Optimization of Network Infrastructures.

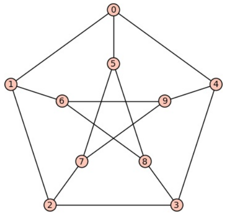
A little bit of green in networks and other problems of placement and management of resources.

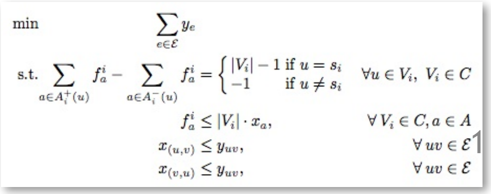
Frédéric Giroire

Université Côte d'Azur/CNRS/Inria COATI*

HdR — October 23, 2018

*Combinatorics, Optimisation et Algorithms For Telecommunications

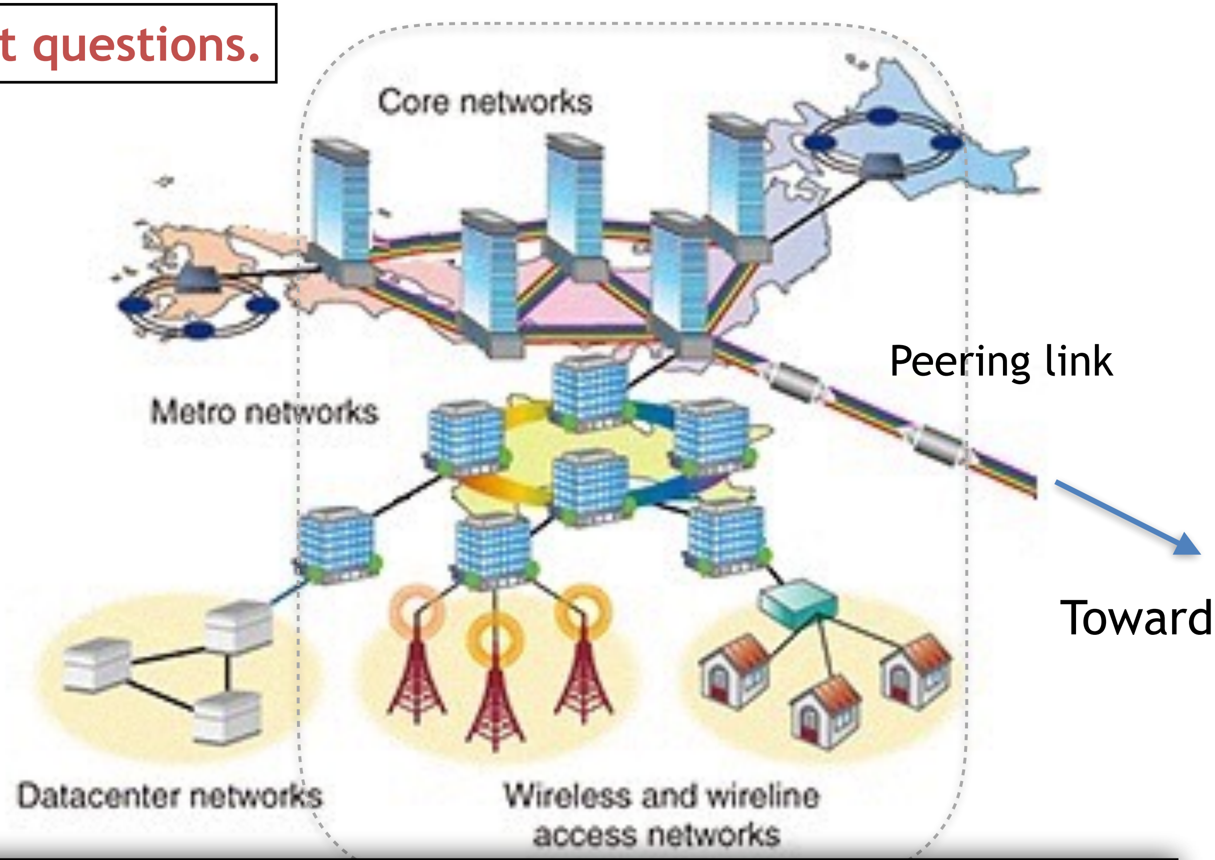




$$\begin{array}{lll}
\min & \displaystyle\sum_{e\in\mathcal{E}} y_e & \\
\text{s.t.} & \displaystyle\sum_{a\in A_i^+(u)} f_a^i - \sum_{a\in A_i^-(u)} f_a^i = \begin{cases} |V_i|-1 \text{ if } u = s_i \\ -1 \qquad\;\; \text{if } u \neq s_i \end{cases} & \forall u \in V_i,\ V_i \in C \\
& f_a^i \leq |V_i| \cdot x_a, & \forall\, V_i \in C, a \in A \\
& x_{(u,v)} \leq y_{uv}, & \forall\, uv \in \mathcal{E} \\
& x_{(v,u)} \leq y_{uv}, & \forall\, uv \in \mathcal{E}
\end{array}$$

Optimization of Network Infrastructures

Lots of different questions.



Towards other ISPs

I answered some of them using tools from **algorithmics**, **optimization**, **combinatorics (graph theory)**, **simulations** and **experimentations**.

Optimization of Network Infrastructures

Manuscript Content

How to know the traffic?
Probabilistic algorithms for cardinality
PhD Inria Rocquencourt + Paris 6
P. Flajolet and M. Soria

Tool: Analysis of algorithms

Traffic Analysis - Security

PhD
(Rocq.)

2003

2006

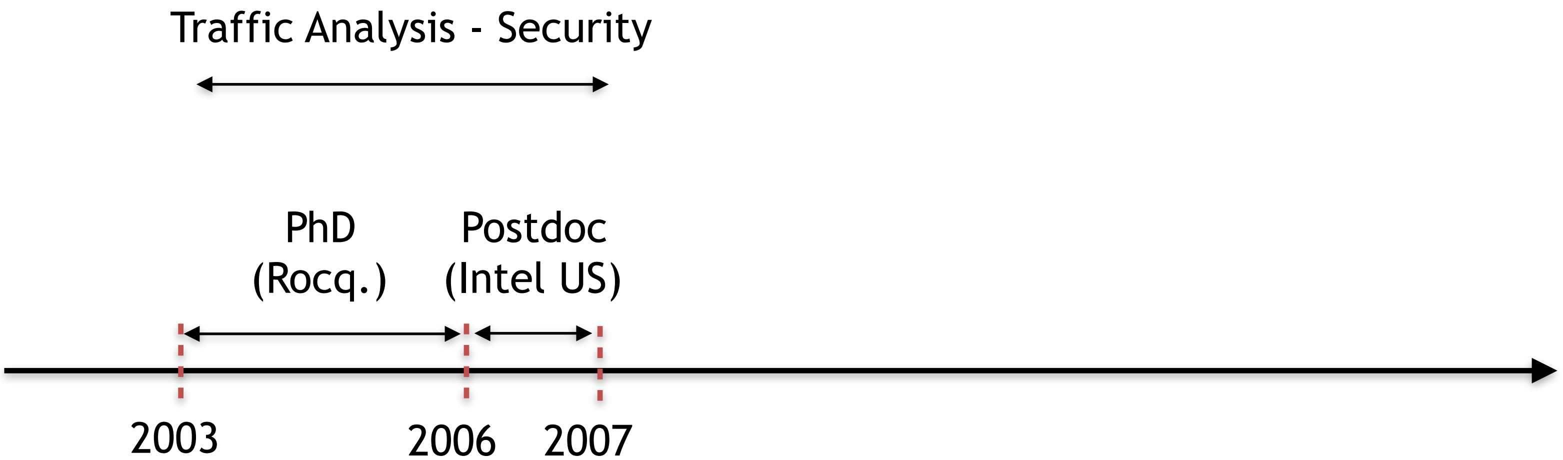
Optimization of Network Infrastructures

How to secure the traffic?
Anomaly detection. DDoS and botnets
PhD
Postdoc Intel Research Berkeley
N. Taft and J. Chandrashekar

Tools: Analysis of Traffic - Algorithms

Manuscript Content

- Appendix Section 1.2
    - Traffic analysis and security
- Part I
- Part 2



Traffic Analysis - Security

PhD
(Rocq.)

Postdoc
(Intel US)

2003

2006

2007

Optimization of Network Infrastructures

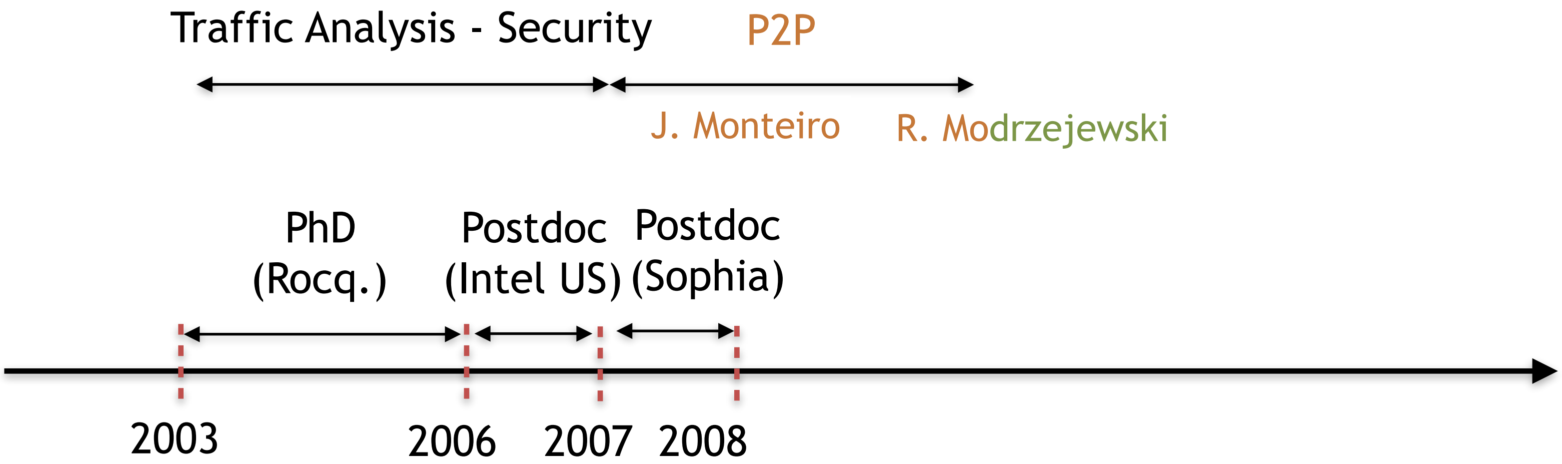
How to back up data?
Analysis of P2P storage systems

Postdoc Sophia
J.-C. Bermond and S. Pérennes

Tools : Markov chains

Manuscript Content

- Appendix Section 1.2
    - P2P storage
    - Traffic analysis and security
- Part I
- Part 2



Traffic Analysis - Security

P2P

J. Monteiro

R. Modrzejewski

PhD (Rocq.)

Postdoc (Intel US)

Postdoc (Sophia)

2003

2006

2007

2008

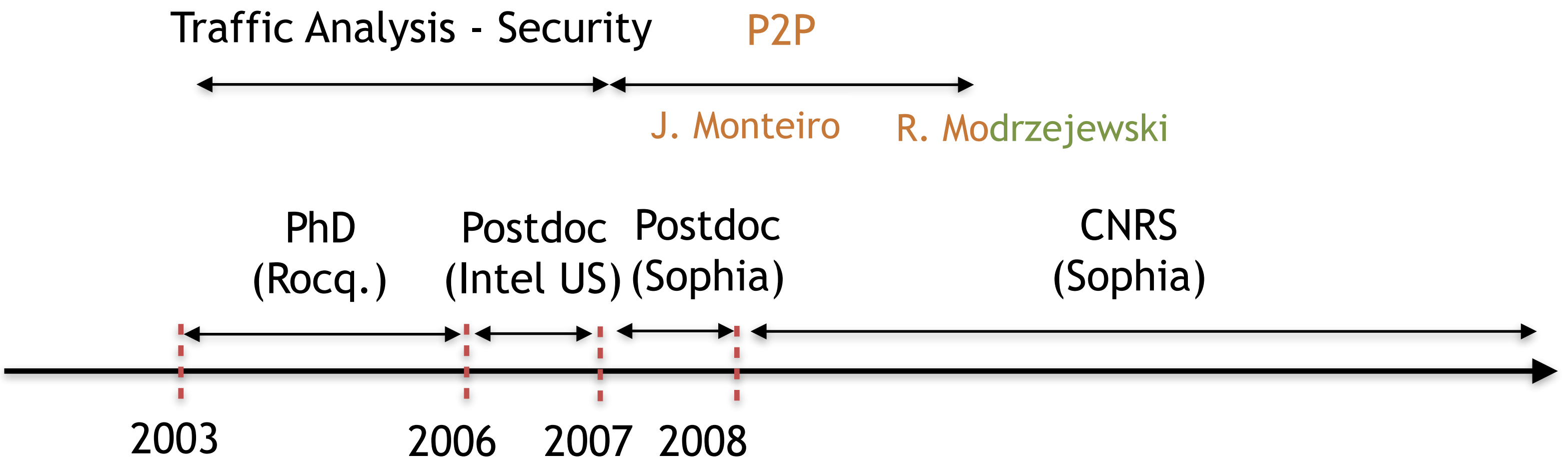
Optimization of Network Infrastructures

How to back up data?
Analysis of P2P storage systems
Section 1.2.3
CR CNRS - I3S/Mascotte

Tools: Markov chains

Manuscript Content

- Appendix Section 1.2
  - P2P storage
  - Traffic analysis and security
- Part I
- Part 2



Traffic Analysis - Security

P2P

J. Monteiro

R. Modrzejewski

PhD
(Rocq.)

Postdoc
(Intel US)

Postdoc
(Sophia)

CNRS
(Sophia)

2003

2006

2007

2008

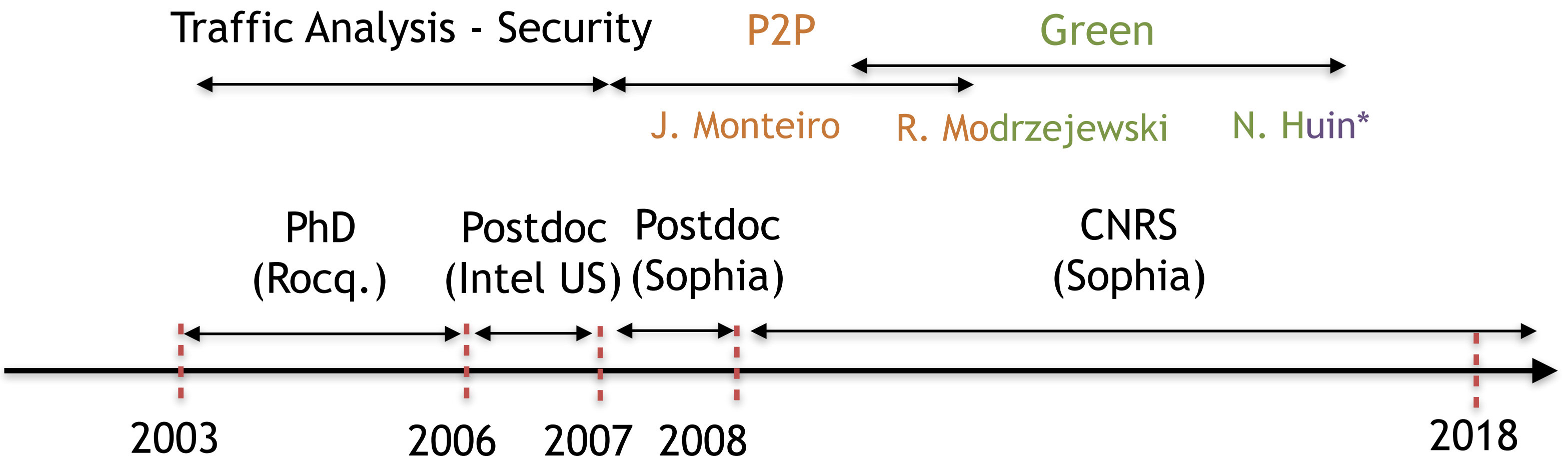
Optimization of Network Infrastructures

How to build green networks?
Energy Aware Routing
Part I

Tools: Optimization, Algorithmics, Simulations

Manuscript Content

- Appendix Section 1.2
  - P2P storage
  - Traffic analysis and security
- Part I Green Networks
  - Chap 3 Practical scenarios
    Chap 4 Graph Theory
- Part 2



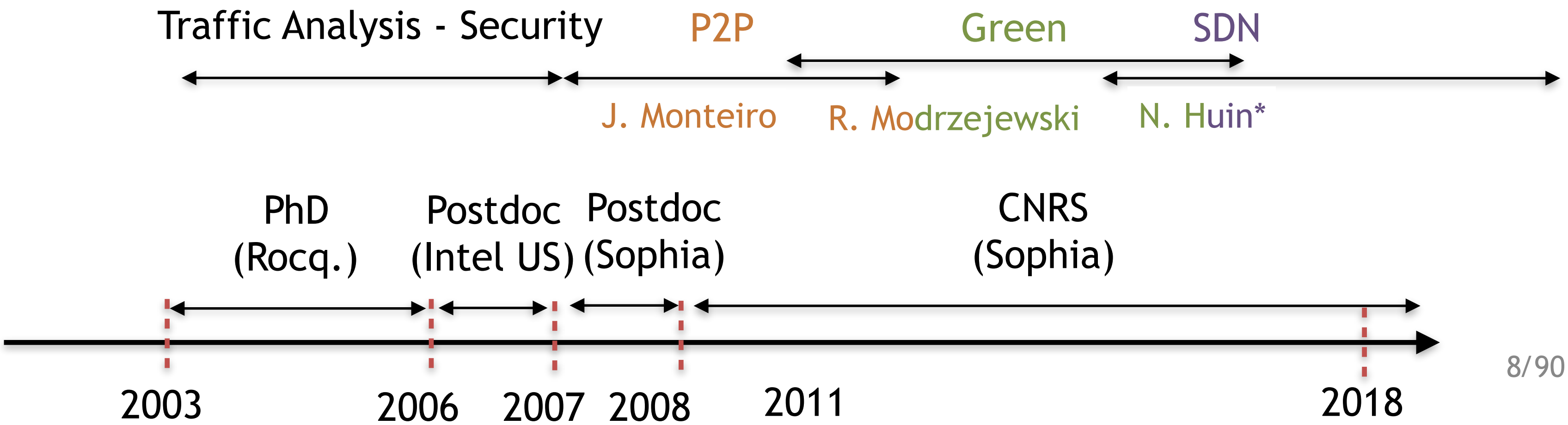
Optimization of Network Infrastructures

How to control the network in real time?
Software Defined Networks (SDN)
Part 2

Tools: Algorithms, FPT, Simulation, Experiments

Manuscript Content

- Appendix Section 1.2
  - P2P storage
  - Traffic analysis and security
- Part I Green Networks
  - Chap 3 Practical scenarios
    Chap 4 Graph The
- Part 2 Virtualized SDN
  - Chap 6 SDN



Traffic Analysis - Security

P2P

Green

SDN

J. Monteiro

R. Modrzejewski

N. Huin*

PhD
(Rocq.)

