Online Management of Virtual Networks

Stefan Schmid
T-Labs / TU Berlin
Virtu Research Areas: The 3 Pillars

Service migration

VNet embeddings

Economics

Virtu + Online Algorithms = ❤
Network Virtualization: High-level Concepts

**Decoupling** services from physical infrastructure
- dynamic virtual network embeddings, sharing of resources, „smarter core“

**Example 1:** A mobile service provider can **move services** to locations where they are most useful: migration vs QoS tradeoff

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

- CPU, mem, OS, ...
- reqs
- bw, lat, ...

on service!
Virtualization Business Roles

Envisioned business roles:

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

- **Virtual network provider (VNP):**
  assembles virtual resources from PIPs into virtual topology, makes negotiations, etc. (e.g., GENI clearinghouse)

- **Virtual network operator (VNO):**
  installation and operation of VNet according to SP needs, e.g., triggering cross-PIP migration, etc.

- **Service provider (SP):**
  uses VNet to offer services (application or transport service)
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??
Virtu + Online Algorithms = ♥

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 Ratio**

Competitive ratio r,

\[ r = \frac{\text{Cost(ALG)}}{\text{cost(OPT)}} \]

Is the price of not knowing the future!

**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!

In virtual networks, many decisions need to be made online: online algorithms and network virtualization are a perfect match! 😊
Online Service Migration 1 PIP, 1 Server: Easy

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 $t$):

$$\text{OPT}$$

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

**Optimal cost to get to configuration where service is at node $x$ at time $t$?**

**Optimal final position?** (Backtrack!)
Online Algorithm: Center to Gravity Approach

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 \( \text{COUNT}(v) \) to count access cost if service was at \( v \) during entire epoch. Call nodes \( v \) with \( \text{COUNT}(v) < m/40 \) active. If service is at node \( w \), a phase ends when \( \text{COUNT}(w) \geq m \): 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:

During the phase, requests may move!

When access cost reaches $m$ here, migrate service to CoG of active nodes!
Online Algorithm: Visualization

Before phase 2:
Online Algorithm: Visualization

Before phase 3:
Online Algorithm: Visualization

Epoch ends!

- active
- inactive
Online Algorithm: Analysis

Competitive analysis?

\[ r = \frac{\text{ALG}}{\text{OPT}} \leq ? \]

**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 costs, detailed topology not known, etc.

**Theorem**

Competitive ALGs still exist 😊!

Idea: Two layers of 1 PIP algo...
Reality is more complex...: Multiple Servers

Multiple servers allocated dynamically depending on load, etc.

**Theorem**

Competitive ALGs still exist 😊!

Idea: via configurations...
Network virtualization architecture and prototype:
Prof. Anja Feldmann, Gregor Schaffrath, Stefan Schmid (T-Labs/TU Berlin)

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

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

Economics
Arne Ludwig (TUB)

TUB students on bord...
Ernesto Abarca, Afshin Hormozdiar, Johannes Grassler, Lukas Wöllner etc.

A joint project with Deutsche Telekom, TU Berlin and NTT docomo:
D. Jurca, A. Khan, W. Kellerer, K. Kozu and J. Widmer
Thank you for your interest!