On Design Principles for Self-Organizing Network Functions

Zwi Altman[o], Mehdi Amirijoo*, Fredrik Gunnarsson*, Hendrik Hoffmann[Δ], István Z. Kovács[+], Daniela Laselva[+], Bart Sas[x], Kathleen Spaey[x], Abdoulaye Tall[o], Hans van den Berg[Φ], Kristina Zetterberg*

*Ericsson Research Linköping, Sweden
[x]iMinds / University of Antwerp, Antwerp, Belgium
[o]Orange Labs Issy-les-Moulineaux, France
[+]Nokia Aalborg, Denmark
[Φ]TNO, Delft, Netherlands
[Δ]TU Braunschweig Braunschweig, Germany

***Abstract*— With an increasing number of SON functions deployed in cellular radio networks, conflicts between the actions proposed by independently-designed and distributed SON functions may arise. The process of minimizing the occurrence, and the consequences, of such conflicts is referred to as SON coordination. SON coordination can be achieved either by a separate entity, a SON coordinator, controlling the actions of each SON function during operation, or in the design of the SON functions as such, through a conflict free SON design. In both cases, the SON functions should be designed in a way compliant with the coordination method used. This paper proposes a number of SON design principles to apply in order to achieve this. In the case of an operational SON coordinator, SON functions should be able to deal with the possible actions that are taken by the SON coordination function in order not to worsen the problems experienced in the network. In the case of conflict free SON design the aim is at removing potential conflicts already in the design phase. The ambition of this paper is not to provide a complete set of design guidelines that address all aspects. Instead, the target is to open up the discussion on the distributed SON design principles in the community.**

Keywords—SON interaction; SON conflict; SON coordination

I. INTRODUCTION

The trend is that cellular radio access networks are increasing in complexity, with a growing number of radio access technologies in concurrent use. Operators presently add the LTE and WLAN systems on top of large-scale GSM/GPRS and UMTS/HSPA deployments. Therefore, mechanisms that facilitate operations such as planning, deployment, optimization, and maintenance play an essential role to reduce operational effort and cost for the operators.

First steps towards a self-management system have already been taken with the standardized Self-Organizing Networks (SON) solutions in 3GPP, as well as with dedicated solutions provided by network equipment and software vendors. The first phase of stand-alone SON use cases and possible solutions that were initiated in Next Generation Mobile Networks (NGMN) and pursued in 3GPP ([1-3]) focused on self-configuration (such as dynamic plug-and-play configuration, automatic neighbor relationships) and self-optimization mechanisms (such as mobility robustness optimization, load balancing, interference management, coverage and capacity optimization, energy savings). For general SON discussions we refer to [4-7].

With the increasing number of SON functions deployed in all network nodes operating at the same time, there can be issues with conflicts between actions proposed by different distributed SON functions. These conflicts could for example occur when two SON functions want to change the same parameter in different direction, or when one SON function influences the target of another SON function negatively. Whenever there are risks that conflicts arise, then such conflicts need to be handled in some manner, or at least the negative implications from the conflicts need to be under control. For example, the EU FP7 project SEMAFOUR on self-management for unified heterogeneous radio access networks [13,17] investigates means to ensure such proper interaction between different SON functions.

One approach to dealing with (potential) conflicts in a self-organizing network is to use a SON coordinator, a separate entity controlling the actions of the SON functions during operation [8-14]. Another approach is to aim at a conflict free design of the SON functions as such [15,16]. Obviously, the SON functions need to be compliant with the considered approach to avoid or resolve conflicts. In this paper, a number of SON design principles are proposed to achieve this.

The motivation of this paper is not to provide a complete set of design guidelines that address all aspects but rather to open up the discussion on the distributed SON design principles in the community.

