On the Benefit of Virtualization

Strategies for Flexible Server Allocation

or/and: How to allocate resources when you don’t know the future?

Dushyant Arora

Anja Feldmann

Gregor Schaffrath

Stefan Schmid

T-Labs / TU Berlin

Co-authors:

Network virtualization architecture and prototype:

Anja Feldmann, Gregor Schaffrath, Stefan Schmid (T-Labs/TU Berlin)

VNet embeddings

Guy Even and
Moti Medina (Tel Aviv Uni),
Carlo Fürst (TUB)

Service migration

Dushyant Arora (BITS) and
Marcin Bienkowski (Uni Wroclaw)

Economics

Arne Ludwig (TUB)

Implementation

Ernesto Abarca,
Johannes Grassler,
Lukas Wöllner, etc.

Note: Focus here not limited to clouds!

A joint project with Deutsche Telekom, TU berlin and NTT docomo:

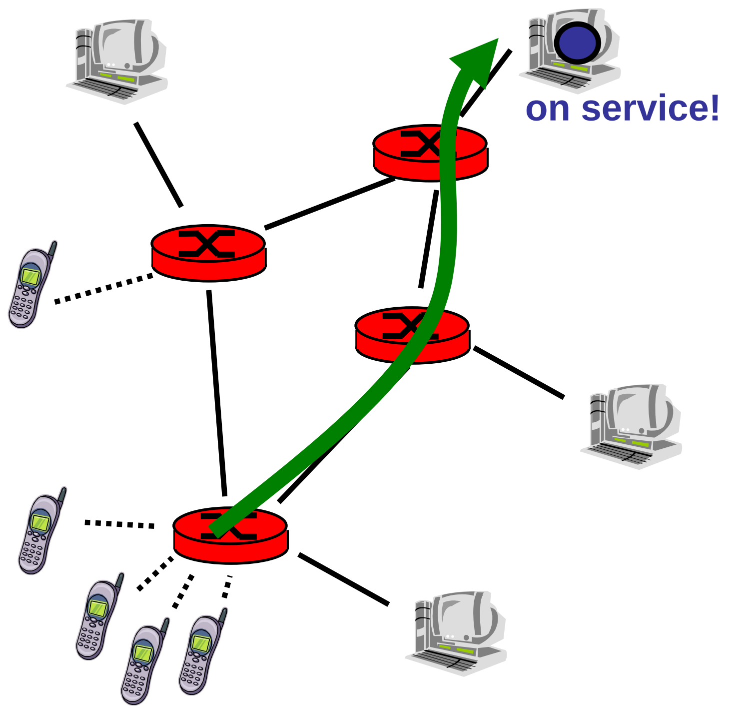
D. Jurca, A. Khan, W. Kellerer, K. Kozu and J. Widmer

Network Virtualization: High-level Concepts

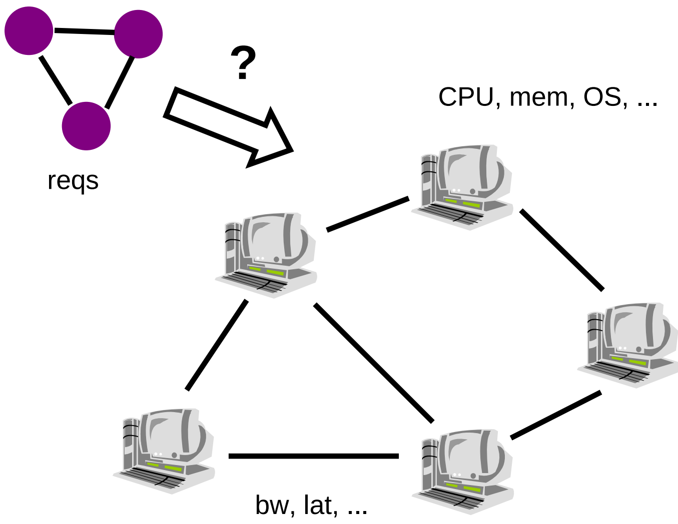
Decoupling services from physical infrastructure

- dynamic virtual network embeddings, sharing of resources, „smarter core“
- not only node but also link virtualization (e.g., VLANs, OpenFlow, ...)

Example 1: A mobile service provider can move services to locations where they are most useful:



Example 2: Virtual networks (VNets) can be allocated where the least resources are used, or where most energy can be saved, or...:



Previous work: Virtualization Business Roles

Actors in the Internet today: service providers and ISPs

- ISP: provide access (own infrastructure, rental, or combination), „connectivity service“ (e.g., Telekom, AT&T, ...)
- Service provider: offers services (e.g., Google)
- More roles exist today, often hidden in one company

Envisioned business roles:


PIP

Physical infrastructure provider (PIP):

owns and manages physical infrastructure („substrate“), supports network virtualization (e.g., GENI: no federation, one PIP only)


VNP

Virtual network provider (VNP):

assembles virtual resources from PIPs into virtual topology, makes negotiations, etc. (e.g., GENI clearinghouse)


VNO

Virtual network operator (VNO):

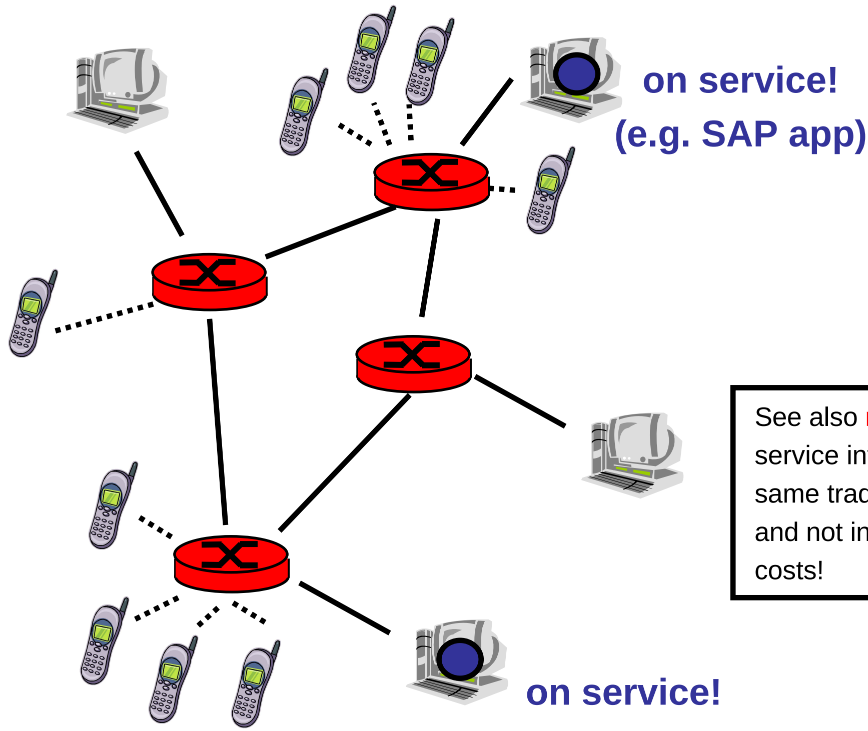
installation and operation of VNet according to SP needs, e.g., triggering cross-PIP migration, etc.


