Deliverable D3.1

Identification of suitable classes of methods for parameter optimization

Grant Agreement	257513	
Date of Annex I	25-07-2011	
Dissemination Level	Public	
Nature	Report	
Work package	WP3 – Network Empowerment	
Due delivery date	01 September 2011	
Actual delivery date	14 September 2011	
Lead beneficiary	ALUD	Markus.Gruber@alcatel-lucent.com

SEVENTH FRAMEWORK PROGRAMME

FUTURE NETWORKS

Authors	ALUD - Harald Eckhardt, Markus Gruber (Editor), Siegfried Klein ALBLI - Lester Ho NEC - Johannes Lessmann, Zarrar Yousaf NKUA - Eleni Patouni, Apostolis Kousaridas, Konstantinos Chatzikokolakis, Damianos Kypriadis, Nancy Alonistioti UNIS - Velmurugan Ayyadurai, Frederic Francois, Stylianos Georgoulas UPRC - Kostas Tsagkaris, Panagiotis Demestichas, Vera Stavroulaki, George Athanasiou, Panagiotis Vlacheas, Yiouli Kritikou, Nikos Koutsouris, Aimilia Bantouna, Dimitris Karvounas, Evagelia Tzifa, Assimina Sarli, Evangelos Thomatos, Marios Logothetis, Andreas Georgakopoulos, Louiza Papadopoulou VTT - Petteri Mannersalo, Helena Rivas

Executive summary

This deliverable focuses on network optimization tasks and the selection of methods suitable to achieve these tasks. The corresponding question we would like to answer is how to choose changes in network parameters that give us an improvement for a given set of key performance indicators. In principle there are two types of problems: those where a model is available giving us an estimate what a parameter change would approximately result in and those where there is no such model and where the best parameter changes can only be guessed.

The first chapter focuses on two classes of methods where no model is available, i.e. changes in parameters will result in changes of the key performance indicators that are hardly or not at all predictable. In this context the role of randomness in two completely disjunctive classes of algorithms, namely evolutionary algorithms representing heuristic methods on the one hand and a gradient descent approach representing a solution designed for convex optimization problems on the other hand, is analysed. We show that evolutionary algorithms can be successfully applied for non-convex problems where the search space is large; a major advantage of this class of algorithms is its flexibility – a telecom researcher can choose from a wide variety of different flavours of the method class (e.g. genetic algorithms, genetic programming). In this deliverable we analyse evolutionary algorithms as a placeholder for methods with random elements. For the specific example of multihop relay-assisted cellular networks, we have clearly shown that evolutionary algorithms outperform methods like simulated annealing as well as mixed integer linear programming. The major advantage of evolutionary algorithms is the fact that a solution can be found relatively quickly and that the state space is explored comprehensively so that ending up in a local optimum becomes less likely. The next steps of our work in this field will be the investigation of the suitability for optimizations within already running networks where trying out unfavourable parameter configurations may have a severe impact.

By contrast, gradient descent approaches are typically applied to convex problems. However, wireless access networks, due to their statistical and non-deterministic properties, reveal problems that are only nearly, but not strictly convex. In this case, we propose to adapt the gradient descent approach by injecting a certain level of noise, similar to the evolutionary algorithms, such that the solution does not get stuck in local optima.

The following chapters then focus more on use cases rather than on the methods itself. The chapter on Governance (use case 6 in deliverable D4.1) extends the method-oriented view by an integrated view on the network. Instead of solving problems for a wireless access network alone, it reveals an additional dimension of complexity by looking at both the wireless and the core domains of the network in an end-to-end way. We show that this complexity can successfully be addressed by using policies that control and coordinate the performance of the entire network, not only individual network domains. Also, there are concrete instances where evolutionary algorithms are applied. This chapter addresses a unification/federation aspect and thus acts as a link to the work on UMF (Unified Management Framework) (deliverable D2.1). With respect to state-of-the-art work, this chapter focuses on a joint end-to-end management of networks composed by multi-vendor/multi-technology segments whose governance is policy-based. Additional effort will have to be spent to answer the question what specific methods are best-suited for this particular purpose.

The next chapter covers a large variety of different facets of load balancing (mainly referring to use case 3 in deliverable D4.1). Load balancing techniques can namely be used in the access domain of the network, in the backhaul/core domain or at the interface of both. A load balancing framework was developed integrating all these aspects in a global view. The analysed aspects also include load balancing between different radio access technologies, interference coordination, and transmission power adaptation as well as access point (de-) activation. The presented work exceeds state-of-the-art work by, e.g., taking into account aspects like energy efficiency or by combining existing load balancing approaches for a better performance. This work bears significant potential, as load balancing is viewed from a multitude of perspectives. The final integration, also in terms of best-suited methods, of these related pieces into one mosaic will be the work of the second project year.

In conclusion, this deliverable elaborates on a class of methods that is particularly eligible for non-deterministic situations (as they appear in wireless access networks). Furthermore, solutions for two particular fields are proposed which both require an integrated view on the network: first of all governance spanning over several network domains (wireless and wireline) and secondly load balancing spanning over several network domains (access, backhaul, and core).

Table of Content

Foreword

The Network Empowerment work package (WP3) will provide the most efficient methods to deliver a toolbox of solutions covering selected operator scenarios. It covers all the work needed to study, design and to evaluate various algorithms with self-x and cognitive capabilities (hereafter termed methods) together with the requirements for their embodiment into network functions to assure trustworthy federation of heterogeneous networks. The work in this work package is based on use cases' problems and should provide the best-suited methods to solve these problems. WP2 as the integration part of UniverSelf will eventually embody the algorithms designed in WP3 into the network through the design of enabling mechanisms and facilities.

The main focus of this deliverable (D3.1) is the optimization of system parameters, whereas deliverables D3.2/D3.3 rather focus on actions that are taken on certain observations (which not only includes learning, but also control theoretical aspects), and deliverable D3.4 focuses on the role of cooperation strategies between control loops and network entities.

This deliverable summarizes the results of Task 3.2 of the work package on Network Empowerment achieved so far. The main purpose of Task 3.2 is to find the best-suited method for a given parameter optimization problem. More specifically, the goals of Task 3.2 are to

- Identify what method is best-suited to optimize parameters of a given self-x (self-configuration, self-optimisation and self-management) problem
- Study the effectiveness of self-optimization for a given problem at runtime
- Adapt, tailor and if necessary extend the identified optimization methods to the specific needs of the given problems, evaluate the effectiveness of the methods, and iteratively modify and refine the techniques to suit specific network scenarios

The word "problem" hereby refers to use case problems as defined in Section 7.2.2 of deliverable D4.1. When describing the individual chapters of this deliverable in the introduction we will briefly sketch the use case problems – a more detailed table with the mapping between use case problems and task forces (in which the work package on Network Empowerment is essentially organized) can be found in the section mentioned above.

1 Introduction

The first chapter of this deliverable, which is on random elements, focuses on methods and their refinements without limiting itself to a very specific use case. This appears to be paradox as the work in WP3 is based on use cases' problems and the very spirit of the Network Empowerment work package is to find the right method for a given problem with an open mind. Evolutionary algorithms, however, have become such a standard tool for non-convex problems that its suitability for this class of problems may depend more on the specificities of the algorithm than on the choice of the method. In other words, wrongly using this type of algorithm may result in solutions that are avoidably bad. This is why we have taken advantage of the fact that we have a critical mass of experts on this class of algorithms in the project to dive considerably deeply into this topic of wide interest. We have also demonstrated that for the problem of multi-hop relay-assisted cellular networks evolutionary algorithms outperform simulated annealing as well as mixed integer linear programming. The concrete use case problems according to Section 7.2.2 of deliverable D4.1 that are addressed by the task force on random elements are

- "Design of distinct SON functionalities in network nodes to efficiently self-configure and self-optimize network resources" (UC4-1)
- "Design of different SON functionalities operating simultaneously to achieve one or several performance objectives" (UC4-2)
- "Govern radio access networks by means of high level policies triggering coordinated SON functionalities" (UC4-3)
- "To invoke […] selected RANs and request for an offer in terms of the quality which the RAN can provide" (UC6-4)

The chapter on Governance and autonomic management of cross-domain segments not only addresses a critical inter network domain aspect, in this case a bridge between the wireless and wireline world, but also demonstrates the link to the work package on the Unified Management Framework which should be able to coordinate optimizations with potentially competing goals across different network domains[1]. It directly maps to the use case "Operator-governed, End-to-end Autonomic joint Network and Service Management" defined in D4.1 and maps to the use case problems

- "To invoke […] selected RANs and request for an offer in terms of the quality which the RAN can provide" (UC6-4)
- "Invocation of backhaul/core segment" (UC6-5)

The chapter on Load balancing mainly relates to the use case "Dynamic Virtualization and Migration of Contents and Servers" defined in D4.1. This work again benefits from the fact that the topic is approached from different perspectives out of a critical mass of partners working in this field. In particular, various aspects from wireless to backhaul/core domains are addressed as well as (in the future) a service perspective. The concrete use case problems addressed in the load balancing task force are

- "To develop strategies that would reduce the load (traffic, processing, signalling etc.) on the core network segments and data centres for efficient delivery of data/service/application to the mobile user" (UC3-1)
- "Develop algorithms that would leverage the virtualization techniques and cloud concepts for seamless migration of resources/services/functions context" (UC3-3)
- "Develop simulation models to understand the implication of decentralizing the resources/functions/services from the core and migrating them towards the access and backhaul segments and their impact on the network architecture and the corresponding network entities" (UC3-4)
- "Design of distinct SON functionalities in network nodes to efficiently self-configure and self-optimize network resources" (UC4-1)

[1] of a single network operator, as per UniverSelf scope.

- "Govern radio access networks by means of high level policies triggering coordinated SON functionalities" (UC4-3)
- "Invocation of backhaul/core segment" (UC6-5)
- "To ensure the designated QoS levels by means of corrective actions and by issuing alarms and warnings to the upper layers in case these corrective actions cannot remedy the problematic situation" (UC6-8)
- "Evaluation of the network governance tool, in terms of examining whether the generated policy rules and the applied configuration actions meet the initial business requirements" (UC7-2)
- "Decision making processed based on semantic models and inference engines must be supported for self healing purposes" (UC7-6)

Please note that service management related optimization, and in particular the last two problems related to use case 7, have not been addressed in this deliverable and are subject to future work in the project.

Generally, in the Network Empowerment work package we have discussed requirements towards the UMF. The outcome was the following:

- High-level policies (e.g. coverage or capacity) with absolute goals and/or relative weightings are given to the UMF by the operator
- The methods (or in other words, processes dealing with use case problems) will provide affected control parameters, metrics, and key performance indicators
- The UMF passes low-level policies on to the individual methods/processes and – where needed – provides a utility function
- Methods/processes also need the possibility to report failures to the UMF

The UMF and its relation to methods/processes are exemplarily addressed in the chapter on Governance and will be further elaborated in the second project year and future releases of the WP2 and WP3 deliverables. Likewise, further integration of the presented three chapters is planned for the second project year.

2 Methods with random elements

2.1 Introduction

Self-optimization in autonomous networks often involves solving multi-objective problems that are complex due to the interactive nature of the network nodes. For example, when optimizing the radio transmission parameters in a wireless network, making changes in one base station can have a detrimental impact on neighbouring base stations. On top of that, the optimization also takes place in highly dynamic conditions with varying user traffic, network topologies and physical environments. Because of this, deriving mathematical models that are able to predict the behaviour of these networks can be difficult to do, making heuristic optimization applied to simulation models or feedback from real-world networks a more practical approach.

In this chapter, the application of optimization techniques with random elements is presented, namely evolutionary algorithms (EAs) and a gradient descent with noise. The two techniques represent two basic approaches to network optimization: EAs are techniques that perform a search over a large area in the search space, while the gradient descent with noise performs the search of neighbouring states. These two approaches have different implications with regards to how they can be implemented in the network, which is discussed in the following sections. Firstly, the basic mechanics of these two techniques are presented, and applications on different network optimization problems are given, with comparisons in performance over other approaches in the state-of-the-art given. The application areas used are in radio network optimization (UC4) and routing for MPLS traffic engineering (UC5, UC6). At a later stage of the project the results will be integrated into the work on the Unified Management Framework (UMF, see also D2.1).

2.2 Techniques with random elements

Here, two different techniques are described: evolutionary algorithms (EA) and steepest descent with noise. The two techniques involve the search of the solution space for optima, using random elements in order to explore the search space, but they do so with different amounts of disruptiveness. Network optimization problems are often NP-hard, with a search space that is very large and hard to define, with many local optima. EA is a technique that is able to perform the search for a solution over a very wide area of the search space to the risk of converging to local optima, but at the same time typically converging to a solution quickly. Generally, however, EAs and the other associated techniques can randomly jump between different areas of the search space. While this helps avoid being stuck at low optima, it can be disruptive if applied to real-time optimization.

Another technique that is less disruptive is one that has a more gradual traversal of the search space. Here, a steepest descent technique with noise is used for that purpose. The approach is intended to be less disruptive when applied to certain real-time network optimization scenarios.

The two different techniques given here therefore give two different approaches to network optimization, and give insight into how they can be practically applied to real networks.

2.2.1 Evolutionary algorithms

An evolutionary algorithm is an optimization algorithm that uses concepts that are based on biological evolution. It is a population-based approach that uses genetic operations such as mutation and recombination on candidate solutions over multiple generations in order to produce a solution with a high fitness.

The basic steps involved in setting up a problem for EA are the following:

- The specification of a fitness function. The fitness function is a form of objective function that is used to calculate the fitness (i.e. optimality) of a solution. In the context of network optimization, examples of the fitness can be network performance such as throughput, energy efficiency or load. For multi-objective optimization, the fitness can be formulated as an aggregate of different metrics.
- Creating a representation of solutions in a form that can be manipulated by the genetic operators. These are called chromosomes, and can be in many different forms, depending on the problem being addressed. For example, a binary string chromosome representing a numerical value or a network resource allocation, or a decision tree representing a decision making program.
- The specification of the EA parameters to be used. This includes the parent selection procedure, probabilities of mutation and recombination, population size, and the termination condition. The

termination condition can be when a certain level of fitness is achieved or after a set number of generations.

The process of evolving a solution involves finding the right combination or sequence of chromosomes, driven by the fitness function. This process is illustrated in Figure 1, where the steps involved are as follows:

1. Initialize the population with random solutions.
2. Evaluate the performance of each solution in the population using a fitness function. This can be done by testing the solution, e.g. in a network simulation model.
3. Populate the next generation with offspring by applying genetic operators (mutation, recombination) on individuals with high fitness.
4. Repeat steps 2-3 until stop conditions are met.

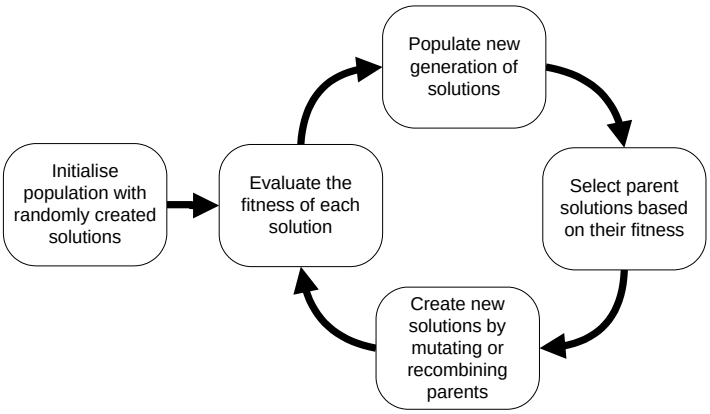


Figure 1: Sequence of EA process

Figure 2 illustrates roughly the mechanics of EAs. The red circles represent the location of candidate solutions in the fitness landscape. The fitness landscape illustrated here has several local optima. In the early phase, the population-based search basically starts with a random search. As the EA process iterates, the characteristics of solutions with high fitness is retained and those with low fitness is discarded. Over more iterations, the evolution process starts to converge onto peaks in the fitness landscape.

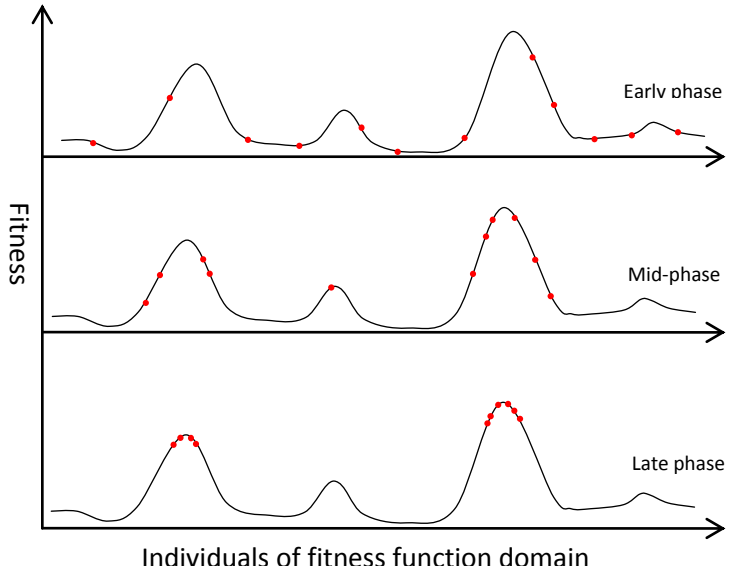


Figure 2: Typical progress of EA illustrated in terms of population distribution

There are several different forms of EAs, which differ in terms of the specifics of implementation and their application to specific problems. Here, two techniques are applied. Genetic algorithm (GA) is arguably the best-

known, where the chromosome is represented as a string, and is used here to optimize network configurations [1]. Genetic programming (GP) is a technique used to evolve computer programs that perform a certain task [2], and has chromosomes that represent computer routines such as parse trees or equations. It is used here to generate algorithms used in the network nodes.

EAs have several advantages. As the search starts out over multiple locations in the fitness landscape, it is able to locate several different optima and is thus less prone to getting stuck at a low optimum point. In many cases, the fitness achieved increases rapidly at the beginning before flattening out later on [3], so a good solution can also be found quickly. The specification of the fitness function can be done using high-level network performance metrics, and can be adapted easily to include multiple objectives.

On the other hand, when applied to network optimization, a large proportion of the candidate solutions produced during the evolution are highly sub-optimal. This means that it would not be possible to evaluate them in a real operating network as this would be highly disruptive. A model of the network is required instead. This currently limits the application of the technique to perform optimization in an offline manner.

2.2.2 Steepest descent with noise

In contrast to the optimization approaches described above, the stochastic behaviour is caused by the system response. The optimization algorithm can be with or without stochastic exploration of the state space. But the system response is composed of two components: the response related to the exploration and a stochastic component. The source of the stochastic component is caused by the fact that the system is not a model but a real world system, which responds slightly differently even if the same control parameters are applied, e.g. in wireless networks this is caused by different user distributions, services requested and radio propagation fluctuations. From the data seen so far, the problem is convex on a large scale, but due to a noise component, the convexity gets lost when looking at the data on a small scale. Steepest descent, which is a method to find an optimum numerically and iteratively, is well suited for convex problems, i.e. for searching the optimum on the large scale. But the small scale noise disturbs a straight-forward steepest descent algorithm e.g. it tends to get stuck in local optima. So, a problem-specific noise-removal stage is necessary, which is partly heuristic as the noise, which is not an added white noise, cannot be separated by simple means yet.

The approach is to use problem specific filters to suppress the noise. The optimization algorithm itself can be reduced for this study to some well known method like steepest ascent/descent. The strength of the approach is to use a real system instead of a model for testing; the weakness is that a solid statistics is needed for each exploration step.

2.3 Optimisation of parameters using techniques with random elements

2.3.1 Evolving coverage optimization algorithms in femtocells using Genetic Programming (GP)

Femtocells are low-power, low-cost cellular base stations that use a wireline connection for backhaul. They are usually deployed in a plug-and-play manner, without manual cell planning, and therefore require self-optimising capabilities to automatically set the radio parameters. While single femtocells have been deployed in residential homes, one other application of femtocells is to be deployed in a group to provide contiguous coverage over a larger area, such as an office building. When deployed in this manner, one of the aspects that requires optimization is the coverage. Here, the problem of jointly adjusting coverage of a group of femtocells is investigated.

The considered objectives of coverage optimization in group femtocells are to:

- Reduce the coverage gaps within the femtocell group's intended area of coverage (Figure 3).
- Reduce the leakage of coverage outside the intended area of coverage. This is to reduce the amount of signalling originating from mobility procedure requests from macrocell users, lowering the power consumption and reducing interference caused by pilot channel transmissions in co-channel deployments (Figure 3).
- Perform load balancing to prevent overloading.

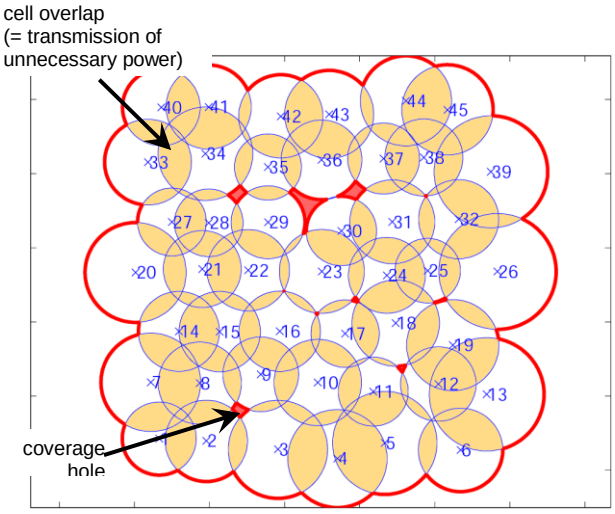


Figure 3 : Illustrative example of coverage in a femtocell group.

As the femtocells utilise omnidirectional antennas, the adjustment of coverage is done by modifying its pilot channel transmit power. The three objectives can result in conflicting requirements. For example, increasing the coverage would reduce coverage holes, but doing this may increase the leakage of coverage outside, and cause congestion due to the larger number of users being covered. Conversely, reducing coverage would reduce leakage and load, but can cause coverage holes to occur. Therefore, the adjustment of coverage has to be done in order to balance the conflicting objectives.

The proposed approach is to utilize genetic programming (GP) to generate distributed coverage algorithms that would adjust the coverage of the femtocells in order to balance the three objectives. GP is an evolutionary computation technique used to evolve algorithms that perform a certain task, specified by a fitness function. It is important to note that genetic programming in this case is *not* being used to optimize parameters, but to automatically create the algorithm used in the network that optimizes parameters (Figure 4), so it can be viewed as a machine learning approach. Currently, femtocells deployed commercially use fairly straightforward approaches to adjusting coverage. This includes, in residential femtocells, setting the coverage based on measurements done on the macrocell received power, or even based on manual cell planning in enterprise femtocell or metrocell deployments. In these cases, the power is typically adjusted to satisfy one objective, namely coverage, rather than balancing the need for multiple objectives. The coverage is also fixed once it has been set, i.e. it does not change dynamically according to changes in the traffic. Therefore, a technique to dynamically adjust the coverage of femtocells according to multiple objectives in a distributed manner is looked at.

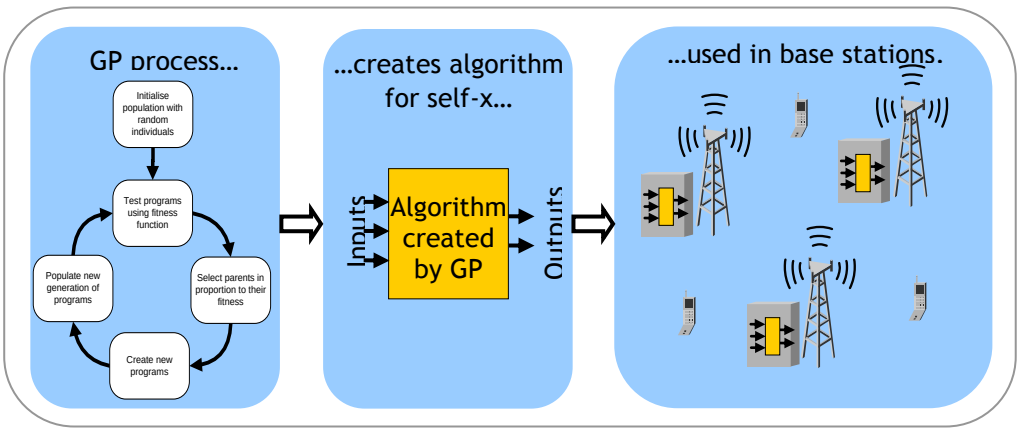
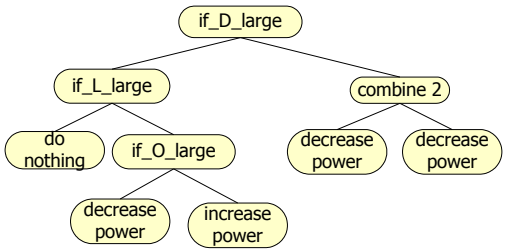


Figure 4: Overview of the application of the GP process

In GP, programs (or algorithms, we use the two terms here inter-changeably) are represented in a parse tree structure made up of functions (branches) and terminals (leaves). Figure 5 shows an example of a parse tree, and the pseudocode of the program represented by the parse tree.




```
if D>D_thr
 if L>L_thr
  donothing;
 else if O>O_thr
  decreasepow;
 else
  increasepow;
else
```

Figure 5: Example parse tree, and pseudocode.

These functions and terminals (the yellow blocks in Figure 5) are specified by the designer, and can be viewed as the "building blocks" used to construct the programs. Using the right combinations of program functions and terminals will result in a useful program.

The algorithms evolved are distributed (i.e. they are run separately on each femtocell), and use statistics of measurements made locally. The algorithm works by first collecting statistics pertaining to load, mobility events, and pilot power over a period, called here as the update period T_u. Once the update period is over, the coverage is changed based on the statistics collected. Here, a short update period of T_u = 100s is used to evolve an algorithm that changes the femtocell coverage to react to quick changes in traffic, such as the appearance of hotspots.

The fitness function F_{femto}, is given by:

$$F_{femto} = \frac{F_H W_H + F_L W_L + F_M W_M}{W_H + W_L + W_M}$$

where

F_H is the fitness associated with coverage gaps,

F_L is the fitness associated with load,

F_M is the fitness associated with leakage of coverage,

W_H, W_L and W_M are the weights used to modify the impact of each fitness component. Their values can be set to any real number, and serve to place more importance of a fitness components relative to the others. While the weights can be modified so that each femtocell is given its own individual weighting, because the evaluation of the fitness is done to give a good overall behaviour of the network, the same weights and fitness are applied for the whole network.

The weights assigned to each objective (W_H, W_L and W_M) can be used in order to optimize the fitness function, so that different priorities can be given to the objectives. Here, we set the weights to be all equal to 1, giving the same priority to all three objectives.

The algorithms are generated using building blocks called functions and terminals. As the algorithms are constructed as a parse tree (see Figure 5), the functions and terminals used are the branch and leaf nodes in the tree. Using the specified functions and terminals, the genetic programming process is used to create the coverage optimization algorithms.

The functions are set up as if-else loops that consider the load, handovers and pilot powers of a femtocell, essentially forming something similar to a decision tree. The description of these functions is given in Table 1.

Table 1: Function list

Function Name	Description
if_L_large	If L_i is higher than a threshold L_{thr}, execute branch 1, else, execute branch 2.
if_H_large	If H_i is higher than a threshold H_{thr}, execute branch 1, else, execute branch 2.
if_M_large	If M_i is higher than a threshold M_{thr}, execute branch 1, else, execute branch 2.
combine2	Execute branches 1 and 2 consecutively.
combine3	Execute branches 1, 2 and 3 consecutively.

L_i is the mean load of the femtocell in the update period i. M_i is the number of macrocell user mobility events received by the femtocell in the update period i. H_i is the proportion of mobility events between the femtocell and the macrocell to the total number of mobility events of the femtocell in the update period i. The functions combine2 and combine3 do not use any statistics as they serve to string together other functions and terminals so that they can be run consecutively.

The terminals (Table 2) are basically the actions in which the femtocell would perform with regard to the pilot powers. These are to increase or decrease the pilot power by a set value. We have used 1dB as the increment value.

Table 2: Terminal list

Terminal Name	Description
increasepow	Increase the pilot power by a set increment.
decreasepow	Decrease the pilot power by a set decrement.
donothing	Do nothing (i.e. keep the pilot power unchanged).

2.3.1.1 Genetic programming results

A simulation scenario is used to evaluate the performance of the GP evolved algorithms. The layout of the office building used is shown in Figure 6, with different wall types modelled. 12 femtocells are deployed within the building. The femtocells are assumed to use omnidirectional antennas, and each has a maximum user capacity of 8 simultaneous voice calls. There is adequate macrocell underlay coverage throughout the building. The macrocell underlay uses a separate carrier and location area code from the femtocells. The femtocells operate in closed-access mode, and only serve femtocell users in an access control list.

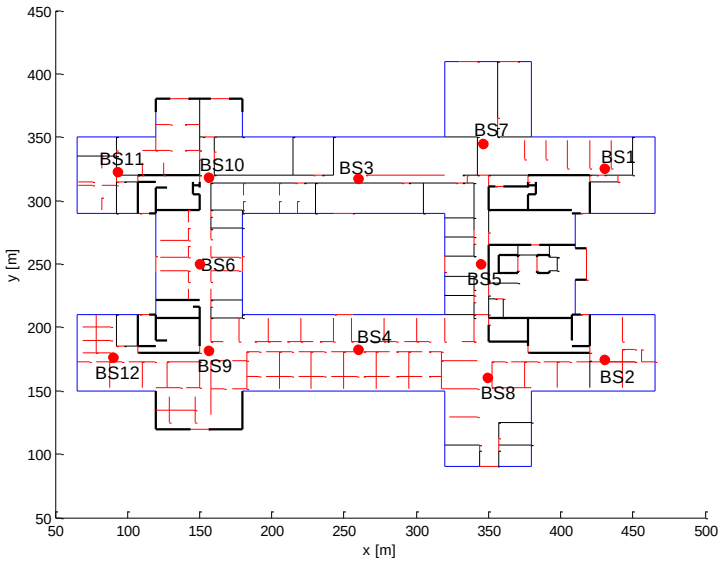


Figure 6 : Layout of office building and location of deployed femtocells

The movement of users in the building is modelled using a waypoint-based mobility model, with users moving between waypoints at a speed of $1ms^{-1}$. In addition to the waypoint mobility model, user hotspots are also modelled in order to create temporary high loads at different femtocells at different times. Macrocell users walk along the outside of the building at a speed of $1ms^{-1}$, along two paths on the northern and southern side of the building.

A GP evolution was run for 50 generations, with a population size of 50 for the three loading conditions: low, medium and high. A roulette wheel selection is used to determine the genetic operations, with a mutation probability of 0.4, crossover probability of 0.4 and reproduction probability of 0.2. The algorithm tree generated generally decreases the coverage when the leakage and load is high, and increases the coverage when coverage gaps are high. The amount by which the coverage is increased or decreased is dependent on the different combination of events (leakage, load, coverage gap).

A static coverage is used as a benchmark for comparison, where femtocells are configured with a fixed pilot power of -29 dBm. This pilot power was chosen by manually evaluating the performance of all different pilot power values and selecting the value that gives the highest average fitness for the three loading conditions.

Table 3 shows the resulting performance figures under overloaded conditions. The average load supported by the femtocells is significantly lower than the average load requested as overload conditions occur very often and last for much longer periods. However, the load supported by the femtocells with the coverage algorithm is approximately 20% higher than the load supported by the fixed coverage deployment. This illustrates the ability of the algorithm to converge to a coverage configuration that balances the load of highly loaded femtocells to neighbouring femtocells. The number of macrocell user mobility requests with the coverage algorithm is significantly lower compared with the fixed coverage deployment. The number of mobility events between macrocells and femtocell is slightly higher due to the femtocells creating temporary coverage holes when changing coverage when load balancing, but this is not significant.

