Green Networks: Reducing the Energy Consumption of Networks

Ken Christensen
Department of Computer Science and Engineering
University of South Florida
Tampa, Florida USA 33620
christen@cse.usf.edu
http://www.csee.usf.edu/~christen

Seminar talk at University of Perugia
May 31, 2010

Thank you – Gratzie

A big thank you to Gianluca Reali for inviting me to give this talk. Thank you to everyone for their wonderful hospitality.

Acknowledging my students…

Some of the work presented here was done by past and present students including,

- Chamara Gunaratne (PhD in 2008)
  - Early Proxying and Ethernet work
- Miguel Jimeno (PhD in 2010)
  - Proxying (especially for applications)
- Mehrgan Mostowfi (MS in 2010, continuing to PhD)
  - Recent Ethernet work

Where do I come from?

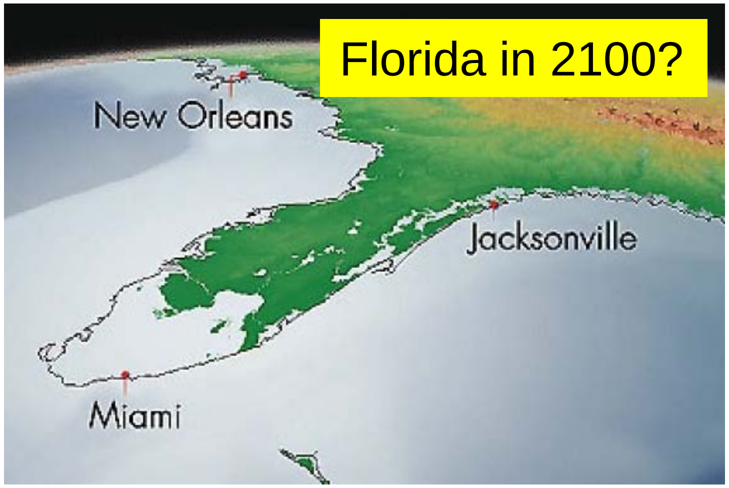
University of South Florida and Tampa

47,000 students
9th largest in the US

http://wwp.greenwichmeantime.com/time-zone/usa/florida/map.htm

Yes, we have lots of alligators

Why Green Networks?



From U.N. Intergovernmental Panel on Climate Change

From http://www.atmos.washington.edu/~bitz/PSC/future.html

One of the most urgent challenges of the 21st century is to investigate new technologies that can enable a transition towards a more sustainable society with a reduced CO_2 footprint.

One way to be "green"…

Just have less and do less

- No houses, no cars, no travel, no PCs, no Internet, etc.

North Korea at night.
A model green society?
I don't think so…

From http://strangemaps.wordpress.com/2007/12/16/218-koreas-dark-half/

Notion of comfortable conservation

"I mean using less energy for identical performance, measured in whatever way the consumer wishes."

- Richard Muller (*Physics for Future Presidents*, 2008)

In network speak, same QoS for less energy

Product lifecycle and green

Lifecycle of "stuff" (including ICT equipment)

Focus of this talk

Production → Use → End-of-use

Production
- Cleaner mining
- Cleaner manufacturing
- Use less toxic materials
- Use less materials overall
- Use less energy overall

Use
- Use less energy
- Extend lifetime

End-of-use
- Recycle materials
- Refurbish for reuse

1 kWh = $0.10

Energy consumed by a PC*
- Production = 2000 KWh
- Life (5 yrs) = 4200 KWh

* E. Williams, "Revisiting Energy Used to Manufacture a Desktop Computer: Hybrid Analysis Combing Process and Economic Input-Output Methods," *Proceedings of IEEE International Symposium on Electronics and the Environment*, pp. 80-85, 2004.

Roadmap of this talk

This talk has three major topics

- Briefly quantifying energy use of ICT
- Reducing direct energy consumption for Ethernet
- Future challenges

A fourth topic if time permits is

- Reducing induced energy consumption

Key definitions

Direct energy use

- Energy used by network links and equipment, but not end devices

Induced energy use

- Incremental additional energy used for a higher power state of end devices needed to maintain network connectivity

Quantifying the energy use of ICT

How much energy does ICT use?

... the Internet is part of this.

A quick look at energy costs

In the USA

Recall that power is W and energy is Wh

- 1 kWh is about $0.10 (in the US)
- 1 TWh is about $100 million
- 1 W for 1 year is about $1 (actually, it is $0.88)

St Lucie, Florida
Twin nuclear units
About 11 TWh/year

A quick look at energy costs <u>continued</u>

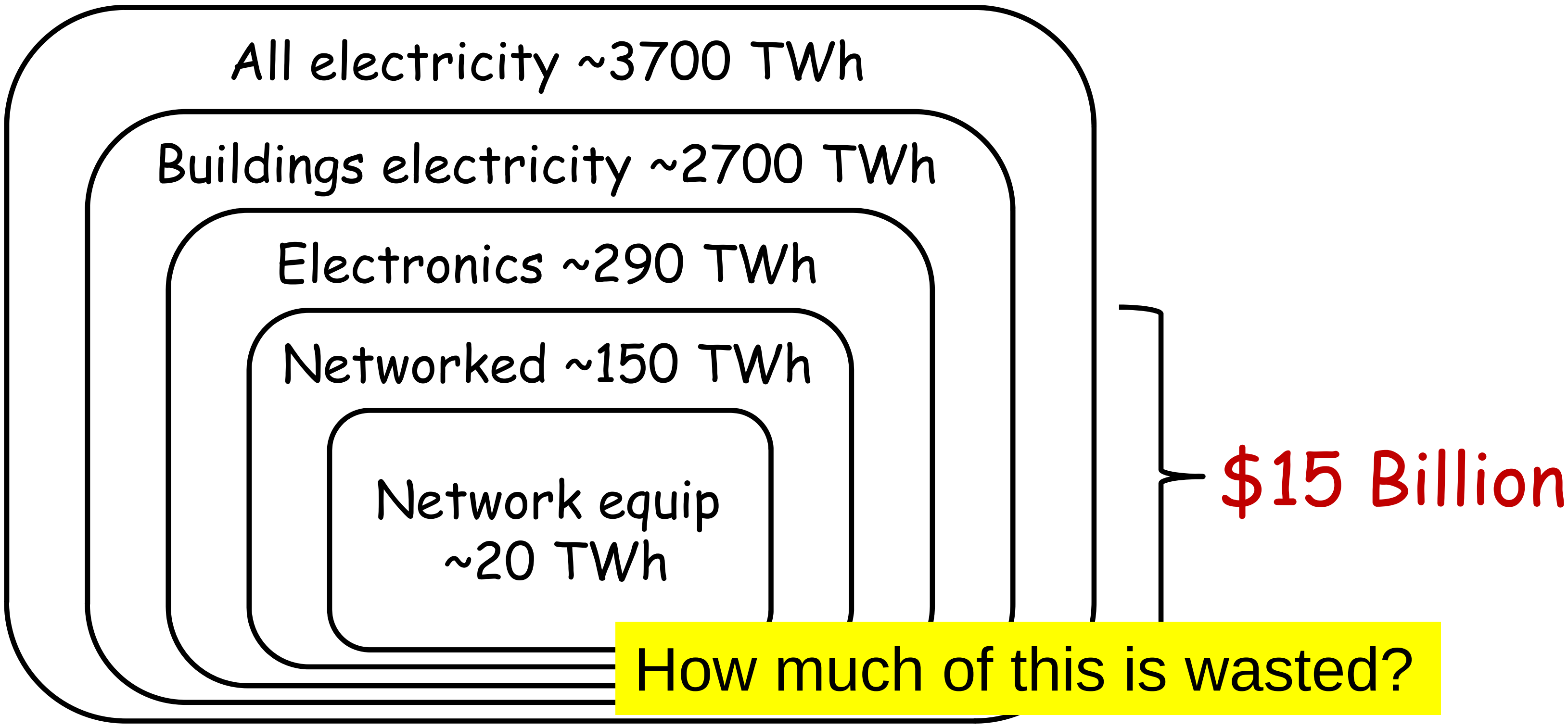
In Italy

- 1 kWh is about €0.085 (for typical a residence)
- 1 TWh is €85 million
- 1 W for 1 year is about €0.75

Larderello, PI
Geothermal
About 4.8 TWh/year

Electricity use – big picture

Electricity use in the USA (2006, from LBNL)

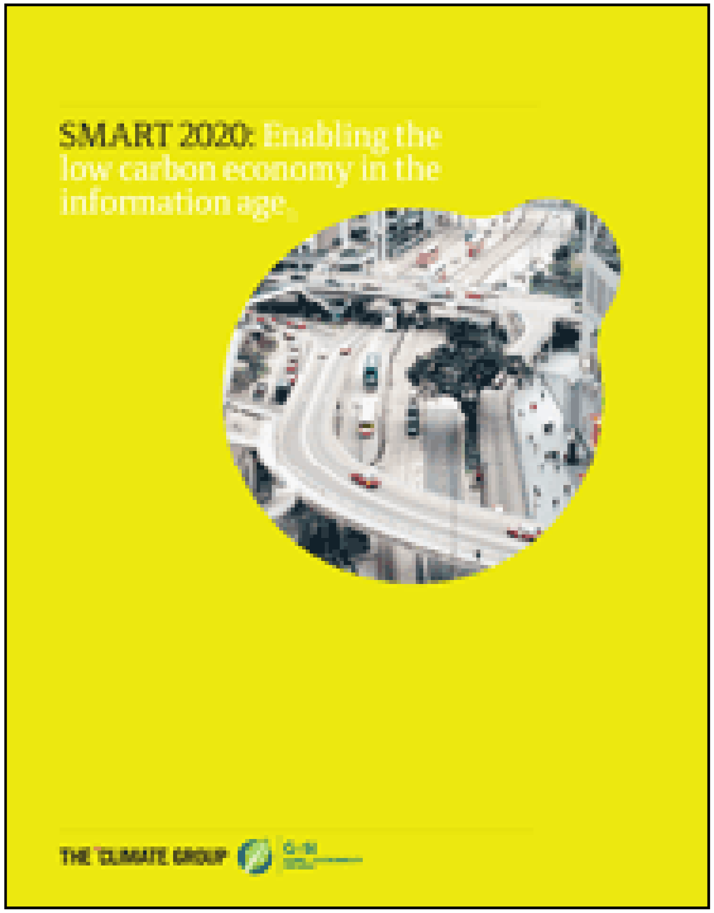


How much of this is wasted?

How much can be saved?

