Analysing Efficiency of New Load Distribution Algorithm with Horizontal Scaling

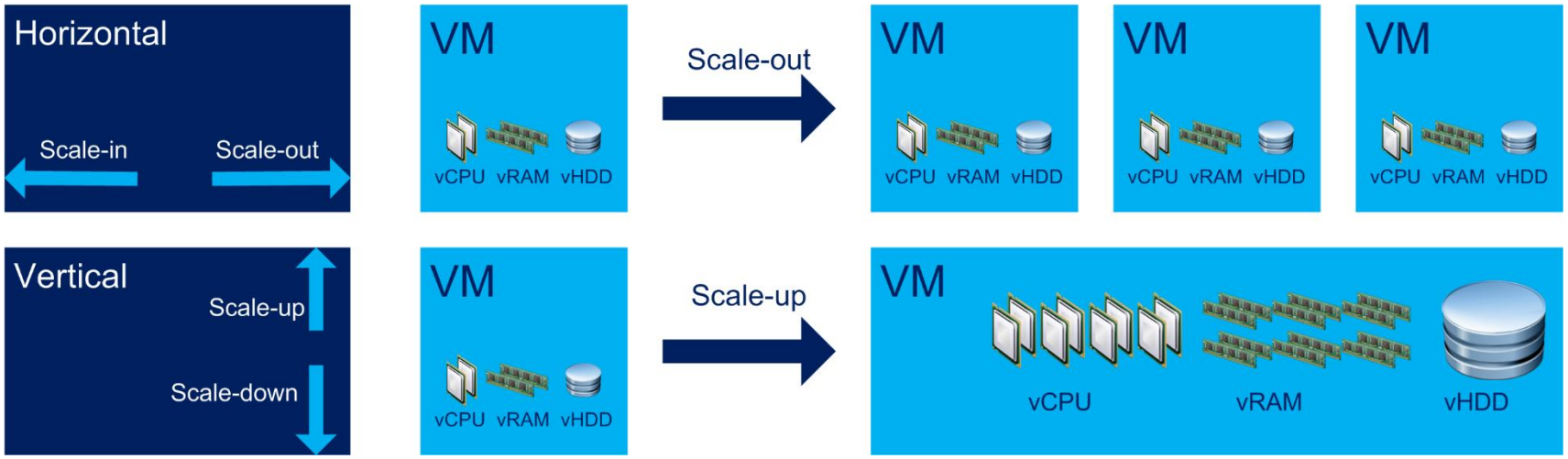
Loránd Nagy, Attila Hilt, László Jánosi, and Gábor Járó

HTE Infokom, Tapolca, Hungary

October 13. 2016.

Introduction
Distributed systems and scaling

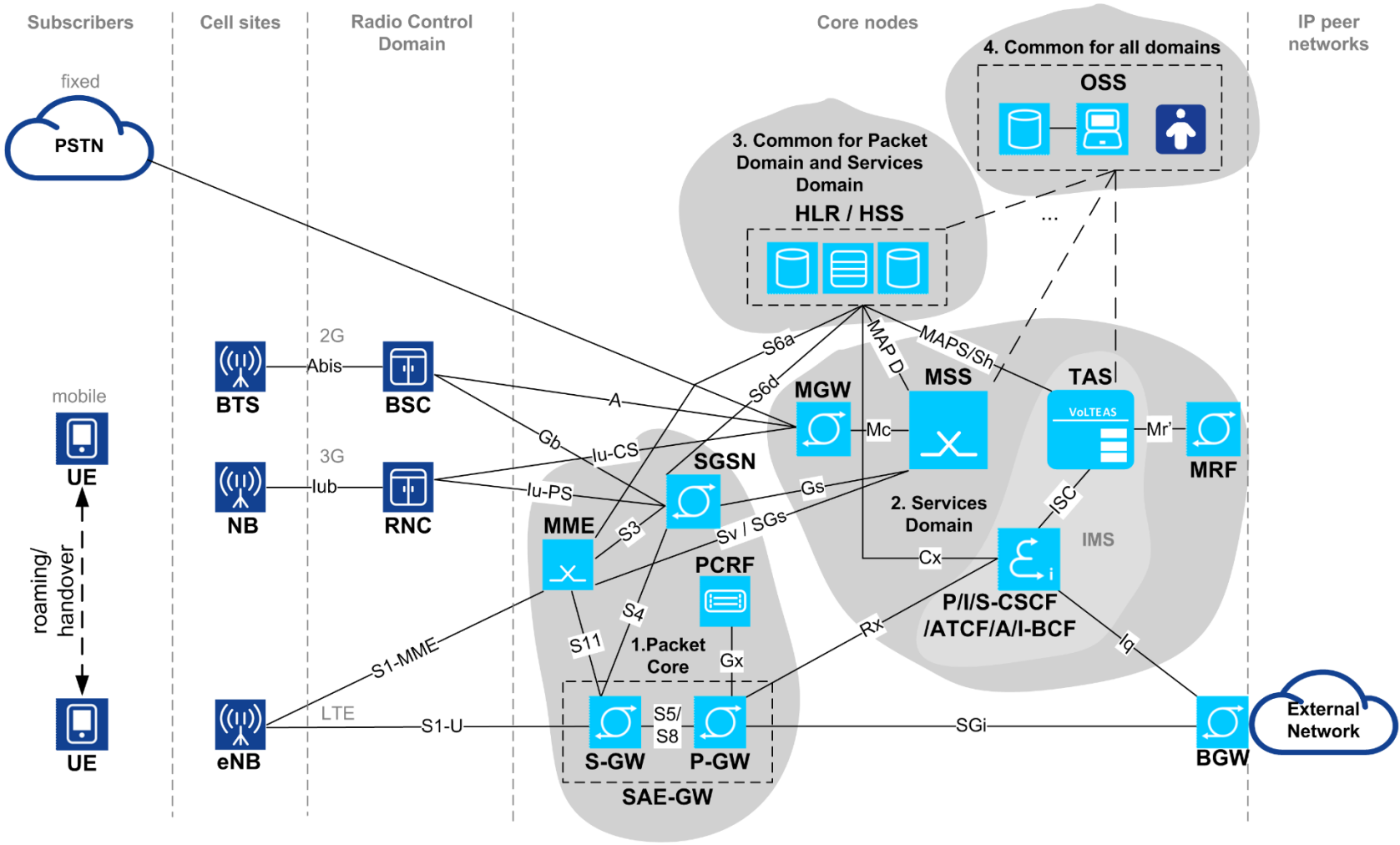
- In most systems there are many tasks that can be processed independently
- Therefore the system load can be handled by multiple *functional units*
- If the system load varies through time, the amount of required resources is not constant
- The total system resource amount can be changed via scaling; horizontal or vertical
- In this paper we consider only horizontal scaling
- In this paper the modelled system will be a telecommunication network element