SP

Service provider (SP):

uses VNet to offer services (application or transport service)

This Paper: Online Service Migration



See also next talks on live migration and service interruption cost (not clear whether same tradeoff exists here, as isolated VNets and not in-band), as well as energy costs!

Access pattern changes, e.g., due to mobility (commuter scenario), due to time-of-day effects (time-zone scenario), etc.

... when and where to move the service, to maximize QoS and taking migration cost into account?

Similar tradeoffs in clouds, content distribution networks, etc.!

Dealing with Unpredictable Demand?

How to deal with dynamic changes (e.g., mobility of users, arrival of VNets, etc.)?

Online Algorithm

Online algorithms make decisions at time t without any knowledge of inputs / requests at times t’>t.

Competitive Analysis

An *r-competitive online algorithm* ALG gives a worst-case performance guarantee: the performance is at most a factor r worse than an optimal offline algorithm OPT!

Competitive Ratio

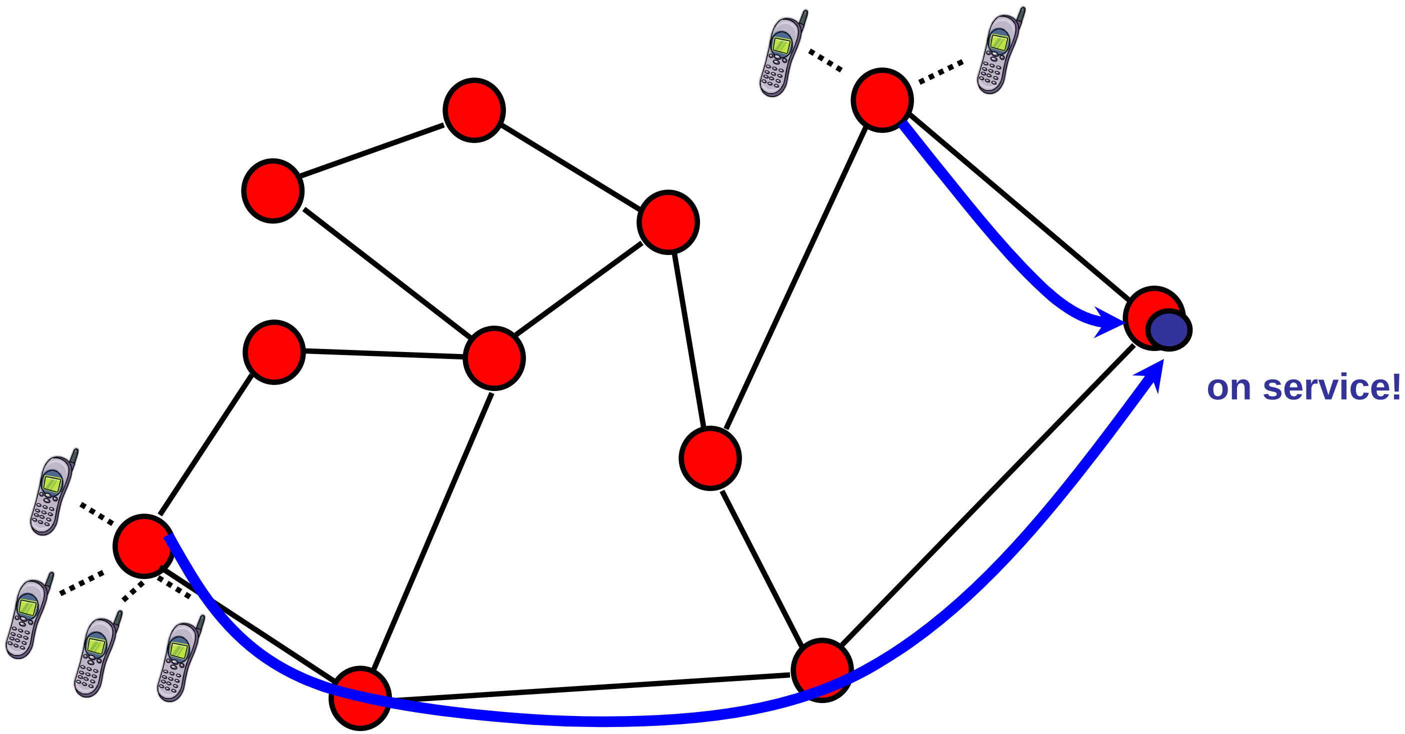
Competitive ratio r,

r = Cost(ALG) / cost(OPT)

Is the price of not knowing the future!

In virtual networks, many decisions need to be made online: online algorithms and network virtualization are a perfect match! ☺

Online Service Migration



Assume: one service, migration cost *m* (e.g., service interruption cost), access cost 1 per hop (or sum of link delays).

When and where to move for *offline algorithm* or *optimal competitive ratio*?

Optimal Offline Algorithm

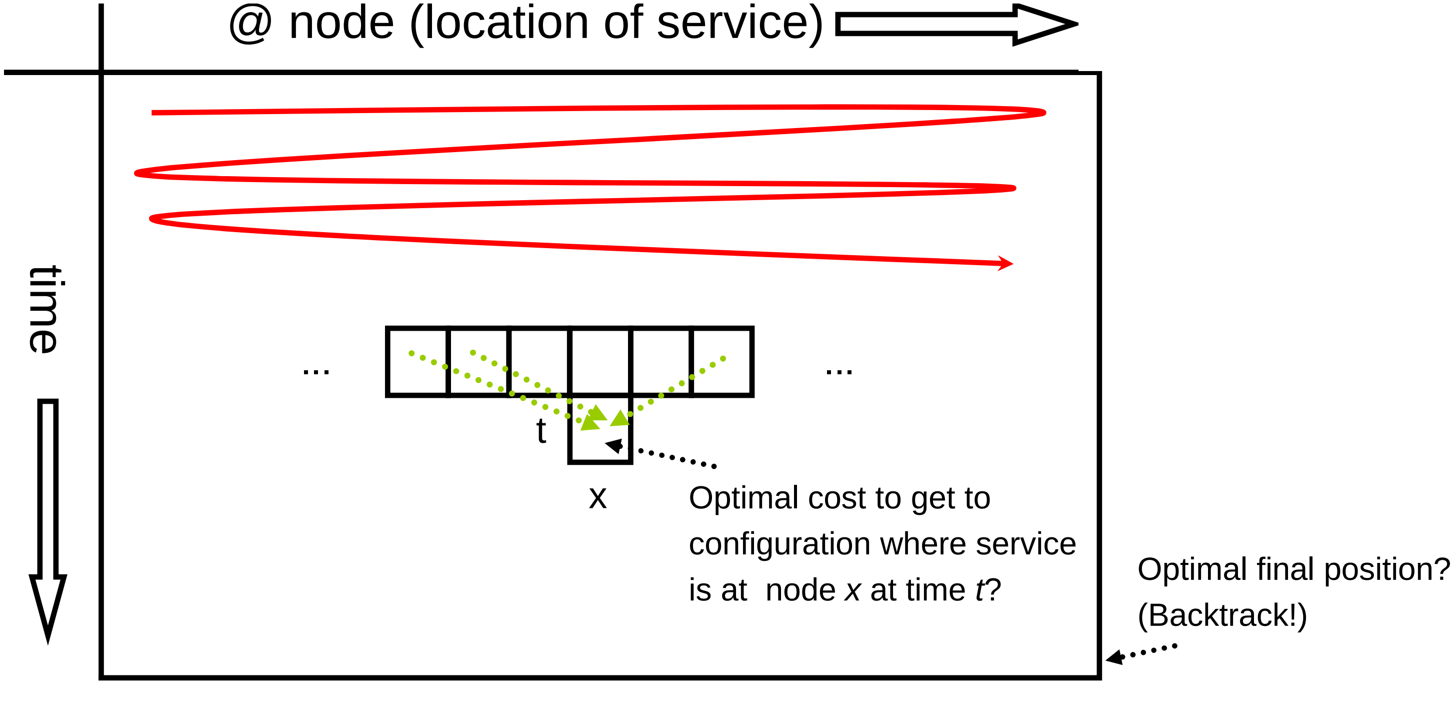
Can be computed using dynamic programming!

