Models for a 3G Network's GGSN

Florian Metzger

University of Vienna
Faculty of Computer Science

DOC Forschungsseminar
2013/11/13

Relation to Thesis

Planned thesis parts

1. Investigation of TCP-based video streaming techniques
   - Protocol survey and classification
   - Deriving a model
   - Measurements with the model
2. Evaluation of a 3G core network
   - Investigation and evaluation of the control plane
   - **Modeling and simulating load**
3. Measuring video streaming in a 3G network

Presentation based on MMB'14 submission *"A PDP Context Load Model and Virtualization Gain for a Mobile Network's GGSN"*

Motivation

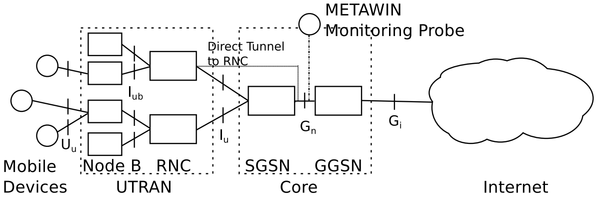
Mobile network planning and dimensioning today

- based on expected user traffic
- good algorithms and tools for placing radio towers and planning radio propagation
- core network and control plane usually not given much consideration

Our approach

- presents queuing models for a GGSN in the core network
- models simulated with data from a real network
- can be used to dimension for control plane
- offers more scaling options

GTP Tunnels and Dataset

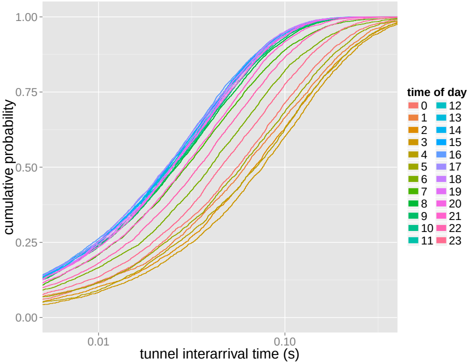


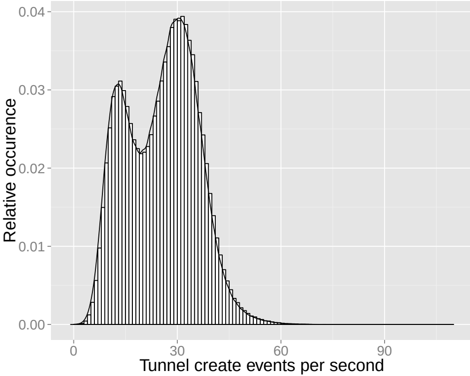
- Any user traffic in a 3G net is encapsulated into tunnels
- GPRS Tunneling Protocol (GTP) used between SGSN and GGSN
- Tunnel state (PDP Context) held at and signaled between core nodes through create/delete/update messages

Recorded dataset

- One week long passive measurements in an operator's core network (METAWIN, April 2011)
- 2.2Bn anonymized user traffic records, 410M GTP tunnel management messages

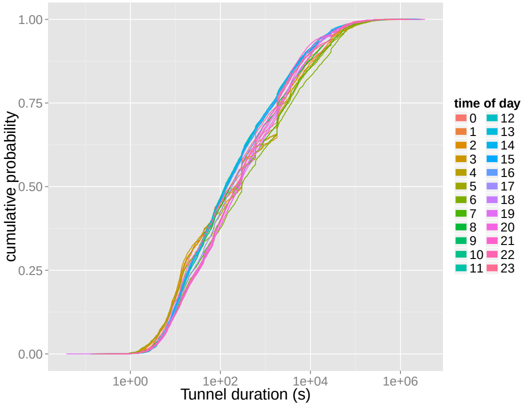
Tunnel Arrivals





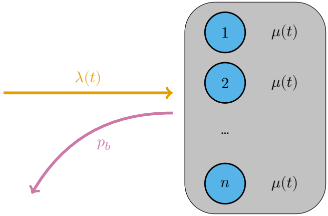
- Strong time of day dependence with busy hour in the early afternoon

Tunnel Durations



- Only slight dependence on time of day
- Much stronger influence of user device type, OS, or network timers (not shown here)

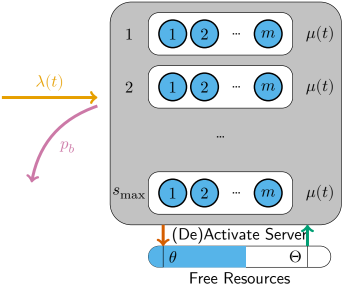
Monolithic GGSN Queuing Model



- Poisson tunnel arrival process with rate $\lambda(t)$, adjusted for the time of day
- GGSN can serve n tunnels in parallel, limited by network/processing load and signaling/state overhead
- Tunnels have a duration of $\mu(t)$ with a general distribution
- If GGSN is full, reject new tunnels with blocking probability p_b