Postdoc
(Intel US)

Postdoc
(Sophia)

CNRS
(Sophia)

2003

2006

2007

2008

2011

2018

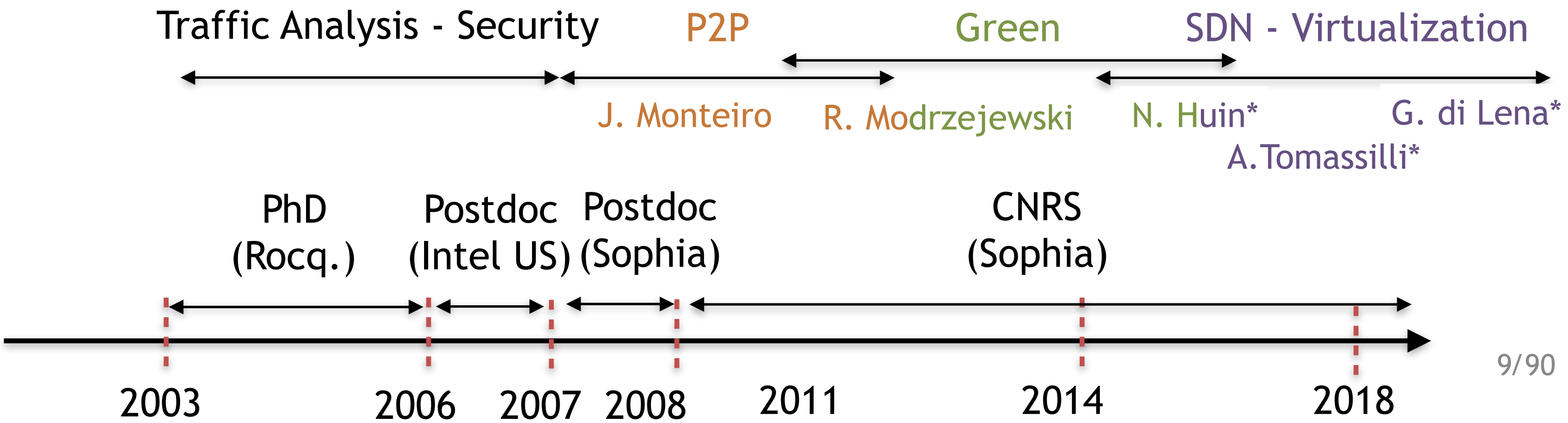
Optimization of Network Infrastructures

How to place (virtual) resources?
Provisioning of Virtual Network Functions
Part 2

Tools: Column generation,
Approximation algorithms.

Manuscript Content

- Appendix Section 1.2
  - P2P storage
  - Traffic analysis and security
- Part I Green Networks
  - Chap 3 Practical scenarios
    Chap 4 Graph Theory
- Part 2 Virtualized SDN
  - Chap 6 SDN
  - Chap 7 NFV
  - Chap 8 Green networks in practice



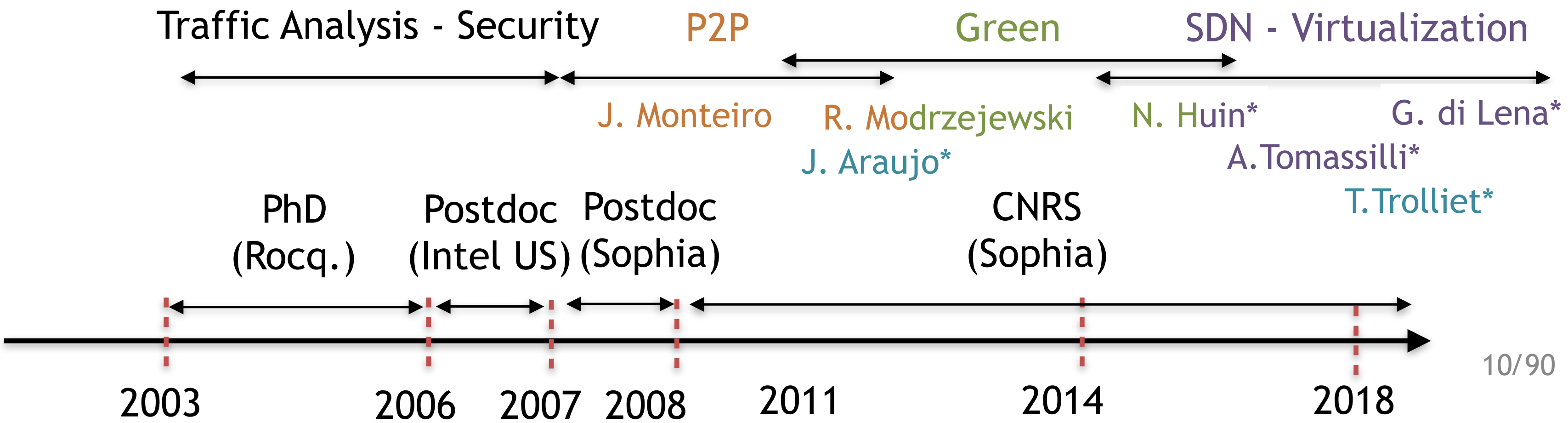
Optimization of Network Infrastructures

How to model networks with graphs?
Coloring for wireless,
Random graphs for social networks

Appendix 3.

Tools: Graph theory.

Manuscript Content
- Appendix Section 1.2
  - P2P storage
  - Traffic analysis and security
  - Graph theory
- Part I Green Networks
  - Chap 3 Practical scenarios
    Chap 4 Graph The
- Part 2 Virtualized SDN
  - Chap 6 SDN
  - Chap 7 NFV
  - Chap 8 Green networks in practice



Traffic Analysis - Security

P2P

Green

SDN - Virtualization

J. Monteiro

R. Modrzejewski

N. Huin*

G. di Lena*

J. Araujo*

A.Tomassilli*

T.Trolliet*

PhD
(Rocq.)

Postdoc
(Intel US)

Postdoc
(Sophia)

CNRS
(Sophia)

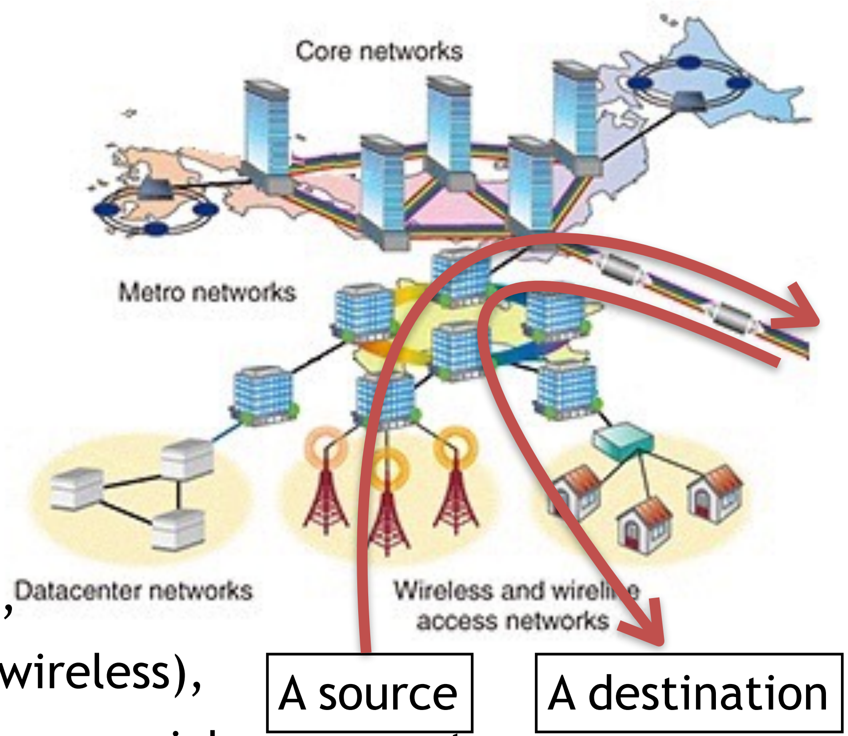
Supervised students

- **Phd Supervisions (3 in progress)**
  - 2018-2021. Giuseppe di Lena with Thierry Turletti, (EPI DIANA), and Chidung Lac (Orange Labs). **Resilience of virtualized networks.**
  - 2017-2020. Thibaud Trolliet with Arnaud Legout (EPI DIANA). **Analysis of large social networks.**
  - 2016-2019. Andrea Tomassilli with Stéphane Pérennes.
    Next generation virtualized networks.
- **Phd Supervisions (4 defended) :**
  - 2014-2017. Nicolas Huin with Dino Lopez (SIGNET).**Energy-efficient Software Defined Networks.**
    Now : Huawei Research Lab, Paris.
  - 2010-2013. Remigiusz Modrzejewski with J.-C. Bermond. **Content Distribution and Storage.**
    Now : Google, Dublin, Irlande.

- **Phd Supervisions (defended) :**
  - 2009-2012. Julio Araujo with Jean-Claude Bermond et Claudia Linhares (Ceara, Brazil). **Graph Coloring and Graph Convexity.**
    Now : Assistant Professor Fortaleza, Brésil.
  - 2007-2010. Julian Monteiro with Olivier Dalle and S. Pérennes. **Modeling and Analysis of P2P Data Storage Systems.**
    Now : Team leader in Cittati Tecnologia. Co-Founder and CTO of Lejour startup.
- **Postdocs: 2**
  - 2012-2013. Luca Chiaraviglio with Joanna Moulierac. **Energy-efficient Networks.**
    Now : Assistant professor (Tenure-Track) University Roma Tor Vergata.
  - 2011-2012. Yaning Liu with Joanna Moulierac. **Energy-efficient Networks.**
    Now : JCP-Consult R&D research and management of european projects.
- **Masters: 10 students.**

Routing

How to route?



However: not a simple shortest path.

Due to:

- capacities and multiple demands (NP-hard),
- heterogeneous infrastructures (e.g. wired/wireless),
- private owners of the infrastructure with commercial agreements,
- frequent failures.

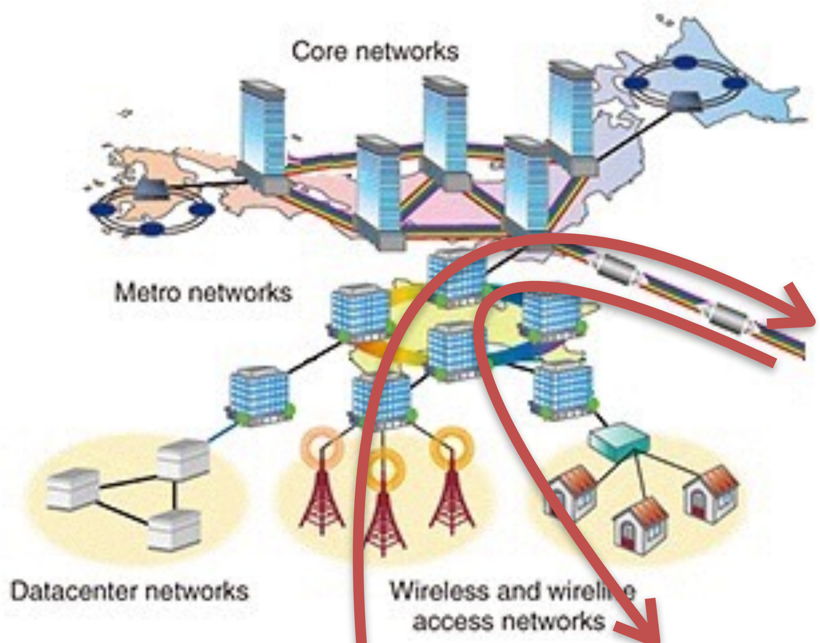
What’s needed from a routing solution?

- Efficient in terms of delay/bandwidth/cost/failure protection.

Routing

How to route?

Legacy solutions: Distributed protocols, such as OSPF, BGB.



A source

A destination

Context has changed.

Routing: A new Context

- Career Routing. Introduction

1. Routing in an energy aware world.
   (Part I)

2. Routing in an SDN world without (a lot of) rules.
   (Part 2. Chapter 6)

3. Routing in a virtualized world.
   (Part 2. Chapter 7)

- End of the route ? Conclusions and Perspectives

Routing: A new Context

- Career Routing. Introduction

1. Routing in an energy aware world.
   (Part I)

2. Routing in an SDN world without (a lot of) rules.
   (Part 2. Chapter 6)

3. Routing in a virtualized world.
   (Part 2. Chapter 7)

- End of the route ? Conclusions and Perspectives

Energy Consumption of Networks

- ICT electricity consumption represents 2-10% of global consumption.
  - Telecom infrastructure and devices account for 25% of ICT’s energy consumption.

3.7 TWh ~ 0.5 billion US$

Corporate Responsibility

9.9 TWh ~ 1.5 billion US$

- Politics. Challenge of the European Commission: a 20% improvement in energy efficiency by 2020.

- Networking research community.
  - Pioneering work [Gupta et al. SIGCOMM 2003]
  - Strong interest from 2008
  - ANR-JCJC DIMAGREEN 2009-2012 (leader)

What can we do? Basic Principles of Power Management

To save Energy we can:

- use more efficient chips and components
- power manage components and systems

To power manage: two main methods

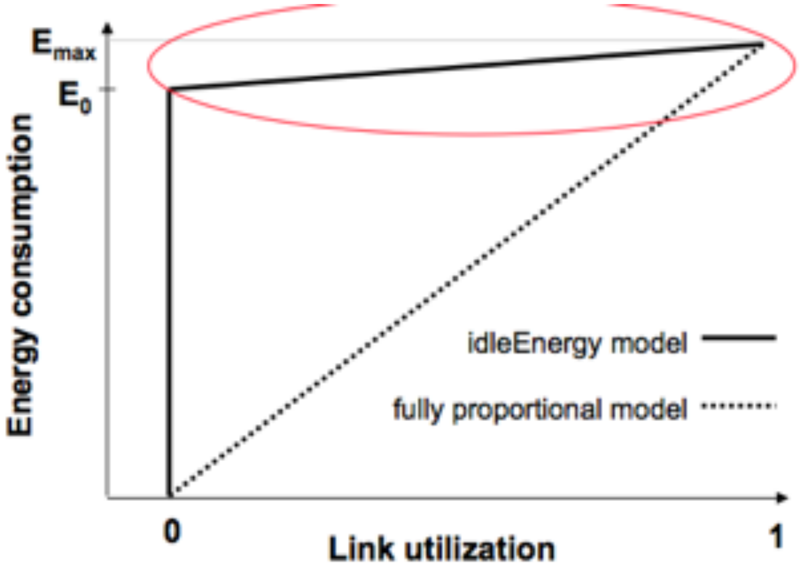
- Do less work: e.g. transmit less in networks. [Nedevschi et al. NSDI 2008.]
- Turn-off devices: not being used [Beloglazov, CCGrid 2010]
  - e.g., floating point unit, disk drive, server in a cluster.

"Most electronics are lightly utilized"

- 2/3 of PC energy used when no one present
- Typical commercial server utilization: ~15 to 20%
- Typical (edge) network link usage: few percents.

Energy Consumption of Networks

Measurements campaigns on routers: small influence of the traffic load on energy consumption on [Chabarek et al. Infocom08]:



—> To save energy: **switch-off** interfaces, chassis.

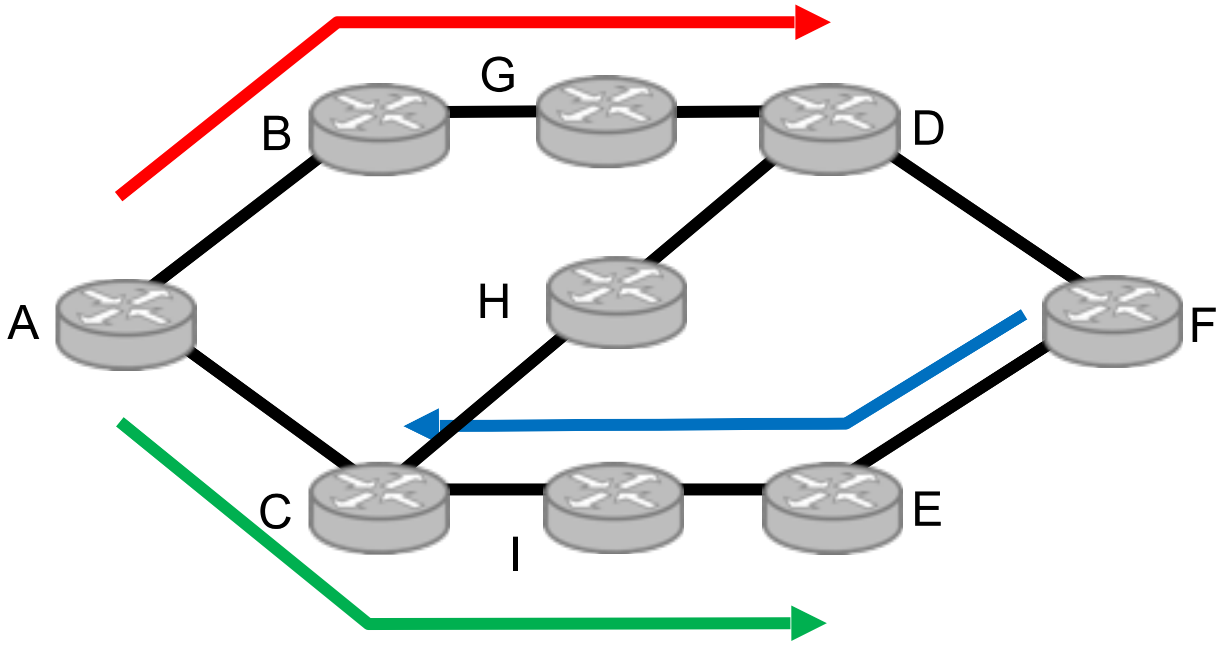
Energy Aware Routing (EAR)

Path between:

A and D

F and C

A and E



Legacy routing: using shortest paths.

