Deliverable D3.4

Cooperation Strategies and Incentives

Grant Agreement	257513	
Date of Annex I	25-07-2011	
Dissemination Level	Public	
Nature	R: Report	
Work package	WP3-Network Empowerment	
Due delivery date	01 December 2011	
Actual delivery date	16 December 2011	
Lead beneficiary	UCL	Marinos Charalambides, M.Charalambides@ee.ucl.ac.uk

Authors	UCL: Marinos Charalambides (editor), Alex Galis, Daphne Tuncer, Stuart Clayman UniS: Stylianos Georgoulas, Velmurugan Ayyadurai, Ning Wang, Chaojiong Wang ALUD: Markus Gruber, Siegfried Klein NKUA: Evangelos Kosmatos, Eleni Patouni, Apostolis Kousaridas, Nancy Alonistioti, George Katsikas, Alexandros Apostolidis, Panagiotis Spapis UPRC: Panagiotis Demestichas, Evagelia Tzifa, Kostas Tsagkaris, Vera Stavroulaki, Yiouli Kritikou, Nikos Koutsouris, Assimina Sarli, Andreas Georgakopoulos, Evangelos Thomatos, Aimilia Bantouna, Dimitris Karvounas, Panagiotis Vlacheas, George Athanasiou, Aristi Galani FT: Zwi Altman, Richard Combes, Xavier Gelabert Doran, Berna Sayrac VTT: Petteri Mannersalo Fraunhofer: Mikhail Smirnov NEC: Johannes Lessmann, Zarrar Yousaf, Paulo Loureiro TIS: Antonio Manzalini TID: Beatriz Fuentes ALBLF: Pierre Peloso
Reviewers	UniS: Stylianos Georgoulas TCF: Gerard Nguengang

Executive Summary

The main goal of the Task 3.4 “Cooperation strategies and incentives” is “identifying methods and strategies best-suited for cooperation of entities in the relevant use cases” (DoW). Towards this end, this deliverable, which is the first result of this task, presents the outcomes of the project’s ongoing effort in this direction. In contrast to deliverables D3.1 and D3.2/D3.3 that are the outcome of tasks T3.2 and T3.3, the main focus of D3.4 is not to provide methods or learning techniques to improve the performance of the individual optimisation engines of different entities in the network but to examine how such interacting entities can jointly (co)operate in a constructive way. In a way, these individual optimisation engines elaborated in isolation the tasks T3.2 and T3.3 are treated in a joint manner in the task T3.4.

More specifically following the identification of related problems in the project use cases through the use case definition and refinement process, where the role of cooperation can be considered imperative, this deliverable puts forward specific cooperation strategies to address these problems while ensuring at the same time that they are compliant with the existing UMF design and specification.

This deliverable addresses the role of cooperation strategies in three different domains and provides cooperation strategies that are tailored for these specific targeted areas; that are:

- Cooperation strategies for traffic engineering,
- Cooperation strategies for Self Organising Networks (SONs), and
- Cooperation strategies for network stability

Section 2 puts its focus on entities running traffic engineering optimisations; it explains the issues that may arise in such cases in a non-cooperative environment and puts forward specific cooperation strategies that can overcome these issues.

Section 3 focuses on the wireless access domain and provides both a mathematical model capable of coordinating interacting and conflicting functionalities as well as some practical examples as instantiations of this mathematical formulation.

Finally, Section 4 identifies and reflects on more generic network stability issues and presents design principle and methodologies as well as tools (formal verification) for pinpointing but also overcoming their detrimental effects.

The deliverable concludes by summarising the outcomes of these ongoing activities under task T3.4, their importance for addressing the project use case and achieving the overall project objectives and the task’s future steps.

In addition, extensive auxiliary entries in the appendix present details background and the state of the art analysis in the aforementioned areas as well as some more technical details of the proposed strategies in this deliverable.

Table of Contents

Foreword

This deliverable has been developed in the context of Work Package (WP) 3 “Network Empowerment”, which is dedicated to the study, design and evaluation of various algorithms with self-x and cognitive capabilities together with their requirements for their embodiment into network functions. Within the project, the long-range objective is twofold:

1. To demonstrate with concrete use-cases how self-management can be implemented, and thus propose methodologies to existing or emerging management issues of operators,
2. To identify the essential features of self-management functions, and help define the characteristics of a Unified Management Framework (UMF), that would allow a soft embedding of such autonomic functionalities. The work about the UMF is essentially covered by WP2.

Previous WP3 deliverables (D3.1-D3.3) have focused on optimisation issues and on issues related to network behaviour and knowledge. More specifically, deliverable D3.1 focuses on optimisation issues that arise in network management, while deliverable D3.2 and its public part D3.3 are concerned with some remaining aspects of network management, typically those related to observing the network behaviour, building knowledge about it and extracting aggregated information about events of interest, in order to drive control actions over the network. Due to the distributed nature of network management functionality, interactions between relevant mechanisms are inevitable, which can have a negative impact on the operation of the network. Deliverable D3.4 addresses this issue by investigating cooperation strategies that must take place in self-management functions in order to guarantee safe network configurations and that global management objectives are met.

1 Introduction

The work presented in this deliverable is concerned with embedded cooperation strategies, a necessary feature of distributed self-management solutions where decision making entities work together to achieve global high-level management objectives and at the same time ensure that their resulting configurations are consistent with each other. It investigates relevant issues that arise in both wired and wireless communication infrastructures and proposes methodologies and mechanisms, which aim to achieve safe and reliable configuration of network resources in a cooperative manner.

Network empowerment refers to the embedding of intelligence, in the form of control loops, inside network equipment to realise distributed self-management functionality. This can result to faster response to emerging network events and also to less management overhead when gathering monitoring data compared to centralised solutions external to the network. The joint operations of many optimisations that need to be performed are convoluted by the distributed nature of the algorithms that realise them, and by the number of objectives that should be satisfied. This is because the mechanisms implementing self-management functionality across the network can interact with each other, which can have adverse effects on the network operation. Lack of cooperation between distributed decision making entities, and mechanisms in general, can lead not only to reduced end-to-end performance but also to oscillatory behaviours and unstable configurations.

The presented work focuses on specific problems identified in the use cases, defined in WP4, that arise as a result of the decentralisation of management functionality and the need of concurrent deployment of optimisation mechanisms. A set of solutions are proposed, which, through cooperation of network entities, can allow self-management capabilities to coexist harmoniously under different objectives. Furthermore, in each of the subsequent core sections we describe how the proposed functionality maps to specific components of the UMF. The latters are a framework and architecture for the management of future networks and services being developed by the project.

The organisation of this document is based on the work of individual WP3 Task Forces (TFs) with each core section presenting the work carried out within a specific TF. Section 2 presents the work of task force TF3.4C, which focuses on the cooperation between Traffic Engineering mechanisms operating at different entities in the network. This work covers coordinated approaches in cellular and core IP networks for the purpose of optimising resources, cooperation techniques between P2P and core networks for cost-efficient resource utilisation and coordination of radio parameter configurations so that interference in multi-hop cellular networks can be avoided. Section 3 presents the work of task force TF3.2A that evolves around the SON interaction problem in LTE networks. Various conflicts that can introduce instabilities and oscillations in LTE use cases are identified and a mathematical model for coordinating self-optimising functionalities is described. Application of the proposed approach is demonstrated through representative examples regarding specific SON algorithms. The work of task force TF3.3F on network stability is described in Section 4. Potential instabilities are identified, which can arise due to configuration inconsistencies introduced by multiple control loops in distributed settings. Issues related to context dissemination and to reliable routing of traffic in the presence of multiple optimisation objectives are investigated. Finally, concluding remarks are presented in Section 5.

Cooperation strategies as presented in this deliverable represent the artefacts and design for the UMF functional blocks in the emerging UMF specification. The follow-up of this document will account for the progress made in solving each problem of the relevant task forces as they relate with the project 6 use-cases (see Deliverable D4.1) and the developments of UMF (see deliverable D2.1), and will progress towards a more unified vision of self-management cooperation strategies. Finally, the methods used to solve the identified problems represent the current working assumption; the suitability of the various methods will be assessed in future work.

2 Cooperation Strategies for Traffic Engineering

2.1 Introduction

Traffic engineering (TE) itself is a well-studied area; in principle it refers to a set of approaches that try to “map” traffic demands onto the available physical topology and resources with the objective to improve certain performance objectives, e.g. minimise delay and jitter, maximise throughput, balance the load on the available paths between source-destination pairs etc.

While standalone TE approaches have been thoroughly studied, what has not been adequately studied, however, is the joint operation of such approaches and mechanisms; that is, when a TE approach is applied in a network entity it does not take into account the effect that it can have to other networking entities and their objectives. This means that a TE-driven optimisation at one entity may degrade the performance at another entity, which in turn may counter-react in response to this degradation, improve its own performance but -in turn- degrade the performance at another entity.

This means that in end-to-end delivery chains, such as the ones addressed by UniverSelf, the lack of coordination and cooperation between TE mechanisms operating at different networking entities can lead not only to reduced individual and end-to-end performance but also to oscillatory behaviours and unstable situations. In the rest of this section we will present cooperation strategies put forward and developed in the context of the UniverSelf project that aim to streamline the operations between various entities in an end-to-end delivery chain and lead to better overall performance.

This coordinated and cooperative behaviour is envisioned to be guided by the Unified Management Framework (UMF) [5], which should provide the policies and constraints that will drive the behaviour of -otherwise- non-cooperative TE mechanisms. It is worth noting that the term network entity can vary from an individual router to a whole network segment meaning that the cooperation strategies presented in the rest of this section can cover a wide granularity of networking scenarios and can be reused/combined in various instantiations of end-to-end chains, as these may be created according to the deployment plans of an operator.

2.2 Related Work

TE approaches are usually categorised based on two metrics that are; the *number* of the TE decision making entities within a network segment and the *dynamicity* of the decision making mechanism itself.

With respect to the first metric, TE approaches can be either centralised or distributed; in the former there exists only one decision making entity whereas in the latter there can be multiple decision making entities. While centralised approaches –theoretically- provide an ideal situation, that is one single entity with global view and knowledge takes the decisions, they do suffer from scalability problems and also resilience (single point of failure) can be an issue. Distributed solutions are more scalable, however -even within a single network domain- the need for coordination between these multiple decision points arises.

With respect to the second metric, TE approaches can be either offline or online. In the former case the TE optimisations are run at long time-scales (e.g. hours, days or even weeks/months), either pre-determined ones or triggered by certain events. In the latter case, the TE optimisations are run at shorter time-scales (e.g. minutes) triggered by more dynamic events.

It is common for offline centralised approaches to perform *global (network scale)* optimisations and drive the required reconfigurations since the infrequent invocation of these optimisations allows for network-wide information to be collected and analysed and also for sophisticated optimisation engines to be used, since the time needed to run the TE optimisation and derive the reconfiguration that has to be enforced is not of crucial importance. On the other hand, distributed online TE approaches often have to rely only on partial network information and enforce *local* reconfigurations, relying as well as on simpler optimisation engines.

For a more comprehensive state of the art analysis on TE approaches the interested reader can refer to Section 9.1.1 of the appendix.

It is also apparent that the nature of the TE approaches directly influences the nature -and viability- of the required coordination and cooperation between the entities running these TE approaches. E.g. in case of two domains running centralised TE optimisations, coordination is needed between the two distinct decision making entities, whereas in the case of distributed approaches the number of the entities that need to cooperate and coordinate their decisions increases accordingly.

Before proceeding to a more detailed analysis of the considered and developed cooperation strategies for TE in the context of UniverSelf we will briefly present the relation and mapping of these mechanisms to the UMF.

2.3 Relation with UMF

Figure 1 gives an overall view of the mapping to UMF functional blocks (FBs). The logic behind the selection of the specific FBs is as follows: each individual TE approach maps onto the Solution Selection and Elaboration FB (SSE_FB) and relies on monitored information from the Monitoring FB (MON_FB) and triggers for re-optimisations coming also from the Solution Evaluation and Assessment FB (SEA_FB).

The cooperation strategies reside on the Cooperation FB (CO_FB) and based on some wider-scale monitored information and evaluation of the solutions they can influence the parameters and objectives of the individual TE engines.