A view from the Climate Group

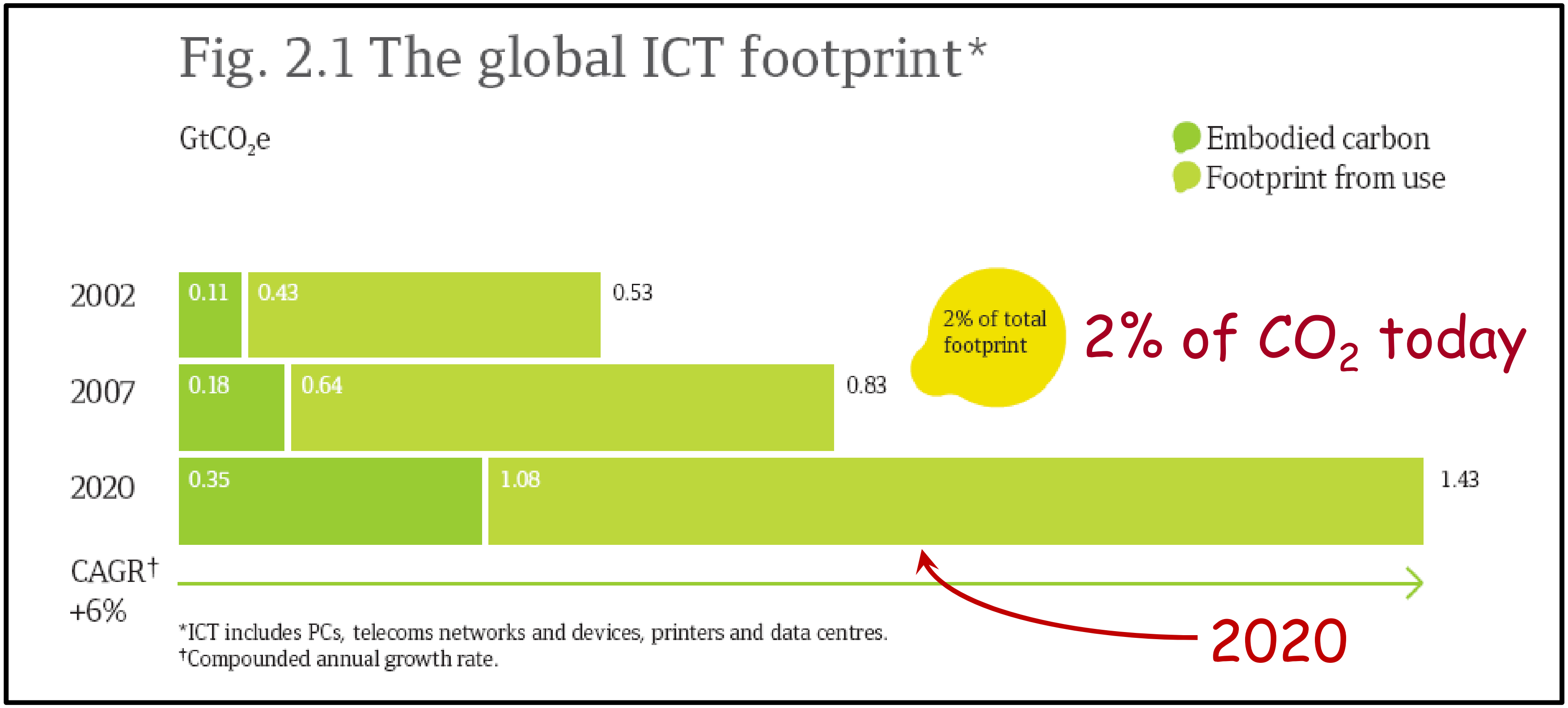
The SMART 2020 report



- Focus is on ICT's role in reducing greenhouse gases
  - Both of and by ICT
- A view of the world in 2020
  - Taking into account "likely" technology developments
- Supporting organizations
  - Include Cisco, Intel, HP, Sun, national telecoms, and telecom operators

Global ICT CO_2 footprint

Today ICT is 2% of global CO_2



From SMART 2020 report

ICT CO_2 > Aviation CO_2

"The global information and communications technology (ICT) industry accounts for approximately 2 percent of global carbon dioxide (CO_2) emissions, a figure equivalent to aviation."
- Gartner Group, Inc. (2007)

- ICT use growing faster than airline traffic
- Greater impact by "fixing" ICT than airplanes

ICT energy use – the PC

The end user PC is the biggest energy consumer

"Desktop computing accounts for 45 percent of global carbon emissions from information technology."
- govtech.com

"Most PC energy use in the US occurs when no one is there, and this is greater than the total energy use of all network equipment."
- Bruce Nordman (LBNL)

Network energy use in Italy

Statistics for Italy

17.5 million broadband users, population of Italy is 60 million

TABLE I.
(A) 2015-2020 NETWORK FORECAST: DEVICE DENSITY AND ENERGY REQUIREMENTS IN THE BUSINESS-AS-USUAL CASE (BAU). EXAMPLE BASED ON THE ITALIAN NETWORK.

	power consumption [W]	*number of devices* [#]	*overall consumption* [GWh/year]
Home	10	17,500,000	1,533
Access	1,280	27,344	307
Metro/Transport	6,000	1,750	92
Core	10,000	175	15
		Overall network consumption	1,947

From: R. Bolla, R Bruschi, K. Christensen, F. Cucchietti, F. Davoli, and S. Singh, "The Potential Impact of Green Technologies in Next Generation Wireline Networks – Is There Room for Energy Savings Optimization?", submitted to *IEEE Communications*.

Network energy use in Italy <u>continued</u>

Another statistic for Italy…

Today, the energy needed by the Telecom Italia's Network

is more than **2.000.000.000.000Wh** (>2TWh)

representing nearly 1% of the total National energy demand,

second user only to the National Railways

From: F. Cuccheietti, "Energy Efficency – An Enabler for the Next Generation Network," Presentation by Telecomm Italia, Bruxelles, January, 30 2006.

Reducing direct energy consumption

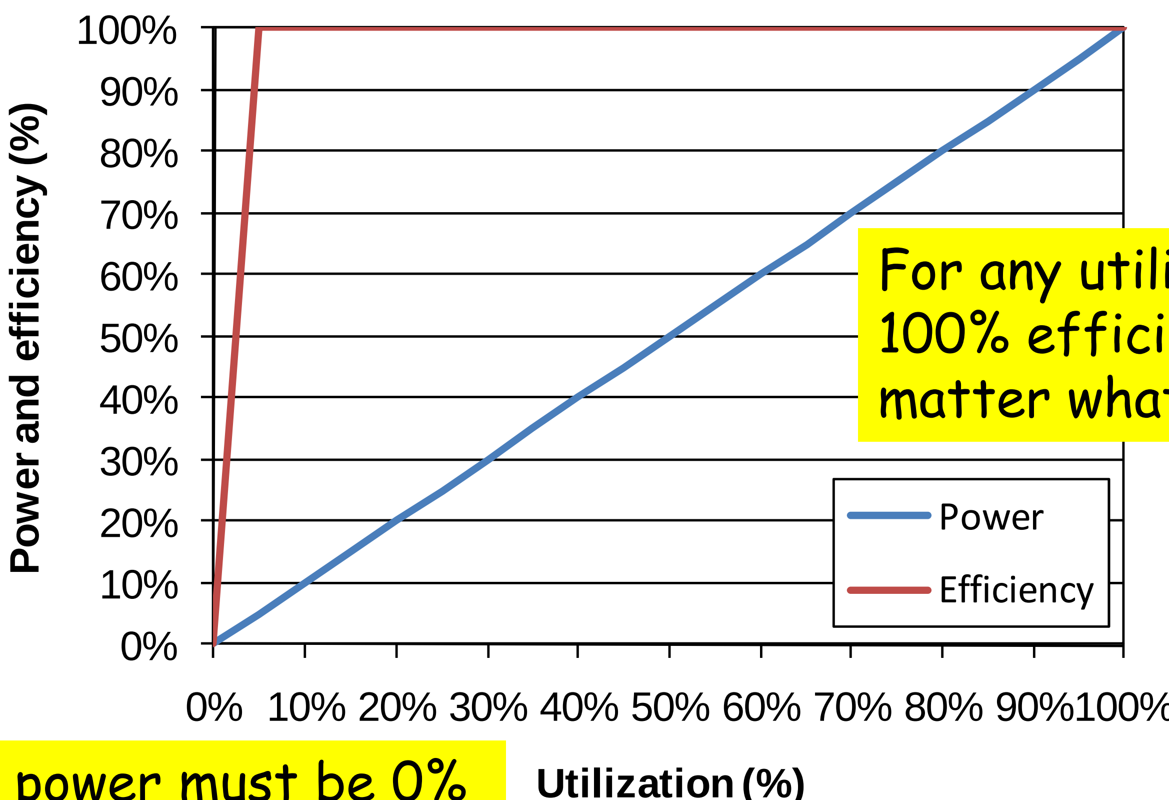
Can we reduce energy used by Ethernet?

... this is Energy Efficient Ethernet

The goal is energy-proportional

We seek energy-proportional computing

- Define efficiency as power divided by utilization



For any utilization you have 100% efficiency. It doesn't matter what utilization you run.

Idle power must be 0%

Adapting link data rate to load

Moving toward energy-proportional links

- Links are typically lightly utilized and will stay that way
  - See Odlyzko and others
- When link utilization is low, do not need "high bandwidth"
- Lower data rates consume less power
- Idea is to explore if and how links could adapt their data rate to load
  - High data rate for high load
  - Low data rate for low load (most of the time!)

Open questions in adapting to load

There are many open questions

- What is the *mechanism* for adapting to load?
  - How is the link data rate changed
- What is the *policy* for adapting to load?
  - When is the link data rate changed
- What about the delay and loss for switching between rates?
- What about oscillation – is it stable?
- Fundamentally, what is the trade-off between energy savings and performance?

Adaptive link rate (ALR) for Ethernet

Goal: **Save energy by matching link data rate to utilization**

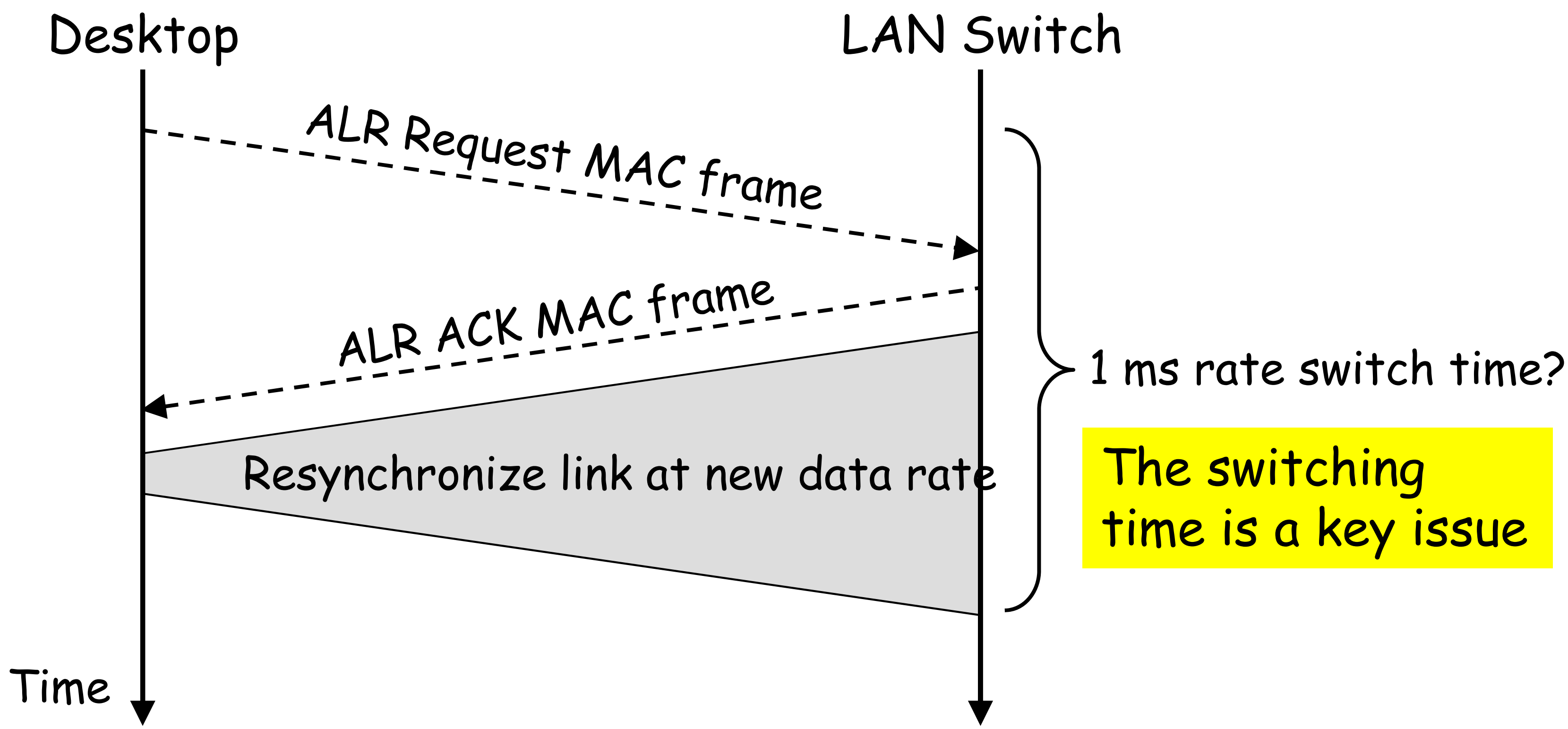
- **Change (adapt) data rate in response to utilization**
  - Use 10 or 100 Mb/s during low utilization periods
  - Use 1 or 10 Gb/s during high utilization periods
- **Need new *mechanism***
  - Current auto-negotiation is not suitable (100s of ms)
- **Need *policies* for use of mechanism**
  - *Reactive policy* possible if can switch link rates "quickly"
  - *Predictive policy* is needed otherwise

