to validate the effectiveness of solutions

Energy savings are hard to estimate because we lack good power models.

... so we are building our own ...

Device power = Base power

+ Static power per port
+ Energy per packet * packet rate
+ Energy per bit * bit rate

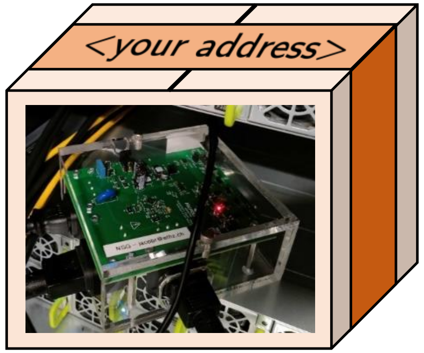
f(device config)

We have power models now.
We need to validate them!

Academics have limited access
to devices used in the field.

Can we measure yours?

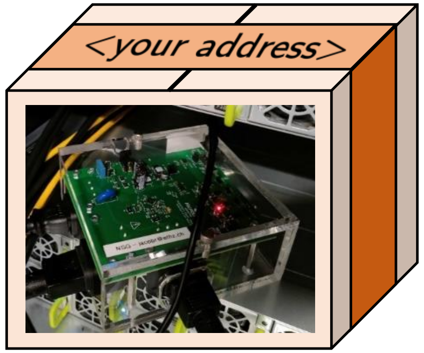
- We sent you hardware
- You plug it in
- Data is pushed into powerDB!



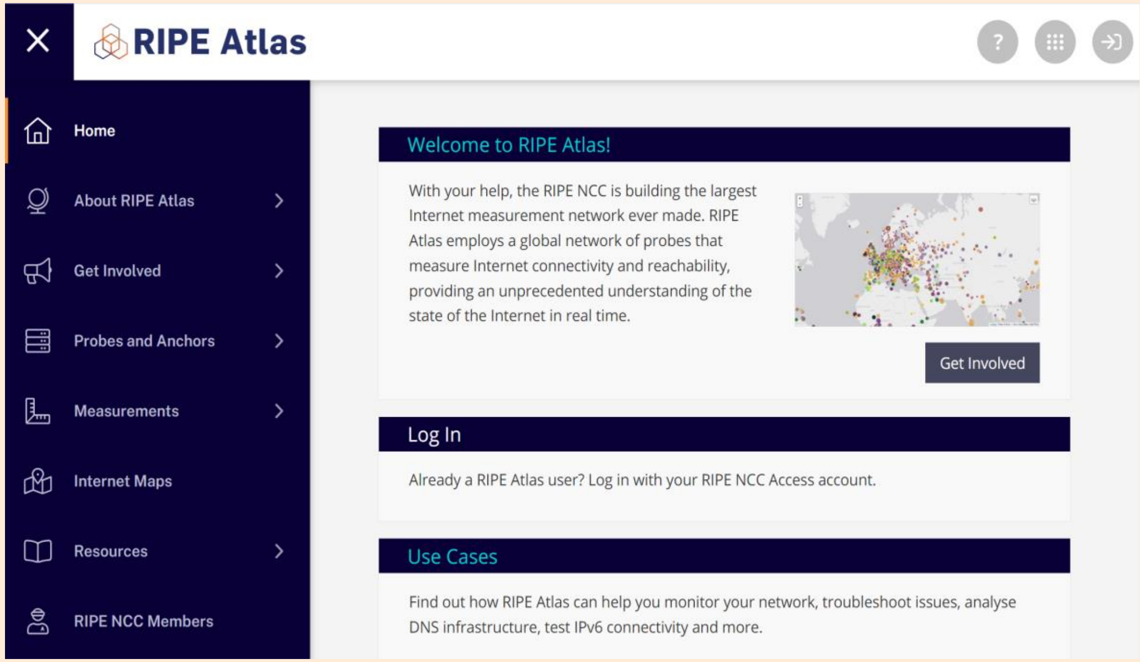
Academics have limited access
to devices used in the field.