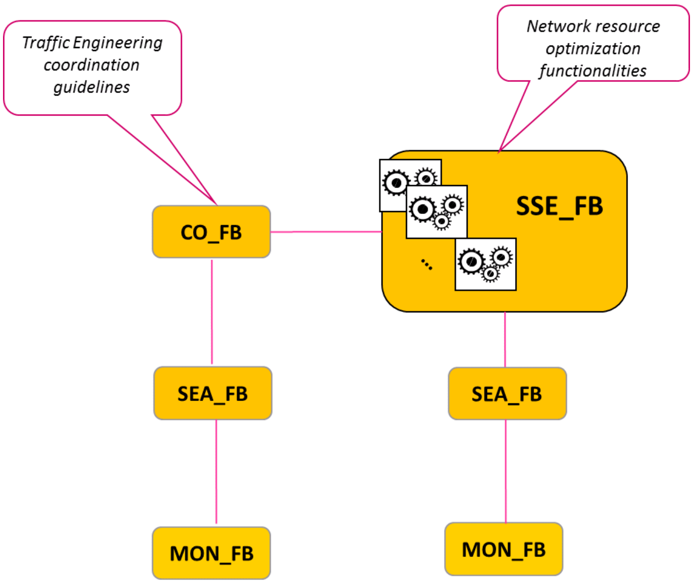


Figure 1: Relation with UMF functional blocks

2.4 Target Areas and Methodologies

In this subsection we present in detail the specific areas for which cooperation strategies have been introduced and considered to improve performance.

In brief the following networking scenarios, corresponding to valid project Use Case problems, have been considered.

1. Cooperation between enodeBs' and relay nodes in multi-hop cellular networks for the purpose of interference minimisation and user throughput maximisation
2. Cooperation between core network segments and overlaying service (P2P) networks for improving both network and P2P objectives, depending on which one is deemed more important
3. Cooperation between routers within a core network segment for the purpose of load balancing, and
4. Cooperation in cellular networks with the objective of load balancing across all parts of a cellular network (access, backhaul and core).

2.4.1 Dynamic Inter-Cell Interference Coordination in Multi-hop Cellular Network

2.4.1.1 Introduction

One of the challenges of today's wireless operators is the way to handle the impact of increase in traffic, demand for high bandwidth applications and interference mitigation through the use of self-organising network (SON) functionalities for dynamically coordinating radio parameter configurations. Using orthogonal frequency division modulation (OFDM) systems, orthogonality characteristics of the subcarriers will combat inter symbol interference (ISI) by using synchronised reference signals inside a cell. Hence, multiple subcarriers can be combined to provide high bit rate. With OFDMA, network access control to provide higher spectrum efficiency is performed by reusing the same sub-carriers in all neighbouring cells with flexible frequency allocation. However, the challenge to support more users with required Quality of Service (QoS) in the present generation wireless networks requires appropriate interference avoidance techniques. Interference avoidance requires dynamic resource allocation with coordination control for frequency reuse inside and with neighbour cells. For example, when two users in different cells, use the same frequency resource (chunk) simultaneously, then the Signal-to-Interference and Noise Ratio (SINR) associated with these chunks can drop to a very low value, resulting in a bad resource utilisation and lower system performance.

Multi-hop cellular networks, with multiple relay nodes (RNs) forming small cells and coordinating with donar cells with Evolved NodeBs (eNodeBs) is one of the approaches proposed in LTE-Advanced systems to improve network performance with spectral efficiency. These kinds of RNs are usually operator-deployed access points that transmit data from eNodeBs to end user equipment (UE) and vice versa, as shown in Figure 2.

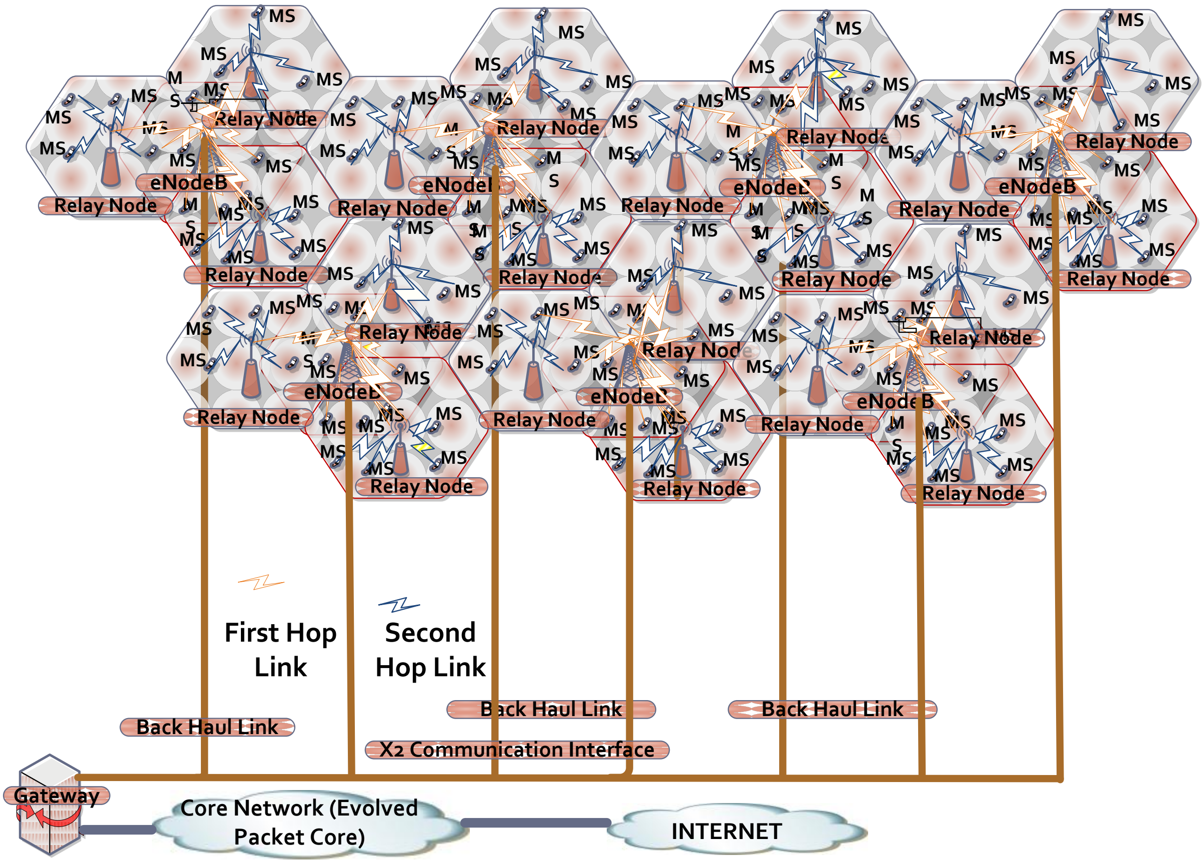


Figure 2: Network layout

The radio resources are allocated for the relays in an orthogonal manner and in-band half duplex transmission based on a time slot basis. The first time slots are reserved for the direct and relay UE transmission. The second time slots are reserved for the direct UEs and relay backhaul transmission. The users are clustered into direct hop transmission users associated directly with eNodeBs and multi-hop transmission users associated with RNs. RNs are positioned so as to increase signal strength and to improve reception in poor coverage areas and dead spots in the existing networks (e.g., cell edges, tunnels). This kind of deployment facilitates ubiquitous coverage and better capacity by making the wireless fading channel quality due to its multipath nature to be better with the received signal quality for cell edge users. However, it adds one more dimension of complexity for the radio resource configuration due to the need for multiple frequency reuse and information exchange between RNs and eNodeBs. Thus, a multi-hop cellular network makes inter-cell interference coordination more challenging.

2.4.1.2 Dynamic Inter-Cell Interference Coordination Framework

The multi-hop cellular network system consists of a network layout representing the minimum size LTE configuration with 7 cells having eNodeBs in the centre with sites of 3 sectors covered with extra RNs as shown in Figure 2. Each eNodeB is equipped with 120° directional transmit antennas, while the RNs and UEs receive antennas are considered to be omni-directional but with multiple inputs and multiple outputs (MIMO) systems. The system uses cell-specific orthogonal reference signals. UEs know the reference signals of neighbouring sectors and they are able to determine interference separately. It is evident that for a downlink transmission to a UE in any sector of RNs, one of its first-tier sectors is likely to be the most dominant interferer. Let us consider Figure 3 as an example.

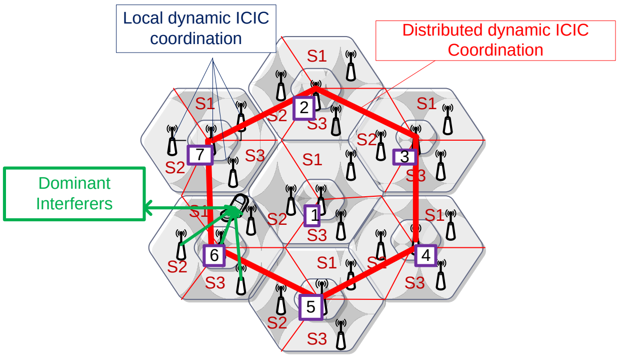


Figure 3: Example of inter-cell interference scenario

Due to relative locations and antenna directivity, a UE in RN in sector 1 of eNodeB6 may receive the most dominant interference from sector 2 of RN or sector 3 of RN (depending on the UE location), or sector 2 or sector 3 of eNodeB1, or from sector3 of eNodeB7 or sector 1 or sector 2 of eNode6. A cell edge UE experiences lower path loss due to RNs placement and receives significant interference from the sectors of the nearby cells. Also, UEs closer to the serving sector may experience severe interference from the neighbouring sectors of their own cell. As a consequence, these UEs are susceptible to see more poor quality chunks having low SINR. Any optimal or sub-optimal allocation scheme that aims to maximise network throughput may overlook such disadvantaged UEs as they are less attractive to contribute to the total throughput compared to those closer to the RN or eNodeBs. Therefore, it is very important to avoid interference on such UEs in order to guarantee their minimum required rates.

To determine frequency resource (chunk) restrictions optimally, a utility maximisation problem is formulated; for more details the interested reader may refer to Section 9.1.2.1 of the appendix.

The proposed inter-cell interference coordination scheme is comprised of two separate algorithms; one is located at the eNodeBs and RNs level that prepares the chunk restriction requests and the other resides at the central controller that resolves restriction request conflicts. The working principle of the scheme can be given as below:

- UEs send channel state information (CSI), including information on two most dominant interference received from their first-tier sector RNs to the serving RNs or eNodeBs they are attached to.
- Each eNodeB prepares a utility matrix based on the channel states and UEs service status
- Each eNodeB iteratively apply genetic algorithms to the utility matrix to find chunk restriction requests for each of its dominant interferer neighbours.
- Each eNodeB forwards the restriction request list to the central entity.
- The central entity processes requests from all involved eNodeBs and resolves conflicting requests based on the utility values in an optimal manner.

The central entity then forwards to each eNodeB a decided set of chunks that are to be used by its scheduler.

2.4.1.3 Performance Evaluation

A total of 7 cell sites (i.e., 21RNs + 7 eNodeBs giving 28 hexagonal regions) are considered in the simulations as shown in Figure 3. The UEs are randomly distributed in the 7 centres and 21 RN sectors within a minimum and maximum radius in each sector. The system parameters used in the simulations can be found in Section 9.1.2.1 of the Appendix.

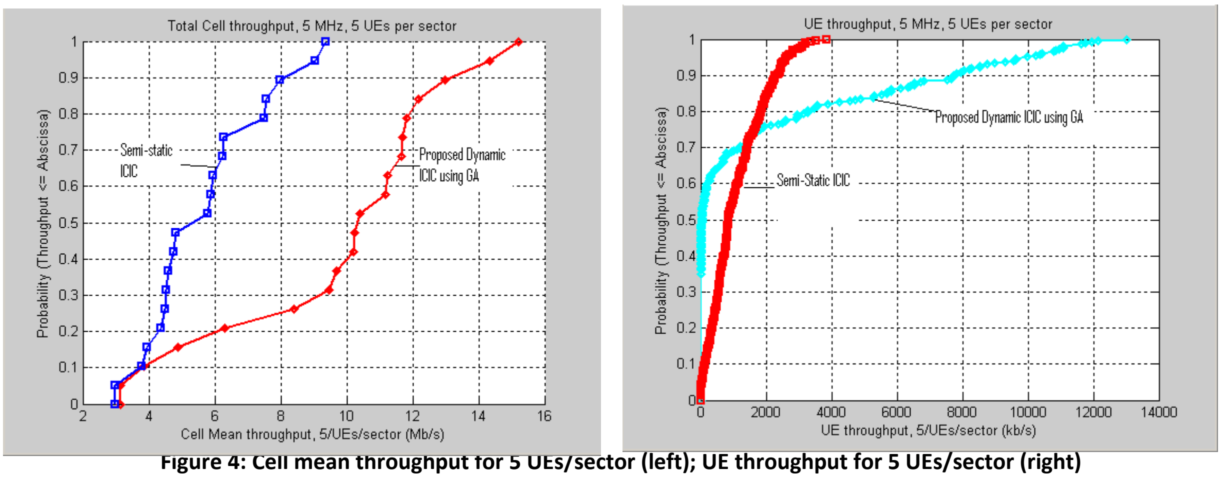


Figure 4: Cell mean throughput for 5 UEs/sector (left); UE throughput for 5 UEs/sector (right)

Figure 4 shows the Cumulative Distribution Function of average cell mean throughput and of average UE throughput. As shown, the proposed dynamic ICIC scheme provides considerable performance improvement compared to the semi-static ICIC -which does not involve any coordination after the initial network planning- in multi-hop cellular networks. Using the proposed dynamic ICIC method, the cell mean throughput gain increases around 40% to an overall of 15 Mbps with very few users suffering from throughput reduction compared to semi-static ICIC methods. It also provides significant UEs throughput gain of around 6 Mbps increases compared to the semi-static ICIC method with an increase of 70% in overall UEs throughput.