Load distribution in telecommunication network elements

Network functions in Cloud environment

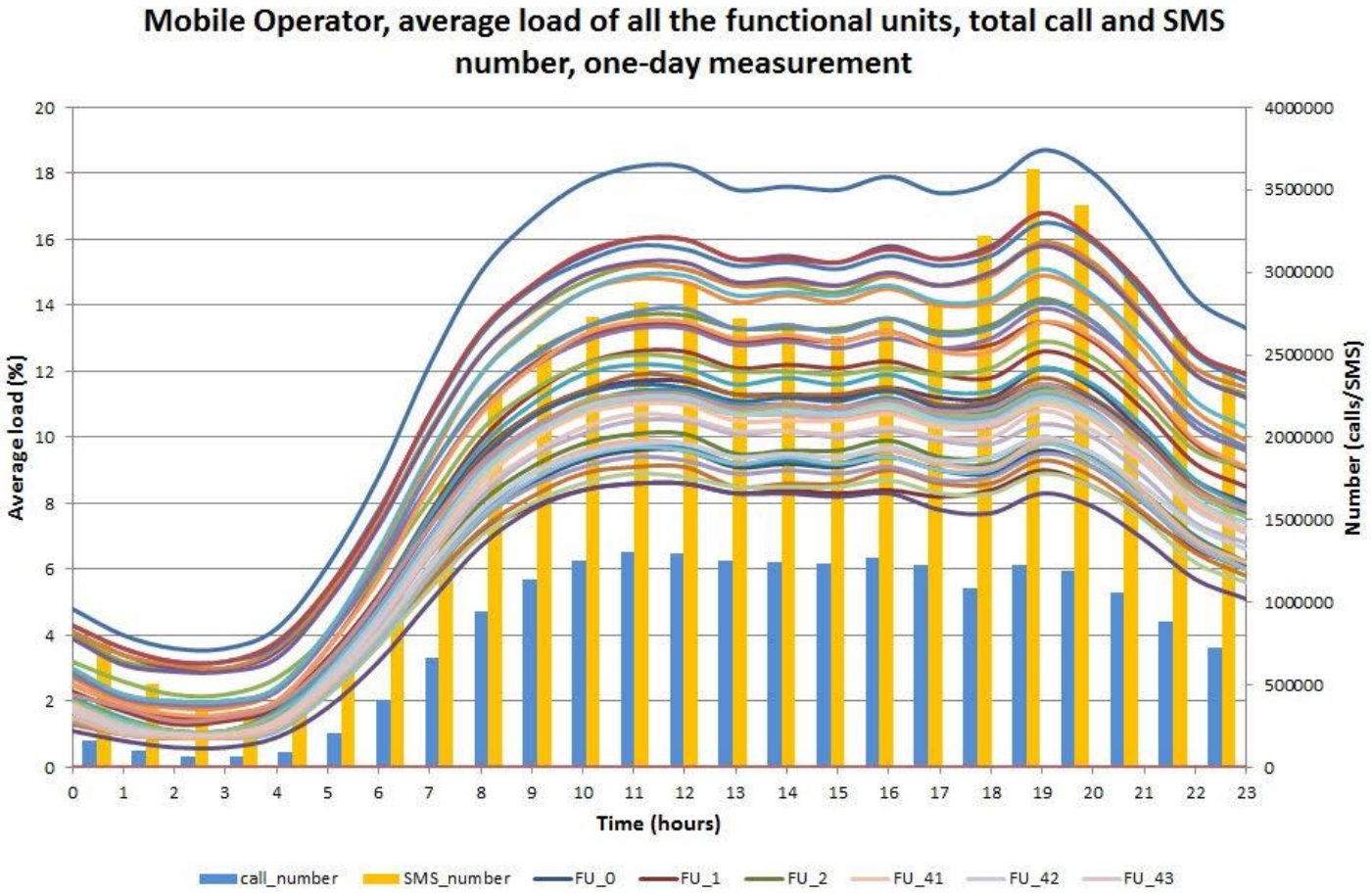
- Traditional solution is static load distribution i.e. the method does not change with the changes of the load
- In virtualized network functions [3] the possibility of dynamic scaling [1], [2] appears
- Our modelling target is the signaling part of the core network



