SON Coordination for parameter conflict resolution: A reinforcement learning framework

Ovidiu Iacoboaiea[†‡], Berna Sayrac[†], Sana Ben Jemaa[†], Pascal Bianchi[‡]

(†)Orange Labs, 38-40 rue du General Leclerc 92130, Issy les Moulineaux, France

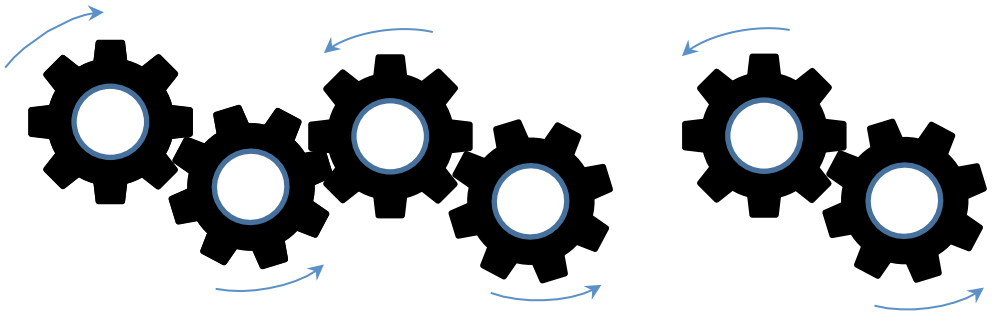
(‡) Telecom ParisTech, 37 rue Dareau 75014, Paris, France

Presentation agenda:

- Introduction
- System Description
- SON Coordination
- Simulation Results
- Conclusions and Future Work

Introduction to SON & SON Coordination

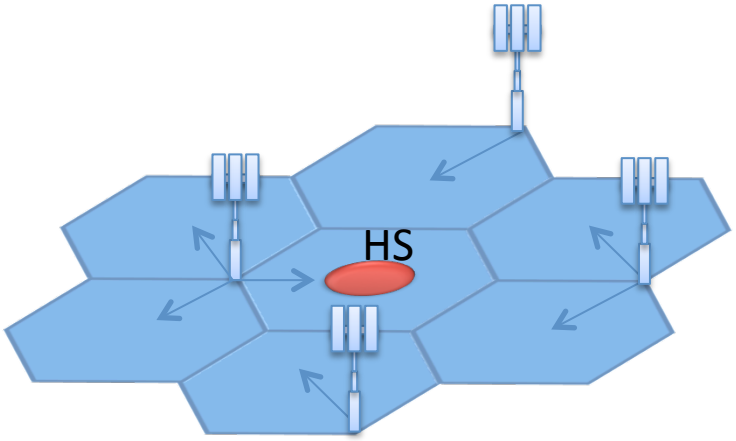
- SON functions are meant to automate network tuning (e.g. MLB, MRO, etc.) in order to reduce CAPEX and OPEX.
- A SON instance is a realization/instantiation of a SON function running on one or several cells.
- In a real network we may have several SON instances of different SON functions, this can generate conflicts.



SON instance 1
(e.g. MLB instance)

SON instance 2
(e.g. MRO instance)

System Description: MLB vs MRO



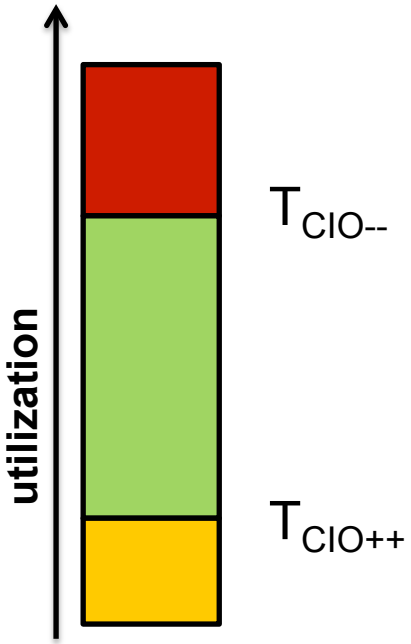
We consider:

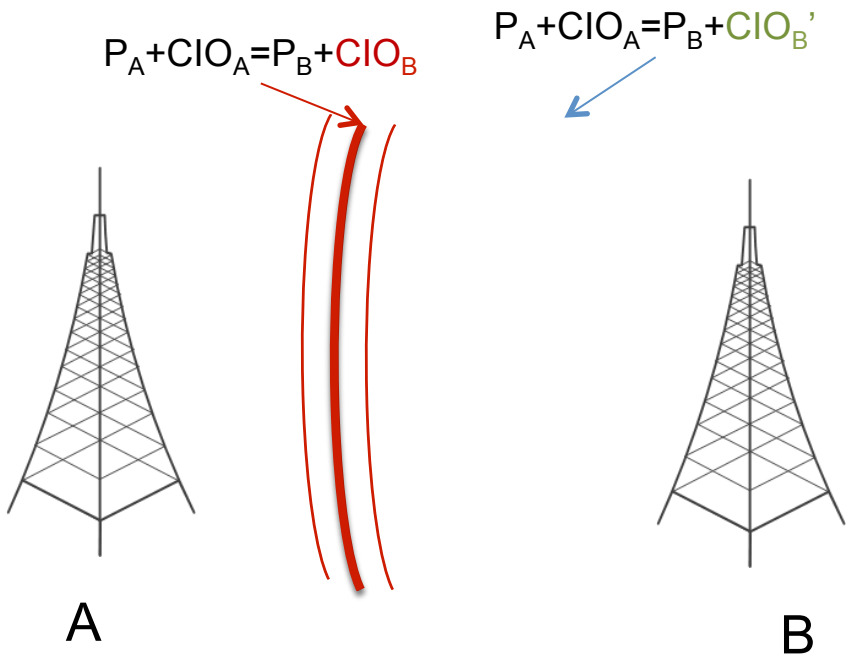
- 2 SON functions: MLB and MRO
- a network segment of N=7 cells
- on every cell we have a MLB instance and a MRO instance
- one HotSpot (HS)
- FTP like traffic
- all SON instances are synchronized (time window 5 min)

System Description: MLB

Looking at base station B:

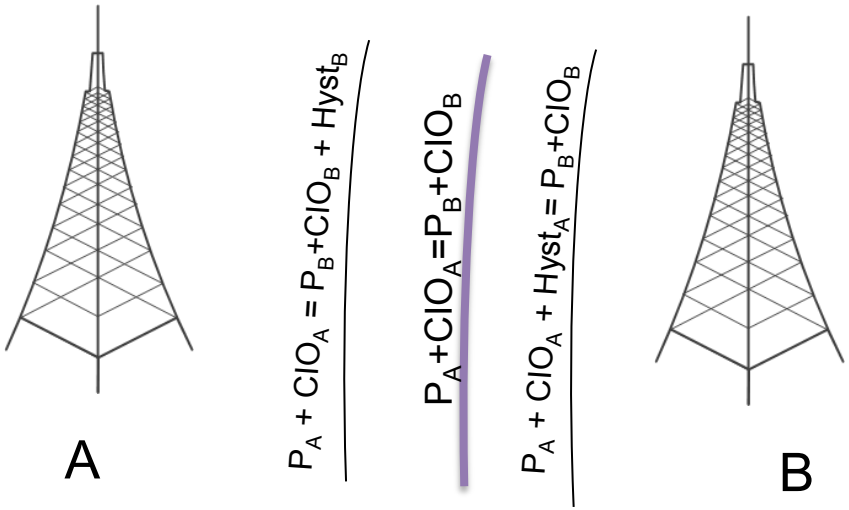
- if(utilization > T_{CIO--}) ➔ CIO_B--
- if(utilization < T_{CIO++}) ➔ CIO_B++ (for going back to the default configuration)
- else ➔ Relax (void)





CIO range (-12dB:0dB/ step-size 3dB)

