Dynamic System Level Simulations in faster then realtime

Presentation at ITG Workshop, UMIC Aachen, 14july2011

Andreas Lobinger

(Ingo Viering, Martin Döttling, Szymon Stefanski)

The Why and How

- We want to observe what ‘the system’ does.

- Therefore we build a mathematical model.
- We write a program in accordance with that and equip it with reasonable parameters (or just sweep) and observe ‘the system’.
- For a nearer definition of ‘the system’:
  - A set of basestations
  - A (larger) set of UEs
  - An radio interface between them

- Dynamic: We expect changes over time OR expect interactions within ‘the system’.
- Changes over time: service/traffic, user position, basestation parameters etc.

Remember: We need a model

- In Radio Simulations there are two classical paths:

1. Drop the interaction and work with constant signals.
2. Model in detail.

- For the 2nd approach, the question: Which level of detail?

So let‘s say, you want to look at a LTE DL system

- The level of detail (for our model) is (seems) to be defined by the details of the system.
- As LTE DL is an OFDMA system, we need to look at OFDM symbols ?
- … or resource blocks i.e. symbols by set of subcarriers?
- … or transport blocks i.e. a set of resource blocks scheduled together?

- The answer lies somewhere in the middle, but what we need to cover is
- RRM (radio resource management) aka scheduling (+link adaptation)

And the interaction?

- To make a long story short: It‘s Interference (the I in SINR)

- Interference usually (at least in this LTE DL sense) is an uncoordinated, unpredictable, non-controllable term in the SINR calculation

- AND: it‘s a sum

- Nice thing: It depends on the load in other cells and only on that.

- First Idea:

  Formulate user specific SINR as function of Signal, other cells Signal and load

Back to RRM

- Scheduling and Link Adaptation is the modern tooling to approach the boundary of channel capacity: Shannon bound
- So turn it around: Can we use a channel capacity to derive the load?
- -> in every time step, from the user SINR and a certain expected throughput (think packet-size) we can calculate the needed resources
- Simplification: fix the expected throughput (CBR, GBR service), assume QoS scheduling
- Second idea:

  In every time step calculate a cell load based on user SINR and a look at capacity

… so closing the loop

- SINR = function(load)
- Load = function(SINR) ?

- We use an estimate of the load by looking at the previous time step.

What‘s this time step all about?

- Our (main) idea is to keep the (simulation) time step short enough to be able to look at all dynamic effects, but (way) longer than the real RRM granularity.

- Also to make a long story short: 1s to 50ms

Program performance, simple MATLAB implementation

Simulation setup:
600s (= 600 steps)
800 users (~15u / cell)
57 cells

MATLAB profiler

Function Name	Calls	Total Time	Self Time*
cart2sph	600	10.841 s	10.841 s
pathloss_and_beampatterns2	600	23.786 s	8.761 s
distance_subroutine	600	8.674 s	8.487 s
channel_and_sinr_log	600	35.320 s	6.801 s
son	1	56.417 s	4.947 s
inpolygon>vec_inpolygon	1202	4.120 s	4.120 s
log10	1803	3.888 s	3.888 s
logSum	600	3.092 s	3.013 s

Angle in hor/ver

Guess what…

Wrap-around in channel

SINR calc.

main

Wrap-around in mobility

Interference calc.

The steps for simplifications

- Abstract the RRM by a channel capacity (average)
- Calculate the user SINR by estimated cell loads
- Calculate the cell load by user SINR, a scheduling assumption and user traffic.
- Does it work?
- OK, so when does it not work?

Contacts and references

Andreas Lobinger andreas.lobinger@nsn.com

Paper:

ICC09

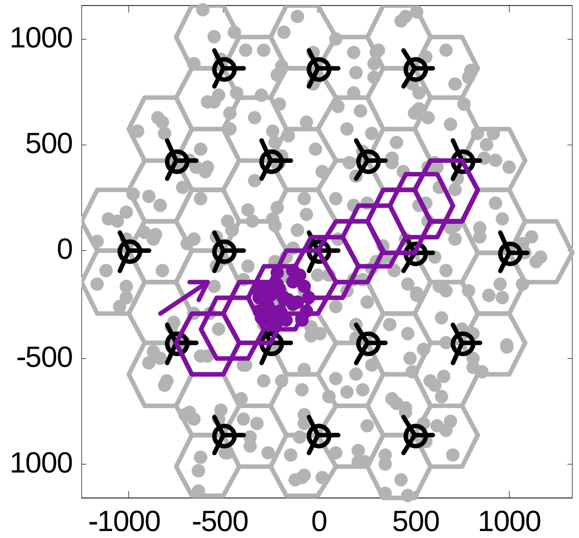
„A mathematical perspective of self-optimizing wireless networks“
Viering, Döttling, Lobinger