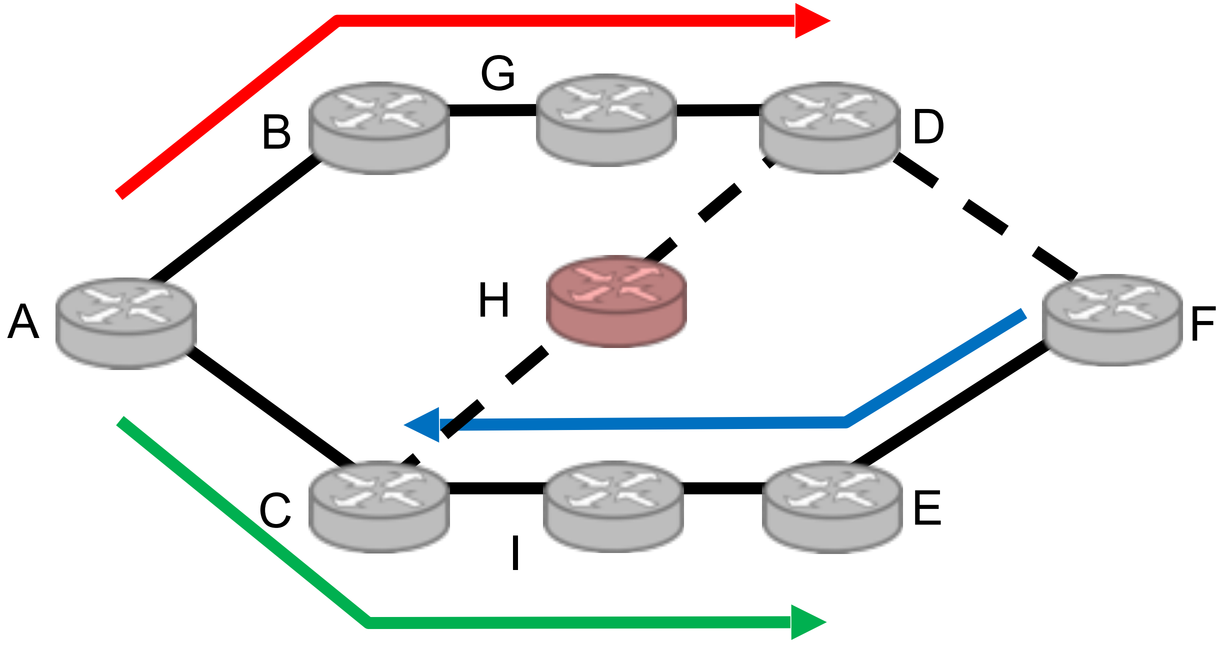
Energy Aware Routing (EAR)

Path between:

A et D

F et C

A et E



Shortest path routing

Putting unused network equipments (routers and/or links) into sleep mode

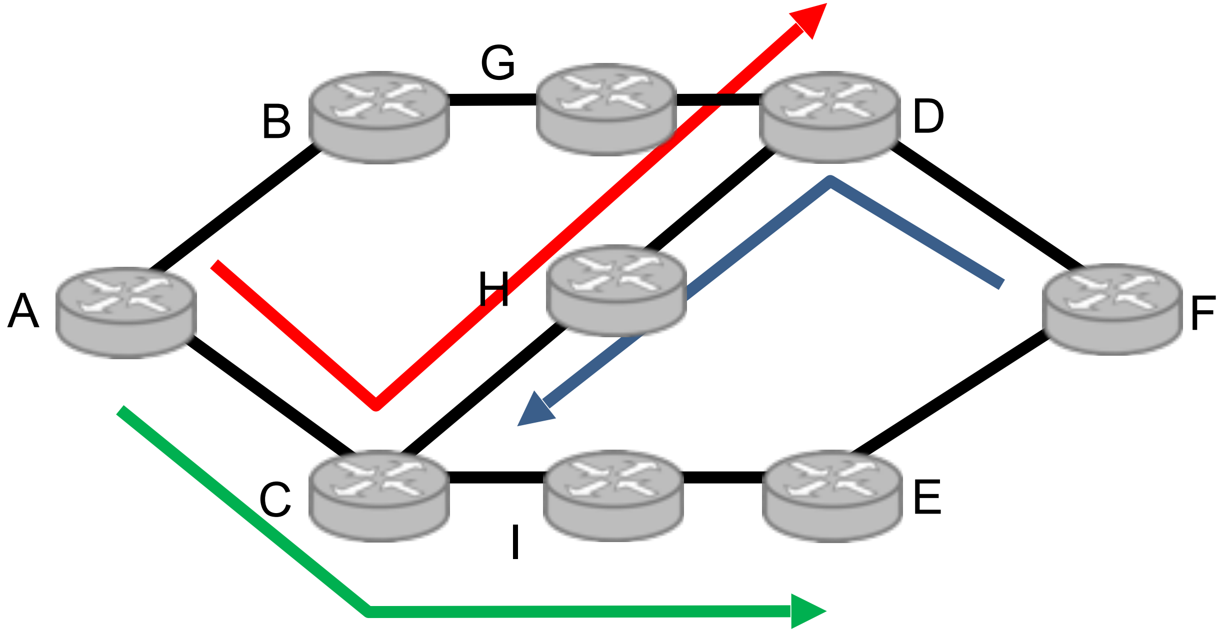
Energy Aware Routing (EAR)

Path between:

A et D

F et C

A et E



EAR: Routing requests while minimizing the number of active network equipments

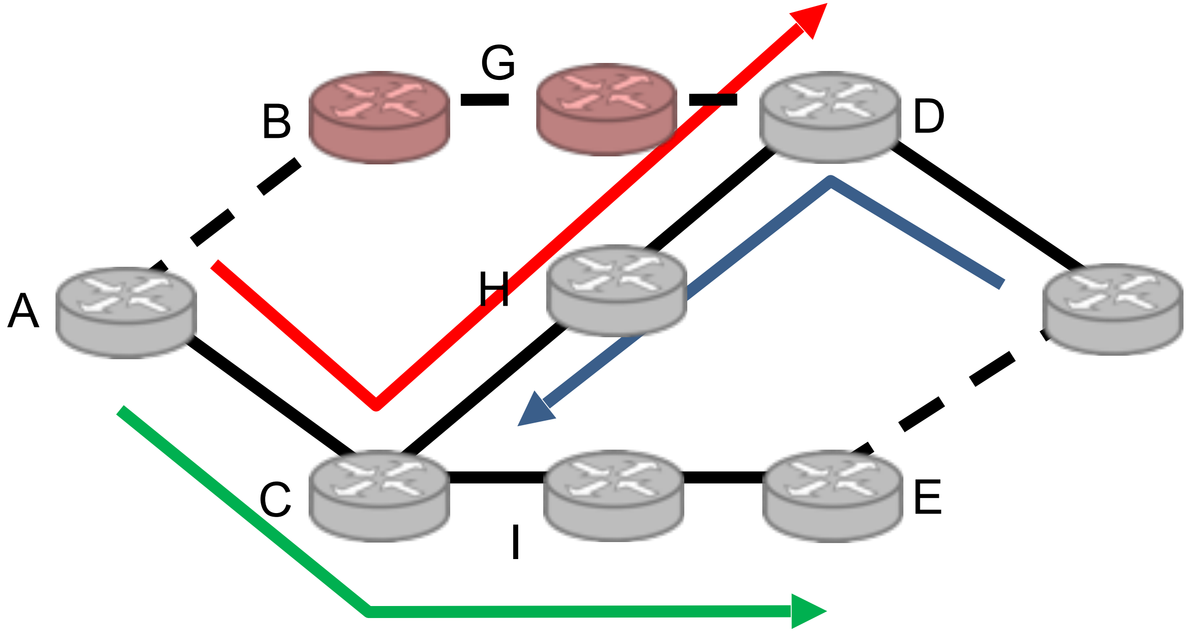
Energy Aware Routing (EAR)

Path between:

A et D

F et C

A et E



Energy Aware Routing

EAR: Routing requests while minimizing the number of active network equipments

Energy Aware Routing (EAR)

- Closely related to classic problems.
  - Routing: Maximum multicommodity flow with a specific cost function [Even. SIAM Journal of Computing 1976]
  - Design: Finding the minimum cost to build a network [Johnson et al. Networks 1978].
  - OR: instance of the Fixed charge Transportation Problem [Gray. Operations Research 1971]

- But with new angle (e.g. dynamics), new applications (energy cost and specifics) and a lot of open problems.

My contributions

- Practical Scenarios [Chap. 3]
  - Study of ISP networks [Chap. 3.1]
  - Using redundancy elimination [Chap. 3.2]
  - For content delivery [Chap. 3.3]

- Using Algorithm Complexity and Graph Theory
  - Hardness results (No-APX) [Chap. 3.1]
  - Theoretical bounds for specific topologies (grids, rings, trees, etc...) and all-to-all [Chap. 4]

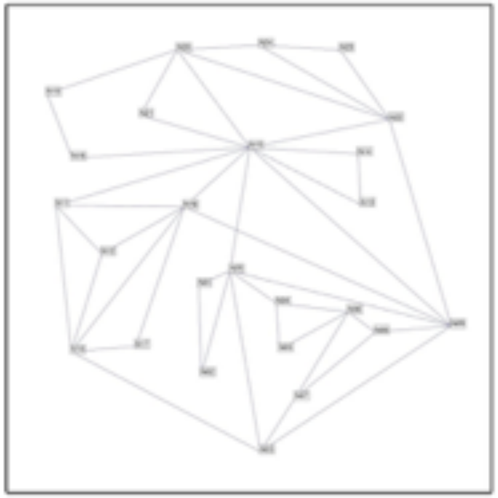
My contributions

- Practical Scenarios [Chap. 3]
  - Study of ISP networks [Chap. 3.1]
  - Using redundancy elimination [Chap. 3.2]
  - For content delivery [Chap. 3.3]

- Using Algorithm Complexity and Graph Theory
  - Hardness results (No-APX) [Chap. 3.1]
  - Theoretical bounds for specific topologies (grids, rings, trees, etc...) and all-to-all [Chap. 4]

Results on ISP Topologies

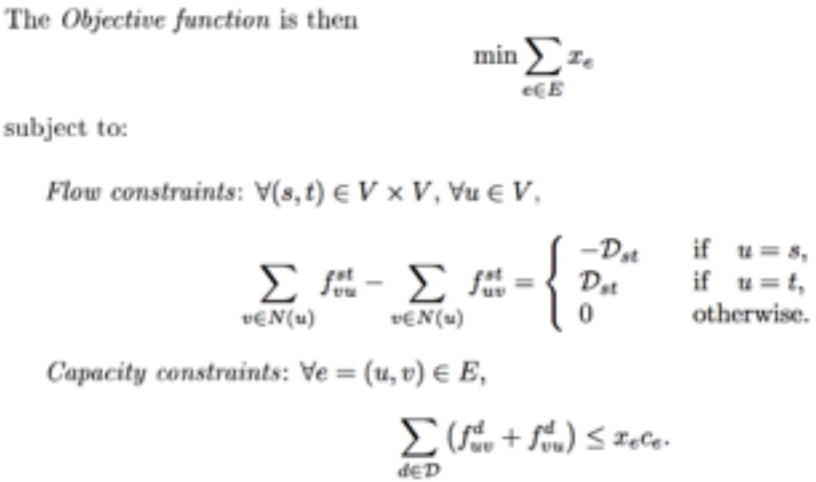
- Topologies extracted from SNDLib
- Tested how many interfaces can be spared for different ranges of operation on 10 topologies
- Also looked at the impact on route length and on fault protection



[Greencom 2010]
[Chapter 2012]

EAR Modelling

Linear program



The *Objective function* is then

$$\min \sum_{e \in E} x_e$$

subject to:

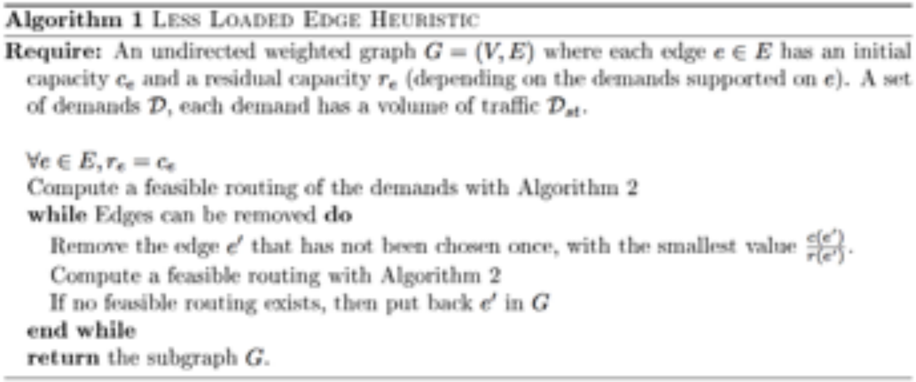
Flow constraints: $\forall (s,t) \in V \times V, \forall u \in V,$

$$\sum_{v \in N(u)} f^{st}_{vu} - \sum_{v \in N(u)} f^{st}_{uv} = \begin{cases} -\mathcal{D}_{st} & \text{if} \quad u = s, \\ \mathcal{D}_{st} & \text{if} \quad u = t, \\ 0 & \text{otherwise.} \end{cases}$$

Capacity constraints: $\forall e = (u,v) \in E,$

$$\sum_{d \in \mathcal{D}} (f^d_{uv} + f^d_{vu}) \leq x_e c_e.$$

Heuristic algorithm

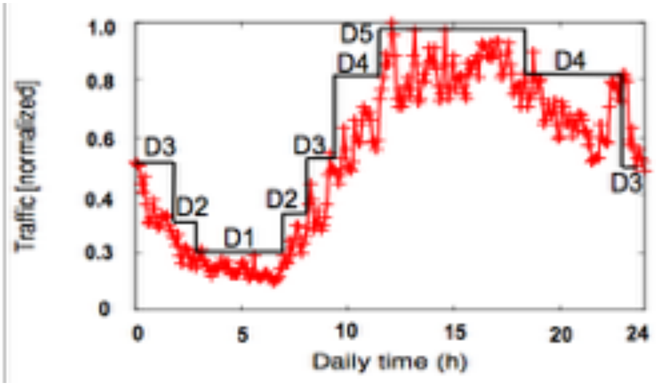


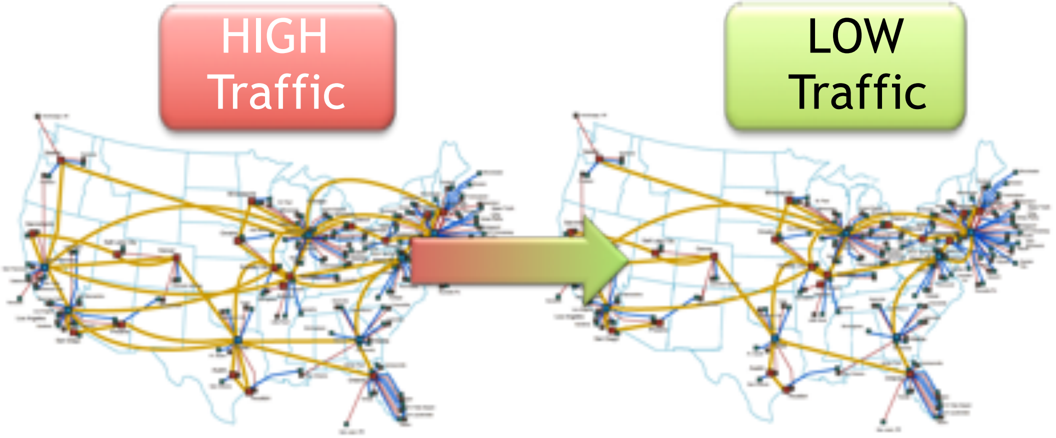
```
Algorithm 1 LESS LOADED EDGE HEURISTIC
Require: An undirected weighted graph $G = (V, E)$ where each edge $e \in E$ has an initial
  capacity $c_e$ and a residual capacity $r_e$ (depending on the demands supported on $e$). A set
  of demands $\mathcal{D}$, each demand has a volume of traffic $\mathcal{D}_{st}$.

  $\forall e \in E, r_e = c_e$
  Compute a feasible routing of the demands with Algorithm 2
  while Edges can be removed do
    Remove the edge $e'$ that has not been chosen once, with the smallest value $\frac{c(e')}{r(e')}$.
    Compute a feasible routing with Algorithm 2
    If no feasible routing exists, then put back $e'$ in $G$
  end while
  return the subgraph $G$.
```

Principle: remove the least loaded edge.