Problem with load distribution and load balancing

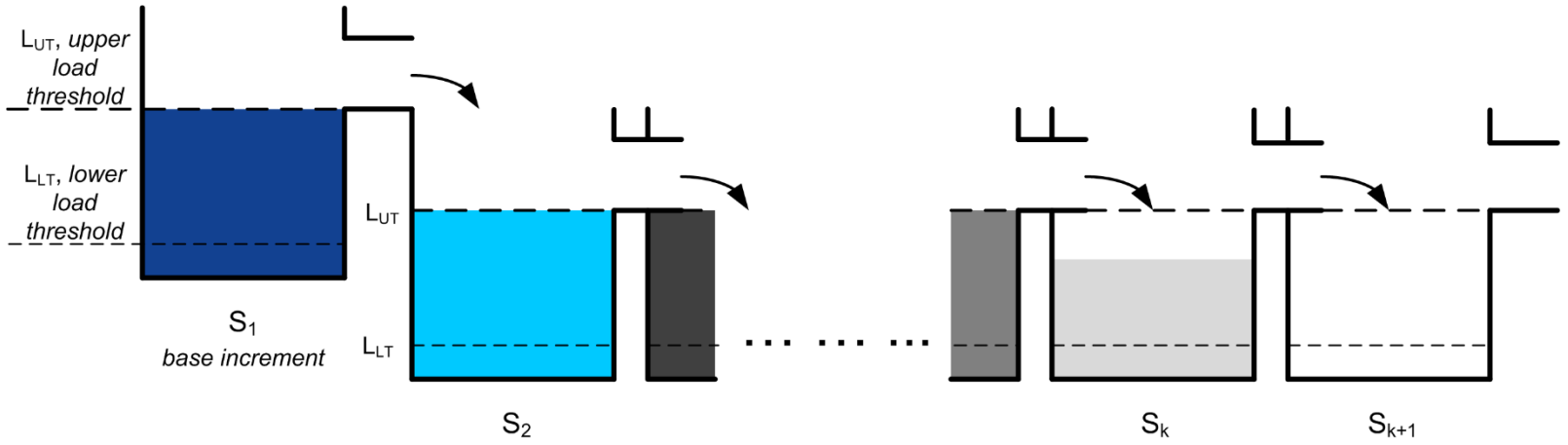
Challenges of current solutions

- The goal of current solutions is to distribute requests evenly
- Instead of load, requests are distributed, although there is a strong correlation
- Static system topology is designed to handle the maximum design load
- Typical operation handles less load
- Load distribution can be modelled with the Round-Robin algorithm



New load distribution algorithm
Requirements for the new logic

- Be able to handle groups of functional units (sets) to achieve a bigger scaling step
- Set load thresholds to limit the maximal load of active sets
- Instead of even distribution allow one set (lower order) to be less loaded and assign the design load to the other active (higher order) sets
- The changes of the system load will appear on the lower order set, allowing better scalability
- Randomize call distribution between the functional units



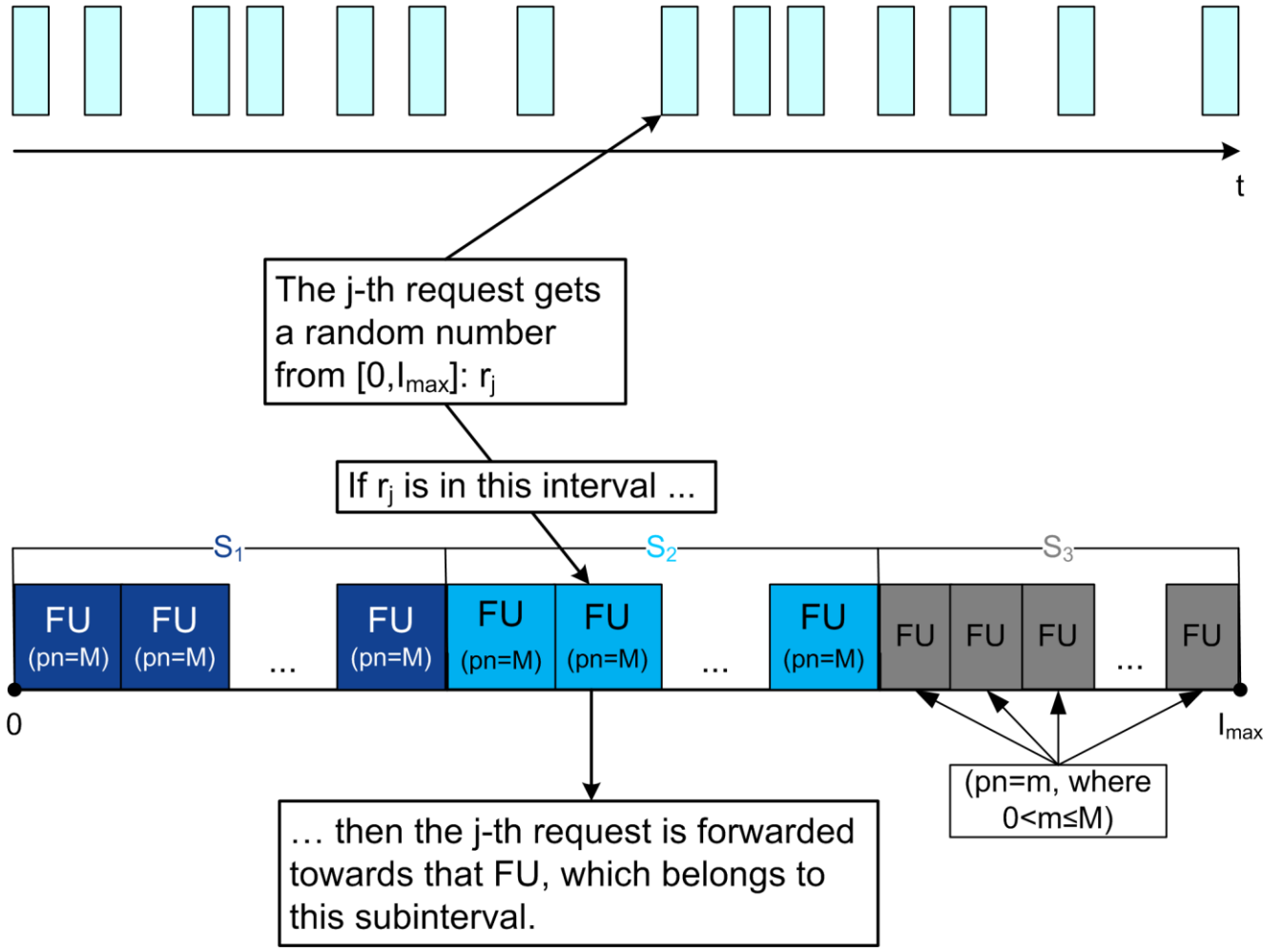
New load distribution algorithm
Solution proposal