The paper is organized as follows. Section II discusses SON coordination in more detail and describes some proposed strategies from literature. When SON coordination is considered via a centralized SON coordinator entity, distributed SON functions will be impacted, and correspondingly Section III proposes some design principles for distributed SON functions under centralized SON coordinator actions. Section IV discusses a different approach where the design principles instead aim at distributed SON functions that can operate in a conflict free manner without a centralized coordinator. Finally, Section V provides some conclusive remarks and a summary.

II. SON COORDINATION

The process of minimizing the occurrence of SON conflicts and the consequences of SON conflicts is referred to as SON coordination. SON coordination can be achieved by i) an external entity, a *SON coordinator*, controlling the actions of the SON functions during operation, *operational SON*

© 2014 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

DOI: 10.1109/ISWCS.2014.6933396

coordination, or ii) in the design of the SON functions as such, through a *conflict free SON design*. One SON coordination method must not exclude the other, but the two methods can be seen to complement each other.

In the SEMAFOUR project, the *operational SON coordination* function (SONCO) is introduced for SON coordination during operation [18]. The SONCO considers a SON function to be a 'black box', where algorithm input and output are known, but the algorithm itself is unknown. The tasks of the SONCO are, among others, to detect conflicts between simultaneously running SON functions and resolve/prevent the conflicts during run-time. The SONCO can also undo actions it recently enforced. This type of SON coordination sets requirements on the design of the SON functions in terms of stability to the actions of the coordinator entity, which is further discussed in Section III.

Another approach for SON coordination is to design the SON functions so that they are conflict free right from the beginning. The coordination mechanism proposed in [16] fits in this kind of approach and provides a mathematical tool for conflict-free SON design. We recall below the framework proposed in [16] and then discuss the insights it gives for studying stability of simultaneously operating SON functions. We also discuss how the proposed coordination mechanism can be used for applying the design principles (see Section IV) to allow conflict-free operation of the SON functions.

According to [16], a SON function is a control loop which adapts a parameter x according to a function of Key Performance Indicator(s) (KPI), denoted here as $f(x)$. For example, f could be the gradient of a function we wish to maximize or the difference between a KPI and its target value. This control loop can be represented as an ordinary differential equation (ODE) in the following form

$$\dot{x} = f(x)$$

In practice, the discretized version of the ODE is used for the SON function, e.g. $x_{k+1} = x_k + \epsilon f(x_k)$ with ϵ being a small number. Bringing together I SON functions gives us a set of differential equations written as

$$\dot{\boldsymbol{x}} = F(\boldsymbol{x}) \quad (1)$$

where $\boldsymbol{x}$ is the vector $\boldsymbol{x} = [x_1, \ldots, x_I]^T$ of all the parameters, and F the vector $F = [f_1, \ldots, f_I]^T$ of updates. Using Hartman-Grobman theorem [20], eq. (1) can be linearized to get a system of linear ODEs

$$\dot{\boldsymbol{x}} = A\boldsymbol{x} \quad (2)$$

The stability of the system in eq. (2) can be simply characterized by the eigenvalues of the matrix A. If their real parts are all strictly negative ($A^T + A$ is negative definite) then the system is stable. The coordination mechanism proposed in [16] consists in applying a linear transformation to the right hand side of eq. (2) in order to get a stable system. The new system would be written as $\dot{\boldsymbol{x}} = CA\boldsymbol{x}$, where C is chosen so that $(CA)^T + CA$ is negative definite.

Each line of the matrix C, C_i, modifies the corresponding SON function taking into account the KPI functions (or the updates' vector $F(x)$) of the other SONs, $\dot{x}_i = C_i F(\boldsymbol{x})$. It is noted that if $C_{i,j} = 0$, then the parameter i will not be influenced by the update j. The ideal case would be to have a diagonal C, so that every SON function uses only its own update, yet the system remains stable. However, such a coordination matrix may not exist, e.g. when the SON functions are highly interacting with each other. We can also have a block-diagonal matrix C meaning that we regroup the SON functions so that there is interaction inside a group but not between groups. As in the diagonal case, the existence of such a matrix depends on the specific case studied.