Results on ISP Topologies



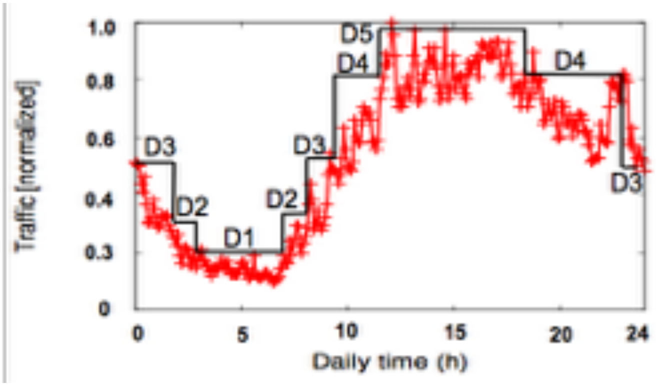


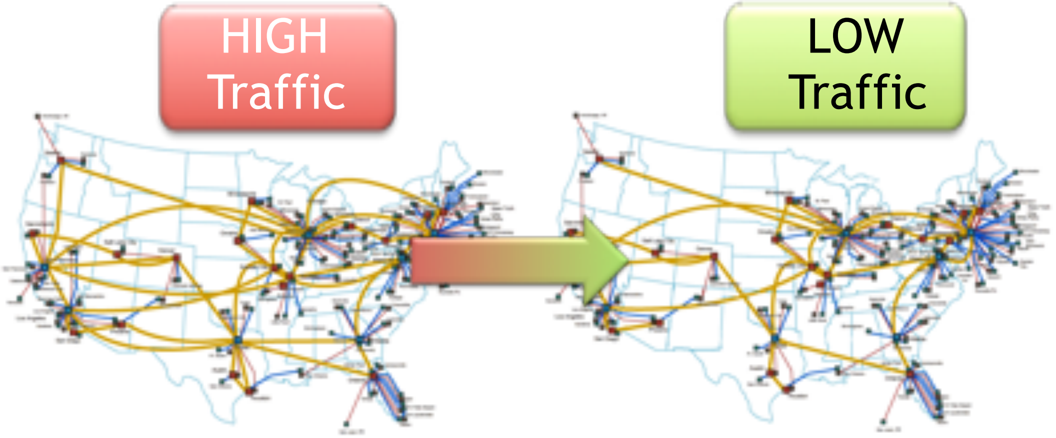
HIGH Traffic

LOW Traffic

- Between 30-60% of spared network equipements for usual range of operations
- But with impact on
  - route length (however limited average impact)
  - failure protection (usually single existing path)

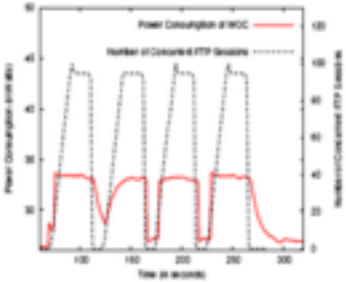
Results on ISP Topologies





- Failure protection:
  - Add fault protection constraints
  - Impact depends on the technology: How long to switch back on vs rerouting time? [experiments of Chap 8]

My contributions



- Practical Scenarios [Chap. 3]
  - Study of ISP networks [Chap. 3.1]
  - Using redundancy elimination [Chap. 3.2]
  - For content delivery [Chap. 3.3]

- Using Algorithm Complexity and Graph Theory
  - Hardness results (No-APX) [Chap. 3.1]
  - Theoretical bounds for specific topologies (grids, rings, trees, etc...) and all-to-all [Chap. 4]

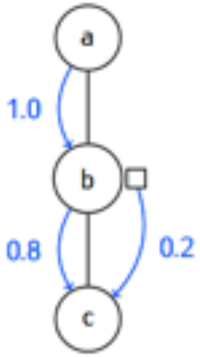
[ComCom 2015]

[Networking 2012]

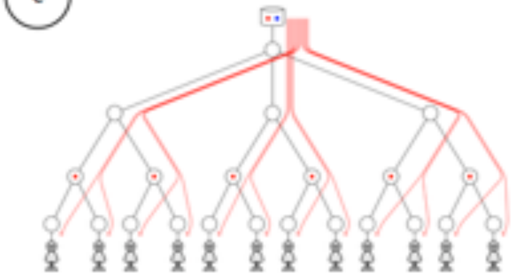
My contributions

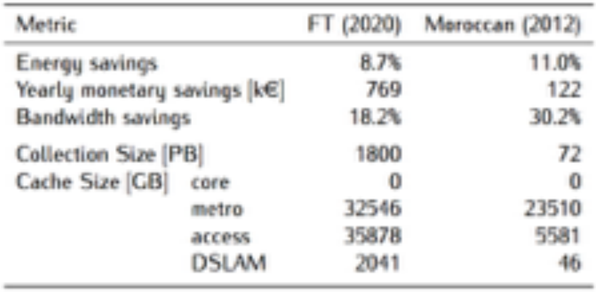
- Practical Scenarios [Chap. 3]
  - Study of ISP networks [Chap. 3.1]
  - Using redundancy elimination [Chap. 3.2]
  - For content delivery [Chap. 3.3]

- Using Algorithm Complexity and Graph Theory
  - Hardness results (No-APX) [Chap. 3.1]
  - Theoretical bounds for specific topologies (grids, rings, trees, etc…) and all-to-all [Chap. 4]



Level(3)
HIGHWINDS
Akamai
Limelight
Ustream





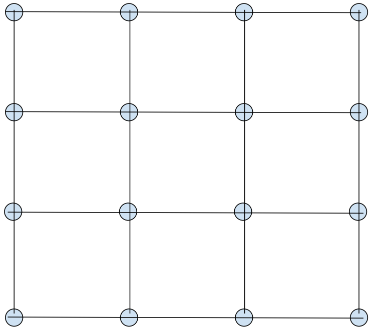
Metric		FT (2020)	Moroccan (2012)
Energy savings		8.7%	11.0%
Yearly monetary savings [k€]		769	122
Bandwidth savings		18.2%	30.2%
Collection Size [PB]		1800	72
Cache Size [GB]	core	0	0
	metro	32546	23510
	access	35878	5581
	DSLAM	2041	46

[CompJ 2016]

[Globecom ICC 2013]

My contributions

- Practical Scenarios [Chap. 3]
  - Study of ISP networks [Chap. 3.1]
  - Using redundancy elimination [Chap. 3.2]
  - For content delivery [Chap. 3.3]



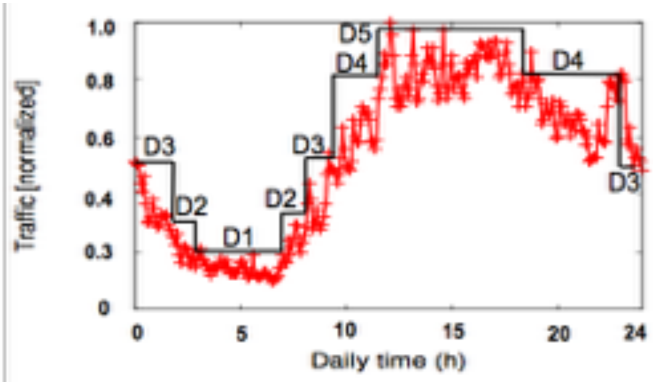
- Using Algorithm Complexity and Graph Theory
  - Hardness results (No-APX) [Chap. 3.1]
  - Theoretical bounds for specific topologies (grids, rings, trees, etc...) and all-to-all [Chap. 4]
    - Grid [Chap 4.2]
    - Given a number of edges and nodes, what are the graphs with the lowest forwarding indices? [Chap. 4.3]

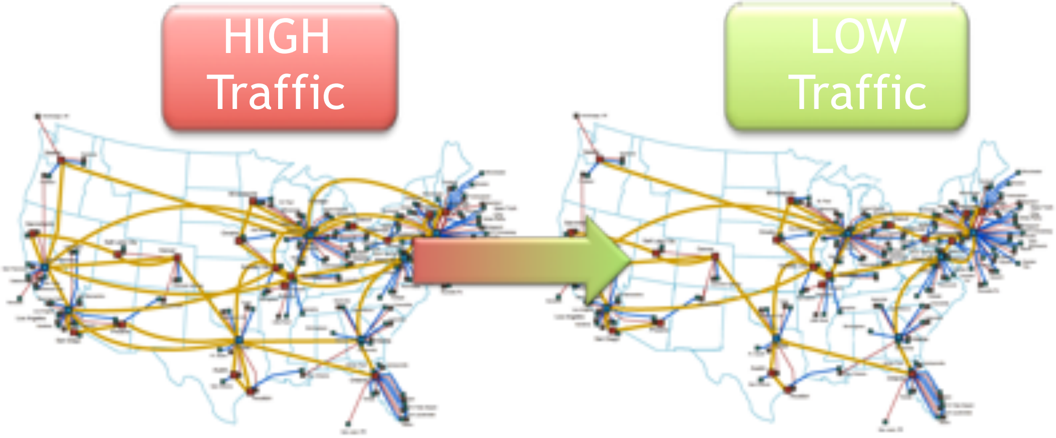
[DAM 2018]

[Inoc Iwoca 2015]

Energy Efficiency

- Core of solutions for energy efficiency: dynamic adaptation of resource usage to traffic changes.

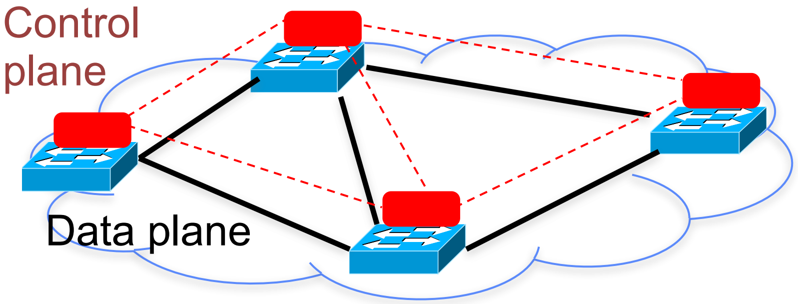




Other applications: energy efficient data centers (virtual machine assignment), wireless networks (base-station assignment)…

Legacy networks

However, network operators reluctant to change the routing.



- **Router=closed systems.** Any change has to be done **manually.**

- Networks are managed by complex **configurations.**

—> Important difficulties to deploy new protocols

-> Energy efficient solutions **not yet successfully implemented** in networks.

Routing: A new Context

- Career Routing. Introduction

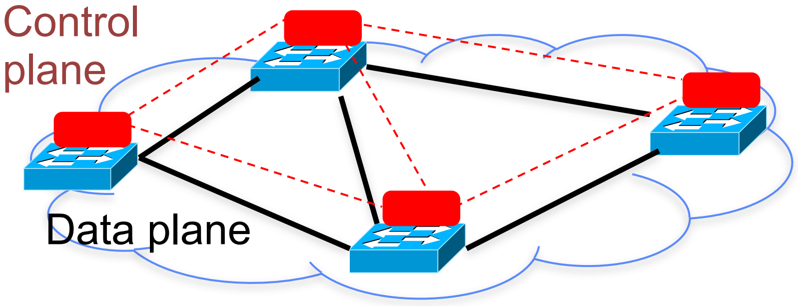
1. Routing in an energy aware world.
   (Part I)

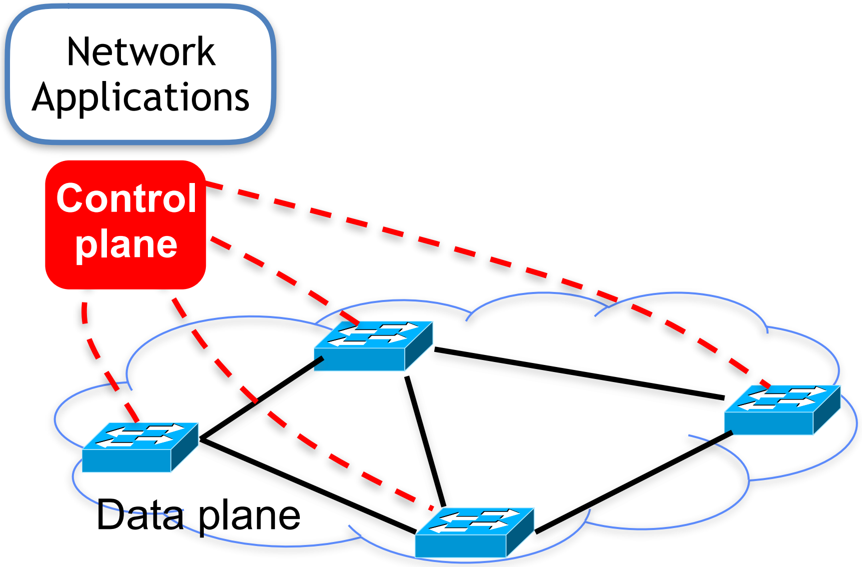
2. Routing in an SDN world without (a lot of) rules.
   (Part 2. Chapter 6)

3. Routing in a virtualized world.
   (Part 2. Chapter 7)

- End of the route ? Conclusions and Perspectives

Software Defined Networks





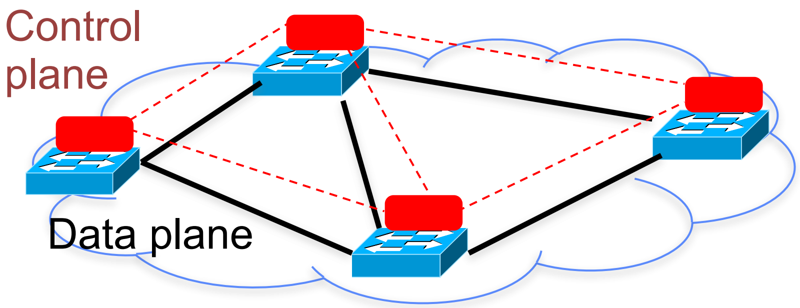
- **Router=closed systems.** Any change has to be done **manually.**
- Networks are managed by complex **configurations.**

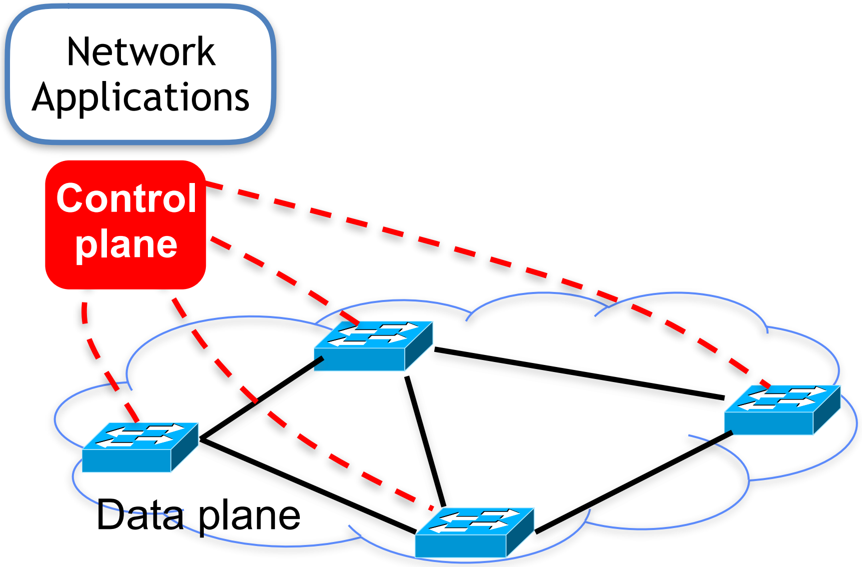
—> Important difficulties to deploy new protocols

- Intelligence implemented by a **centralized controller** managing **elementary** switches
- SDN conceives the network as a **program.**

—>Allows the deployment of advanced (dynamic) protocols

Software Defined Networks





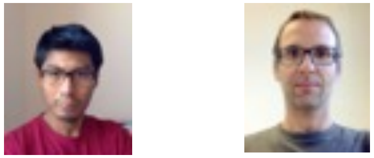
—> SDN has the potential to put into practice energy efficient solutions

- Intelligence implemented by a **centralized controller** managing **elementary** switches
- SDN conceives the network as a **program.**

—>Allows the deployment of advanced (dynamic) protocols

SDN and Energy efficiency

- Topic of a project between COATI and SIGNET



- Inside the axis Energy of labex UCN@Sophia
- Two Ph.D. students:
  - Nicolas Huin, 2014-2017
  - Myriana Rifai, 2014-2017

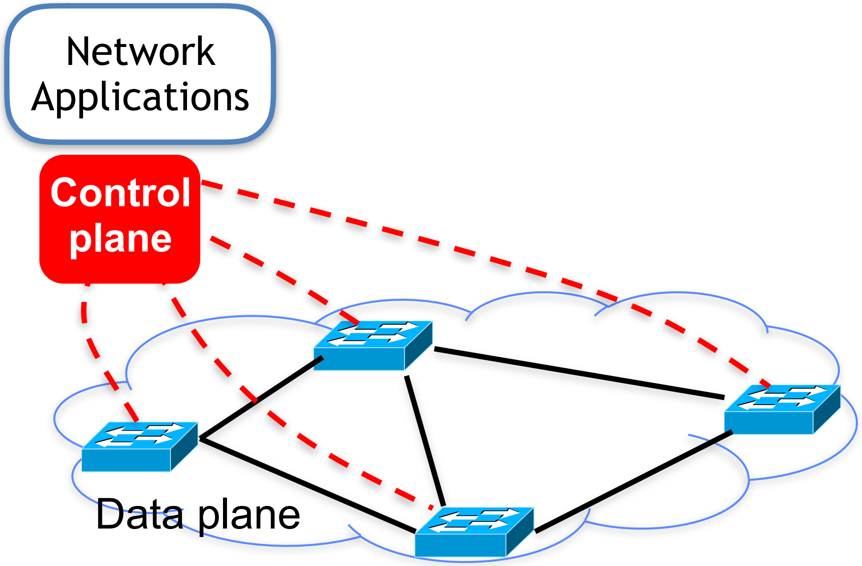
Software Defined Networks

- Pushed by open source communities + large software and telecommunication companies.
  - **Large eco-system:** Open Flow / Open Day Light / Open Stack / Open vSwitch
  - **Software companies:** Google B4 large scale experiment on its inter-data center networks [Jain 2013].

B4 worldwide deployment (2011)

  - **Telcos:** e.g. AT&T targets 75% of network functions as a software by 2020.

SDN Challenges



- **Defining the architecture.**
  - e.g. northbound **APIs** to enable real network programmability
- **Security**
  - e.g. single point of failure
- **Scalability of the SDN environment**
  - e.g. avoiding Control – Data Plane communications overhead

- **avoiding excessive flow table entries**

Problem

- SDN enables dynamic routing but at the cost of **limited forwarding table size.**

Legacy rule:

Destination	Output port

1 tuple

SDN rule:

Input port	Vlan ID	Vlan pcp	Src. MAC	Dst. MAC	Src. IP	Dst. IP	...	Action

40 tuples

- SDN rules are flow-oriented –> more complex
- SDN forwarding tables stored with **TCAM memory** which is expensive, power-hungry and with a limited size.