- Assign priority numbers to the sets
- Represent sets (and units) with corresponding intervals on the real line
- Assign interval lengths:

$$L_{FU_k} = P_{FU_k} \times F_{FU_k}$$

where P is the priority number, F is the free capacity of the functional unit

- For every call generate a random number with uniform distribution on the union of the intervals $[0, I_{max}]$
- Assign the request to the unit that corresponds to the interval *"hit"* by the random number

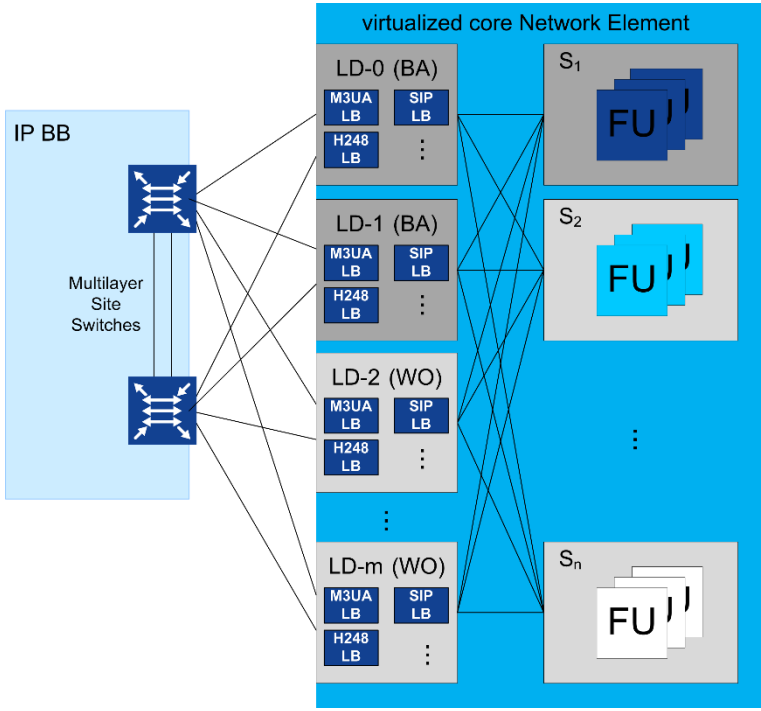


Comparing models for existing and the new method
Efficiency of load distribution

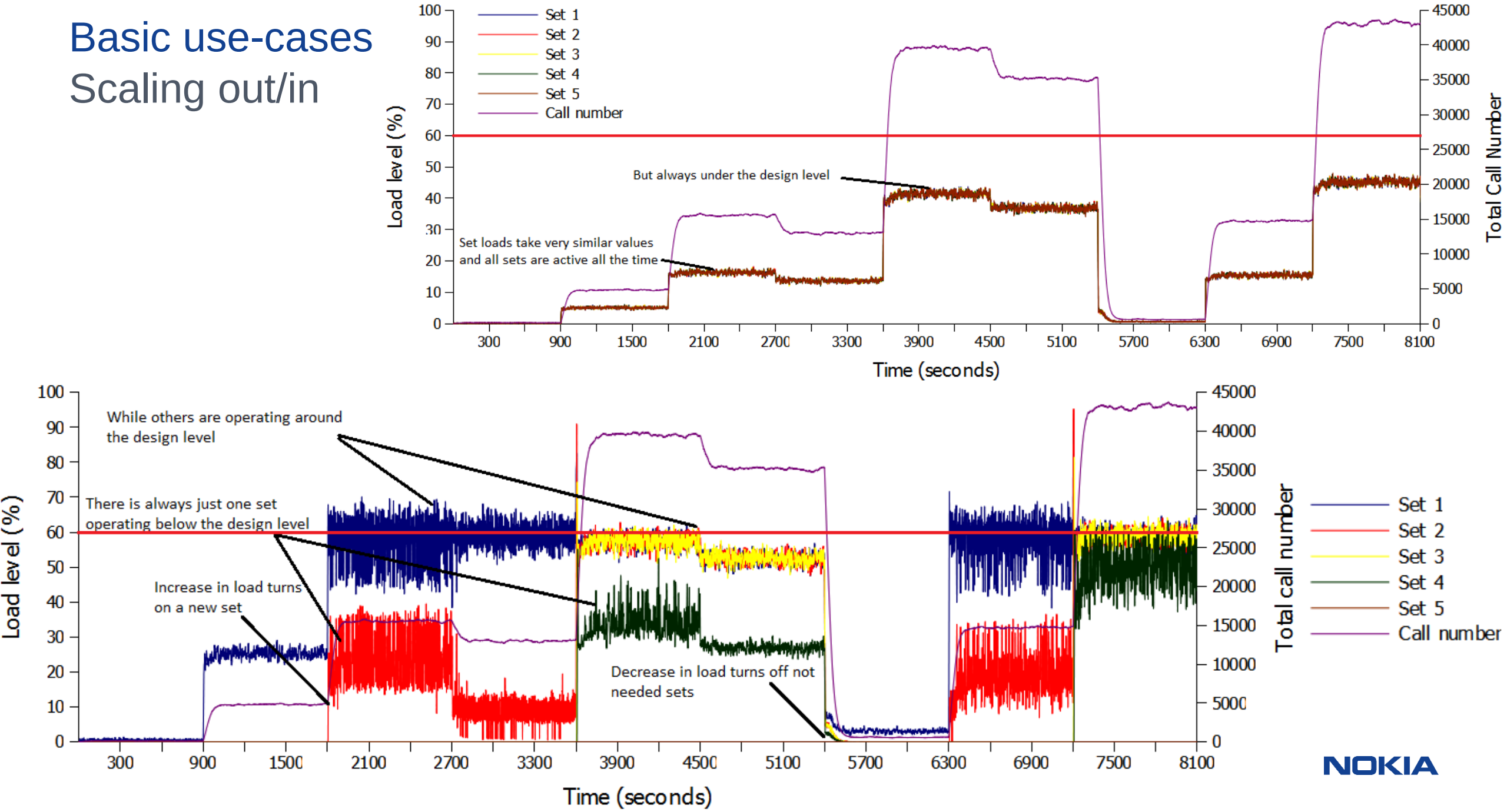
- The existing solution is modelled with Round-Robin
- The new solution is described in [5], called the Horizontal Scaling Algorithm (HSA)
- Simulation is required for both, algorithm logic and call traffic both have to be simulated
- Call arrival model is an inhomogeneous Poisson process
- Call length models are more challenging [4], for simple simulations exponential and log-normal distributions can be used
- Rate-function of the Poisson process can be created based on traffic measurements
- Rate function is piecewise constant (measurement granularity), definition can be improved by spline interpolation
- Goal is to keep as many sets in gracefully shut down (GRSD) state as possible