While the focus in [16] is on designing appropriate stabilizing feedback matrix C, we will in Section IV address design principles that will have an impact on the properties of matrix *A* itself already in the design phase.

III. Design Principles for SON with SON Coordinator

When in a network a SON coordination function is deployed, the actions of the SON functions are being monitored and altered in order to avoid and mitigate conflicts in the network [18]. When the SON coordination function detects conflicts there are a number of possibilities to resolve these conflicts:

- Enabling/disabling/suspending SON functions
- Stopping/suspending/modifying SON actions
- Triggering modification of configuration parameters

It is important that SON functions are designed such that they are able to deal with the possible actions that are taken by the SON coordination function in order not to worsen the problems in the network.

A. Enabling/Disabling/Suspending SON Functions

The most drastic approach that can be taken by the SON coordinator in order to resolve conflicts is to disable certain SON functions entirely. This will prevent a selected SON function from making changes to network parameters such that other, more important, SON functions can change them without interference from the conflicting SON function.

The disabling of a SON function is however not permanent: After a certain amount of time, for instance after a situation of high load is over, the disabled SON function might be enabled again. In this case it is important that the SON function can resume its normal operation as soon as possible again. This might be difficult as during the time that the SON function was disabled both the network situation as the network parameters that are controlled by the SON function might have changed. If the SON function would just be started as if it was the first time that it is enabled or using its state as it was just before it was disabled it would take a relatively large amount of time before the algorithm would be able to operate optimally again.

Design Principle 1.1: Updating Internal State

While they are disabled by the SON coordinator SON functions should keep updating their internal state based on changes in the network such that once they are enabled again they are able to swiftly operate optimally again.

Note that with updating the internal state, the updating of the internal view of the network situation is meant, and not the internal updating of the network parameters that are controlled by the SON function.

As an example take an Mobility Robustness Optimization (MRO) SON function. This function will monitor certain KPIs like the Call Drop Ratio (CDR) and Ping-pong Handover Ratio (PPHOR) and adapt the handover parameters like *Hysteresis* and *Time-to-trigger* (TTT) according to a certain policy. When the SON function is disabled it should continue to monitor the CDR and PPHOR such that when it is enabled again it knows what values it can currently expect for these KPIs such that it will not try to improve these KPIs relative to some earlier state and/or that it does not have to measure these values for a while in order to know the current situation.

B. Stopping/Suspending/Modifying SON Actions

A less drastic action that can be taken by the SON coordinator is to prevent or modify network parameter changes made by the SON function. Changing a network parameter modification can for instance be done when two SON functions try to modify a certain network parameter at more or less the same time by taking an average value or changing the magnitude.

As other SON functions that are active in the network might also change the network parameters that are controlled by a SON function certain network parameters can also change without a request from the SON function. SON functions must thus be aware that the network parameter changes they propose are not necessarily carried out as requested and that network parameters can be changed without request. Therefore they should be designed such that they are able to deal with this fact.

Design Principle 1.2: Monitoring Output

SON functions should monitor the values of the controlled network parameters and take the current values into account when evaluating the system performance and taking decisions.

The SON coordination function that is being designed in the SEMAFOUR project also requires the SON functions being able to undo previous actions up to a certain extent [18]. The MRO SON function should for instance check the current handover (HO) parameters before it takes a decision to change them. If another SON function changed the HO parameters to a different value, the MRO function might wrongly conclude that a previous change was insufficient to change the HO performance and change the HO parameters to even more extreme values. Instead, the MRO SON should change the values of the HO parameters relative to their current values in the direction that will resolve the current performance issue.

C. Modifying Configuration Parameters

When repeated conflicts occur in the network there might be a configuration parameter conflict. The configuration parameters determine the policy that is being pursued by the different SON functions. In the SEMAFOUR architecture these configuration parameters are for example set by the Policy-Based SON Management (PBSM) function [18]. This function translates a high-level goal like improving coverage into the different parameter configurations of the SON functions. When the SON coordination function detects such a problem the PBSM or some other entity, for instance a human operator, is informed which will then adjust the parameters of the SON functions in order to harmonise the goals that are pursued by the SON functions. This means that the SON function cannot assume that its configuration parameters will be fixed for a very long time.