→ Constraint on number of forwarding rules (around 1000)

How do we deal with small rule tables?

- **Eviction** (e.g., **LRU**) or remove the least interesting rule when a new rule must be added.
  -> Frequent contact with controller

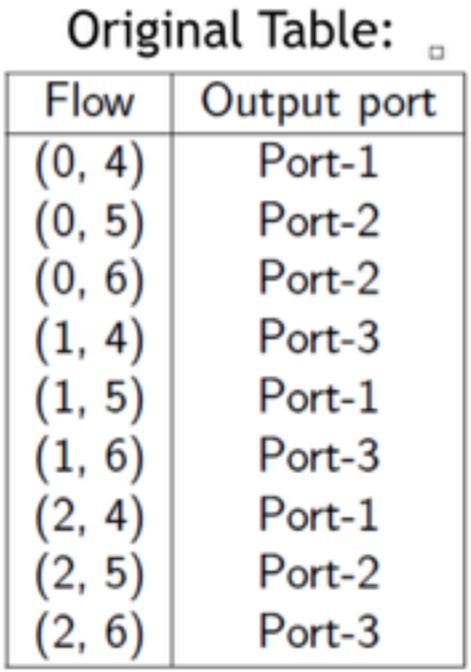
- **Split and distribute the rules** in network [cohen et al. 14]

- Use a minimum number of paths. Xpath: **Relabeling and aggregation of paths** [Hu et al, '15].
  -> Increased path length and thus delay

- Decrease rule size by matching only on **small tag in packet header** [Kannan et al, '13][Banerjee et al, '14].
  -> Need to modify end hosts.

- Our solution: **Compressing using wildcard rules.**

Compression problem

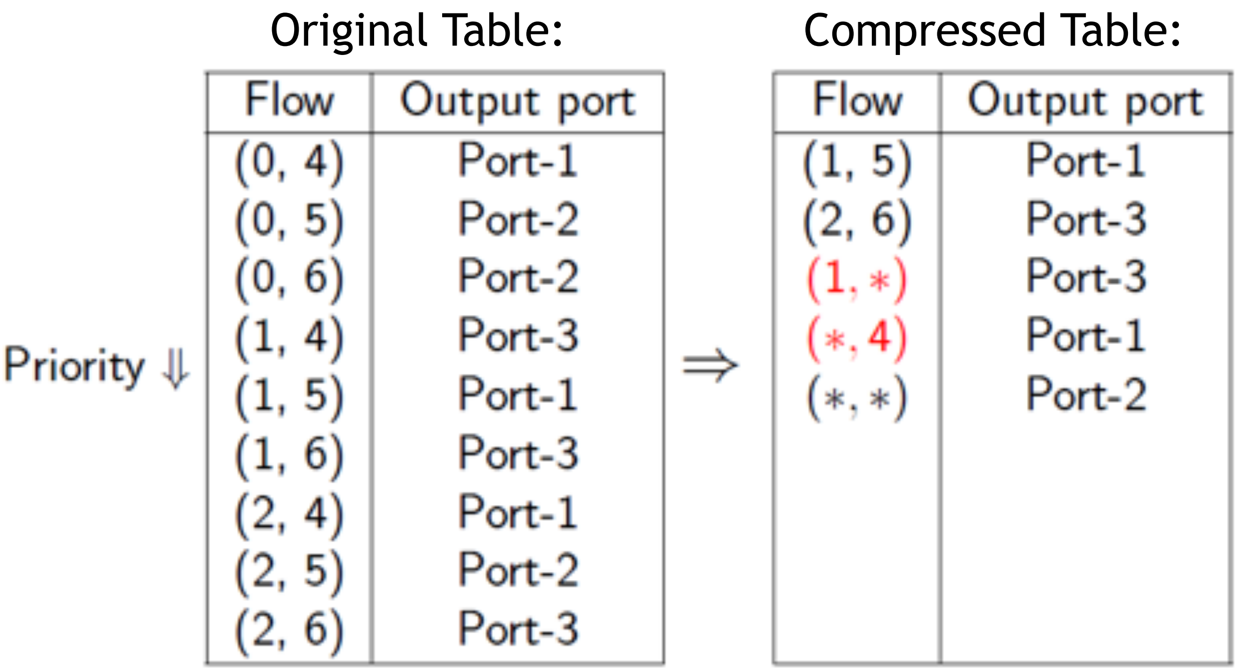


Original Table:

Flow	Output port
(0, 4)	Port-1
(0, 5)	Port-2
(0, 6)	Port-2
(1, 4)	Port-3
(1, 5)	Port-1
(1, 6)	Port-3
(2, 4)	Port-1
(2, 5)	Port-2
(2, 6)	Port-3

Reduce the size of the table using wildcard rules and default rule.

Compression problem



Original Table:

Priority ⇓

Flow	Output port
(0, 4)	Port-1
(0, 5)	Port-2
(0, 6)	Port-2
(1, 4)	Port-3
(1, 5)	Port-1
(1, 6)	Port-3
(2, 4)	Port-1
(2, 5)	Port-2
(2, 6)	Port-3

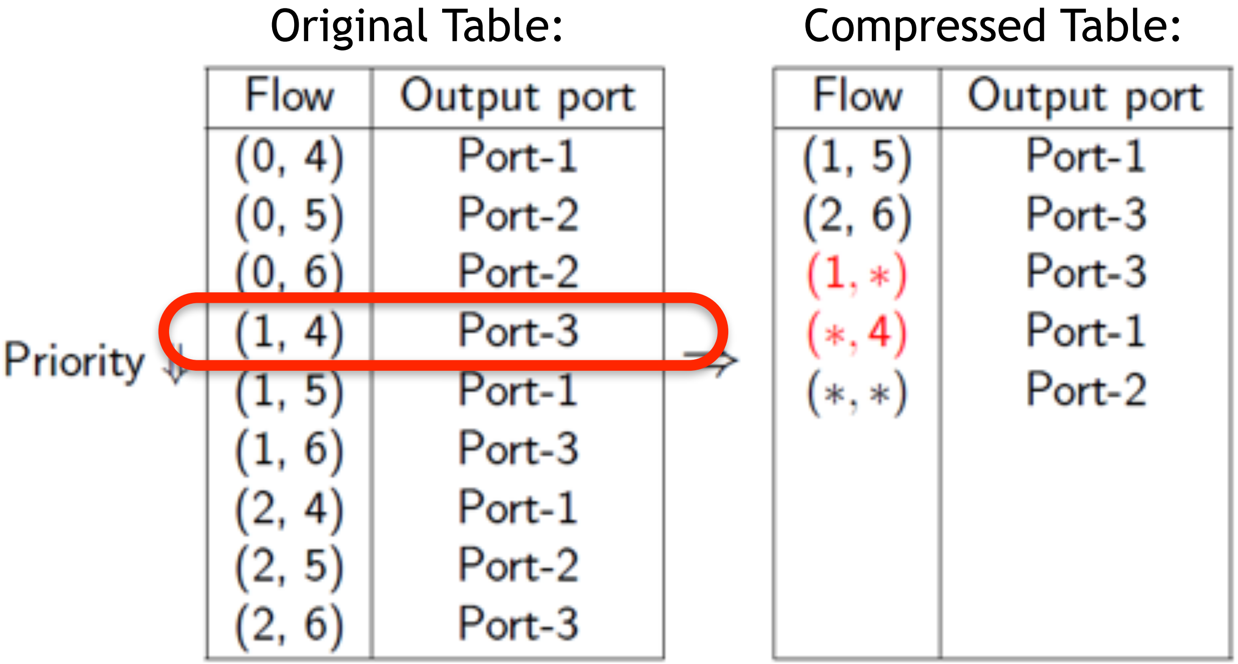
⇒

Compressed Table:

Flow	Output port
(1, 5)	Port-1
(2, 6)	Port-3
(1, ∗)	Port-3
(∗, 4)	Port-1
(∗, ∗)	Port-2

Reduce the size of the table using wildcard rules and default rule.

Compression problem



Original Table:

Flow	Output port
(0, 4)	Port-1
(0, 5)	Port-2
(0, 6)	Port-2
(1, 4)	Port-3
(1, 5)	Port-1
(1, 6)	Port-3
(2, 4)	Port-1
(2, 5)	Port-2
(2, 6)	Port-3

Priority ↓

⇒

Compressed Table:

Flow	Output port
(1, 5)	Port-1
(2, 6)	Port-3
(1, ∗)	Port-3
(∗, 4)	Port-1
(∗, ∗)	Port-2

Beware the order. The **first matching rule** is applied.

Example: If `(*,4)`→`1` is before `(1,*)`→`3`, then `(1,4)` will be routed through `1`, and not `3`.

Contributions

Problem: how to routing using compression while minimizing energy consumption.

My Method

- Algorithmic complexity
- Efficient solutions
  - with proved warranties
  - Optimization methods
  - Heuristic solutions
  - Decompose the problem
- Test in practice
  - Simulations
  - Experimentations

Contributions

Problem: how to routing using compression while minimizing energy consumption.

1. Study of the algorithmic problem of compressing a two dimensional routing table using wildcard rules with an order on the rules. [Algorithmica 2018. Short version INOC 2015]

My Method

- Algorithmic complexity
- Efficient solutions
  - with proved warranties
  - Optimization methods
  - Heuristic solutions
  - Decompose the problem
- Test in practice
  - Simulations
  - Experimentations

Contributions

Problem: how to routing using compression while minimizing energy consumption.

1. Study of the algorithmic problem of compressing a two dimensional routing table using wildcard rules with an order on the rules. [Algorithmica 2018. Short version INOC 2015]
2. Modeling and Simulations on ISP networks. [Computer Communications 2018 . Short version Globecom 2014]

My Method

- Algorithmic complexity
- Efficient solutions
  - with proved warranties
  - Optimization methods
  - Heuristic solutions
  - Decompose the problem
- Test in practice
  - Simulations
  - Experimentations

Contributions

Problem: how to routing using compression while minimizing energy consumption.

1. Study of the algorithmic problem of compressing a two dimensional routing table using wildcard rules with an order on the rules. [Algorithmica 2018. Short version INOC 2015]
2. Modeling and Simulations on ISP networks. [Computer Communications 2018 . Short version Globecom 2014]
3. Experiments for an SDN data center network. [Computer Networks 2018. Short version Globecom 2015]

My Method

- Algorithmic complexity
- Efficient solutions
  - with proved warranties
  - Optimization methods
  - Heuristic solutions
  - Decompose the problem
- Test in practice
  - Simulations
  - Experimentations

Contribution 1: Compressing two dimensional routing tables with orders

- **Algorithmic complexity**
  - Link with a classic graph problem, **Feedback Arc Set**.
  - Hardness results:
    - Polynomial pour 1 port
    - **NP-complete** 2 ports or more

Solved on open problem stated in [Suri et al. Algorithmica 2003]

- **Efficient solutions with proved warranties**
  - Approximation algorithms
    - a **simple 3-approximation** for List Reduction: Direction-Based Heuristic.
    - a 4-approximation for Routing List.
  - Study of the Fixed Parameter Tractability (FPT). **Polynomial kernels** for most of the problems considered.

Contribution 1: Compressing two dimensional routing tables with orders

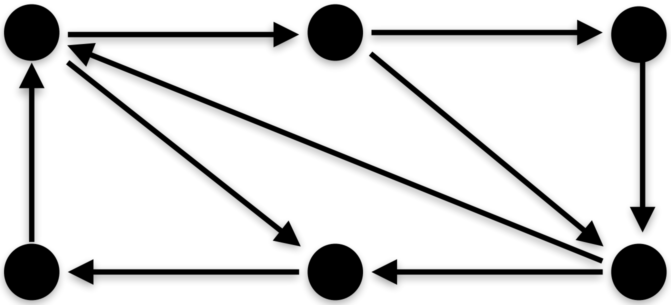
- **Algorithmic complexity**
  - Link with a classic graph problem, **Feedback Arc Set**.
  - Hardness results:
    - Polynomial pour 1 port
    - **NP-complete** 2 ports or more

Solved on open problem stated in [Suri et al. Algorithmica 2003]

- **Efficient solutions with proved warranties**
  - Approximation algorithms
    - a **simple 3-approximation** for List Reduction: Direction-Based Heuristic.
    - a 4-approximation for Routing List.
  - Study of the Fixed Parameter Tractability (FPT). **Polynomial kernels** for most of the problems considered.

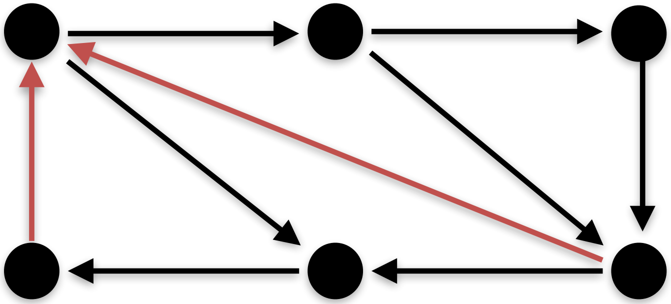
Contribution 1: Compressing two dimensional routing tables with orders

Feedback Arc Set: set of edges removing all cycles in a digraph



Contribution 1: Compressing two dimensional routing tables with orders

Feedback Arc Set: set of edges removing all cycles in a digraph



Decision problem: “can all cycles be broken by removing k edges?” is one of Karp’s NP-complete problem.

Contribution 1: Compressing two dimensional routing tables with orders

Link with Feedback Arc Set: A simple example with 2 ports

(Src, Dst)	Output port
(1, 2)	**0**
(1, 4)	**0**
(3, 1)	**0**
(3, 2)	**1**
(3, 4)	**1**
(4, 5)	**0**
(6, 1)	**1**
(6, 4)	**0**
(6, 5)	**0**

Representation as a matrix

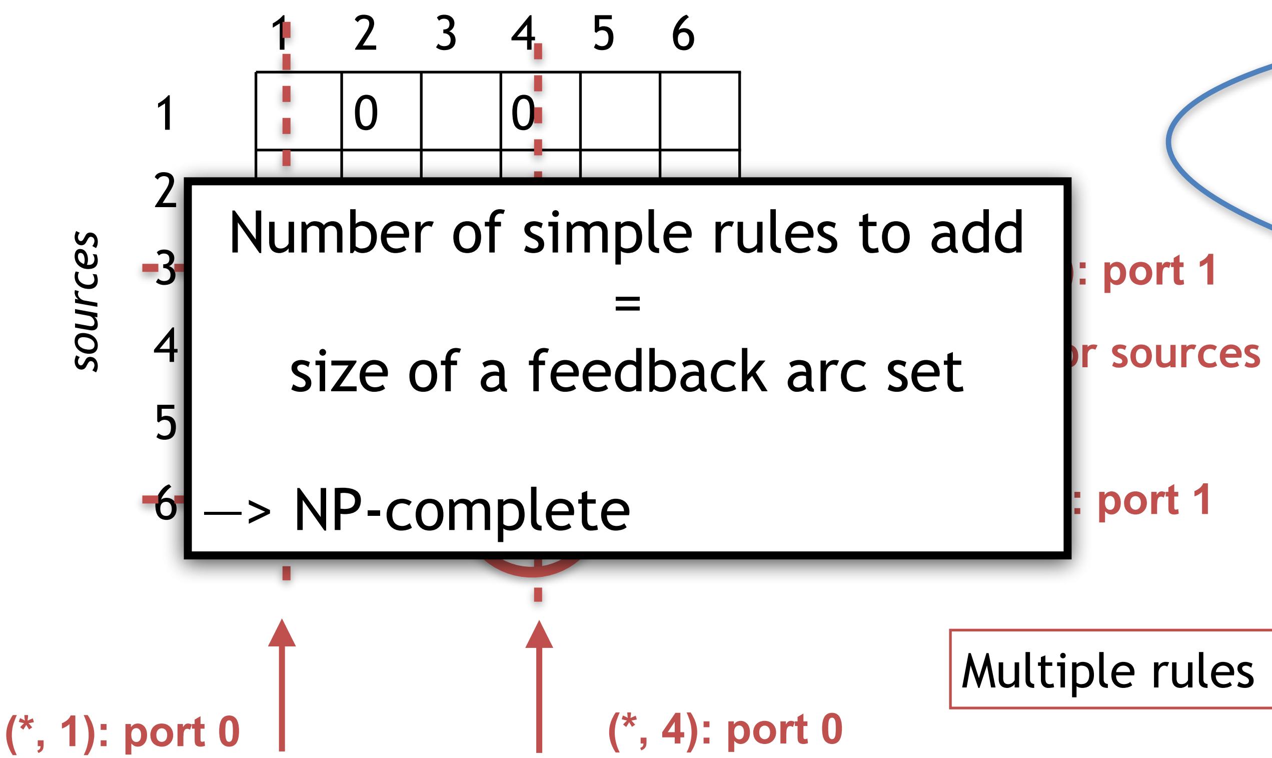
destinations

sources

	1	2	3	4	5	6
1		0		0		
2						
3	0	1		1		
4					0	
5						
6	1			0		0

Contribution 1: Compressing two dimensional routing tables with orders

A simple example with 2 ports



Number of simple rules to add

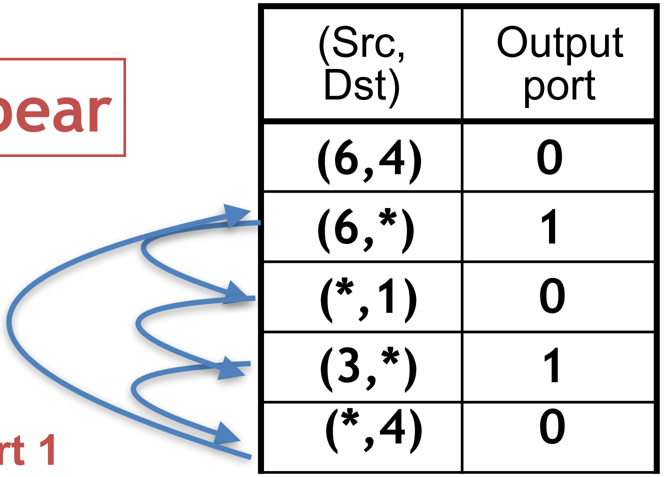
=

size of a feedback arc set

—> NP-complete

Aggregation for destinations

Cycle appear



(Src, Dst)	Output port
(6,4)	**0**
(6,*)	**1**
(*,1)	**0**
(3,*)	**1**
(*,4)	**0**

Multiple rules -> importance of order

Contribution 1: Compressing two dimensional routing tables with orders

- **Algorithmic complexity**
  - Link with a classic graph problem, **Feedback Arc Set.**
  - Hardness results:
    - Polynomial pour 1 port
    - **NP-complete** 2 ports or more

Solved on open problem stated in [Suri et al. Algorithmica 2003]

- **Efficient solutions with proved warranties**
  - Approximation algorithms
    - a **simple 3-approximation** for List Reduction: Direction-Based Heuristic.
    - a 4-approximation for Routing List.

