Cognitive Management of Self -Organized Radio Networks Based on Multi Armed Bandit

Tony Daher, Sana Jemaa, Laurent Decreusefond

Cognitive Management of Self - Organized Radio Networks Based on Multi Armed Bandit

Tony Daher, Sana Ben Jemaa
Orange Labs
44 Avenue de la Republique 92320 Chatillon, France
Email:{tony.daher,sana.benjemaa}@orange.com

Laurent Decreusefond
Telecom ParisTech
23 avenue d'Italie, 75013 Paris, France
Email:laurent.decreusefond@mines-telecom.fr

***Abstract*—Many tasks in current mobile networks are automated through Self-Organizing Networks (SON) functions. The actual implementation consists in a network with several SON functions deployed and operating independently. A Policy Based SON Manager (PBSM) has been introduced to configure these functions in a manner that makes the overall network fulfill the operator objectives. Given the large number of possible configurations (for each SON function instance in the network), we propose to empower the PBSM with learning capability. This Cognitive PBSM (C-PBSM) learns the most appropriate mapping between SON configurations and operator objectives based on past experience and network feedback. The proposed learning algorithm is a stochastic multi-armed bandit, namely the UCB1. We evaluate the performances of the proposed C-PBSM on an LTE-A simulator. We show that it is able to learn the optimal SON configuration and quickly adapts to objective changes.**

I. Introduction

Because of the increase in the number of wireless devices that are accessing mobile networks and the emergence of services generating bigger amounts of traffic, mobile data traffic is growing drastically [1]. To cope with this growth, mobile networks are expected to become more dense, with different layers of cells and many technologies deployed. Such networks are complex to manage, in particular with operators seeking to maintain the quality of service of the users, while sustaining their profits level. The 3rd Generation Partnership Project (3GPP) has already anticipated this issue by introducing the Self-Organizing Networks (SON) [2] in its 8th release. SON functions are currently deployed independently, each replacing a specific operational task. It is agreed that deploying simultaneously such various functions in a network, without any kind of orchestration and coordination, will hardly lead to an optimal operation of the network in overall [3]. Making these functions cooperate to respond as a whole to the operator objectives is crucial. The question is then how to design a global SON management system that coordinates and orchestrates the SON functions, in order to fulfill the operator's objective.

978-1-5386-3531-5/17/$31.00

SON functions behave differently depending on the configuration of the algorithm they are running (step size, thresholds, parameter intervals ...) and the cell context (cell location, technology ...) where they are deployed. The SON function configurations are hence a leverage for the operator to control the deployed SONs in a way that fulfills its objectives. The task of the manager is thus to find the best configuration for each instance of each deployed SON function, taking into account the cell context where it is deployed as well as the interaction with other deployed functions. Furthermore, SON functions are usually designed by Radio Access Network (RAN) vendors in a proprietary manner i.e. they are seen by the operator as almost black boxes [3]. Consequently, taking account of the increasing complexity of networks and SON functions, this task becomes complex. A possible solution for this problem is through Reinforcement Learning (RL) [4]: introducing an agent that learns the best configurations by monitoring the network's Key Performance Indicators (KPIs).

In this paper we study the use of RL, more precisely the Multi-Armed Bandit (MAB) [5], to automate and improve the management of SONs by learning the optimal configuration of the deployed SON functions from the real network. Our contribution can thus be summarized by the following: 1) modeling the SON management problem as an RL problem, 2) proposing the MAB as an appropriate RL technique to solve the problem and 3) assessing the performance of the proposed approach using an LTE-A system level simulator. The rest of the paper is organized as follows. Section II presents the state of the art on the management of SON and the applications of MAB in wireless networks. Section III describes the proposed management model using MAB. Scenario description and performance evaluation are presented respectively in sections IV and V. Section VI concludes the paper.