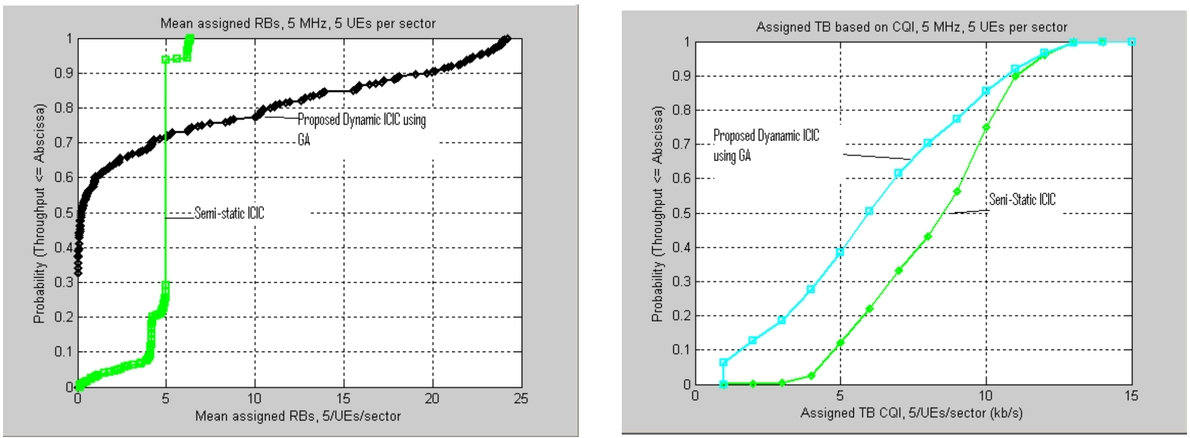


Figure 5: Mean assigned chunk/sector (left); Assigned TB CQI/sector (right)

As shown in Figure 5, the proposed dynamic ICIC scheme provides considerable performance improvement regarding the mean assigned frequency resources compared to semi-static ICIC by increasing the chunk usage around 75%. Using the proposed dynamic ICIC method a maximum full chunk usage is possible with slight reduction due to centralised control. It also provides significant transport block CQI with the proposed dynamic ICIC method achieving a 30% increase compared to the semi-static ICIC method, which does not involve any cooperation through a central entity.

2.4.2 Cooperative-TE between Core Network Segments and Specialised Service Networks

2.4.2.1 Introduction

TE techniques have been extensively investigated for achieving cost-efficient resource utilisation in networks. The basic TE strategy is to perform optimised traffic routing and forwarding configurations in order to achieve desired network performance targets. Optimised traffic delivery paths are computed based on the mapping of the static long-term traffic demand (i.e. the traffic matrix - TM) onto the underlying network topology. To do this, the network operator needs to accurately predict the overall traffic demand between all ingress and egress routers. Nevertheless, with the increasing popularity of peer-to-peer (P2P) applications in recent years, the prevalent traffic patterns in operational networks have become increasingly difficult to capture. The key challenge for operators in dealing with traffic flows incurred by P2P overlays is the uncertainly in determining communication endpoints, which are under the control of end users. More specifically, given the ad hoc peer selection and content swarming behaviours, the overall P2P traffic demand has become much more difficult for network operators to forecast. As such, solely relying on a static traffic matrix in order to perform traditional application-agnostic traffic engineering (AATE) does not seem to be practically effective.

More recently, proposals have appeared suggesting cooperation between applications and the underlying network in order to achieve “win-win” solutions. In particular, the Application Layer Traffic Optimisation (ALTO) framework is currently being investigated by the IETF [4]. According to this approach, a dedicated ALTO server maintained by the network operator is responsible for providing necessary network information to the P2P overlay for supporting network-friendly peer selection. ALTO-based traffic optimisation is no longer completely network-centric, as is the case with the traditional AATE. Instead, traffic optimisation can be “indirectly” enforced at the application layer through manipulating traffic delivery paths in the network.

In this section we put forward fully-cooperative and semi-cooperative strategies to guide the interactions between TE optimisations at the network level and such overlaying specialised service networks and we validate through simulations the effect that these can have with respect to the satisfaction of their individual objectives, compared to selfish non-cooperative behaviours.

2.4.2.2 Cooperation Framework

We model the core segment (with AATE) and the P2P overlay (with fully-, semi-, or non-cooperative behaviours) as two autonomous and rational players who play best-reply dynamics: one player chooses the best response based on the other’s decisions in the previous round. Specifically, AATE aims to optimise the overall network performance through tweaking routing and/or forwarding decisions of customer traffic (including both P2P flows and non-P2P background traffic without differentiation) at the network layer. The changed routing/forwarding behaviours made by AATE for P2P traffic is then taken as input by the P2P overlay to take further actions.

In the fully-cooperative case, the P2P overlay aims to exploit opportunities (by reselecting partner peers) for further improving the network performance which could not be achieved through routing or forwarding optimisation by AATE in the previous round. That is the P2P overlay does not give any importance to its own application-layer objectives but only cares about the AATE objectives of the core segment. In the semi-cooperative case, the P2P overlay considers the best trade-off between application-layer requirements and network performance according to its own strategy. Finally in the non-cooperative case, the P2P overlay selfishly performs peer reselections according only to application-layer objectives. As a result, the overall traffic performance can be affected by the adverse impact from the non-cooperative P2P overlay.

The peer reselection action taken by the P2P overlay may further influence the overall traffic distribution within the network, possibly requiring further AATE operations from the core segment point of view. This is typically the case for semi- and non-cooperative cases where the objectives of the two players are inconsistent or even conflicting with each other. As a result, multiple rounds of bargaining interactions between the P2P overlay and AATE can be conceived. Throughout such an iterative process, the P2P overlay and AATE adjust their own decisions according to each other’s input from the previous round.

The AATE objective considered was the minimisation of network congestion (network cost) whereas the P2P application-layer objective was the minimisation of the delay between the interconnected peers. For more details the interested reader can refer to [6].

2.4.2.3 Performance Evaluation

In our simulations we considered the GEANT network topology [7] with 23 nodes and 74 links as the core segment. With respect to the P2P overlay we considered 20 concurrents P2P sessions with each session attracting up to 1200 peers. To evaluate the effect that the amount of P2P traffic can have on the performance of the various strategies we considered 3 cases where the P2P traffic accounts for 20% of the overall network traffic (low), 50% (mediums) and 80% of the overall traffic (high) respectively.

We model the three distinct P2P overlay behaviours (non-cooperative, semi-cooperative and fully-cooperative) to interact with AATE for 100 rounds in our evaluation. AATE initiates the interaction processes, so that AATE takes a turn at every odd round and the P2P overlay takes a turn at every even round. Since the fully-cooperative P2P and AATE have consistent objectives in optimising the overall network cost, such a scenario is obviously able to achieve the best results (out of the three scenarios) at the network side, and hence it is used as the reference one in order to evaluate the network performance of the other two. In addition to the network performance we also investigate P2P side performance, such as end-to-end delay. In this case, the non-cooperative P2P overlay, whose objective is solely to reduce delay), can be regarded as the reference point to evaluate the application-oriented performance of the semi- and fully-cooperative P2P behaviours.

2.4.2.3.1 Effect of Strategies on Network Cost

Figure 6 shows the overall network cost of the non- and semi-cooperative P2P interacting with AATE. These are the relative ratios against the fully-cooperative scenario whose overall network cost converges after the 2nd iteration according to our results, thanks to the completely consistent objectives between the two entities.

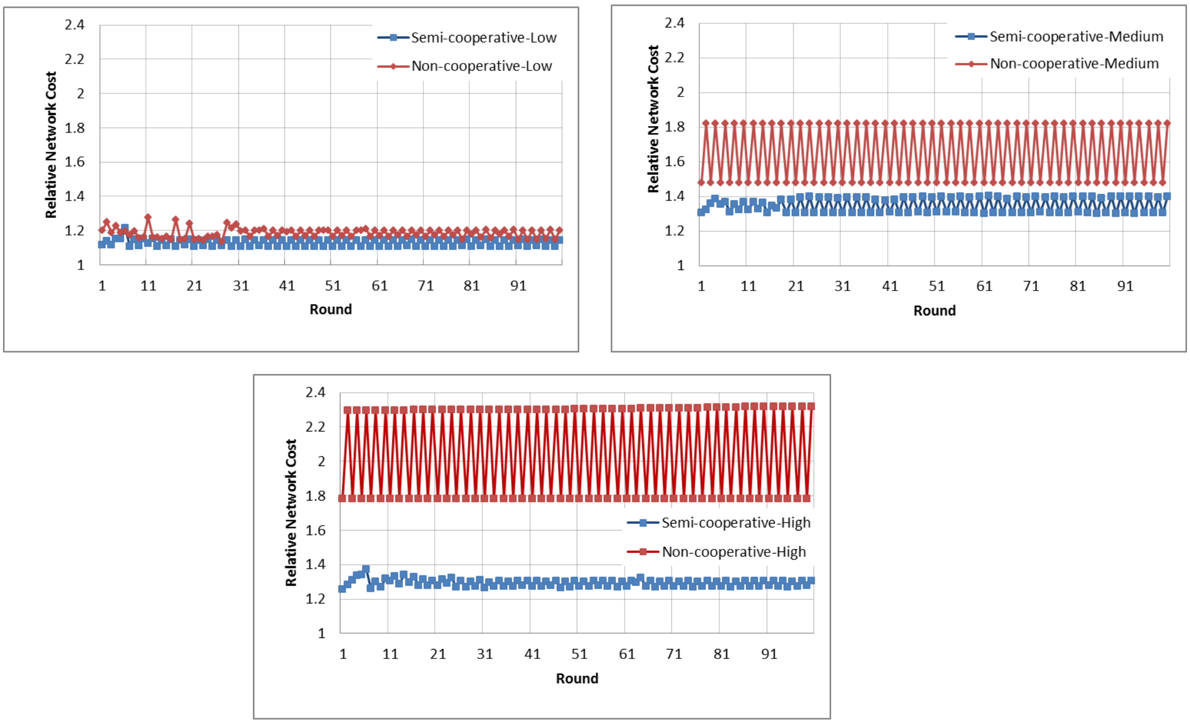


Figure 6: Network cost of the non- and semi-cooperative P2P versus fully-cooperative P2P

It is clear from Figure 6 that the overall network performance achieved by the semi-cooperative scenario is constantly better than the non-cooperative one, due to the more network-friendly objective. On the other hand, we can clearly observe the oscillation behaviours of the non- and semi-cooperative cases. The reason is that the interaction between AATE and the P2P overlay is processed in an interleaved manner where AATE first obtains the best network cost solution, but the outcome of the P2P selfish behaviour then leads to deteriorating network performance due to the inconsistent objectives with AATE. In response to the affected network performance caused by P2P overlay, AATE needs to re-compute the splitting ratios with the aim to regain the original performance. It is particularly worth mentioning that the oscillation degree (defined as the relative ratio between the maximum and minimum values across the 100 rounds) of the non-cooperative P2P becomes higher as the P2P traffic proportion increases (12%, 23% and 30% in the low, medium and high cases respectively). This is because with the increase of the P2P traffic proportion, more peers have the opportunity to perform selfish peer (re-) selections according to their own delay objectives. Such a significant traffic pattern change results in a larger optimisation space for AATE in the next round that aims to regain the original optimised performance.

2.4.2.3.2 Effect of Strategies on P2P Objectives

We now investigate the end-to-end delay for P2P sessions. The same evaluation methodology is adopted as to the one used in evaluating network-oriented performance. Nevertheless it is also worth mentioning that we use the delay of the non-cooperative P2P case as the reference point, given it is expected to achieve the best delay performance due to the selfish behaviour at the P2P side. On the other hand, it is also important to note that, like the network cost, the actual end-to-end delay for the fully-cooperative scenario converges after the 2nd round as indicated before, while the reference curve from the non-cooperative case has oscillations. This is because the optimised delay achieved by the selfish peer selection can be significantly impacted by the follow-up AATE operation that aims at network resource optimisation. It is not difficult to infer that the next round of peer selections pulls back to a re-optimised delay. Once again, we observe that the degree of oscillation in medium and high proportion P2P traffic is higher than that in low P2P traffic.