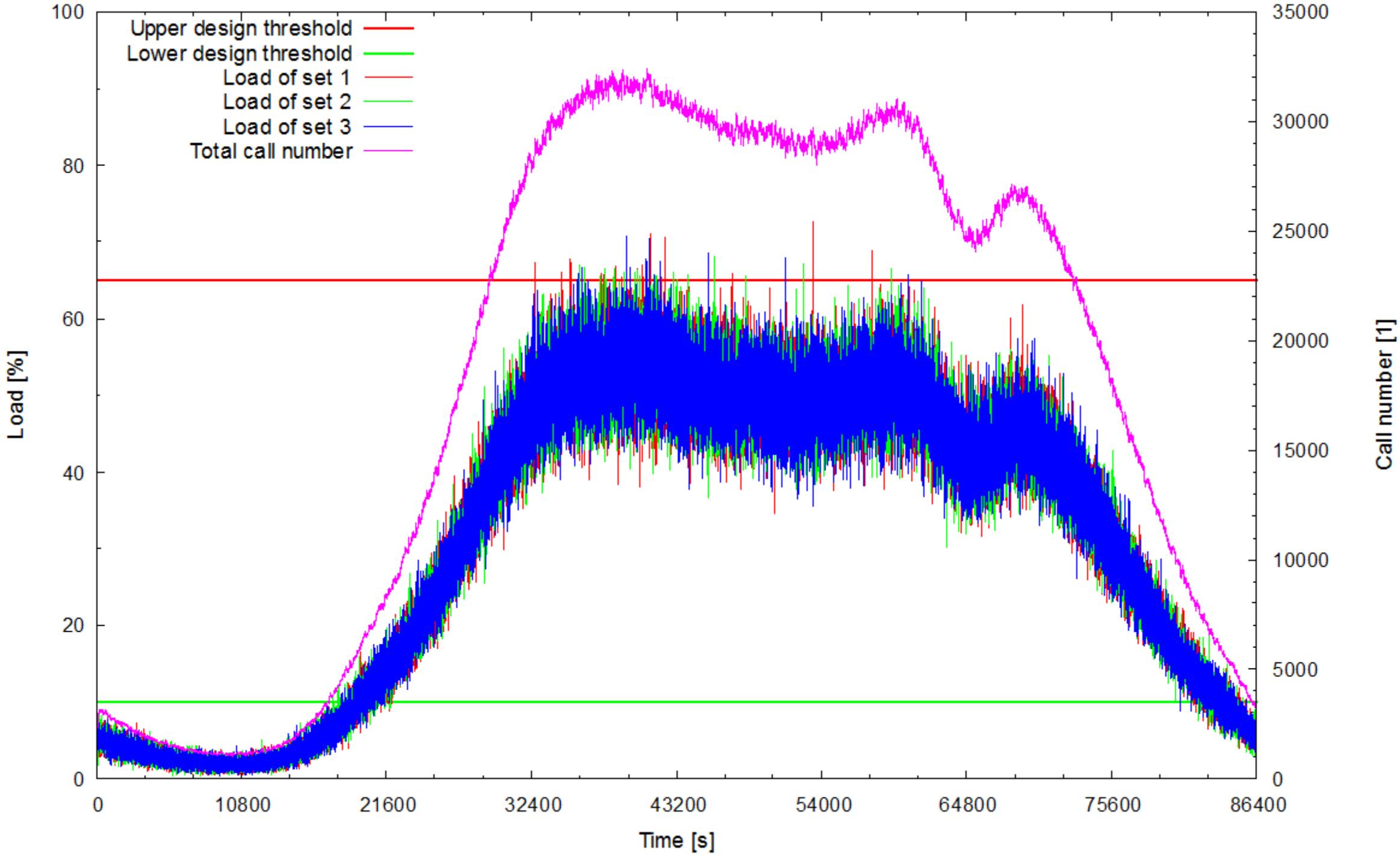
Simulation
Program for comparing models

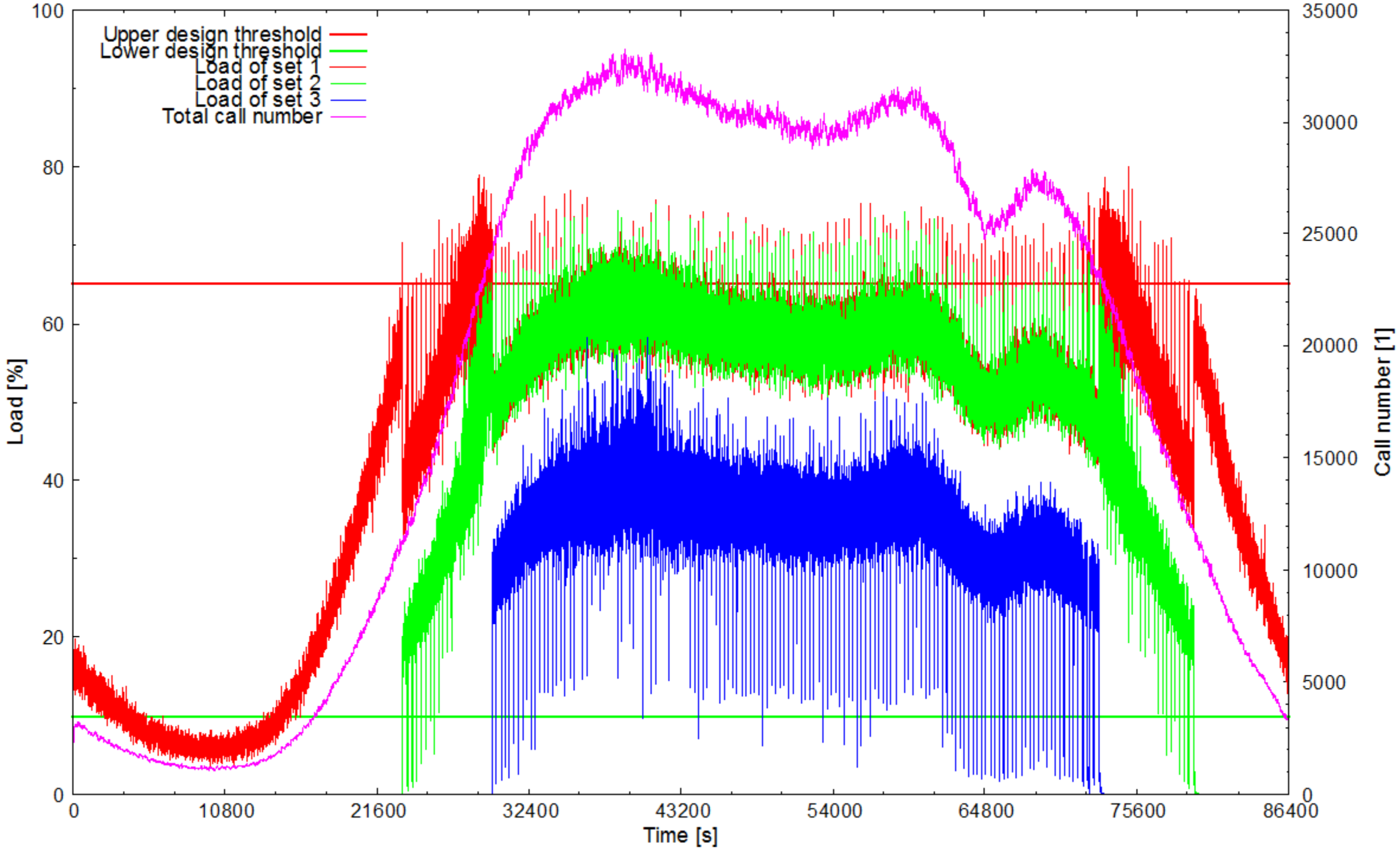
- Simulation was written in c++ with object models for calls, units, sets etc.
- Call length distribution was log-normal with 90 seconds expected value and 50 seconds standard deviation
- Both algorithms (RR and HSA) used the same simulated call set
- First simulations were to determine the different behaviour of the models in prepared use-cases (when scaling was required)
- Later simulations aimed at simulating real traffic profiles
- One day simulation with measured (re-scaled) traffic profile and system capacity based on real network design were created