Independent of PC power management

One possible ALR mechanism

Use a MAC frame handshake between ends

ALR must be supported in both ends

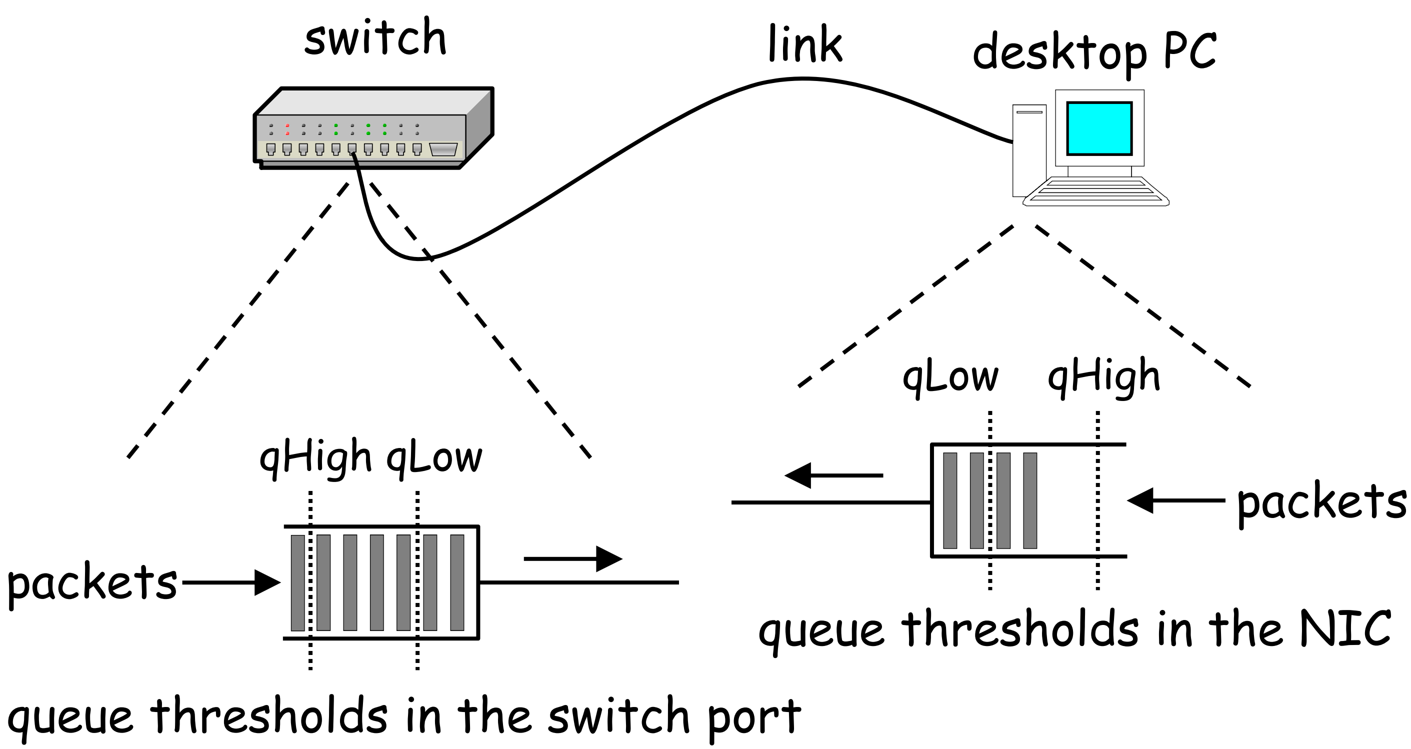


The switching time is a key issue

One possible ALR policy

Dual-threshold policy

- If queue is above qHigh then switch to high rate
- If queue is below qLow then switch to low rate



A lot of work done with ALR

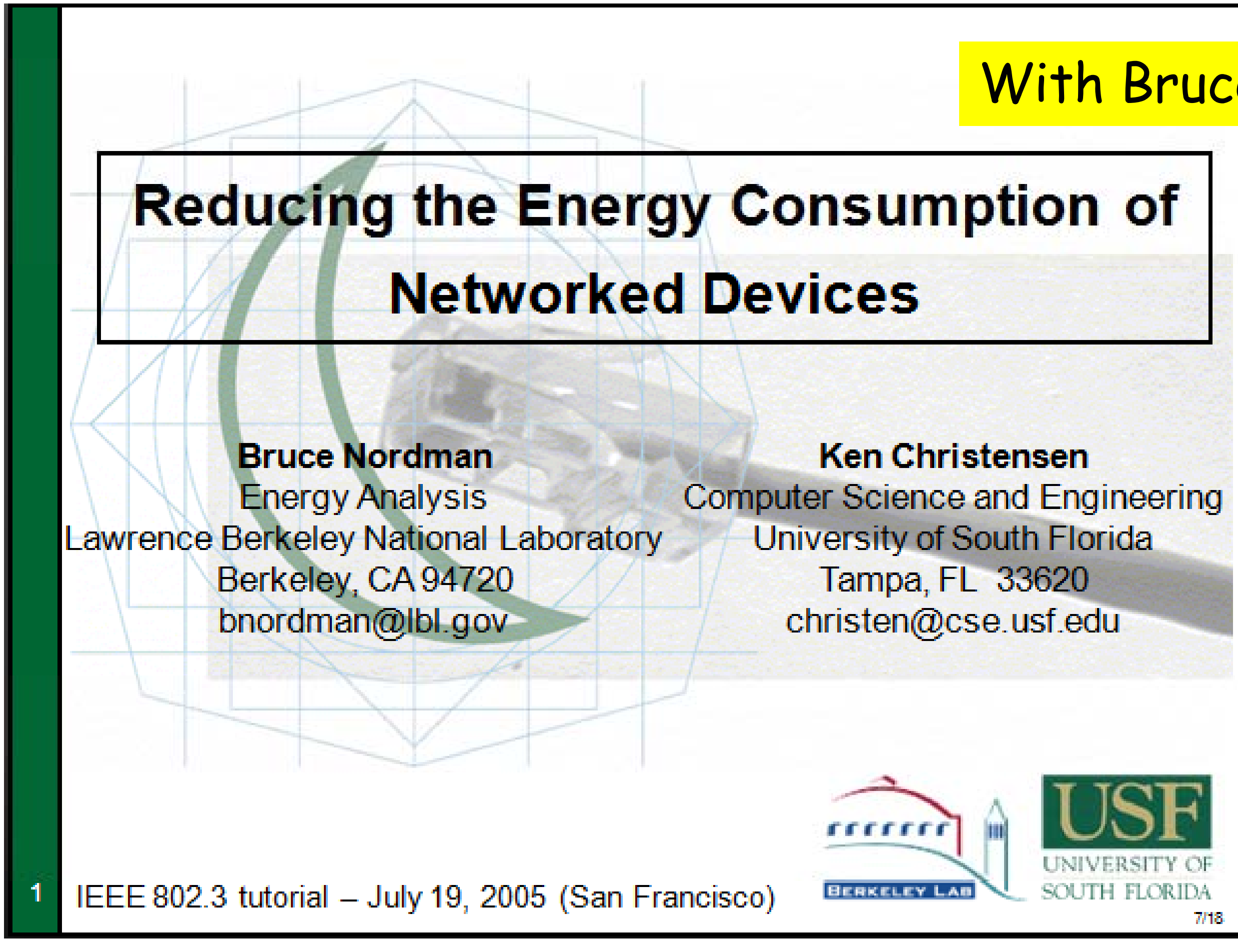
We did a lot of work with ALR

- Studied performance of ALR policies
- Effect of switching time studied
- Simulation and analytical models built
- Published findings in *IEEE Transactions on Computers*
- However, ALR was not adopted by IEEE 802.3
  - Issues with switching time
  - Issues with complexity of a mechanism

ALR and IEEE 802.3

ALR presented to IEEE 802.3 in July 2005

With Bruce Nordman



ALR and IEEE 802.3 <u>continued</u>

- **Adaptive Link Rate to IEEE 802.3 in 2005**
  - A Study Group was formed
  - Mike Bennett from LBNL is the chair
- **Became "Energy Efficient Ethernet"***
  - IEEE 802.3az task force
- **ALR became RPS, which then became LPI**
- **Standard expected to be approved in late 2010**
- **Vendors are sampling products already**
  - Broadcom and Realtek

* Logo by Glen Kramer of Teknovus, Inc. (full permission for use granted via email dated January 27, 2007)

EEE in EPA Energy Star

EPA Energy Star for Computer Servers, Tier 2

- "Energy Efficient Ethernet: All physical layer Ethernet in servers covered by the Computer Server specification must meet the Energy Efficient Ethernet (IEEE 802.3az) standard upon its approval by the IEEE."*

ENERGY STAR

To be in computer (PC) spec later

* From ENERGY STAR® Version 1.0 Program Requirements for Computer Servers, Tier 2: PRELIMINARY

Complexity of ALR handshake

Summary slide from a presentation by Dave Law

> ### Summary
>
> - Lost speed change REQ, ACK or NACK
>   - Very rare event
>   - But need to handle with minimum disruption
> - Do we really need a ACK
>   - Let the link drop be the ACK
> - Timeout, repeat N times or keep repeating
>   - REQ packets
>   - Resend, sending nothing otherwise
>   - NACK packets
>   - Send once then if REQs continue send again
>   - Minimum loss of bandwidth for NACKer
>
> IEEE 802.3 EEE SG – September 2007 Interim Meeting — Page 12

Looks like we need a complicated protocol.

From D. Law, "Packet loss in protocol based speed change," September 2007.