Table 3: Performance results with overload conditions

	Coverage algorithm	Fixed coverage
Average requested femtocell user load	90.08 Erlangs	90.08 Erlangs
Average load supported by femtocells	80.91 Erlangs	64.89 Erlangs
Average macrocell user mobility requests per pass	0.1835	3.9998
Average femto ↔ macro mobility events to experienced by a femtocell user per hour	0.0938	0.0627

Figure 7 shows the snapshot of the coverage at the end of the simulation, showing low leakage of coverage outside. Figure 8 shows the pilot powers of the femtocells in a low and high load scenario. The femtocells are deployed with a starting pilot power of -30 dBm and converge onto their respective pilot powers, after which slight changes are made due to occasional macrocell occurrences of mobility events, where the algorithm attempts to cover coverage gaps. The changes to the femtocell pilot powers in the high load scenario are made more frequently as overload conditions occurs more often, particularly when hotspots appear, but the pilot powers still remain fairly stable after readjustments.

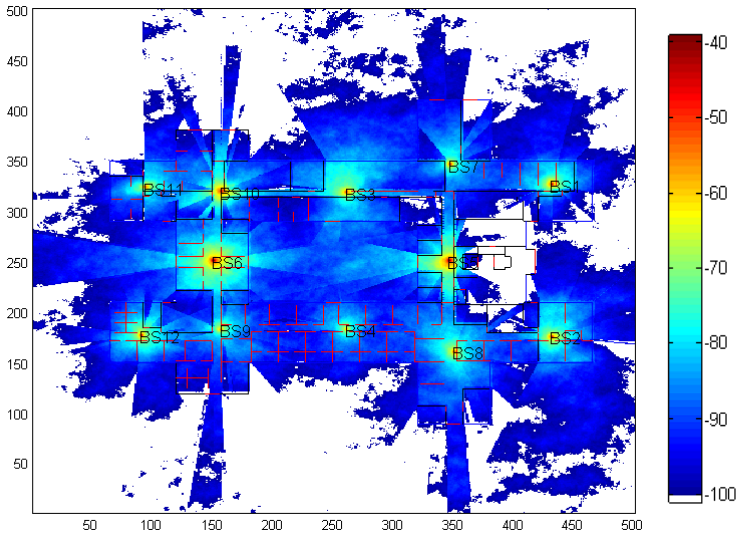


Figure 7: Snapshot of coverage at the end of high load simulation. Colourbar shows the received pilot channel power in dBm.

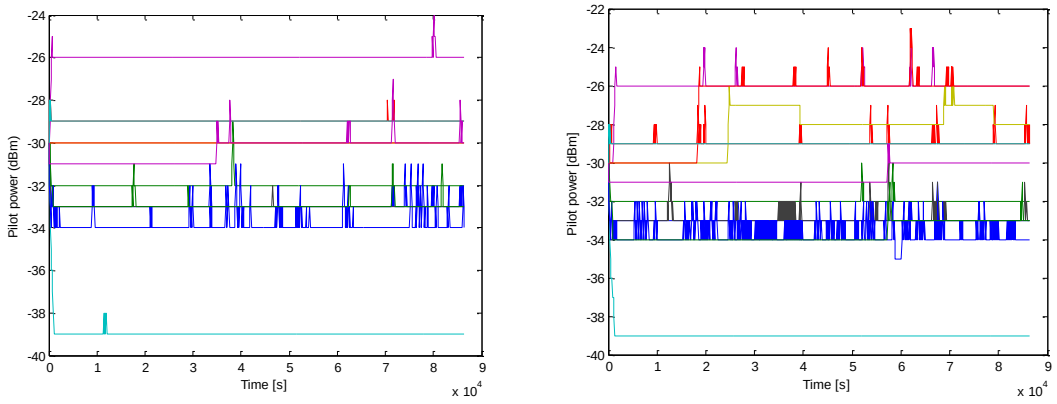


Figure 8: Femtocell pilot powers over time for low load (left) and high load (right) scenarios. Each line indicates the power level of one femtocell. Self-optimization of antenna tilts using steepest descent

2.3.1.2 Description

Individually adjusting the vertical antenna down tilt angle of every sector allows optimizing the coverage and the capacity of a cellular system. Up to now, this was achieved by human intervention and the achievements were measured by drive tests. The problem is a convex optimization problem. Today, convex optimization problems are easy to be solved by standard mathematical means. However, if the possible input is not known in advance and is based on noisy measurement data, in contrary to a closed mathematical form, the convexity is easily broken, even if the underlying problem is still mostly convex. Automatically and adaptively stripping off the noise without damaging the true data is the issue to be solved here. The new approach [8] is to save effort by applying an autonomous self-optimization.

The optimization target is defined by a utility function and an optimization area is selected. The utility function is composed by a component representing the cell edge properties and a component representing the capacity. For the cell edge, the 5% quantile from the CDF of the spectral efficiency is used, for the capacity the mean spectral efficiency is used. Both are combined by a weighted sum with the weights ten and one, respectively. The weights can be chosen arbitrarily and reflect the trade-off between coverage and capacity (which eventually is a decision of the operator). This per-cell utility is then averaged over the optimization area. The optimization area includes those neighbour cells potentially affected by changing the antenna tilts in a small region around the current optimization centre, including the centre cell. The algorithm uses the real system in operation to verify any changes. So, stability and reliability is an essential issue, as the users shall feel no degradation of their services during the iterative steps of the optimization. Exploring bad performing areas of the solution space must be done with care and rarely. Randomness must be bounded with respect to the gain to risk ratio. No prediction model is needed, as the true traffic is used to calculate the utility. Verifications are costly in the sense that they require sufficiently long measurement periods to be statistically significant. The mobility of the users and their variable request for services causes ‘noise’ on the measured utility. The algorithm must be able to cope with such statistical properties of the system response.

The algorithm in brief: In the first step, gradient based optimization slightly moves the antennas to get a feedback by how much the utility changes. From this, a gradient can be calculated, which tells in which direction the most improvement can be expected. Direction means here a weighting of the antenna change when all involved antennas are moved together in the second step. The second step now changes the antennas following the gradient direction. The issue here is how long to follow the direction, as the gradient is only an estimate. The algorithm continues until the utility begins to get definitely worse again, i.e., until reaching a clear optimum despite the noise. The parameter describing how long the gradient direction was followed is called alpha in the section below.

2.3.1.3 Results

In each optimization step, the selection of the best parameter is one of the problems to be solved. Figure 9 shows for a certain optimization step the noisy system response depending on the chosen parameter alpha (blue line, partially shown). The task here is to search for the alpha with the highest utility gain.

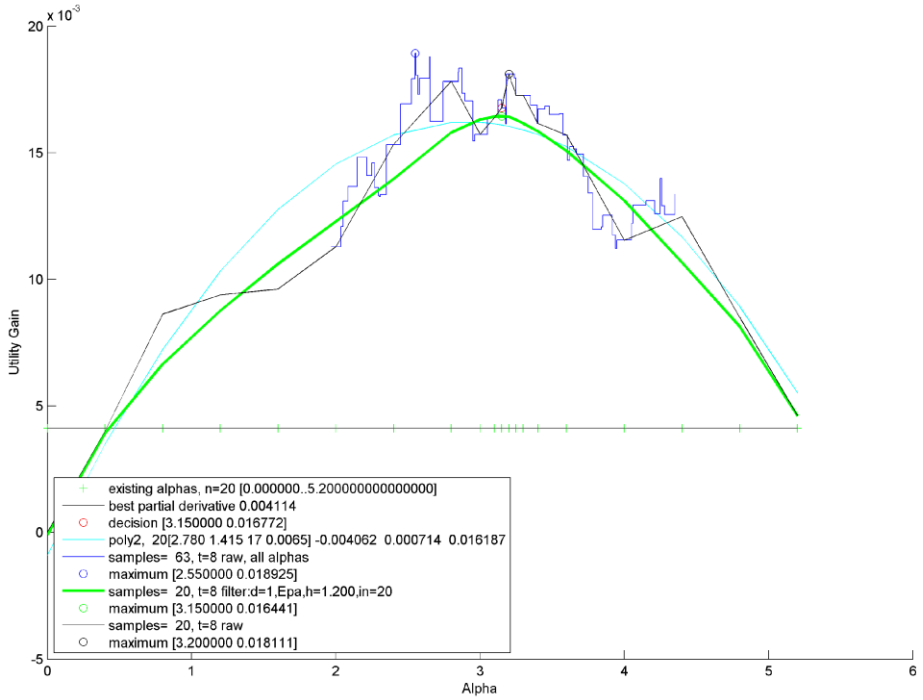


Figure 9: Utility gain with different alpha

Legend description: curves have their maximum indicated by a circle. The coordinates of the maximum are indicated in brackets as [Alpha, UtilityGain], except for the polynomial approximation (poly2 curve in cyan) where the number of points, error measures in brackets and the polynomial coefficients of the parabola are indicated. The alphas evaluated (existing alphas) are marked as green ticks on the horizontal line (which indicates the utility gain of the best partial derivative) and the respective legend indicates the number of points and the alpha-range covered. The decision taken is marked by a red circle, which is always above the green circle (obtained by smoothing), but has the height of the real data value.

To solve this task, a subset of alphas is adaptively selected: the range and density of the requested points is automatically and iteratively adjusted according to the samples seen so far. The probed alphas are shown as green tick markers on the reference line, they get denser around the estimated maximum and the total search range is automatically adapted to the interval [0, 5.2]. This range is defined by the alphas examined so far. The reference line is the best partial derivative achieved during the previous processing step of gradient calculation.

The selected points for alpha (named existing alphas) define a much smaller but sufficient subset of the potential data (the utility gain of the subset is shown as black line), as measuring all potential alphas would be much too costly in a true operational system.

The smoothing filter (green line) applied to the selected alphas (black) is intended for significantly removing the noise but still keeping the overall coarse shape. The result is typically a convex curve, well suited for a gradient descent optimization. The green circle indicates the maximum of the filtered utility gain. The red circle indicates the decision for the alpha as selected for the next iteration cycle, which is typically not the maximum out of the potential alphas, but a consolidated value based on the filtered data.

For comparison, a second order polynomial is also fitted on the subset of alphas (cyan), but not used for decision making, as it typically fits worse than the filter shown.

In conclusion, in the presence of noise, optimization problems can lose their convenient property of being convex, which makes the solving much more troublesome. After applying a filter, the noise can be removed, which makes the problem again convex or divides the problem at least in a very limited number of convex regions. The filtering algorithm still contains some heuristics as it has to be tuned to the properties of the system response and the noise observed. The target is to make also the remaining hand-tuning parameters to be self-tuning to get a robust, completely automatic optimization, even in case of a system changing over time.

2.3.2 Multihop relay-assisted cellular networks parameter optimization using GA to maximize multiuser throughput

The big challenge for broadband wireless systems is to provide a right balance between traffic demands with coverage range to offer good signal quality and service reliability at a reasonable cost. In a cellular network, system spectral efficiency more broadly includes the notion of coverage range. Multihop relay-assisted cellular networks, has widely been recognized as a promising technology to improve coverage range, user throughput, and traffic demand. The relay-assisted transmission technique will be widely used in the next-generation wireless systems to provide more uniform data rates to users who are scattered over a cell, and to save the transmit power of a mobile station (MS) in the uplink. Compared to the base station (BS) cost, deploying multihop relay stations (RSs) can reduce infrastructure cost to provide improved signal transmission quality. In addition, since multihop RSs do not need wireline connections to the core network, the relay-assisted cellular networks can be quickly deployed on a large scale.

However, transmission through a RS needs two transmission phases, which may degrade traffic demand requirement. Therefore, one interesting issue in the multihop cellular network (MCN) is to determine whether a two-hop transmission is necessary. Furthermore, it is an important task to investigate the impact of RS location on link reliability and traffic demand requirement. Specifically, if the relay stations are deployed far away from BS, the user at cell boundary can receive stronger signal from RS. However, the longer hop distance between BS and RS will decrease the relay link capacity. Therefore, determining appropriate relay location to achieve the trade-off between communication reliability and traffic demand requirement is an essential issue in MCN. In order to provide solutions in real time, in cases where there is large number of users associated to multihop links with users sharing a common traffic channel, an optimal solution is required to satisfy all users with higher throughput, by placing the RSs at a proper location inside the BS cell range to provide higher signal coverage range and to share the resource blocks effectively between direct links and multihop links active users. A way to achieve the optimal solution to fast convergence of air interface parameters with traffic demand requirements is done using a linear optimization technique such as Mixed Integer Linear Programming (MILP) that searches a vast solution space to provide fitness value. However, Genetic Algorithms (GAs) are a class of intelligent search technique to find optimal solutions by choosing a proper problem representation and corresponding evolutionary search operators to match the fitness value. They may not reach the global optimum in some cases, but they can normally reach an acceptable suboptimum solution quickly.

2.3.2.1 Proposed Approach

The genetic algorithm (GA) is tailored to specifically solve the multihop relay placement and user association to provide the required user throughput as [6]. The individual representation and genetic variations are specifically designed to suite the characteristics of the multihop relay-assisted cellular network problem. Furthermore, a population adjustment method is used to enhance its search ability. The solution to multihop relay placement and user association first require a two-tier genetic structure in order to encode the BS and RS selection and MS assignment separately as shown in Figure 10.

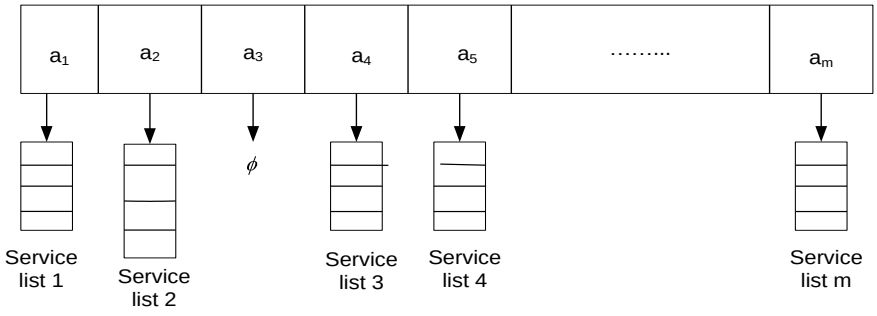


Figure 10: Two-tier chromosome representation

Given a set of MSs (*m*) and a set of BS and RS (*b, r*) with their location information, encode a mapping *M* from *m* to *b* and *r* as a two-tier chromosome. At the higher level, i.e., the BS and RS activation level, an array of length $m = |b|$ represents the BS and an array of length of length $m = |r|$ represents the RSs. Each locus of *i* of

this chromosome stands for a BS and RS as a_i, referring to its service list containing all the MSs assigned to it. If there is no MSs connected to RS (i.e., this RS is not needed), its service list is ϕ . This is referred to as the MS assignment level. For a feasible solution, the total length of the service lists should add up to $|m|$, and for each a_i the total traffic demand in the list must not exceed the BS and RS service requirement.

However, the separation is based on a more inherent difference of the information embedded in a solution, i.e., site activation and user assignment. The division of information into two tiers separates the semantics embedded in an individual. That is, the activation of a BS, RS and the assignment of an MS to an activated BS, RS are encoded in two separate domains. This allows controlling the genetic variations at these two levels independently, which turns out to be fairly powerful. This two-tier genotype is distinctive from the most common representations of GA solutions to combinatorial optimization problems. Multi-level encoding model in [5], use the BS site activation, antenna type selection, and antenna configuration are encoded as three levels.

The proposed GA approach method evolves a population of individuals with adaptive size in the generational mode to approach the optimum. The process starts with randomly generating a population P_0 of a given size. Next, each individual's fitness value in this initial population is evaluated. Then, the process enters a generational iteration outlined as follows.

- Step 1. Randomly pair up individuals of population P_t ($t=0$ at start);
- Step 2. Crossover each pair of individuals to generate $|P_t|$ offspring;
- Step 3. Repair the offspring of previous step;
- Step 4. Mutate the offspring;
- Step 5. Repair the output of previous step;
- Step 6. Evaluate offspring;
- Step 7. Calculate the next population size $|P_{t+1}| = f(|P_t|)$
- Step 8. Choose by truncation selection the next population P_{t+1} from the competition pool consist of $|P_t|$ parent and $|P_t|$ offspring individuals;
- Step 9. Go to step 1 if termination criterion is not met.

The iterative process stops when the best fitness value in the population has remained the same for s (stagnation threshold) individual evaluations. This termination condition will signal if the evolution stagnates. The measure of how fast the algorithm leads the process to a possibly global/local optimum before stagnation b recording the number of individual evaluations elapsed so far.

The coverage range parameter gives the percentage of target MS locations examined which can support the signal to noise ratio (SNR) required to satisfy service requirement. The traffic demand parameter gives an expectation of mean per-user gain with respect to RS (multihop link) with BS (direct link). This optimization task aims to meet the optimal RS placement at the radio access network (RAN) and also the assignment of users to either BS (direct link) or RS (multihop link).

An extended service in a MCN single cell with a BS in the centre supported by RS sharing the available traffic channel resources to provide required service to direct hop and multihop users. Each RS is placed at a particular location to cover the users at the cell edge without affecting the user throughput. The RS position is matched to the propagation conditions (path loss, large scale macro diversity), frequency reuse and channel state. Each user has to be associated with a direct link (BS) or multihop link (RS) that can be translated to provide the required throughput by matching the traffic channel resources with channel state. However, each user experiences a different channel state among the different available traffic channels shared. As a result each link provides a different data rate when associated to different RSs or with BS according to the channel state. According to this each user has to be associated with a particular link to balance traffic with the channel resources.

In order to determine the appropriate RS site activation with BS position and user association pattern to satisfy the available traffic channel resource requests, a genetic algorithm is employed. The genetic algorithm (GA) takes into account the RS site activation, user association, frequency reuse pattern with traffic channel resources for each user, RS, and BS. The optimal solution obtained, determines the optimal traffic channel resources to be allocated to which user's group by either fixed resource planning (FRP) or dynamic resource planning (DRP) for proper RS signal quality to satisfy multihop link as that of direct hop link user throughput. In order to evaluate the optimality of each candidate solution, the solution's fitness value is equal to the amount of optimal traffic channel resources with optimal user association to satisfy the traffic demand requirements.

The optimal solution (i.e. the solution that all users have been associated with the available traffic channel resources to provide the required data rate) has fitness equal to 0.

Many efforts have been devoted to control the critical parameter in evolutionary computation. In addition to initializing a proper population size beforehand, it is also possible to later on dynamically adjust the population size during evolution to improve the performance of an evolutionary algorithm. An empirical method for variable population size is proposed in [4].

2.3.2.2 Genetic algorithm results

In order to evaluate the proposed GA search for fitness-iterations with linear optimization technique of MILP, a single cell MCN simulation is performed, the results of which are presented in this section. The simulations were based on a scenario according to which there is a BS at the centre which provides coverage range to an area where there are direct active users. The RS are placed at appropriate locations matching the good propagation conditions for extending coverage range of cell edge users as shown in Figure 11. The total user throughput depends on the good signal quality, frequency reuse pattern and traffic channel resources to match proper user association. The users are requesting maximum throughput by properly associating with direct links or multihop links depending on the location for access. The total requested resources from BS and RS to provide required throughput for all the users is based on the traffic demand requirement. The BS with RSs has 100 users or 300 users uniformly distributed at a range of 1km to share the available traffic channel resources in order to provide the required service requirement by proper user association to satisfy the traffic demand.

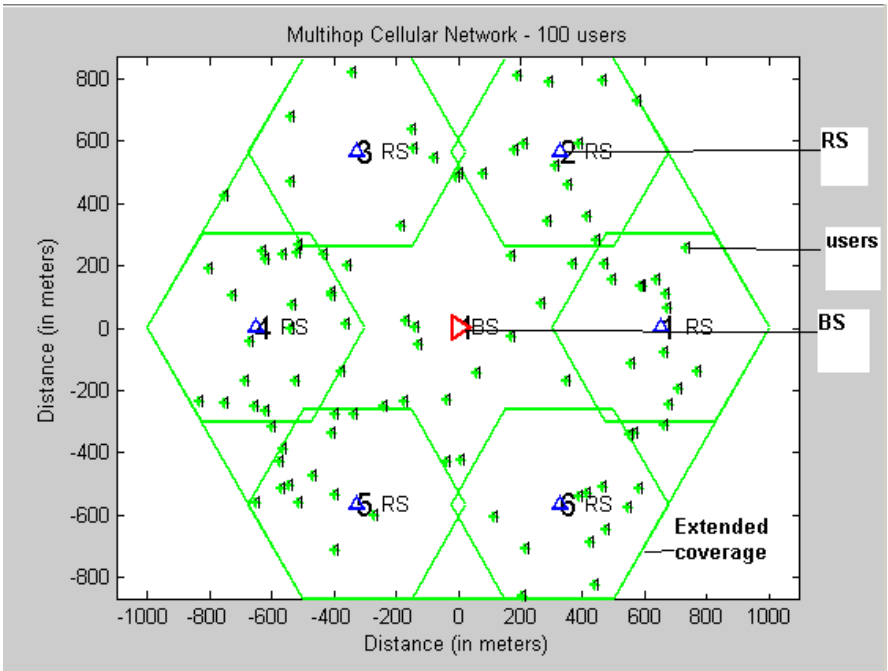


Figure 11: Multihop Cellular Network with BS and RS with user association

The parameters of the GA are the following:

- Chromosome size: 100 and 500
- Initial Population size: $|P_0| = 200$
- Termination stagnation threshold s = 10000
- Crossover rate: 0.7
- Mutation rate: 0.3

In the following Mixed Integer Linear Programming (MILP) optimization is used. The main principle of the MILP functionality is that every possible state *s* (solution) of the system (optimization problem) has a flow demand for uplink and downlink (which indicates the solution's placement). The objective is to find the state *s* with the RS placement (i.e. with the maximum coverage range). In each optimization step a neighbouring state *s'* of the current state *s* is computed. The optimal value of making the transition from *s* to *s'* is specified by an

acceptance limit, that depends on the RS placement propagation conditions of the two states, and on a global traffic demand varying parameter C_i^{λ} called the available resources. The multihop cellular network system's resources have a fixed value between BS and RS at the start of the optimization and it decreases by a decrease factor in every step for active user association.

In order to evaluate the convergence evaluation metrics *m* of GA with MILP, the fitness-iterations (i.e. how fast the algorithm finds the best solution) and resolution (i.e. simulation resolution of minute) have been selected. Figure 12 shows the fitness value of the solution found at every step of the GA with MILP for a group of 100 users and 300 users. The MILP found the best solution after 1000 iterations and the resolution taken is also higher. The corresponding GA's best solution is found after 300 iterations the resolution taken is lower than MILP. From the above we can conclude that the MILP converges slower to find the fitness value than GA's proposed one and also gave the values not optimal for some cases.

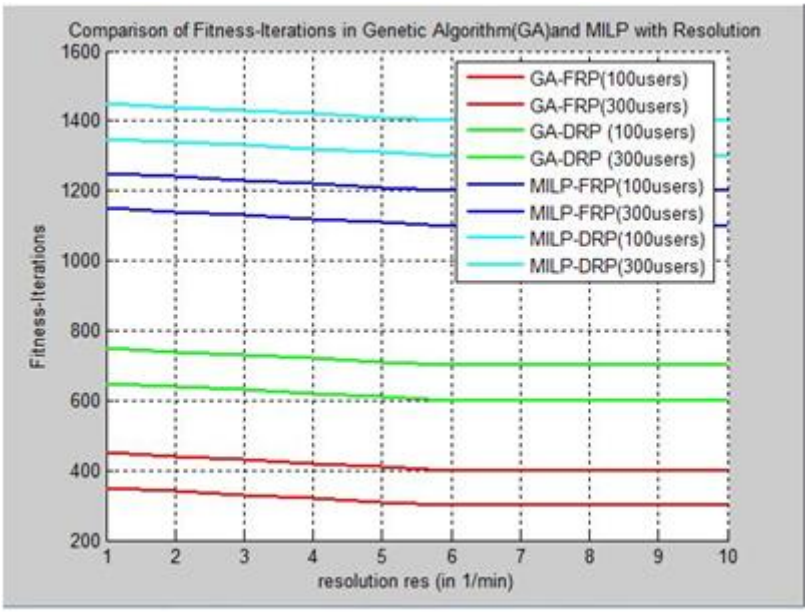


Figure 12: Comparison of Fitness-Iterations in GA and MILP with resolution

In Figure 13, the fitness value of the solution found at every iteration of the MILP with GA is presented. The GA found its best solution at the step 300 and those solutions had fitness value 0.05 and keep falling afterwards to 0. The corresponding MILP's best solution found at the step 1000 and that solution had fitness value approximated to 0.09 and keep falling afterwards to around 0.05. GA converges to its optimal solution and even after a small number of iterations it is better than the final solution of MILP.

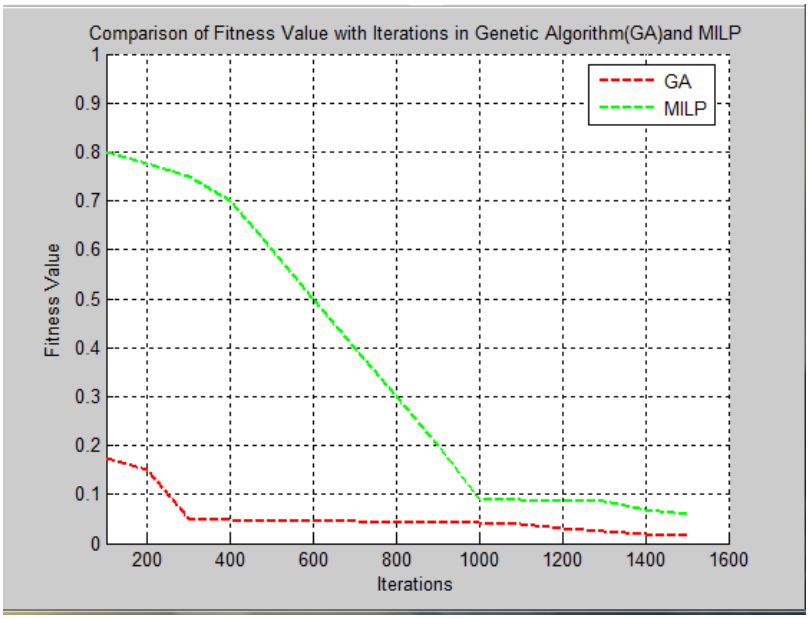


Figure 13: Convergence of fitness values

2.3.3 Routing for MPLS traffic engineering using GA

In Future Networks environments, there exist strict requirements for specific levels of network QoS in terms of changing load conditions. The dynamic route adaptation of individual traffic flows allows the utilisation of underutilised resources and improves the system performance. In this direction, Traffic Engineering (TE) techniques are employed for the parameter optimization of legacy protocols (OSPF, MPLS), dealing with the inherent disadvantages of these protocols. Moreover, in order to preserve the overall system performance and stability in high standards, new TE optimization techniques should be utilized. This work provides solutions to these problems using evolutionary algorithms and more specifically Genetic Algorithms.

In this context, our approach deals with technical problems addressed within both UC5 and UC6. As regards UC5, genetic algorithms address the parameters optimization problem for legacy protocols. As regards UC6, genetic algorithms address the problem of invocation of backhaul/core segment σ. Such algorithms allow fast convergence to near optimal solution for problems with unclear size of search space (e.g. routing optimization problem); this advantage makes them suitable for investigating the handling of RAN requests.

Genetic algorithms are suitable for function optimization problems with large search space. Their main advantage is the fast convergence to near optimal solution and they are less likely of getting stuck in local optima. Our approach models the TE problem as a function optimization problem and formulates it as a genetic algorithm, taking advantage of the aforementioned benefits. More specifically, the TE problem is formulated as a multi-commodity flow problem, which is a NP-complete problem, in order to find near optimal flow patterns for a given set of requests, considering a network topology. Bandwidth requests are characterized feasible or infeasible with regards to capacity constraints along network links. Infeasible requests are rejected while the algorithm attempts to minimize overall network congestion and maximize potential for traffic growth.

The genetic algorithm fitness function minimizes the sum of link utilisation of all the links with regards to capacity constraints and link costs. The fitness function is computed as follows:

$$F = \sum_{(i,j)\epsilon\, E} [c_{ij} * \sum_{r=1}^{k} x_{ij}^{r}]$$

where:

- x_{ij}^{r} indicates edges leaving node *i* towards neighbouring nodes for the *r*-th flow
- c_{ij} the cost of link *ij*
- *k* is the number of flows
- *E* represents the number of edges.

Fitness function is subject to the following five constraints:

$$\sum_{r=1}^{k} x_{ij}^{r} \leq cap_{ij}$$

$$\sum_{j:(i,j)\,\epsilon\,E} x_{ij}^{r} - \sum_{j:(i,j)\,\epsilon\,E} x_{ji}^{r} = 0\ ,\forall\ r\ \epsilon\ [k], \forall\ i : i\ \neq s_r, d_r$$

$$\sum_{j:(i,j)\,\epsilon\,E} x_{ij}^{r} - \sum_{j:(i,j)\,\epsilon\,E} x_{ji}^{r} = B_r\ ,\forall\ r\ \epsilon\ [k], \forall\ i : i\ = s_r$$

$$\sum_{j:(i,j)\,\epsilon\,E} x_{ij}^{r} - \sum_{j:(i,j)\,\epsilon\,E} x_{ji}^{r} = -B_r\ ,\forall\ r\ \epsilon\ [k], \forall\ i : i\ = d_r$$

$$x_{ij}^{r} \geq 0$$

Where:

- x_{ji}^{r} indicates edges coming towards node *i* from neighbour nodes for the *r*-th flow
- cap_{ij} indicates the capacity of link *ij*
- B_r is the requested bandwidth for flow *r*,
- d_r and s_r are the destination and source node of flow *r*
- Vector *X* has #flows $*|E|$ elements, where $|E|$ is the number of edges.

The first constraint is related to the capacity limit of each *ij* link. The following three constraints refer to the load conservation of each node given a feasible solution. More specifically, the second constraint implies that if a node is not the source, or the destination node of the r^{th} flow, then it should not conserve any load. The third and the fourth constraint are similar to the second, but they are conformed to the source node and the destination node respectively. So, for the source node a positive value of B_r, which is the requested bandwidth for the r^{th} flow, should be preserved as the node's load. Accordingly, a negative value of B_r should be preserved for the destination node. Finally, the last constraint ensures that all edges are assigned with a non negative value of load.

As mentioned before, infeasible bandwidth requests are rejected. In order to model this behaviour extra edges with infinite cost between source and destination are required. These edges are utilized if and only if the network load of the bandwidth requests exceeds the capacity of the links along route between source and destination.

2.3.3.1 Traffic engineering results

Early validation results in terms of key network metrics such as time requirements and admission rate have been also measured. Future work comprises comparison between our formulation and other commonly used function optimization techniques that are supported by the Global Optimization toolbox of Matlab.

Experiments are conducted on a wide range of different scenario simulations. Initial validation has been realised targeting key network metrics such as load distribution in the network links and algorithm convergence time. The focus of this validation lies in the comparison with other optimization techniques such as constrained non linear minimization technique.

Figure 14 illustrates the network topology with 5 routers. Preliminary results were produced for this topology and 2 bandwidth flows. The first is between router 1 and 3 and the second is between 2 and 4. As it can be seen two infinite cost edges with respect to bandwidth flows have been added (blue colour edges).