Design Principle 1.3: Adaptability of Goals

SON functions should be designed such that they are able to deal with configuration parameter changes and can quickly operate optimally again after a configuration parameter change.

IV. DESIGN PRINCIPLES FOR CONFLICT-FREE SON

In order to analyse potential SON function conflicts, we distinguish between

- inter-SON function aspects, i.e. how different SON functions mutually interact
- intra-SON function aspects, i.e. how instances of a specific SON function implemented in different nodes interact

To address differently such aspects, some specific system models are needed, which are presented in the next subsection. Moreover, inter- and intra-SON functions are further analysed in the following subsections.

A. System Model

The considered system model is similar to the one used in [16] and presented in Section II. However, in order to separate different SON functions from different instances of the same SON functions, we consider N different SON functions. Each SON function $n;\ n=1{:}N$ has in total m_n instances, which means that in total there are $I = \sum_n m_n$ SON functions. The set of SON function instances of SON function n is denoted I_n. For clarity, the SON functions are sorted by function type, so that

$$I_n = \left\{ i;\ \sum_{k=1}^{n-1} m_k < i \leq \sum_{k=1}^{n} m_k \right\}$$

As an example, consider only distributed SON functions with N functions associated to each of the M cells in a network, in total $I=NM$, with $m_n = M,\ \forall n,$ and

$$I_n = \{i;\ M(n\text{-}1) < i \leq Mn\},$$

where SON function i is tuning the parameter x_i by $\dot{x}_i = f_i(\boldsymbol{x})$. In addition, the SON function i is considering a measurement y_i, possibly with the objective to meet a target y_i^t. By introducing the corresponding vectors $\boldsymbol{y} = [y_1, \dots, y_I]^T$ and $\boldsymbol{y^t} = [y_1^t, \dots, y_I^t]^T$, the extended set of differential equations can be written as

$$\dot{\boldsymbol{x}} = F(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{y}^t) \quad (3)$$

As in Section II, this can be linearized in a similar manner, for example as

$$\dot{\boldsymbol{x}} = A\boldsymbol{x} + \boldsymbol{B}(\boldsymbol{y}^t - \boldsymbol{y}) \quad (4)$$

Note that a sorting of the SON functions gives matrix block structures, based on the *N* different SON functions:

$$A = \begin{bmatrix} A_{11} & \cdots & A_{1N} \\ \vdots & \ddots & \vdots \\ A_{N1} & \cdots & A_{NN} \end{bmatrix}, B = \begin{bmatrix} B_{11} & \cdots & B_{1N} \\ \vdots & \ddots & \vdots \\ B_{N1} & \cdots & B_{NN} \end{bmatrix} \quad (5)$$

where block A_{nk} describes how the parameters of SON function *k* have an implication on parameters of SON function *n,* i.e. the inter-SON function aspects, diagonal blocks A_{kk} describe how different instances of the same SON function interact, i.e. the intra-SON function aspects, and blocks in *B* how different measurements impacts different SON functions.

In the discussion above, each SON function instance *i* is associated to one parameter x_i which also is seen as the state of that SON function instance. Note that this can be generalized to representations with multiple states per SON function instance, but the slightly simpler model with only one state per SON function instance will be used in the sequel without loss of generality.

B. *Inter-SON Function Aspects*

One challenge with SON conflicts is the many potential inter-SON function conflicts. As seen in the *A* matrix in eq. (5), the A_{nk} blocks represent inter-SON function interactions. Furthermore, measured quantities can be used by multiple SON functions and thereby create an inter-relation and potential issue. If it is possible to impact the system already in the design phase, then it can be possible to affect such interactions and inter-relations. In the following, we describe a set of design principles that the system itself has attractive properties such as mutually independent inter-SON functions. This implies that the overall system becomes easier to analyse and design.