An entirely new approach

IEEE 802.3az opened-up possibility of PHY change

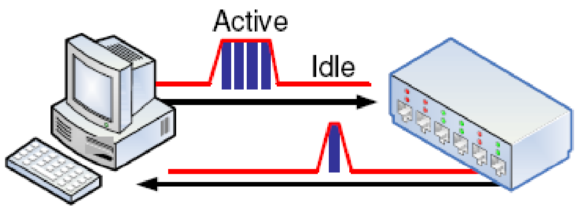
- The open issues with switching rates lead to the possibility of changing the PHY
- Would likely have to change the PHY in any case to enable fast switching (the 10 Gb/s link training issue)

EEE is based on Low Power Idle (LPI)

Slide from November 2007 IEEE 802.3az meeting…

7

Active/Idle Toggling with 0BASE-x Concept



The better idea

- Principle: Transmit data at fastest rate then return to idle
  - Energy savings come from power cycling between active/idle states
- Active/Idle toggling could be used *instead* of PHY rate shifting
  - Offers the best energy efficiency on links with lower utilization
  - Integrates well with existing PC power management schemes (e.g. ACPI)
  - Clock & power gating (on/off) is easier than rate shifting
- Asymmetrical operation would provide even better energy efficiency
  - Each direction could enter active & idle states independently
  - Most end-node traffic is heavily weighted toward either send or receive
  - Tx & Rx data paths already operate independently above the PHY

Energy Efficient Ethernet

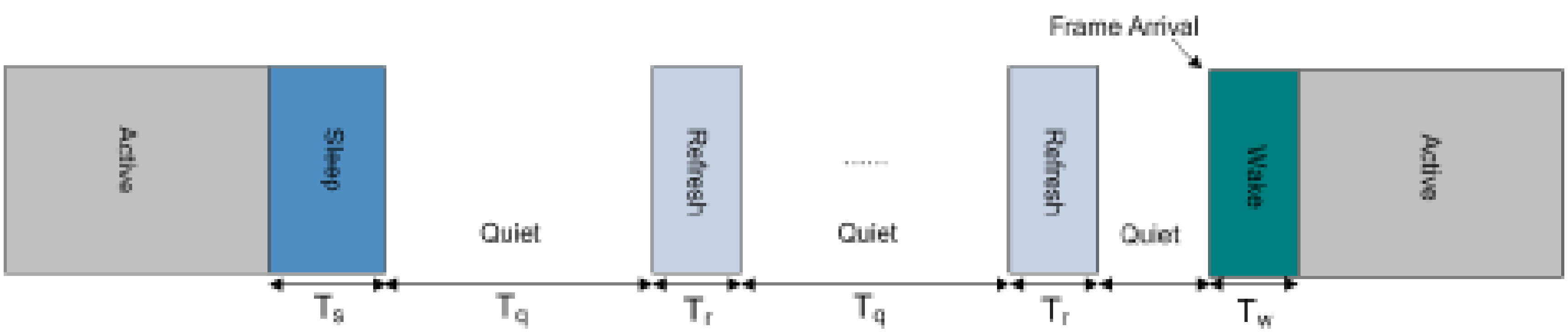
intel

Seminar talk at University of Perugia
May 31, 2010

USF UNIVERSITY OF SOUTH FLORIDA

How LPI works

- **Between packets the PHY "goes to sleep"**
  - Sleep is idle = about 10% of full power
  - Periodic refreshes to keep synchronized
- **LPI has wake-up and sleep transitions**
  - First packet after an idle incurs a wake-up transition
  - After last packet in a burst a go to sleep transition



LPI overhead

LPI has overhead from T_w and T_s

- Can measure frame efficiency for single packet case

$$Efficiency = \frac{T_{Frame}}{T_{Frame} + T_w + T_s}$$

Protocol	Min T_w (μs)	Min T_s (μs)	T_{Frame} (1500B) (μs)	Frame eff.	T_{Frame} (150B) (μs)	Frame eff.
100Base-Tx	30	100	120	48%	12	8.5%
1000Base-T	16	182	12	5.7%	1.2	0.6%
10GBase-T	4.16	2.88	1.2	14.6%	0.12	1.7%

Performance evaluation of EEE

IEEE COMMUNICATIONS LETTERS, VOL. 13, NO. 9, SEPTEMBER 2009 1

Performance Evaluation of Energy Efficient Ethernet

P. Reviriego, J. A. Hernández, D. Larrabeiti, and J. A. Maestro

- The first published work on EEE performance evaluation
- "The results show that although EEE improves the energy efficiency, there is still potential for substantial further energy savings as in many cases most of the energy is wasted in waking up and sleeping the link."

Reviriego et al.

Define key time periods from IEEE 802.3az

- T_s = Time to enter low power mode (goto sleep)
- T_w = Time to exist low power mode (wake-up)
- T_q = Time spend in quiet (low power mode)
- T_r = Refresh time to periodically align both ends
- T_{frame} = Time to transmit a frame itself

- Thus, the total time for transmission of one frame is

$$T_w + T_{frame} + T_s$$

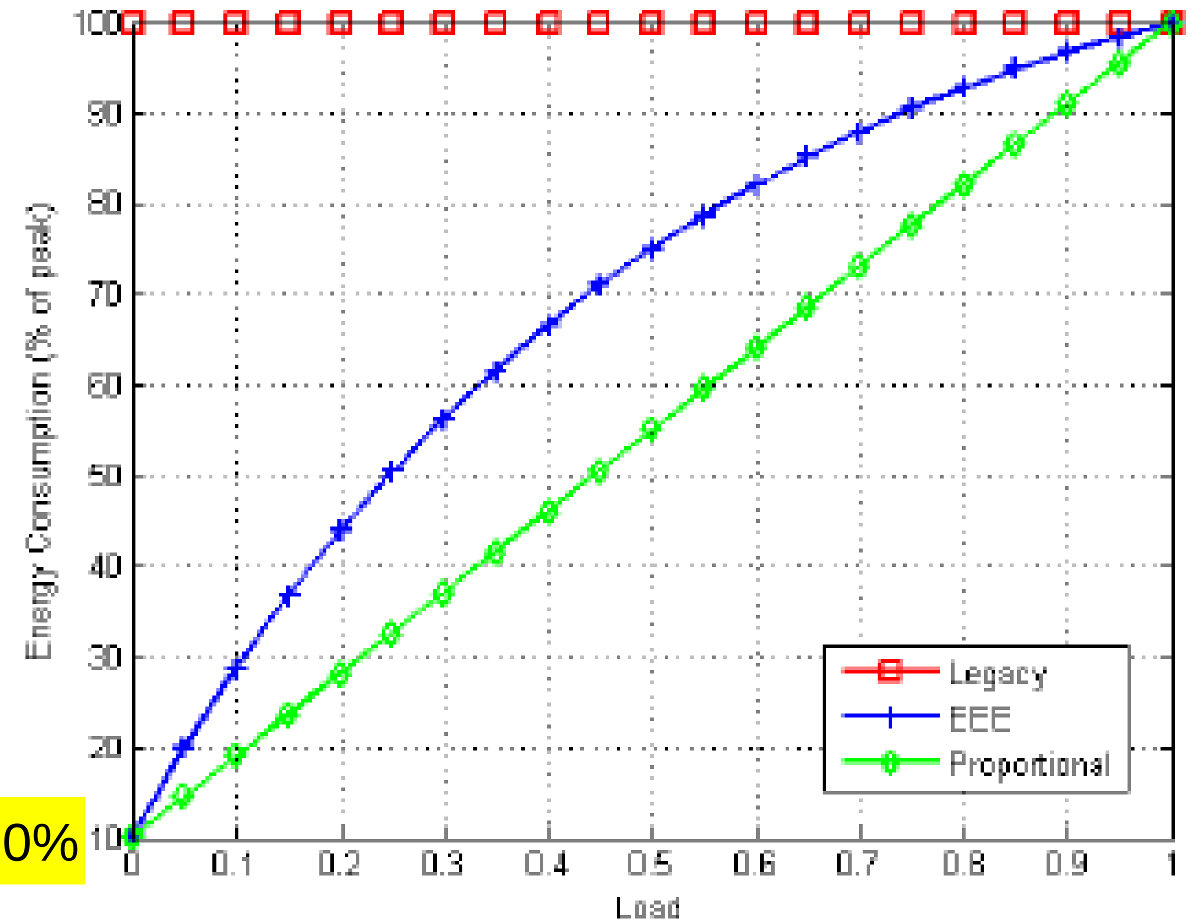
Reviriego et al. continued

Simulation model to study EEE overhead

- A MatLab script
- Poisson arrivals
- Fixed length frames (1250 bytes)
- Assumed that LPI power is 10% of active power
- Studied power consumption as a function of utilization
  - For 100 Mb/s, 1 Gb/s, and 10 Gb/s
  - Key parameter values taken from standard