Used for practical applications presented later

  - Study of the Fixed Parameter Tractability (FPT). **Polynomial kernels** for most of the problems considered.

Direction-Based Heuristic

Compress using source aggregation, destination aggregation or default rule
$\Rightarrow$ Take the best table

- For each source `s`, **# source aggregation**
  - Find the most occurring port k.
  - Add the aggregation rule `(s,*):k`
  - Keep the non matching rules `(s,t):k'` with higher priority

  -> Table T_s

- Do the same for each destination `t` **# destination aggregation**

  -> Table T_t

- Find the most occurring port **# default port**
  Add the aggregation rule `(s,*):k`
  Keep the non matching rules `(s,t):k'` with higher priority

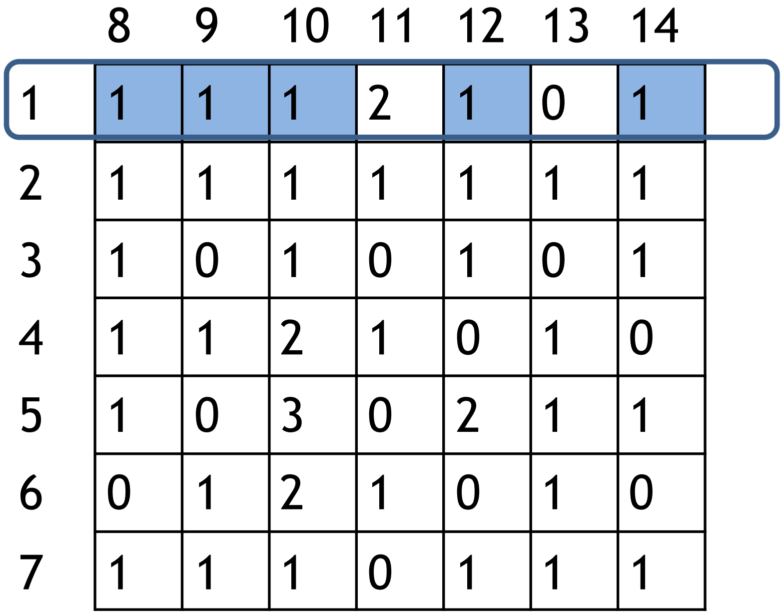
  -> Table T_d

- Choose min(T_s,T_t, T_d)

Direction-Based Heuristic

	8	9	10	11	12	13	14
1	1	1	1	2	1	0	1
2	1	1	1	1	1	1	1
3	1	0	1	0	1	0	1
4	1	1	2	1	0	1	0
5	1	0	3	0	2	1	1
6	0	1	2	1	0	1	0
7	1	1	1	0	1	1	1

Direction-Based Heuristic



	8	9	10	11	12	13	14
1	1	1	1	2	1	0	1
2	1	1	1	1	1	1	1
3	1	0	1	0	1	0	1
4	1	1	2	1	0	1	0
5	1	0	3	0	2	1	1
6	0	1	2	1	0	1	0
7	1	1	1	0	1	1	1

(1,11)→2
(1,13)→0
(1,*)→1

(Src, Dst)	Output
(1, 11)	**2**
(1, 13)	**0**
(1, *)	**1**

Direction-Based Heuristic

	8	9	10	11	12	13	14
1	1	1	1	2	1	0	1
2	1	1	1	1	1	1	1
3	1	0	1	0	1	0	1
4	1	1	2	1	0	1	0
5	1	0	3	0	2	1	1
6	0	1	2	1	0	1	0
7	1	1	1	0	1	1	1

(2,*)→1

(Src, Dst)	Output
(1, 11)	**2**
(1, 13)	**0**
(1, *)	**1**
(2, *)	**1**

Direction-Based Heuristic

	8	9	10	11	12	13	14
1	1	1	1	2	1	0	1
2	1	1	1	1	1	1	1
3	1	0	1	0	1	0	1
4	1	1	2	1	0	1	0
5	1	0	3	0	2	1	1
6	0	1	2	1	0	1	0
7	1	1	1	0	1	1	1

(3,9)→0
(3,11)→0
(3→13)→0
(3,*)→1

(Src, Dst)	Output
(1, 11)	**2**
(1, 13)	**0**
(1, *)	**1**
(2, *)	**1**
(3,9)	**0**
(3,11)	**0**
(3,13)	**0**
(3,*)	**1**

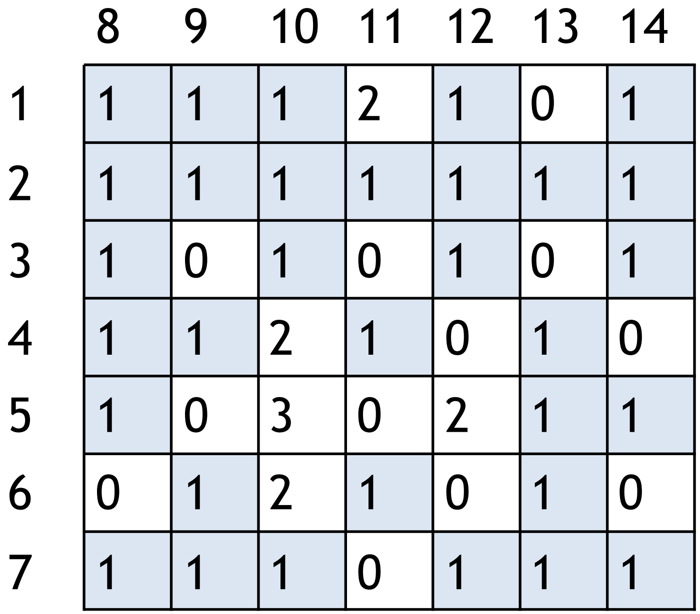
Direction-Based Heuristic

	8	9	10	11	12	13	14
1	1	1	1	2	1	0	1
2	1	1	1	1	1	1	1
3	1	0	1	0	1	0	1
4	1	1	2	1	0	1	0
5	1	0	3	0	2	1	1
6	0	1	2	1	0	1	0
7	1	1	1	0	1	1	1

(6,8)→0
(*,8)→1

The heuristic does the same for the destinations.

Direction-Based Heuristic



	8	9	10	11	12	13	14
1	1	1	1	2	1	0	1
2	1	1	1	1	1	1	1
3	1	0	1	0	1	0	1
4	1	1	2	1	0	1	0
5	1	0	3	0	2	1	1
6	0	1	2	1	0	1	0
7	1	1	1	0	1	1	1

Table :
Non matching rules
Default rule (*,*)→1

Theorem: This gives a 3-approximation of the List-reduction problem:

Input: A set C of communication triples and integer z

Output: sav(C) is the maximum number of saved rules.

Contribution 1: Compressing two dimensional routing tables with orders

- **Algorithmic complexity**
  - Link with a classic graph problem, **Feedback Arc Set.**
  - Hardness results:
    - Polynomial pour 1 port
    - **NP-complete** 2 ports or more

Solved on open problem stated in [Suri et al. Algorithmica 2003]

- **Efficient solutions with proved warranties**
  - Approximation algorithms
    - a **simple 3-approximation** for List Reduction: Direction-Based Heuristic.
    - a 4-approximation for Routing List.

Used for practical applications presented later

- Study of the Fixed Parameter Tractability (FPT). **Polynomial kernels** for most of the problems considered.

FPT Algorithms

FPT **Algorithms**: computation time in

$$f(z) \cdot |x|^{O(1)}$$

with

- z (well chosen) parameter,
- f any function.

⇒ if z is small, inputs can be solved by brute force.

Theorem: For every l ≥ 1, List-Reduction-with-k-ports admits a linear kernel and so is FPT.

A rule is **isolated** if its shares neither source nor destination.

Kernelization algorithm:

1- Remove isolated rules ⇒ table S_1.

2- Apply the Destination-Based Heuristic ⇒ table T .

3- If $|S_1| - |T|$ = $sav_{DBH}(S_1) \geq z$, return 'Yes'.
Else return T.

Proof of validity: If T is returned, we show that $|T| \leq (4z - 4)k$.

Contributions

Problem: how to routing using compression while minimizing energy consumption.

1. Study of the algorithmic problem of compressing a two dimensional routing table using wildcard rules with an order on the rules. [Algorithmica 2018. Short version INOC 2015]
2. Modeling and Simulations on ISP networks. [Computer Communications 2018 . Short version Globecom 2014]
3. Experiments for an SDN data center network. [Computer Networks 2018. Short version Globecom 2015]

My Method

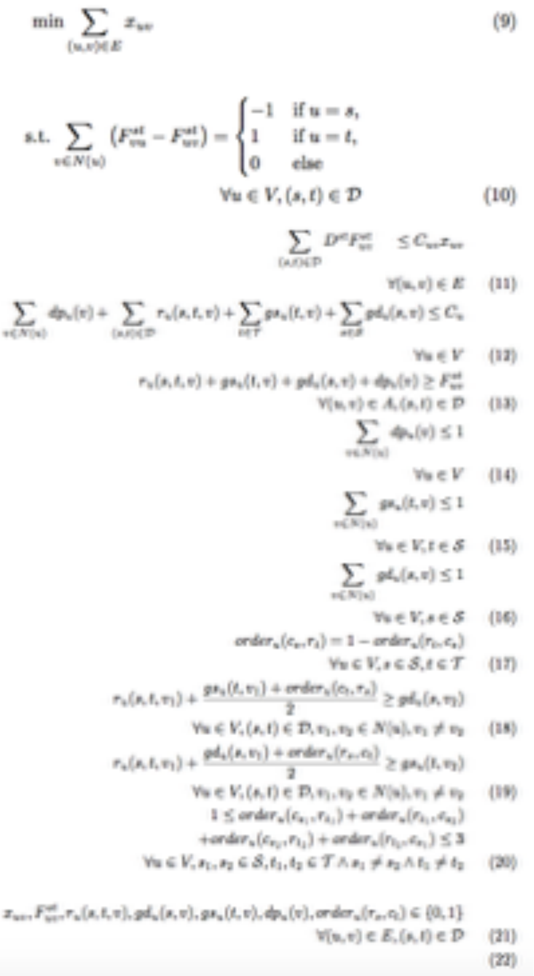
- Algorithmic complexity
- Efficient solutions
  - with proved warranties
  - Optimization methods
  - Heuristic solutions
  - Decompose the problem
- Test in practice
  - Simulations
  - Experimentations

Contribution 2: Energy efficient routing in an SDN ISP network.

Problem can be modeled by an **ILP.**

The ILP **runs for a small network:** Atlanta.

Propose several **efficient heuristic solutions** (using different compression methods) for larger networks.



$$\min \sum_{(u,v)\in E} x_{uv} \tag{9}$$

$$\text{s.t.} \sum_{v\in N(u)} \left(F^{st}_{vu} - F^{st}_{uv}\right) = \begin{cases} -1 & \text{if } u = s, \\ 1 & \text{if } u = t, \\ 0 & \text{else} \end{cases} \qquad \forall u \in V, (s,t) \in \mathcal{D} \tag{10}$$

$$\sum_{(s,t)\in\mathcal{D}} D^{st} F^{st}_{uv} \le C_{uv} x_{uv} \qquad \forall (u,v) \in E \tag{11}$$

$$\sum_{v\in N(u)} dp_u(v) + \sum_{(s,t)\in\mathcal{D}} r_u(s,t,v) + \sum_{t\in\mathcal{T}} ga_u(t,v) + \sum_{s\in\mathcal{S}} gd_u(s,v) \le C_u \qquad \forall u \in V \tag{12}$$

$$r_u(s,t,v) + ga_u(t,v) + gd_u(s,v) + dp_u(v) \ge F^{st}_{uv} \qquad \forall (u,v) \in A, (s,t) \in \mathcal{D} \tag{13}$$

$$\sum_{v\in N(u)} dp_u(v) \le 1 \qquad \forall u \in V \tag{14}$$

$$\sum_{v\in N(u)} ga_u(t,v) \le 1 \qquad \forall u \in V, t \in \mathcal{S} \tag{15}$$

$$\sum_{v\in N(u)} gd_u(s,v) \le 1 \qquad \forall u \in V, s \in \mathcal{S} \tag{16}$$

$$order_u(c_s, r_t) = 1 - order_u(r_t, c_s) \qquad \forall u \in V, s \in \mathcal{S}, t \in \mathcal{T} \tag{17}$$

$$r_u(s,t,v_1) + \frac{ga_u(t,v_1) + order_u(c_s, r_t)}{2} \ge gd_u(s,v_2) \qquad \forall u \in V, (s,t) \in \mathcal{D}, v_1, v_2 \in N(u), v_1 \ne v_2 \tag{18}$$

$$r_u(s,t,v_1) + \frac{gd_u(s,v_1) + order_u(r_t, c_s)}{2} \ge ga_u(t,v_2) \qquad \forall u \in V, (s,t) \in \mathcal{D}, v_1, v_2 \in N(u), v_1 \ne v_2 \tag{19}$$

$$1 \le order_u(c_{s_1}, r_{t_1}) + order_u(r_{t_1}, c_{s_2}) + order_u(c_{s_2}, r_{t_2}) + order_u(r_{t_2}, c_{s_1}) \le 3 \qquad \forall u \in V, s_1, s_2 \in \mathcal{S}, t_1, t_2 \in \mathcal{T} \wedge s_1 \ne s_2 \wedge t_1 \ne t_2 \tag{20}$$

$$x_{uv}, F^{st}_{uv}, r_u(s,t,v), gd_u(s,v), ga_u(t,v), dp_u(v), order_u(r_s, c_t) \in \{0,1\} \qquad \forall (u,v) \in E, (s,t) \in \mathcal{D} \tag{21}$$

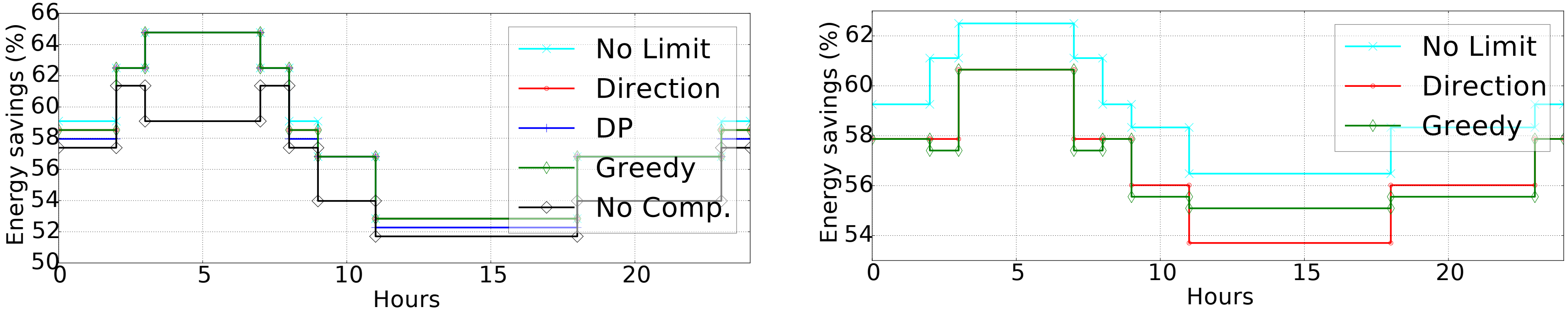
(22)

(a) atlanta (b) germany50 (c) zib54

(d) ta2

Contribution 2: Energy efficient routing in an SDN ISP network.

Energy savings of different solutions for two networks



Zib54 (52 nodes, 216 links)

Ta2 (81 nodes, 162 links)

- **Take aways:**
  - **No feasible solutions** without compression for some networks (or with only the default port)
  - **With compression, results almost as good than without the limit due to TCAM memory** for SDN: between 52% and 65% of savings.

Contributions

Problem: how to routing using compression while minimizing energy consumption.

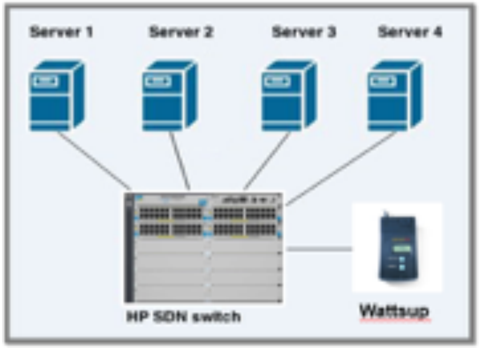
1. Study of the algorithmic problem of compressing a two dimensional routing table using wildcard rules with an order on the rules. [Algorithmica 2018. Short version INOC 2015]
2. Modeling and Simulations on ISP networks. [Computer Communications 2018 . Short version Globecom 2014]
3. Experiments for an SDN data center network. [Computer Networks 2018. Short version Globecom 2015]

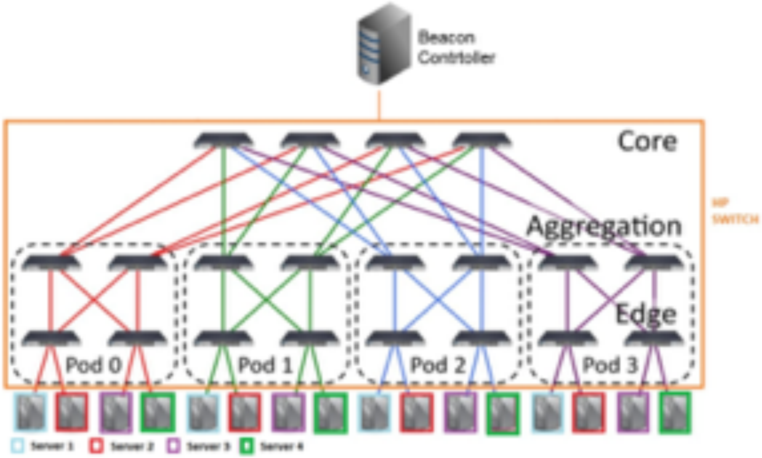
My Method

- Algorithmic complexity
- Efficient solutions
  - with proved warranties
  - Optimization methods
  - Heuristic solutions
  - Decompose the problem
- Test in practice
  - Simulations
  - Experimentations

Contribution 2: Experiments for a data center network

- Small experimental platform.