$\rightarrow$ Non-stationary Erlang loss model $M_t/G/n/0$

Virtualized GGSN Queuing Model



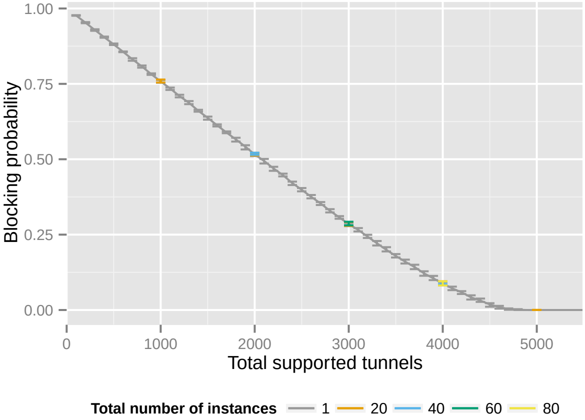
- Same arrival and serving time process, no queue
- Hypervisor distributes tunnels and starts on demand up to s_{max} virtualized GGSN instances, each with capacity m
- Additional blocking when new instances are not switched on fast enough, or instance overhead if not shut down when unused
- System scales up (larger instances) and out (more instances)

Simulating the Model

- No exact mathematical solution available for a $M_t/G/n/0$ model
- Use queuing simulation instead of stationary analysis
- SimPy3 based discrete event simulation[1]
- One week simulated period, omitted startup phase, 10 repetitions
- Arrival process with exponential distributions fitted to dataset, four time of day slots ($\lambda = \{10.67, 24.53, 29.25, 23.50\}$ before normalization)
- Tunnel duration CDF fitted with a rational function
- Scenario variable parameters: n, m, and s_{max}
- Evaluate and compare both models based on
  - Blocking probability
  - resource and instance usage

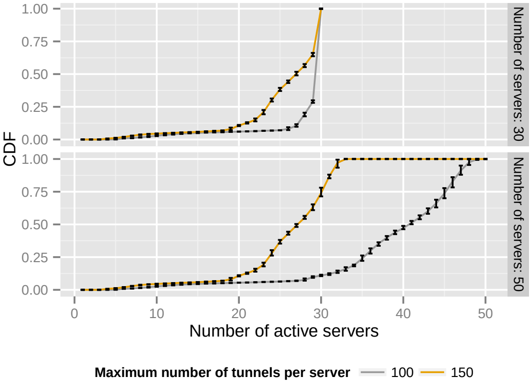
[1] https://github.com/fmetzger/ggsn-simulation

Blocking Probability



- Monolithic and virtualized GGSN scale equally with supported tunnels
- Negligible to no impact on p_B if virtualized model is scaled by tuning s_{max} instead of m

Virtualized GGSN Resource Usage



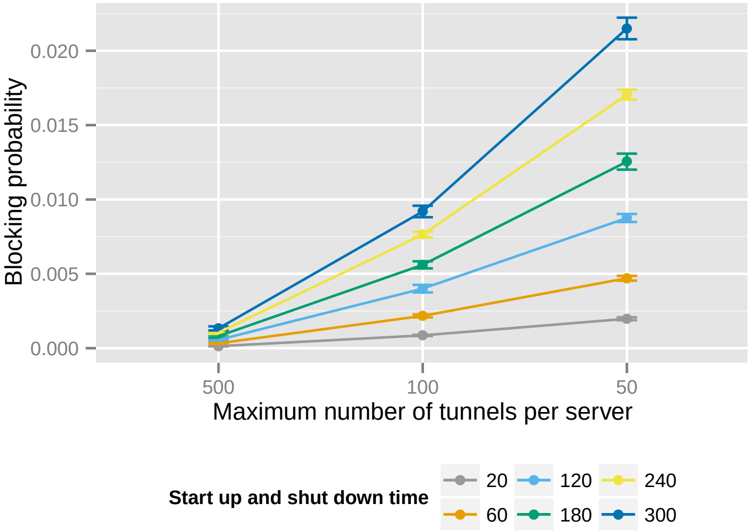
- Unused instances can be shut down for increased energy efficiency compared to monolithic model