EURASIP Journal on Wireless Communications and Networking (online)

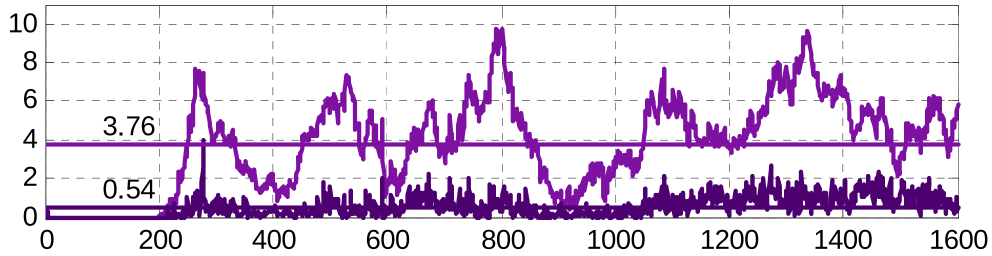
“Efficient Uplink Modeling for Dynamic System-Level Simulations of Cellular and Mobile Networks” Viering, Lobinger, Stefanski

Usecase : SON -> load balancing with moving load

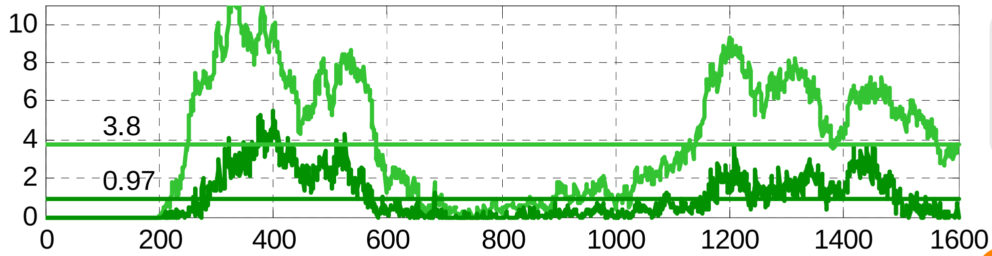
scenario 1, network picture



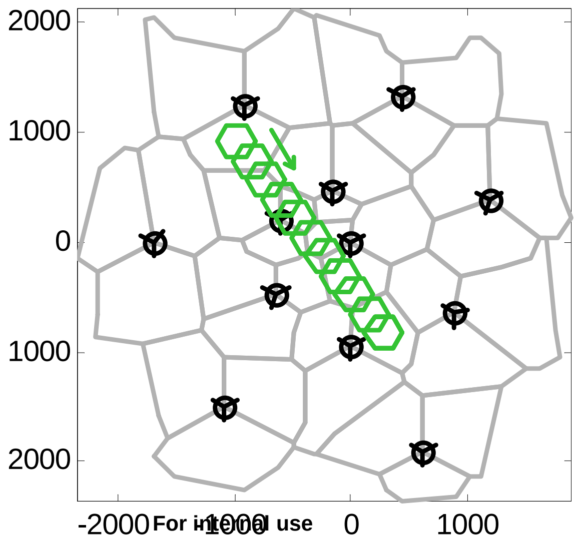
scenario 1, load over time, 40 users in hotspot, 5 users equal load