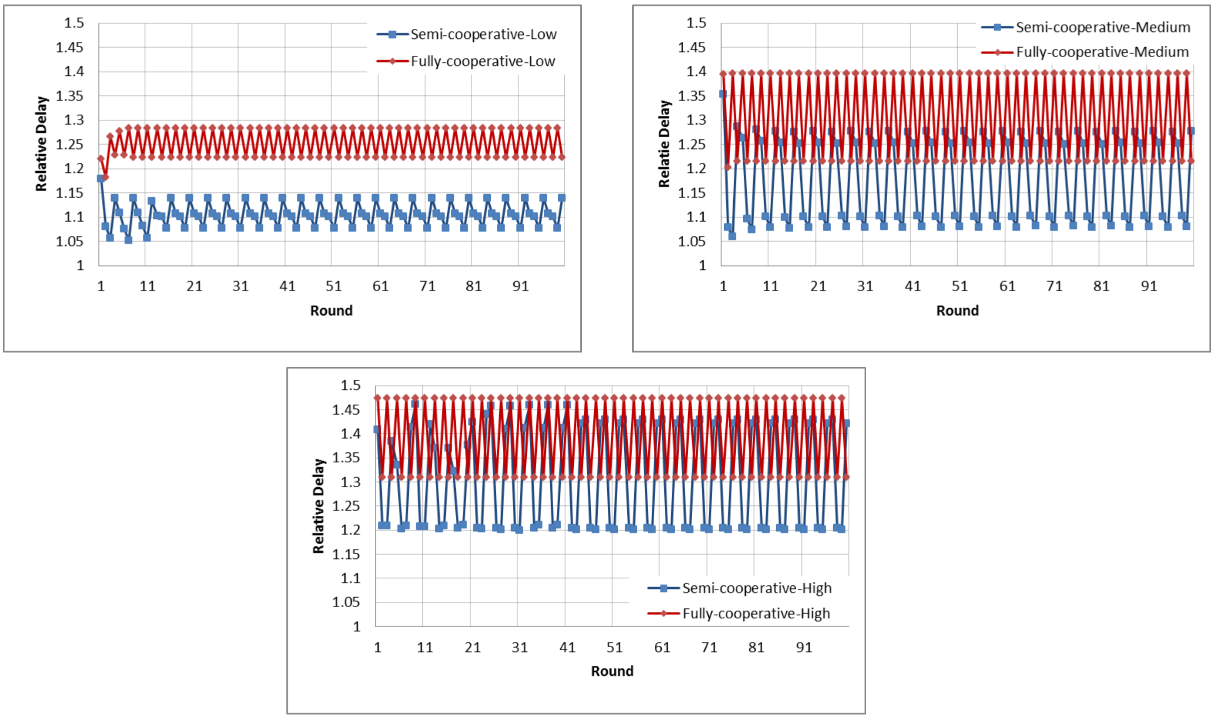


Figure 7: Delay for the semi-cooperative and fully-cooperative P2P versus non-cooperative P2P.

2.4.3 Decentralised and Adaptive Online Traffic Engineering

2.4.3.1 Introduction

Today’s Traffic Engineering (TE) practices mainly rely on off-line settings that use traffic demand estimates to derive network configurations. However, because of their static nature, these practices do not take network and traffic dynamics into account and can lead to sub-optimal overall performance. To cope with unexpected traffic variations and network dynamics, approaches that can dynamically adapt routing configurations and traffic distribution are required. Despite recent proposals to enable adaptive traffic engineering in plain IP networks [8][9][10], current approaches normally rely on a centralised TE manager to periodically compute new configurations according to dynamic traffic behaviours.

This section presents DACoRM (Decentralised Adaptive Coordinated Resource Management), a novel adaptive TE approach for IP networks in which traffic distribution is dynamically adapted according to real-time network conditions. The approach allows for the traffic between any pair of end points in the network to be balanced across several paths according to splitting ratios, which are (re-)computed by the network nodes themselves in real-time. The set of possible routes, enabled by multi-topology routing (MTR) [11], is determined by an off-line configuration process, e.g. [22], and is not modified by the adaptive scheme. The adjustment of the splitting ratios relies on run-time information about the network state and does not require any prior estimates of traffic demand. Most importantly, new configurations are not computed by a centralised management entity that has a global view of the network, but instead, the source nodes coordinate among themselves to decide on the course of action to follow. Each source node is responsible for adjusting the ratios of its locally originating traffic based on the result of the coordination.

2.4.3.2 Overview and System Design

The proposed online TE system performs adaptive resource management by dynamically adjusting the splitting ratios according to network conditions. The TE re-configuration actions performed are decided in a coordinated fashion between a set of source nodes forming an in-network overlay (INO) for communication purposes.

Based on the path diversity provided by configuring the different virtual topologies, the proposed approach controls the distribution of traffic load in the network in an adaptive and decentralised manner through re-configuration actions. The objective of this adaptive control is to dynamically balance the traffic load such that traffic is moved from the most utilised[1] links towards less loaded parts of the network. The traffic demand between each S-D (source-destination) pair is divided into n sets at source nodes, with each set being associated to one of the n MTR topologies T. The proportion of traffic assigned to each set is determined by splitting ratios, which are used to distribute incoming flows at source nodes.

Flows are routed to their destination according to the configuration of the topology they have been assigned to. Splitting ratios are not pre-computed by an off-line process as in other approaches, e.g. [12], but are instead adapted dynamically by the source nodes themselves, even without centralised control as is the case in [6]. New splitting ratios are computed by a re-configuration algorithm that executes only at source nodes, which allows them to react to traffic dynamics in an online fashion by adjusting the proportion of traffic assigned to each topology. If a link gets congested for instance, the nodes can automatically decide to re-configure the splitting ratios and hence move some of the load on that link to less utilised parts of the network. The adaptation is performed periodically in short time scales, every 5-10 minutes.

In order to realise the proposed adaptive resource management scheme a set of components need to be deployed at source nodes as depicted in Figure 8. Each source node S maintains locally both static and dynamic information related to each of the traffic flows originating at that node. Note that we refer to a traffic flow as the volume of traffic between a source node and a destination node. This information is stored in two tables that we call the Link Information Table (LIT) and the Demand Information Table (DIT) respectively. The LIT contains static information about the links traversed by the paths of all the locally originating flows. For each link, its stores the link capacity, references to the S-D pairs that use that link for routing their associated traffic flow in at least one topology, and for each of these S-D pairs, the involved topology(ies). Based on this information, source nodes can efficiently determine if a flow contributes to the load of a link in the network, and if so, on which topology(ies). The DIT contains dynamic information related to each flow entering the network at the source node. It maintains the current and previous splitting ratios assigned to each topology, as well as the associated traffic volume for the current time interval.

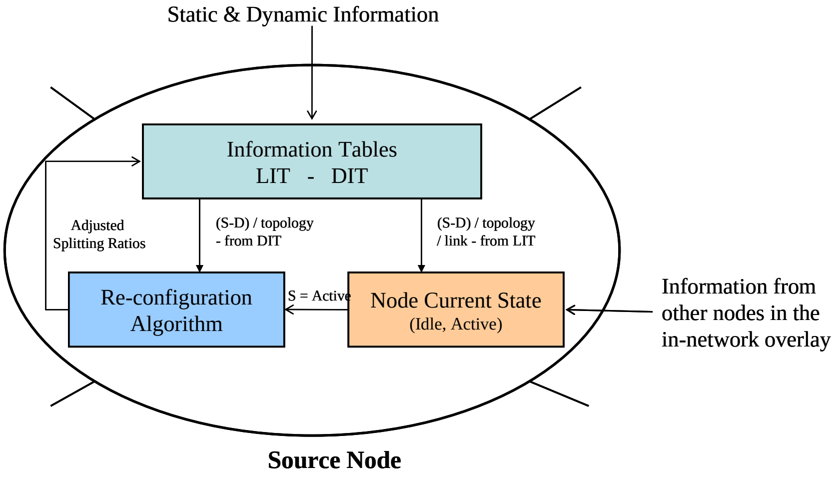


Figure 8: Components overview at the source node level

Based on information stored in the tables and on information received through the INO, a source node can determine its current state, which can be either idle or active depending on whether or not it needs to perform re-configuration. When in the latter state, the node executes the re-configuration algorithm over its locally originating flows to determine the new splitting ratios that can decrease the utilisation of the most utilised link

---

[1] The utilization of a link is defined as the ratio between the link load and the link capacity

in the network. If with these new splitting ratios no other link in the network gets overloaded, the adjusted splitting ratios are updated in the corresponding DIT and enforced at the next time interval.

2.4.3.3 Coordinated Resource Re-configuration

Performing a re-configuration involves adjusting the traffic splitting ratios for some of the S-D pairs for which traffic is routed across the link with the maximum utilisation in the network (noted *lmax*). This means that more traffic is assigned to topologies not using *lmax* to route traffic thus decreasing the traffic volume assigned to topologies that do use *lmax*.

The splitting ratios for each traffic flow are configured only by the corresponding source node. In realistic scenarios, links in the network are used by multiple flows and therefore, several source nodes may be eligible to adapt the ratios of flows traversing *lmax*. Due to the limited network view of individual source nodes, actions taken by more than one node at a time may lead to inconsistent decisions, which may jeopardise the stability and the convergence of the overall network behaviour. For instance, in the process of shifting traffic away from *lmax*, the different reacting nodes can re-direct traffic flows towards the same links thus potentially causing congestion. To avoid such inconsistent decisions, the adaptive scheme is designed so that only one source node is permitted to change the splitting ratios of one of its local traffic flows at a time. When an adaptation is required, the source nodes coordinate through the INO to select one of them that will compute and enforce the new ratios. This selection is based on source node contribution to the load of *lmax* in terms of number of flows. The selected node is responsible for executing the re-configuration algorithm over its locally originating traffic flows with the objective to re-balance the network load.

The INO of source nodes is built during the initial configuration of the network in an off-line manner. Its formation is based on the identification of ingress nodes in the physical network, i.e. the nodes which are potential sources of traffic. In the case of a PoP (Point of Presence) level network, for instance, each node is a potential source of traffic and would therefore be part of the INO. Each node N in the INO is associated with a set of neighbours – nodes that are directly connected to the INO – with direct communication only possible between neighbouring nodes.

Although different types of INO topologies can be used, e.g. ring, star, full-mesh, in this paper we concentrate on a full-mesh topology, where there exists a direct virtual link between all source nodes. Such a topology is depicted in Figure 9 where the four ingress nodes on the physical network are logically connected in a full-mesh. This topology offers a greater flexibility in the choice of neighbours with which to communicate since all source nodes belong to the set of neighbours. However, the choice of the INO topology may be driven by different parameters related to the physical network, such as its topology, the number of source nodes, but also by the constraints of the coordination mechanism and the associated communication protocol.

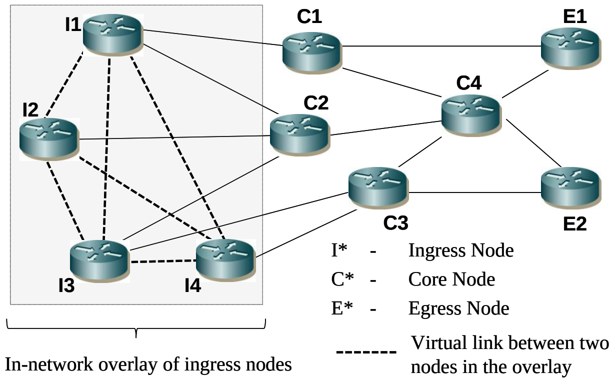


Figure 9: Example of a network and its associated full-mesh in-network overlay of ingress nodes

The overall objective of DACoRM is to balance the load in the network by moving some traffic away from highly utilised links towards less utilised ones. To achieve this objective, the proposed adaptive resource management scheme successively adjusts the splitting ratios of traffic flows through a sequence of re-configurations. At each iteration, DACoRM identifies the link with the maximum utilisation, *lmax*, and a set of other heavily utilised links, S_{HU}, in the network. S_{HU} is defined as the set of links in the network with an utilisation within a% of the utilisation of *lmax*. This information is shared among the source nodes in the INO and is used to select one of them that will compute new ratios. The selected source node is responsible for modifying the splitting ratios of one of its traffic flows contributing to the load on *lmax* such that: a) some traffic is moved away from lmax, and, b) the diverted traffic is not directed towards links in the set S_{HU}. The adaptation process terminates if a successful configuration cannot be determined or if it reaches the maximum number of permitted iterations (a parameter of the algorithm).