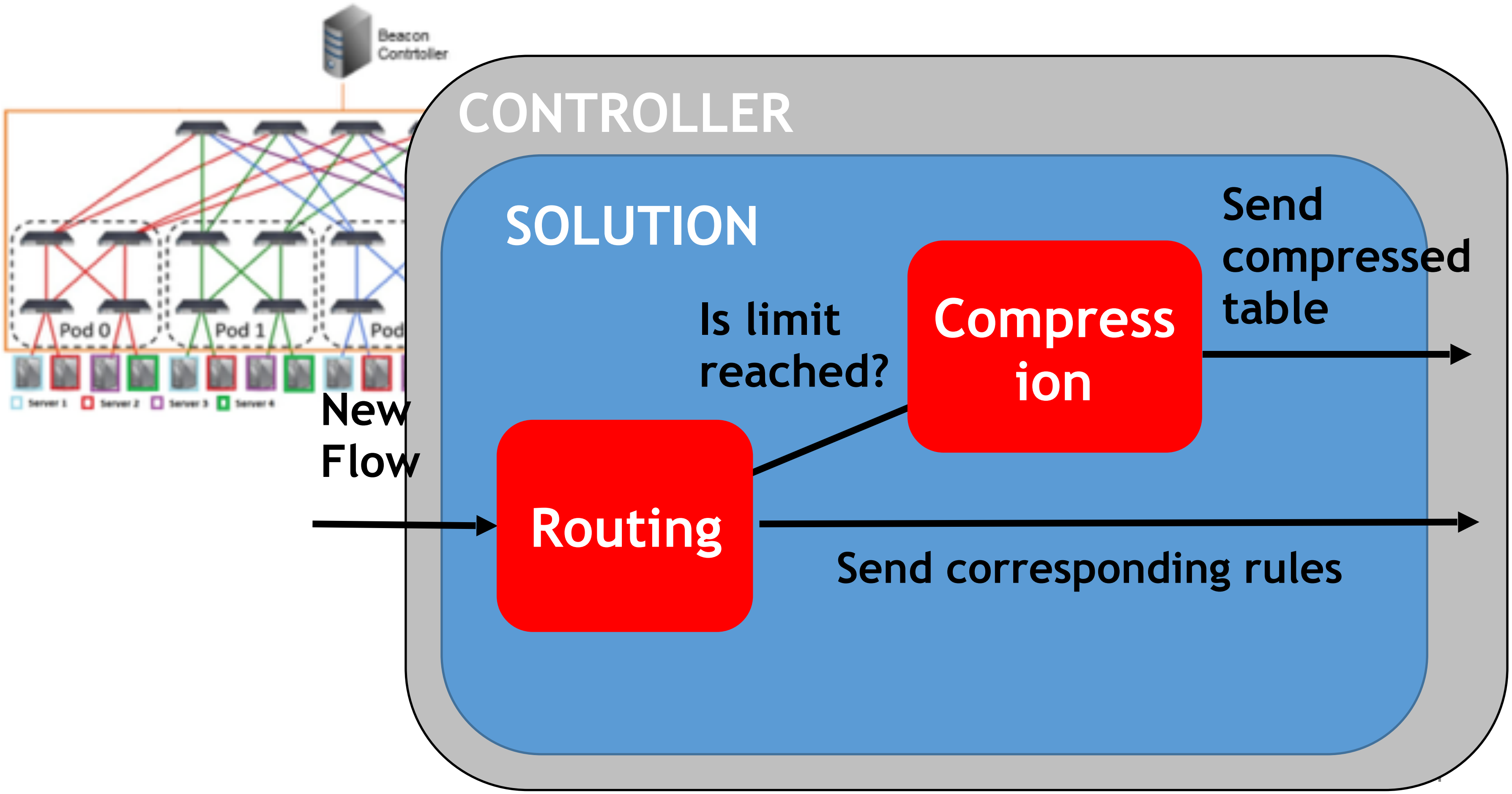




-> allowed to test impact of contacts to controller and controller optimizations

  - on packet delay
  - on packet losses

Contribution 2: Experiments for a data center network



CONTROLLER

SOLUTION

New Flow

Routing

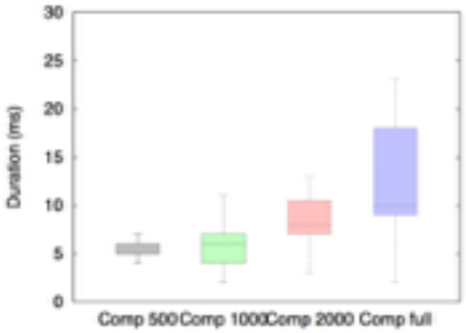
Is limit reached?

Compression

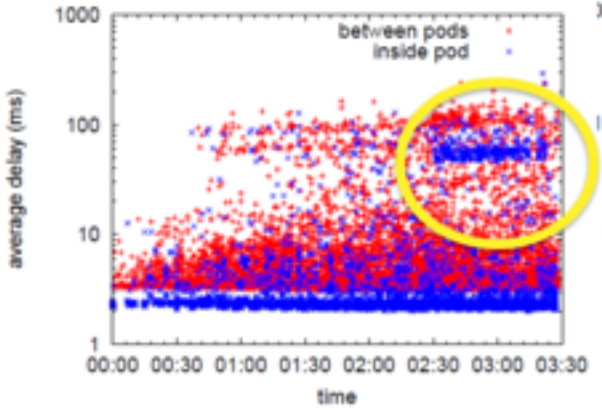
Send compressed table

Send corresponding rules

Contribution 2: Experiments for a data center network

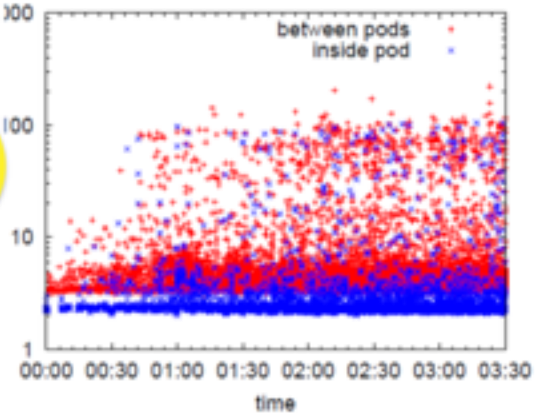


Time compression+ table modification = few ms



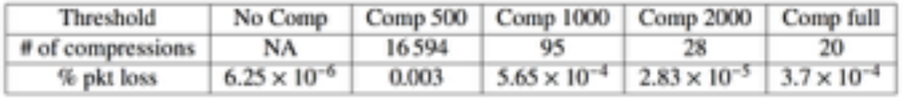
Without Compression

When the switch reaches its limit, no more rules installed
-> need to contact the controller for every packet received
-> high delay



With Compression

No problem. Delay is not increased.



Threshold	No Comp	Comp 500	Comp 1000	Comp 2000	Comp full
# of compressions	NA	16 594	95	28	20
% pkt loss	6.25×10^{-6}	0.003	5.65×10^{-4}	2.83×10^{-5}	3.7×10^{-4}

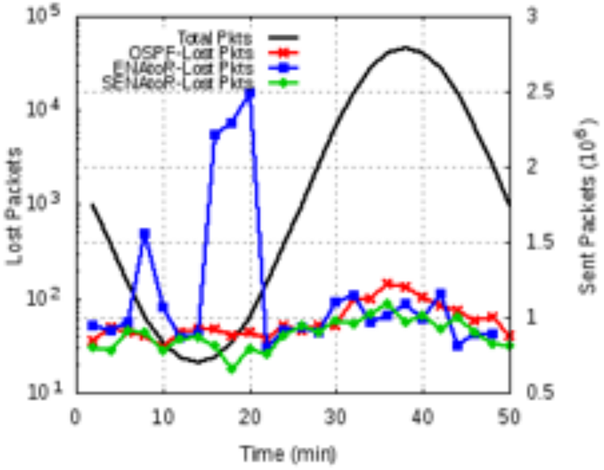
Loss rate ≈0%

Part 2 - Conclusions

- We provided efficient algorithms for doing multifield compression of routing tables: compression rates from 66 to 90%.

- Solutions for routing in an SDN world in datacenter or ISP networks
  - using compression
  - with smooth dynamic equipment extinction [Chap8]
    [Trans. Green Networking 18. Globecom 17.]

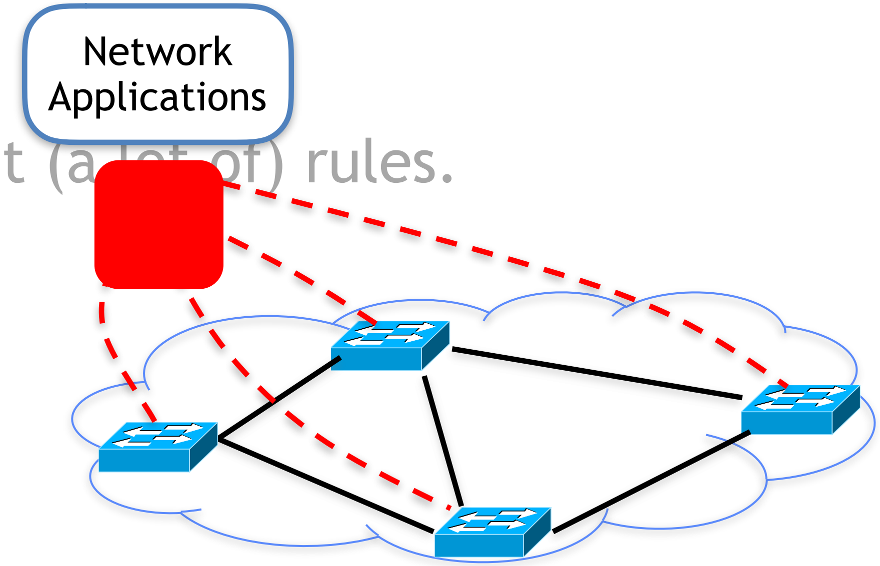
  - while preserving network stability.
    - no increased delay
    - no impact on failure rate
    - failure tolerant



Routing: A new Context

- Career Routing. Introduction

1. Routing in an energy aware world.
   (Part I)
2. Routing in an SDN world without (a lot of) rules.
   (Part 2. Chapter 6)
3. Routing in a virtualized world.
   (Part 2. Chapter 7)



- End of the route ? Conclusions and Perspectives

Network Function Virtualization

- Legacy networks implements network functions using expensive specific hardware called middleboxes.

- The NFV initiative allows functions to be run on general hardware using Virtual Machines.

Network Function Virtualization

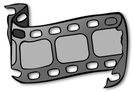
- Legacy networks implements network functions using expensive specific hardware called middleboxes.

- The NFV initiative allows functions to be run on general hardware using Virtual Machines.

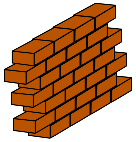
- Solve problems of cost, capacity rigidity, management complexity, and failures

Service Function Chaining

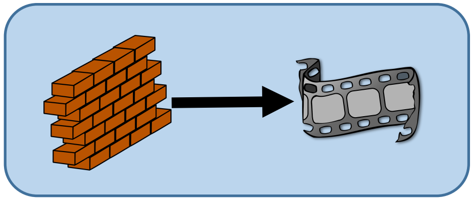
Service Function Chain: **ordered** chain of network functions to apply to flows on the network


Video optimization

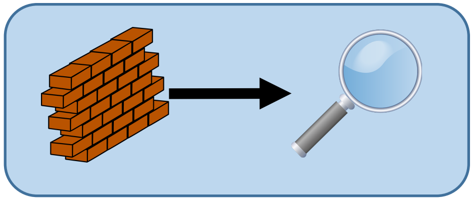

Deep packet inspection


Firewall

SFC A



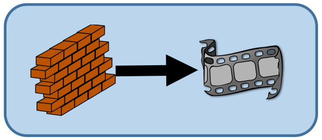
SFC B



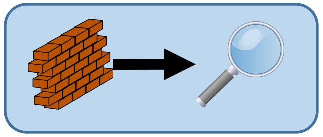
Problem: place VNFs to satisfy the **ordering constraints** of the flows with the goal of **minimizing the total setup cost** (e.g. license fees, network efficiency, or energy consumption)