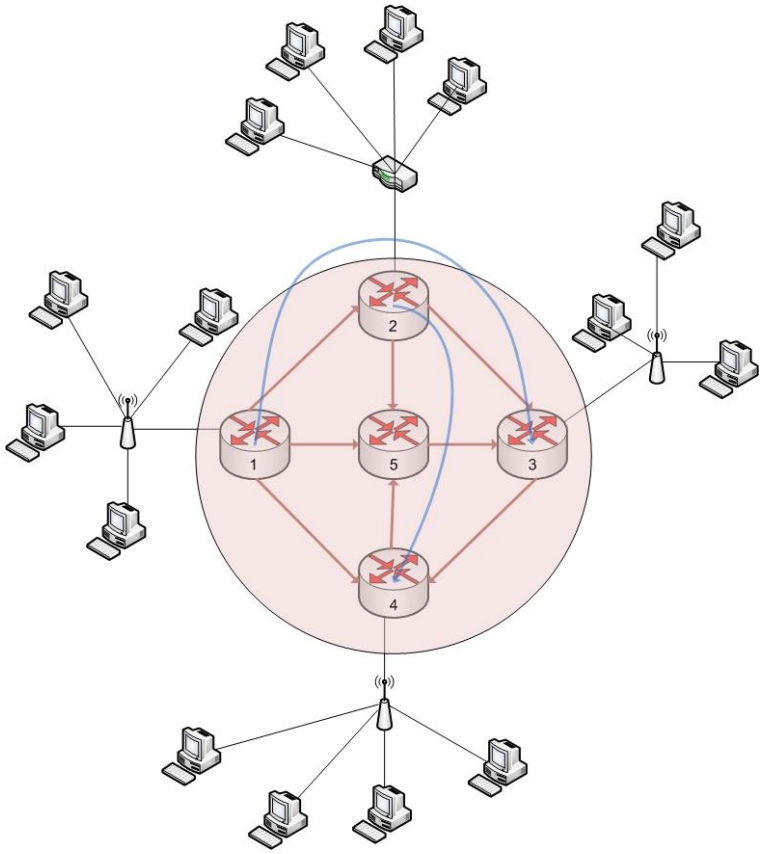


Figure 14: Network Topology

Initial simulation results based on this network topology are shown on Figure 15. Fmincon is a function included in the Optimization Toolbox of Matlab that attempts to find the minimum value of a constrained nonlinear multivariable function. As it can be observed even for a simple network topology, the GA approach outperforms Fmincon in terms of load distribution as the load is more spread out among the links compared with the results for Fmincon, which uses fewer links compared with GA. Moreover, Fmincon fails to void congestion (i.e. network link 4 is almost fully utilized). So, if an emergency situation occurs GA will reject less traffic flows than Fmincon.

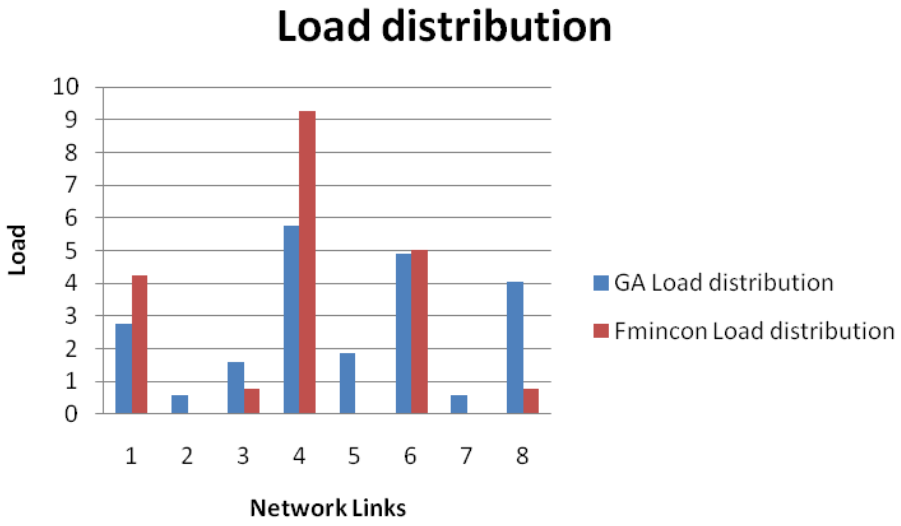


Figure 15: GA and Fmincon load distribution

2.3.4 OFDM resource allocation to users using GA

Another group of parameters that can be optimized is the allocation decision of OFDM resources at the Radio Access Network. This optimization task aims to meet the optimal resource utilization at the RAN and also the satisfaction of the users' request. In [52],[53] a genetic-based algorithm for adaptive bit and subcarrier allocation is presented. In this piece of work the algorithm takes into account the channel state of each subcarrier, the number of allocated subcarriers to each user, the requested number of bits transmitted per symbol (requested bit rate) and the actual number of bits transmitted per symbol (actual bit rate). The fitness value of each solution depends on the total transmit power requested if this solution is adopted. A familiar approach is presented in [7], but in this piece of work a different chromosome structure is used. In [7] each chromosome consists of N genes (each gene represents a subcarrier) and each gene is divided into two parts. The first part represents the user who is assigned to the subcarrier, and the second part uses two bits to represent the modulation used. In [54] the proposed genetic-based algorithm uses chromosomes that consist of an equal number of subcarrier genes and each gene represents the user who is assigned to the subcarrier that this gene indicates. For each solution the water-filling algorithm is used in order to determine the amount of data bits that are going to be transmitted by the available subcarriers according to the channel state. Then each solution is evaluated according to the total transmission power requested if this solution is adopted.

The OFDM resource allocation genetic algorithm that is proposed in this section is designed in line with the work which has already been done in the state of the art. The main difference of the proposed algorithm is the optimization objective which is the provision of the appropriate resources in order to satisfy the QoS request of the network users. An advantage of the proposed optimization algorithm in contrast with the state of the art is that it is in line with the context of the policy-driven resource allocation as it takes into account the high-level policies that the Network Operator (NO) have selected to enforce. This is so as the QoS of each user that belongs to a user class is defined by the high-level goals of the NO, and by this way any possible change of the high-level goal of the NO regarding the QoS of a user class is appropriately being addressed by the optimization algorithm as the optimization objective changes in line with the QoS level change.

2.3.4.1 Description

Each cell of the network is controlled by a BS which has been assigned with a set of available subcarriers and which provides certain services to a set of active users inside this cell. Each user has a different QoS requirement that can be translated to a provided throughput requirement. Also each user experiences a different channel state among the different available subcarriers. As a result each subcarrier provides a different data rate when assigned to different users (according to the channel's SINR value). According to this each user has to be allocated with the appropriate amount of resources that will provide him the requested QoS.

In order to determine the appropriate OFDM resource allocation pattern that satisfies all the users' requests, a genetic algorithm is deployed. This genetic algorithm (GA) takes into account the channel state of each user at each resource element and the requested QoS of each user. The final solution found by the algorithm, determines which resource elements (subcarriers in our case) are going to be allocated to which user (i.e. the allocation pattern). In order to evaluate the optimality of each candidate solution, the solution's fitness value is equal to the amount of the users' requested data rate that remains unsatisfied. The optimal solution (i.e. the solution that all users have been provided with their requested data rate) has fitness value equal to 0.

2.3.4.2 Problem formulation

A further formulation of the previously defined problem is presented in this section.

First of all we assume that variable $i \in [0,N)$ represents a subcarrier and variable $j \in [0,K)$ represents a user. Also the following variables are defined in order to help to the formulation of the problem.

- $\forall j \; \exists \, q_j$ which represents the requested QoS in terms of throughput of each user according to the network operator's (NO's) policies.
- $\forall j,i \; \exists \, s_{j,i}$ which represents the channel state (SNR or signal to noise and interference ratio = SNIR) of subcarrier *i* if it is assigned to user *j*. This piece of information can be derived from feedback from the users or by the pilot subcarriers that are used.

- $p(s_{j,i})$ is the data rate that can be provided to user *j* by the subcarrier *i*, based on the given channel state.
- $\forall j,i \; \exists \; u_{i,j} = \begin{cases} 1, \text{if subcarrier i is assinged to user j} \\ 0, \text{otherwise} \end{cases}$, which is the subcarrier allocation vector and determines which subcarrier is going to be allocated to which user.
- $\forall$ j $\exists \; a_i = \sum_{j=0}^{N-1} u_{j,i} \cdot p(s_{j,i})$, which represents the achieved provided data rate to user *j* by all the subcarriers that have been allocated to him.

The objective is:

$$\min \sum_{j=0}^{K-1} (q_j - a_j), \text{where } (q_j - a_j) = 0 \text{ if } q_j \le a_j .$$

Subject to:

$$\sum_{i=0}^{N-1} u_{i,j} \ge 1, \forall \text{ j} \in [0,\text{K})$$

(each user has to be allocated with at least one subcarrier)

$$\sum_{j=0}^{K-1} u_{i,j} = 1, \forall \text{ i} \in [0,\text{N})$$

(each subcarrier can be allocated just to one user)

$$\sum_{j=0}^{K-1} \sum_{i=0}^{N-1} u_{i,j} \le N$$

(not more than the available subcarriers can be allocated)

2.3.4.3 Genetic algorithm modelling of the problem

The objective of the presented genetic algorithm is to allocate the proper number of subcarriers at the system's users according to the requested QoS that each user has. To model this allocation problem, the chromosome needs to contain information about which subcarrier is going to be allocated to which user.

According to this the structure of the chromosome that has been selected is the following:

The chromosome is separated into blocks. The number of blocks is equal to the number of subcarriers that the system will assign at the network segment that we observe and the indicator of each block corresponds to the subcarrier with the same indicator. Each block is a binary string whose decimal representation corresponds to a user indicator. The block indicated subcarrier is going to be assigned at the user that the block represents. A short example of the chromosome structure follows.

010	110	. . .	111
Subcarrier 0 is going to be assigned at user 2	Subcarrier 1 is going to be assigned at user 6		Subcarrier n is going to be assigned at user 7

So if there are N subcarriers to be allocated at the network segment that we observe and *K* users, the length of the chromosome will be: $N \cdot \log_2(K)$.

Each solution of the algorithm corresponds to a specific $u_{i,j}$ array. In order to evaluate each solution the appropriate fitness function has to be selected. The fitness function has to evaluate the solution according to how much the users' requested QoS is been satisfied. Hence for each solution the corresponding a_j vector (for all *j*) is calculated and then the fitness function computes the difference between the required and the

achieved data rate of every user as: $\sum_{j=0}^{K-1}(q_j - a_j)$. By this fitness value it is clear how much the users are satisfied. The fitness value 0 means that all users are satisfied.

2.3.4.4 Results

In order to evaluate the introduced algorithm's performance, some experimentation was performed, the results of which are presented in this section. The experimentations were based on a scenario according to which there is a base station (BS) which provides services to an area where there are 22 active users. The users are requesting services with different QoS (for simplicity there are two types of provided services, one with 1024Kbps and another with 256Kbps requested throughput). The total requested throughput from all the users is equal to 7168Kbps. The BS has 100 total available subcarriers in order to serve the requested load.

Figure 16: Graphical representation of the scenario

The parameters of the GA are the following:

- Chromosome size: 500
- Population size: 50
- Tournament selection size: 20
- Elitism size: 3
- Crossover rate: 0.7
- Mutation rate: 0.3

In order to compare the performance of the introduced genetic algorithm, a simulated annealing (SA) algorithm is employed. The main principle of the simulated annealing functionality is that every possible state *s* (solution) of the system (optimization problem) has an energy level *E(s)* (which indicates the solution's cost). The objective is to find the state with the minimum energy (i.e. with the minimum cost). In each algorithm step a neighbouring state s' of the current state *s* is computed. The probability of making the transition from *s* to s' is specified by an acceptance probability function *P(e,e',T)*, that depends on the energies *e = E(s)* and *e' = E(s')* of the two states, and on a global time-varying parameter *T* called the temperature. The system's temperature has an initial value at the start of the algorithm and it decreases by a decrease factor in every algorithm's step.

The parameters of the SA are the following:

- Initial temperature: 100 degrees
- Decrease factor: 0.96

As performance evaluation metrics of the algorithm, the convergence (i.e. how fast the algorithm founds the best solution) and optimality (i.e. how optimal is the solution found) have been selected. In Figure 17 the fitness value of the solution found at every step of the two algorithms is presented. The SA found its best solution at the step 1800 and that solution had fitness value equal to 106. The corresponding GA's best solution

found at the step 2100 and had fitness value equal to 26. From the above we can conclude that the SA converged faster than the GA, but the solution that GA converged to was closer to the optimal solution than the corresponding solution of the SA

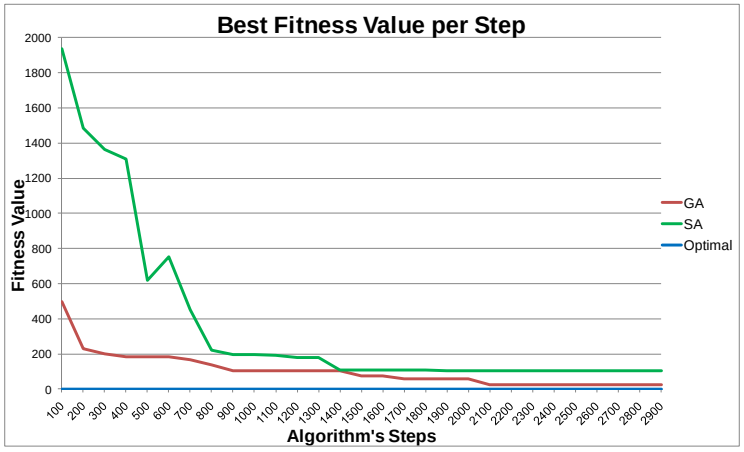


Figure 17: Best solution found by the algorithms per step

The optimality of the best solutions found by the two algorithms is represented in Figure 18. The solution of the SA left 6 users unsatisfied (the provided throughput was less than the requested) and the corresponding solution of the GA left just 1 user unsatisfied.

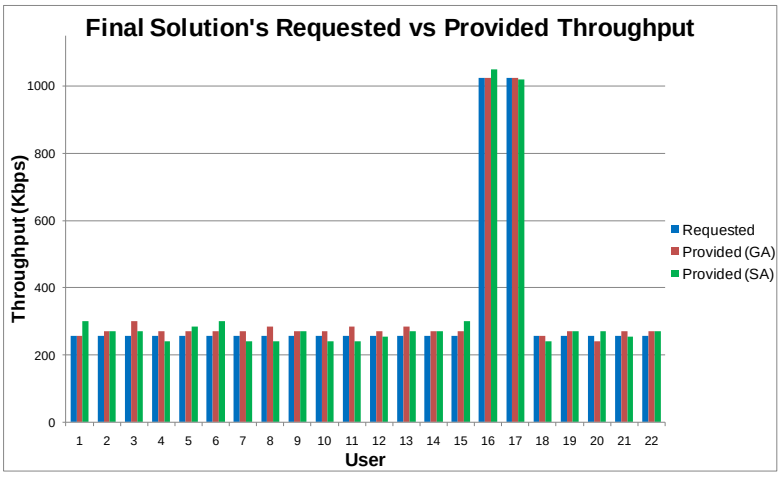


Figure 18: Requested and provided throughput per user by the two algorithms

2.4 Discussion

In this chapter, the use of techniques with random elements to perform various complex optimization problems in networks, namely evolutionary algorithms (GA and GP) and steepest descent with noise is presented. The optimization techniques looked at have been applied to various areas within the network: from the radio resource to backhaul network optimization, showing the applicability of these techniques in an end-to-end manner is possible. The simulation results showed the ability of these techniques to obtain comparatively good results. As per the nature of these techniques, there is a certain amount of random exploration of the state space in order to find good solutions. One way of overcoming this issue is to use, as with the steepest descent with noise technique, a more gradual or controlled injection of noise such that the changes introduced when searching for the state space is not too drastic, allowing for its use in online optimization in a network. However, there is always the risk of being stuck in a local optimum if too little noise is used. Given their different properties described above, the use of EAs would tend to be suited for an "offline" optimization approach, where the techniques are used to optimize the network using a priori knowledge of the network, such as topology and environment, and then the solutions produced are applied to the network.

With EAs, the nature of the approach involves a large amount of randomness at the start of the optimization process, but a solution can be found quickly after several iterations. The state space is also explored more comprehensively, so being stuck in an unfavourable local optimum is less likely with EAs. Nevertheless, the use of EAs in an online manner may not be desirable as it could cause too much disruption in a network whilst evolving a solution, and EAs would be more practically applied in an offline manner. However, the development of a model-building technique to automatically create a network model based on real-time information such as network measurements and topological information could conceivably be used to perform an online optimization using EAs in the future. The EA approaches used here share the common properties of having the solutions represented in a chromosome form, and use the same principles of the selection of individuals with highest fitness and the application of mutation and recombination. The development of common techniques for the refinement of EAs, such as optimization of population size, and mutation and recombination rates would apply commonly to all these techniques.

The techniques that are used here make use of objective functions (or fitness/utility functions) that use high-level performance metrics such as throughput, capacity and link utilisation. These objective functions can be adapted fairly easily to take other metrics into consideration. This means that any user or operator requirement can be translated and integrated as an objective to the network optimization technique in a relatively straightforward manner. For example, the requirements can be translated to objective functions automatically at the Unified Management Framework and passed on to the optimization algorithms, giving us an autonomous network approach to manipulating the network behaviour.

The open issues that have been identified with the current study are as follows:

- A detailed mapping of the techniques to specific areas of the use cases defined in the project, and the requirements with regards to the interfaces that would be used within the network and the UMF.
- The implementation of EAs in an online manner. There are several approaches that can be investigated on this. The most straightforward includes a study of how quickly the algorithms can converge to a solution in a particular application area, and whether these solutions can be obtained and implemented quickly in a network in an online fashion. A more long-term area of study is an investigation into the use of model-building processes that would enable the automated creation of up-to-date network models that can be used to apply EA optimizations for problems where the network topology and environment is changing dynamically.

3 Governance and autonomic management of OFDM/MPLS segments

This chapter focuses on the solution of problems that stem directly from a project's use-case, namely UC6 "Operator-governed, end-to-end, autonomic joint network and service management". The term "operator-governed" implies the ability of the operator to be able to express and set its business goals at a relative high level, without having to resort to manual lower level configuration goals, and still see these business goals being translated to lower level goals and commands and being translated in network and service configurations able to meet these business objectives. It also implies that, even though the various decision entities at the lower levels have the ability to drive network and service reconfigurations so as to adapt to changes in the traffic conditions -without explicitly requiring the intervention of the operator-, the operator is still kept in the loop, and is informed about these changes. The operator is also alerted in cases where, despite all the efforts of the lower level decision entities, the business goals cannot be met and therefore a "system-wide" intervention or change in business goals themselves is needed. The root cause leading to such situations is also provided to the operator so that they can reason about the potential remedial actions.

In brief, UC6 deals with the situation that a Network Operator (NO) wants to deploy new services and/or accommodate new traffic on top of its multi-vendor and multi-technology infrastructures, with the focus being placed on IP/MPLS (Multiprotocol Label Switching) backhaul/core segments and OFDM based Radio Access Networks (RANs). Several problems need to be tackled in the context of service deployment so as to achieve a coordinated, end-to-end performance, and they have to do with the different manifestations of heterogeneity in the considered underlying networks (Figure 19) namely, heterogeneity in the used technologies and domains, in the equipment coming from different vendors and of course in the management systems used even for the same kind of technology/domain.

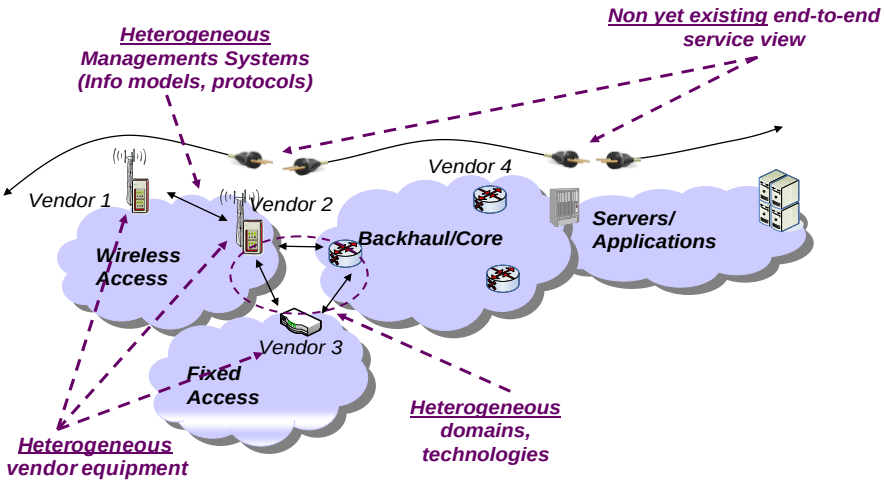


Figure 19: Heterogeneous nature of the network

Although there have been efforts towards integrating and automating the service deployment processes, they didn't actually succeed to provide a complete solution. Although they may be elaborate enough they exhibit loose or no integration and partial and no automation at all. Humans are still highly involved into the processes and pretty often they require in depth skills and expertise to perform manual work, e.g. for translating service requirements to network configuration, for configuring elements etc. Operators rely on tools and processes that are not as flexible as they should be, on the contrary they are static and are based on worst case planning, thus resulting in an undesirable over-provisioning of resources. In general, the above negatively impact the CAPital and OPerational EXpenditures (CAPEX/OPEX), also associated with the time needed for the operator to accommodate the new service/traffic request.

Given the described problems, the objective is to provide a unified, goal-based, autonomic management system for the service deployment and/or new traffic accommodation on top of heterogeneous networks encompassing both OFDM-based RANs and MPLS-based backhaul/core segments. To this effect the use case aims at finding solutions that will:

- Enable operators to describe their goals and objectives, through high-level means and govern their network. Low-level (system specific) policies seem not adequate enough for determining the desired performance across the whole network.
- Achieve policy-based operation of Radio Access Network (OFDMA-based) and Backhaul/Core Network (IP/MPLS-based) segments, which is optimized with respect to QoE/QoS (and energy) efficiency, taking into account metrics derived in network nodes and end-user devices and in line with the operator objectives.
- Achieve coherence between these segments through cooperation, negotiation and federation

During the use case lifecycle, selected RANs and Backhaul/Core network segments will be called to investigate the satisfaction of the business level service/traffic requests (and their associated QoS levels), coming from operators. This actually calls for formulating and solving both RAN and Backhaul/Core network optimization problems that take into account operator goals (policies). This can be split as follows, as also depicted in Figure 20:

1. *Autonomic management of OFDM-based RAN segments*: This a) aims at RAN parameter optimization based on operator policies, b) can be applied to OFDM(A) resource allocation, relay selection and link positioning in multi-hop cellular OFDM networks and c) can resort in stochastic optimization methods/algorithms and metaheuristics (e.g. genetic algorithms) or game theoretic approaches.
2. *Autonomic management of MPLS-based backhaul/core segments:* This aims at Ingress/Egress node selection and route optimization and generally, optimization of traffic engineering based on operator policies, it involves Label Switched Path (LSP) configuration in IP/MPLS case (backhaul side), as well as Gateway (GW) (e.g., Service GW (SGW), Packet Data Network GW (PDN-GW)) (re)selection/configuration, GW migration/dimensioning (e.g. at the core part), policy-based, green traffic engineering, and can rely on maximum-flow/graph-theory or bio-inspired approaches
3. *Impact of governance – policies in the algorithms above:* It involves Autonomic adaptation of objective functions, utility functions, action policies, constraints etc.

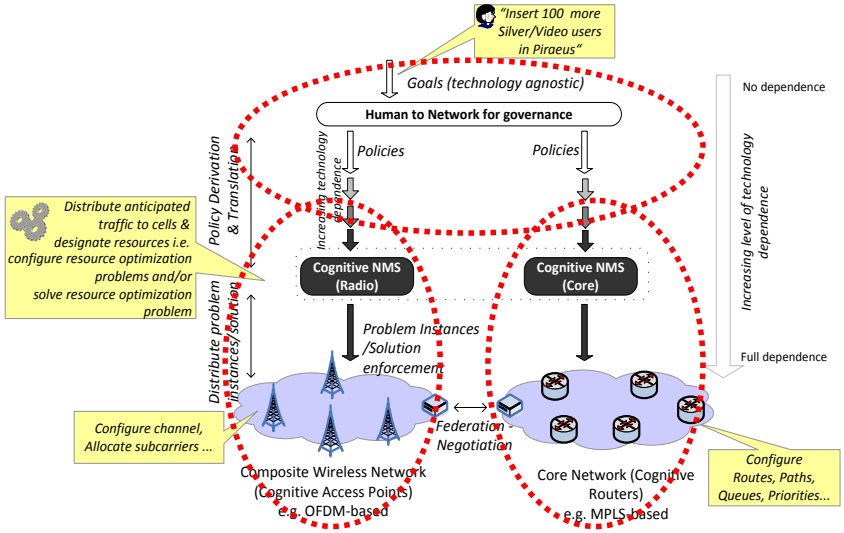


Figure 20: Mechanisms envisaged

Network governance can be defined as the ability of a Network Operator to control, manage and intervene in the operation of an even autonomic network. Governance can also be considered as a high level mechanism which involves all functionalities necessary to address the gap between high-level specification of human operators' objectives and existing resource management infrastructures towards the achievement of global goals. In order to do so, governance encompasses Human-to-Network (H2N) communication interfaces for the introduction of policies and business goals to the network.

All in all, in order to achieve end to end joint network management/governance, the desired performance has to be mapped into high-level policies that should affect the network as a whole. In the opposite direction, there comes the need to have common performance metrics (probably technology agnostic) across the network in order to monitor its performance in a joint manner.

The above designate the relationship of the mechanisms envisaged herewith with the Unified Management Framework (UMF). More specifically and first, UMF should have the required information so as to formulate a goal for the mechanisms, second, it should provide the means for communicating such goal to the mechanisms, and last but not least, it should be able to receive and handle the possible instabilities of the mechanisms. As it will be made clearer in the sequel of this chapter, the particularities of each specific optimization problem and its application environment can be sent to the so called governance entity, goals can be formulated as high level policies, with utility functions being good candidates among others, and that allow relative weightings among different objectives and finally, feedback to governance entity that calls for altering goals (policies, utility functions etc.) is expected for reporting instability issues.

The specific proposed optimization mechanisms that will be presented in the rest of this section extend the state-of the art from an algorithmic point of view by considering combinations of network and service objectives that have not been addressed in the past as well as proposing new or adapting existing methodologies accordingly so as to be applicable in this new context.

The optimization mechanisms also are designed so as to exhibit an inherent level of autonomicity in their behaviour. As long as the specific goals to be enforced by each optimization mechanism -received through the governance entity- can be met, no further manual intervention is needed. Manual intervention is needed in case these goals cannot be any longer met. In such cases, guidance from the governance entity that has a more complete end-to-end view, and therefore a more complete reasoning and understanding of the cause of these violations, is needed.

3.1 Policy-based autonomic network management - State of the Art

In the context of autonomic network management there have been done many works that address problems which are derived from this context. Some of the most representative ones are presented in this section. The work in [59], presents a summarization of the potential challenges that derive for the autonomic network management from the conceptual, functional, architecture, information and behavioural perspectives. In this work there are also presented the main autonomic network architectures that are used and a bio-inspired autonomic management framework that enables self-governance behaviour of the network and which elaborates in the minimization of the human intervention. In [60], an autonomic network management architecture named FOCALE is introduced. This management architecture takes into account human-specified business goals, that determine how resources in the network should be collectively utilized, context-aware policy management processes, in order to adapt the management control loops used to meet changing user requirements, business goals, and environmental conditions, and a combination of information and data modelling, augmented by ontological data, to enable the system to learn and reason about itself and its environment. The adopted approach in that work is to minimize human intervention and to focus it more on business concerns than on low-level device configuration.

Policy-based network management is a well-known research object that has attracted the interest of many works. In works like [64] and [65], the semantics of policies, the language by which the policies need to be formed and how policies affect adaptation mechanisms of applications to environmental or contextual changes in mobile systems, are addressed. The work done in [61] introduced how policies can coexist with traditional hierarchical management systems in order to enable QoS guaranteed Differentiated Service provision over IP networks. Considerations on policy-based network management issues, like how policies can be considered as means for programmable and extendable management systems and how hierarchical policies can be formed, as well as the functional architecture, named TEQUILA, that enables the hierarchical policy-based management of IP networks, were presented.

In the sense of end-to-end policy-based network management the work in [62] presented an end-to-end management framework for high performance switching networks. The framework proposed consists of three different layer management components named Network and Protocol Management (NPM), Management Computing System (MCS), and Application-Centric Management (ACM). By these management components the framework can address issues of utilizing existing network management tools, core management issues in order to provide system management services to enable the development of efficient proactive management, and issues required to develop application specific management techniques and manage applications so they can meet their requirements in real-time. In the same end-to-end sense, the authors in [63] presented an end-to-end policy based network management architecture. In this approach the target is to ensure that the

business policies and needs of the network operator will be supported and met by the network, the applications and services it supports.

Finally, in the context of autonomic joint network and service management the work in [66] presents a service delivery framework by which the service providers can create and deploy QoS guaranteed services over IP. The work is based on the introduction of an Autonomic Service Architecture (ASA) which exploits the application of autonomic management principles (i.e. the automated management of computing resources encompassing the characteristics of self-configuration, self-optimization, self-healing, and self-protection) in order to ensure the delivery of telecommunications services over IP networks.

The main difference of the work presented in this chapter compared with the state of the art is that in this work the main emphasis is given to the joint end-to-end management of the network. This means that in this approach the main focus is given at the policy-based governance of networks, which are composed by multi-vendor and multi-technology segments, as a whole, regardless of the individual network segments' nature. In this approach the human intervention is limited in the high-level governance of the network (that includes tasks like high-level business goal expression) and it tends to be eliminated from all the other management tasks like network configuration or network monitoring.

3.2 Reference problem formulations (From use case to problem statement)

As stated in the introductory section, this chapter focuses on the formulation and solution of parameter optimization related problems that derive from Use Case 6 (see also D4.1). Particularly, in this context, RAN and Backhaul/Core network segments are called to investigate ways to satisfy the business level service/traffic requests (and their associated QoS levels), coming from operators as part of a new service/traffic deployment.

In the use-case description document, these problems are referred to as "Invoke Radio Access Networks" and "Invoke Backhaul/Core Networks" respectively, and they are represented and described using the "black-box" methodology. The use of black-boxes to describe a problem hide the details of the system internals and requires no or minimum knowledge to assess its behaviour or external response and on the contrary it places focus on the needed inputs for the problem to be solved and expected outputs as an outcome from the application of a solution method. These inputs and outputs, which can be used to provide reference formulations, they are depicted and described in more detail in Figure 21 and Figure 22 for RANs and Backhaul/Core Networks, respectively.

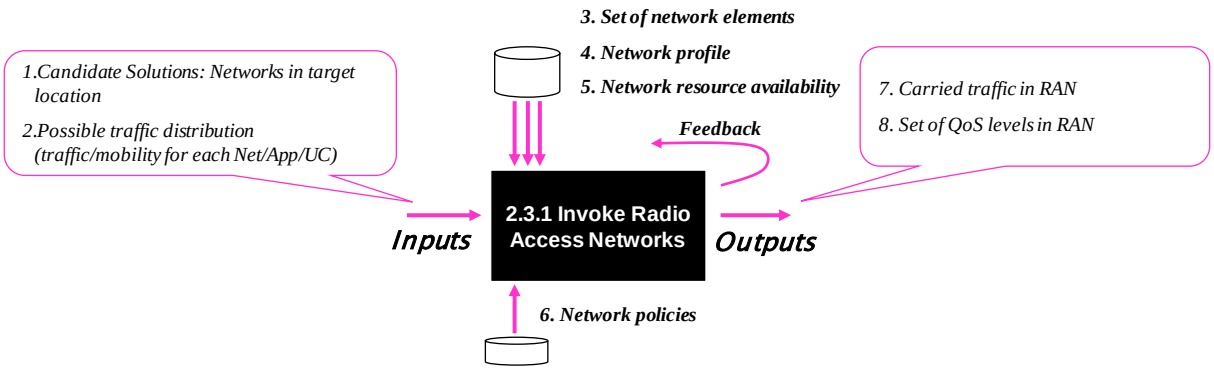


Figure 21: Invoke RAN black box

Where the inputs consist in:

1. the set of radio access networks available in the target location (the location where a specific traffic/service request exists),
2. the traffic and mobility distribution (in the available networks in the target location),
3. the set of available network elements (eNodeBs, Relays etc.) in the target location (in accordance with policies),
4. the network profile of each available element,
5. the resource availability (current context) of resources related to transmission, storage, computing load, energy; which may includes metrics of the form “probability that resource parameter will be at a certain range given the location and time”,
6. the currently valid network policies,

   and the expected outputs are:

7. the carried traffic that the RAN can support (in terms of number of sessions that can be accommodated, or the amount of throughput that can be supported) and
8. the set of QoS levels that can be offered to a specific user class of the RAN.