Filling out a for optimal server configuration (at node *u* at time *t*):

OPT

opt[u,t] = $\min_{v \in V}$ {opt[t-1][v] + MIG(v,u) + ACC(u,t)}

Visualization:



Online Algorithm

Idea: Migrate to center of gravity when access cost at current node is as high as migration cost!

Time between two migrations: *phase*

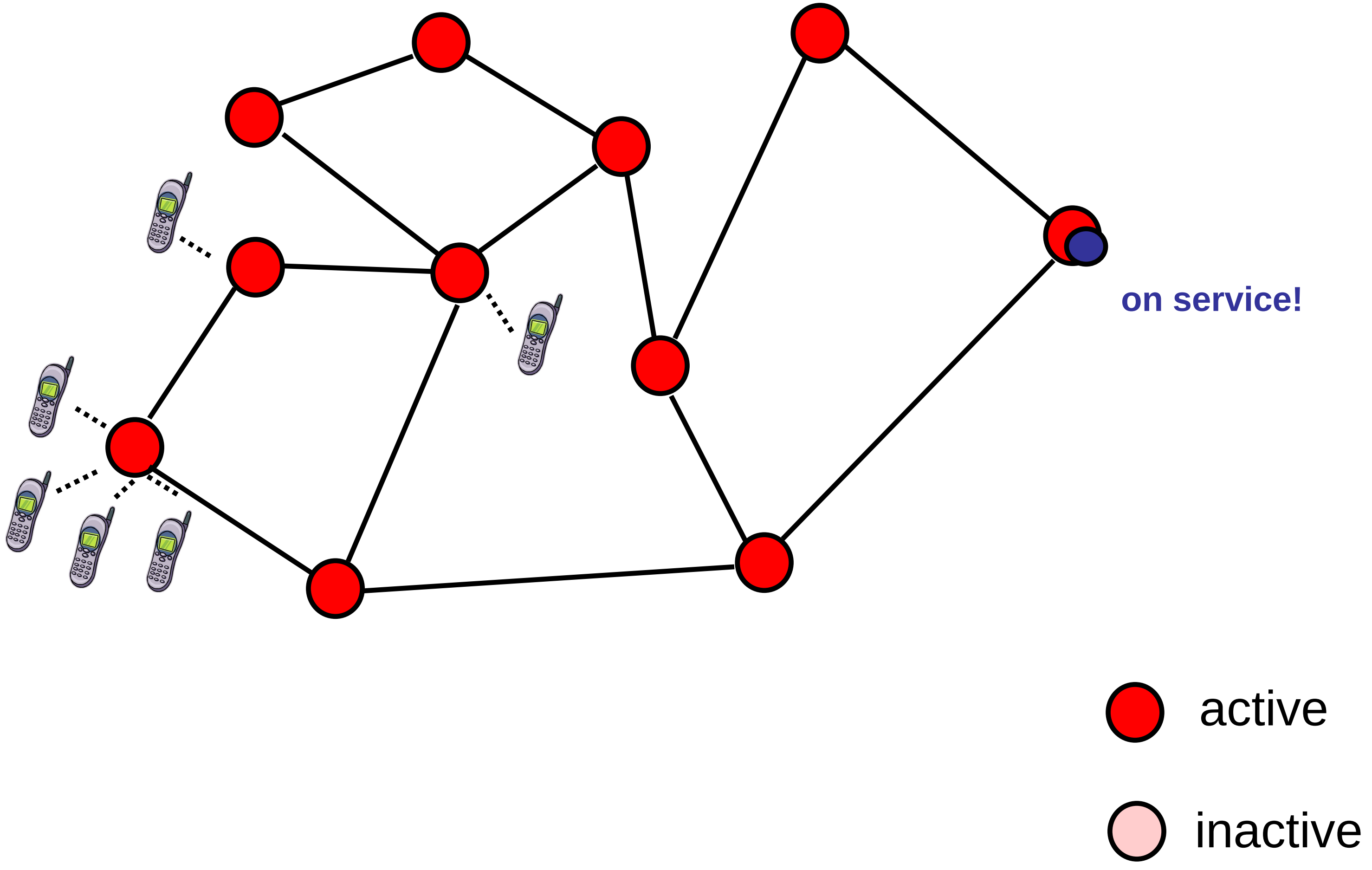
Multiple phases constitute an *epoch*

ALG

For each node *v*, use COUNT(v) to count access cost if service was at *v* during entire ***epoch***. Call nodes v with COUNT(v) < *m*/40 ***active***. If service is at node *w*, a ***phase*** ends when COUNT(w)≥m: the service is migrated to the ***center of gravity*** of the remaining active nodes („center node“ wrt latency or hop distance). If no such node is left, the epoch ends.