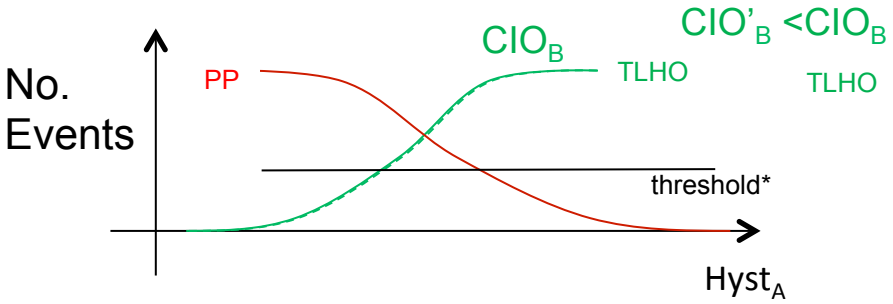
System Description: MRO



For Cell A:

- **Too Late HOs (A→B)**

- **Ping-Pongs (A→B→*)**
- Too Early HOs (A→B(RLF)→A)
- Wrong Cell HOs (A→B(RLF)→*)



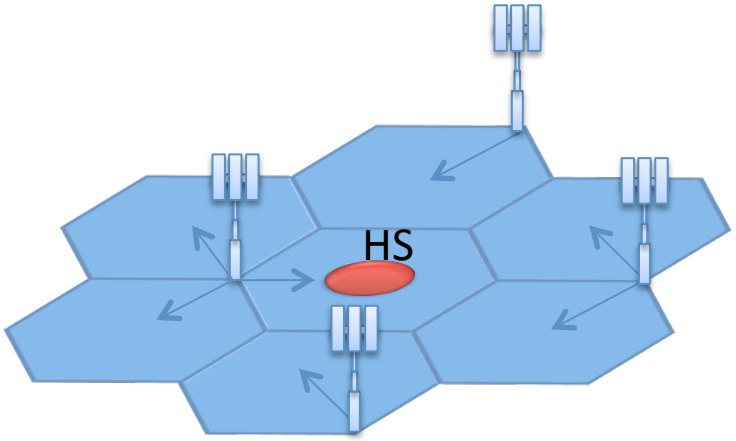
* For this example we considered the same threshold for all the parameters

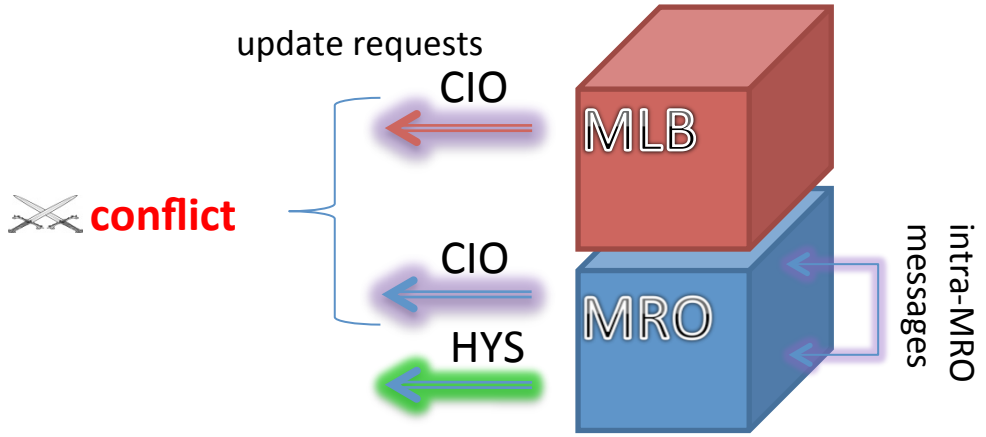
MRO target: reduce the number of Too Late HOs, Too Early HOs, Wrong Cell HOs, Ping-Pongs