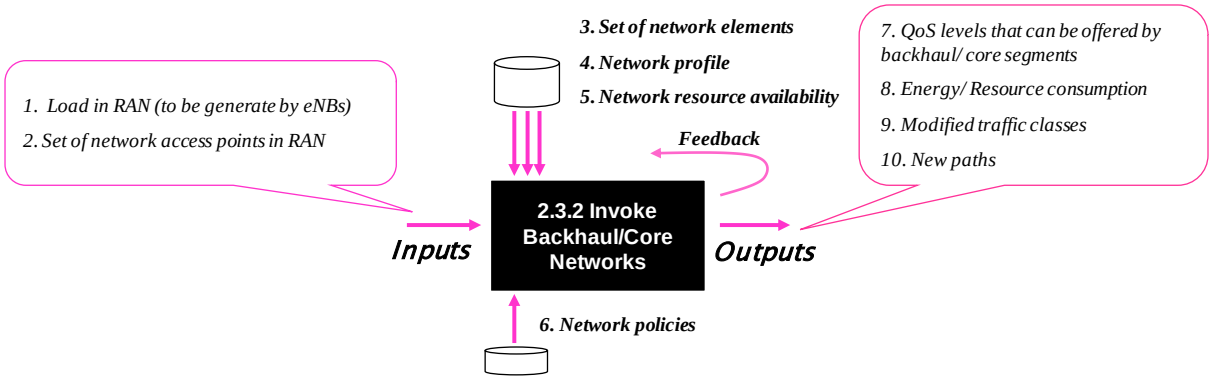


Figure 22: Invoke Backhaul/Core Networks black box

Where the inputs consist in:

1. the amount of traffic load that will be generated in the RAN network by the potential network elements,
2. the set of network access point in the RAN that connect to the Core network,
3. the set of Core network elements (ingress and egress nodes) that can be used in particular location and time period, in accordance with policies,
4. the network profile of the ingress and egress nodes e.g., network interfaces, transmission resources, etc.,
5. the resource availability (current context) of resources related to transmission, storage, computing load, energy; which may include metrics of the form “probability that resource parameter will be at a certain range given the location and time”,
6. the network policies that are active in ingress and egress nodes,

   and the expected outputs are:

7. the QoS levels that are offered by the backhaul/ core segments,
8. the consumption of energy and resources (Green concept),
9. the modified traffic classes in the Backhaul/Core segment,
10. new paths that are created in the Backhaul/Core network segment.

3.3 The role of policies

As already stated, the role of policies and the way they affect the optimization problems/algorithms hold a prominent position in the governance and autonomic management of RAN and core segments. This can be also extracted from the reference problem formulations that are tackled herewith, where policies comprise important inputs in both the RAN and the Core segments invocation tasks, both being derived from the high-level operator goals or locally and currently (in the time of invocation) valid ones. Anywise, policies are used to define the optimization objectives that need to be achieved and also the constraints that should be taken into account in each of the corresponding optimization problems. Some discussion with respect to the variant types of policies and their usage is deemed important here.

According to the authors of [12], there can be three types of operator, pre-defined policies: action policies, goal policies and utility function policies. Action policies are rules of the form IF *<condition>* THEN *<action>*. They put forward specific actions that the system will take in pre-defined system states defined by the conditions. Goal policies specify one or more desired system states instead of specifying the actions, and let the automated management decide on the optimum actions that lead to the desired system state(s). Utility function policies are similar to goal policies but instead of having a binary repartition for the desired and non-desired states, they put the system states on a grey-scale defining a utility function value for each one of them.

As underlined in the work of Kephart and Das of IBM Research [11] utility function policies are more appropriate for automated management of high-level goals/objectives than traditional action policies and goal policies. Compared to goal policies, they provide a finer degree of preference between several potentially desirable system states. Compared to action policies, they provide a higher-degree of autonomy by avoiding the human administrator to specify the particular action in order to achieve the desired system state(s).

In general, utility functions have been used in many fields and a variety of applications including economics [13], e.g. for expressing the reasonable preferences of a consumer in different circumstances, in business management, e.g. as in [14] where they have been applied to express the service level agreement between a computer-based service provider and a client, and of course in the field of artificial intelligence [15] as a form of preference specification. Furthermore utility functions have been used widely in autonomic computing systems in order to achieve self-management [1].

More explicitly in the context of autonomic computing systems in [11], there are several application environments that are acting as autonomic elements and they are trying to optimize their utilities in both element and system level. Each one of them is assigned with a certain amount of resources, which is specified by a resource arbiter. As a way to estimate how efficient its assigned resources are used, each application environment is empowered with an Application Manager that computes a service-level utility value. The corresponding utility function derives from the policies that the system's administrator has selected to use. According to the utility value of its state, the application environment tries to optimize (by several optimization mechanisms) the resource utilisation by altering some control parameters and by this way to achieve the element's desired performance. Also each application environment computes a resource-level utility value that specifies the value to the application environment of obtaining a possible level of resources. These resource-level utility values are sent to the resource arbiter and then the resource arbiter re-computes the resource allocation to the application environments that optimizes the global system's utility.

In the context of Governance of OFDM/MPLS segments and in a high level view, each network segment (RAN/Core) can be considered as an application environment that acts like an autonomous element. Each network segment is also empowered with a segment manager (SM) that calculates the service and resource level utility value of the segment and determines the optimization actions that have to take place in order to maximize the segment's performance. Also each network segment consists of the corresponding network nodes, i.e. routers and switches for the core segment of the network as well as base stations and relays for the radio access segment of the network as shown in Figure 23.

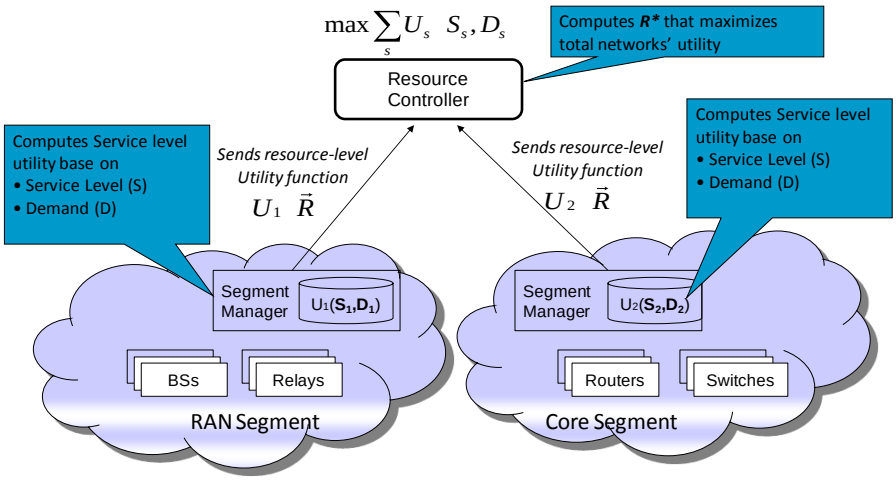


Figure 23: Network governance architecture

The service-level utility function of the network segment *i* is of the form $U_i\ \mathbf{S}_i, \mathbf{D}_i$, where S_i is the service level space and D_i is the demand space in segment *i*. In order to keep the segment's utility value as optimal as possible, given a fixed amount of resources, the SM triggers the appropriate optimization mechanisms, which have as objective the maximization of the utility function of the segment. In order to keep the network's total performance as close to the desirable value as possible, each SM also sends the resource-level utility value $U_i\ \mathbf{R}$ of its segment to a Resource Controller (RC), which is responsible for the allocation of resources among the network segments. The resource-level utility value indicates how valuable each level of resources is to the segment. The RC periodically computes a new resource allocation vector R* that maximizes the network's global utility $\sum_i U_i\ S_i, D_i\ = \sum_i \hat{U}_i\ \vec{R}_i$ and has the form: $R^* = \arg\max_R \sum_i \hat{U}_i\ R_i$.

The above processes can be considered as intra and inter domain/segment control loops, respectively, that enable the operation of the network in an autonomous self-x way. This means that after having defined the high-level goals, the network operator does not intervene further in the configuration of the network, and all the appropriate configurations-optimization decision enforcements are performed autonomously by the network.

In general, what derives from the above is that the utility function policies are the kind of policies that give more freedom to the system to take the appropriate optimization decision and introduce a more autonomous way of managing/governing the network. In addition, the described framework needs to be studied along with the development of UMF as also highlighted in the introduction. Accordingly, in the following of this chapter several policy-based parameter optimization problems are formulated based on the reference formulation above, however they do not always specify the exact policy framework for their policy-based operation. This will be finalized as part of the next steps of this work and being in conjunction with UMF specifications' releases in WP2.

3.4 Application into RAN and Core problem instances

In this chapter policy-based parameter optimization problems are formulated based on the reference formulations above and are applied in problem instances of both RAN (Section 3.4.1) and Core (Section 3.4.2) segments, whereas the use of operator policies for the governance of SON related functionalities is also discussed (Section 3.4.3).

3.4.1 RAN segment

3.4.1.1 OFDM resource allocation

3.4.1.1.1 Description

Although the whole RAN segment takes the appropriate optimization decisions in order to achieve the desirable performance, it can be considered that the same task is done by each cell of the RAN separately. In this perspective the utility function policies that the NO uses to manage the network and determine its desirable performance, are taken into account by each cell separately. In order to achieve the desirable performance in element level, each eNodeB that controls a specific cell, deploys the appropriate optimization algorithms that can optimize potential parameters, which determine the cell's performance. Such a parameter is the allocation of OFDM resources to users.

In order to optimize the used resources utilisation and achieve the desirable performance of the network segment, the system monitors potential performance evaluation metrics, and according to these metrics' values the system by the UMF determines the objectives that the segment has to achieve. This is expressed by the utility function that each element of the segment has to maximize.

According to this, in the following experimental scenario, the initial utility function policy that is used by the NO has as an objective to satisfy the users' requested QoS. The corresponding resource allocation that satisfies this objective imposes a degradation of the system's spectral efficiency. Then in order to improve this, the utility function policy changes and the corresponding objective is to increase the spectral efficiency (by minimizing the received interference at the terminals) subject to satisfy the users' requested QoS.

3.4.1.1.2 Results

The following simulation results are based on the experimental scenario that was described above. The problem statement according to which the simulations were made consists of a single cell with 300 available subcarriers and 45 users that belong to different service classes. There are three service classes, the first class' services request 256 Kbps throughput, the second class' services request 512 Kbps and the third class's services request 128 Kbps. There are 15 users in the first class, 6 in the second class and 24 in the third class. As performance evaluation metrics of the RAN, the Requested – Provided Throughput per User (T) and the System's Spectral Efficiency (SE) were selected. Furthermore the inputs that are required in this case are the channel state information (CSI) of the available subcarriers to the users, the requested QoS by the users in terms of throughput and the amount of interference caused to each of the users at each of the available subcarriers.

Initially, the target of the NO is to provide the appropriate service level to the users in order to satisfy their QoS requirements. According to this, a utility function that takes into account the user request satisfaction is adopted. As a way to maximize the utility value of the cell, the appropriate optimization algorithm (subcarrier allocation algorithm) is selected to be used. The algorithm's objective is to allocate the appropriate set of subcarriers to each user; that would provide him at least the requested QoS level. The appropriate resource allocation is found by the algorithm and then it is applied.

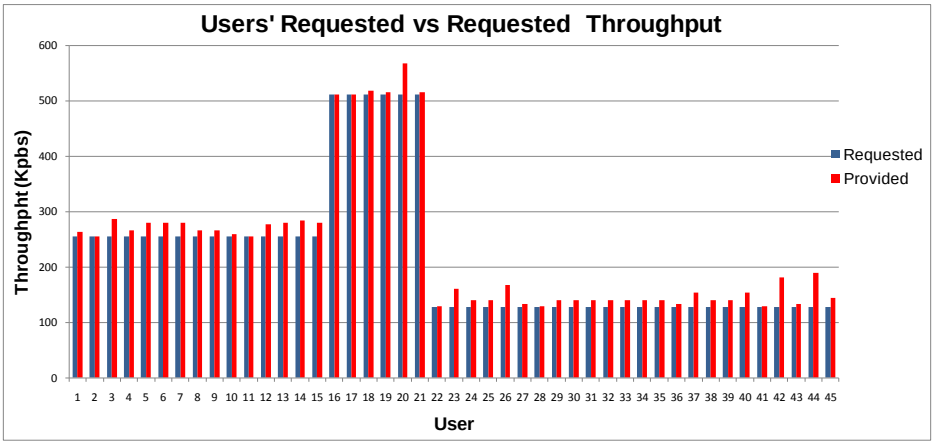


Figure 24: Provided and Requested throughput per user according to the first allocation

After the allocation of the resources, the system monitors the performance evaluation metrics that have been selected. As sown in Figure 24: Provided and Requested throughput per user according to the first allocation Figure 24, the users are fully satisfied (as the provided throughput is at least equal to the requested) and as a

result of this the corresponding metric T, which counts only the amount of unsatisfied throughput and not the over-satisfied, is equal to 0. On the one hand the total provided throughput in the cell is equal to 10.702 Kbps. On the other hand the total received interference by the terminals is equal to -37.9567 dBm and the Spectral Efficiency (SE) is equal to 2.15bits/Hz.

In order to improve the system's spectral efficiency, the utility function changes. The new utility function takes into account also the spectral efficiency except of the QoS satisfaction. In order to maximize the new utility function a new resource allocation optimization algorithm is deployed. The new optimization algorithm has as objective to minimize the total received interference while it maintains the QoS request satisfaction as constraint. The resource allocation that is determined by the new algorithm has as a result the provided throughput to every user to be at least equal to the requested as sown in Figure 25. This means that the T metric is maintained equal to 0.

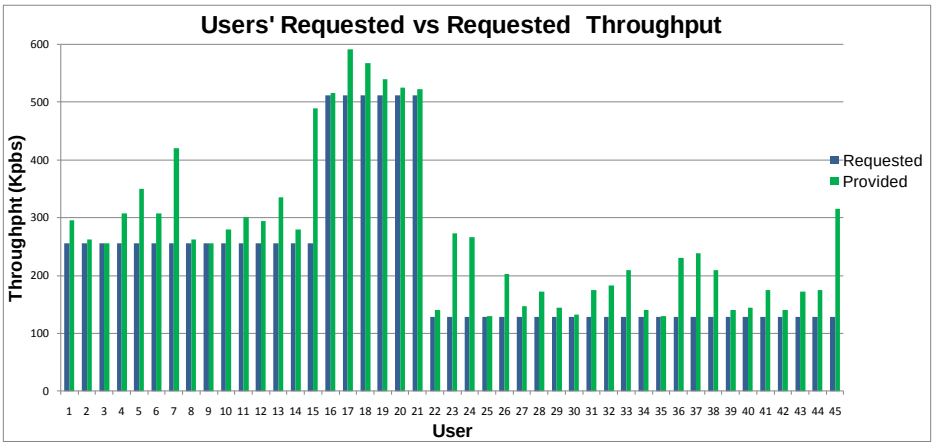


Figure 25: Provided and Requested throughput per user according to the second allocation

The total provided throughput now is equal to 12.348 Kbps. Also the received interference at every terminal is decreased, as Figure 26 illustrates and the total received interference is equal to -43.5766 dBm and the corresponding Spectral Efficiency (SE) metric is equal to 2.98bits/Hz. By this way the total cell's performance is maintained as close to the desirable value as possible.

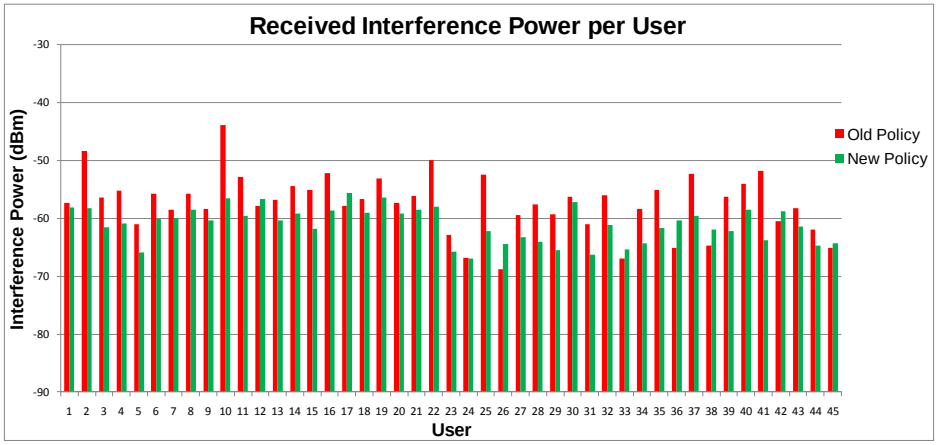


Figure 26: Received interference per user at the first and the second allocation decisions

3.4.1.2 Multi-hop selection

3.4.1.2.1 Motivation

Except from the OFDM resource allocation to the users, the user link selection is also a very important parameter that determines the achieved performance (in terms of throughput and signal-to-noise ratio) in the RAN segment. In order to provide satisfactory signal-to-noise-ratio (SNR) to users, especially at the cell edge, one solution is to decrease the cell radius. This result in more base stations (BSs) required per area thus escalating the infrastructure costs. Also, a smaller cell radius causes higher inter-cell interference, thereby calling for interference management techniques such as sectorisation and adaptive interference cancellation. An alternate solution being employed in next generation cellular systems [31] is to deploy low-cost relay stations (RSs) in each cell. The deployment of RSs has two key benefits – increase in cell capacity, and coverage extension.

Multihop cellular network planning is one of the 3GPP release 10 requirement in LTE-A networks for coverage and capacity optimization. An operator needs a plan to establish the coverage area of the BS at the centre to satisfy the RSs extension to cover the cell edge users with two hops to satisfy the future subscriber demands and density. The operator sets coverage policies, for placing RSs at a distance of maximum spectral efficiency by taking the propagation conditions with large scale macro diversity to satisfy the coverage conditions. The operator also sets capacity policies to control the sharing of the multihop channel resources to satisfy the subscriber demands by assigning with proper links either direct, or multihop link with sufficient capacity. The capacity policies assumed by the operator can be either fixed resource allocation (FRP) or dynamic resource allocation (DRP). These policies associate the subscriber demand and density with either direct link or multihop link based on available channel state.

The proposal is a general formulation of the coverage and capacity optimization problem based on operators' policies for the RAN networks. A multihop cellular network model is assumed with BS at the centre with RSs covering the cell edge users. Each BS may have traffic demands from a particular direct link users or each RS may have traffic demands from a particular multihop user. The objective of the optimization is to place RSs at a proper location (in an offline manner) to provide required coverage and to associate subscribers to satisfy the proportional fairness goal of the UMF to satisfy the required capacity in uplink and downlink. The traffic association is split among the available direct link paths at the granularity of a flow, to avoid effects that lead to performance degradation, and the multihop link paths are computed and re-computed (if it is necessary) offline by the operator, since this will become a common practice in future networks. Each user has a specified uplink and downlink demand, and the local link capacities between a user and the RS are also given as shown in Figure 27 below.

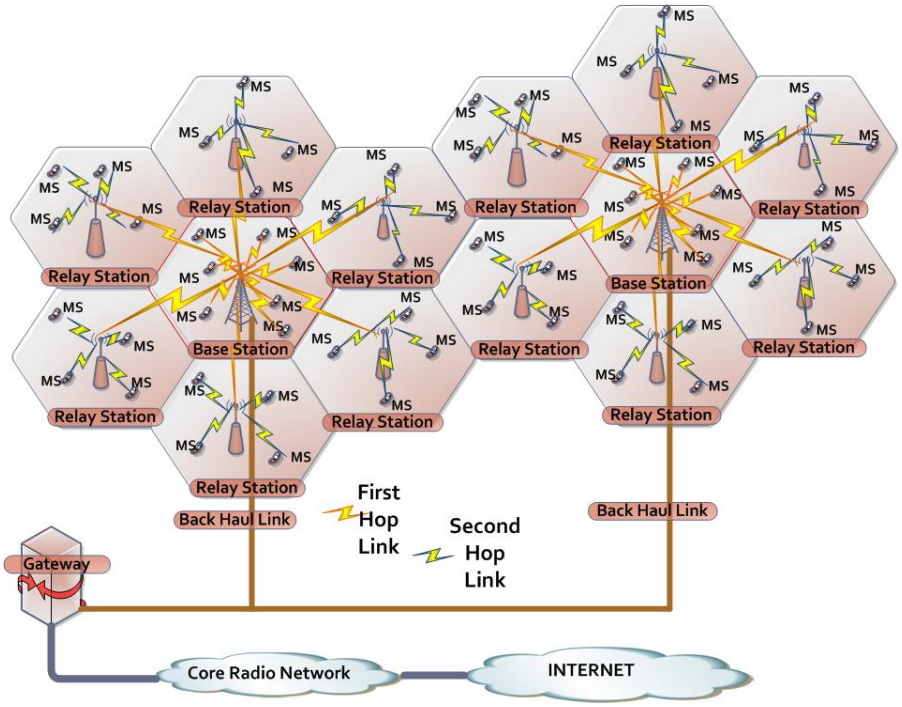


Figure 27: Multihop cellular network with BS and multiple RSs giving coverage and capacity

3.4.1.2.2 Optimization problem formulation

In the following we will use a genetic algorithm as discussed in the chapter on Random elements. A comparison is made between the policy changes affecting the optimal decisions for placement and allocation. Suppose there are *N* subscribers and one BS in the system, and they are represented by the set *V = {0, 1… N}*, where the BS is represented by the index 0. Let $V_R \subseteq V$ be the set of nodes where the installation of relay stations are feasible. Using direct link capacity from BS, from node *i* to node *j*, denoted by C_{ij}^{B} represent the capacity in terms of bit per second. Using relay access link from node *i* to node *j* is denoted by C_{ij}^{L}. Assume a cellular system of regular hexagonal cells with edge length *D*. Each cell has a BS and six RSs situated symmetrically around the BS at a distance d_r. The focus is primarily on both up-link and down-link communication. Let the total bandwidth available for uplink be W_u and the one for downlink be W_d units. Assume the user density in the cell λ, i.e., the average number of MSs in the region of area *A* is λA. Let *N(i)* be the set of nodes that interfere the operation of node *i*, where $i \in V_R$. Moreover, when there are N_C available relay link channels, let $\Lambda = \{1,2,\ldots, N_C\}$ be the relay link channel set. Furthermore, for each subscriber i there is a pre-specified uplink demand, u_i, and downlink demand, d_i. Given the above as an input to the problem, the following are the decision variables. X_i^{λ} gives the distance for relay placement and C_i^{λ} gives the bandwidth sharing between BS and RS.

$$X_i^{\lambda} = \begin{cases} 1 & \text{if an RS which uses channel } \lambda \text{ is installed at node i} \\ 0 & \text{otherwise} \end{cases} \quad i \in V_R, \lambda \in \Lambda$$

$$C_i^{\lambda} = \begin{cases} 1 & \text{if the capacity of BS or RS which uses channel } \lambda \text{ is installed at node i} \\ 0 & \text{otherwise} \end{cases} \quad i \in V_R, \lambda \in \Lambda$$

f_{ij}^{d} -> downlink direct link flow from node i to node j (bps) i,j $\in V_R$, i≠ j

f_{ij}^{u} -> uplink direct link flow from node i to node j (bps) i,j $\in V_R$, i ≠ j

h_{ij}^{d} -> downlink relay link flow from node i to node j (bps) i,j $\in V$

h_{ij}^{u} -> uplink relay link flow from node i to node j (bps) i,j $\in V$

The goal is to find the relay placement with Fixed resource allocation (FRP) or Dynamic resource allocation (DRP) in the system, which satisfies all the subscribers' demand and network constraints. The optimization formulation is as follows

$$U_{\Omega} = \min_{X} : \sum_{i \in V_R} \sum_{\lambda \in \Lambda} X_i^{\lambda} * C_i^{\lambda} \quad (1)$$

$$Subject\,To$$

$$\sum_{i \in V_R} f_{i0}^{u} + \sum_{i \in V \setminus \{0\}} h_{i0}^{u} = \sum_{i \in V \setminus \{0\}} u_i \quad (2)$$

$$\sum_{i \in V_R} f_{0i}^{d} + \sum_{i \in V \setminus \{0\}} h_{0i}^{d} = \sum_{i \in V \setminus \{0\}} d_i \quad (3)$$

$$\sum_{j \in V_R, i \neq j} f_{ji}^{u} + \sum_{j \in V \setminus \{0\}} h_{ji}^{u} = \sum_{j \in V_R, i \neq j} f_{ij}^{u} \quad \forall i \in V_R \setminus \{0\} \quad (4)$$

$$\sum_{j \in V_R, i \neq j} f_{ji}^{d} - \sum_{j \in V \setminus \{0\}} h_{ji}^{d} = \sum_{j \in V_R, i \neq j} f_{ij}^{d} \quad \forall i \in V_R \setminus \{0\} \quad (5)$$

$$\sum_{j \in V_R} h_{ij}^{u} \geq u_i \quad \forall i \in V \setminus \{0\} \quad (6)$$

$$\sum_{j \in V_R} h_{ij}^{d} \geq d_i \quad \forall i \in V \setminus \{0\} \quad (7)$$

$$\sum_{\lambda \in \Lambda} X_i^{\lambda} \leq 1 \quad \forall i \in V_R \quad (8)$$

$$\sum_{j \in V, j \neq i} \frac{h_{ij}^{d} + h_{ij}^{u}}{C_{ij}^{L}} + \frac{h_{ji}^{d} + h_{ji}^{u}}{C_{ji}^{L}} \leq (1 - \sum_{\lambda \in \Lambda} X_i^{\lambda}) k + 1 \forall i \in V_R \quad (9)$$

$$\sum_{j \in V} h_{ij}^{d} + h_{ji}^{u} \leq k \sum_{\lambda \in \Lambda} X_i^{\lambda} \quad \forall i \in V_R \quad (10)$$

$$f_{ij}^{u} + f_{ij}^{d} \leq C_{ij}^{B} \sum_{\lambda \in \Lambda} X_i^{\lambda} \quad \forall i \in V_R, j \in V_R, i \neq j \quad (11)$$

$$f_{ij}^{u} + f_{ij}^{d} \leq C_{ij}^{B} \sum_{\lambda \in \Lambda} X_j^{\lambda} \quad \forall i \in V_R, j \in V_R, i \neq j \quad (12)$$

$$X_i^{\lambda} + \sum_{j \in N(i)} X_j^{\lambda} \leq 1 \quad \forall i \in V_R, j \lambda \in \Lambda \quad (13)$$

The objective (1) minimizes the number of RSs placement to satisfy the demand for users in uplink and downlink to improve user throughput. Constraints (2) and (3) verify that the amount of traffic entering and exiting the base station equals the uplink and downlink demands, respectively. Constraints (4) and (5) verify that the amount of traffic entering each RS matches the amount of traffic exiting each RS. Constraints (6) and (7) verify that the uplink and downlink demands are met, respectively. Constraint (8) verifies that at most one channel can be assigned to an RS. Constraints (9) and (10) work together with an arbitrary large number *k*. If an RS is placed at node i, then $\sum_{\lambda \in \Lambda} X_i^{\lambda} = 1$, and the right hand side of constraint (9) is 1. Constraints (11) and (12) ensure that positive direct link traffic between node *i* and *j* exists only if an RS is placed at node *I* and an RS is placed at node *j*. Finally, constraint (12) ensures that no two RSs which use the same channel are placed in each other's interfering zone.

3.4.1.2.3 Results

Simulations were done using Matlab with one BS and six RSs for placement and channel allocation with the extended service area shown in Figure 28 with 100 and 300 subscriber stations (MS). The total spectrum size is 20 MHz which is divided into 100 resources, and therefore bandwidth = 180 kHz. The required SINR at the receiver is assumed to be 10 dB. With fixed resource allocation policy (FRP), 40% of resource allocation to BS

and 60% shared with the six RSs with optimum placement. With dynamic channel allocation the MSs associated with RSs share the channels based on the number of active users in each RS area. Figure 28 gives the user throughput with RS optimum placement with fixed resource allocation (FRP) and dynamic resource allocation (DRP). The same scenario repeated with 300 users and 300 resources for user throughput comparison with FRP and with DRP. A user throughput of around 10 Mbps is approximated for 100 users using DRP in the simulation.

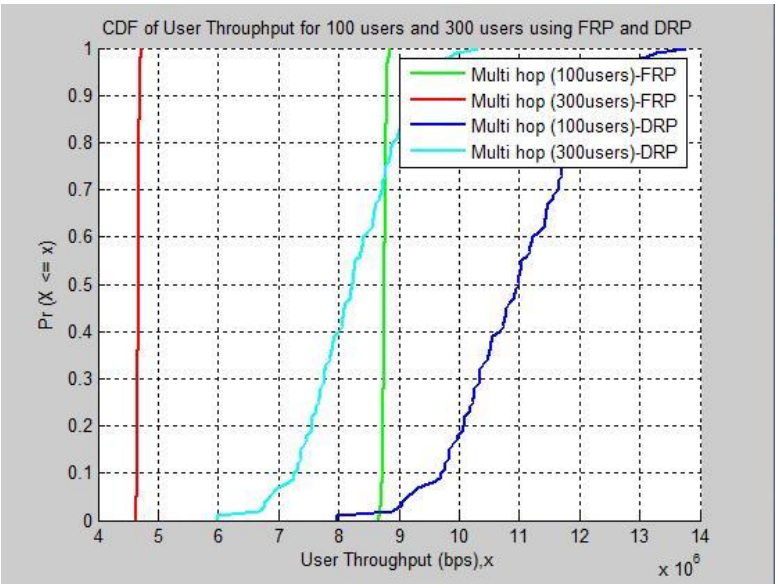


Figure 28: CDF of User throughput for 100 users and 300 users using FRP and DRP

A dynamic user arrival scenario is based on calls arriving as per Poisson process with rate θ. The location of an arriving call has uniform distribution over the cell. Moreover, locations of the arriving calls are assumed to be independent and identically distributed (i.i.d). The arriving call is accepted if sufficient resources are in the region of the call arrival; otherwise the call is blocked. The holding times of the accepted calls are i.i.d. exponential random variables with mean $1/\mu$. The aim is to obtain the call blocking probability in the cell. The comparison of call blocking rate with FRP (100 users and 300 users) and the DRP (100 users and 300 users) is shown in Figure 29. As the Erlang load increases, the call blocking probabilities of the FRP and DRP increase at different rates. The blocking probability of DRP enhanced cell is much lower than that of the FRP cell.

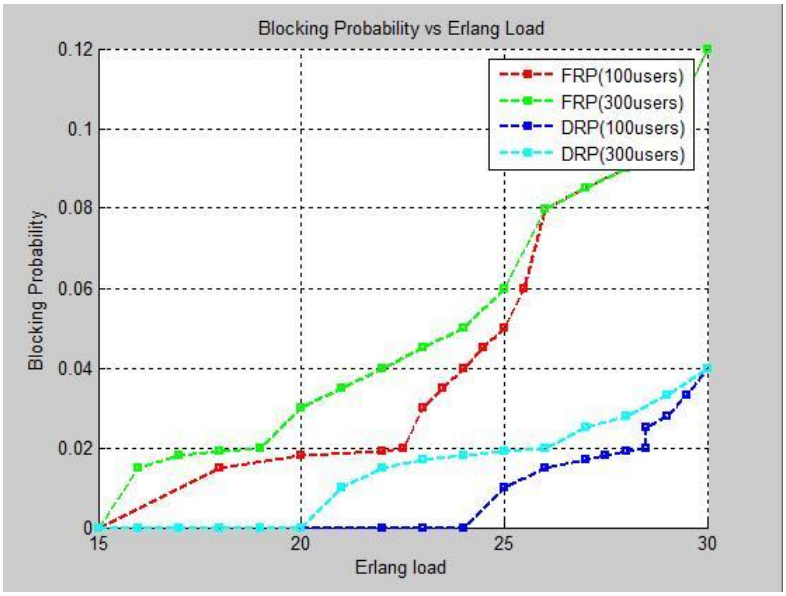


Figure 29: Blocking Probability with Erlang Load

3.4.2 Backhaul/Core segment

3.4.2.1 Energy Aware Traffic Engineering (1st approach)

Traffic Engineering (TE) receives huge attention as one of the most important mechanisms seeking to optimize network performance. The authors in [16] give an overview of the TE approaches that emerged the last years and placed focus on two major issues: quality of service (QoS) and network resilience. A general classification of these traditional-objective TE approaches is: Intradomain vs. Interdomain [17], MPLS-based vs. IP-based [18][19], Offline vs. Online [20][21], Unicast vs. Multicast [22][23]. The following work is inspired by these traditional TE approaches. Recently, routing, rate adaptation and network control are mobilized towards energy-efficient network operation [26][27]. Unfortunately, none of these approaches provide a general problem formulation in the direction of "coupling" the traditional TE objectives with the modern objectives, like energy-awareness. Our work is an attempt to modernize the research in this field.