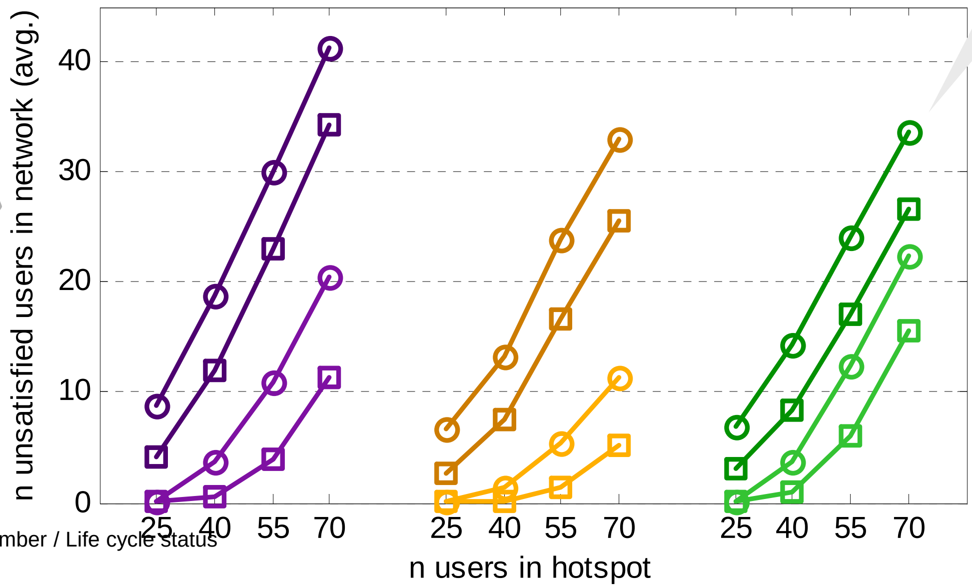
scenario 6



scenario 6, network picture



scenario 1 | 2 | 6 overall comparison

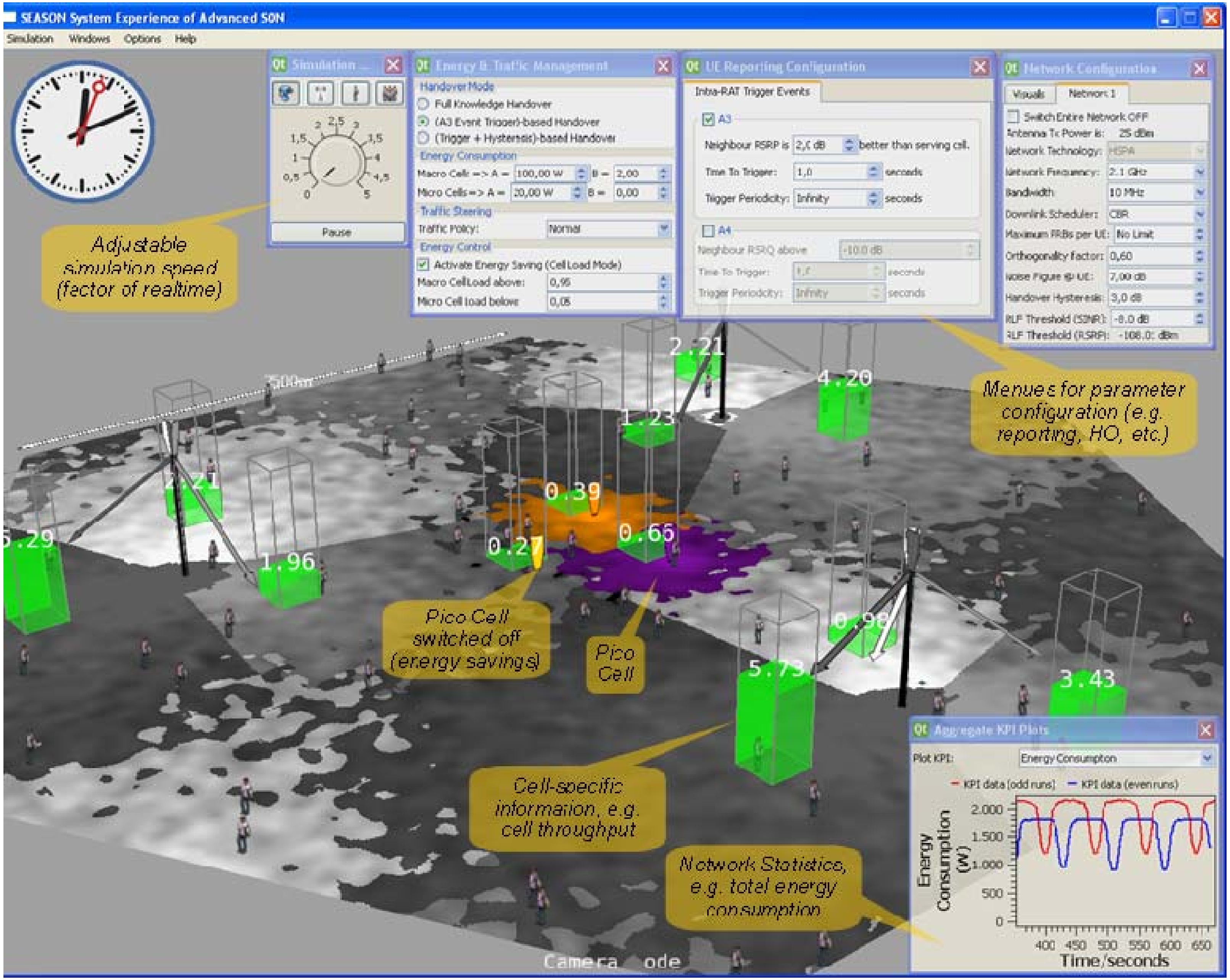


10 experiments per point

3 x 16 = 48 settings

48 x 10 x 1600s -> **210h realtime**

Another Simulator example



Nokia Siemens Networks