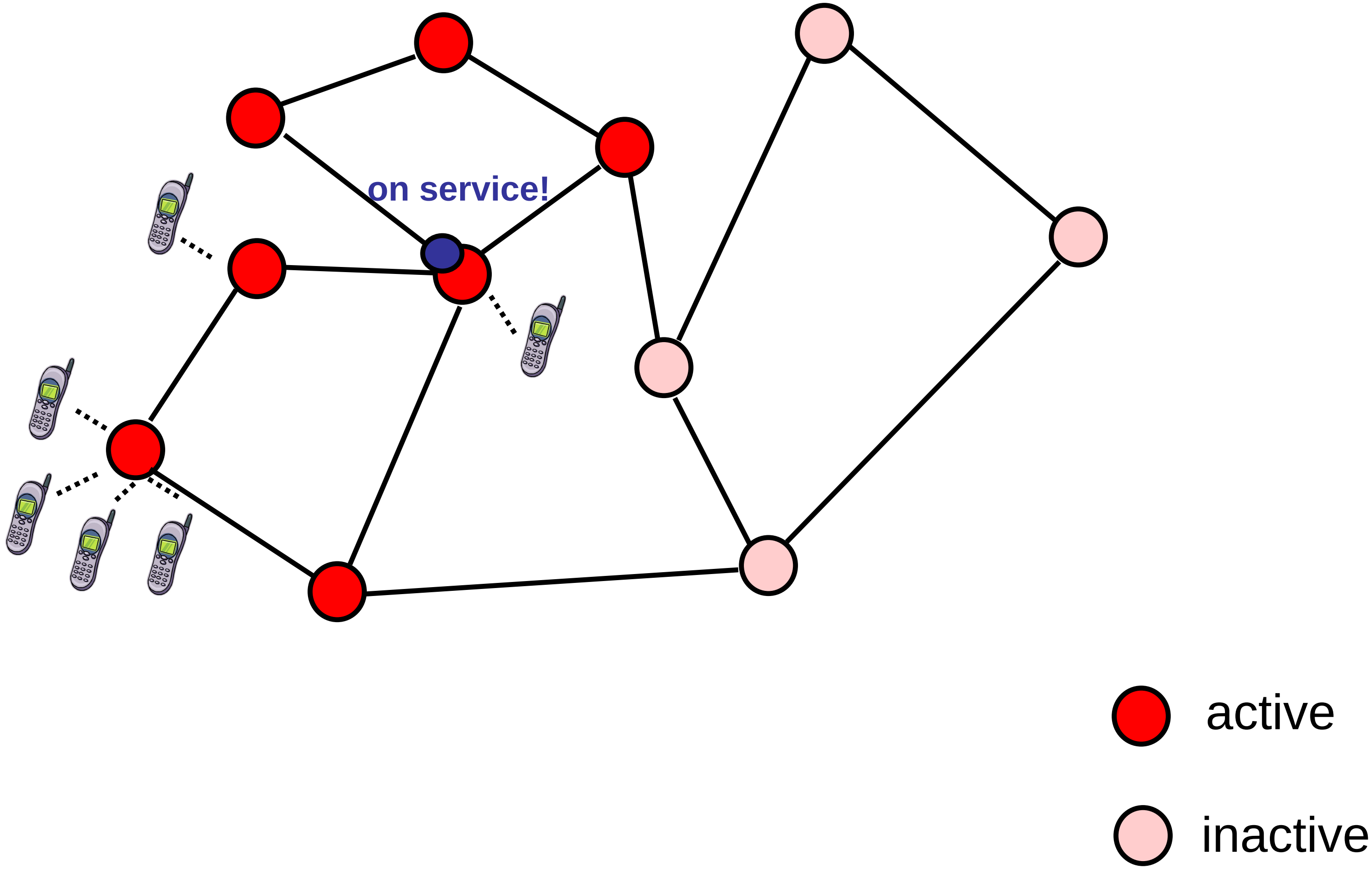
Online Algorithm: Visualization

Before phase 1:



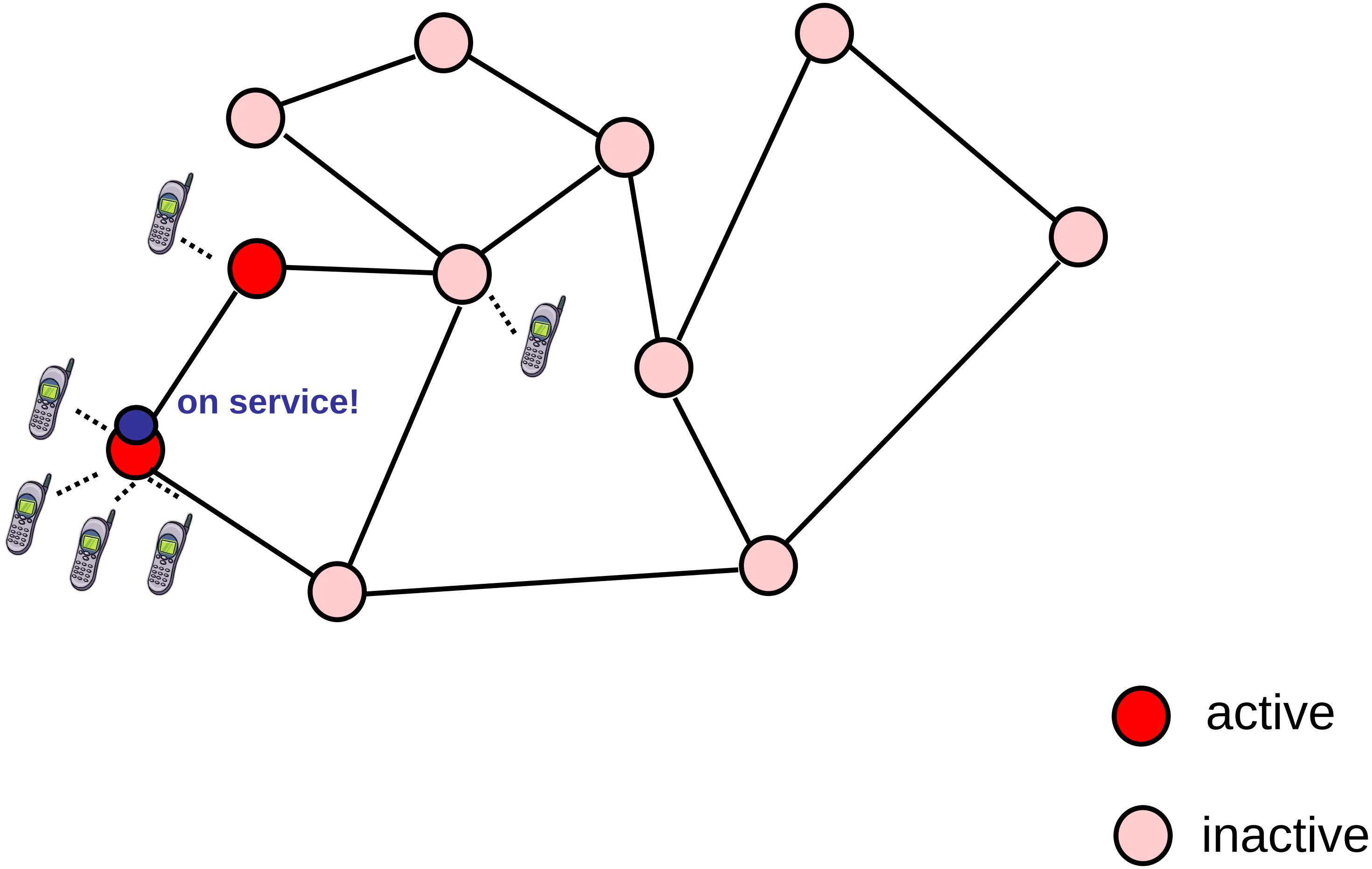
Online Algorithm: Visualization

Before phase 2:



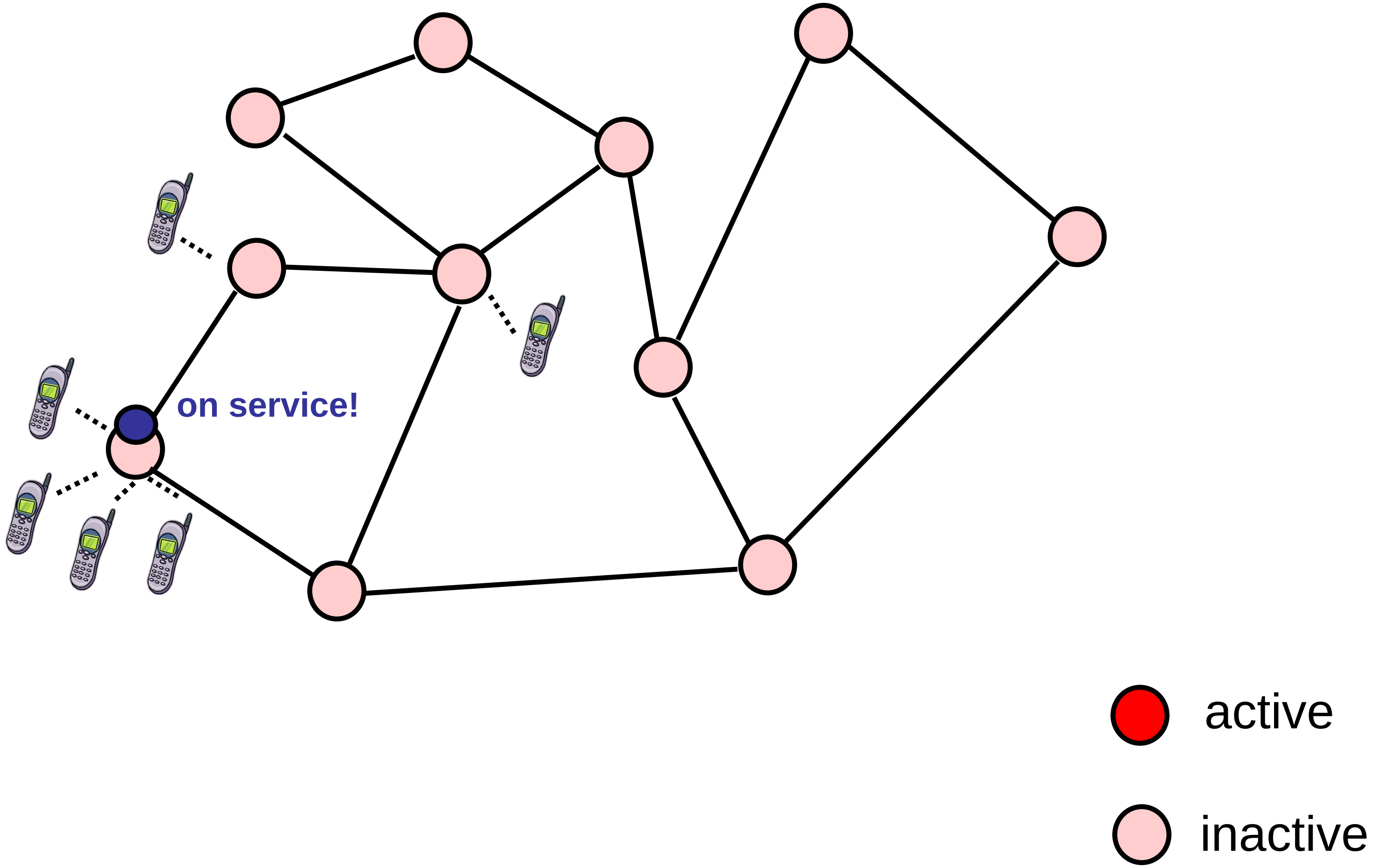
Online Algorithm: Visualization

Before phase 3:



Online Algorithm: Visualization

Epoch ends!



Online Algorithm: Analysis

Competitive analysis?

r = ALG / OPT $\leq$?

Lower bound cost of OPT:

In an epoch, each node has at least access cost *m*, or there was a migration of cost *m*.

Upper bound cost of ALG:

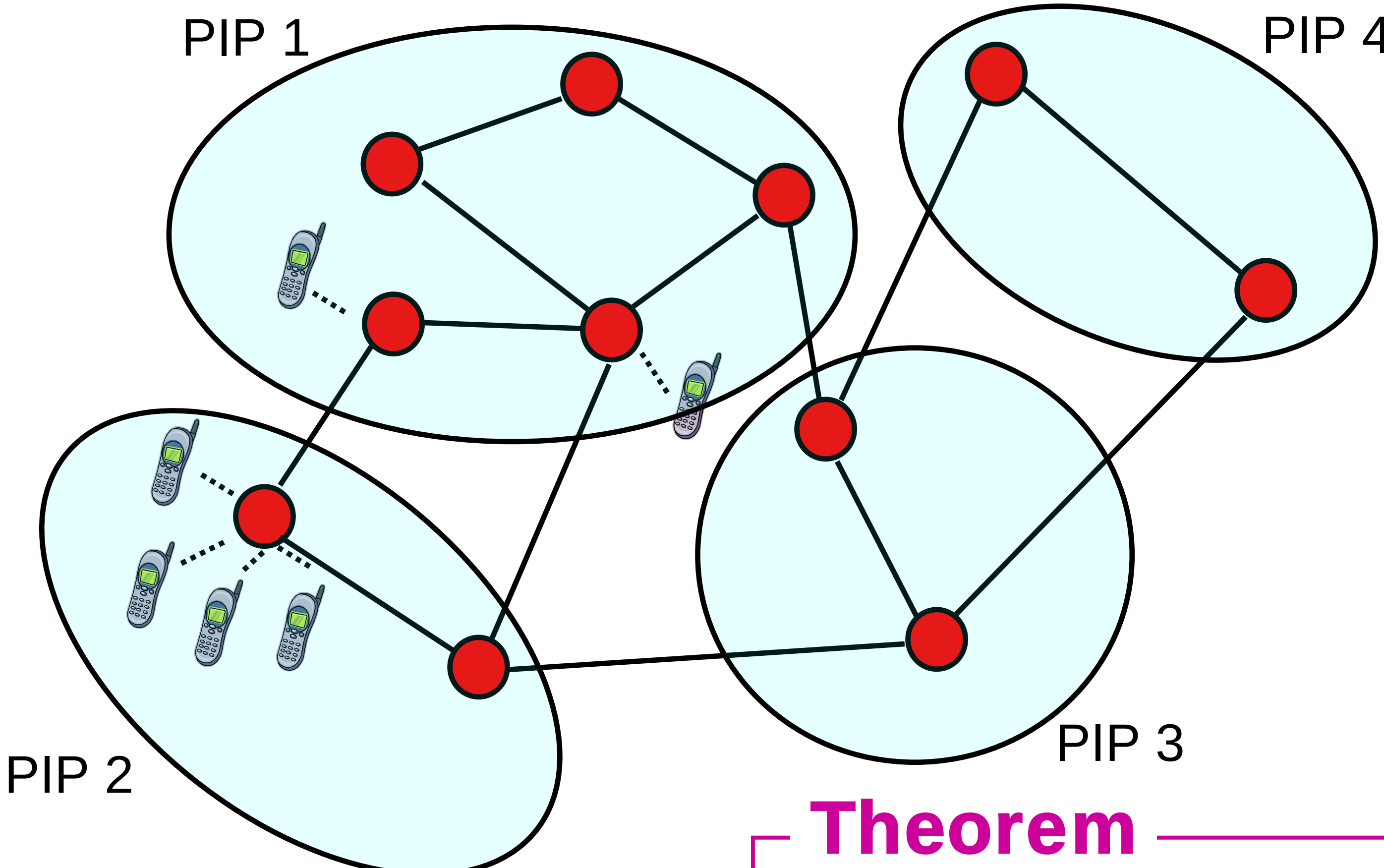
We can show that each phase has cost at most *2m* (access plus migration), and there are at most *log(m)* many phases per epoch!