We give a general formulation of the load balancing and the energy efficiency problems for the operator's networks. Then, we present a distributed *Energy-Aware Traffic Engineering* scheme that follows the guidelines provided by the theoretical study. We consider a network model, where each ingress router (which is the router that initiates the traffic in the Core segment) may have traffic demands for a particular egress router (which is the "drain node" of the traffic in the Core segment) or set of routers and assume multiple paths (MPLS tunnels) to deliver traffic from the ingress to the egress routers. Traffic is split among the available paths at the granularity of a flow, to avoid effects that lead to performance degradation [28], and the paths are computed and re-computed (if it is necessary) offline by the operator, since this is a common practice in today's networks.

3.4.2.1.1 Load balancing oriented problem formulation

We assume that for each ingress-egress node pair i the traffic demand is T_i and multiple paths P_i could be used to deliver the traffic from the ingress to the egress node. A fraction of the traffic in *i*, x_{ip} is routed across path p ($p \in P_i$). Table 4 contains the definition of the variables used in our problem formulation.

Table 4 : Definition of variables

Variables	Description
L	Set of links in the network
IE	Set of Ingress to Egress node pairs
e_l	Energy consumption of the port connected to link l
P_i	Set of paths of Ingress to Egress node pair i
T_i	Traffic demand of Ingress to Egress node pair i
a_l	Binary variable: 0 if link l is sleeping, 1 if link l is active
u_l	Utilisation of link l
c_l	Capacity of link l
x_{ip}	Fraction of traffic of Ingress to Egress node pair i, sent through the path p
r_{ip}	Traffic of Ingress to Egress node pair i, sent through path p
P_l	Set of paths that go through link l
L_i	Set of links that are crossed by the set of paths P_i
E	Demand of the operator in energy consumption

We formulate the problem of optimal splitting the traffic of each pair $i \hat{1} IE$ across the available paths, assuring that the maximum link utilisation (total traffic on an active link divided by the link capacity) in the network is minimized (balanced and stable network operation is assured [29]). Then, we introduce energy-awareness by identifying the set of links in the network that could be turned into sleeping mode. Therefore, we formulate the problem of finding the optimal set of "sleeping" links in order to achieve minimum energy consumption in the communication (sum of the energy consumption of the active links):

$$\min_{x_{ip}} \max_{l \in L} \sum_{i \in IE} \sum_{p \in P_i} a_l \frac{x_{ip} T_i}{c_l},$$

subject to:

$$x_{ip} \geq 0, \forall p \in P_i, \forall i \in IE$$
$$c_l \geq \sum_{i \in IE} \sum_{p \in P_l} x_{ip} T_i, \forall l \in L$$
$$\sum_{p \in P_i} x_{ip} = 1, \forall i \in IE$$
$$a_l = \{0,1\}, \forall l \in L$$
$$x_{ip} = [0,1], \forall p \in P_i, \forall i \in IE$$

$$\min_{a_l} \sum_{l \in L} e_l a_l,$$

subject to:

$$x_{ip} \geq 0, \forall p \in P_i, \forall i \in IE$$
$$a_l - u_l \geq 0, \forall l \in L$$
$$c_l \geq \sum_{i \in IE} \sum_{p \in P_l} x_{ip} T_i, \forall l \in L$$
$$u_l = \sum_{i \in IE} \sum_{p \in P_i} \frac{x_{ip} T_i}{c_l}, \forall l \in L$$
$$\sum_{p \in P_i} x_{ip} = 1, \forall i \in IE$$
$$a_l = \{0,1\}, \forall l \in L$$
$$x_{ip} = [0,1], \forall p \in P_i, \forall i \in IE$$

The previous constraints ensure that: the fraction of traffic for a specific node pair *i* sent across a path cannot be negative, the capacity of each link cannot be outreached, the traffic splitting through the available paths meets the traffic demands and the utilized links cannot be turned into sleeping mode.

3.4.2.1.2 Heuristic Energy-Aware Traffic Engineering mechanism

We present a *distributed heuristic mechanism,* which approaches the optimal *energy-aware TE solution*. The main constituents of the proposed mechanism are the following *low-complexity* and *distributed* algorithms:

- **Load Balancing (LB)**: *Given the* a_l *values for the links in the network, find the corresponding* x_{ip} *values that provide balanced network operation in terms of link utilisation.* In order to provide an efficient solution we investigate for each ingress-egress node pair the paths that goes through the maximum utilized link. Then, we "relieve" this link by moving a portion of traffic Δx and provisioning it proportionally to the rest paths (inverse procedure of progressive filling). This procedure continues till convergence to the optimal x_{ip} values (converge to optimal solution based on [29]).
- **Energy Saving (ES)**: *Given the* x_{ip} *values resulted from* **LB***, find the maximum set of links that could be turned into sleeping mode.* For each node pair in IE we find the routers that are part of the active routes and turn the lines of their network card that are not used (by any path in the network) into sleeping mode.

The proposed mechanism (Figure 30) gets as input the operator's request (*E*), as far as the energy consumption is concerned. Then, **LB** and **ES** are executed by each ingress-egress node pair i to balance the utilisation of the links (that belong to their paths), and turn the non-utilized links into sleeping mode. Next, the new energy consumption level is compared to E in order to realize if we have reached the desired state. If not, the heuristic mechanism continues by excluding the path p with the minimum $x_{ip}T_i$ (lightest path). The heuristic mechanism iterates based on the updated P_i values, optimizes x_{ip} and a_l values $\forall p \in P_i, l \in L_i$ and finally, stops when the operator's energy consumption goal is achieved.

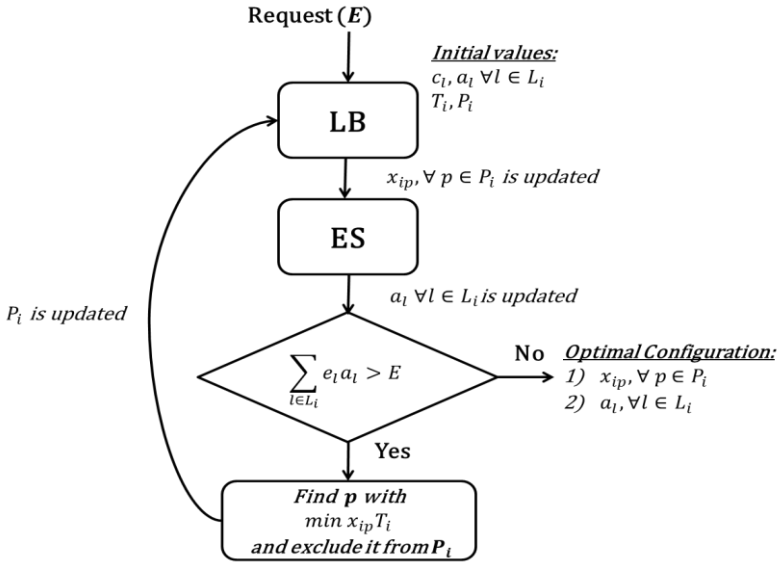


Figure 30: Heuristic energy-aware load balancing mechanism

3.4.2.1.3 Evaluation

We present the evaluation study of the proposed scheme. The validation methodology that is adopted uses the optimal solutions as a benchmark in the direction of evaluating Energy-aware Traffic Engineering (*ETE)*. We consider a network topology where four ingress nodes send traffic to four egress nodes. We are using IBM ILOG CPLEX Optimizer [30] to find the optimal solutions and evaluate the proposed mechanisms.

Figure 31 depicts the maximum link utilisation in the network vs. the total traffic demands (traffic that must be served in the network). We observe that the performance of *ETE* is bounded by the optimal solutions of the load balancing (OptLB) and the energy saving (OptES) problems and varies based on the operator's demands (e.g. Alg10 represents the performance of *ETE* when 10% energy saving is requested i.e. *E=10%).* A general outcome is that *ETE* performs close to the optimal load balanced network performance, while satisfying the operator's demands (related to the saved energy).

Figure 32 depicts the percentage of saved energy vs. the total traffic demands in the network. We observe that the operator's demands are satisfied by *ETE* while ensuring the balanced network operation (close to optimal). In other words, ETE tends to behave like an optimal load balancer in the network, influenced by the minimum energy saving level *E* that is desired from the operator.

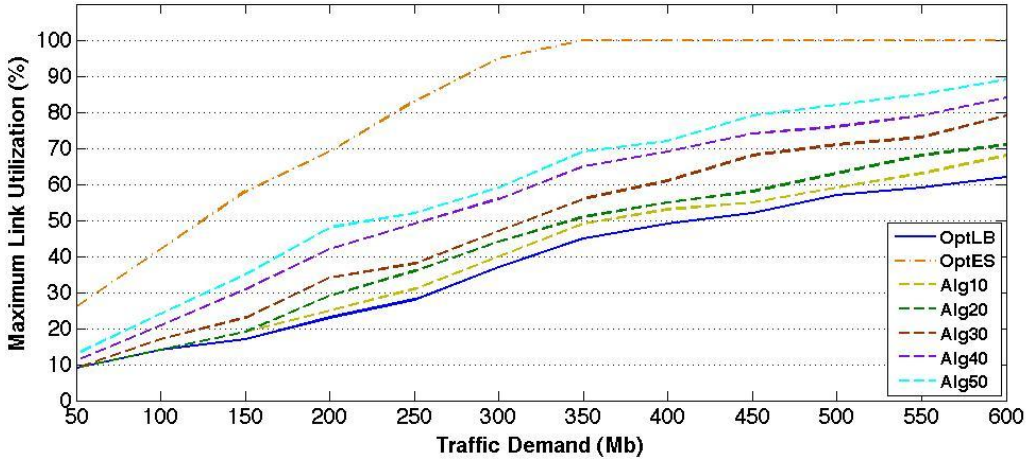


Figure 31: Maximum link utilisation vs. total traffic demand

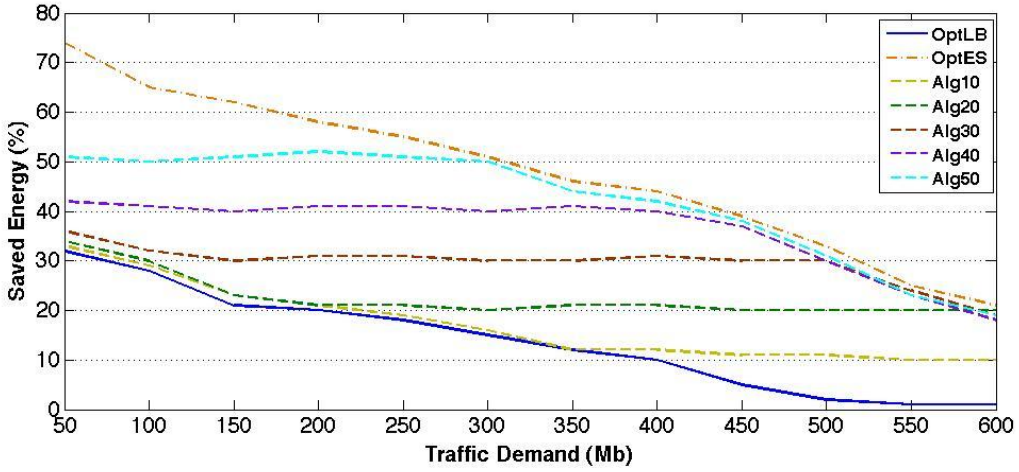


Figure 32: Percentage of saved energy vs. total traffic demand.

Table 5 presents simulation results related to the execution of *ETE*. The first column contains the operator's request, as far as energy saving is concerned. In addition, in the next two columns we observe the percentage of the links that are turned into sleeping mode and the routes that are excluded in order to approach the corresponding *E* values. The last column presents the average iterations of *ETE* till convergence. *ETE* converges after a small number of iterations, proving in this way its lightweight operation.

Table 5 : ETE Performance

Requested percentage for energy saving *(E)*	Percentage of "sleeping" links	Percentage of routes excluded	Average algorithm iterations till convergence
10%	9%	3%	3
20%	20%	11%	5
30%	28%	18%	7
40%	39%	24%	10
50%	48%	38%	12

3.4.2.2 Energy Aware Traffic Engineering (2nd approach)

3.4.2.2.1 Motivation

One important operational goal of a NO is the energy efficient operation in the Backhaul/Core segment of the network. In order to achieve this, several Energy Aware Traffic Engineering (ETE) methods can be deployed. There are several approaches of ETE as shown in Figure 33. At this stage, we have concentrated on offline intra-domain ETE. Although there have been recent proposals towards adaptive energy-aware reconfiguration of networks on the fly, it is important to note that changing the routing configuration of the network too frequently may incur instability problems, and this is in particular the case in IP routing due to the network re-convergence problem.

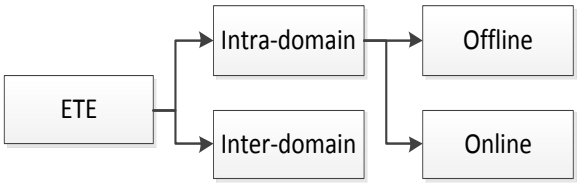


Figure 33: Different categories of ETE

As such, our strategy focuses on an *offline* approach to compute a reduced network topology (through link removal operations) with optimised energy saving performance, which can be applied in a *time-driven* manner during off-peak time on daily basis. Such an approach is ideal for those operational networks with relatively regular traffic dynamics patterns such as the GEANT network [32].

Our approach can be used by network operators which want to deploy core network configurations (in line with the end-to-end goals) that meet certain energy efficiency objectives together with QoS related objectives. These target values for these objectives should be provided by the UMF, which has a holistic view of all the segments (core and RAN) that participate in an end-to-end network and service topology, as well as knowledge of the optimization engines available at each segment and their required inputs. These inputs, as provided by the UMF through policies, will drive the optimization engine behaviour.

The inputs that can be set through policies by the UMF in our approach are:

1. the time-frequency of the optimization; the optimization is re-triggered whenever a new traffic matrix is provided as input
2. the maximum link utilisation (MLU) that needs to be enforced; setting a maximum link utilisation target can help to achieve delay and load-balancing objectives and also ensure that in case of link failures it is easier that the affected traffic can be redirected without causing severe congestion to the remaining links.
3. the energy efficiency target, that is the number of links that should be switched off

As it will be shown in the next section, given these two policy-driven objectives the optimization mechanism attempts to switch off as many links as possible ensuring though that the remaining network topology is fully connected. In case the energy efficiency target cannot be met, i.e. not as many links as required can be switched off, the UMF should be notified about this and consider remedial actions in the form of "relaxing" its MLU objective –if this can be compensated through actions in the other segments- in favour of the energy efficiency objective.

3.4.2.2.2 Greedy Algorithm for Link Selection for Sleeping

At this stage our target is to design an efficient link removal approach according to one single traffic matrix. A greedy algorithm has been initially designed as shown in Figure 34. From this starting point, we will be able to make further extensions with "oblivious" techniques such that the enhanced algorithm will be able to compute a robust topology configuration for multiple traffic matrices during the off-peak time. More specifically, the resulting topology is able to (1) achieve optimized energy efficiency performance during the overall off-peak time on daily basis, and (2) avoid potential network congestions due to the traffic dynamics during the period (e.g. traffic upsurge when the reduced network topology is applied). Towards this goal, sophisticated consideration on the trade-off between energy efficiency target and network robustness should be taken. Initial research on robust network configurations has pointed to [8] where a Mixed Integer Non Linear Programming (MINLP) solver is applied to find a network topology which will equally satisfy several traffic matrices for inter-domain traffic engineering.

Another interesting issue is how to determine the off-peak time duration. A simple scenario is for the network administrator to regulate the start/end of the off-peak time, for instance between 7PM and 7AM on daily basis. Based on such a pre-determined duration, the corresponding calculation of the reduced network topology with link removals will become relatively simple – the only concern is that the topology should satisfy all the traffic matrices falling into that *pre-determined* duration. However, it is easy to infer that the overall energy-saving effect might not be comprehensively optimized since the determination of the off-peak duration does not take into account the traffic dynamics pattern. For instance, a less-reduced network topology configuration coupled with a longer off-peak window size might yield better energy saving performance than a completely greedy configuration which can be applied only at a short duration. As such, a more advanced approach is to apply a "joint" optimization mechanism including the determination of *both* link removal *and* the window size duration of the off-peak period in order to achieve global optimized configurations during daily network operations.

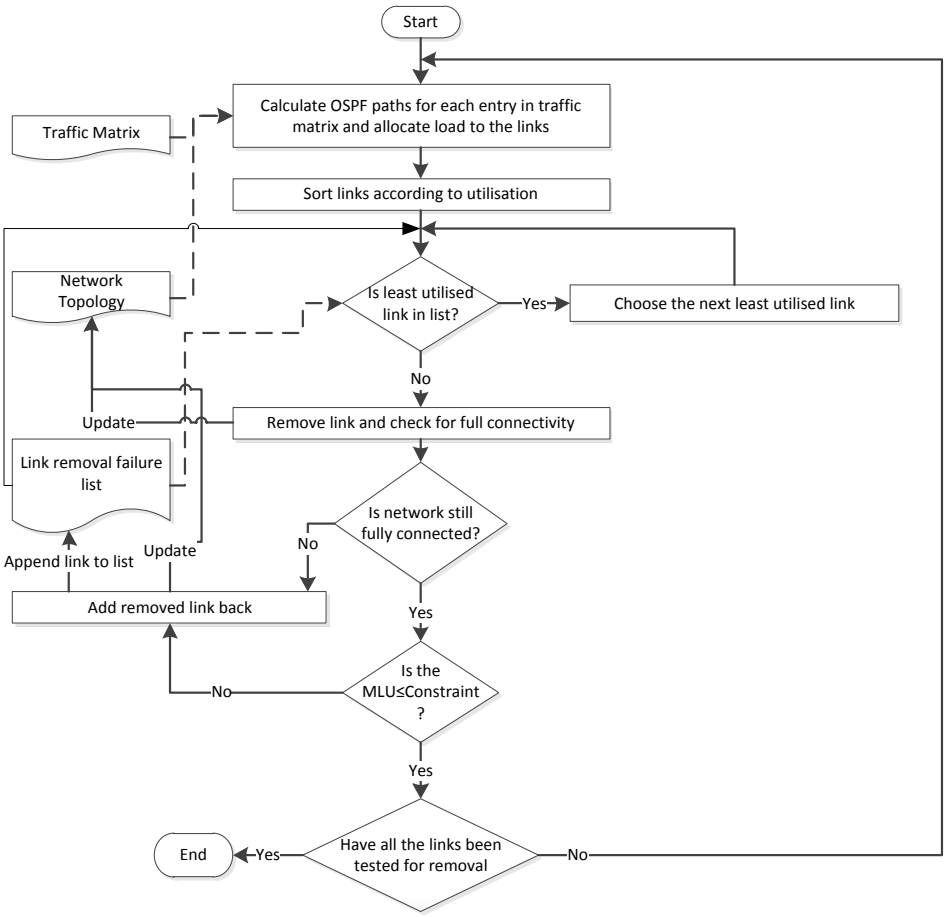


Figure 34: Flowchart for greedy algorithm

3.4.3 Governance of Self-Organizing Network (SON) functionalities through operator policies

Governance of SON mechanisms allows the operator to have automatic pre-defined/controlled reactions of SON mechanisms in accordance with high-level operator goals. The operator can also determine in an automated manner which SON(s) to activate on certain (series of) event(s) and with what parameter settings. To this end, a SON coordination mechanism that realizes the necessary actions has to be put in place as an interface between the SON governance and the low-level individual SON mechanisms. Each SON functionality periodically transmits to the SON coordinator the information needed to construct the *model*, i.e. the relationship between the optimized metrics and the tuned parameters. The SON coordinator constructs the *overall system model* (relationships between all optimized metrics and tuned parameters) empirically and also using the *a priori* (i.e. expert) knowledge on the behaviour of network metrics with respect to the network parameters.

The SON coordinator transmits the *overall system model* to the SON governance mechanism that also takes as input the high-level operator goal(s) from the human network administrator. By using the *overall system model* and the high-level operator goal(s), the SON governance mechanism then determines the optimum actions according to the pre-defined operator policy.

If based on a utility function policy (as utility function policies introduced in the related work section), the SON governance mechanism knows the *high-level utility* as a function of the *system state* that comprises the optimized metrics together with some high-level metrics that cannot be directly linked to the SON metrics (like cell-edge user performance, user satisfaction, fairness etc.). Using the *overall system model*, the SON governance mechanism calculates the parameter-level utility. The aim of the SON governance mechanism is to find the optimum set of SON (output) parameters that maximizes this high-level utility.

The SON governance mechanism is not operational all the time, but it intervenes when the high-level utility falls below a certain threshold. If the *overall system model* as well as the model for the high-level metrics is

known with sufficient precision, then the SON governance mechanism computes the optimum set of SON parameters and transmits the optimum parameter values to all SONs.

On the other hand, if the overall system model or the model for the high-level metrics is not known with sufficient precision, then it may be preferable to use action policies, since the automated management needs the accurate models that link the high-level goals to the individual SON metrics in order to come up with the optimum actions. The possible actions that the SON governance mechanism may instruct to the SONs can be triggering ON/OFF for a pre-defined period, modifying the permissible ranges of the parameters/metrics and modifying the objective function of the SON. The optimum action vector at time instant *t* is a function of the system state at *t* and involves the operator policies. However, in the action policy case, the optimum actions are not automatically determined by the SON governance mechanism. They are manually entered by the human administrator for each particular system state.

The functioning of the governance of SON functionalities using both action and utility function policies are depicted in the block diagram of Figure 35

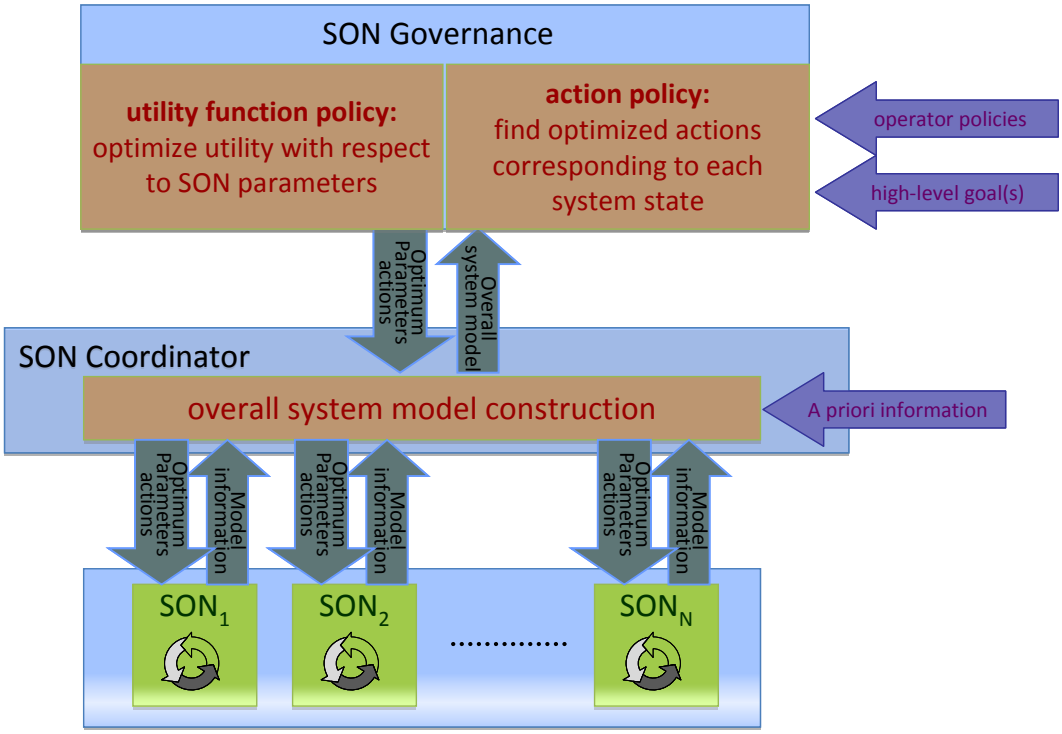


Figure 35: Governance of SON functionalities using action and utility function policies

3.5 Discussion

This chapter is based on the ascertainment that an autonomic deployment and/or traffic accommodation of a new service by a network operator on top of multi-vendor and multi-technology (IP/MPLS and OFDM) infrastructures would require fully coordinated performance across the whole network; in particular, this entails the proper formulation and solution of parameter optimization problems in both RAN and core segments, as well as the investigation of the way policies can be used as a means for "governing" this optimization by imposing high level goals that are capable of efficiently controlling and coordinating the performance of the whole network independently of the network segments.

To this effect, the chapter presents, formulates and solves policy-based parameter optimization problems applying to both RAN and core segments. More specifically, policy-based OFDM resource allocation and multi-hop link selection are presented for the RAN case. For the backhaul/core case the focus is placed on policy-based energy and load balancing-aware traffic engineering mechanisms. The use of operator policies for the governance of SON related functionalities is also discussed.

The policy-based optimization problems populate specific Functional Blocks (FBs) of the UMF. In particular, the responsibility of solving the problems addressed herewith, referring to the RAN and Backhaul/Core invocation phases, is undertaken by the so-called "Solution Selection and Elaboration" functional block of the UMF as per Release 1 of the UMF Specifications (see also D2.1). The joint end-to-end coordination of the RAN and core segments refers to cooperation strategies and as such, it is to be tackled within Task 3.4. The work in this chapter also strongly relates to both the "Governance" and the "Policy Derivation and Management" FBs, namely in terms of the way that policies are used to affect the optimization problems and the way that these policies are affected (properly modified) as an outcome of the results of the latter.

In the future, the proposed methodologies used to drive the optimization engines for the problems tackled in this chapter will be further validated and additional methodologies will be considered. The aim in doing so is to find the best-suited method for each problem addressed; in this context criteria for suitability will be not only the ability of the methodology to optimize against the specific objectives for given network and service scenarios but also its ability to be robust to changes in the network and service conditions. In other words be as autonomic -and applicable to as many scenarios- as possible minimizing the need for manual intervention. Further work will be on issues related to the cooperation between core and RAN segments through the policies so that fully coordinated end-to-end performance across the whole network may be achieved. The consideration of this work in the context of assurance phase during which the focus shall be also placed on more real time/online handling, i.e. monitoring, admission control etc., is also part of further work. Finally, more elaboration on the policies in conjunction with the work conducted in WP2/Task 2.3 with respect to governance and policy framework is going to take place in the future work related to this chapter.

4 Load balancing

4.1 Introduction

Load balancing is a principal mechanism for resource management, designed to distribute workload and resources between different network entities in a balanced way. The exact definition of the term “balanced” depends on resource types, trigger types and network domains. For example, objectives for a load balancing algorithm may be the optimized usage of services, network resources or control-plane data with respect to the given constraints and utility functions. The load balancing functionalities can be triggered by congestion, resource or computational shortage, QoS and fairness control, mobility and energy-efficiency management, etc. Mechanisms operating across different network domains are also needed. These include, e.g., multi-access load balancing (i.e. across networks), intra-access load balancing (e.g. across cells), intra-backhaul load balancing (e.g. traffic engineering, routing, in-network caching) and inter-gateway load balancing (e.g. across service gateways). The operational time scales of different load balancing solutions vary a lot. The offline traffic engineering mechanisms, like re-configuration of label switching paths (LSPs) and overall network capacity planning have operational times from minutes up to years, whereas online mechanisms should be able to react with dynamics of order of seconds and much less [33].

The UNIVERSELF research on load balancing is strongly driven by the use cases (UCs). In principle, all UCs contain load balancing related issues; many of these have already been considered in the previous chapters. These include, e.g., evolutionary algorithms for load balancing problems in Section 2.3.1 and load balancing applied in OSPF-based traffic engineering approach in Section 3.4.2.1.1. In this chapter, we consider UC3 (Section 4.5), UC4 (Sections 4.4, 4.6, 4.7, and 4.8) and UC6 (Section 4.9). The studied load balancing scenarios can be categorized into two classes: solutions for wireless access and solutions for core networks. Note that in a later phase of the UNIVERSELF project, algorithms for end-to-end load balancing as well as co-operation strategies will be developed. This specifically includes virtualization techniques for balanced allocation of resources in the wireless access and backhaul.

4.2 Solution space

Within the context of load balancing, the partners will contribute novel/autonomous strategies and solutions that will address load balancing and resource management from different perspectives while focusing on different network domains (i.e., access, core, backhaul, service etc.) of a common reference network framework. The overall load balancing solution space will cover the following approaches:

- Intra RAT handover and/or interference coordination in the access domain
- Inter/Intra RAT handover with reference to content migration in the core and backhaul domain
- Prediction methods for attaining user satisfaction in the service domain
- Selective (de-)activation of access points in the access domain
- Centralized energy-aware capacity-optimized traffic engineering

With respect to the wireless access in the LTE, detailed investigation will be carried out to determine and compare the feasibility of changing the handover region or perform inter-cell interference coordination for load balancing.

Within the backhaul and core domains, a load balancing strategy will be developed in view of resource management for managing large volumes of mobile content data. This will be achieved by developing a decision making framework comprising of decision modules, flexible expert systems and triggering mechanisms to be applied for balancing load at both the network level and service level. The proposed framework will utilise the Network Expert System (NES) proposed in [35] and combine it with an event collection and distribution system proposed in [36] and [37]. The developed system will make use of Self Organizing Maps (SOM) techniques to classify the load levels of APs and then trigger decision paradigms.

As a proactive strategy, prediction models will also be developed to accurately predict future loads and trigger load balancing on time. The load will be balanced by either performing a handover or carrying out protocol reconfiguration. The work will extend the previous work [41] by developing prediction models that will predict future values of a user satisfaction metric.

Another approach will take into account (de-)activation of selected (or group of) APs/BSs in a self-aware cluster for capacity and radio resource optimization while maintaining continuous coverage service. This work will build upon the previous work of coverage optimization undertaken in [45], [46] and [47] by developing a self-aware Domain Manager. The Domain Manager will create a Local physical Topology Graph (LTG) and Cluster Topology Graphs (CTG) by taking into account the monitoring data and the operational status of member APs as well as terminals. Based on this the Domain Manager will make decisions regarding (de-)activation of APs and shifting load from deactivated APs to activated ones.

In the access domain, the implication of failed BSs on the overall network load will also be taken into account. The goal is to autonomously handle network failures by self-configuring/healing and self-optimizing network resources. One approach will be to efficiently and autonomously compensate the cell outage by adapting transmission powers, with minimum interference, of neighbouring BSs and shifting the affected mobile users along with their load to them. The work will be based on the framework proposed in [48] but enriched with a richer set of parameters and self-x algorithms.

Last but not least, energy-aware traffic engineering for load balancing will also be addressed within the load balancing solution space. Recently, routing, rate adaptation and network control are mobilized towards energy-efficient network operation [26][27]. Unfortunately, none of these approaches provide a general problem formulation in the direction of “coupling” the traditional traffic engineering objectives with the modern objectives (like energy-awareness). This work will attempt to “modernize” the research in this field. This work will be based on and extend [25], which discusses the idea of dynamically turning part of the network operations into sleeping mode, during light utilisation periods, in order to minimize the energy consumption. In this context, a challenging task will be to identify the main parts of the Internet that dominate its power consumption and investigate methods for improving energy consumption [24].

All the methods/algorithms that will be developed within the scope of balancing load and resource management will be undertaken within the operational and functional scope of SON and will be fully coupled with the UMF at a later stage.