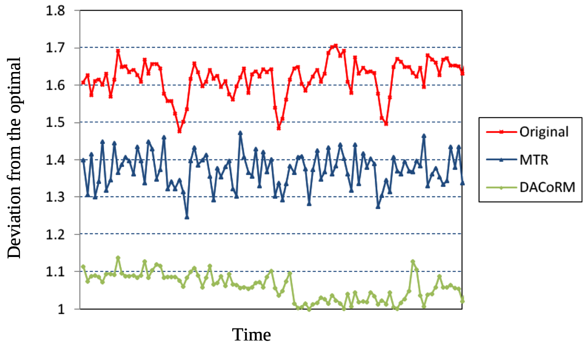
The adaptation process consists in adjusting the splitting ratios of some traffic flows such that the load-balancing objective is satisfied. At each iteration of this process, the selected source node is responsible for executing a re-configuration algorithm. The objective of the re-configuration algorithm is to determine if re-configuration can be performed on one of the local traffic flows. More precisely, the algorithm considers each local traffic flow at a time and tries to adjust its splitting ratios. These are adjusted such that the ratios related to the topologies that use *lmax* to route the traffic flow are decreased while the ratios related to the alternative topologies not using *lmax* are increased.

2.4.3.4 Experimental Evaluation

The performance of the proposed approach has been evaluated using the real PoP-level topology of the Abilene network and the traffic matrices available from [13] that provide traffic traces for 5 minute intervals during a 7 day period. The Abilene network topology consists of 12 PoP nodes and 30 unidirectional links. To analyse the performance of the proposed adaptive scheme in terms of maximum utilisation (max-u) in the network, we compare the results achieved by DACoRM with the results obtained by three other schemes: (a) Original scheme, where the original link weight settings are used in the original topology and no adaptation is performed; (b) MTR scheme, where the computed virtual topologies are used to provide path diversity and initial random splitting ratios are applied, but no further adaptation of these ratios is performed; and (c) Optimal scheme, where the TOTEM [14] toolbox is used to compute the optimal maximum link utilisation for each traffic matrix.

The objective of the comparison with the MTR scheme, where no adaptation is performed, is to evaluate the performance of the proposed adaptive resource management scheme, which performs periodic re-configurations, in terms of resource utilisation gain. For the experimentation, 4 topologies were used and reconfigurations were performed every 5 minutes.

Figure 10 presents the results of the evaluation and quantifies the average deviation from the optimum over a period of one week. This corresponds to more than 2000 traffic matrices that thus represents a wide variety of traffic conditions. The optimum is calculated using the TOTEM implementation with knowledge of the overall traffic demand. This is not possible to compute at run-time given that traffic matrices are not available. The proposed approach achieves a near optimal result with an average deviation of less than 10% from the optimum, while the other two schemes do not perform as well. The reason for MTR performing better than using only original link weight settings is that traffic is more evenly balanced over the different links in the network.



	Deviation from the optimal (%)
Original scheme	54 %
MTR scheme	34 %
DACoRM	7,2 %

Figure 10: Deviation of the maximum utilisation from the optimum in the Abilene network

On average, it takes 100ms for DACoRM to determine a new configuration, i.e. for computing and enforcing new splitting ratios. This amount can be considered negligible compared to the frequency at which the adaptive resource management scheme is invoked, i.e. every 5 minutes.

2.4.4 Collaborative End-to-End Load Balancing for Cellular Networks

2.4.4.1 Introduction

A lot of work has been performed on resource management and load optimisation aspects, e.g. see Section 9.1.1.4 of the Appendix. However, the solutions have mostly addressed the different parts of the network independently: there is already early standardisation on load balancing in the access network (e.g. scheduling, cell load balancing), there are algorithms on load balancing in the backhaul or core network (e.g. static/dynamic or centralised/decentralised traffic engineering) and even in the service domain (e.g. codec adaptation based on congestion).

Ideally, all the various areas of load optimisation (or more general: resource management) would be considered together. First, a single load balancing strategy might not be enough to relieve an imbalance situation, so a combination of actions might be needed. Second, performing an optimisation in one area can have significant implications for the load situation in others. E.g., if a bulk of user devices is forced to attach to a different base station to alleviate imbalances on the access side, this can considerably change traffic demands going from the old and new base station through the backhaul network to the core. In the worst case, some rebalancing must be done in the backhaul network as a consequence. A third reason why the different optimisation strategies should be considered together is that independent optimisations in each of the areas might not lead to a result which is optimal in a network-wide sense. Only coordinated action can ensure global optimisation.

2.4.4.2 Collaborative Resource Management Framework

To enable efficient and robust end-to-end resource management, we propose a collaborative resource management framework composed of a Collaborative Resource Manager (CRM) and a set of Resource Management Agents (RMA). Figure 11 illustrates the framework with reference to a generic cellular network infrastructure. The CRM has a global view of the entire system and it coordinates the different network domains for efficient utilisation of resources. The CRM is connected to the various network domains via well-defined interfaces termed as the RMAs, which act like a domain controller for local resource management.

The CRM is composed of three sub-layers following the related work [24]. The Monitor plane is responsible for the monitoring of context/content information at the respective network domain. The Control plane issues control and management commands to trigger and enable efficient resource utilisation amongst the various entities. It is based on sophisticated control algorithms ensuring collaborative resource management. The Management plane is used for the management of CRM functions related to alarms, rules and policies enunciation, charging/billing, etc.

The RMAs have the ability to coordinate and control the heterogeneous management system associated with different vendor devices at a particular domain. The RMAs specify rule-based resource management decisions locally at their respective network domains and continually submit resource status and management decision reports to the CRM's monitor plane. The CRM will be continually running control loops monitoring the decisions of the different RMAs. The CRM can then direct the RMAs to take specific resource management measures at their respective network domain. This orchestration of the RMAs by the CRM would enable optimum end-to-end utilisation of resources and the system achieving global balance.

In principle, the CRM can supersede the management decisions of the individual RMAs in situations where a resource adjustment in one domain may cause a bottleneck in another network domain. E.g., an RMA decision for shifting load from one access link onto another may overload an associated backhaul link causing service degradation and underutilising a core link that is part of the session.

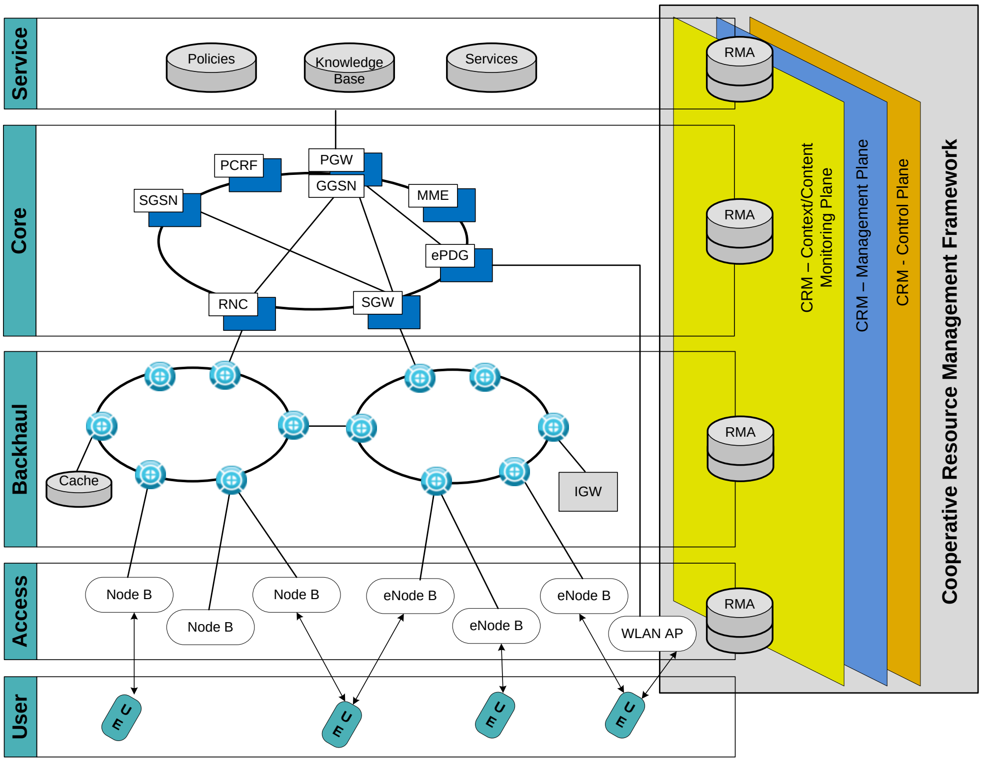


Figure 11: End-to-end resource management framework (high level system architecture) with respect to the reference network architecture

The main components of a CRM include resource management functions needed for efficient global load balancing and decision support functions required to take into account the resource conditions in all affected domains. Among the supported functionalities are e.g. connection management, link management, routing management, multi-homing, context management, QoE/QoS joint management, etc.

2.4.4.3 Algorithmic Framework

Based on the general architecture described above, CRM and RMAs are collaboratively responsible for ensuring end-to-end load balancing. Figure 12 visualises an end-to-end load balancing algorithm triggered either periodically (timer expiration) or on-demand (if a current load imbalance situation is detected). In case that the load imbalance was detected in one of the RMAs, the trigger for detecting the load imbalance itself may already give indications towards the possible root cause for the imbalance situation. In general, if a root cause can be identified by an RMA or the CRM, an efficient load rebalancing action can directly be undertaken (e.g. rerouting of traffic in the backhaul after a link failure).

If the root cause cannot be clearly identified, or if a local reaction strategy does not resolve the problem, the CRM has to become active and poll all involved RMAs for a cost/benefit evaluation. The backhaul RMA could, e.g., analyse possible LSP reconfigurations, the potential cost of executing them, and the expected load relieve. Once cost/benefit evaluations of the different RMAs are collected at the CRM, a sequence of load balancing actions across different RMAs is computed in increasing cost order. The CRM then requests the involved RMAs one-by-one to execute their domain-specific load balancing action. As soon as the imbalance situation is resolved, the process is terminated. If the combination of all computed actions does not resolve the problem, a manual intervention is needed and an alarm is passed to the network management system.

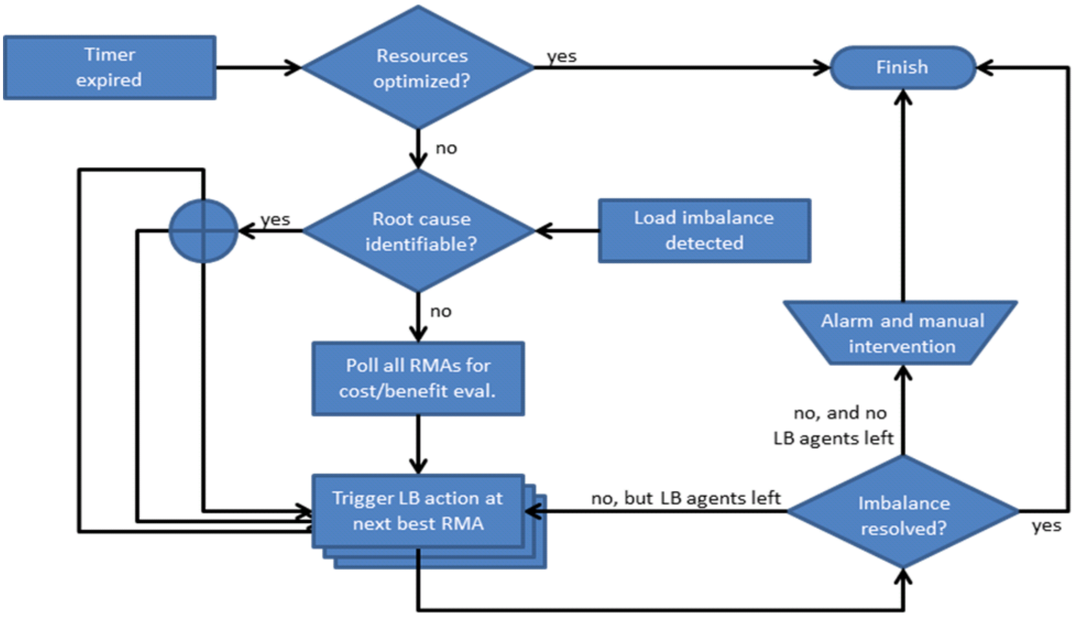


Figure 12: End-to-end load balancing algorithm

It is important to note the following issues. First, the interaction of CRM and RMAs clearly requires a domain-agnostic protocol. However, this is beyond the scope of this work. Second, having RMAs try to remediate imbalance situations locally before propagating them to the CRM helps to keep the majority of load imbalance reactions inside the affected domain without signalling and computation at the CRM. Using the CRM only for situations that cannot be resolved locally might come at the expense of reduced global performance. However, the proposed chain of responsibility leads to a natural balance between overhead and optimality. Third, by the nature of mobile networks, load gets more and more aggregated and thus smoother towards the core network. The fastest and highest load fluctuations will generally occur closer to the access. Hence, handling imbalances at in the access domain is the most vital part in the execution of the proposed approach as the aggregation and core network are not much affected anyway.