Reviriego et al. continued

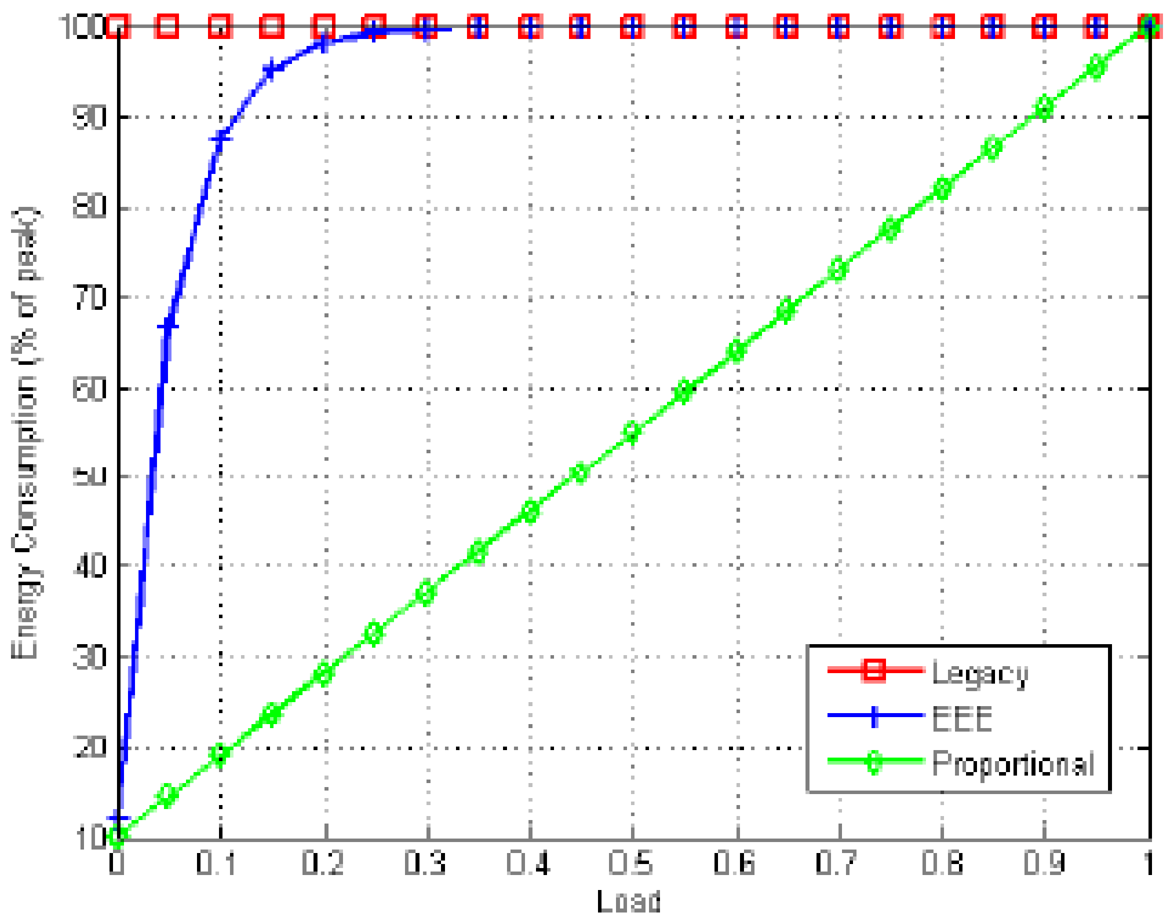
Results for 100 Mb/s



Base is 10%

Reviriego et al. continued

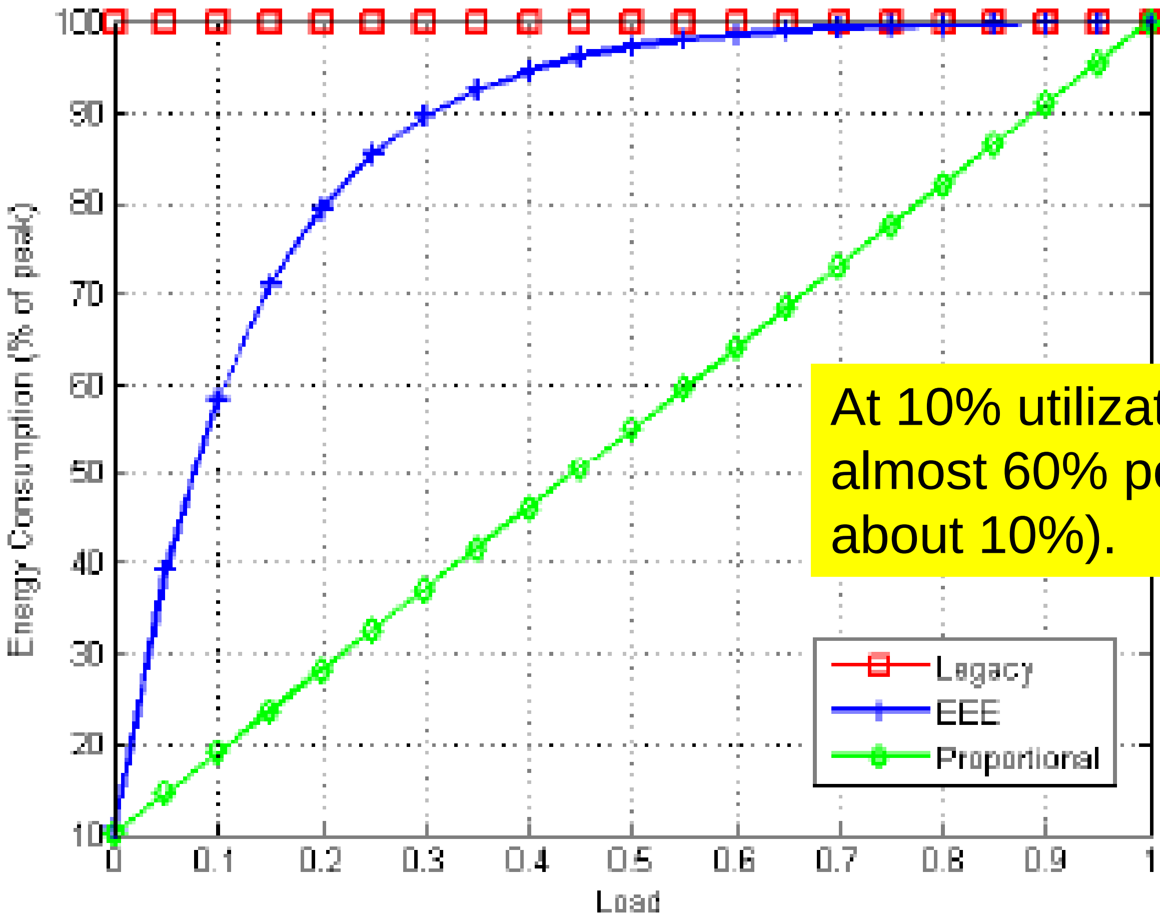
Results for 1 Gb/s



Reviriego et al. continued

Results for 10 Gb/s

Probably of most interest



At 10% utilization consumes almost 60% power (ideal is about 10%).

Reviriego et al. continued

Summary of Reviriego et al.

- EEE can have large deviations from proportional (ideal)
- Energy efficiency of EEE is poor for small frames
- "… recommend frame scheduling algorithms that maximize the efficacy of EEE"
  - Group frames before waking-up link to minimize overhead
- Needs further consideration

More EEE performance evaluation

IEEE 802.3az: The Road to Energy Efficient Ethernet

Authors: K. Christensen, P. Reviriego, B. Nordman, M. Bennett, M. Mostowfi, and J.A. Maestro

- Submitted to IEEE Communications magazine in March 2010
  - For their special issue on Green Communications
- CSIM models by me (and student, Mehrgan Mostowfi) and ns-2 models by Pedro Reviriego and Juan Maestro
- Energy savings from Bruce Nordman
- History of IEEE 802.3az from Mike Bennett (chair task force)

CSIM EEE model

Developed key models in CSIM

- CSIM is a process oriented simulation engine
  - A C function library
  - From Mesquite Software

- Simple single-server queue model with EEE added
  - Customers have deterministic service time

- Adds a T_WAKE delay for first packet to leave queue

- Adds a T_SLEEP delay for last packet to leave queue
  - "last packet" means queue is now empty

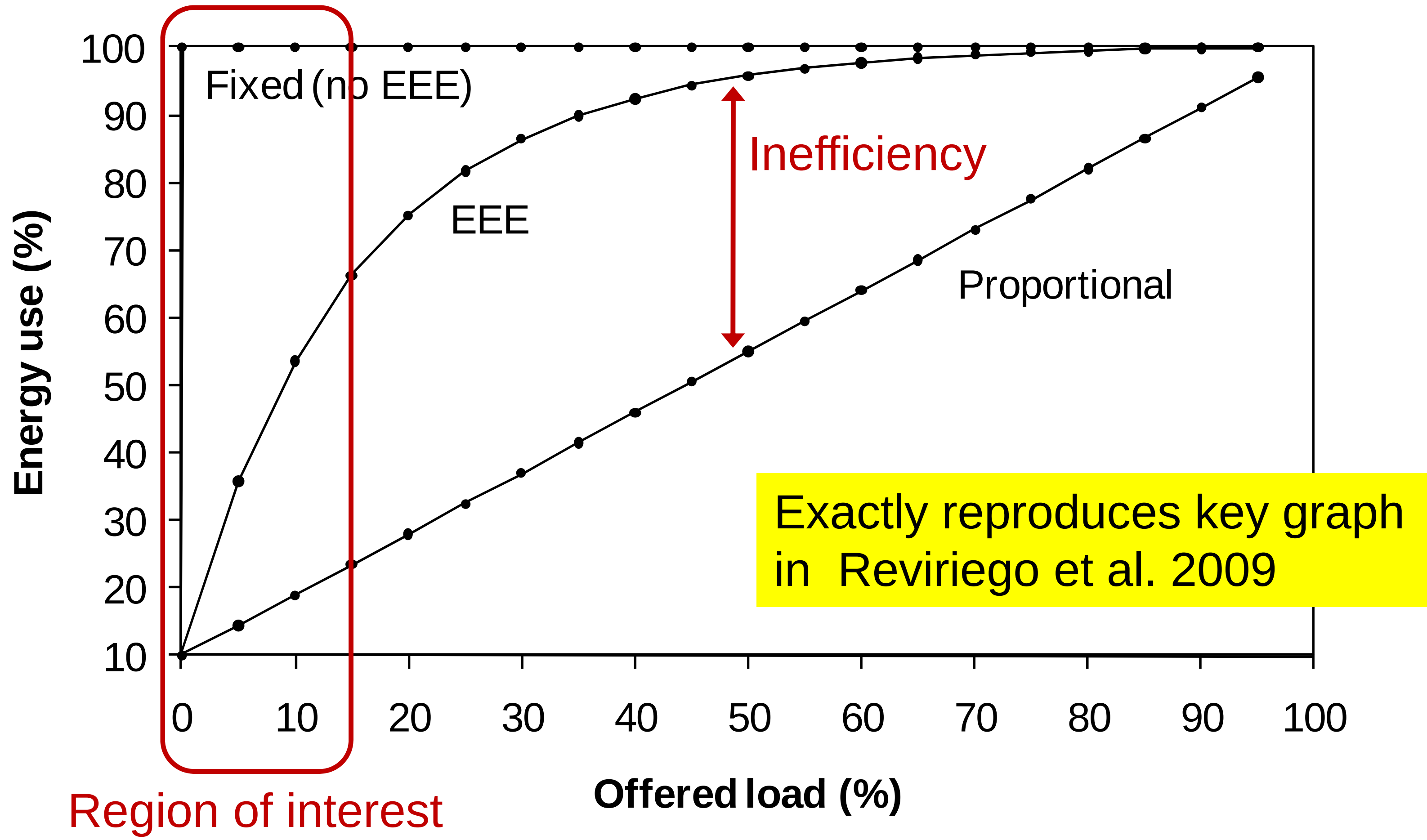
EEE model experiment

Ran an experiment for 10 Gb/s

- For 10 Gb/s
  - T_WAKE = 4.16 μs
  - T_SLEEP = 2.88 μs
  - For 1500 byte packet service_time = 1.2 μs
- Assume that idle power use is 10% of full power use
- Vary offered load from 0% to 95%
  - Poisson arrivals, fixed length packet
- Measure link utilization
  - Note that link utilization will be greater than offered load due to EE overhead

EEE model results

Results for 10 Gb/s



Need to "fix" inefficiency

Idea – packet coalescing to improve efficiency

- Coalescing will reduce EEE overhead
  - More packets per T_WAKE and T_SLEEP overhead
- Trade-offs are
  - Added packet delay
  - Increased burstiness of departure process

FSM for coalescing

Specify coalescing operation with an FSM

- The FSM has two states: ON and OFF
  - In OFF state generated packets are buffered, but not sent
  - In ON state packets are sent
    - » Packets in buffer at time of entry into state are sent first