The rest of this chapter is organized as follows. The overall UNIVERSELF load balancing framework in case of mobile networks is sketched in Section 4.4 focusing on the downlink transmission of the intra-LTE intra-frequency radio access. It indicates that a potential gain of more than 20% is possible and suggests several strategies how to collect these gains in practice. Section 4.5 describes a self-organizing map based expert system which is applied to load balancing in wireless access. The solution is also implemented as a proof-of-concept prototype. User satisfaction, a metric defined by response times, is used as a trigger for load balancing functions in Section 4.6. The developed load prediction model allows predicting future values of user satisfaction and thus enables the proactive management of load balancing requests. Section 4.7 studies the dynamic deactivation and/or re-activation of access points and the respective load balancing phase that follows those. Section 4.8 develops solutions to recover from a base station failure event in a wireless network. The main goal is to find a balance between increased transmission ranges and interference resulting from the need to increase transmission power nearby the failed base station. Finally, Section 4.9 proposes a distributed Energy-Aware Traffic Engineering scheme that provides load balancing and energy-awareness in accordance with the operator’s needs. The proposed scheme is “governed” by a low-complexity heuristic algorithm that is executed in an autonomous manner, using monitoring of the status of the network and making automatic decisions

4.3 Load balancing framework

As indicated above, load balancing can involve different domains of the network. Research work has been performed in various areas. Figure 36 shows an integrated overview, which spans all operator domains from the service domain all the way to the user device. For each domain, the left-hand side of the figure shows which context information can be obtained in order to collect solid information based on which load balancing decisions can be taken. The right-hand side shows which load balancing actions can actually be performed in the different domains. The depicted network topology is of course only an example. It was chosen so as to include multiple radio access technologies (RATs), since inter-RAT handover is one of the considered load balancing options in our work.

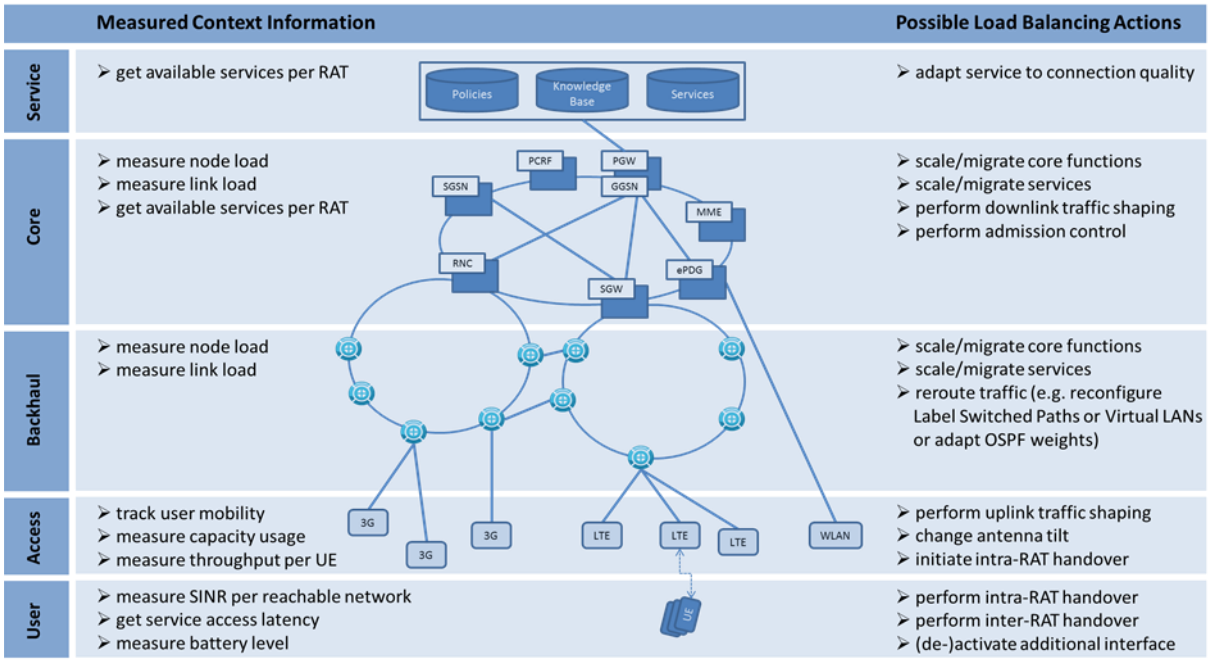


Figure 36: Overall (end-to-end) load balancing view.

As can be seen from the figure, there are many possible actions that could be taken to alleviate load imbalance situations. In many cases, a combination of different actions will be the most appropriate or effective reaction to imbalances. However, the precise sequence in which different actions are taken, and the question which load balancing action is preferred over which other one, completely depends on the preferences of each individual operator. Some operators might consider codec adaptation less intrusive than reconfiguration of Label Switched Paths (LSPs) in the backhaul, for some operators, inter-RAT handovers will not be an option since they might not own multiple RATs themselves. The following sections will present individual load balancing strategies which touch on the one or other context element, operator domain or possible remediation action included in Figure 36. Specifically, the subsequent sections are related to Figure 36 as follows:

- Section 4.4 covers intra-RAT handover and interference coordination, i.e. load balancing actions in the access domain
- Section 4.5 deals with both intra-RAT and inter-RAT handover, but the work discussed there will later be extended to content migration, i.e. load balancing in the core and backhaul domain
- Section 4.6 introduces prediction methods for user satisfaction with the goal of guiding the two considered load balancing strategies handover and protocol reconfiguration, i.e. access and service domain
- Section 4.7 considers access point (de-)activation and the resulting handovers as a means for load balancing in the access domain
- Section 4.8 describes transmission power adaptation of base stations after base station failures, which effectively implies intra-RAT handover, i.e. load balancing in the access domain
- Section 4.9 presents centralized energy-aware capacity-optimized traffic engineering, i.e. (re-) configuration of LSPs and thus load balancing in the backhaul and core domain

4.4 LTE wireless access

This section focuses on the wireless access domain. In particular, we consider the case where cells are in overload where overload refers to a situation in which the number of physical resource blocks is not sufficient for the traffic. Load balancing in this context refers to changing parameters in the overloaded cell (and potentially its neighbour(s)) such that the overloaded cell is either relieved by some terminals or alternatively by interference. More specifically, we investigate the role of changing handover parameters in order to change

the handover region for the purpose of load balancing and alternatively we investigate the role of inter-cell interference coordination for the purpose of load balancing. We then compare these two strategies with each other and show that a combination of both strategies comes along with the highest benefits.

In the following we elaborate on the wireless access of a 3GPP LTE network and its terminus with an emphasis on downlink communication. Then attention is turned to the operator's view, which basically is to balance user satisfaction with the cost of operating the network. Load balancing can increase the overall throughput in the radio access network. This statement is supported with an excerpt of simulation results from a preceding study. The remainder of this section identifies strategies for load balancing and summarises possible solutions. The work presented in the following essentially builds on [34].

The LTE wireless access network consists of base stations, which are named enhanced node B (eNB) and which are carefully placed in its coverage area. To access this network, the user employs a terminal device called user equipment (UE). Between any UE and any eNB there is a radio channel, which can be seen as an end-to-end communication path for signals. Signals conveyed through these channels experience a path loss L in dB, which is approximated by $L=128.1+37.6 \log_{10}(d)$, where d is the distance in kilometres. This formula is explicitly given here, to illustrate the high dependency of path loss on the distance to the eNB.

If a user decides to communicate, he sets up a call. As part of this procedure, an eNB in its vicinity is selected as serving eNB and transport of data from the network to the user is achieved by transmitting data via this channel. Multiple transmissions may be executed by the different eNB of the network in parallel. The channel's quality is defined by thermal noise and the signals received at the mobile. A typical measure for its quality is the Signal to Interference plus Noise Ratio (SINR), which can also be used to derive the channel capacity.

To achieve a certain data rate, bandwidth has to be allocated to a transmission. An LTE system maintains bandwidth in units of so-called Physical Resource Blocks (PRB) of 180 kHz each; a common 10 MHz LTE deployment commands 50 of these. The channel quality in the coverage area varies considerably. Therefore the number of PRBs which are required to provide a defined data rate to a user depends on his position.

Each eNB forms a radio cell with a certain coverage area. Caused by the mobility of the users, the serving eNB of a UE has to be changed to another eNB, if it leaves the coverage area and enters into a new cell. Changing the serving eNB is known as handover.

4.4.1 Network operation

Network operators have to find an efficient balance between cost and user satisfaction. The former means that the number of eNB locations is kept low, whereas the latter comprises quite complex considerations. A simple approach to maintain fairness is to attach each UE to the eNB providing the strongest signal and to allocate each user an equal share of the PRB of the associated radio cell. Under the further assumption, that all users have infinite data to transmit, this approach leads to a fair network.

A more realistic view of fairness is to guarantee each active user a minimum data rate. Of course, the network may allocate a higher data rate to each user, but this may be reduced to the guaranteed rate to ensure efficient network operation.

One consequence of this approach is that a call setup may be blocked due to the fact, that there are no PRB available to handle it. Also, due to fluctuations in channel capacity caused by user mobility, and due to handovers, the cell may run out of PRB to serve all UE and may be forced to drop the call of a mobile. From the user's perspective, these call drops are considered even more dissatisfactory than the call blocks. It is the task of the Call and Admission Control (CAC) to prevent these situations. Yet CAC also offers a way to increase the overall user satisfaction by dropping individual UE. This approach releases PRB, which can be used to keep other users satisfied.

Another consequence of this approach is that there may be partially loaded radio cells. The network can draw an advantage from this fact by either handing over UE from a highly loaded to a lower loaded cell or by using the PRB in the low loaded cell in a way that avoids interference in the highly loaded cell and thereby increases the resource efficiency therein.

The first approach, i.e. selecting another serving eNB than the one providing the highest signal, leads to high interference for the shifted mobile and therefore suffers from decreased resource efficiency. Yet, if this is done carefully, the total number of users in the network may be increased. The second approach, i.e. coordination of interference, leads to increased resource efficiency and therefore it is promising especially in situations of a high overall load in the network. Also the combination of both approaches is promising.

4.4.2 Simulation results

The diagrams below show the results of a simulation experiment executed to find out the potential gain of the respective approaches. In a network with a hexagonal cell layout, this experiment focuses on seven cells. In a centre cell and in the six surrounding other cells, the load is controlled separately. It is then observed how much load can be carried by this sub-network. The experiment fixes the load in the other cells and then increases the load in the centre cell as long as the sub-network carries more than 97% of the offered load. The gain shown in the diagrams is the relative change of throughput for a specific approach in comparison to not doing any load balancing at all. The two diagrams are different with respect to the user density at the centre cell's edge, which is almost even for Figure 37, while in Figure 38 the user density is configured so that 50% of all users have a SINR of 0 dB or less.

Figure 37 and Figure 38 show the results of these experiments. The curve labelled "ic" shows the gain of the interference coordination approach, the curve labelled "ho" shows the gain of the approach to select another than the eNB providing the best signal as a serving eNB and the curve "ic+ho" shows the gain of the combined approaches. The individual approaches of "ic" and "ho" show a contrary behaviour over the other cell load; "ho" pays off in low loaded environments, while "ic" pays off in highly loaded environments. The most interesting observation is that an appreciable gain of 13% and more for the combined scheme "ic+ho" can be observed over the entire range of other cell load. In addition, a higher user density at the centre cell's edge leads to even higher gains for the "ho" and "ic+ho" approaches. In all situations, the gain of the combined approach is better than the gain of any of the individual approaches alone.

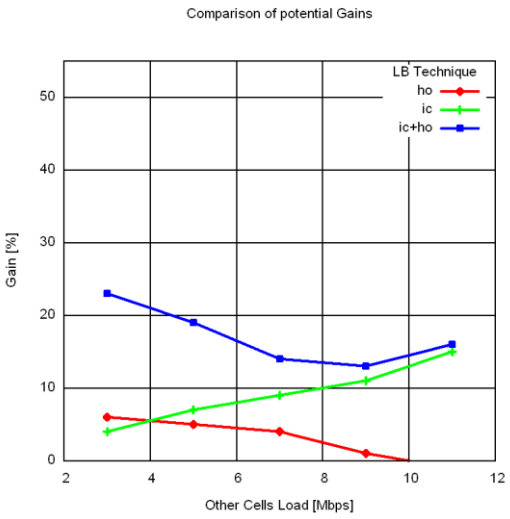


Figure 37: Potential gain of load balancing.

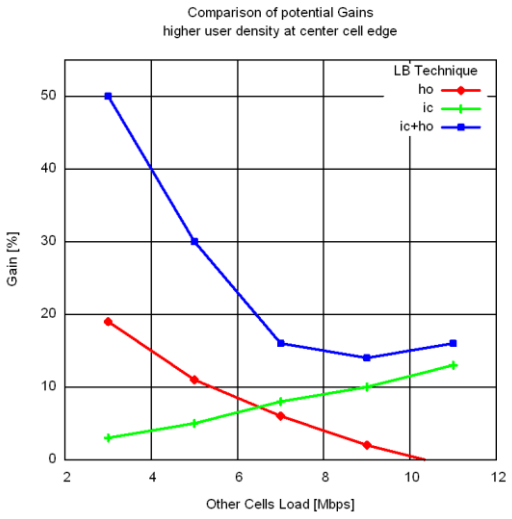


Figure 38: Potential gain of load balancing for an increased user density at the centre cell edge.

4.4.3 Methods for parameter optimization

The aim of load balancing is to increase the total number of satisfied users in the LTE radio access network. So far three strategies have been identified to deal with load balancing in the mobile access:

- Handing over mobiles to a neighbour cell (change of serving eNB).
- Increasing PRB efficiency by interference coordination (network wide coordination of PRB assignment to simultaneous transmissions).
- Dropping of individual users (releases PRB which can be used to keep other users satisfied – this is an absolute emergency action).

Solutions to draw the maximum advantage of these strategies are:

- Brute force calculation of all possible combinations (which is not feasible due to the computing effort).
- Also based on the brute force approach, yet the solution space is reduced by excluding obvious non candidates.
- Hypothetical calculation obtained from measurements of UE. E.g. all eNB providing a signal below are certain threshold of the maximum received signal can be safely ignored, because they provide an SINR at which data transmission is impossible.
- Introduction of fuzzy states and rule based state transitions. This may be complemented with a self-learning approach.

In this section the network operator's basic view on a wireless network has been identified – a high number of satisfied users at reasonable cost of network operation. Simulation results have been provided which clearly show, that potential gains of more than 20% can be achieved by load balancing in the context of LTE radio access networks. Different strategies to balance the load in the LTE radio access in downlink transmission direction have been listed and solutions have been proposed how to find the optimum operating point. Potentially, this enables a network operator to serve more users at the same quality of service w/o additional investment.

4.5 Expert system for hand-over decisions

Content delivery is one of the main applications of the current and future networks. The ever increasing number of wireless devices moving large amount data over the networks requires novel resource management solutions. These should support distributed decision making as well as co-operation between different load balancing mechanisms, e.g., operating in service and network layers. A step towards this is development of a decision making framework where decision modules, flexible expert systems and triggering mechanisms are combined. This approach is first applied into network level load balancing. Analogous solutions will be used in service level load balancing in future work.

In this section, we present a network expert solution for handover decisions for wireless access. Specifically, we describe of a proof of concept prototype where this solution is tested in an overload scenario. In the prototype, we currently utilize a Network Expert System (NES) [35] together with an event collection and distribution system [36] [37] to achieve load balancing in the access network. The decisions are based on the classification of the access point loads by self-organizing maps (SOM). Depending on the outcome of the classification, different actions are triggered, e.g. send a congestion trigger, suggest to handover, force to handover, etc. In addition to SOM, fuzzy classifiers are going to be implemented and tested in the prototype. Performance and scalability of the different classification approaches will be evaluated through simulations later.

The overall scenario is closely related to Use Case 3. The system consists of wireless devices equipped with multiples interfaces, which can be used at the same time and may support different technologies (WLAN, 3G and LTE). The devices run a content delivery service, which in this case is a multi-access capable BitTorrent supporting localization of data [38] [39]. The ultimate goal is an end-to-end load balancing solution, where the load balancing decisions are made co-operatively both at the network and at the service level. Additionally, each end user tries to optimize his/her QoE. In this report, we focus only at the network selection from the access point perspective; see Figure 39. Later on decision mechanisms for content migration will be developed. Cooperation between these different level decision mechanisms will be addressed in D3.4.

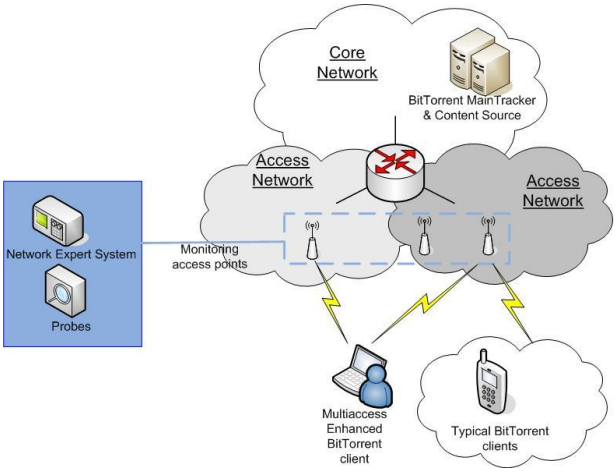


Figure 39: Multi-access prototype scenario.

NES is a SOM expert system, which operates according to defined rules and policies. Probes are monitoring the APs and sending the information to NES. In order to detect congestion, current available bandwidth, radio interface quality and packet loss are monitored. In a future work also the number of clients connected to each access points (APs) is going to be considered. Before using NES in the actual decision making, it needs to be trained to classify different states and map these to the congestion levels. After the training phase, NES can compare the information received from the probes with its knowledge base and generate triggers indicating the state of the AP. The possible AP states are shown in Table 6.

Table 6: Access point states.

Access point state					
Low traffic	Medium traffic	High traffic	Almost congested	Congested	Badly congested

The decision modules receive the AP status updates from NES. Since the NES implementation is memory less, each decision module keeps a history of the previous states to avoid unnecessary handovers. Decisions can be made centralized or in each AP separately. Based on the information received from NES, the decision mechanism can recommend or force a handover. The decision may also depend on the states of neighbouring APs if this information is available. In case of congestion, choosing randomly devices to force/suggest to handover is the easiest solution, but also more intelligent approaches can be implemented.

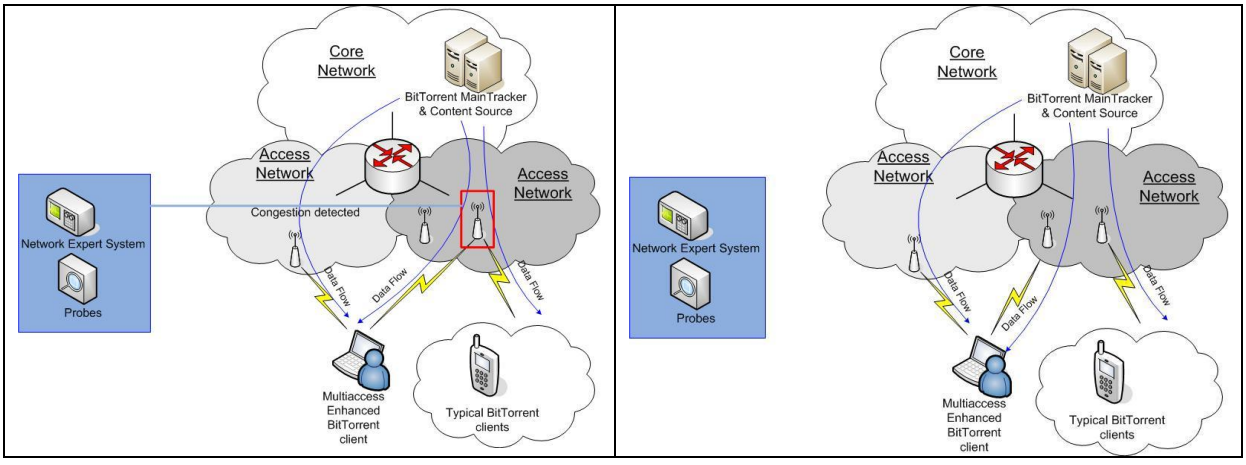


Figure 40: Congestion detection and handover.

An example of the realized scenarios in the prototype is shown in Figure 40. The left panel shows NES monitoring an AP which is congested (denoted by a red box). NES generates triggers indicating the congestion of the AP as well as the states of the neighbouring APs in the same wireless segment. In the current implementation, the decisions are made centralized and the decision module sends the control messages directly to a selected client. Another option would send the triggers to the APs, which would individually decide when and which clients to drop. The right panel of Figure 40 shows the wireless connections after the multi-access enhanced client has received a handover suggestion/order together with the state information of neighbouring. In this example the client connects to a neighbouring non-congested AP. Note that even if the control message was "force to handover", the multi-access client would also have the possibility to close the interface and continue receiving the data through another existing wireless link. E.g. if the client tries to optimize energy usage and none of neighbouring APs is in low traffic state, such a decision would make sense.

As a summary, we have developed a network expert based solution for load balancing in wireless access. The current implementation is using SOM for event classification. Fuzzy logic, possible enhanced with learning mechanisms, and service-level load balancing will be considered in the next phases of the work.

4.6 Resource management – control plane

The dynamic management of Future Networks requires the modelling, incorporation and integration of suitable mechanisms and algorithms for the network decision making process. An important part of such procedure is the end-users' load-balancing, which is related to their decision-making requests (control plane data). Such load balancing algorithms specify decisions based on the present load conditions of the system computational resources. The possibility of predicting future loads is also very important. This will enable to proactively manage the system resources, triggering load balancing mechanisms on time [40].

Related work on the resource management of control data include a) the modelling of the decision-making procedure and b) the load-balancing actions. As regards the first challenge, the work by Balbo and Serazzi [44] and Litoiu [43], [42] uses multiclass queuing networks for the system model for the derivation of the network bottlenecks and the bounds of the response time under asymptotic and non-asymptotic conditions. Besides them, several approaches have been addressed for the development of approximation techniques to estimate performance measures such as queue lengths, sojourn times and throughput [55]. In the aforementioned approaches the implications of reconfiguration decisions at network level are not addressed. Moreover, the system bounds for each framework are not discussed in consideration of the overall load and reconfiguration overhead in conjunction with the user and device classes and respective request patterns. Concerning the second challenge, our work leverages on the prediction models proposed in [40], to realise the prediction of future values of the network response time and consequently the user satisfaction metric. To this end, we propose an innovative prediction-based load balancing scheme which allows for the proactive management of the saturation in terms of handling the decision-making requests. In conclusion, the introduction and adaptation of existing methods in order to discuss optimization issues in autonomic and reconfigurable telecommunication systems has not been considerably investigated in the literature and forms one of the key directions of this work.

The work presented here specifically deals with an appropriate model of the network decision making process for mobile devices adaptation. Two main adaptation alternatives are assumed: handover and protocol reconfiguration. We consider two classes of mobile devices: reconfigurable and autonomous; the difference between them lies on the degree they support decision making functionality. An algorithmic framework for the management of the decision making requests for reconfiguration or handovers is proposed. This work is based on the introduced metric of user satisfaction, which is based on the network response time for serving the decision making requests. Such a framework is important for guiding the load balancing/relocation of mobile terminals so as to achieve offloading, based on the values of the user satisfaction. This framework, which is based on previous work [41], is extended with load prediction models that allow to predict future values of user satisfaction. More specifically, we consider the load-prediction models applied in Web-based systems [40]. Such models are not based directly on resource measures but on the representation of the load behaviour of system resources. Such models ensure that not only a limited view of the resources is provided but also a view of the behavioural trend. The predicted values are used to proactively trigger the load balancing of the decision-making to avoid the saturation of the computational resources, based on the predicted value of the user satisfaction. At this point it should be noted that this work is mainly related to Use Case 4: SON and SON collaboration according to operator policies.

4.6.1 Algorithmic framework

This section proposes an algorithmic framework for the management of decision-making requests in a system. The key concept addressed is the dynamic computation of the system capacity in terms of computational resources and the management of the requests that exceed the system capacity. In this work, the system capacity is defined as the number of simultaneous decision-making requests that can be handled by the network; it should be noted that the system capacity is computed separately per class of mobile devices as there are different response time requirements per class [41].

The system capacity is affected by both the number and frequency of the requests and dynamically changes according to these parameters. In order to be able to capture these requirements and dynamically compute if the system capacity is exceeded, we need to introduce a metric that will enable the comparison between different allocations of the decision-making requests. In this direction, based on similar concepts in [42], we introduce the user satisfaction: a metric of the satisfaction of the user based on the network response time for serving the decision making requests. High values of the user satisfaction imply fast serving rate of the decision-making requests by the network whereas low values of the user satisfaction imply slow serving rate of the decision making requests and that the system capacity is close to its limits. At this point we should state that out-of-band control traffic is considered in the network response time since we consider a reconfiguration network service that is related only to the control traffic caused by decision-making requests.

At first the user satisfaction degree is dynamically computed during real-time based on network response time measurements, per class of mobile device. Next, we define the user satisfaction threshold: a threshold for the lowest possible value of the user satisfaction. If the user satisfaction is found to be lower than this threshold, then the requests reallocation procedure is triggered. In this work, we consider that the requests reallocation is applied in the neighbouring network nodes that handle similar requests. The requests reallocation is realized based on the user satisfaction metric. Specifically, once a node triggers the reallocation procedure, the satisfaction metric for the neighbouring nodes is computed. Next, a negotiation procedure is triggered for the selection of the appropriate neighbouring nodes to participate in the reallocation procedure. In particular, the concept of the reallocation threshold is defined: the minimum acceptable value of the node' user satisfaction that allows its participation in the reallocation procedure. Once the neighbouring nodes are selected, then the mobile devices reallocation is applied. More specifically, we define as mean reallocation satisfaction the mean value of the degree of satisfaction of the participating nodes in the reallocation procedure. The nodes with lower value than the mean reallocation satisfaction should allocate a percentage of the serving mobile devices to the nodes with higher user satisfaction than the mean reallocation satisfaction. The actual percentage can be either fixed (static) or can be computed in a dynamic manner targeting the fair request reallocation according to the user satisfaction of the nodes.

Such framework is extended with load prediction models which enable the prediction of future values of the network response time and consequently the user satisfaction metric. To this end, we employ a two-step approach. Initially the representation of the resource load conditions l_i at time t_i is computed based on the measured raw data of the network response time – this is realised by the load-tracker module. Next, the future values of the network response time are forecasted based on a set of load tracker values for the network response time. This is realised by the load prediction module [40].

The proposed scheme enables the management of decision-making requests coming from both reconfigurable and autonomous mobile devices - the difference between them lies on the degree they support decision making functionality. The presented mechanism is to be integrated within the network nodes handling the decision-making requests and is fully autonomous. The network operator should manage the scheduling and the realisation of the resource measurements as well as the definition of the required thresholds for the implementation of the decision making procedure (e.g. user satisfaction threshold, reallocation threshold).

4.6.2 Modelling and computing user satisfaction: the trigger metric for load-balancing

Based on the proposed algorithmic framework, we need to define the user satisfaction metric as a function of the network response time. As analysed in the previous section, user satisfaction is the key metric for load balancing: if the computed or the predicted value of the user satisfaction is above the user satisfaction threshold, then the load balancing procedure is triggered. First of all, we define as network response time the response time experienced by a mobile device making a decision-making request to the network side. We differentiate the network response time per class of mobile devices. Therefore we define the response time of

class c R_c as the response time experienced by a class c mobile device making a decision-making request. We also define as user satisfaction SA_c, the normalized distance of the network response time R_c from the maximum value of the response time $R_c^{\max}$ to the interval of the maximum response time minus the minimum response time $R_c^{\min}$. Therefore, user satisfaction is analysed as follows:

$$SA_c = \frac{R_c^{\max} - R_c}{R_c^{\max} - R_c^{\min}}$$

At this point it should be noted that the maximum and minimum values of the response time are not static but dynamically varying based on the number of the decision-making requests and the average time between requests. For a given system, the response time and the respective maximum and minimum values of the response time can be computed using mean value analysis (MVA), an iterative technique for the analysis of closed queuing network models, which has very high computational complexity. Therefore, instead of computing the user satisfaction, we compute an approximate value of the user satisfaction. This is realized by computing the bounds of the maximum and minimum response time.

Therefore, the approximate value of user satisfaction is given below:

$$\overline{SA_c} = \frac{R_c^{Up} - R_c^{Ms}}{R_c^{Up} - R_c^{L}}$$

In this direction, the computation of user satisfaction requires to also compute the upper and lower bounds of the network response time and measure the network response time. To realize the bounds computation, we consider the analysis by Litoiu and Balbo, Serrazzi [43] [44], and propose a methodology and respective analytical model for the computation of the approximate value of the user satisfaction. Such methodology concerns the bounds computation for the response time for distributed systems with multiple resources and workload mixes.

As regards the load prediction, we consider the models presented in [40]. More specifically, as regards the load tracker, both linear and non-linear load trackers are available. We consider the samples of the network response time s_i at time t_i, and a set of collected n measures denoted as $\overrightarrow{S_n}(t_i) = (s_{i-1},.., s_i)$. In addition, the load tracker is denoted as a function $LT(\overrightarrow{S_n}(t_i)) : \Re^n \rightarrow \Re$ that takes as inputs $\overrightarrow{S_n}(t_i)$ and gives a representation of the resource load conditions l_i at time t_i [40].

In this work we employ we Exponential Moving Average (EMA) load tracker, which is the weighted mean of the n resource measures of the set $\overrightarrow{S_n}(t_i)$, where the weights decrease exponentially. The EMA-based load tracker $LT(\overrightarrow{S_n}(t_i))$, for each time t_i where $i > n$, is equal to

$$\mathrm{EMA}(\overrightarrow{S_n}(t_i)) = \alpha * s_i + (1-\alpha) * \mathrm{EMA}(\overrightarrow{S_n}(t_{i-1}))$$

where the constant $\alpha = \frac{2}{n+1}$ is the smoothing factor.

In addition, we denote the load prediction as a function $LP_k(\overrightarrow{L_q}(t_i)) : \Re^q \rightarrow \Re$ which takes as input the set of q values $\overrightarrow{L_q}(t_i) = (l_{i-1},.., l_i)$ and returns the predicted value at time t_{i+k}, where k>0. In this work, we consider the load predictor employed in [40], which is based on the linear regression of two available load tracker values. Each predictor is characterized by the following set of values:

- The predicted window k which represents the size of the prediction interval
- The past time window q, where q is the distance between the first l_{i-q} and the last l_i load tracker value

The load predictor of the load tracker is the line that intersects of the two points (t_{i-q}, l_{i-q}) and (t_i, l_i), and it returns $\hat{l}_{i+\kappa}$ that is the predicted value of the load tracker $l_{i+\kappa}$ at time t_{i+k}, which is given by [40]:

$$LP_k(\overrightarrow{L_q}(t_i)) = m * (t_{i+k}) + \alpha$$

where $m = \frac{l_i - l_{i-q}}{q}$ and $a = l_{i-q} - m * t_{i-q}$.

4.6.3 Application of the model in a case study system

We consider the application of the model in the case study system described in [41]. Such system comprises two types of physical entities: mobile devices and network nodes. Mobile devices include both reconfigurable and autonomous mobile devices. The networks nodes in our system model include: a) the nodes that receive the decision-making requests (e.g. eNB) and b) the context server (CS) which handles the requests for the retrieval of protocol configurations and protocol component metadata from network repositories.

At this stage, the results of our work focus on the application and the evaluation of the load prediction mechanisms in the load balancing framework. Specifically following the above mentioned mechanisms, we computed the future values of the user satisfaction for each class of mobile devices. This was realised by applying the load tracked functions first and then the load prediction functions on the outputs of the load tracker. The results for the EMA load tracker for both types of mobile devices are presented in the figures below. Such results were derived through MATLAB simulations using the MATLAB Simulink toolbox. Specifically Figure 41 illustrates the normalized difference between the user satisfaction as it was derived from measurements of the network response time versus its predicted value, for the reconfigurable devices. It should be noted that the predicted window *k*, that represents the size of the prediction interval is equal to 15 steps. Figure 42 presents the same metrics for the autonomous mobile devices class.