? Can we measure yours?

- We sent you hardware
- You plug it in
- Data is pushed into powerDB! 🚀



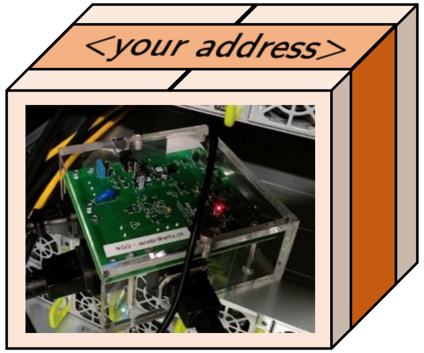
Vision akin to a
RIPE Atlas for Power Data



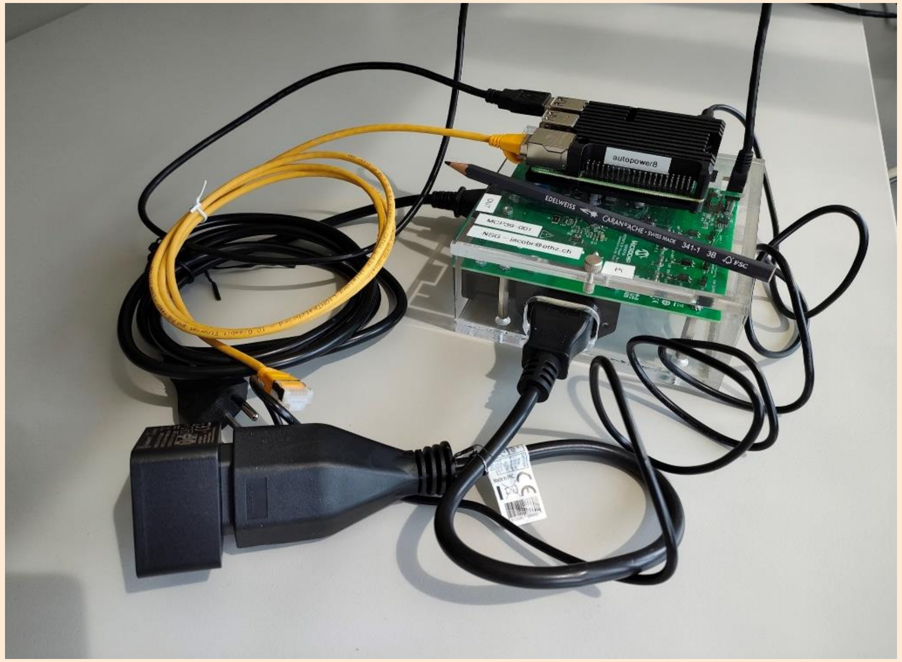
Academics have limited access to devices used in the field.

Can we measure yours?

- We sent you hardware
- You plug it in
- Data is pushed into powerDB!



Our measurement units are ready to go!



Quantifying the savings from link sleeping needs more work.