Theorem

ALG is log(m) competitive!

Reality is more complex...: Multiple PIPs

Migration across provider boundary costs transit/roaming costs, detailed topology not known, etc.

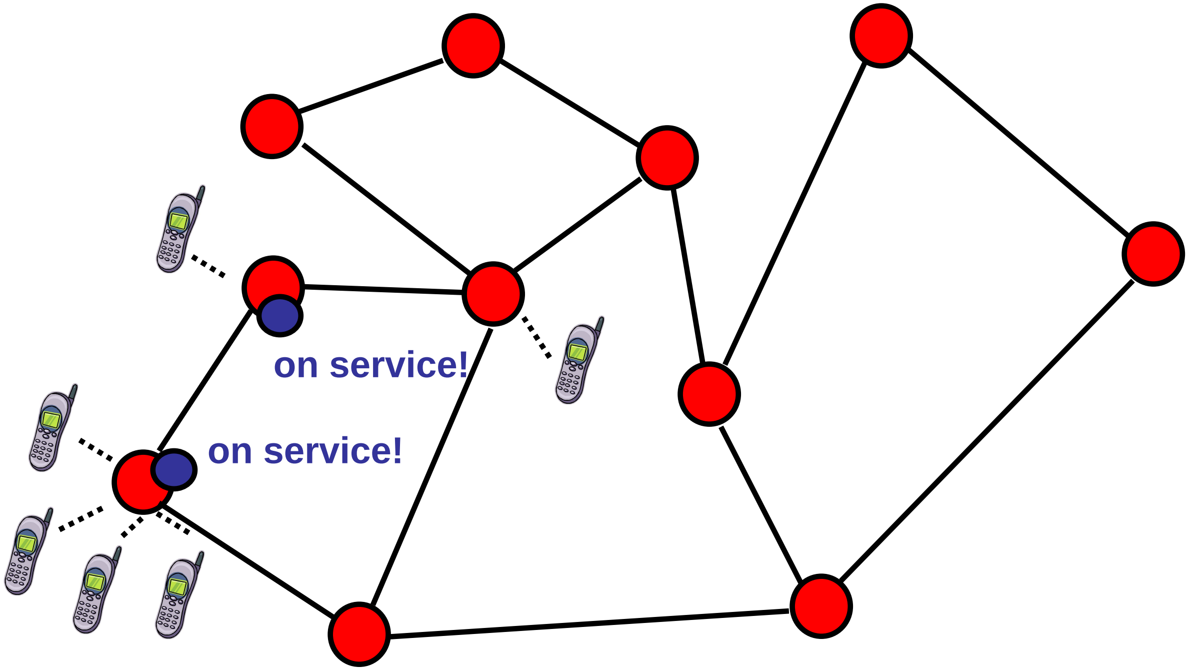


Theorem

Competitive ALGs still exist!

Reality is more complex...: Multiple Servers

Multiple servers allocated and migrated dynamically depending on demand and load, etc.



Theorem

Competitive ALGs still exist!

On the Benefit of Virtualization: Strategies for Flexible Server Allocation

Dushyant Arora, Anja Feldmann, Gregor Schaffrath, Stefan Schmid
Deutsche Telekom Laboratories / TU Berlin, Germany

1 Introduction

Network virtualization [2] is an intriguing paradigm which loosens the ties between services and physical infrastructure. The gained flexibility promises faster innovations, enabling a more diverse Internet and ensuring coexistence of heterogeneous *virtual network (VNet)* architectures on top of a shared substrate. Moreover, the dynamic and demand driven allocation of resources may yield a "greener Internet" without sacrificing (or, in the presence of the corresponding migration technology: with improved!) quality-of-service (QoS) / quality-of-experience (QoE).

This paper attends to a fundamental challenge in the field of network virtualization: the flexible allocation and migration of servers. As a generic use case, we consider a network operator offering a flexible service to a set of dynamic or mobile users, and we present a model that captures the main cost factors in such a system. This allows us to shed light on the benefit of the flexible allocation and the use of migration.

Although our cost model is described from a network virtualization perspective, it is not limited to such architecture requests, and *offline algorithms* where the (e.g., periodic) demand is known ahead of time. Both algorithms are applicable to various delay models (access latency, delay due to different load functions, etc.). Moreover, we also describe an optimal offline but static algorithm that allows us to *quantify the cost-benefit tradeoffs of dynamic resource allocation*, and thus to shed light on fundamental questions such as the use of migration compared to solutions using static servers. For example, our simulations show that the overall cost can be higher (by up to hundred percent), if resources are static, in particular if the demand dynamics is moderate.

The content of this workshop paper is based on the longer *arXiv report 1011.6594*; please refer to the technical report for more details.

2 A Flexible Service Provider

Our work is motivated by the network virtualization paradigm that decouples services from the underlying physical infrastructure (the *substrate*) and for which we are in the process of developing a prototype architecture [8]. However, the model and tradeoffs studied in the

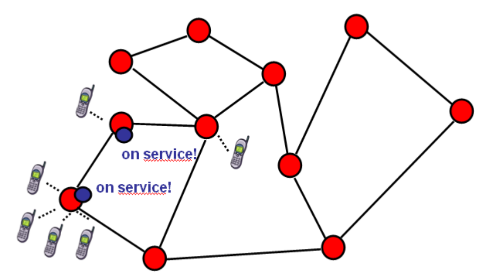
Very general cost model

- detailed study of cost factors
- access cost that depend on latency and load
- servers have running costs (unlike many classic problems such as online facility location or metrical task systems)

Online and offline algorithms for various scenarios

Focus on use of flexible allocation (compared to static allocation)

- under what dynamics is flexibility better?