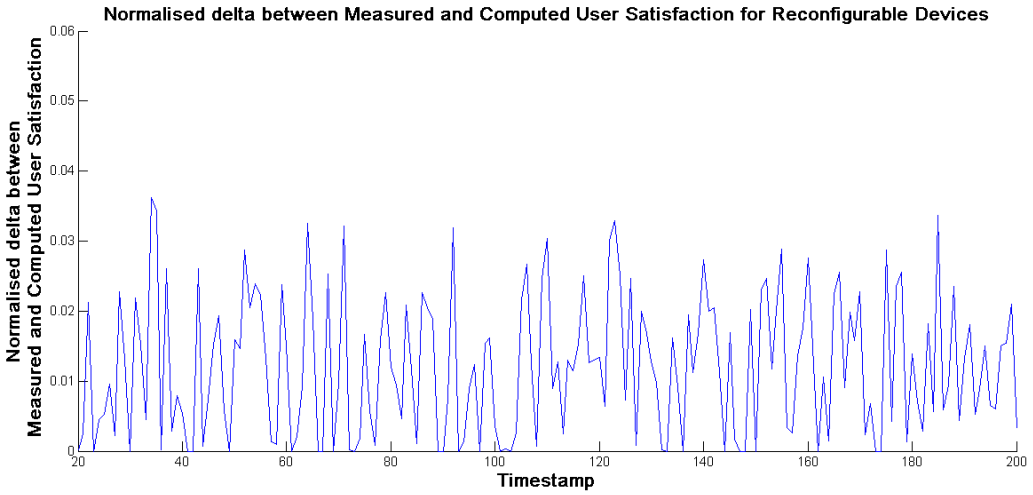


Figure 41: User satisfaction versus its predicted value for the reconfigurable mobile devices.

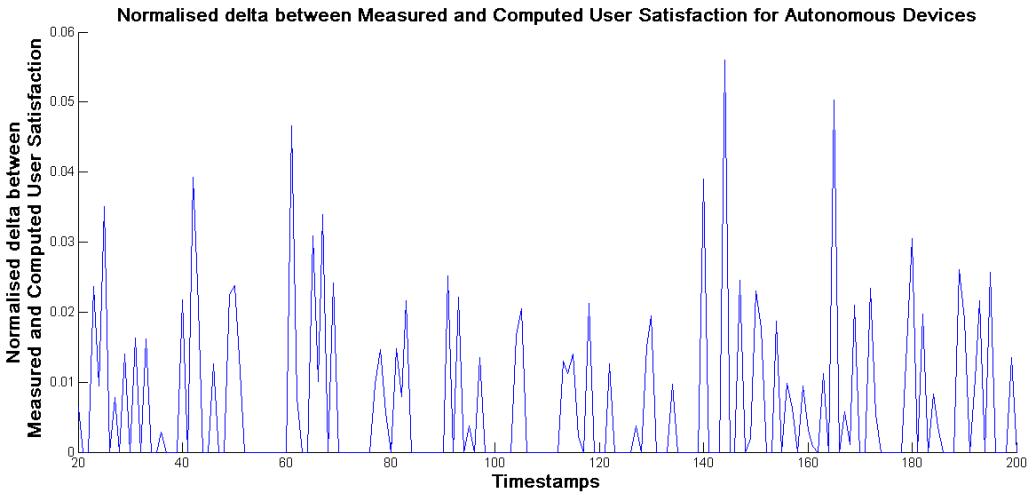


Figure 42: User satisfaction versus its predicted value for the autonomous mobile devices.

The results show that the prediction of the user satisfaction approaches very well its real value. Therefore, the introduction of the prediction functionality in the load-balancing of the decision-making requests will enable the proactive management of such requests, improving the network management procedure. In addition, the prediction for the reconfigurable mobile devices is slightly better than the one for the autonomous mobile devices. Such conclusions reveal that the introduction of autonomicity in the mobile devices affects the network management of the decision-making requests. This is due to the fact that it is the prediction and proactive management is not as accurate as for the reconfigurable mobile devices.

4.7 Dynamic AP switch ON/OFF and load balancing scheme for coverage and capacity optimization

In this work we study the dynamic deactivation and/or re-activation of one or more APs under certain conditions and the respective load balancing phase that follows the deactivation or reactivation action. This approach follows existing research work for coverage optimization, focusing on specific configuration parameters that could be either offline or online managed [45][46][47]. In such cases the de-activation of a group of APs could improve the network performance and avoid harnessing radio and energy resources. In specific geographical areas, the potentially available wireless resources (i.e. APs) over cover for a long period of time, the existing capacity requirements (e.g., throughput, number of users). Thus, the de-activation of a set of APs could be beneficial for over-served network areas, with possible re-activation when the network conditions necessitate more capacity.

In recent years there has been a continuous increase in the number of wireless access points (e.g., IEEE 802.11) that are placed at several areas (offices, homes etc.) following the vast proliferation of capacity requirements for emerging and future Internet services. All these access points (APs) are often not part of the same administrative entity, and the configuration of their locations and operational features are not necessarily planned for the “network welfare”. The uncontrolled operation of wireless APs results in dense topologies of APs, especially in urban areas, with high coverage or frequency overlapping. Hence, one of the challenges is the continuous coverage and capacity optimization of the next generation wireless communication networks, taking also into account the volatile network conditions.

At this point we should note that the presented mechanisms are related to Use Case 4: SON and SON collaboration according to operator policies. In this work we study the dynamic deactivation and/or re-activation of one or more APs under certain conditions and the respective load balancing phase that follows the deactivation or reactivation action. This approach follows existing research work for coverage optimization, focusing on specific configuration parameters that could be either offline or online managed [45] [46] [47]. In such cases the de-activation of a group of APs could improve the network performance and avoid harnessing radio and energy resources. In specific geographical areas, the potentially available wireless resources (i.e. APs) over cover for a long period of time, the existing capacity requirements (e.g., throughput, number of users).

Thus, the de-activation of a set of APs could be beneficial for over-served network areas, with possible re-activation when the network conditions necessitate more capacity. Each AP collaborates with the domain manager by providing topological and monitoring data in order to solve the optimization problem. The domain manager coordinates the management actions that require a greater view of the network area. The area that the domain manager controls is labelled as cluster. The borders of the cluster are defined by the APs that are controlled by the respective Domain Manager.

The scheme for the coverage optimization and the respective load balancing actions that are triggered after the AP deactivation or reactivation is analysed below and it is illustrated in Figure 43. Each AP scans periodically the wireless medium in order to discover the neighbouring APs (physical topology), thus building/updating its adjacency matrix. The MAC addresses and the channels used by each neighbouring AP are sensed and kept locally. Furthermore, each member node (AP) requests the associated UEs to provide the MAC addresses list of the sensed APs. The AP collects periodically the above data, builds its Local physical Topology Graph (LTG), and transmits LTG to the Domain Manager. Moreover, each AP provides to the Domain Manager information about its operational status: a) the number of the associated UEs, b) the used capacity (downlink/uplink), as well as c) the total available capacity of the AP. The list of the transmitted parameters is the following: Number of associated UEs to AP_i ($\mathrm{AP}_{\mathrm{UEs,i}}$), Capacity used by AP_i ($\mathrm{AP}_{\mathrm{Cap,i}}$), maximum available capacity for each associated terminal ($\mathrm{AP}_{\mathrm{CapAvail,i}}$), reserved capacity for each associated UE_j ($\mathrm{UE}_{\mathrm{Cap,j}}$), $\mathrm{A(i)}$ set of APs that sensed by AP_i (one hop away APs), and $\mathrm{A_e(i)}$ set of APs that sensed by UE e that is associated to AP i.

According to the population of the cluster, the Domain Manager receives the monitoring data (e.g., LTG) and the local operational status ($AP_{Cap,i}$, $AP_{CapAvail,i}$) from the member nodes and updates its self-awareness about the cluster area by building the Cluster-level Topology Graph (CTG). The Domain Manager undertakes to characterize the existing load levels in the cluster area, consisting of *n* APs, through the Capacity Usage Ratio,

$$CUR = \sum_{i=1}^{n} \frac{\mathrm{AP}_{\mathrm{Cap,i}}}{\mathrm{AP}_{\mathrm{CapAvail,i}}}$$

Thereinafter, the Domain Manager proceeds to the identification of the coverage optimization opportunities for a Low Load or High Load situation (Level 2 Situation awareness). An optimization opportunity for a Low Load situation indicates that there is the possibility to de-activate one or more APs; the goal is to avoid the harnessing of resources in the cluster area, without concurrently reducing the effective geographical coverage of the APs. Similarly, in a high load situation the Domain Manager estimates the necessity to re-activate an AP in order to address the increased capacity requirements. The Coverage Optimization Opportunity Coefficient (*COOP*) is given by:

$$\mathrm{COOP} = \mathrm{CUR}^{\mathrm{OF}} \quad (1)$$

where the Overlapping Factor (*OF*) denotes the coverage of the APs that constitute the cluster area and it is provided as follows:

$$\mathrm{OF} = \frac{\mathrm{Edges}}{\mathrm{CN} * (\mathrm{CN} - 1)} \quad (1)$$

The *Edges* parameter corresponds to the number of the existing connections (i.e. overlaps) among the APs of the cluster, while *CN* is the number of APs (i.e. Cluster Nodes) that constitute a cluster. Through equation (1), the *CUR* is associated with APs *OF* in the corresponding network area. This allows the more effective interpretation of the information that *CUR* provides in order to identify optimization opportunities for a low loaded (less capacity needed) or for a high loaded (more capacity needed) situation.

Thus, the Domain Manager of each cluster should identify the appropriate load levels in the cluster-defined network area ($\sum_{i=1}^{n} \mathrm{AP}_{\mathrm{Cap,i}}$), where a de-activation or re-activation check could be triggered. The process for APs de-activation checking is triggered if:

$$\mathrm{COOP} < \text{de-actUB} \quad (2)$$

where $de\text{-}actUB$ denotes the upper bound for APs de-activation and it is calculated as follows:

$$\text{de-actUB} = \mathrm{LL} * \log_2(\text{diameter})$$

Diameter is the greatest distance between any pair of nodes in the formed cluster that the domain manager manages, while *LL* and *HL* denote the Low Load and High Load thresholds respectively, which are initially set by the network operator. The process for APs re-activation checking is triggered if:

$$\text{COOP} > \text{re-actLB} \tag{3}$$

where $re\text{-}actLB$ denotes the lower bound for APs reactivation and it is calculated as follows:

$$\text{re-actLB} = \text{HL}*\log_2(\text{diameter})$$

A high *OF* is useful in order to address a low loaded situation (low *CUR*), since there are more opportunities for the UEs to be handed over, without reducing the access capabilities at the geographical area that the APs cover. In the case of a high load status (high *CUR*), a low or a medium-dense network area provides more opportunities for the re-activation of an AP (previously de-activated). The re-activation of an AP in a high-dense network area of APs (if the capacity requirements do not require that), will increase further the overlapping of the selected channels and thus affecting the noise and the bit error rate (BER) levels in the network area.

If the *CUR* of the network cluster area has reached the level, which satisfies in equation (3), then the domain manager proceeds to build the list of candidate APs for de-activation and selects the most appropriate. The list of the candidate nodes for de-activation includes those APs, where all associated UEs ($AP_{UEs,i}$) could be transferred to a neighbouring $AP \in A_e(i)$ by satisfying all $UE_{Cap,\ j}$ requirements. Then, the domain manager selects AP_m, with the minimum ratio: $\frac{\text{AP}_{\text{Cap,m}}}{\text{AP}_{\text{CapAvail,m}}}^{\text{OF}_\text{m}}$ where OF_m is the overlapping factor between AP *m* and its one-hop away nodes; equation (2) is used for the calculation.

The scheme for load balancing after the access point de-activation is presented below: The terminals that are associated to the AP_m are handed over to neighbouring sensed APs. Each terminal is allocated to the AP with the maximum Received Signal Strength (RSS) from those APs that have CUR less than the average CUR of the cluster ($\sum_{i=1}^{n}\frac{\text{AP}_{\text{Cap,i}}}{\text{AP}_{\text{CapAvail,i}}}$). If there is no such AP then it is selected the AP that has the minimum CUR and is over the average CUR of the cluster.

In the case that the *CUR* level in the cluster has increased up to point that in equation (4) is satisfied then the process for APs re-activation is initiated. The Domain manager undertakes to identify if there is a de-activated AP that could be enabled in order to the serve the increasing capacity requirements. If more than one AP are available for re-activation, then the domain manager selects AP *p*, which one-hop away neighbours (*k*) have the maximum: $\left(\sum_{i=1}^{k}\frac{\text{AP}_{\text{Cap,i}}}{\text{AP}_{\text{CapAvail,i}}}\right)^{\text{OF}_p}$, where OF_p presents the overlapping factor between candidate AP *p* for re-activation and the *k* neighbouring APs. The goal is to find the area with the highest load and the less overlapping factor.

The scheme for load balancing after the re-activation of an access point is presented below: After the re-activation of AP *p* the domain level engine builds the list of terminals that are already associated to an AP and also have the capability to sense the new AP *p.* Based on the previous list we prioritize those terminals to handover which are associated to an AP that has CUR higher that the average CUR of the cluster ($\sum_{i=1}^{n}\frac{\text{AP}_{\text{Cap,i}}}{\text{AP}_{\text{CapAvail,i}}}$). The process stops when the CUR of the new AP is over the average CUR of the cluster.

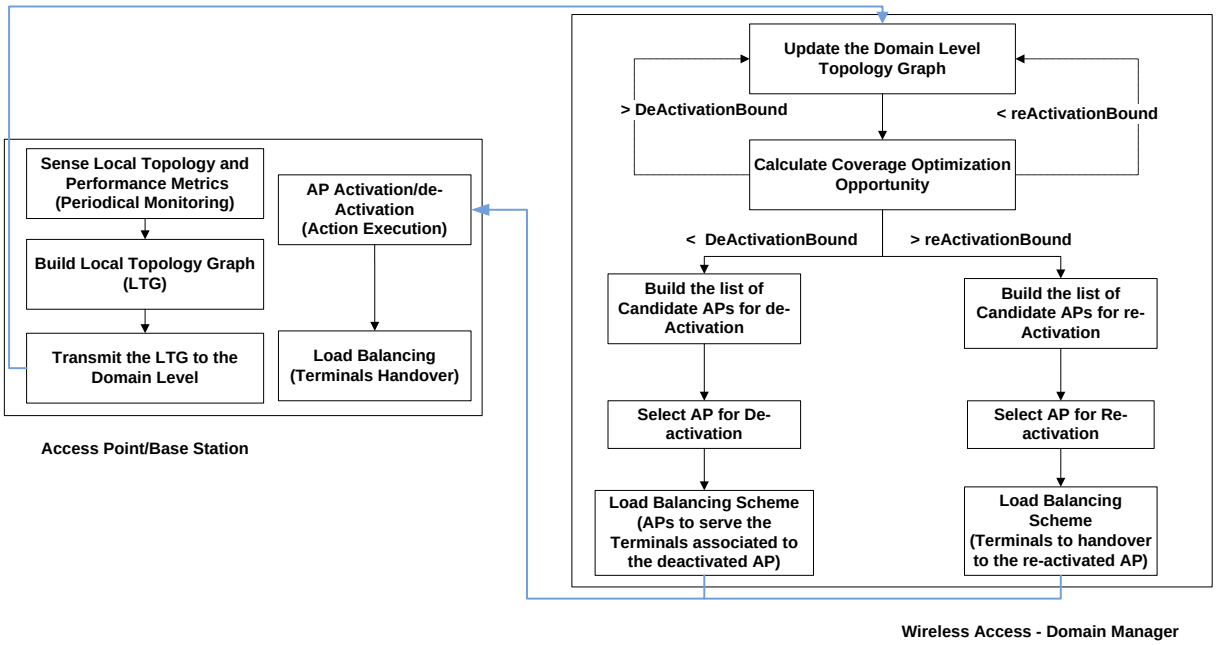
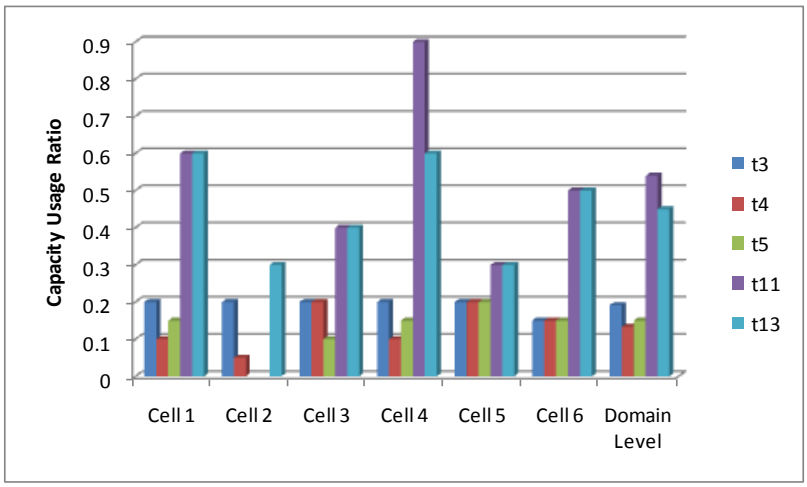


Figure 43: Dynamic Switch On/Off and Load Balancing Scheme.

Figure 44 depicts the variation of the COOP value for a case study network area that consists of six cells, having an overlapping factor, OF = 0.53. COOP value changes due to the variation of the end user traffic and the actions for AP deactivation or AP re-activation. At time instance t4, AP de-activation is triggered, since COOP is below the deactivation threshold (Figure 44-a). As it is depicted in (Figure 44-b) Cell 2 is deactivated and the associated terminals are handed over to Cell 4. The average CUR is increased in order to avoid resources underutilisation. On the other hand, at t11 a re-activation trigger arises, due to an increase of the end user traffic for all cells. Consequently, Cell 2 is activated and terminals are handed over to the re-activated cell. The capacity usage ratio of cell 4 is reduced, as well the domain level capacity usage ratio, by avoiding the overutilisation of radio resources and thereinafter blocking or interference issues. Furthermore, AP de-activation is an energy saving method, especially for low traffic periods.

Taking into account the above analysis and discussion, the COOP metric provides a correlation between the capacity that is offered in cluster area and the overlapping of the cells. High levels of COOP indicate overutilisation of the available cells. Thus, the associated configuration action (AP re-activation) requires a load balancing scheme, where the main criterion is the offloading of congested APs. On the other hand, for low levels of COOP, which indicates underutilisation of the available resources, the load balancing scheme that follows the associated configuration action (AP de-activation) mainly considers the RSS level of neighbouring APs that terminals can sense.

(a)



(b)

Figure 44: (a) Domain-Level COOP Metric variation, (b) Capacity Usage Ratio per AP and at the domain level

The provided scheme is fully autonomous. The network operator or the network administrator should predefine only the thresholds for the triggering of the optimization scheme for AP activation and re-activation.

4.8 Configuration optimization for self-healing actions

The question that this work addresses is what actions are needed in order to handle and recover a base station failure event in a future wireless network. This can be achieved by reconfiguring the radio parameters of the network under several certain limitations (i.e. SINR values, cell load, etc.). This approach follows existing research work on radio coverage optimization, cell outage management in both WLANs and LTE networks. Namely the approach in [56] addresses key aspects for the development of cell outage management algorithms including an overview of potentially useful measurements and a set of appropriate control parameters (e.g. handover failure rate, interference measurements, cell load). The authors do not practically develop algorithms for cell outage detection and compensation. [57] proposes a distributed method to address accidental AP failure by maximizing network coverage and system capacity at the same time. To ensure all of the APs be able to acquire enough information from their neighbours, a station-assisted beacon-transmitting mechanism is proposed. In [58], the authors investigate the problem of coverage planning in WLANs, which is crucial in cell outage management. Most of the proposed formulations aim at maximizing the capacity of either the overall network or the single end user. Our work extends state of the art work by proposing an advanced fuzzy-logic based approach for future wireless networks, to simultaneously deal with the reassignment of the affected mobile devices to neighbouring base stations while minimizing the generated interference.

Goal and motivation of this work is to handle as efficiently as possible failures in a wireless network and to compensate for a cell in outage. The target is twofold:

- To compensate (i.e. reassign to neighbouring base stations) as many affected mobile stations as possible by increasing the transmission power of the neighbouring base stations,
- To generate as little additional interference as possible. (The extra interference is due to the prior increment of the transmission power.)

This work is related to Use Case 4: SON and SON collaboration according to operator policies. More specifically, this work is most related to the first identified problem the UC4 attempts to solve, i.e., the design of distinct SON functionalities in network nodes to efficiently self-configure and self-optimize network resources. Regarding future steps, this work may also be related to identified problem two in order to handle multi-objective problems or identified problem three in order to take into account operator policies during the self-healing process.

4.8.1 Solution framework

For the solution of this specific use case, this work is based on the framework specified in [48]. The general framework, which describes the solution to such a problem, is shown in Figure 45:

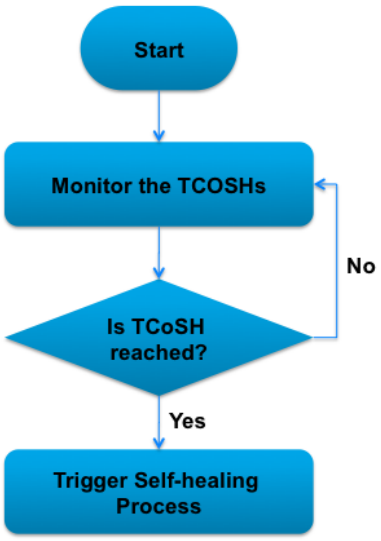


Figure 45: Generic Self-healing Framework.

Regarding the specific case we examine, the general scenario is depicted in Figure 46. The solution approach is described in the following subsections.

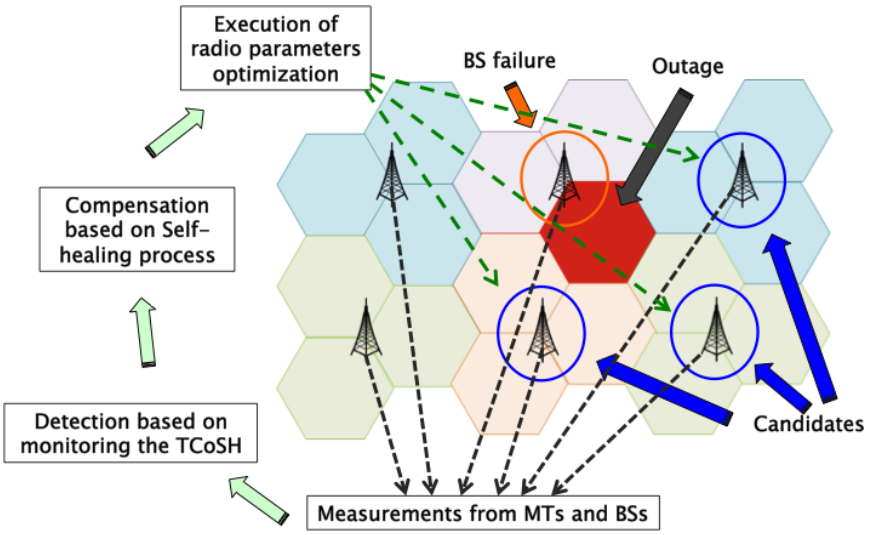


Figure 46: General Self-healing scenario.

4.8.2 Description of trigger conditions and self-healing process

TCoSH (Trigger Condition of Self-Healing) is a rather simple relation, which is based on four (4) identified parameters:

- Incoming / outgoing handover failure rate
- Relative load indicator
- Interference measurements
- Reference Signal Received Power (RSRP)
- These four parameters are evaluated in terms of a set of specified thresholds via a simple approach, and in case the TCoSH is reached, the self-healing process is triggered.

In order the self-healing process to execute efficiently and correctly, a set of information is required to the domain manager so as to take the optimum decision to this specific problem. This information includes:

- Information from the eNodeBs
  - Neighbouring list (i.e. IDs of the neighbouring eNodeBs)
  - IDs of mobile terminals assigned to the eNodeB
  - Information about the status of the eNodeB (i.e. Tx power, etc.)
  - Signal measurements
- Information from the UEs
  - Signal measurements

This information is periodically aggregated to the domain manager in order to be as updated as possible. Based on this information the flow of the process is described as follows:

- Domain manager (DM) identifies a failed eNodeB based on the specified TCoSH.
- DM retrieves all neighbouring eNodeBs from its neighbouring list.
- DM retrieves the IDs of the assigned mobile terminals (MTs).
- DM checks if these MTs have been reassigned to any of the neighbouring eNodeBs. (If yes, the self-healing action may be executed at a later stage.)
- DM feeds the parameter optimization algorithm with all required information so as to take the best decision for the compensation.

The parameter optimization algorithm is described in the next sub-section.

4.8.3 Parameter optimization algorithm

The parameter that will be optimized in this specific scenario is the Tx power of the neighbouring eNodeBs, if needed. The optimization algorithm is based on fuzzy logic, which is a problem-solving methodology based on

control system theory. This algorithm takes as input a set of related parameters and exports as output the new Tx power of the neighbouring eNodeBs in order to compensate for the cell outage.

More specifically the inputs for the fuzzy logic system are the following:

- Percentage of Tx Power (current Tx Power / max. allowed Tx Power)
- Cell load
- # of MTs assigned
- Average RSS (RSS of neighbouring eNodeBs and MTs for this eNodeB based on a specified weight)

For each of the aforementioned parameters, as well as the output parameter, the respective membership functions have been specified. For instance, the membership function of the Tx Power is shown in Figure 47:

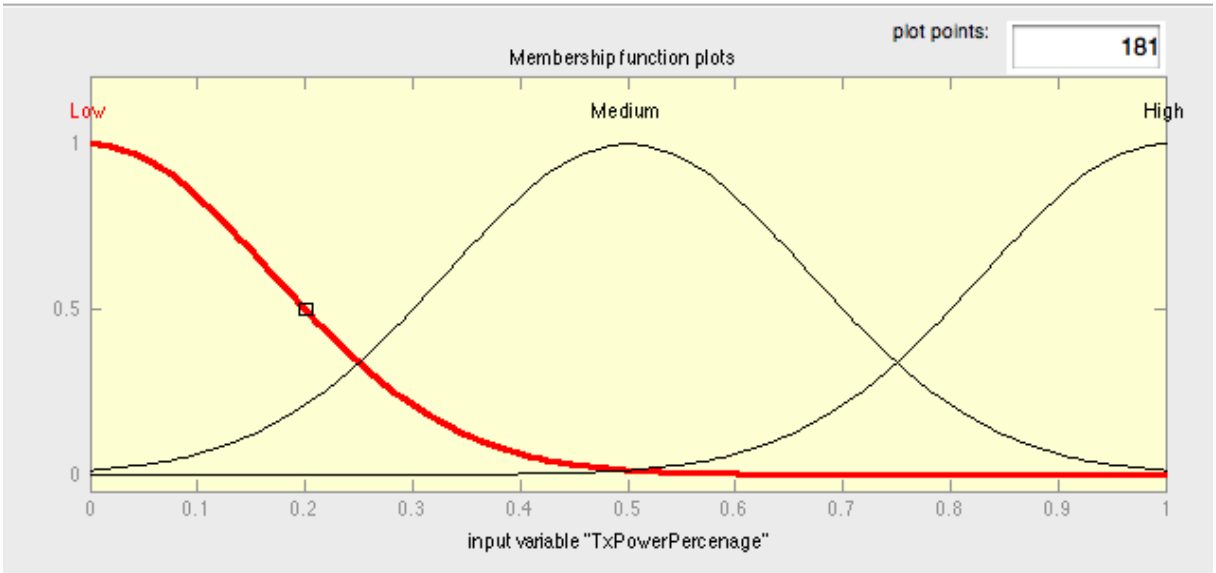


Figure 47: Tx Power Percentage Membership Function.

Then a set of rules is defined based on which the input variables are transformed to the output variable. A subset of these rules is presented in Table 7:

Table 7: Description of a subset of applied rules.

If (NeighbouringRSS is High) then (TxPowerAdjustment is Low)
If (NeighbouringRSS is Low) and (CellLoad is Low) and (NumberMTs is Low) and (TxPowerPercentage is Low) then (TxPowerAdjustment is High)
If (NeighbouringRSS is Low) and (CellLoad is High) and (NumberMTs is High) and (TxPowerPercentage is AboveMedium) then (TxPowerAdjustment is BelowMedium)

Finally the relation of the input variables and the output variable is depicted in Figure 48. In this figure, the relations of NeighbouringRSS and TxPowerPercentage to TxPowerAdjustment parameters are presented based on the aforementioned rules. For example, if the NeighbouringRSS is high (close to 1) then the TxPowerAdjustment is low, despite the TxPowerPercentage value. In addition, it should be noted that this figure illustrates the outcome of the decision part of the parameter optimization algorithm.

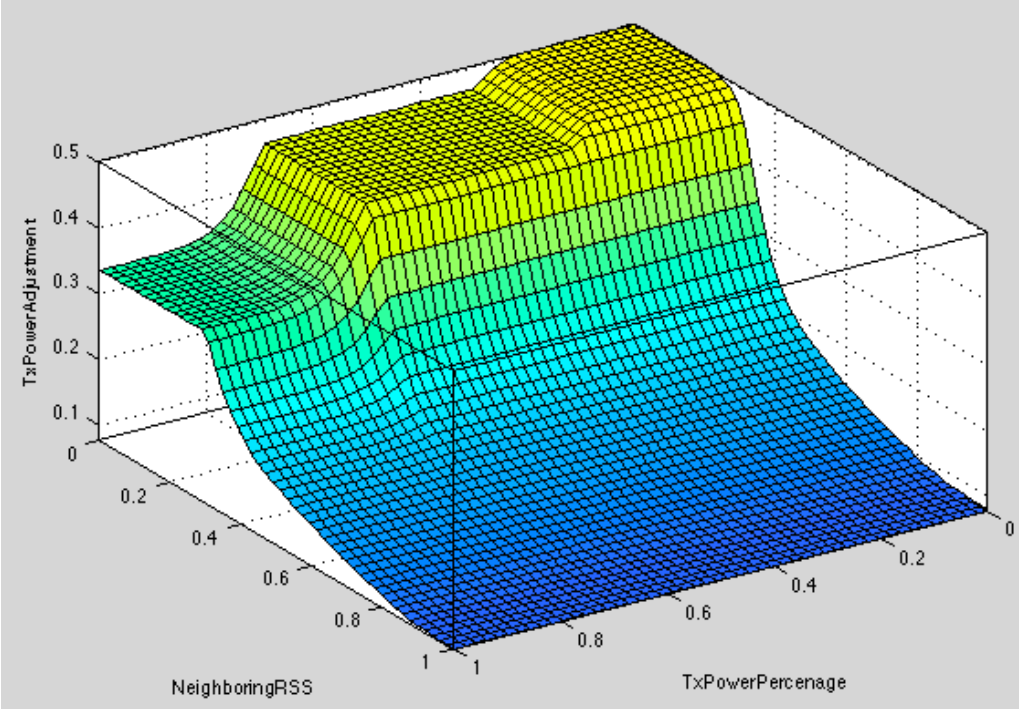


Figure 48: Relation of NeighbouringRSS and TxPowerPercentage to TxPowerAdjustment.

This work deals with the self-healing of future wireless networks, focusing on radio parameters reconfiguration targeting the effective management of the affected mobile devices. This is realized by fulfilling two contradictive requirements: increasing the transmission power of the neighbouring base stations that will handle the affected devices while minimizing the additional interference. To this end, this mechanism allows us to maximize the number of affected mobile devices that will be reassigned to neighbouring base stations, optimizing the load balancing procedure. The proposed scheme is fully autonomous. The network operator should only specify the TCoSH threshold and the individual thresholds of the parameters taken into account in the trigger conditions of the self-healing process.

4.9 Optimal load balancing through sophisticated energy-aware traffic engineering

One of the main challenges in the Future Internet era will be the increase of QoS-demanding applications that need to be supported. Unavoidably, this will lead to increased traffic that has to be served by the deployed networks. Moreover, the energy consumption and the configuration/management complexity will be

significantly increased, too. Current network infrastructures and their associated management systems face problems in keeping up with such stringent requirements. Therefore, it is essential to reconsider the design and the management of the Future Networks. An important goal in this direction is to achieve the best ratio of performance to energy consumption and at the same time assure manageability. This work presents a distributed Traffic Engineering scheme that provides load balancing and energy-awareness in accordance with the operator's needs. Results from trace-driven simulations confirm the capability of our scheme to meeting the needs of Future Networks.