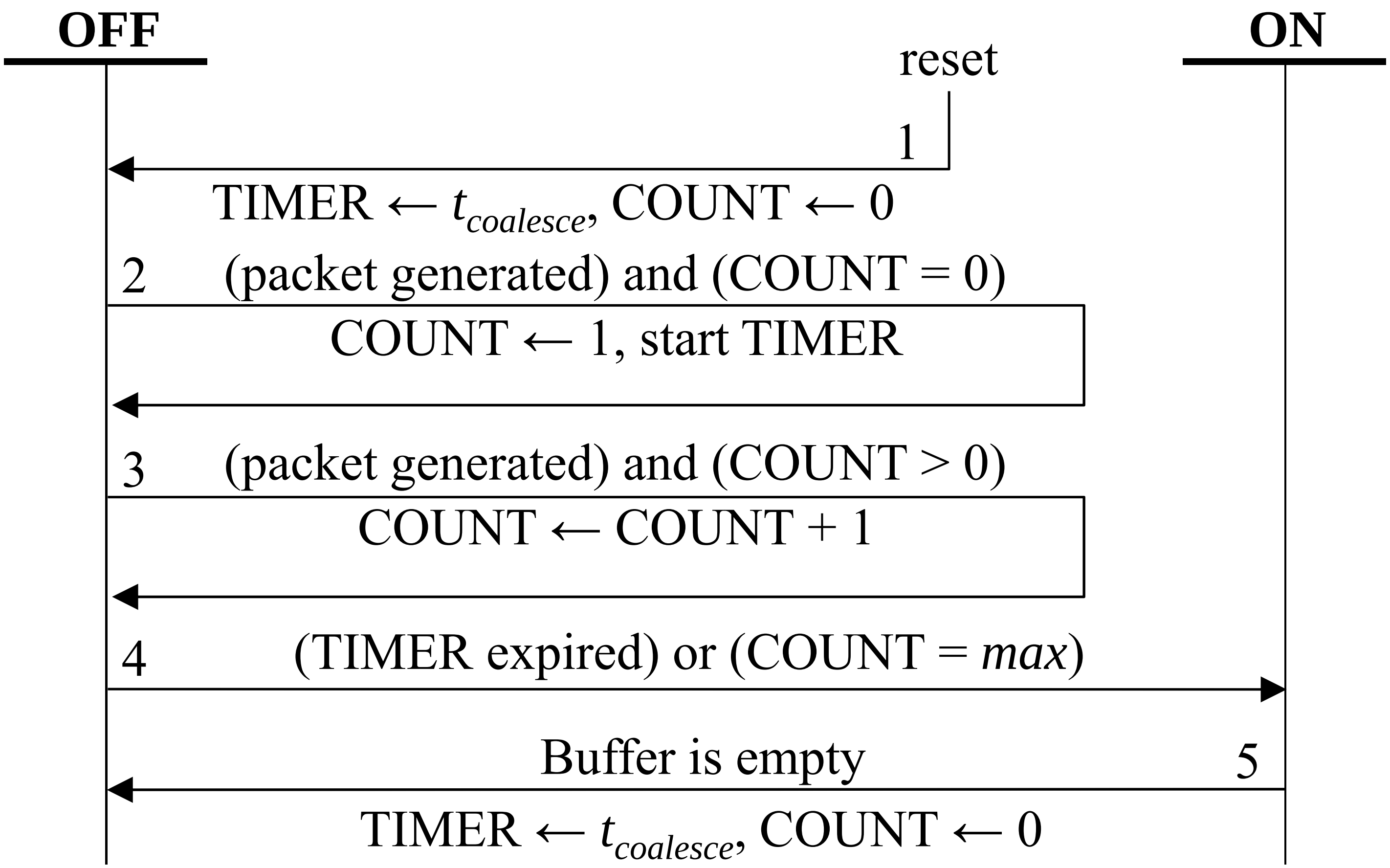
- Key variables

TIMER	Timer for coalescing
COUNT	Packet counter for coalescing
$t_{coalesce}$	Initial timer value for TIMER
max	Maximum count for generated packets

FSM for coalescing <u>continued</u>

The FSM

Only when buffer is empty does transition (5) occur. Thus, more than COUNT packets can be sent each time the ON state is entered.



CSIM model for coalescing

CSIM model

- More complicated than EEE model
- Uses a separate process for the coalescer
- Uses CSIM "wait" event – event is set by a time-out or when coalescer capacity is reached

EEE with coalescing experiment

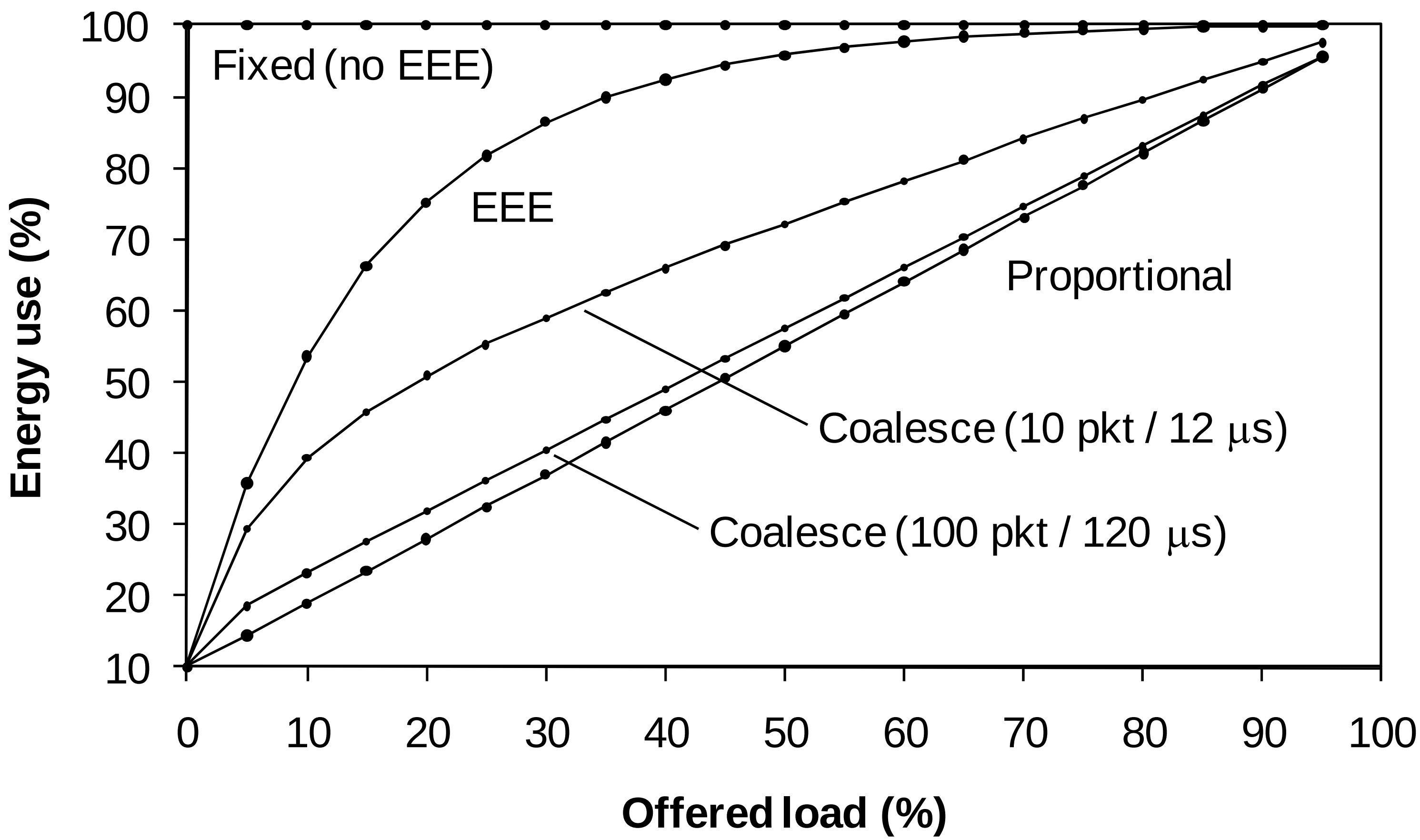
Repeat previous 10 Gb/s experiment

- For 10 Gb/s
  - T_WAKE = 4.16 μs
  - T_SLEEP = 2.88 μs
  - For 1500 byte packet service_time = 1.2 μs
- For coalescing
  - $max = 10$, $t_{coalese} = 12$ μs
  - $max = 100$, $t_{coalesce} = 120$ μs
- Assume that idle power use is 10% of full power use
- Vary offered load from 0% to 95%
  - Poisson arrivals, fixed length packet

EEE with coalescing results

Results for 10 Gb/s with coalescing

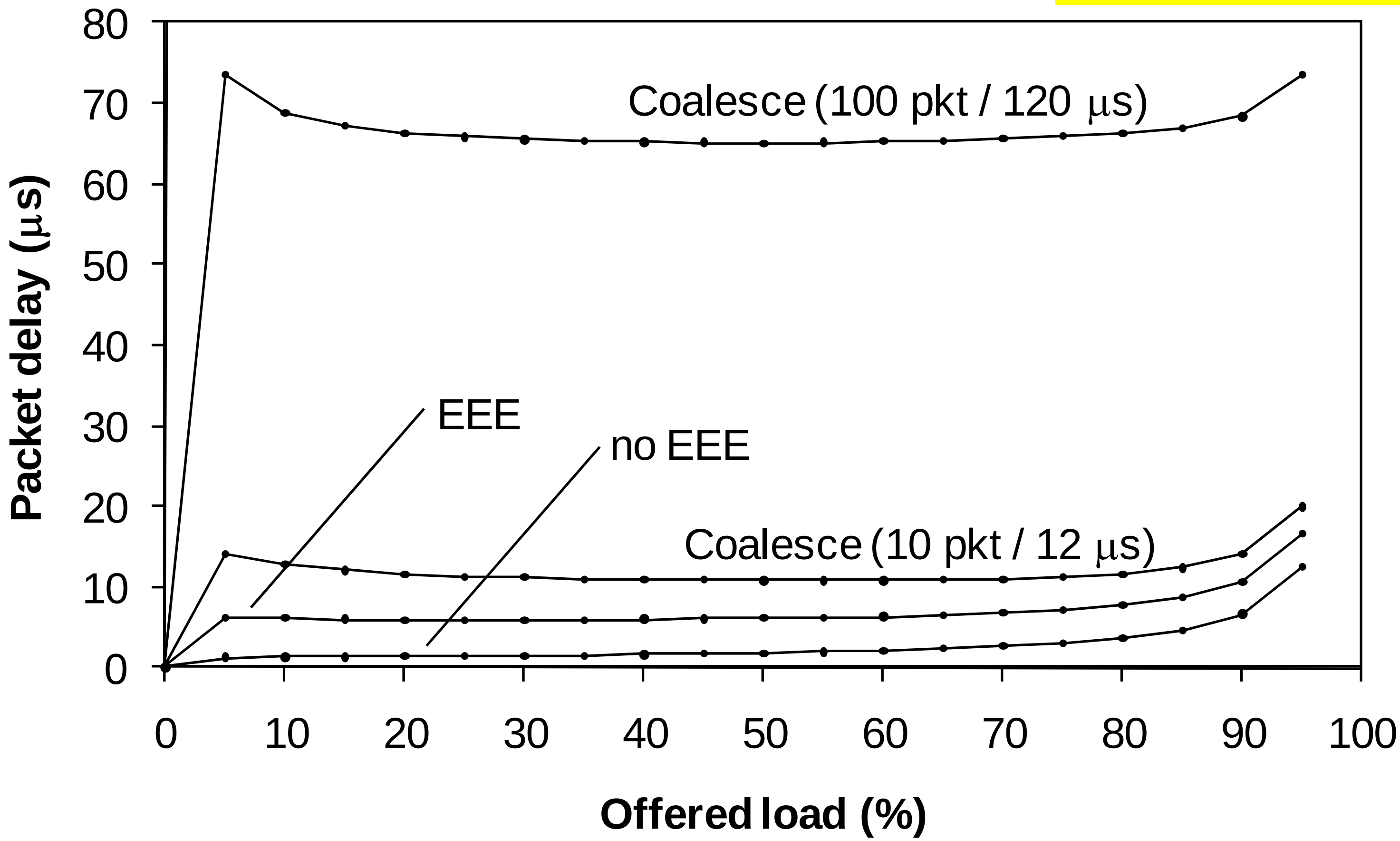
Note significant improvement.