The first principle concerns inter-SON functions affecting the same parameters, which cause interactions. Therefore, the first design principle is to forbid such interactions.

Design Principle 2.1: Separation of Control

Each SON function i *of type* n, $i \in I_n$ *controls and affects only parameters* x_j *associated to the same SON function type,* $j \in I_n$ *and any SON function* l *of a different type* k *cannot control that parameter* x_i.

In the current framework, the state variable of and the parameter controlled by a SON function instance are the same, which means that this design principle implies independence between inter-SON function states.

As an example, consider the SON functions MRO (*n=1*) and Mobility Load Balancing (MLB) (*n=2*). It is a classical potential conflict, since both these SON functions can be seen as controlling the handover triggering point, or handover offset *Δ*. Separation of control can be implemented by separating the mobiles into two groups:

1. Mobiles subject to regular mobility
2. Mobiles that will be enforce to change cell due to load balancing reasons

Mobiles in each of the groups are configured with separate parameters and are handled by the two different SON functions. MRO will tune the parameter Δ_1 and configure mobiles in group 1 with the updated parameter. MLB on the other hand will tune the parameter Δ_2 and configure mobiles in group 2 with the updated parameter.

Note that these groups can be split in time (sometimes MLB is active, and sometimes not), over users (users subject to MLB in one group, and the remainder in the other group) or a combination.

Separation of control means that the off diagonal blocks A_{nk} of the A matrix are all zero, resulting in independence in the A matrix between states associated to different SON functions. However, there may still be remaining interactions via the measurements if two SON functions use the same measurement, or at least correlated measurements. The Separation of Concern design principle avoids such interactions by restricting the use of the same measurements:

Design Principle 2.2: Separation of Concern

Each SON function i *of type* n $i \in I_n$ *uses only its specific measurement* y_i, *and any other SON function j of type k* $j \in I_k$ *cannot use that measurement. Furthermore, the measurements are made independent of each other,* $E\{y_i, y_j\}=0$, $\forall i \in I_n, \forall j \in I_k, n \neq k$.

The design principle implies that there are no interactions between SON functions via the measurements. Together with separation of control, the states associated to different SON functions will be independent. The B_{nk} blocks of the *B* matrix in eq. (5) are zero, simplifying the analysis of the network considerably. Again, the principle is made more intuitive via an example. Consider the SON functions MRO (*n=1*) and MLB (*n=2*). Both these SON functions can take advantage of handover performance measurements such as handover failure and radio link failure statistics in the serving node, denoted *M.* Separation of concern can be implemented by separating statistics so that each SON function is based on statistics that is independent from statistics used by other SON functions. Typically, this can be done by separating the statistics per group:

- The statistics M_1 is gathered from mobiles in group 1, subject to regular mobility
- The statistics M_2 is gathered from mobiles that have been enforced to change cell due to load balancing

MRO will tune its parameter based on statistics M_1, while MLB will tune its parameter based on statistics M_2. As before, the statistics may not always be gathered for a particular group (MLB may not always be active).

Another example is the following. We consider two SON functions operating in parallel namely a load balancing (LB) algorithm and an outage compensation algorithm (OC). The LB SON tunes the Cell Individual Offset to balance the cell loads and the OC SON adjusts the transmit power of traffic channels to achieve a target outage. The outage is defined as the proportion of users with data rate below a pre-defined target. These two SON functions have been shown to be unstable when operating simultaneously [16, Section IV.B.2]. Using the coordination mechanism from [16], the two SON functions have been stabilized.