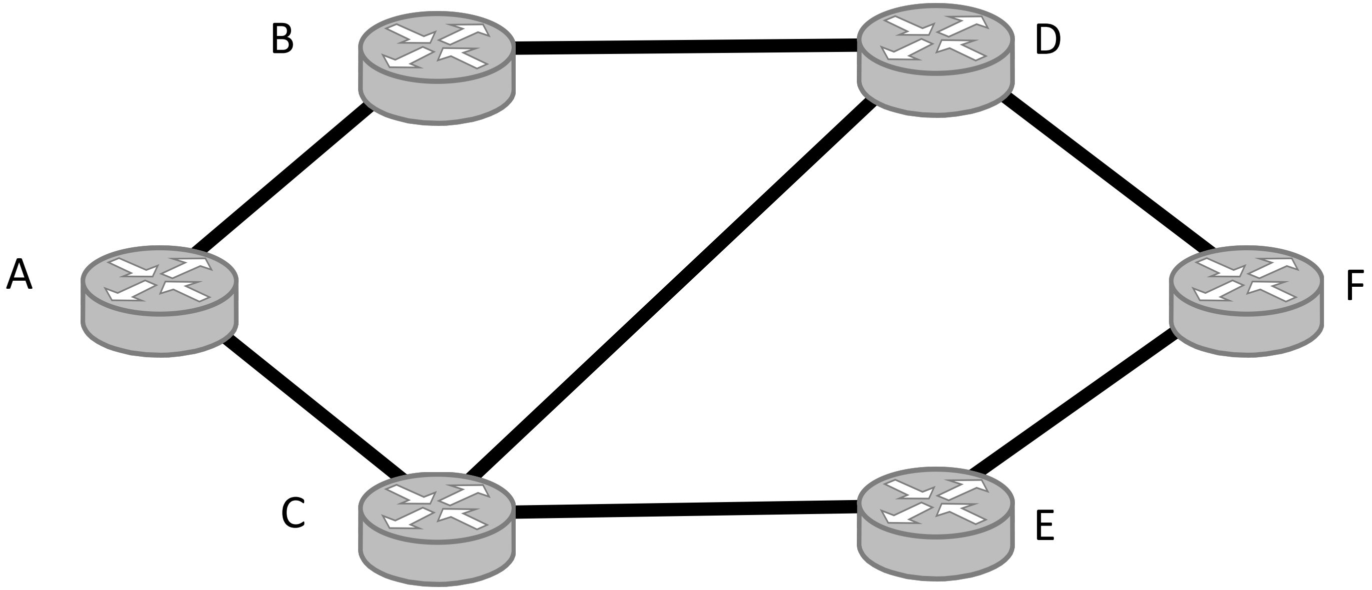
Example of Service Function Chains

3 flows: A to F — SFC A



A to E
F to C } SFC B





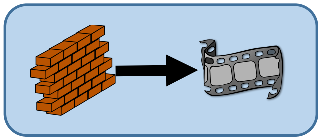
Example of Service Function Chains

3 flows: A to F

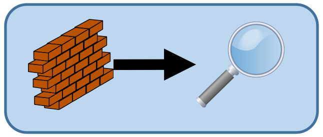
A to E

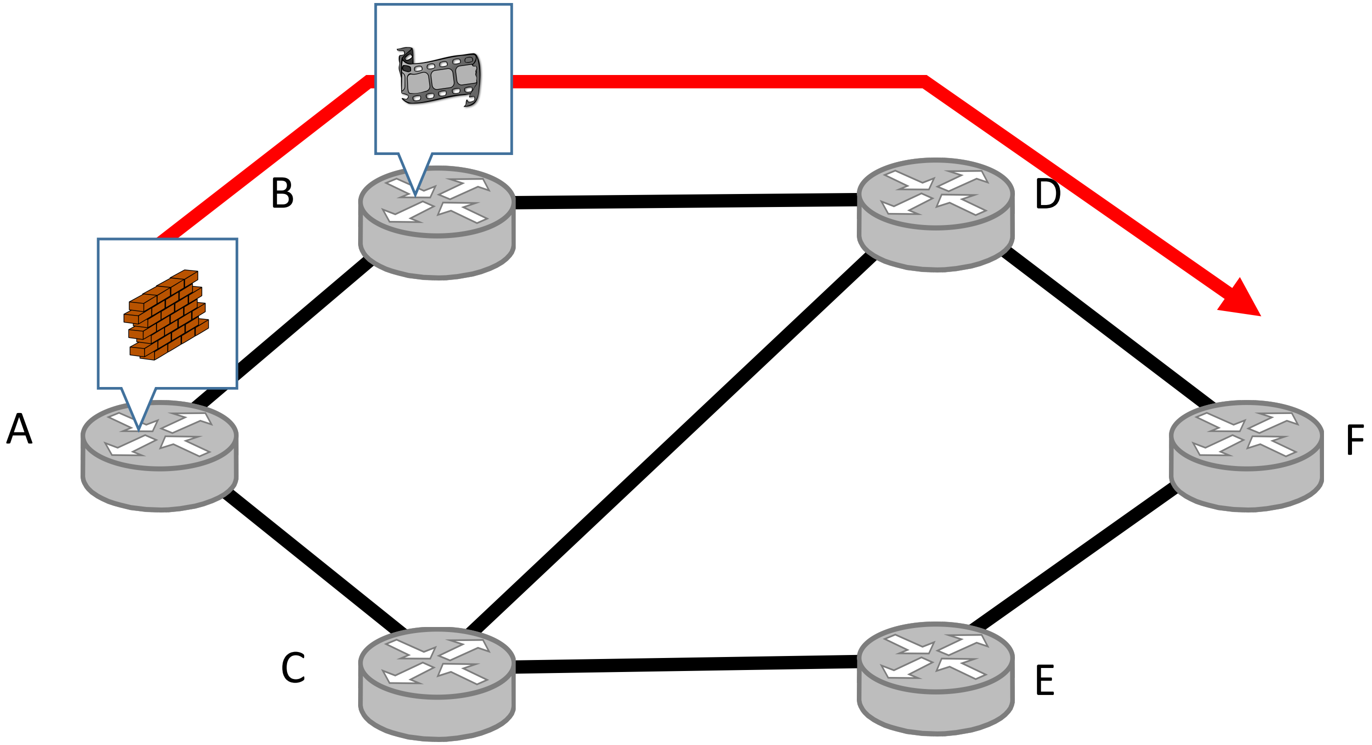
F to C

SFC A



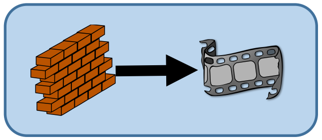
SFC B



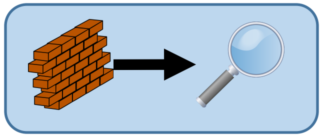


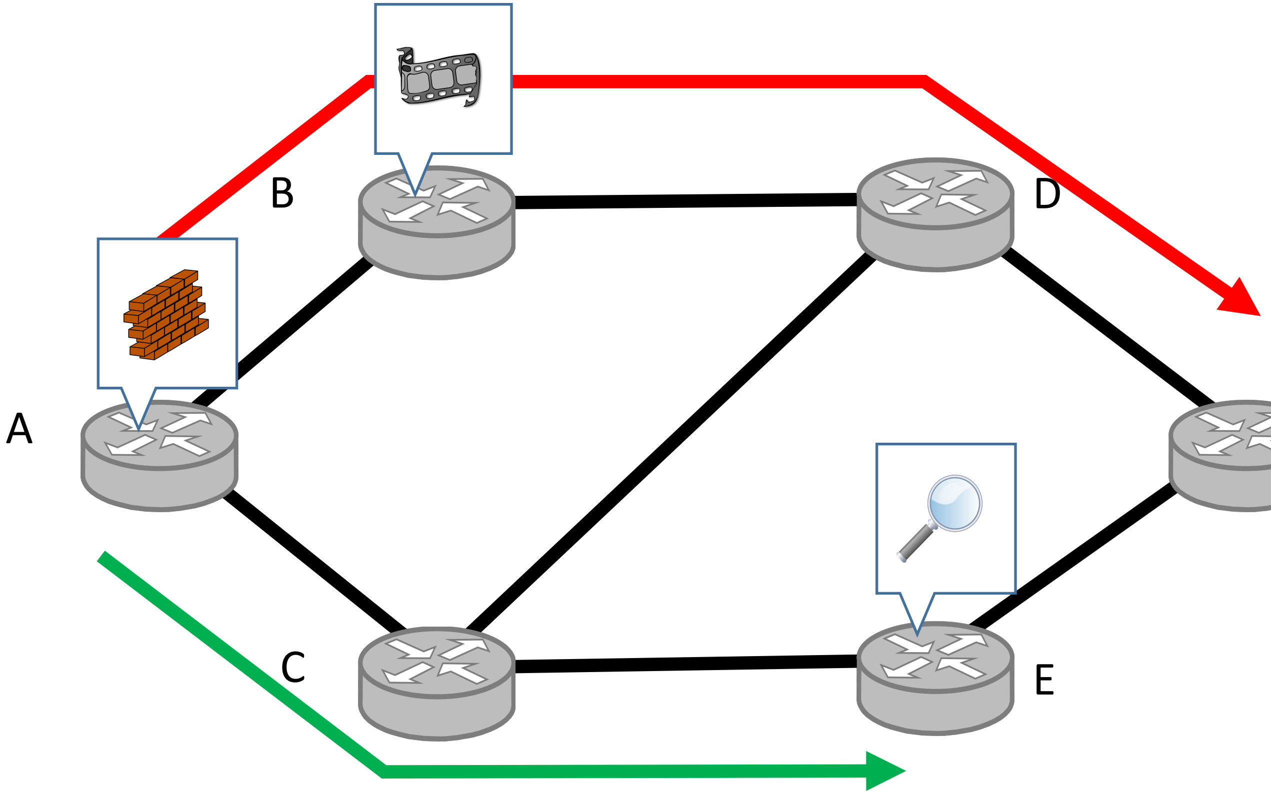
Example of Service Function Chains

3 flows: A to F — SFC A



A to E }
F to C } SFC B

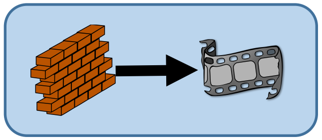


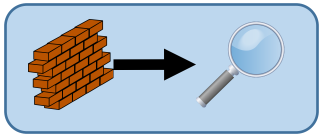


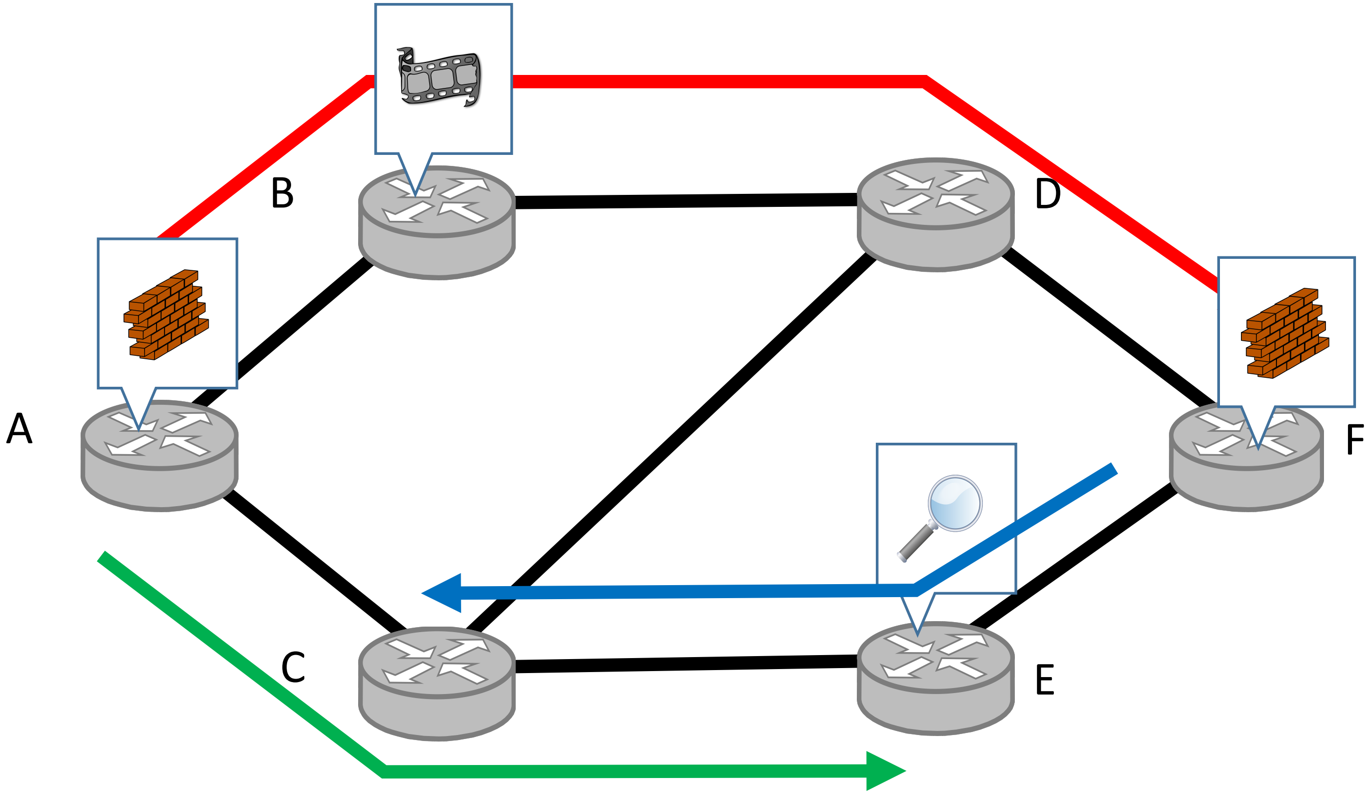
Example of Service Function Chains

3 flows: A to F SFC A

A to E }
F to C } SFC B







Related Work

- Heuristics
  - [Kuo et al. Infocom 2016] Maximizing the total number of admitted demands -> no warranties

- ILP based
  - [Mehraghdam et al. Cloudnet 2014] Minimizing the number of used nodes or the latency of the paths.

-> problem of scalability

- Approximation Algorithms
  - [Cohen et al. Infocom 2015] Minimizing setup cost near-optimal approximation algorithms with theoretically proven performance.
  - [Sang et al. Infocom 2017] Minimizing the total number of network functions. But one single network function.

-> *leave the placement of virtual functions with chaining constraint as an open problem.*

Contributions

- **Service Function Chain provisionning**
  1. using **Column Generation** [Several papers including ICC 2017-2018, ToN 2018]

     -> improved the scalability of ILP models

  2. with **Approximation Algorithms** [INFOCOM 2018]

     -> *"First approximation algorithms taking into account ordering constraints."*

     \+ optimal on trees + validation

Routing: A new Context

- Career Routing. Introduction

1. Routing in an energy aware world.
   (Part I)
2. Routing in an SDN world without (a lot of) rules.
   (Part 2. Chapter 6)
3. Routing in a virtualized world.
   (Part 2. Chapter 7)

- End of the route ? Conclusions and Perspectives

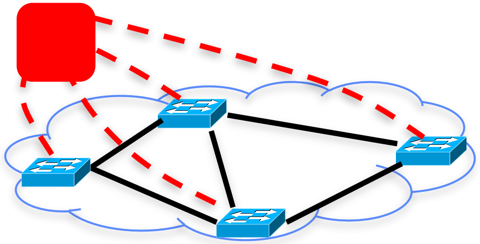
Conclusion

- I solved several problems of routing and placement of (virtual) resources.

- Lots remains to be done.

Current work and Perspectives

- Several major revolutions:
    - Diffusion in the industry of software defined networks
    - of network virtualization
    - Convergence network and data center architectures
    - 5G/IoT/M2M

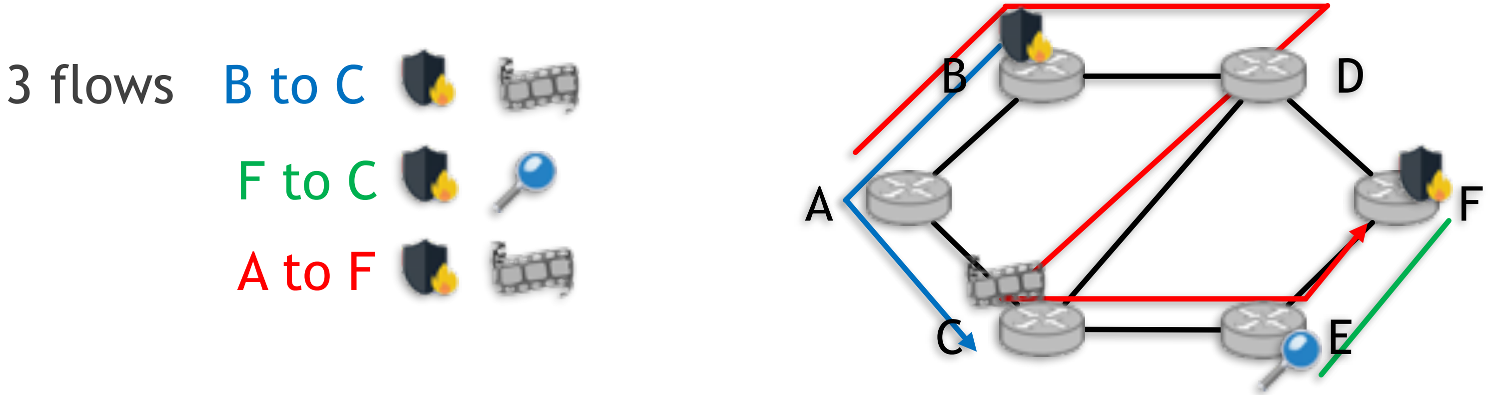


-> New algorithmic problems.

Perspectives

- Routing and placement of dynamic resources.

  -> study of the reconfiguration of virtual ressources.



3 flows B to C

F to C

A to F

PhD. Adrien Gausseran 2018-2021
(J. Moulierac and N. Nisse)

Perspectives

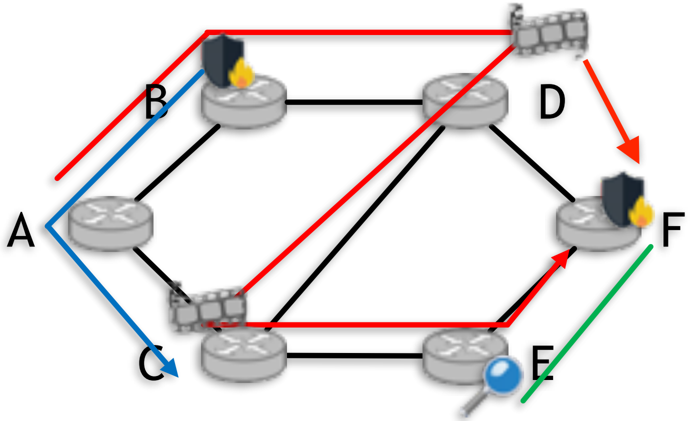
- Routing and placement of dynamic resources.

  -> study of the reconfiguration of virtual ressources.

3 flows B to C 🛡 🎞

F to C 🛡 🔍

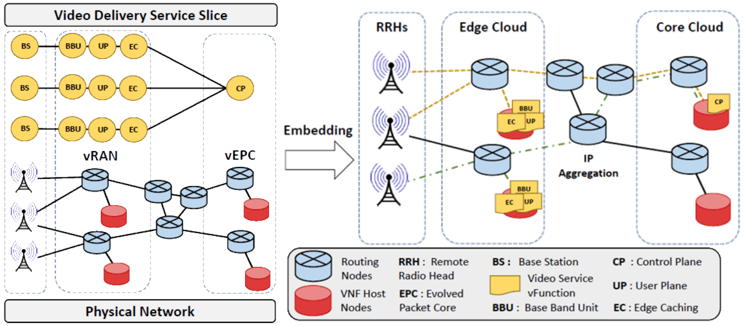
A to F 🛡 🎞

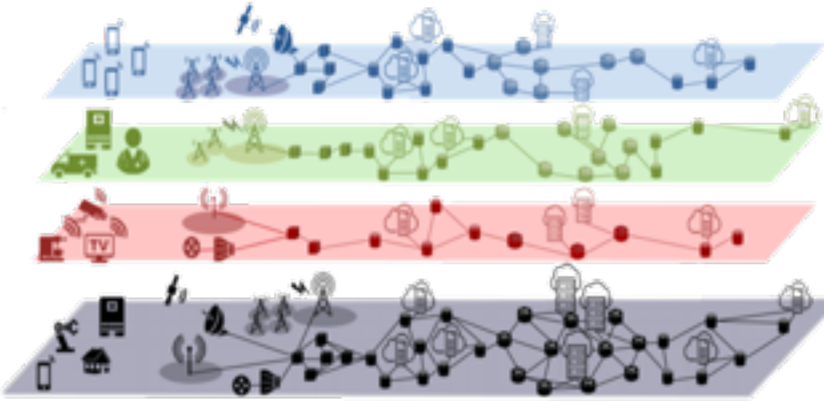


PhD. Adrien Gausseran 2018-2021
(J. Moulierac and N. Nisse)

Perspectives

- Mapping of virtual networks on physical networks. e.g slicing.





Mobile Slices

Critical Slices

IoT Slices

Physical Network

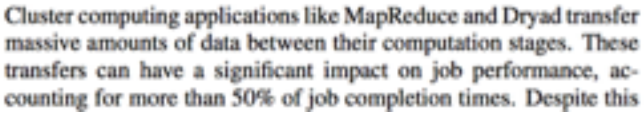
PhDs of A. Gausseran and G. di Lena 2018-2021 (Orange, C. Lac and T. Turletti)

Perspectives

- Joint optimization of applications and network traffic.

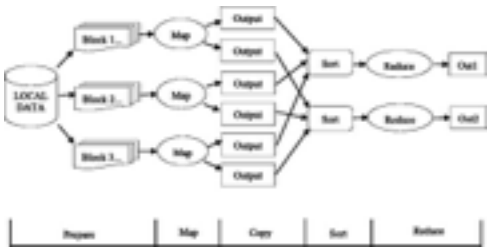
  -> introduction of a new scheduling framework scheduling data center workflows + network routes.

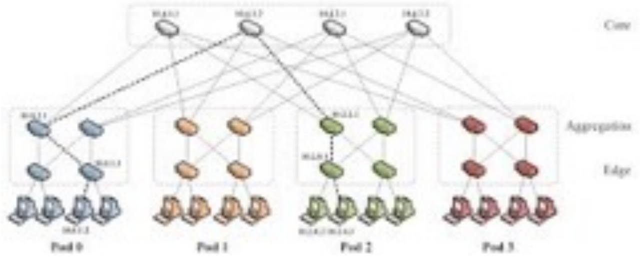
PhD. A. Tomassilli 2016-2019
with S. Pérennes



Cluster computing applications like MapReduce and Dryad transfer massive amounts of data between their computation stages. These transfers can have a significant impact on job performance, accounting for more than 50% of job completion times. Despite this

[SIGCOMM Orchestra 2011]





THANKS FOR YOUR ATTENTION!