II. State of the Art

A global SON management framework has already been introduced in the context of the European project Semafour [3]. In [3] and [6], the authors propose a Policy Based SON Manager (PBSM) that translates

automatically operator objectives into SON function configurations. The proposed translation is based on the so called SON Function Models (SFM). The SFMs are the mapping of the SON function configuration to the resulting KPIs. This mapping is obtained through extensive offline simulations performed once. Then, the PBSM deduces the global system performance by comparing the combination of the individual SFMs to the operator's KPI objectives. This combination does not take into account potential interdependencies between the individual SON functions. Moreover, the SFMs are obtained by simulation and do not reflect the network reality, especially that the SON function's behavior depends on the context of the cell where it is deployed. Therefore, we propose to introduce cognition into the PBSM, that we henceforward call C-PBSM, through an RL algorithm that configures the deployed SON functions by learning the optimal configuration from the real network.

MAB is an RL problem, formulated as a sequential decision problem, where at each iteration an agent is confronted with a set of actions called arms, each when pulled, generates a reward. The agent is only aware of the reward of the arm after pulling it. The objective is to find a strategy that permits to identify the optimal action, while maximizing the cumulated rewards obtained during the process. There are different formulations of the MAB in the literature [7], such as stochastic, adversarial, contextual, linear, etc. In our case we consider the stochastic MAB (this choice is motivated in the following sections). MAB has already been studied and used in wireless access networks and showed to be useful in many problems for example in resource allocation problems [8], interference coordination [9] as well as spectrum allocation [10].

III. Policy Based SON Management Using Multi Armed Bandit

A. Problem Formulation and Motivation

We consider a network section where the traffic is stationary and unequally distributed. To make the PBSM cognitive i.e. C-PBSM, we introduce a learning agent on top of the SON functions as shown in figure 1. This agent takes actions by configuring all the instances of the SON functions in the network. The set C of possible actions is defined as follows. Let S be the set of SON functions deployed in the network. Let N_s be the number of deployed instances of a SON s and V_s be the set of possible SON function Configuration Values (SCV) sets for SON s, $\forall s \in S$ (i.e. assuming that all the instances of the same SON function have the same set of possible SCV sets). Then we define the action set as: $C = (V_{s_1})^{N_{s_1}} \times (V_{s_2})^{N_{s_2}} \times ... \times (V_{s_{|S|}})^{N_{s_{|S|}}}$ where $s_1, s_2, ..., s_{|S|} \in S$.

For each action $c \in C$, the C-PBSM receives a reward r, evaluated based on a combination of network KPIs,

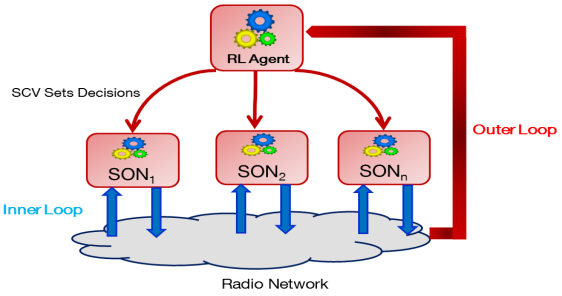

Fig. 1: C-PBSM based on RL

once all the individual SON functions have converged as depicted on figure 2 (we consider that $\forall c \in C$, all the SONs converge after a sufficient time). The agent is aware of the reward of an action only after applying it to the network for a sufficient time: at each iteration t, the agent picks $c \in C$ and then evaluates the corresponding reward $r_{t,c}$. Under the conditions of traffic stationarity and SON convergence, the observed KPIs converge towards the same distribution, independently of the previous actions, $\forall c \in C$. It is then reasonable to consider that, for the same configuration combination c, the observations of r are i.i.d. random variables following an unknown probability distribution. The best action (the one that satisfies best the operator's objective targets) is defined as the action that has the highest expected reward:

$$c^* = \underset{c \in C}{\operatorname{argmax}}(\mathbf{E}(r_c)) \quad (1)$$

The agent's task is to sequentially explore the set of actions C in the real network in order to find c^*. The question is: what should be the agent's exploration strategy so that it reaches as fast as possible its objective (finding c^*), while minimizing suboptimal decisions (i.e. minimizing the iterations where the agents choses $c \neq c^*$). This is also known as the exploration/exploitation dilemma. The C-PBSM's sequential learning process is represented in figure 2. In the following we will present