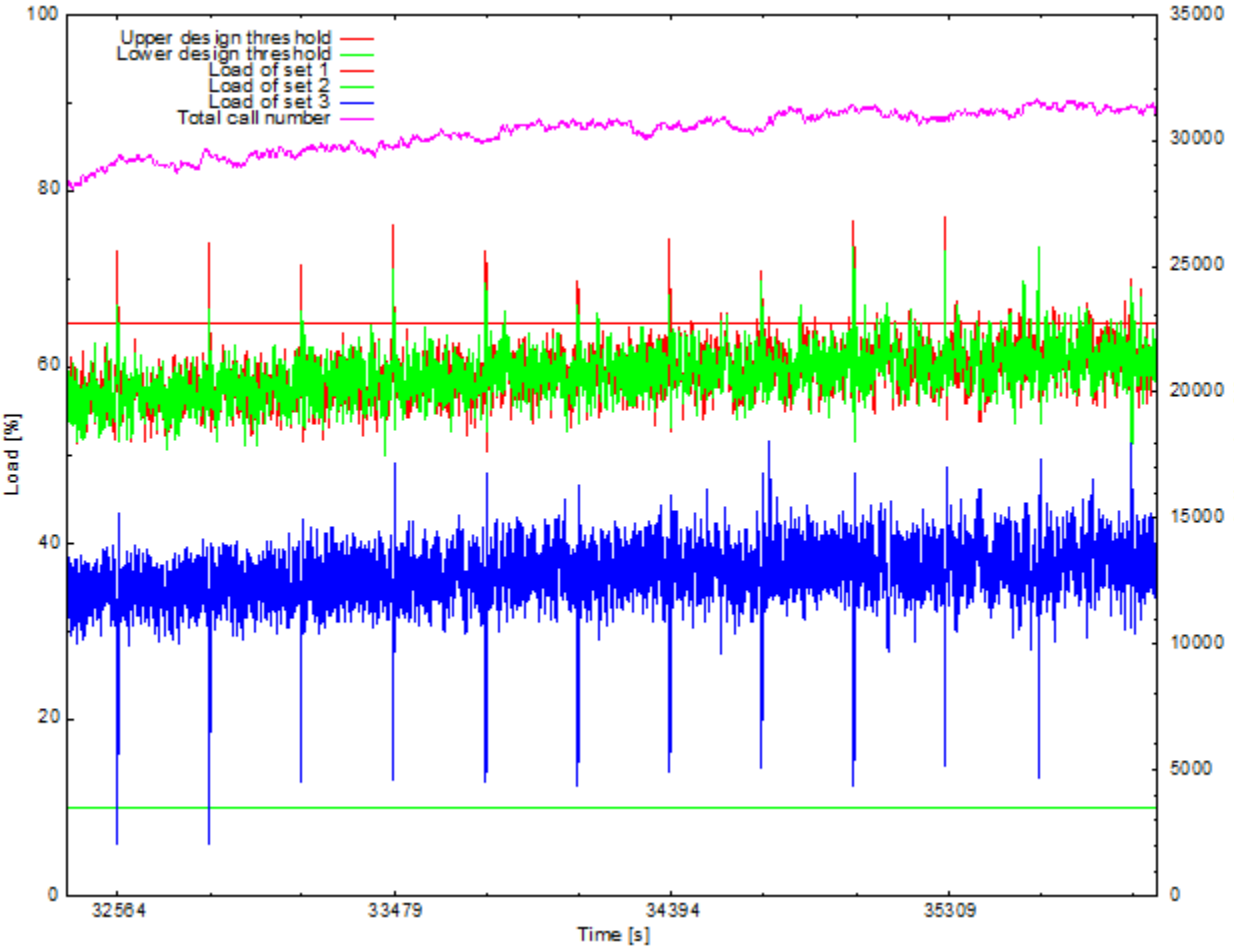
Basic use-cases
Scaling out/in

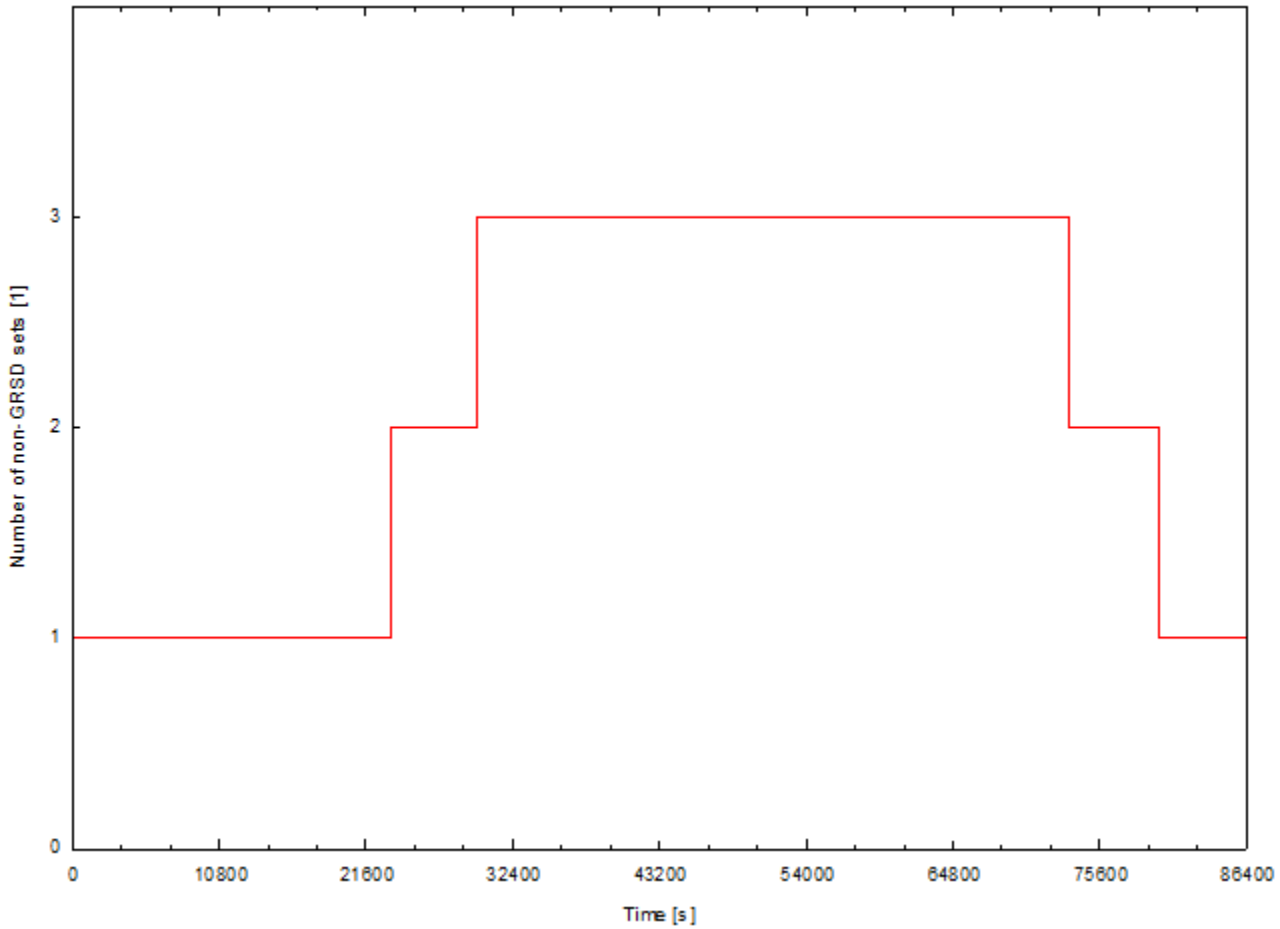



Upper design threshold
Lower design threshold
Load of set 1
Load of set 2
Load of set 3
Total call number
Load [%]
Call number [1]
Time [s]
0
20
40
60
80
100
0
5000
10000
15000
20000
25000
30000
35000
0
10800
21600
32400
43200
54000
64800
75600
86400


Upper design threshold
Lower design threshold
Load of set 1
Load of set 2
Load of set 3
Total call number
Load [%]
Call number [1]
Time [s]
0
20
40
60
80
100
0
5000
10000
15000
20000
25000
30000
35000
0
10800
21600
32400
43200
54000
64800
75600
86400

Challenges of the simulation
Unnecessary switching and GRSD state





Conclusions

Results so far and further possibilities

- Load distribution algorithm was patented [5]
- Simulations show that efficiency can be improved by scaling
- HSA supports graceful shutdown of units / sets effectively
- Decreased capacity – during periods of low load – does not create service degradation at load increase
- Multiple – allowing / strict – thresholds and sliding window averages could improve the algorithm
- Analysis of the effect of parameter values on efficiency and quality could be performed
- The overload handling of HSA could have a modified logic to improve the overload handling