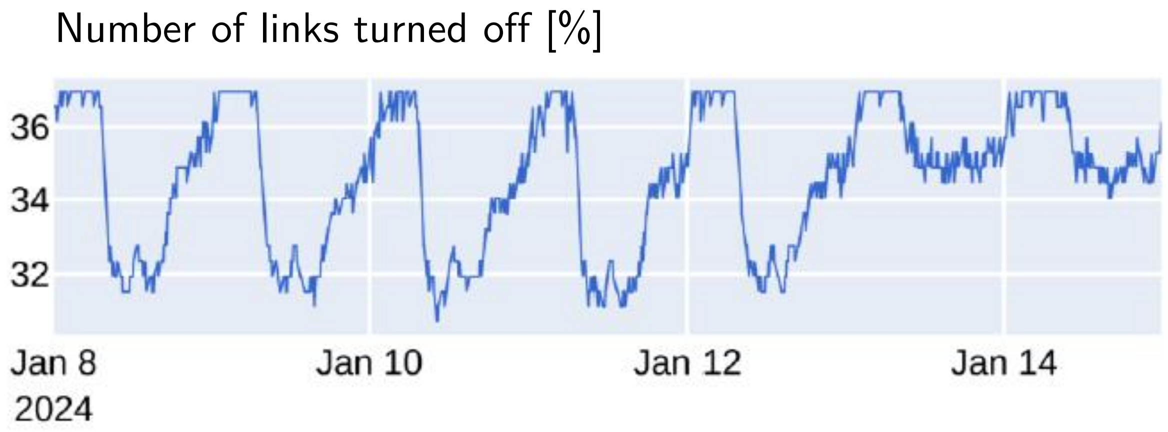
We need

1 Power data
to understand better where power goes

2 Power models
to predict the effects of changes

3 Testing
to validate the effectiveness of solutions

Hypnos evaluation is promising but has important limitations.



- No flow-level data

  We do not know exactly where traffic gets rerouted away from sleeping links.

- No “live” data

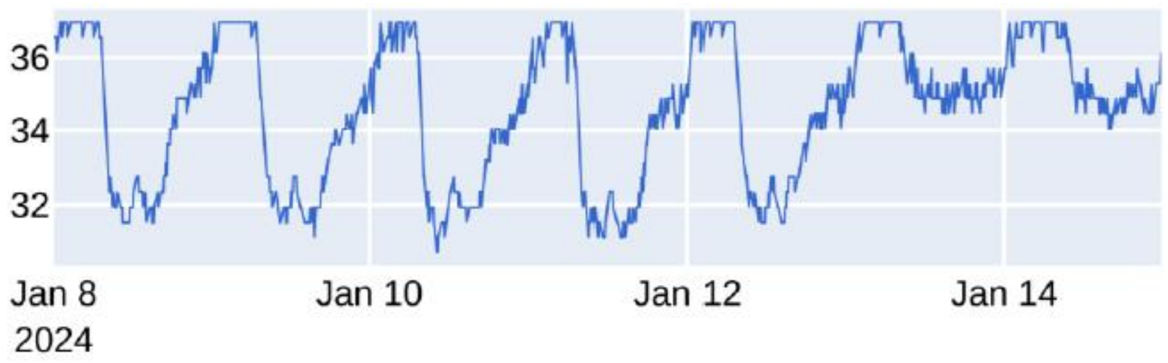
  We only have 5-minute averages on link loads.

- We cannot guarantee that Hypnos would not have created congestion.

- The evidence suggests the risk is very small.

The only way to know if link sleeping works and how much it saves is to try it out.

Number of links turned off [%]



Are you interested?

Simple heuristics appear enough to implement link sleeping in practice.

On Switch LAN, we can

- Turn 1/3 links off
- Avoid congestion

Similar results for the SURF network.

Quantifying the energy savings from link sleeping needs more work.

We need

1 **Power data**
to understand better where power goes

2 **Power models**
to predict the effects of changes

3 **Testing**
to validate the effectiveness of solutions

We need your help
to help your network.

We need your help
to help your network.

We need data.

- Academics have ideas
  sometimes even good ones!
- Operators have power

We need your help
to help your network.

We need data.

- Academics have ideas
  sometimes even good ones!
- Operators have power to pay for every month.

We need your help
to help your network.

We need **data**.

- Academics have ideas
  sometimes even good ones!
- Operators have power to pay for every month.
  to change things in their network.

Let's work together

Yes, we know what NDAs are.