Conclusion

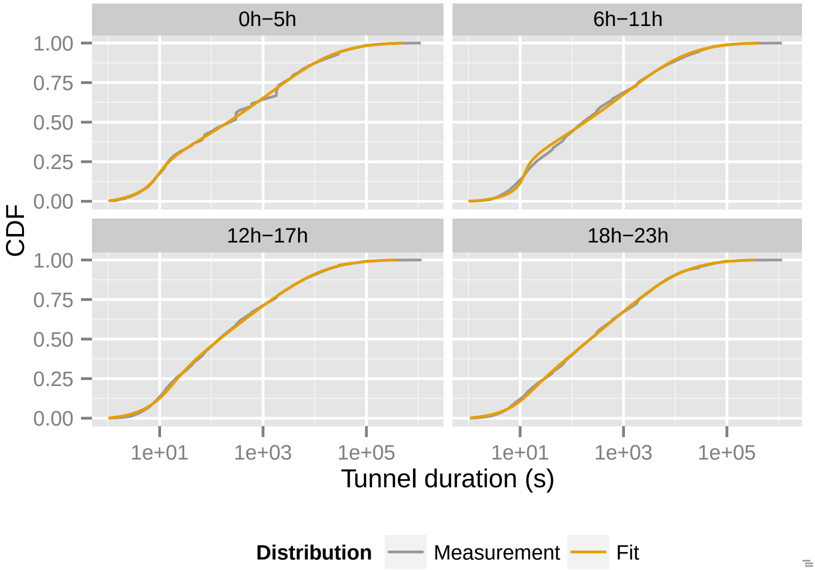
- Investigated tunnel properties in core network dataset
  - Non-stationary Poisson arrivals
  - Tunnel duration with general distribution
- Erlang loss models for tunnel load at a mobile core network's GGSN
  - Monolithic GGSN representing today's makeup
  - Virtualized GGSN proposal with improved scalability and efficiency
- Simulative evaluation of the model
- Enable mobile network dimensioning based on tunnel blocking rate instead of only user traffic volume

Thanks!

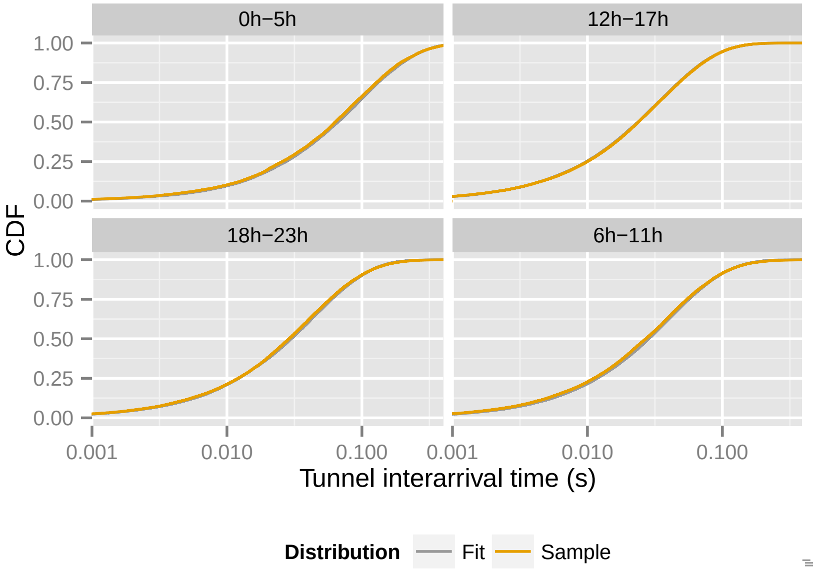
Questions?


Blocking probability
0.020
0.015
0.010
0.005
0.000
500
100
50
Maximum number of tunnels per server
Start up and shut down time
20
120
240
60
180
300

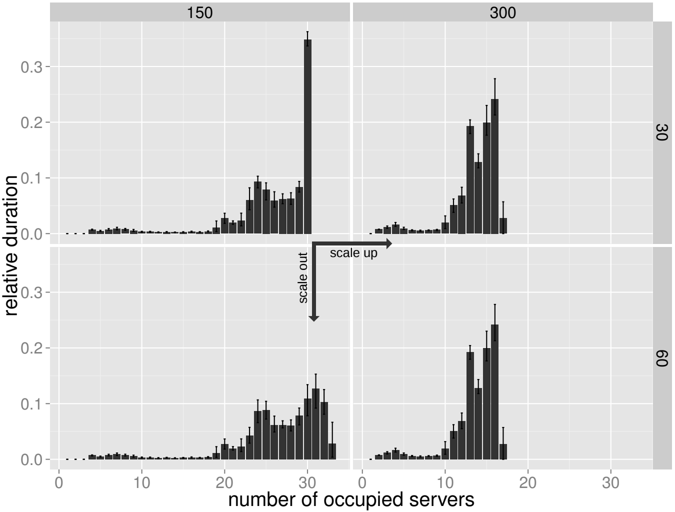
Exponential Arrival Process Fits



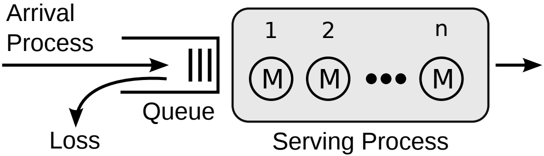
Serving Time Rational Functions Fit



Scaling Up or Out with a Virtualized GGSN



Queuing Models



Described by Kendall's Notation $A/S/c/q$

- Distribution of the arrival process A
- Distribution of the serving time S
- Number of Servers c
- Queue Length q
  - $q = \infty$ no loss will occur
  - 0 loss/blocking system, no queue
- Evaluate
  - Average queue length and server occupation
  - Blocking probability