EEE with coalescing results <u>continued</u>

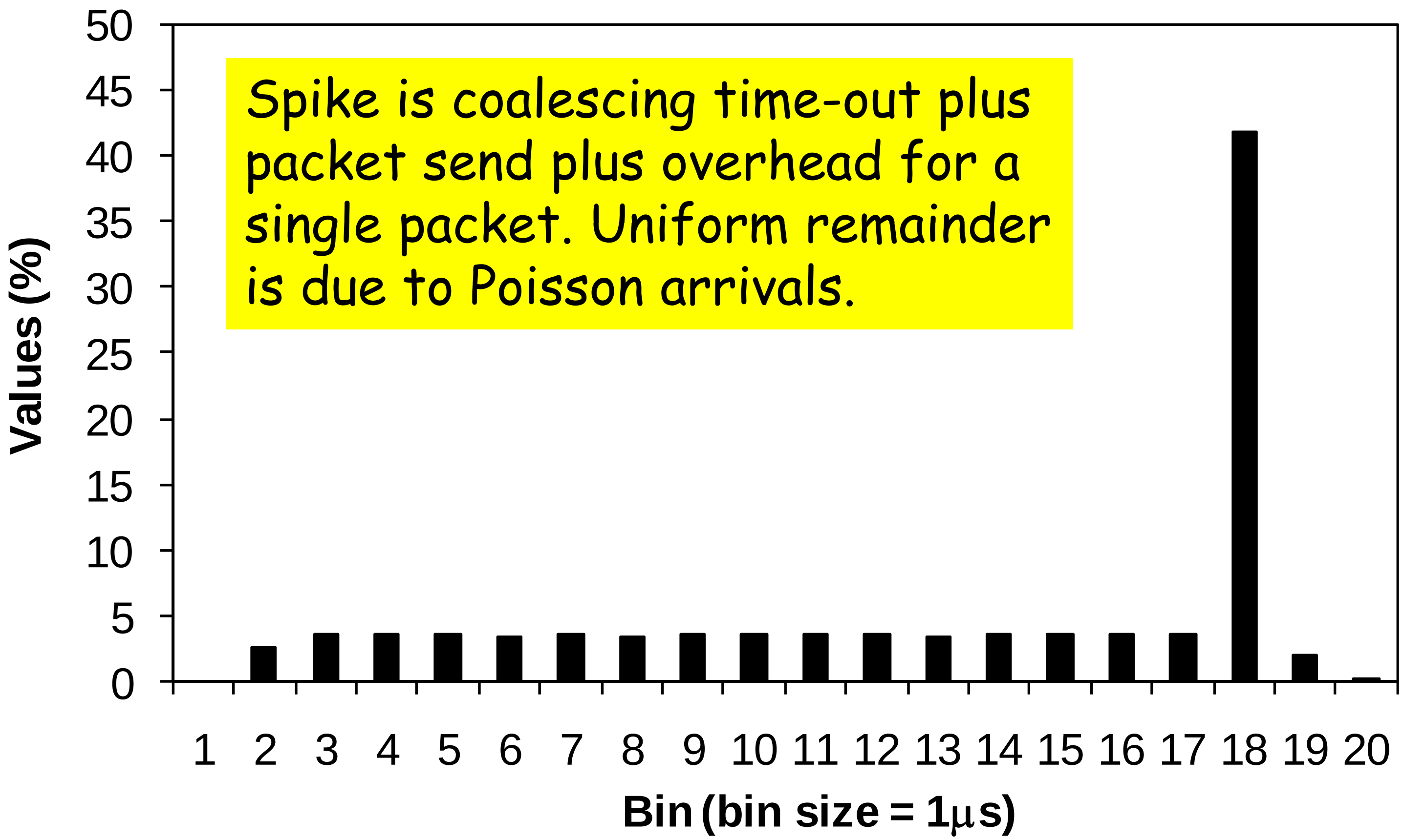
But, what about the added delay?

This is our trade-off



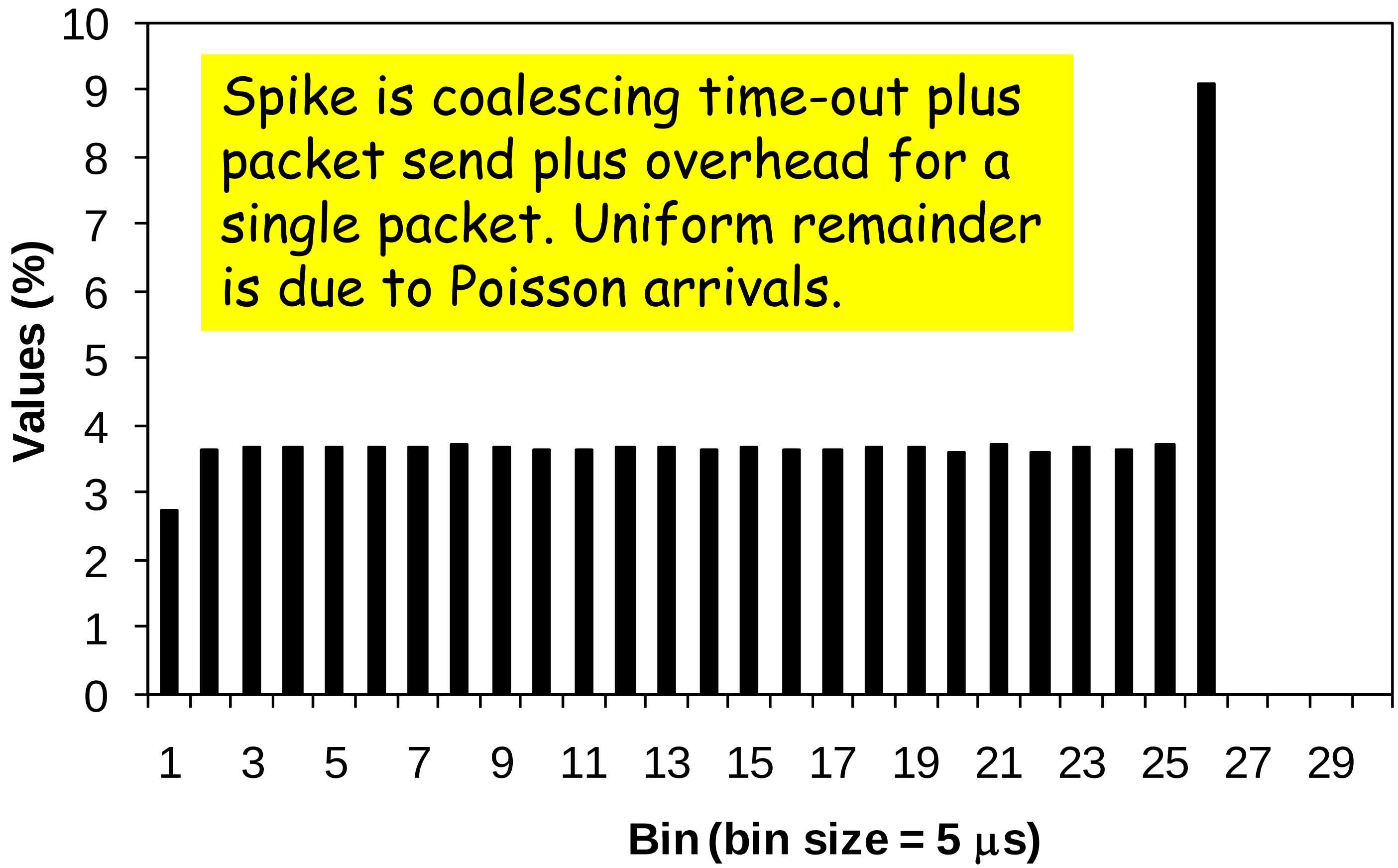
Distribution of EEE delay

For 10 packets / 12 μs and 10% offered load



Distribution of EEE delay <u>continued</u>

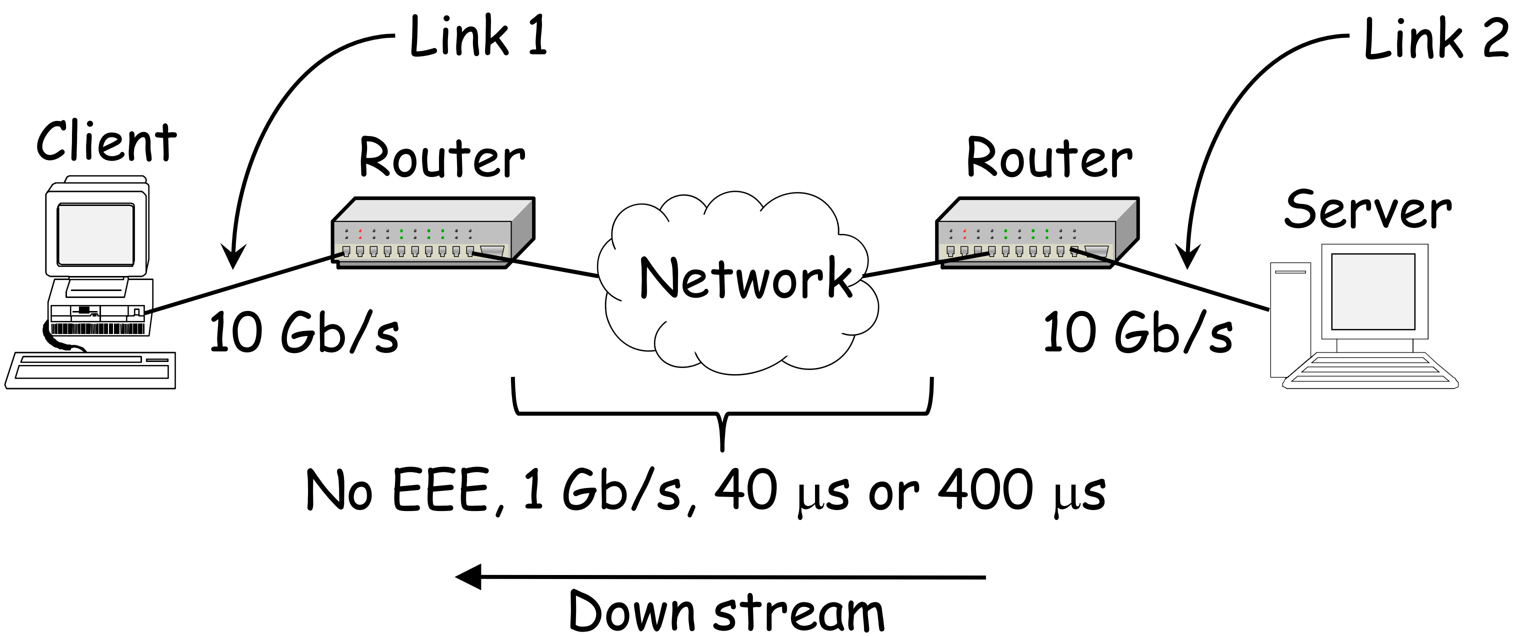
For 100 packets / 120 μs and 10% offered load



EEE file transfer

File transfer experiment with an ns-2 model

- File transfer for a 1 GB file, server to client
- Coalescing implemented in ns-2 (same parameters)



EEE file transfer continued

File transfer experiment parameters

- Router buffer was 100 packets
- Used ns-2 TCP Linux agent and Sack1 receiver
- TCP maximum window size of 400 packets
- The client and sever links (to the network) were 10 Gb/s
  - Without EEE (standard Ethernet)
  - With EEE
  - With EEE and coalesce-1 and coalesce-2
- Coalescing on host and router interfaces

EEE file transfer <u>continued</u>

Key measurements for the model

- File download time
- Utilization on link 1 and link 2
  - Does not include EEE overhead (only packet time)
- Energy use on link 1 and link 2
  - 100% maximum, 10% minimum

EEE file transfer model results

Link utilization

Results are "as expected" here, one ACK per data packet (ACK packets are about 1/24 the size of a data packet).