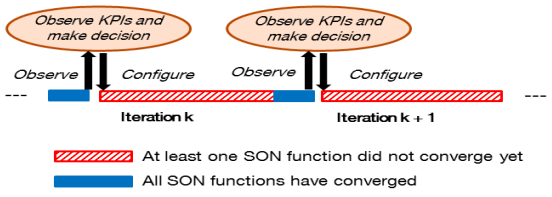

Fig. 2: C-PBSM sequential learning process

the MAB and show that we can find good strategies that balance exploration and exploitation.

B. Multi-Armed Bandit

We propose to use a stochastic MAB, which is an RL framework where the rewards of each arm are

supposed to be an i.i.d sequence following an unknown distribution, specific to each arm. Hereafter we consider C the set of arms, ν_c and μ_c are respectively the reward's unknown probability distribution and the unknown average reward of arm c. The MAB learning process is the following:

For $t = 0, 1, 2, ...$:

- The agent selects an arm $c_t \in C$ according to some policy
- The environment outputs a vector of rewards $\mathbf{r}^t = ({r_1}^t, {r_2}^t, ..., {r_n}^t) \in [0, 1]^n$
- Agent observes only ${r_c}^t$

The MAB's objective is to define a strategy that finds the optimal arm, while minimizing the pseudo-regret defined as follows:

$$\overline{R}_n = \underset{c \in C}{max} \mathbf{E}[\sum_{t=0}^{n} r_{c,t} - \sum_{t=0}^{n} r_{c_t,t}] \quad (2)$$

The expectation is taken with respect to the draw of arms and rewards. Let $\mu^* = max(\mu_c)$ for all $c \in C$. Then the pseudo-regret can be written as:

$$\overline{R}_n = n\mu^* - \mathbf{E}[\sum_{t=0}^{n} \mu_{c_t}] \quad (3)$$

The UCB1 algorithm is based on the Upper Confidence Bound (UCB) strategy, and was introduced first in [5]. The algorithm does not need any preliminary knowledge on the rewards distributions. Each arm is associated with an index composed of 2 terms: the first is the empirical average of the perceived rewards and the second is related to the upper confidence bound derived from the Chernoff-Hoeffding bounds.

UCB1 Algorithm
$\alpha > 0$ input constant parameter
for each iteration $t = 0, 1, 2...$
- select arm c_t that maximizes $\widehat{\mu}_{c_t}^{t-1} + \sqrt{\frac{\alpha log(t)}{2{N_{c_t}}^{t-1}}}$
- observe reward ${r_{c_t}}^t$
- evaluate $\widehat{\mu}_{c_t}^t$
- ${N_{c_t}}^t = {N_{c_t}}^{t-1} + 1$
where $\widehat{\mu_{c_t}}^{t-1}$ is the empirical average
and ${N_{c_t}}^{t-1}$ the number of times arm
c_t was pulled at iteration $t - 1$

The UCB1 has theoretical guarantees on the pseudo-regret: in [5] the authors show that the UCB1 achieves an expected regret bound of $O((|C|log(t))/\Delta)$ where $|C|$ is the number of arms and Δ is the difference between the expected rewards of the best and the second best arm. This upper bound matches with the lower bound on the pseudo-regret that can be achieved using any other strategy, making the UCB1 a reasonable solution for the stochastic MAB.