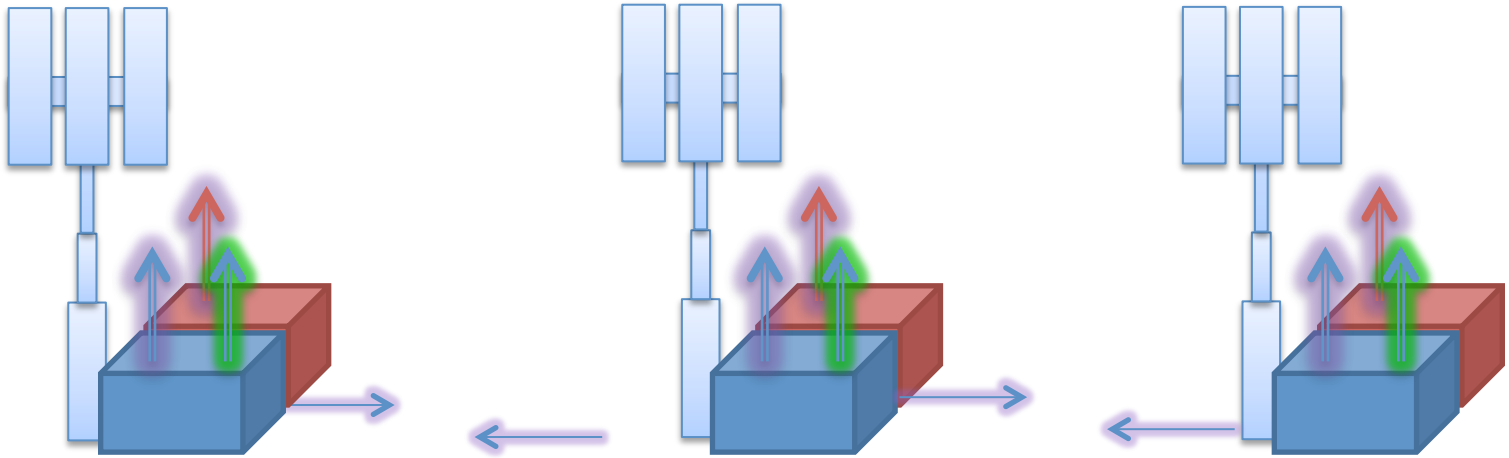
Example: Consider Cell A. as source for all the following events.

- to reduce the No. of Ping-Pongs we increase $Hyst_A$
- to reduce the No. of TooLateHOs we decrease $Hyst_A$
- to do both the above one solution is to first decrease the CIO of the neighbor towards which cell A has the biggest No. of TooLateHOs (assume it is cell B ➔ CIO_B--). Afterwards we will most-likely increase $Hyst_A$.

SON Coordination







SON Coordination

- An update request contains
  - the parameter request: increase/decrease/maintain
  - an information on the criticalness of the request

Reinforcement Learning helps us learn from past decisions:

- create a state space that includes the CIOs of the cells(c)
- target: find the state that maximizes a "reward"
- the "reward" is a function of
  - the happiness indication contained in the update requests from the SON instances
  - weights associated to the SON functions by the operator

Function Init:

Initialize $W(c) = 0$ *and* $e(c) = 0$ *for all* $c \in \mathcal{C}$

Initialize $s = ((0,0,0),(0,0,0))$

Function SONCO:

Observe current state $s' = (c', u')$ *and*

calculate reward $r(s') = \rho(u')$

$\delta \leftarrow r + \gamma W(c') - W(c)$

$e(c) \leftarrow e(c) + 1$

for all $\tilde{c} \in \mathcal{C}$

$W(\tilde{c}) \leftarrow W(\tilde{c}) + \alpha\delta e(\tilde{c})$

if $a'^{C} = a^{C,opt}$ *then* $e(\tilde{c}) \leftarrow \gamma\lambda e(\tilde{c})$