- Call length modelling could be enhanced to reflect real traffic profiles better

[1] Sagar Dhakal, Majeed M. Hayat, Jorge E. Pezoa, Cundong Yang, David A. Bader: ”Dynamic Load Balancing in Distributed Systems in the Presence of Delays: A Regeneration-Theory Approach”, IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 18, NO. 4, Apr. 2007.

[2] Parveen Jain, Daya Gupta: ”An Algorithm for Dynamic Load Balancing in Distributed Systems with Multiple Supporting Nodes by Exploiting the Interrupt Service”, International Journal of Recent Trends in Engineering, Vol 1, No. 1, May 2009.

[3] ETSI GS NFV 002, Network Functions Virtualisation (NFV), Architectural Framework, V1.1.1, Oct. 2013. http://www.etsi.org/deliver/etsi gs/NFV/001 099/002/01.01.01 60/gs NFV002v010101p.pdf

[4] Pedro O.S. Vaz de Melo, Leman Akoglu, Christos Faloutsos, Antonio A.F. Loureiro: ”Surprising Patterns for the Call Duration Distribution of Mobile Phone Users”, Machine Learning and Knowledge Discovery in Databases, Lecture Notes in Computer Science Volume 6323, pp 354-369, 2010.

[5] István Bakos, Gyula Bódog, Attila Hilt, László Jánosi, Gábor Járó: ”Optimized resource management in core network element on Cloud based environment”, Patent PCT/EP2014/075539, Nov. 2014

Thank you for your attention

Q&A

NOKIA