IV. Scenario Description

We consider a 2 layers heterogeneous LTE-A network. A set S of decentralized SON functions is deployed:

a) Mobility Load Balancing (MLB): Deployed on each macro cell. Its objective is to balance the load between macro cells by iteratively comparing the observed cell load to predefined upper and lower load thresholds and tuning the Cell Individual Offset (CIO) parameter accordingly.

b) Cell Range Expansion (CRE): It optimizes the load difference between the pico cell and the macro cell where it is deployed, by tuning CIOs using an iterative process similar to MLB.

c) Enhanced Inter-Cell Interference Coordination (eICIC): Its objective is to protect pico cell edge users (attached to a pico cell because of CRE's offset), from the high levels of interference caused by the closest macro cell (due to the power difference between the pico and the macro cells). To do so, the eICIC tunes the number of Almost Blank Subframes (ABS) in the macro cell frames (ABS contains only control signals transmitted at reduced power).

These functions affect directly two KPIs: the load variance and the user throughput. A well balanced network is in fact more robust to traffic variations and has lower call block rates that is caused mostly by overloaded cells. Under the effect of the CIO, traffic will offload from one cell to another, meaning that users will not attach necessarily to the best serving cell in term of received power. This causes a degradation of the average user throughput in the network. The eICIC's objective however is to improve the throughput of pico cell edge users by requesting ABS from the corresponding macro cell. While this procedure improves the throughput of cell edge users, it affects the resources of the macro cell which may cause a load increase, thus increasing the load variance in the network. It is clear that these functions influence each another when deployed simultaneously in a network.

A. Network Model

To test the UCB1, we consider a section of a 2 layer heterogeneous LTE network as shown in figure 3. The traffic distribution is unbalanced and stationary (high traffic in cells $\{1, 3, 4, 9, 11\}$). Additional traffic hotspots are randomly deployed in certain cells, they are each served by a small cell. We consider the following SON deployment: an MLB on each macro cell, an eICIC on each macro cell where small cells are deployed and a CRE on each small cell. Each of these functions can be configured with SCV sets (table I). Soft configuration means a configuration that reduces the load difference to some extent without reaching high levels of CIO and by aggressive we designate a configuration that seeks to equilibrate the load as much as possible by configuring higher levels of CIO. The total number of possible actions is: $2^{N_{s_1}} \times 2^{N_{s_2}} \times 3^{N_{s_3}}$, where N_{s_1}, N_{s_2} and N_{s_3} are respectively the number of eICIC, CRE and MLB instances in the network. In our case, $N_{s_1} = 5$,

$N_{s_2} = 10$ and $N_{s_3} = 13$. There are approximatively 5×10^{10} possible actions. It is impossible to consider such a big set of actions, especially that the regret of MAB scales linearly with the number of actions as stated in the previous section. In order to reduce the complexity of the algorithm, we configure classes of similar cells instead of individual cells [11]. We consider 2 cell classes: A class of macro cells with a layer of small cells ($C_1 = \{1, 3, 4, 9, 11\}$) and a class of single layer macro cells ($C_2 = \{2, 5, 6, 7, 8, 10, 12, 13\}$). This leaves us with 12 actions for C_1 and 3 for C_2, hence a total of 36 possible SON configurations, which is a reasonable number of actions to apply UCB1.

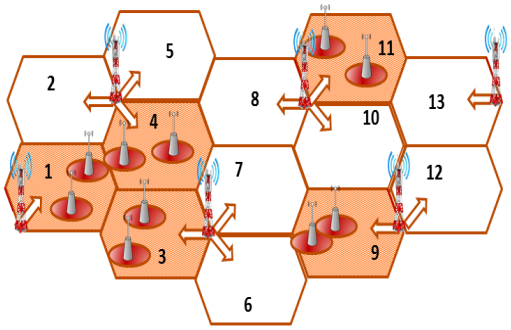

Fig. 3: Network Model

SON	SCV sets
MLB	Off: Function is turned off SCV1: Soft configuration SCV2: Aggressive configuration
CRE	SCV1: Soft configuration SCV2: Aggressive configuration
eICIC	Off: Function is turned off SCV1: Function is turned on

TABLE I: SCV sets behavior description

B. Proposed C-PBSM based on MAB

We define the following KPIs:

- $L_{i,c,t}$ is the load of cell i
- $\overline{L}_{c,t}$ is the average load in the considered section
- $\overline{T}_{c,t}$ is the average user throughput in the considered section
- $\overline{T'}_{c,t}$ is the average pico cell edge user throughput in the considered section

$c \in C$ and t is the iteration. The variables were normalized between 0 and 1. The reward function reflecting the operator's objective is:

$$r_{c,t} = \omega_1(1 - \sigma_{c,t}) + \omega_2\overline{T}_{c,t} + \omega_3\overline{T'}_{c,t} \quad (4)$$

Where the load variance is:

$$\sigma_{c,t} = \frac{\sum_{i=0}^{B}(L_{i,c,t} - \overline{L}_{c,t})^2}{B}$$

B is the number of cells in the considered section and ω_1, ω_2 and ω_3 are weights set by the operator depending on its priorities. They are positive and add up to 1. Furthermore, since r is a linear combination of weights and KPIs, we consider that the agent preserves, besides the empirical average of the perceived reward, an empirical average of the considered KPIs, for each c. This knowledge of these KPIs allows the algorithm to adapt faster to the operator's priority changes.

V. Simulation Results

The C-PBSM is tested using an LTE-A system level simulator based on the 3GPP specifications [12]. It is a semi-dynamic simulator that performs correlated snapshots with a time resolution of 1 second. We take into account path loss and shadowing. Users arrive in the network according to a Poisson arrival, download a file of a fixed size, and leave the network once the download is complete. Users can be dropped due to lack of coverage. We consider only down-link traffic. Table II summarizes the main simulation parameters

Parameters	Settings
Bandwidth	10 MHz
PRBs per eNB	50
Carrier Frequency	2 GHz
Macro ISD	1732 m
Bandwidth	10 MHz
Macro Path Loss to UE	$128.1 + 37.6 \times log_{10}(d[Km])$
Pico Path Loss to UE	$140.1 + 37.6 \times log_{10}(d[Km])$
Antenna Gain	macro: 14 dBi; pico: 5 dBi
Transmit Power	macro: 46 dBm; pico: 30 dBm
Shadowing standard deviation	macro: 8 dBm; pico 10 dBm
Traffic model	FTP, file size: 14 Mbits

TABLE II: Simulation parameters

As a baseline for comparison, we consider the following default configuration of the SON functions: all deployed functions are on and with parameters that are commonly used in real networks (table III). In this scenario, we consider 3 consecutive phases where the operator sets a first set of priorities, then changes twice these priorities in the second and third phase as shown in figure 4. In the first iterations the algorithm is exploring the possible configurations, since it is learning from scratch with no prior information, hence generating low rewards. After a number of iterations, the algorithm

Parameters	Value
Load Low Threshold	0.6
Load High Threshold	0.8
CIO	0 dB to 16 dB with 2 dB steps
ABS	0 to 8 ABS per frame

TABLE III: Default Configuration Parameters

converged towards a configuration, generating stationary rewards. Furthermore, that the algorithm adapts rapidly with objective changes: in the second phase the algorithm passes through a brief phase of exploration, before converging back towards a new configuration. Same for the last phase, the algorithm explores briefly then converges to another configuration.

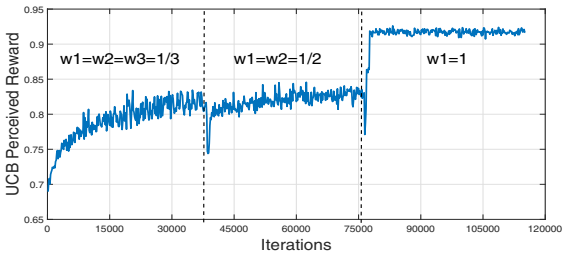

Fig. 4: Perceived Rewards

In figure 5 we compare the average reward generated by the configuration identified as optimal by the UCB1 with the default configuration on the one hand and the optimal configuration identified through an exhaustive search on the other hand. We notice that in the 3 cases the UCB1 succeeds in identifying the optimal configuration, whereas the default configuration is always generating sub-optimal rewards.