2.5 Results

The cooperation strategies presented in this section constitute a first step towards the creation of a toolbox of solutions to cover selected operator scenarios, as these are depicted in the project defined Use Cases. More specifically, taking into account valid Use Case problems, relevant cooperation strategies for traffic engineering were studied, developed and evaluated. These strategies will be further elaborated and tailored to the cooperation properties and needs of the Use Case problems. In addition, alternative strategies for the same problems will be also considered with the objective to identify the best strategy for a given cooperation problem.

Based on the mapping of the strategies and the “controlled” standalone TE approaches onto UMF FBs, the requirements for their embodiment into network functions will also be derived to allow for feasibility in their employment in practice and also maximise their reusability.

2.6 Discussion and Future Work

In this subsection we discuss the main findings of this section on the presented cooperation strategies for traffic engineering and provide some directions for future work.

2.6.1 Dynamic Inter-Cell Interference Coordination in Multi-hop Cellular Network

The observed performance gain in the proposed dynamic ICIC scheme is solely due to dynamic Interference coordination. Enhanced cell throughput in the proposed scheme can potentially allow a smaller number of RNs to cover a region yielding substantial savings in the deployment cost. As it does not require any frequency planning, the proposed scheme is not only effective for RN environment; it can be applied to future femtocell

BSs where user terminals are expected to experience severe interference from neighbouring macrocell eNodeBs.

In the future, further parameter tuning of the scheme will be performed to ensure its proper performance under a variety of load conditions.

2.6.2 Cooperative-TE between Core Network Segments and Specialised Service Networks

With respect to the cooperation framework between core network segments and specialised service networks we showed how the introduction of cooperative behaviour can affect the performance of network-layer and application-layer objectives in case of P2P networks built on top of underlying core segments. In particular, we identified the following two major factors that need to be specifically considered when deploying such approaches: (1) the *degree of consistency* in the optimisation objectives adopted by autonomous parties where some trade-off may be necessary, and (2) the specific proportion of P2P traffic that can be controlled by the application-layer optimiser.

In the future more complicated strategies and negotiation schemes can be considered in order to minimise or even nullify oscillations and ensure timely convergence to a stable and agreed by both parties configuration.

2.6.3 Decentralised and Adaptive Online Traffic Engineering

Source nodes, in the adaptive resource management scheme for intra-domain TE, coordinate among themselves through an in-network overlay to decide on the course of action to re-balance the traffic load across several paths according to network conditions. Unlike off-line TE approaches which rely on static configurations, our approach can efficiently deal with traffic and network dynamics by enabling adaptation of routing configuration in short timescales. The results of our experiments, based on the Abilene network and real traffic traces, show that our approach can efficiently achieve substantial gain in terms of network resource utilisation.

Future work will focus on the design of the communication protocol between source nodes in the in-network overlay. We plan to further analyse the impact of the different factors and parameter settings on the performance of our approach, both in terms of time-complexity as well as resource utilisation gain. We also plan to evaluate our approach in different network topologies and to compare its performance to other adaptive TE schemes.

2.6.4 Collaborative End-to-End Load Balancing for Cellular Networks

A collaborative framework for end-to-end resource management improving global usage of available resources was presented. Both the global orchestration of the entire system's resources and the interfaces to local mechanisms on the different domains are supported by the introduction of the Collaborative Resource Manager and the Resource Management Agents. This approach enables achieving improved resource usage at a global level while trying to handle as many remediation actions as possible at the local level, thus balancing optimality, scalability and signalling overhead.

While introducing the framework, we also described a taxonomy of possible load balancing actions together with the relevant context information according to the different network domains and gave an overview of important design considerations for any collaborative resource management framework.

Through the CRM, the network operators can achieve fault tolerance and resource scalability at a lower cost. This will translate into better services and increased revenue for the operators. The CRM can be looked upon as a shared resource and thus it can substantially reduce the management cost of operators who own and manage a specific network domain.

Future work will focus on developing the mechanisms and algorithms needed to orchestrate the RMAs and their resource management functionalities effectively. Different mechanisms may be launched simultaneously or in series. Consequently, this may change the settings of another running instance or trigger a new resource management mechanism due to feedback loops. Moreover, management mechanisms operate in varying time scales, from sub-seconds up to days, depending on configured network elements. All this means that finding the optimal orchestration of subsystems and their mechanisms is impossible in general. However, good engineering solution could be found. This will also require that information exchange between different entities has to be planned carefully as numerous different status and trigger messages need to be transmitted.

3 SON Interaction

3.1 Introduction

Harmful interactions between management functionalities may occur due to two reasons: either the two functionalities have control over the same network parameter, or the outcome of one functionality is a metric that is used as input by another functionality. In this section this interaction issue is illuminated from different perspectives. At first a comprehensive interaction system is derived for the LTE system from the corresponding 3GPP standard. In a next step, we go beyond LTE and analyse the interaction between coverage/capacity optimisation and load balancing between macro and pico cells in an LTE-Advanced HetNet (heterogeneous networks) context. Other investigated specific interactions include those of coverage/capacity optimisation and energy saving as well as coverage/capacity optimisation and interference coordination.

3.2 Related Work

The outcome and the deliverables of the European project Socrates is the current state-of-the-art on this topic. In Socrates, a combination of two strategies was proposed to avoid harmful interactions for 3GPP LTE: First of all management functionalities that are tightly coupled are treated as a package, i.e. with one solution, and secondly management functionalities that can in principle be separated from each other should be triggered by each other [38]. This section goes beyond Socrates' work by providing an integrated picture of all potential conflicts that may occur in a 3GPP environment and by approaching additional problems that occur in LTE-Advanced, namely in the context of HetNets.

3.3 Relation with UMF

Figure 13 gives an overall view of the mapping to UMF functional blocks (FBs). The operator provides high-level goals (GOV_FB) to the management system that are then translated (PDM_FB) into more technology specific goals that can be interpreted by the individual functionalities. The cooperation between the functionalities (CO_FB) is the core part that also reflects the specific strategy that is chosen (e.g. a trigger approach as described above). This functional block also hosts the SON functionalities themselves. Finally, monitoring (MON_FB) is a trust related elements in order to rule out instabilities and oscillations.

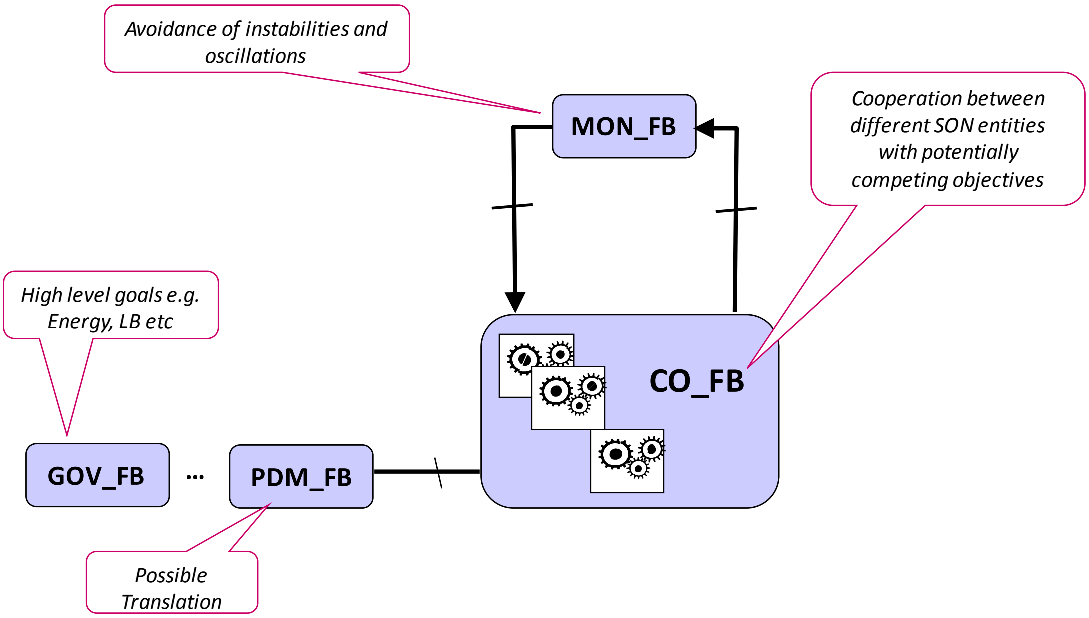


Figure 13: Relation with UMF functional blocks

3.4 Methods, Tools, and Models

3.4.1 SON Interaction in 3GPP

SON stands for self-organising networks and addresses capabilities that allow network nodes to organise and optimise themselves with only a minimum of manual intervention. This capability is particularly important for wireless network operators. New technologies appear on a regular basis (UMTS, LTE,...) while legacy technologies like GSM remain in the field. For each new technology, however, specialised service personnel are required so that cost for operational expenses can get excessively high. For this reason, a number of SON use cases was discussed in 3GPP and presented in [37].

However, these use cases cannot be treated individually and in an isolated way as already discussed in [38]. The reason for this are potential instabilities and oscillations caused by two types of conflicts: control parameter conflicts where multiple SON use cases access and change the same parameter, and metric conflicts where a SON use case influences a metric that is used as input for another SON use case. A clean solution of this problem is absolutely critical to operators´ trust in the network. If there is no such solution, problems may occur sporadically and due to the stochastic nature of wireless networks not necessarily reproducibly while the network is already up and running. In the following we will illustrate the interaction conflict for relevant SON use cases and we will then present possible strategies to deal with the interaction problem in general.

3.4.1.1 SON Interaction Conflicts of 3GPP LTE Use Cases

In UniverSelf, we have devised a figure (Figure 14) illustrating those interaction conflicts of the LTE SON use cases that are listed in [37] (and that we thus consider to be of a certain relevance to industry). The use cases themselves are depicted in yellow and are described in detail in [37]. The potentially affected control parameters are depicted in pink, whereas the influenced metrics are shown in light blue.

Out of the many potentially occurring conflicts, let us have a closer look at one particular conflict as an example, namely the conflict between mobility robustness optimisation and mobility load balancing, in order to thoroughly illustrate at least one instance of a SON interaction conflict. In LTE, there is a set of handover parameters that ensure that the success of a handover attempt is high while at the same time the occurrence of so-called ping-pongs, i.e. handing over a mobile forth and back from one cell to the other due to fast fading or other noisy effects, is low. For the purpose of ping-pong avoidance, there is a handover margin, which is a cell-wide hysteresis threshold, making sure that the mobile is not handed over as soon as there is a better server in sight, and there is a cell-individual hysteresis value called cell individual offset. The cell individual value can be controlled by the mobility robustness optimisation functionality to accommodate particularities of a handover region between a specific cell pair. The problem is that the cell individual offset is also controlled by the mobility load balancing use case in order to offload users from one cell to the others. Obviously, this control parameter can be used to geographically shift the handover region between two cells towards one cell or the other so that this “cell breathing” can be exploited to shrink cells in overload. Anything the mobility load balancing use case will do with this parameter, though, will be perceived as detrimental by the mobility robustness optimisation use case, which will – if no conflict avoidance mechanism is in place – revert the cell individual offset parameter to a value that seems ideal for mobility alone. This ultimately would result in an oscillation.

3.4.1.2 Conflict Avoidance

There are several strategies how to avoid conflicts between SON use cases. In [38] it was proposed to jointly optimise tightly coupled use cases and to otherwise use a trigger strategy, i.e. a use case completes the optimisation and then triggers (an)other use case(s) to re-optimise. In contrast to this approach, in [39] a separation strategy is proposed, according to which use cases are grouped in time domains. Within the time domains a joint optimisation would be performed; slower processes would have priority over faster processes as faster processes are more flexible to react.

Which time domain a use case may be attributed to may depend on several criteria. One criterion is how often an optimisation *can* be performed, for instance. To give an example, the optimisation of handover parameters cannot be performed very frequently since the choice of parameters needs to be evaluated against the handover performance, which in turn can only be determined after a relatively high number of actual handovers. Depending on the traffic in the cell, this may take a long time. Another criterion is the cost of an optimisation. A change of a parameter that has to be explicitly signalled to the mobile by a radio resource control message is more costly for the operator than a change that is communicated by a broadcast message that is sent out periodically anyway. In [39] there is also a thorough analysis of whether it makes sense to re-optimise at all at a given time opportunity to do so.