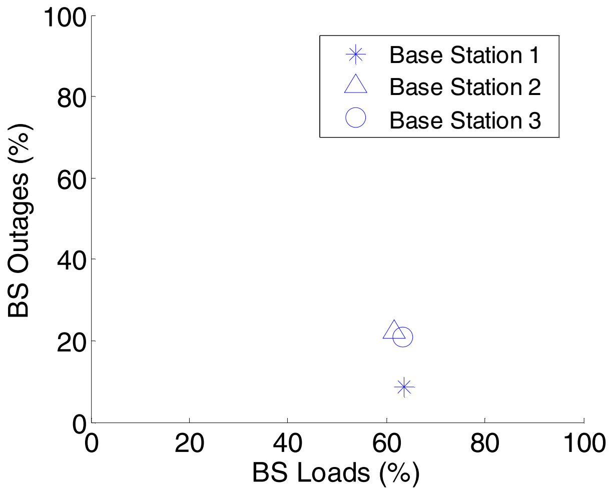

Fig 1: Operation point without separation of concern

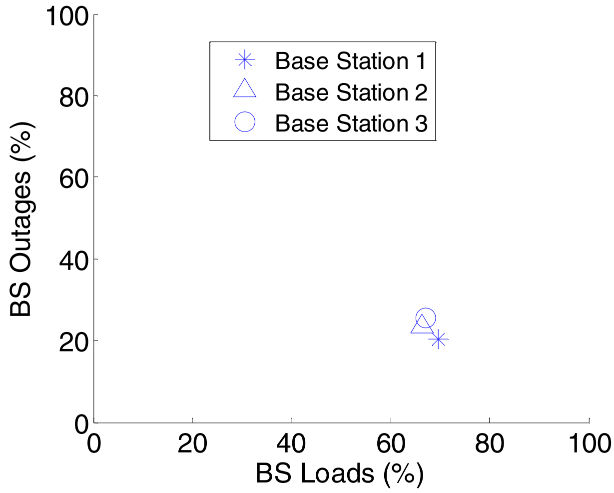

Fig 2: Operation point with separation of concern

However, the goals of the OC SONs are not fully satisfied as shown in Fig. 1. The target for all Base Stations (BSs) has been set to 18% and the final operation point of BS 1, although benefitting from lower outage, is different from the design target. Looking closer at the impact of the coordination mechanism, it has been noticed that to achieve stability, the OC SON mainly uses the LB SON KPI (load imbalance) in addition to its outage KPI so that its target is not achieved. Separation of concern can be applied to the OC SON by enforcing it to use only its own KPI. This is achieved by setting to zero all elements $C_{i,j}$ of the coordination matrix where i is the index of a OC SON, and j is the index of a LB SON. The resulting operation point in the outage – load plane gets closer to the target KPIs as shown in the Fig. 2.

On the other hand, there might be reasons to allow different SON functions to tune the same parameter and/or use at least correlated measurements. One such example is when the two SON functions operate on very different time scales, for example a slow centralized SON function and a much faster decentralized SON function. In such cases, the actions of the faster SON function can be seen as converged when evaluating the situation on longer time scales based on averaged measurements over long time windows.

Design Principle 2.3: Separation of Time Scales

Two SON functions operating at different time scales are separated by time if the actions by the faster function can be assumed to have converged well within the update period of the slower one, and that measurements considered by the slower are averaged enough so that there is negligible correlation between the faster variations of the measurements.

This separation principle implicitly means that the considered SON functions are not expected to require the same reaction time, or time granularity, to the changes to the network conditions, in terms of load, mobility, radio link failures, etc. Requirements from short to long term variations could be anticipated, where short changes are in the order of tens of seconds/few minutes up to changes in terms of several hours / days which characterize long variations. This implies that states associated to different SON functions are independent.

To illustrate an example, we consider three different SON functions expected to be employed in typical heterogeneous network deployments, which include macro, small-cell, and WLAN radio access layers. The exemplified SON functions are taken from the uses cases studied in the SEMAFOUR project [19] as follows:

1. Intra-LTE dynamic spectrum allocation (DSA): Re-allocates spectrum between LTE cells according to the estimated long-term cell load/traffic.
2. LTE-WLAN access network selection (ANS): Re-directs user sessions towards LTE or WLAN systems depending on the estimated medium-term cell load conditions.
3. Intra-LTE traffic steering and mobility load balancing: Controls handovers and traffic load based on the estimated short/medium-term cell load conditions.