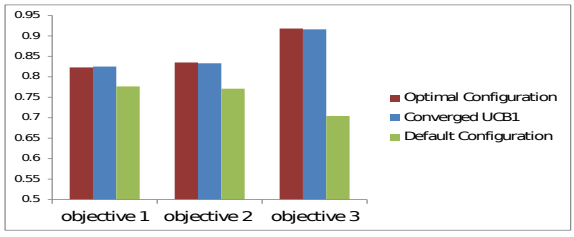

Fig. 5: Average Rewards

We can say that after a learning phase at the start, the C-PBSM converges towards SON configurations that maximize the perceived reward, performing better than a static default configuration. Also the C-PBSM adapts quickly to the operator objective changes.

VI. Conclusion

In this paper we have investigated the integration of MAB for SON management. The proposed C-PBSM configures the deployed SONs in the network based on UCB1, so that it maximizes a reward function reflecting the operator's objective. The novelty in this approach is that the system does not rely on external models to enhance its decision. It is able to learn online through real KPIs measurements and depicts the interaction between the different SONs, since in the learning process the SONs are deployed simultaneously, and various configuration combinations are tested. Simulation results show that the algorithm converges towards the optimal configuration within a reasonable time and adapts rapidly to priority changes. Furthermore, the UCB1 converges to the optimal configuration while minimizing the average regret i.e. minimizing the iterations where the C-PBSM is testing suboptimal configurations in the network.

The next step of our work is to evaluate the performances of the proposed C-PBSM on a real network where the scalability of the MAB algorithm will be the main challenge. We will investigate more evolved cell classification techniques to take into account the complexity of the real network topology.

References

[1] Cisco VNI mobile forecast highlights, 2015-2020

[2] S. Hämäläinen, H. Sanneck and C. Sartori, *LTE self-organising networks (SON): network management automation for operational efficiency,* John Wiley & Sons, 2012.

[3] SEMAFOUR project web page http://fp7-semafour.eu/.

[4] R.S. Sutton and A.G. Barto, *Reinforcement learning: An introduction, Vol. 1, no. 1.* Cambridge: MIT press, 1998.

[5] P. Auer, N. Cesa-Bianchi and P. Fischer, "Finite-time analysis of the multiarmed bandit problem," *Machine learning, 47*(2-3), pp. 235-256, 2002

[6] S. Lohmüller, L.C. Schmelz and S. Hahn, "Adaptive SON management using KPI measurements," *IEEE/IFIP Network Operations and Management Symposium (NOMS)*, 2016.

[7] S. Bubeck and N. Cesa-Bianchi, "Regret analysis of stochastic and non-stochastic multi-armed bandit problems," *Foundations and Trends in Machine Learning,* vol. 5, pp. 1-122, 2012.

[8] A. Feki and V. Capdevielle, "Autonomous resource allocation for dense lte networks: A multi armed bandit formulation," *IEEE Personal Indoor and Mobile Radio Communications (PIMRC)*, 2011.

[9] P. Coucheney, K. Khawam and J. Cohen, "Multi-armed bandit for distributed inter-cell interference coordination," *IEEE International Conference on Communications (ICC)*, 2015.

[10] M. Lelarge, A. Proutiere and M.S. Talebi, "Spectrum bandit optimization," *IEEE Information Theory Workshop (ITW)*, 2013.

[11] S. Hahn t al., "Classification of Cells Based on Mobile Network Context Information for the Management of SON Systems," *IEEE Vehicular Technology Conference (VTC Spring)*, 2015.

[12] 3GPP TR 36.814, "Technical Specification Group Radio Access Network; Evolved Universal Terrestrial Radio Access (E-UTRA); Further advancements for E-UTRA physical layer aspects (Release 9)", March 2010.