We present a heuristic load balancing mechanism, which is based on the previously described policy-based energy-aware traffic engineering approach (see Section 3.4.2.1**).** In this section, we place special focus on the load balancing that is achieved in the network and we validate its performance through simulations that get as input real Internet Service Provider (ISP) traces. Moreover, we discuss several issues related to the implementation/deployment of the proposed scheme in real core/backhaul networks.

We consider a network model, where each ingress router may have traffic demands for a particular egress router or set of routers. We use multiple paths (MPLS tunnels) to deliver traffic from the ingress to the egress routers. We must mention here that traffic is split among the available paths at the granularity of a flow, to avoid reordering TCP packets or similar effects that lead to performance degradation (using efficient traffic splitting approaches, like [28]). In addition, we consider that the paths are computed and re-computed (if it is necessary) offline by the operator, since most of the operator's networks work in this way.

The main "cornerstones" in the proposed mechanism are the following *low-complexity* and *distributed* algorithms (Table 8 contains the definitions of the variables used):

Load Balancing: *Given the* a_l *values for the links in the network, find the corresponding* x_{ip} *values that provide balanced network operation in terms of link utilisation.* In order to provide an efficient solution we investigate for each ingress-egress node pair the paths that go through the maximum utilized link. Then, we "relieve" this link by moving a portion of traffic Δx and provisioning it proportionally to the rest paths (inverse procedure of progressive filling that leads to optimal load balancing based on [49]). This procedure continues till convergence to the optimal x_{ip} values.

Energy Saving: *Given the* x_{ip} *values resulted from* **Load Balancing***, find the maximum set of links that could be turned into sleeping mode.* For each ingress-egress node pair we find the routers that are part of the active routes and turn the lines of their network card that are not used (by any path in the network) into sleeping mode.

Table 8: Variables

Variables	Description
L	Set of links in the network
IE	Set of Ingress to Egress node pairs
e_l	Energy consumption of the port connected to link l
P_i	Set of paths of Ingress to Egress node pair i
T_i	Traffic demand of Ingress to Egress node pair i
a_l	Binary variable: 0 if link l is sleeping, 1 if link l is active
u_l	Utilisation of link l
c_l	Capacity of link l
x_{ip}	Fraction of traffic of Ingress to Egress node pair i, sent through the path p
r_{ip}	Traffic of Ingress to Egress node pair i, sent through path p
P_l	Set of paths that go through link l
L_i	Set of links that are crossed by the set of paths P_i
E	Demand of the operator in energy consumption

The proposed approach (Figure 49) gets as input the operator request, as far as the energy consumption is concerned (E). Then, **Load Balancing** and **Energy Saving** are applied, by each ingress-egress node pair i in order to balance the link utilisation in their paths and put the links that are not utilized into sleeping mode. Next, the new energy consumption level is compared to E in order to realize if we have reached the desired state. If not, the heuristic mechanism continues by excluding the path p with the minimum $x_{ip}T_i$ (lightest path). Traffic engineering adapter controls the aforementioned procedure. The heuristic mechanism iterates based on the updated P_i values, optimizes x_{ip} and a_l values $\forall p \in P_i, l \in L_i$ and finally, stops when the operator's energy consumption goal is achieved.

We must highlight that our approach (Energy-Aware Traffic Engineering, ETE) can be executed in an autonomous/cognitive manner using monitoring and knowledge sharing (Figure 49). In other words, the status of the system is continually monitored in order to apply ETE when needed. Moreover, knowledge (related to configuration actions and different states of the system) is stored for increasing reliability and automation of the system reactions. In this way there is no need for execution of the proposed heuristic mechanism when a "known" event happens in the network (e.g. new request with specific characteristics)

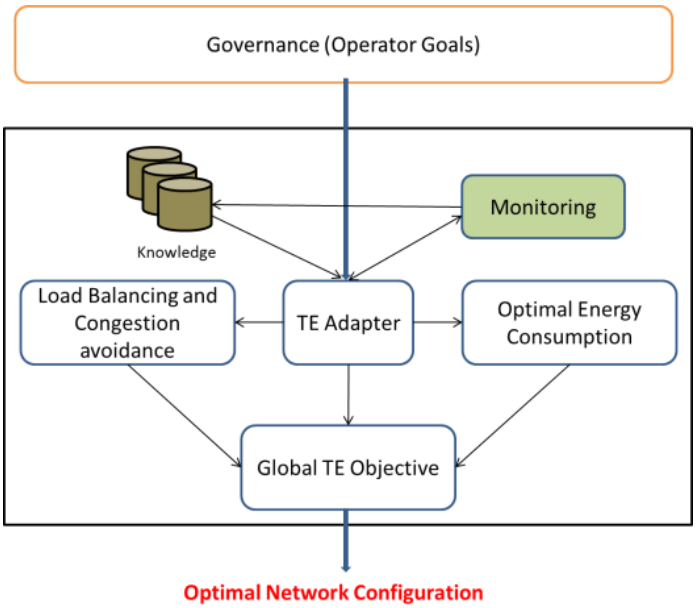


Figure 49: Cognitive/Autonomic Energy-aware Traffic Engineering.

In order to give a thorough presentation of the proposed mechanism we discuss several deployment issues that arise when trying to apply the ETE scheme in real network deployments. Firstly, ETE is executed at each ingress-egress node pair in the network and the main decision mechanism is executed at the ingress nodes. Moreover, supposing that we support MPLS-based operation, we require several label switched paths (LSPs) for each ingress-egress node pair in order to split the traffic to the available routes (the current ISP-class routers can support up to 16 LSPs). Traffic splitting is performed seamlessly using sophisticated mechanisms, like [28]. In addition, for each ingress-egress node pair MPLS-based monitoring is performed (probe request/response) in order to estimate the network and flow performance. Lastly, ETE is implemented on top of the router functionality (software package) handling the basic functionalities that are offered (sleeping mode, etc.).

We present now the evaluation study of the proposed scheme. In order to provide realistic simulation results, we use real ISP topologies and traces provided by Rocketfuel tool [50]. In our simulation, we consider Tiscali (3257) traces and a network topology consisted of 18 routers and 77 links. We compare the performance of ETE to OSPF-TE (Open Shortest Path First – Traffic Engineering) [51] that was applied in Tiscali network when the traces were collected.

Figure 50 depicts the utilisation of the links in the network when ETE and OSPF-TE are applied. We observe that ETE is able to keep the link utilisation at low levels using the minmax link utilisation policy that is adopted. On the other hand, OSPF-TE uses a dynamic procedure to calculate the link weights in order to route the traffic efficiently, which could lead to link overutilisation.

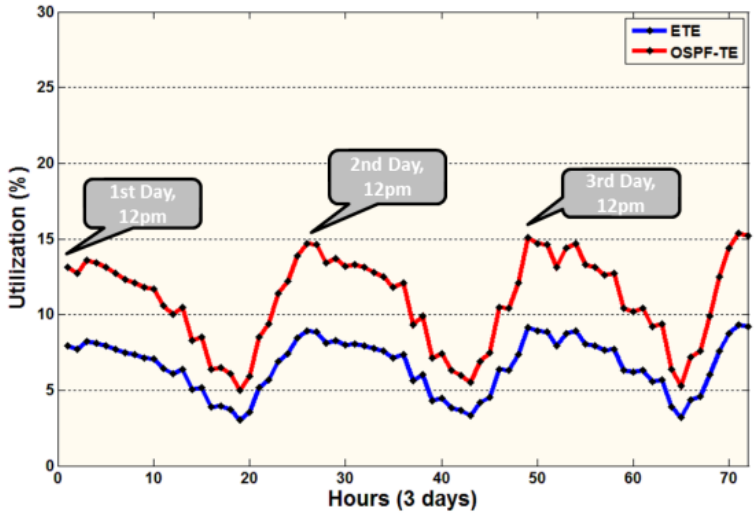


Figure 50: Link utilisation when ETE and OSPF-TE are applied.

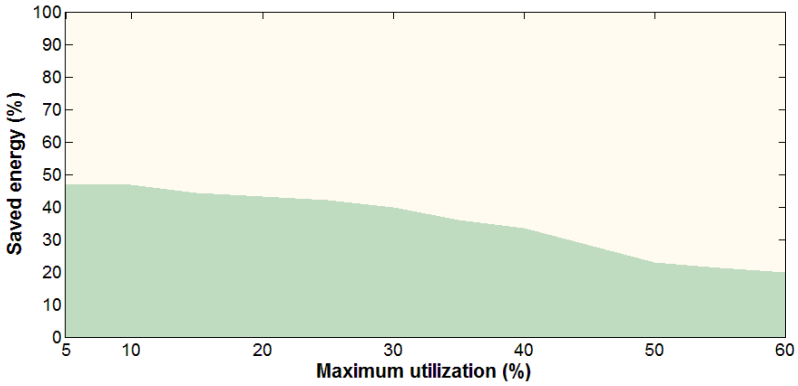


Figure 51: Percentage of saved energy vs. maximum link utilisation.

Then, we plot the percentage of the initially consumed energy that is saved when ETE is applied. Similar to the previous test case, we consider Tiscali traces in order to build a relationship between the load of the traffic that must be supported in the network and the energy saving that could be achieved by ETE. Figure 51 visualizes that when the maximum link utilisation is low, the energy saving is close to 50%. The percentage of saved energy continually drops while the traffic in the network grows and therefore the utilisation of the links is getting high.

Lastly, Table 9 presents the existing trade-off between the energy consumption and the maximum link utilisation in the network. The first column contains the operator's request, as far as energy saving is concerned. In addition, in the next two columns we observe the percentage of the links that must be turned into sleeping mode and the routes that will be excluded in order to approach the corresponding E values. The last column presents the balanced link utilisation that is achieved by ETE for each desired E level. It is obvious that there is an important trade-off between the balanced and energy-aware network operation, which is handled by ETE, based on the operator's goals.

Table 9: ETE performance – trade-off.

Requested percentage for energy saving *(E)*	Percentage of "sleeping" links	Percentage of routes excluded	Maximum link utilisation
10%	5%	2%	6%
20%	18%	8%	13%
30%	24%	13%	21%
40%	36%	18%	42%
50%	45%	22%	58%

In this section we presented an Energy-Aware Traffic Engineering scheme that try to meet the requirements of the future networks and pave the way for new "modern" traffic engineering approaches. The trace-driven simulation results indicate that our approach is capable to achieve load balancing and energy-awareness.

4.10 Discussion

This chapter presented a collection of load balancing approaches designed to autonomously solve various problems originating from the use cases. One part of the work focuses on developing and improving algorithms that would fit into the context of self-organizing networks, and another part tries to determine which already existing solutions are suitable and well-performing in specific scenarios. In both cases, evaluation is done either by simulation or proof-of-concept prototypes. At the moment, the discussed load balancing approaches are linked together via common use cases. Later, a more integrated view on load balancing will be created. The first step towards this goal has been taken by sketching the load balancing framework in case of mobile and core networks and identifying the regions where the different load balancing activities locate. The ultimate goal is a generic end-to-end load balancing framework integrated in the UMF where load balancing related events can trigger a series of selected load balancing mechanisms operating in different parts and layers of the network. This will require end-to-end load balancing and virtualization solutions, which will be studied during the second and third year of UniverSelf.

5 Conclusion

In this deliverable we have worked towards means to provide the network more authority over itself with the help of the right methods, or in other words we have worked towards *Network Empowerment*. This entails suitable methods and the ability to solve problems not only for isolated problems, but also for problems that span across multiple network domains, such as wireline networks and fixed networks. In this context, we have provided first results on refining classes of methods (evolutionary algorithms and a variant of the gradient descent approach), on cross-domain optimization spanning wireless and wireline network segments, and on load balancing in different network domains.

The chapter on random elements discusses several classes of methods that are applied depending on whether suitable models are available for predicting the outcome of changes in parameters or not. Secondly, the aspect of convexity is discussed, but wireless access networks typically come along with statistical properties so that one can get trapped in local optima. The chapter is concerned with injecting the right level of randomness so that converging in local optima can be avoided. In particular in this chapter we have moved closer to finding suitable methods for problems as we have compared the performance of evolutionary algorithms with the performance of both simulated annealing and mixed integer linear programming for the self-optimization of OFDM resource allocation and the self-optimization of MPLS traffic engineering, respectively. In both cases, evolutionary algorithms were better-suited. In the future, a strong emphasis will be on more focused optimizations: not only has the convergence to be quick, the path towards the optimum also has to be "benign" in the sense that the system will avoid trying configurations that will significantly reduce the system performance or even cause system instability. In other words, the network operator may accept that the convergence of a method consumes twice the time if it is ensured that the quality of service is not perceivably degraded during the optimization process. Having stated this, evolutionary algorithms are definitely a promising class of methods as they are able to outperform methods such as simulated annealing or mixed integer linear programming as demonstrated in the chapter on Random elements.

The chapter on Governance presents, formulates, and solves policy based parameter self-optimization problems. The gist of this work is the joint evaluation of resource assignments across wireless and wireline network domains by using policies and also here different methods are compared with each other in order to find the best possible solution. The main difference of the work presented in this chapter compared with the state of the art is that in this work the main emphasis is given to the *joint end-to-end* management of the network. This means that here the main focus is given at the policy-based governance of networks, which are composed by multi-vendor and multi-technology segments, as a whole, regardless of the individual network segments' nature and thus this chapter is a first move towards trustworthy federation. This work also represents the most obvious link towards the UMF, and in the chapter this link is clearly sketched. In the near future, we will establish similar links between the UMF and the other tasks/task forces in the Network Empowerment work package.

The chapter on Load balancing displays a collection of different load balancing approaches in different network environments (radio access network, backhaul, core, and interfaces between these). In this context, both existing and newly developed methods are used. State-of-the-art approaches are extended by methods like self-organizing maps or topology graphs. Also, traffic engineering aspects are extended by including energy awareness. It is now planned to provide a more integrated view with a generic load balancing framework where a certain event may trigger a series of selected load balancing mechanisms.

With the latter two chapters we have provided examples of problems that can by no means be seen from an isolated perspective and that require cross-technology and cross-network domain expertise. Complementarily to this, we have devised methodological solutions for these problems. Besides the question "what method is most suitable for a given problem?", the chapter on random elements conversely addresses the question how a method is applied suitably. The thought behind this is that applying the right method in an inappropriate way can be as damaging as applying an inappropriate method in a favourable way (within the limits of the chosen method). As evolutionary algorithms are such a standard tool for any problems where the search space is too large for more focused methods, we have dedicated some time to study different flavours of this powerful class of methods.

Considering the first results we have presented in this deliverable, we will now focus on the following aspects:

- Provide links between problems and best-suited methods

- Tailor the chosen methods to the specific particularities of the addressed problem
- Strengthen the link between the proposed approaches and the UMF
- Further harmonize the proposed solutions for a given problem into an integrated solution

In conclusion, in this deliverable we strive to get the most efficient self-x methods for the problems given by use cases defined in deliverable D4.1. With this effort, we have done the first step towards a toolbox of solutions for operator problems. Please note that the design of distributed capabilities and mechanisms for the orchestration of these solutions will be addressed in deliverable D3.4.

References

[1] D. E. Goldberg, “Genetic Algorithms in Search, Optimization and Machine Learning”, Addison-Wesley Longman Publishing Co., 1989

[2] J. R. Koza, “Genetic Programming: On the Programming of Computers by Means of Natural Selection”, Volume 1, MIT Press, 1992

[3] A. E. Eiben and J. E. Smith, “Introduction to Evolutionary Computing”, Springer, Natural Computing Series, 1st edition, 2003

[4] T. Hu, and W. Banzhaf, “Evolvability and Speed of evolutionary algorithms in light of recent developments in biology”, Journal of artificial evolution and applications, Hindawi publishing, 2010

[5] H. Meunier, E. G. Talbi, and P. Reininger, “A multiobjective genetic algorithm for radio network optimization”, Proc. of IEEE Congress on evolutionary computation (CEC), pp.317-324, 2000

[6] W.-H. Sheen, S.-J. Lin, and C.C. Huang, “Downlink optimization and performance of relay assisted cellular networks in multicell environments”, IEEE Trans. Vehicular Technology, vol. 59, no. 5, June 2010

[7] H. Ahmadi and Y. H. Chew, “Adaptive Subcarrier-and-Bit Allocation in Multiclass Multiuser OFDM Systems using Genetic Algorithm”, PIMRC 2009

[8] H. Eckhardt, S. Klein, and Markus Gruber, “Vertical Antenna Tilt Optimization for LTE Base Stations”, IEEE Vehicular Technology Conference, Spring 2011

[9] Unified Management Framework, Univerself Project, http://www.univerself-project.eu/

[10] W. E. Walsh et al., “Utility Functions in Autonomic Systems”, First International Conference on Autonomic Computing (ICAC'04), pp. 70-77, 2004

[11] J. O. Kephart and R. Das, “Achieving Self-Management via Utility Functions”, IEEE Internet Computing, 40-48, 2007

[12] J. O. Kephart and W. E. Walsh, “An artificial intelligence perspective on autonomic computing policies”, Fifth IEEE International Workshop on Policies for Distributed Systems and Networks (POLICY'04), 2004

[13] A. Mas-Colell, M. D. Whinston, and J. R. Green, “Microeconomic Theory”, Oxford University Press, 1995

[14] J. Wilkes, “Market Oriented Grid and Utility Computing”, Chapter 4: “Utility functions, prices, and negotiation”, John Wiley & Sons, Inc 2009

[15] S. Russell and P. Norvig, “Artificial Intelligence: A Modern Approach”, Prentice Hall, second edition, 2003

[16] N. Wang, K. Ho, G. Pavlou, and M. Howarth, “An Overview of Routing Optimisation for IP Traffic Engineering”, IEEE Surveys and Tutorials, Vol. 10, No. 1, pp. 36-56, 2008

[17] R. Teixeira, T. Griffin, A. Shaikh, and G.M. Voelker, “Network Sensitivity to Hot-Potato Disruptions”, ACM SIGCOMM, 2004

[18] D. O. Awduche, “MPLS and Traffic Engineering in IP Networks”, IEEE Communications Magazine, vol. 37, no. 12, pp. 42–47, 1999

[19] B. Fortz, J. Rexford, M. Thorup, “Traffic Engineering with Traditional IP Routing Protocols”, IEEE Communications Magazine, vol. 40, no. 10, pp. 118–24, 2002

[20] D. K. Goldenberg, L. Qiu, H. Xie, Y. R. Yang, and Y. Zhang, “Optimizing Cost and Performance for Multihoming”, ACM SIGCOMM 2004, pp. 79–9

[21] A. Elwalid, C. Jin, S. Low, and I. Widjaja, “MATE: MPLS Adaptive Traffic Engineering”, IEEE INFOCOM 2001 pp. 1300–09

[22] M. Kodialam and T. V. Lakshman, “Minimum Interference Routing of Applications to MPLS Traffic Engineering”, IEEE INFOCOM 2000, pp. 884–93

[23] M. Kodialam, T. V. Lakshman, and S. Sengupta, “Online Multicast Routing with Bandwidth Guarantees: A New Approach Using Multicast Network Flow”, IEEE/ACM Trans. Networking, vol. 11, no. 4, pp. 676–86, 2003

[24] K. Hinton, J. Baliga, M. Feng, R. Ayre, and R. S. Tucker, “Power consumption and energy efficiency in the internet”, IEEE Network Magazine, Special Issue on "Energy-Efficient Networks," vol. 25, no. 2, pp. 6-12, 2011

[25] R. Bolla, R. Bruschi, A. Cianfrani, and M. Listanti, “Enabling Backbone Networks to Sleep”, IEEE Network Magazine, Special Issue on "Energy-Efficient Networks," vol. 25, no. 2, pp. 26-31, 2011

[26] A. Cianfrani, V. Eramo, M. Listanti, M. Marazza, and E. Vittorini, “An Energy Saving Routing Algorithm for a Green OSPF Protocol”, IEEE INFOCOM 2010

[27] S. Nedevschi, L. Popa, G. Iannaccone, S. Ratnasamy, and D. Wetherall, “Reducing Network Energy Consumption via Sleeping and Rate-Adaptation”, ACM, USENIX, NSDI, 2008
[28] S. Kandula, D. Katabi, S. Sinha, and A. Berger, “Flare: Responsive Load Balancing Without Packet Reordering”, ACM Computer Communications Review, 2007
[29] D. Bertsekas and R. Gallager, “Data Networks”, Englewood Cliffs, NJ: Prentice-Hall, 1992
[30] IBM ILOG CPLEX Optimizer, http://www-01.ibm.com/software/ integration/ optimization/cplex-optimizer
[31] R. Pabst et al., "Relay-based deployment concepts for wireless and mobile broadband radio," IEEE communication magazine, vol. 42, no. 9, pp. 80-89, 2004
[32] The GEANT network topology, http://www.geant.net
[33] D. Awduche, A. Chiu, A. Elwalid, I. Widjaja, and X. Xiao, “Overview and Principles of Internet Traffic Engineering,”, RFC3272, 2002
[34] S. Klein, I. Karla, and E. Kuehn, “Potential of intra-LTE, intra-frequency load balancing,” IWSON at IEEE VTC, Spring 2011
[35] S. Horsmanheimo, J. Eskelinen, and H. Kokkoniemi-Tarkkanen, “NES - Network expert system for heterogenous networks,” International Conference on Telecommunications (ICT-2010), 2010
[36] J. Mäkelä, K. Pentikousis, M. Majanen, and J. Huusko, “Trigger management and mobile node cooperation,” in Cognitive Wireless Networks; Concepts, Methodologies and Visions Inspiring the Age of Enlightenment of Wireless Communications, Springer, 2007
[37] J. Mäkelä, M. Luoto, T. Sutinen, and K. Pentikousis, “Distributed information service architecture for overlapping multi-access networks,” Multimedia Tools and Applications, Springer, 2010
[38] K. Pentikousis and T. Rautio, “A Multiaccess Network of Information,” in WoWMoM, Montreal, 2010
[39] T. Rautio, O. Mämmelä, and J. Mäkelä, “Multiaccess NetInf: a prototype and simulations,” in Tridentcom, Berlin, 2010
[40] M. Andreolini, S. Casolari and Colajanni, “ Load prediction models in web-based systems,” in VALUETOOLS, Pisa, 2006
[41] E. Patouni, N. Alonistioti and L. Merakos, “Cognitive Decision Making for Reconfiguration in Heterogeneous Radio Network Environments,” IEEE Transactions on Vehicular Technology, vol. 59, no. 4, pp. 1887-1990, 2010
[42] M. Litoiu and J. Rolia, “Object allocation for distributed applications with complex workloads,” in 11th Int. Conf. Comput. Perform. Eval.:Modelling Tech. Tools, London, 2000
[43] M. Litoiu, “A performance analysis method for autonomic computing systems,” ACM Trans. Auton. Adapt. Syst., vol. 2, no. 1, 2007
[44] G. Balbo and G. Serazzi, “Asymptotic analysis of multiclass closed queueing networks: multiple bottlenecks,” Perform. Eval. J, vol. 30, no. 3, pp. 115-152, 1997
[45] NGMN, “Recommendation on SON and O&M Requirements, version 1.1,” 2008
[46] D. Fagen, P. A. Vicharelli, and J. Weitzen, “Automated Wireless Coverage Optimization With Controlled Overlap,” IEEE Transactions on Vehicular Technology, vol. 57, no. 4, pp. 2395-2403, 2008
[47] L. Nagy, “Indoor Radio Network Optimization Using Multi Level Hierarchic Method,” in IEEE WCNC, 2009
[48] 3GPP TS 32.541 v1.4.0, “Telecommunications Management; Self-Healing OAM; Concepts and Requirements”
[49] D. Bertsekas and R. Gallager, Data Networks, Prentice-Hall, 1992
[50] [Online]. Available: http://www.cs.washington.edu/research/networking/rocketfuel/
[51] B. Fortz and M. Thorup, “Optimizing OSPF Weights in a Changing World”, IEEE JSAC, 2002
[52] Y. B. Reddy, “Genetic Algorithm Approach in Adaptive Resource Allocation in OFDM Systems”, International Joint Conferences on Computer, Information and Systems Sciences and Engineering (CISSE 06), 2006
[53] Y. B. Reddy, N. Gajendar, P. Taylor, and D. Madden, “Computationally Efficient Resource Allocation in OFDM Systems: Genetic Algorithm Approach”, Fourth International Conference on Information Technology (ITNG'07), 2007
[54] W. Yongxue, C. Fangjiong, and W. Gang, “Adaptive subcarrier and bit allocation for multiuser OFDM system based on genetic algorithm”, International Conference on Communications, Circuits and Systems, 2005
[55] R. Onvural, “Survey of closed queueing networks with blocking”, ACM Comput. Surv. vol. 22, no.2, pp. 83-121, 1990
[56] M. Amirijoo et. al., “Cell Outage Management in LTE Networks”, 7th COST 2100 Management Committee Meeting, 2009

[57] D. Fan, Z. Feng, L. Tan, V. Le, and J. Song. “Distributed Self-Healing for Reconfigurable WLANs”, WCNC'2010
[58] E. Amaldi, A. Capone, M. Cesana, and F. Malucelli, “Optimizing WLAN Radio Coverage”, IEEE Int. Conf. on Comm., vol.1, pp.180-184, 2004
[59] N. Agoulmine, S. Balasubramaniam, D. Botvich, J. Strassner, E. Lehtihet, W. Donnelly, “Challenges for autonomic network management”, First Conference on Modelling Autonomic Communication Environment (MACE'06), 2006
[60] B. Jennings et al., “Towards autonomic management of communications networks”. Communications Magazine, IEEE Publications, vol. 45, no. 10, pp. 112–121, 2007
[61] P. Flegkas, P. Trimintzios, G. Pavlou, I. Adrikopoulos, and C. F. Calvacanti, “On Policy-Based Extensible Hierarchical Network Management in QoS-Enabled IP Networks”, Policy 2001: 230-246
[62] S. Hariri, Y. Kim, K. Varshney, R. Kaminski, D. Hague, and C. Maciag, “A framework for end-to-end proactive network management”, Network Operations and Management Symposium, vol.1, pp. 280-286, 1998
[63] J. Strassner, D. Raymer, E. Lehtihet, and S. Van der Meer, "End-to-end Model-Driven Policy Based Network Management", Policy 2006
[64] C.K. Wang, “Policy-based Network Management”, WCC-ICCT, 2000
[65] C. Efstratiou, A. Friday, N. Davies, and K. Cheverst, “Utilising the Event Calculus for Policy Driven Adaptation in Mobile Systems”, Proceedings of the 3rd International Workshop on Policies for Distributed Systems and Networks, pp. 13–24, Policy 2002
[66] Yu Cheng et al., “A generic architecture for autonomic service and network management”, Computer Communications, vol. 29, no. 18, pp. 3691–3709, 2006

Abbreviations

3GPP	3rd Generation Partnership Project
AP	Access Point
ACM	Application-Centric Management
ASA	Autonomic Service Architecture
BS	Base Station
CAC	Call Admission Control
CS	Context Server
CAPEX	Capital Expenditure
CDF	Cumulative Density Function
COOP	Coverage Optimization Opportunity Coefficient
CSI	Channel State Information
CTG	Cluster Topology Graph
dB	Decibel
DM	Domain Manager
DRP	Dynamic Resource Allocation
EA	Evolutionary Algorithm
ES	Energy Saving
eNB	Enhanced NodeB
EMA	Exponential Moving Average
ETE	Energy-aware Traffic Engineering
ID	Identity
IP	Internet Protocol
FB	Functional Block
FRP	Fixed Resource Allocation
GA	Genetic Algorithm
GP	Genetic Programming
GW	Gateway
ISP	Internet Service Provider
IP	Internet Protocol
LB	Load Balancing
LSP	Label Switched Path
LTE	Long Term Evolution
LTG	Local Topology Graph
MCN	Multihop Cellular Network
MCS	Management Computing System
MILP	Mixed Integer Linear Programming
MINLP	Mixed Integer Non Linear Programming
MLU	Maximum Link Utilization
MPLS	Multiprotocol Label Switching
MS	Mobile Station
MT	Mobile Terminal
NES	Network Expert System
NPM	Network and Protocol Management
NO	Network Operator
OFDM	Orthogonal Frequency-Division Multiplexing

OFDMA	Orthogonal Frequency-Division Multiple Access
PDN-GW	Packet Data Network Gateway
QoE	Quality of Experience
QoS	Quality of Service
RAN	Radio Access Network
RSS	Received Signal Strength
RSRP	Reference Signal Received Power
RS	Relay Station
SA	Simulated Annealing
SE	Spectral Efficiency
SGW	Service Gateway
SNR	Signal to Noise Ratio
SM	Segment Manager
SOM	Self-Organising Map
SON	Self–Organised Network
SNR	Signal to Noise Ratio
SINR	Signal to Interference plus Noise Ratio
TCoSH	Trigger Condition of Self-Healing
TE	Traffic Engineering
UC	Use Case
UE	User Equipment (=Terminal)
UMF	Unified Management Framework

Definitions

Algorithm – *A concrete step-by-step procedure for calculation. It is an effective method expressed as a finite list of well-defined instructions for calculating a function. Algorithms are used for calculation, data management, and automated reasoning.*

Capability – *The ability to perform actions. It is the sum of know-how and capacity.*

Governance – *A high level mechanism which involves all functionalities necessary to address the gap between high-level specification of human operators' objectives and existing resource management infrastructures towards the achievement of global goals. It relates to decisions that define network expectations, grant control, or verify performance. It consists of either a separate process or part of management processes. These processes and systems are typically administered by a governing function.*

Method – *A general procedure for solving a problem. It is a series of steps or acts for performing a function.*

Model – *A system and/or a representation of postulates, data, behaviour, and inferences presented as a description of an entity or state of affairs. An example of an optimization with a model would be the optimization of the channel capacity in a wireless access network by changing the transmission power of the base station. The Shannon-Hartley theorem tells us that the increase of channel capacity monotonically increases with increasing total received signal power over the bandwidth; and the total received signal power is directly related to the transmission power. Hence the model tells us that if we increase the transmission power of the base station, the channel capacity can be assumed to increase. In this case the model is reflected in a formula, namely the Shannon-Hartley theorem. Furthermore, thanks to the autonomic increase, the described problem also belongs to the class of convex optimization problems where solutions can be found in a straightforward way without getting trapped in local optima. For the class of non-convex optimization problems with models, however, the situation is slightly more complex as there need to be ways to avoid local optima, but the model can still be used to check new parameter configurations before they are actually tried in the network.*

Network empowerment – *Embedded network ability and authority to access and manage information, resources for decision-making and execution elements for changes of network behaviour. It is an approach where management and control functions are distributed and located in or close to the managed network and service elements. The potential benefits are the inherent support for self-management features, higher automation and autonomicity capabilities, easier use of management tools and empowering the network with inbuilt cognition and intelligence. Additional benefits include reduction and optimization in the amount of external management interactions, which is key to the minimization of manual interaction and the sustaining of manageability of large networked systems and moving from a managed object paradigm to one of management by objective.*

Self-optimization – *Selection and adjusting best (network and/or service parameters or behaviours from some set of available alternatives and/or minimize or maximize a utility function by systematically choosing the values of the parameters from within an allowed set in an autonomous way. Self-Optimization is a process in which the system's settings are autonomously and continuously adapted to the traffic profile and the network environment in terms of topology, propagation and interference. Together with Self-Planning and Self-Healing, Self-Optimization is one of the key pillars of the Self-Organizing Networks (SON) management paradigm.*

Management Tool – *Means to produce a management function or to achieve a management task, but that is not consumed in the process. Informally the word is also used to describe a management procedure or process with a specific purpose.*

Use case – *A descriptor of a set of precise problems to be solved. It describes steps and actions between stakeholders and/or actors and a system, which leads the user towards an added value or a useful goal. A use case describes what the system shall do for the actor and/or stakeholder to achieve a particular goal. Use-cases are a system modelling technique that helps developers determine which features to implement and how to gracefully resolve errors.*