The DSA SON function monitors the cell load conditions in a pre-defined cell cluster and re-allocates available spectrum (LTE carriers) between the cluster cells in order to maximize the overall system spectrum efficiency. This mechanism is beneficial when the offered traffic load is changing in a localized geographical area e.g., covered by 1-2 cells, and the temporal variations are expected to be rather slow e.g., between morning, midday, and afternoon periods.. The outcome of the

DSA SON decisions is available to all the involved RAN LTE cells.

The LTE-WLAN ANS is typically utilized in high traffic areas where the LTE capacity is expected to be exceeded only temporally, and there is a sufficient number of WLAN access points deployed. In essence the LTE-WLAN ANS SON function monitors cell load levels, either on LTE only or on both LTE and WLAN [19], and attempts to re-direct the UEs to the access technology which is less loaded. The LTE-WLAN ANS SON should be able to react relatively fast to changes in the offered traffic in the cell but is not required to follow traffic changes due to individual UE traffic patterns.

The TS/MLB SON functions are typically designed to react to faster changes in the cell load, due to UE mobility and changing traffic patterns.

It is worth to note here that for both LTE-WLAN ANS and TS/MLB SON the performance of the LTE layer is subject to the large time scale DSA SON actions.

All three SON functions exemplified above utilize the cell load as input key performance indicator (KPI) to the SON function. The target of *separation of time scales* can be achieved by appropriately selecting the time period over which the cell load is monitored/ estimated/ filtered to derive the input metric to the decision algorithms.

C. Intra-SON Function Aspects

With the block structure of the system, and if the design principles in Section IV.B are considered, then the only interactions that remains are those between different instances of the same SON function – they have become isolated. Such interactions depend on the deployed network as well as the SON function itself. However, with the massive reduction of interactions between SON functions from the design principles, it can now be tractable to analyse these SON functions in theory and/or via simulations.

V. CONCLUSIONS

In this paper, a set of design principles to construct independent SON functions which operate simultaneously in a conflict free fashion is presented, assuming a centralized SON coordinator entity, and a conflict-free SON design, respectively.

In the case of an operational SON coordinator, SON functions should be able to deal with the possible actions that are taken by the SON coordinator. This could mean that the SON functions should be designed such that they may need to update their internal state based on changes in the network even while disabled, take the current values of the controlled parameters into account when evaluating system performance and taking decisions, and such that they are able to deal with parameter changes and can quickly operate optimally again after a parameter change.

In the case of conflict free SON design, the proposed design principles separating the control parameters changed by the SON functions, the measurements used by the SON functions, and, in some cases, the time scales they operate on, will ensure that there are no undesired interactions between different SON functions.

The two SON coordination methods must not exclude the other, but the two methods can be seen to complement each other. A topic for further research is to evaluate the design principles introduced in this paper.

VI. ACKNOWLEDGEMENT

The research leading to these results has been carried out within the FP7 SEMAFOUR project and has received funding from the European Union Seventh Framework Programme (FP7/2007-2013) under grant agreement no 316384.

VII. REFERENCES

[1] 3GPP, “Evolved Universal Terrestrial Radio Access Network ({E-UTRAN}); Self-configuring and self-optimizing network ({SON}) use cases and solutions, TS 36.902.

[2] 3GPP, “Telecommunication Management; Self-Organizing Networks (SON) Policy Network Resource Model (NRM) Integration Reference Point (IRP); Requirements”, TS 32.521.'

[3] Next Generation Mobile Networks (NGMN), “Recommendation on SON and O&M Requirements”, 2008.

[4] C. Prehofer and C.Bettstetter, ”Self-organization in communication networks: principles and design paradigms”, *IEEE Commun. Mag.*, vol.~43, no.~7, pp. 78 -- 85, July 2005.