What keeps your network up at night?
A link sleeping study of Switch LAN

Hypnos – Greek god of sleep

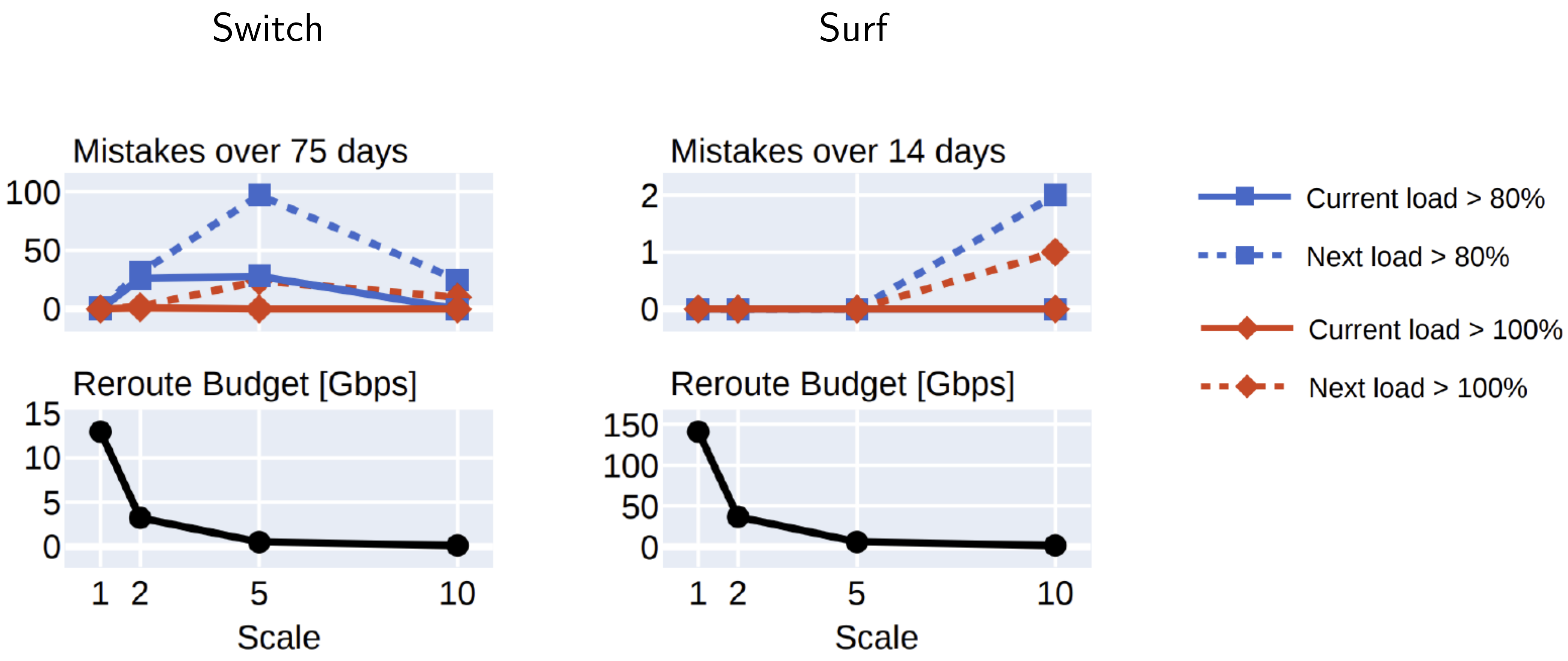
Romain Jacob
jacobr@ethz.ch

Switch Network WG
June 27, 2024

Back up

Savings remain sizable when enforcing a 2-connectedness constraint.

number of links (%)	1-connected	2-connected
ISP 1	85 (36%)	43 (18%)
ISP 2	280 (38%)	52 (7%)


Switch
Surf
Mistakes over 75 days
100
50
0
Mistakes over 14 days
2
1
0
Reroute Budget [Gbps]
15
10
5
0
Reroute Budget [Gbps]
150
100
50
0
1 2 5 10
Scale
1 2 5 10
Scale
Current load > 80%
Next load > 80%
Current load > 100%
Next load > 100%