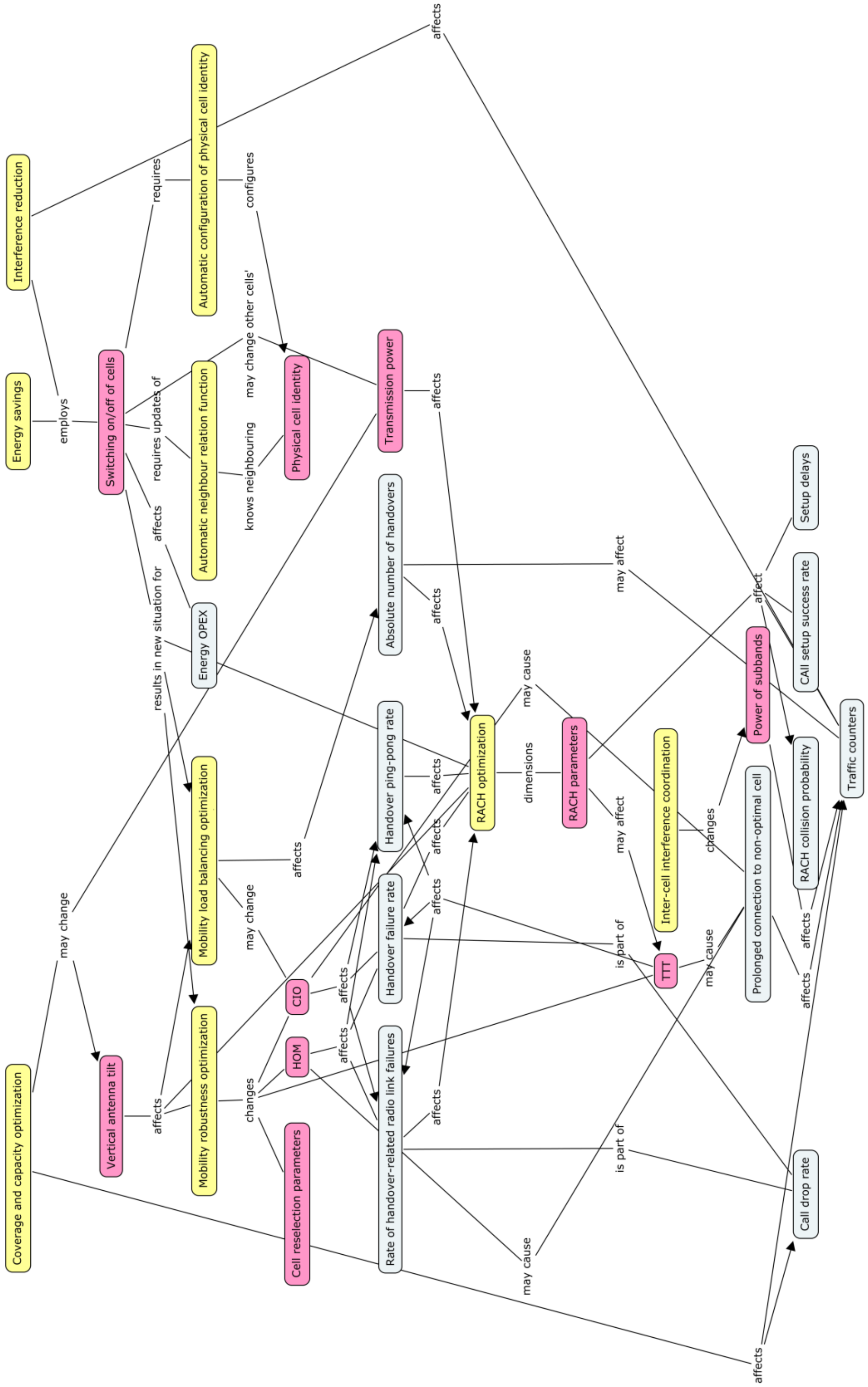


Figure 14: Interaction conflicts of SON use cases as they are listed in [37]

3.4.2 Coordination of SON Entities in a LTE-Advanced Network

This section describes a mathematical model for coordinating self-optimising functionalities. The mathematical model is based on the stochastic approximation method which iteratively optimises parameters of a dynamic system in the presence of noise [44]. Each SON is considered as a black box (denoted hereafter as "SON box"), which receives as input certain indicators and delivers as output a new set of KPIs (Key Performance Indicators) and (possibly) their gradient with respect to the network parameters. They are necessary for the operation of the coordinating entity. The outputs of the SON entities are jointly processed to define new parameter settings which guarantee the coordinated operation of the network. The operator can only access the inputs and outputs of the SON boxes. The proposed solution should allow coordinating a large number of SON boxes, and should be robust, namely the joint operation of the SON entities should not deteriorate the network performance during operation.

3.4.2.1 Mathematical Model

The block diagram for the SON coordination solution is presented in Figure 15. It has three components:

1. The vector of network parameters $\theta = (\theta_1,...,\theta_N)$ which, at time *t*, is denoted as $\theta(t)$. $\theta(t)$ is provided by the coordination entity (see below) and feeds the network (e.g. new radio resource management (RRM) parameter settings) including the SON entities.
2. *N* SON entities, with input: the vector $\theta(t)$ and certain indicators (performance and/or QoS), and output: one or several KPIs (Key Performance Indicator) and their respective gradient, $f_i(\theta), \nabla f_i(\theta)$.
3. A coordination entity with input: $(f_i(\theta), \nabla f_i(\theta))_i$, and a weight vector $w = (w_1,...,w_N)$ provided by the UMF, and output: $\theta(t+1)$ the new parameter settings.

The heart of the SON coordination algorithm is performed by the coordination entity which computes the following:

1. An aggregated utility function $U = U(w_1 f_1,..., w_N f_N)$
2. A parameter update, $\theta(t+1)$, using a stochastic approximation algorithm such as a stochastic gradient.

We note that the KPIs calculated by the SON entities can be highly noisy due to the traffic dynamics and the random characteristics of the radio channel. Hence all quantities $f_i(\theta), \nabla f_i(\theta), U$ and ∇U are corrupted with noise. The stochastic approximation algorithms provide a proof of convergence to a local optimum and hence the self-optimisation process is stable and robust (namely it does not degrade the utility function).

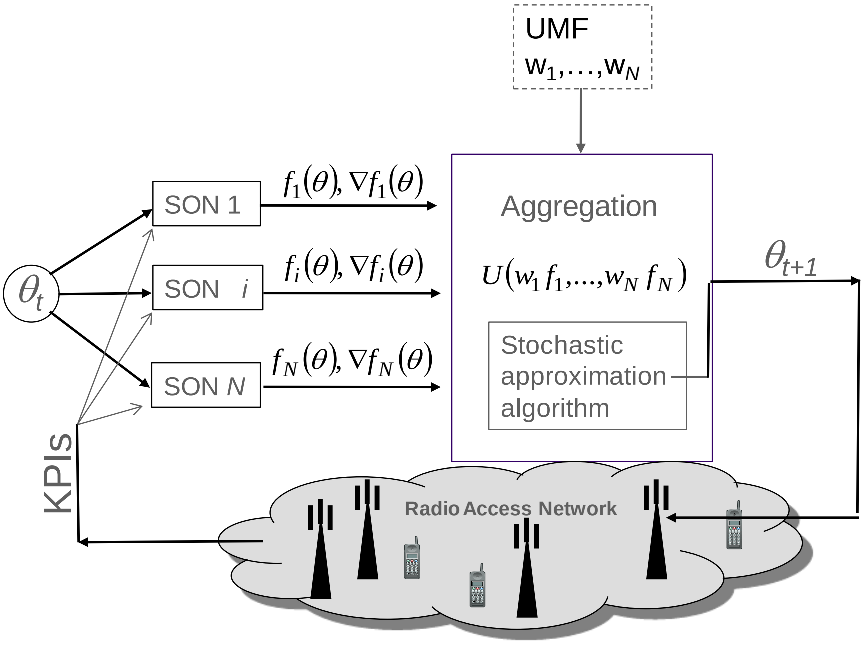


Figure 15: Block diagram for the SON coordination

3.4.2.2 Use Case Description

The use case comprises a heterogeneous network (HetNet) in LTE-Advanced with Macro- and relay stations (Figure 16). In-band relays are considered, namely the station to relay links are radio links sharing the same frequency bandwidth with the station (macro/relay) to mobile links. The two types of links are multiplexed in time or frequency.

Two SON functionalities are considered to guarantee optimal performance (Figure 17).

SON 1: Coverage Capacity Optimisation. Each relay station optimises its coverage by self-adapting its pilot power. When relay coverage is increased, it serves more traffic and hence offloads the macro-station which in turn transmits more data on the backhaul link.

SON 2: Traffic balancing. This self-optimising functionality balances the traffic between the backhaul and the direct (station to mobile and relay to mobile) links.

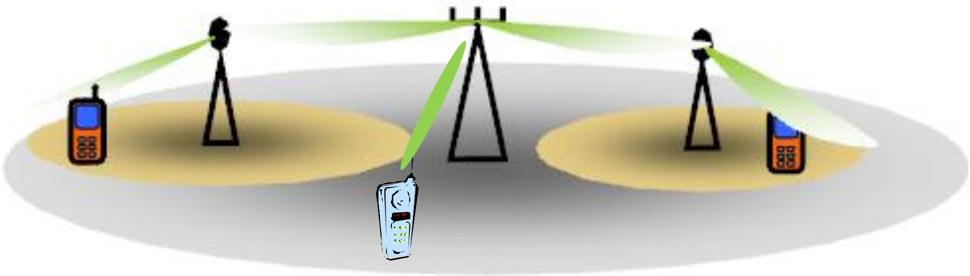


Figure 16: An LTE-Advanced HetNet with macro and relay stations

(a) SON 1: Coverage capacity optimisation

(b) SON 2: Traffic balancing

Figure 17: Two SON functionalities implemented in the LTE-Advaced HetNet

3.4.2.3 Results

An LTE-Advanced HetNet with macro- and relay stations is simulated using a dynamic network simulator. Figure 18 presents an LTE-Advanced site with a macro station and four relay stations. Each colour represents the coverage area of a station: sky blue for the macro, and orange, green, red and yellow for the four relay stations. The upper (lower) two graphs present a network with (without – the reference scenario) the SON functionalities activated. The colour code of the backhaul link is related to the quantity of backhaul traffic: blue: low traffic, red: high traffic.

The two graphs on the right show the worst (bottleneck) link usage (in %) indicator for both backhaul in blue and direct (station to mobile) in green. The usage metric represents the link load. In the lower graph, one can see a highly loaded direct (station to user) link while the backhaul link load is relatively low. The activation of the coordinated SON algorithm results in: (i) much lower (worst) direct link load, and (ii) balanced traffic between the backhaul and direct links. In terms of network performance, the decrease of the worst link load enhances the network capacity. Furthermore, this pushes away the traffic demand points from which network instabilities occur. It is recalled that network instabilities are the point where, due to high load, mobiles start accumulating, and QoS deteriorates (large blocking rates and long delays).