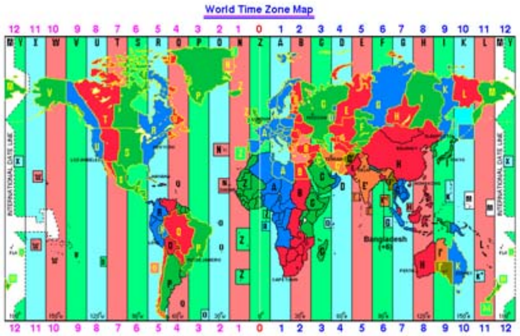
On the Benefit of Flexibility: Dynamics Scenarios

Commuter Scenario

Dynamics due to mobility: requests cycle through a 24h pattern: in the morning, requests distributed widely (people in suburbs), then focus in city centers; in the evening, reverse.

Time Zone Scenario

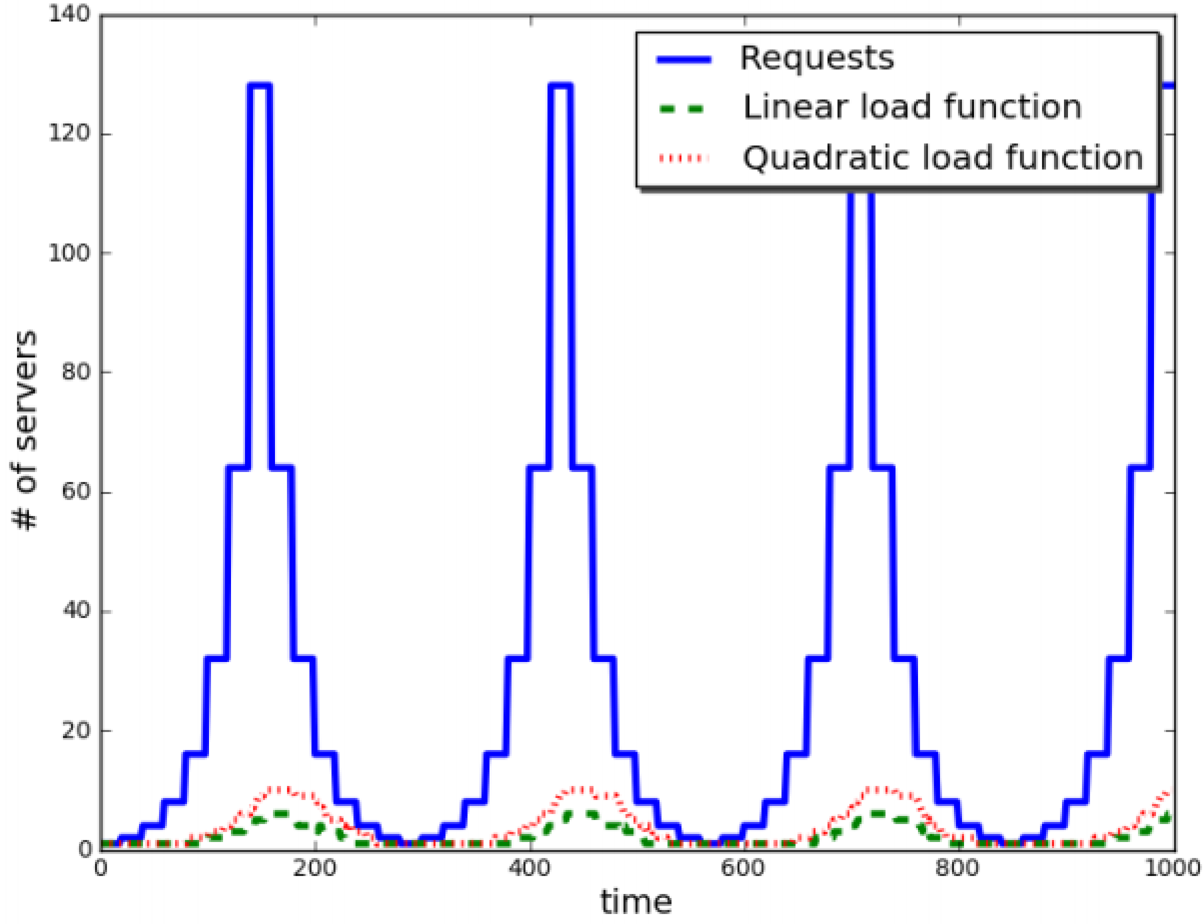
Dynamics due to time zone effects: request originate in China first, then more requests come from European countries, and finally from the U.S.



Static Algorithm

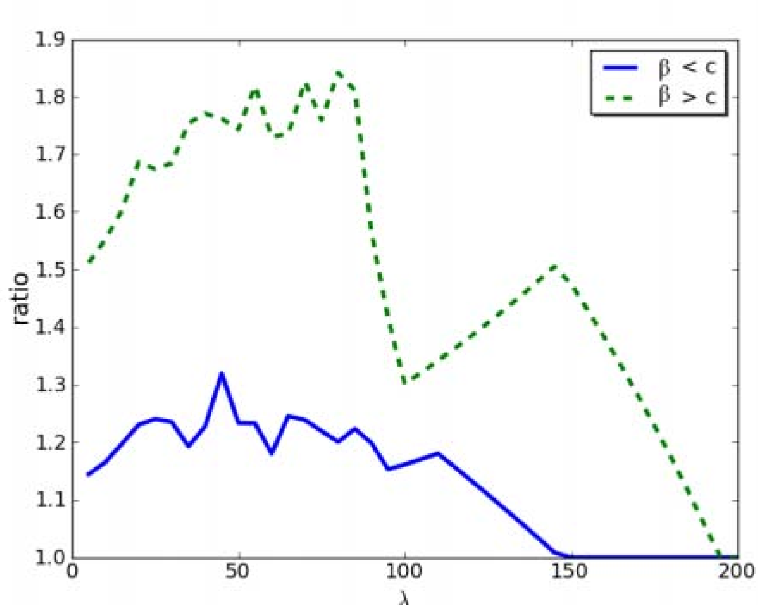
Algorithm which uses optimal static server placements for a given request seq.