		Link utilization	
Config (10Gb/s, 40 μs delay)	**Download time** (s)	**Link 1 up**	**Link 1 down**
No EEE	0.843	4.0%	94.9%
EEE	0.843	4.0%	94.9%
EEE coalesce-1	0.843	4.0%	94.9%
EEE coalesce-2	0.847	4.0%	94.5%
Config (1Gb/s, 400 μs delay)	**Download time** (s)	**Link 1 up**	**Link 1 down**
No EEE	8.28	0.4%	9.7%
EEE	8.28	0.4%	9.7%
EEE coalesce-1	8.28	0.4%	9.7%
EEE coalesce-2	8.34	0.4%	9.7%

EEE file transfer model results continued

Energy use

High energy use for ACKs and for the 1 Gb/s case. Coalescing reduces energy use with little extra download time.

		Energy use (100% maximum, 10% for idle)	
Config (10Gb/s, 40 µs delay)	**Download time** (s)	**Link 1 up**	**Link 1 down**
No EEE	0.843	100%	100%
EEE	0.843	99.9%	99.9%
EEE coalesce-1	0.843	50.6%	99.9%
EEE coalesce-2	0.847	21.3%	99.5%
Config (1Gb/s, 400 µs delay)	**Download time** (s)	**Link 1 up**	**Link 1 down**
No EEE	8.28	100.0%	100.0%
EEE	8.28	65.6%	74.4%
EEE coalesce-1	8.28	38.0%	46.7%
EEE coalesce-2	8.34	17.8%	25.8%

Significance of the added delay

What is the significance of the added delay?

- Increased delay is magnitudes less than end-to-end delay on an Internet path
  - End-to-end on Internet is 10s to 100s of milliseconds
  - Coalescing delay is in the 10 to 100s of microseconds

- Increased burstiness may be an issue
  - But, coalescing is already being done for reducing packet processing load on system CPU

- Coalescing can cause TCP "ACK compression"
  - Returning ACKs come in a burst
  - Studied since early 1990s

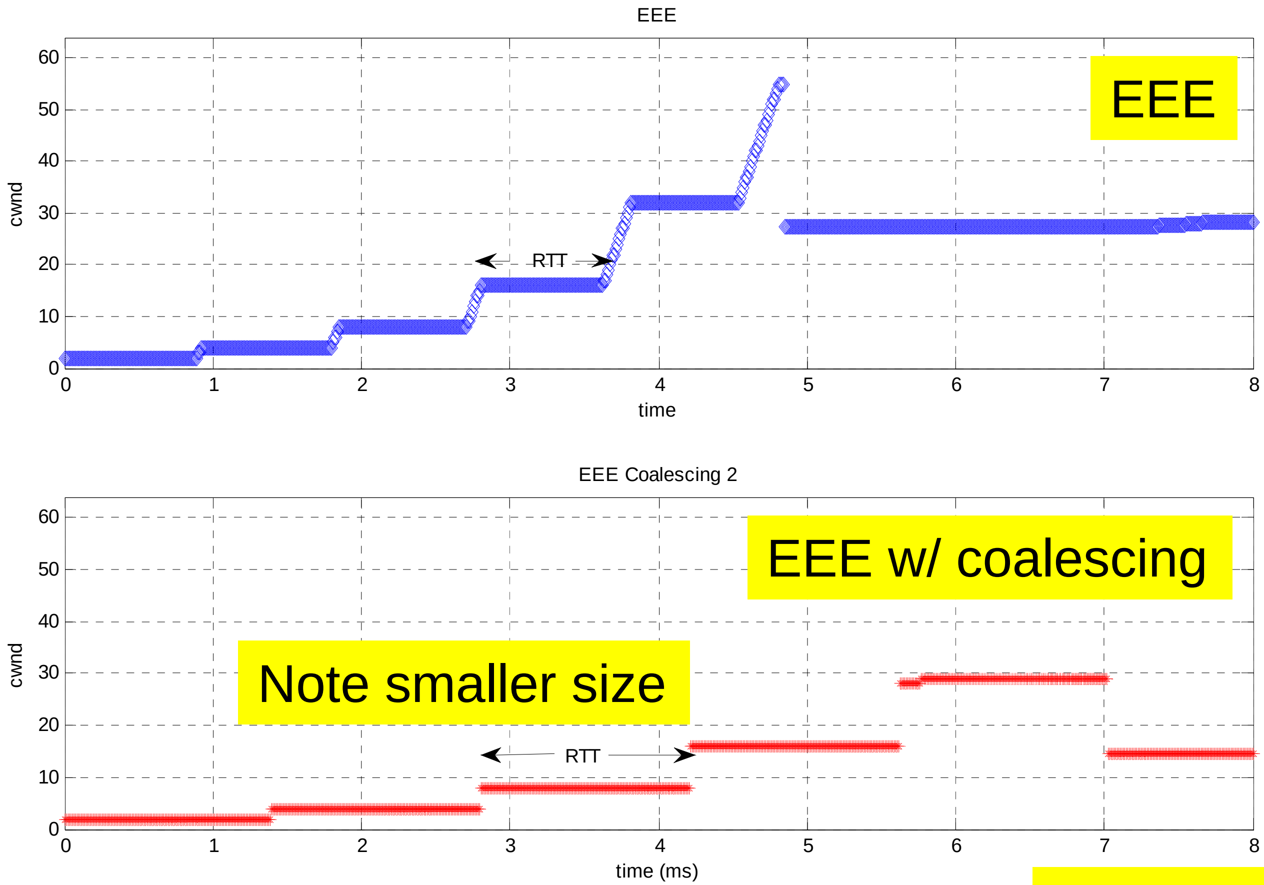
Coalescing and burstiness

A deeper understanding is needed

- Generally, coalescing will increase RTT
  - This requires a larger window size for a given throughput
- Coalescing effects are likely small if,
  - Burst size is much smaller than router and NIC buffers
  - Burst timer is much smaller than RTT
- Should explore how coalescing for reducing CPU load and coalescing to improve EEE efficiency can be combined

Coalescing and burstiness continued

Can explore cwnd growth in slow start with ns-2



Future work for Pedro

Economic benefits from EEE

Estimating the savings

- Savings per link is the difference between fully active and in low power mode

What are the savings from EEE?

What is the additional savings potential from coalescing?

Economic benefits from EEE <u>continued</u>

Assumptions made

- Use known 2008 stock and port count – U.S. only
  - From an estimate made for EPA in 2008
  - Thus, more 1 Gb/s than 10 Gb/s

- Increase data rates, use current power levels, and maintain assumption of low utilization

- Assume large packets, independent arrivals, and 100% PHY power consumption during transitions

Economic benefits from EEE continued

The assumptions

The savings per link may be conservative, also the mix between 1 Gb/s and 10 Gb/s

	1 Gb/s	10 Gb/s	Total
Assumptions			
Savings per link – no data (W)	1	5	—
Link utilization (%)	1 %	3 %	—
Active links (millions)	250	65	315
Electricity cost ($/kWh)	0.10	0.10	—

Economic benefits from EEE <u>continued</u>

EEE savings

The per link savings comes from the previous power graphs (this is the EEE overhead).

	1 Gb/s	10 Gb/s	Total
Results – EEE Savings			
Per link (%)	81 %	82%	—
Per link (W)	0.81	4.10	—
Total (MW)	200	270	470
Total (TWh/year)	1.8	2.3	4.1
Total (million $/year)	180	230	410

Economic benefits from EEE <u>continued</u>

Gain from coalescing

Assumes coalescing gets us to the "ideal" line on the power graphs.

	1 Gb/s	10 Gb/s	Total
Results – Ideal Savings			
Total (TWh/year)	2.2	2.8	5
Total (million $/year)	220	280	500
Coalescing Opportunity			
Percent (last column is average)	22 %	18 %	20 %
Total (TWh/year)	0.39	0.43	0.82
Total (million $/year)	40	40	80

So, $80 million per year from coalescing.

Future challenges

Where can we go from here?

... energy savings *of* and *by* ICT.

Challenges in green networks

Challenges in five areas

1) General (or overall)
2) Network equipment
3) Network hosts
4) Data centers
5) Distributed applications

Challenges in green networks <u>continued</u>

General

- Metrics
  - How do we measure energy-performance trade-offs?
- Models
  - How do we model energy-performance trade-offs?
- Exposing power and usage state
  - Need to be able to remotely determine power/use state
- Architectures for selective connectivity
  - Need mechanisms/protocols for selective connectivity
    - » Includes notions of proxying

Challenges in green networks <u>continued</u>

Network equipment

- Green routers and switches
  - Re-design routers and switches for energy efficiency
- Data caching for energy efficiency
  - Caching to reduce load network and servers
- Traffic shaping for energy efficiency
  - Shaping traffic for short-term shutdown
- Traffic engineering for energy efficiency
  - Routing to consolidate routes for long-term shutdown

Challenges in green networks <u>continued</u>

Network hosts

- Discovery of devices, capabilities, and services
  - Need to be able to discover low-power substitutes

Data center specific

- High bandwidth / low latency for dynamic virtualization
  - Useful for server shutdown
- Move computing work to where power is cheapest
  - "Follow the moon" for data center activity

Challenges in green networks <u>continued</u>

Distributed applications

- P2P, multiplayer games, and virtual worlds
  - Need to address these large and growing energy consumers
- Webcams and sensors everywhere
  - Need to address these large and growing energy consumers

Where are the "best" challenges?

My views…

- I think that the biggest challenges are at the edge
  - Most energy use there
  - Most opportunity for making changes
- Need applications and protocols that allow for and enable hosts and network equipment to sleep
- But… the biggest challenges may be in the "other 98%"
  - Many open networks problems for Smart Buildings
- Be careful to not work on problems already solved
  - Much has now been solved (the "low hanging fruit")
  - Always be able to quantify expected savings and argue that they are sufficient to be of interest

Current work in the lab

Some ideas being worked on…

- Ethernet switch power management
  - Can traffic shaping enable switches to sleep?
- Dual-channel Ethernet link for energy efficiency
  - Low-speed/low-power and high-speed/high-power
- Cooperating proxies to send requests to other machines
  - Notion of a recursive proxy
  - Protocols for discovery

Alessandro's work!

- Demand response for smart appliances
  - Distributed protocols for scheduling appliances in a building

ICT can dematerialize the economy

Our economy is increasingly about…

Moving bits and not atoms

- This is how most of us now earn a living
- Made possible by networks
- Continuing trend may help us be ***comfortably green***

Conclusions

- **ICT has large and growing energy use**
- **EEE will reduce networks energy use**
  – Hundreds of millions of dollars per year in US
- **EEE can be improved with packet coalescing**
  - Tens of millions of dollars per year in US
- **ICT can enable global energy savings**
  – Moving bits and not atoms = less CO_2
- **There are future challenges to be addressed**

Any questions?

Ken Christensen

http://www.csee.usf.edu/~christen/energy/main.html