else $e(\tilde{c}) \leftarrow 0$

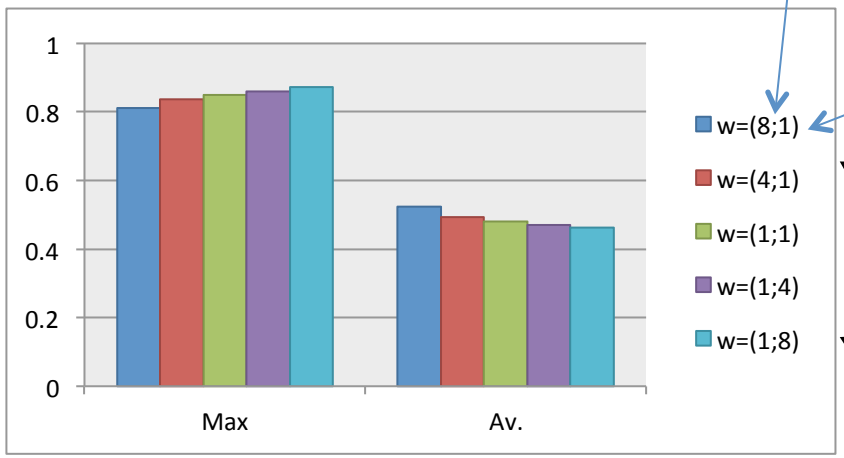
Choose action $a' = (a'^{C}, a'^{H})$ *using policy* π,

Calculate $a^{C,opt}$,

Take action a',

$s \leftarrow s'(\equiv (c,u) \leftarrow (c',u'))$

Simulation Results



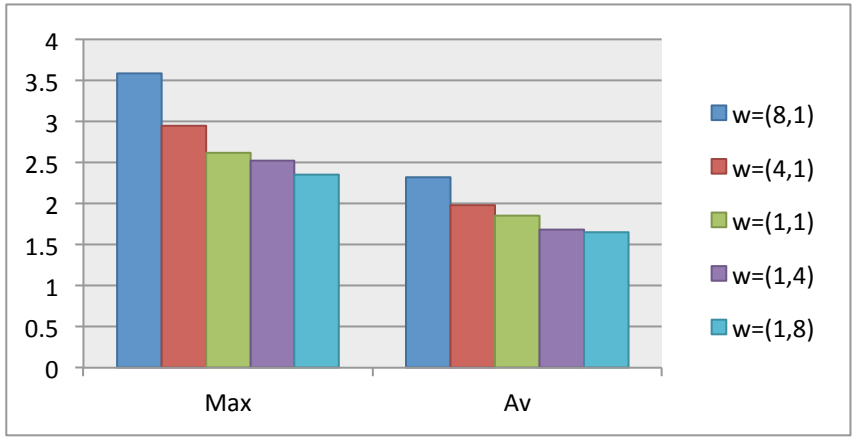
MLB weight

MRO weight

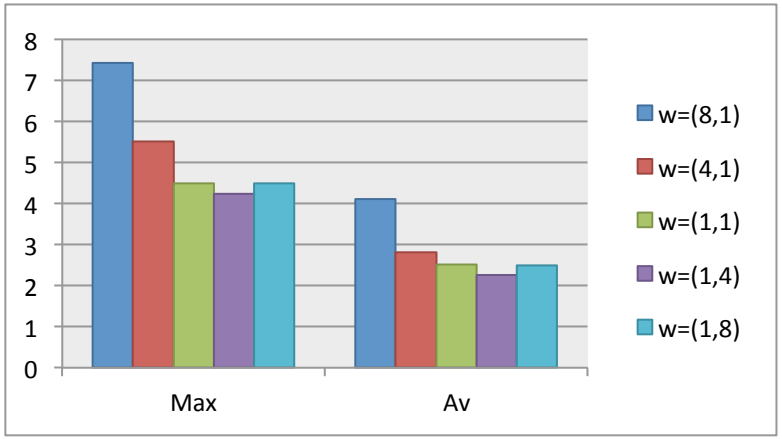
High priority to MLB

High priority to MRO

average load



No. Too Late HOs [#/5min]



No. Ping-Pongs [#/5min]

(Maximum and average over all cells)

Conclusion and future work

- RL proves to have the qualities that allow us to tune the arbitration in favor of one or the other with respect to priorities/weights defined by the operator.
- By using a SON Coordinator we were able to intelligently decide when to accept/deny the requests of the SON instances in order to reach the desired tradeoff between the two SON functions w.r.t. to their corresponding targets (avoiding over-loads for MLB versus decreasing the number of connection failures and ping-pongs due to mobility for MRO).
- Simulation results show that the resulting KPIs reflect the operator priorities.
- Future work will focus on improving the scalability of our solution.

Questions