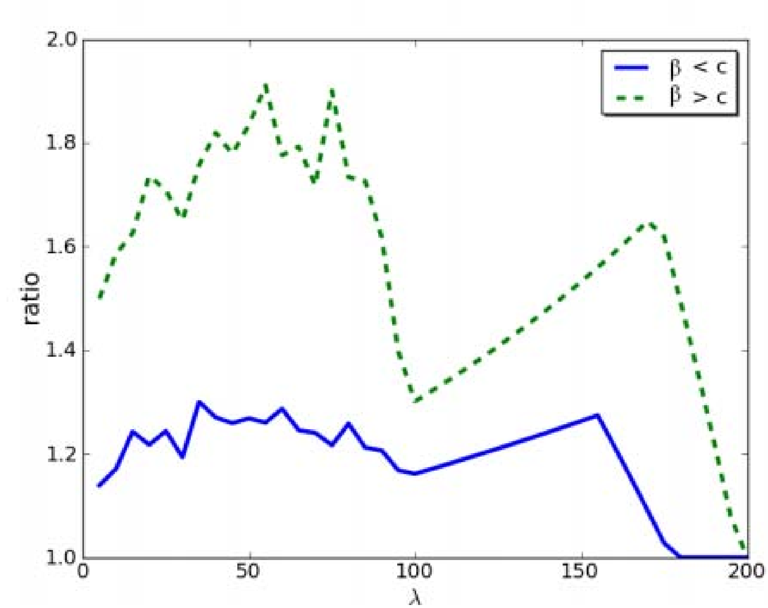
Intuition for Algorithm...



Increasing demand triggers creation of additional servers (more for faster growing load functions).

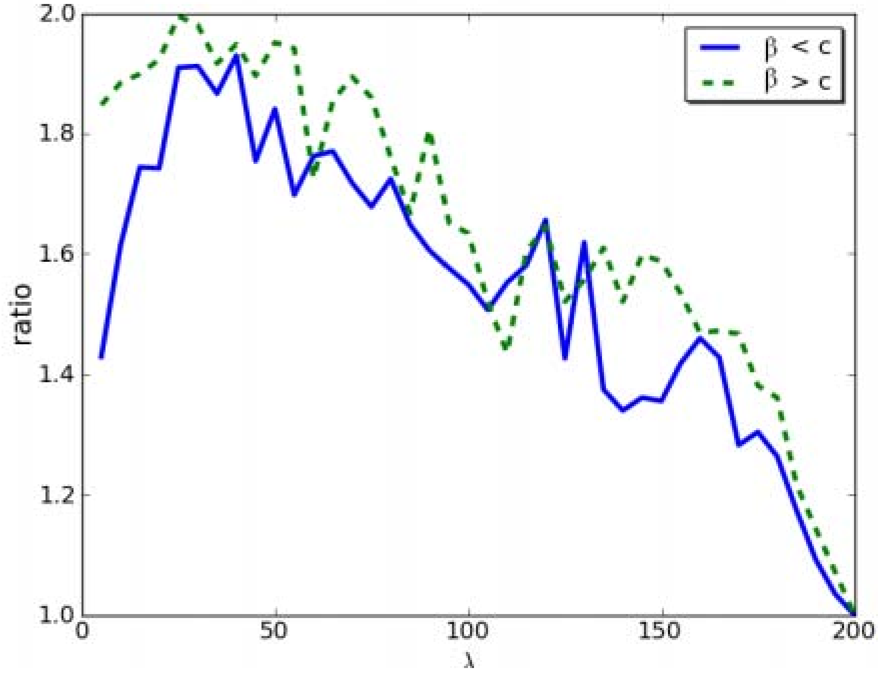
On the Benefit of Flexibility: Commuter Scenario





ALG/STAT as a function of dynamics (static and dynamic load):
For low dynamics and high dynamics, flexibility is less useful
(max gain: almost factor of 2).

On the Benefit of Flexibility: Time Zone Scenario



ALG/STAT as a function of dynamics: for time zone scenario.

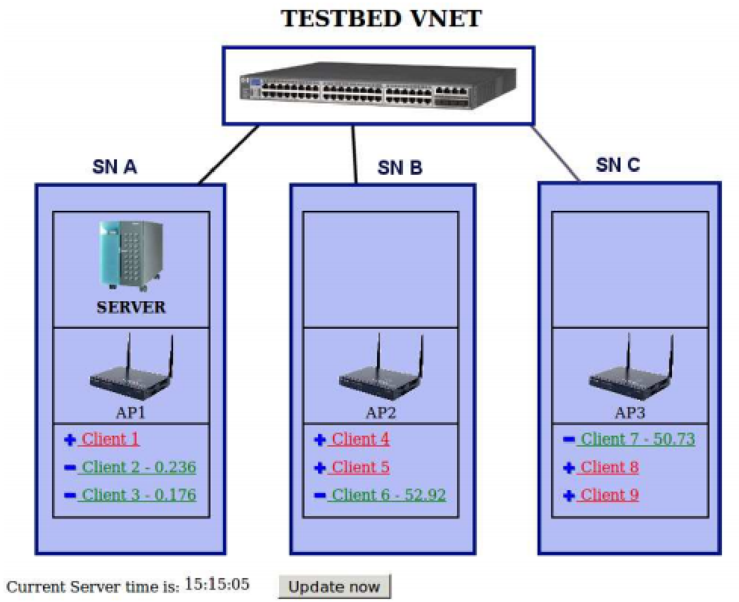
Conclusion and Takeaways

- Flexible server allocation for network virtualization and beyond: generalized model for a challenging problem

- Online perspective: algorithms have to decide without knowing the future; relevant for many aspects of network virtualization

- When useful? Depends on dynamics!

- Streaming migration demonstrator for our network virtualization prototype (VLAN based):



Thank you!

Further reading (e.g., on competitive embedding algorithms):

http://www.net.t-labs.tu-berlin.de/~stefan/

Comparison to Related Work

- Conservative online perspective on resource management: no predictions possible, but with worst-case guarantees

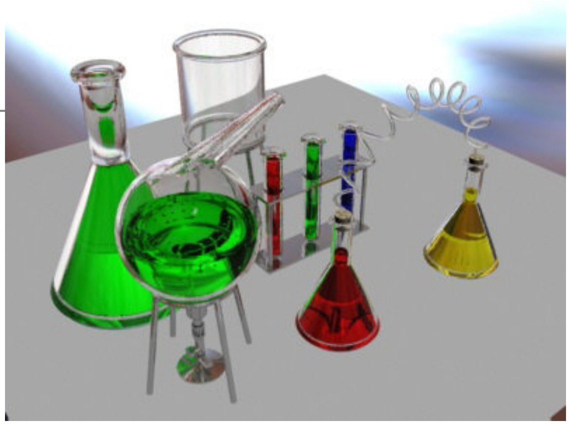
- Detailed costs model for VNet application (multiple PIPs with transit costs, costs depending on scenario: shared NFS, etc.)

- Allows to study the „use of flexibility“ (compared to static algorithms)

- Like dynamic facility location problems where additional facilities can be created, migrated and closed (at non-zero cost) and where facilities have running costs and access costs that depend on load

- Often a special case of metrical task systems but sometimes better bounds can be obtained for the more specific model!

New Resource Allocation Challenges?



- Flexibility of embedding (max-flow problem
  with flexible end-points)

- Migration technology: new tradeoffs

- Economical aspects: new roles, new forms of inter-provider collaboration (roaming, QoS, inter-provider migration, ...)

- Unknown demand and traffic patterns, new models for prediction?