[5] J. Ramiro and K. Hamied, Self-Organizing Networks (SON): Self-Planning, Self-Optimization and Self-Healing for GSM, UMTS and LTE”, *John Wiley & Sons Ltd*, 2011.

[6] S. Hämäläinen, H. Sanneck, and C. Sartori, “LTE Self-Organising Networks (SON): Network Management Automation for Operational Efficiency”, *John Wiley & Sons Ltd*, 2011.

[7] Gunnarsson, F. “Self-Organization”, In Heterogeneous Cellular Networks Theory, Simulation and Deployment, Eds. Chu, X., Lopez-Perez, D., Yang, Y. and Gunnarsson, F., *Cambridge Academic Press*, 2013.

[8] T. Jansen, M. Amirijoo, U. Turke, L. Jorguseski, K. Zetterberg, R. Nascimento, L. Schmelz, J. Turk, and I. Balan, “Embedding multiple self-organisation functionalities in future radio access networks,” in *IEEE 69th Vehicular Technology Conference (VTC Spring 2009)*.

[9] L.-C. Schmelz, M. Amirijoo, A. Eisenblaetter, R. Litjens, M. Neuland, and J. Turk, “A coordination framework for self-organisation in LTE networks,” in *IFIP/IEEE International Symposium on Integrated Network Management (IM)* 2011.

[10] A. Galani, K. Tsagkaris, P. Demestichas, G. Nguengang, I. BenYahia, M. Stamatelatos, E. Kosmatos, A. Kaloxylos, and L. Ciavaglia, “Core functional and network empower mechanisms of an operator-driven, framework for unifying autonomic network and service management,” in *IEEE 17th International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, 2012.

[11] P. Vlacheas, E. Thomatos, K. Tsagkaris, and P. Demestichas, “Operator-governed SON coordination in downlink LTE networks,” in *Future Network Mobile Summit*, 2012.

[12] Z. Liu, P. Hong, K. Xue, and M. Peng, “Conflict avoidance between mobility robustness optimization and mobility load balancing,” in *IEEE Global Telecommunications Conference (GLOBECOM 2010)*.

[13] Eisenblatter, A.; Gonzalez Rodriguez, B.; Gunnarsson, F.; Kurner, T.; Litjens, R.; Sas, B.; Sayrac, B.; Schmelz, L.C.; Willcock, C., "Integrated self-management for future radio access networks: Vision and key challenges," *Future Network and Mobile Summit, 2013.*

[14] Bandh, T.; Schmelz, L.C., "Impact-time concept for SON-Function coordination," *International Workshop on SON (IWSON) 2012*, Paris, France.

[15] Combes, R.; Altman, Z.; Altman, E., "Coordination of autonomic functionalities in communications networks," *Modeling & Optimization in Mobile, Ad Hoc & Wireless Networks (WiOpt), 2013 11th International Symposium on* , May 2013.

[16] Tall, A., Combes, R, Altman, Z. and Altman, E. “Distributed coordination of self-organizing mechanisms in communication networks”, to appear in *IEEE Transactions on Control of Network Systems 2014.* arXiv:1309.5067.

[17] SEMAFOUR, http://fp7-semafour.eu,.

[18] EU FP7 project SEMAFOUR (Self-management for unified heterogeneous radio access networks), Deliverable D5.1: ‘Integrated SON Management Requirements and Basic Concepts’, December 2013, available at http://fp7-semafour.eu.

[19] EU FP7 project SEMAFOUR (Self-management for unified heterogeneous radio access networks), Deliverable D4.1: ‘SON functions for multi-layer LTE and multi-RAT networks (first results)’, November 2013, available at http://fp7-semafour.eu.

[20] P. Hartman, “A lemma in the theory of structural stability of differentialequations,”Proceeding of the American Mathematical Society, vol. 11,pp. 610–620, 1960.