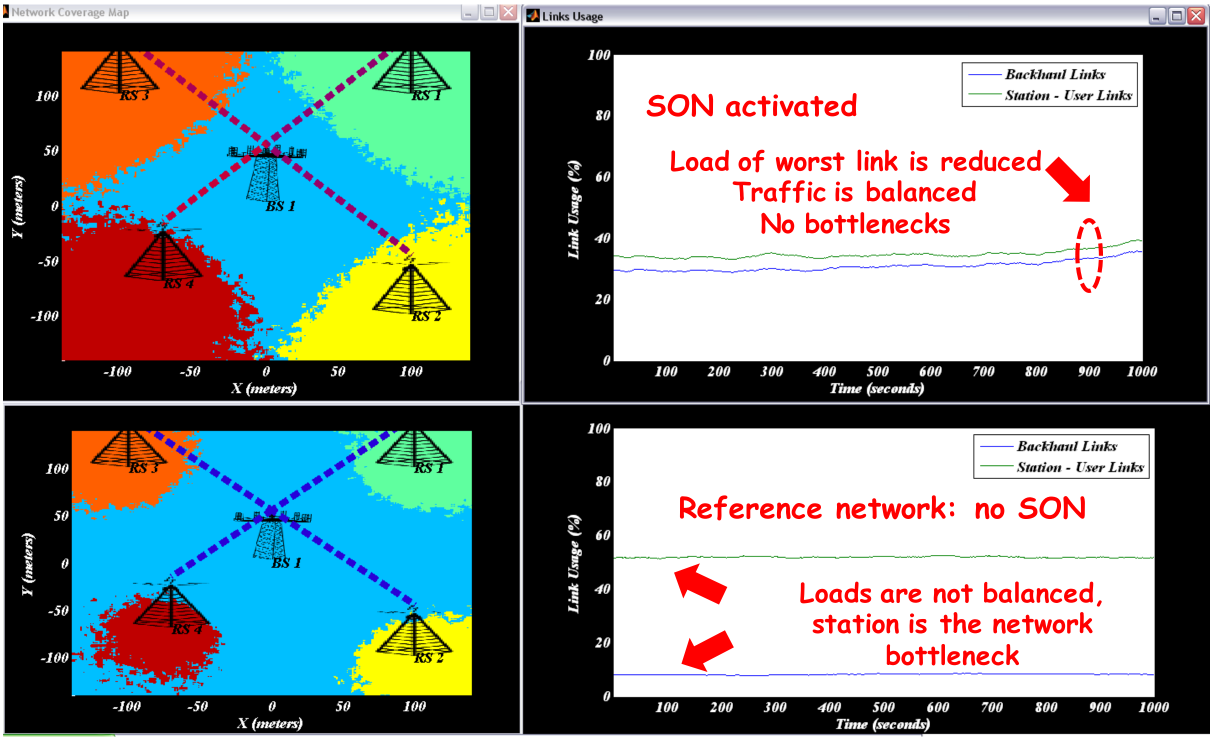


Figure 18: Two SON functionalities implemented in the LTE-Advanced HetNet

3.4.3 Interaction between Dynamic Access Point (AP) Switch On/Off and Energy Saving

In this section we study the impact between dynamic AP switch On/Off and energy saving. AP dynamic deactivation/reactivation is considered as an action under the Coverage and Capacity Optimisation umbrella. We have followed a simulation-based approach in order to identify the interactions/dependencies, focusing on the conflicts of metrics [39]. The algorithm for AP dynamic switch On/Off and the respective load balancing scheme have been presented in D3.1 [40], extending our previous work in [41]. The Concept map has been used for decomposing the specific case of 'Coverage and Capacity Optimisation' as well as for the description of SON interaction (Figure 19). The goal of this Use Case (UC) is the assessment of network capacity and APs overlapping factor (OF) on a specific geographical area. According to the assessment phase the deactivation or reactivation of one or more AP is decided. Deactivation of one or more APs could improve the network performance and avoid wasting radio and energy resources. The reactivation is selected when the network conditions need more capacity.

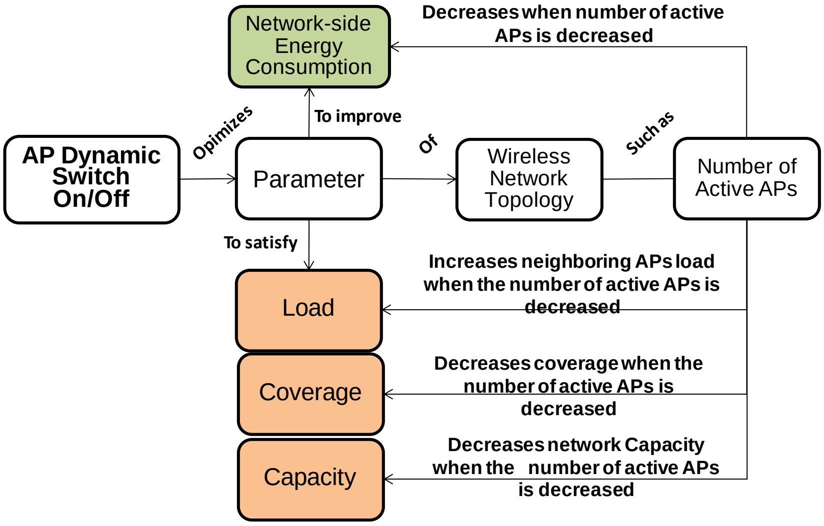


Figure 19: Coverage and capacity optimisation and energy saving

We have performed simulations by using OPNET Modeler v14.5 simulation software [42] in order to identify the effect of the specific optimisation action and other identified metrics. WiFi APs and the respective wireless station nodes operate at the frequency of 2.4GHz and the bandwidth of each channel is 22 KHz. All nodes implement the IEEE 802.11b wireless LAN station model. During the simulation the stations are fixed, while their handover capability is enabled. All access points have the same transmission power (0.002 W) and the data rate is set to 11 Mbps. The duration of the simulation scenario that is presented below is 360 seconds.

The initial topology of the APs and the associated terminals are depicted in Figure 20-a. 10 APs and 21 stations constitute the network graph. Each AP collaborates with the domain level manager by providing topological and monitoring data in order to solve the optimisation problem. The domain manager coordinates the management actions that require a greater view of the network area.

The stations transmit broadcast data traffic to the APs, which undertake to re-transmit it to all associated terminals. Hence, the uplink (UL) traffic of each AP is reaching the level of 50% of the downlink (DL) data traffic. The traffic is mainly generated by the stations that belong to AP 3, AP 7 and AP 5. AP 7 and AP 3 transmit 950Kbps and 50Kbps respectively, while the rest APs have zero or near to zero DL traffic.

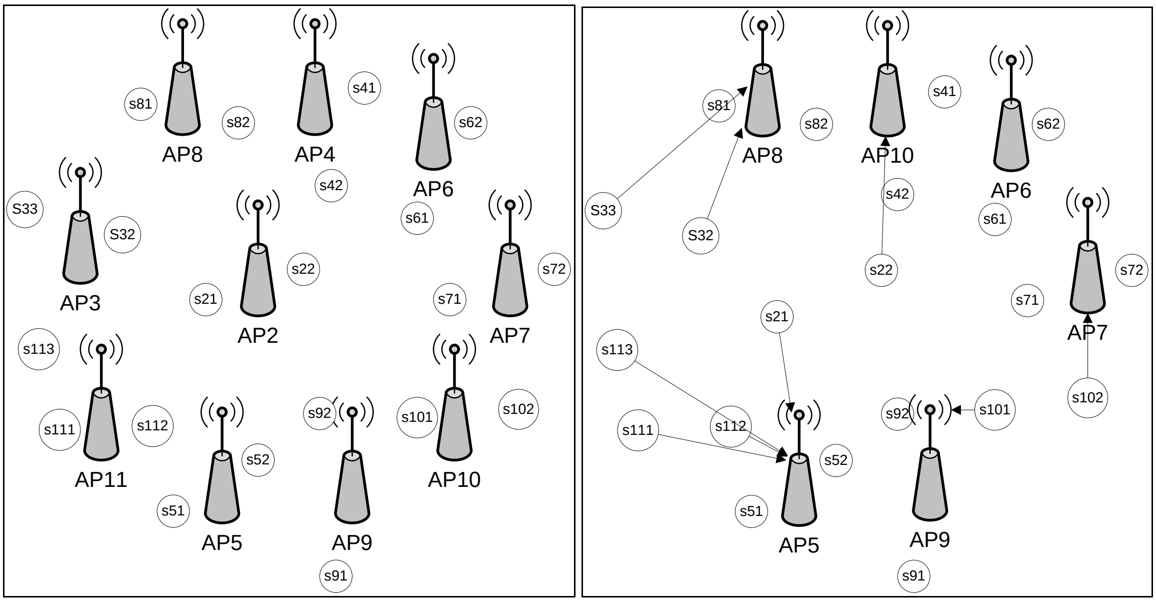


Figure 20: Simulation topology (a) Network graph before Optimisation (b) Network graph after Optimisation

The goal of the coverage management is to provide network connectivity at all desired locations, while capacity management undertakes to provide sufficient bandwidth to satisfy clients' communication needs. The Capacity Usage Ratio (*CUR*) of the network area, which consists of *n* APs indicates the percentage of the used capacity that *n* APs provide:

$$CUR = \sum_{i=1}^{n} \frac{AP_{Cap,i}}{AP_{CapAvail,i}} ,$$

where $\mathrm{AP}_{\mathrm{Cap,i}}$ describes the used capacity, both uplink and downlink, of AP_i, and $\mathrm{AP}_{\mathrm{CapAvail,i}}$ the maximum available capacity of AP_i.

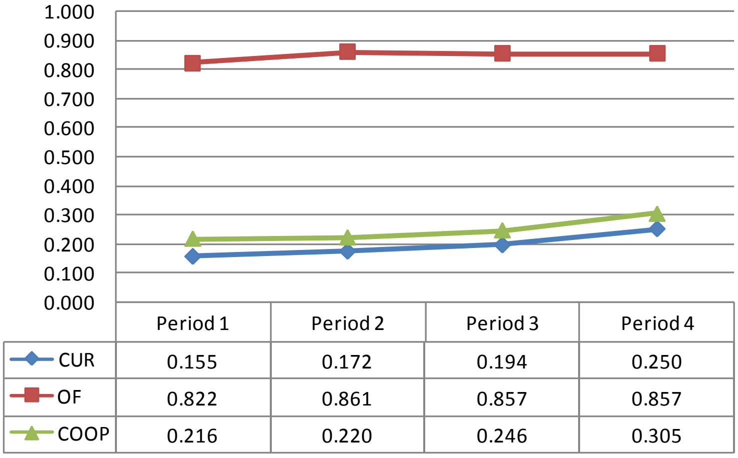
The number of the APs that are deployed and mainly the overlap of their transmission range should be taken into account. The network area for an AP includes the (one-hop) neighbouring access points and those (two-hop) access points that are not within each other's reception range, but are within the reception range of associated clients (i.e., meet the hidden terminal situation). For the calculation of the overlapping factor (OF) of a network area of access point we use the clustering coefficient, based on the graph theory [43]. We assume that *G = (V, E)* is a connected, and undirected graph of AP, with a set of nodes (vertices) *V* and a set of edges *E*. Let *|V|=: n*, *|E|=: e*. Parameter *e* corresponds to the number of the existing connections (i.e. overlaps) among the APs, while n is the number of APs that constitute the network graph. The OF is provided as follows:

$$OF = \frac{2*e}{n*(n-1)}$$

The association of the *CUR* and APs *OF* in the cluster area allows the more effective interpretation of the information that *CUR* provides by taking into account the overlap level of the offered bandwidth in the corresponding network area. For that reason we introduce Coverage Optimisation Opportunity Coefficient (*COOP*), which is given by:

$$COOP = CUR^{OF}$$

The *COOP* metric is useful for the identification of optimisation opportunities for a low load situation, where less capacity is needed as well as for a high load situation, where more capacity is required. A low *COOP* value means that too much capacity is provided in a very dense area, while a too high *COOP* value indicates an overloaded network area, where there is the need for more resources.



	Period 1	Period 2	Period 3	Period 4
CUR	0.155	0.172	0.194	0.250
OF	0.822	0.861	0.857	0.857
COOP	0.216	0.220	0.246	0.305

Figure 21: CUR, OF and COOP values

The domain manager of the network area of the cluster, collects periodically the status and performance information of APs, calculates *COOP* and based on its value decides the deactivation or reactivation of an AP. In the specific scenario the UL and DL traffic is not high. Thus a switch off of APs is decided. Figure 21 describes the variation of the *CUR*, *OF* and *COOP* values during simulation, for the four (periodic) decisions made by the domain manager. The decision making takes place every fifty five second (55”). AP11, AP2, AP3, and AP10 have been selected to be switched off and the respective associated terminals have been finally transferred to those APs that can sense, as it is depicted in Figure 20-b. For the reallocation of stations the scheme presented in [40] has been used. The effective use of resources is associated with the efficient energy consumption. Over utilisation or underutilisation of the resources is not efficient for energy saving.

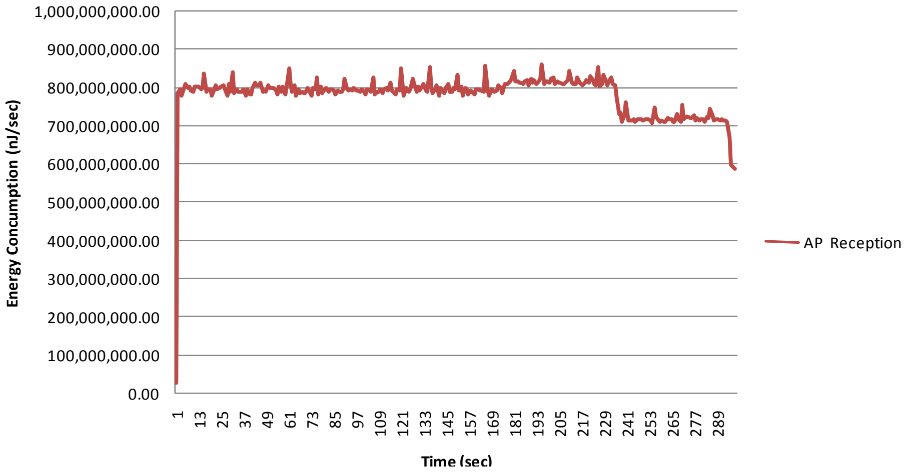


Figure 22: Total energy consumption of APs per second (Reception phase)

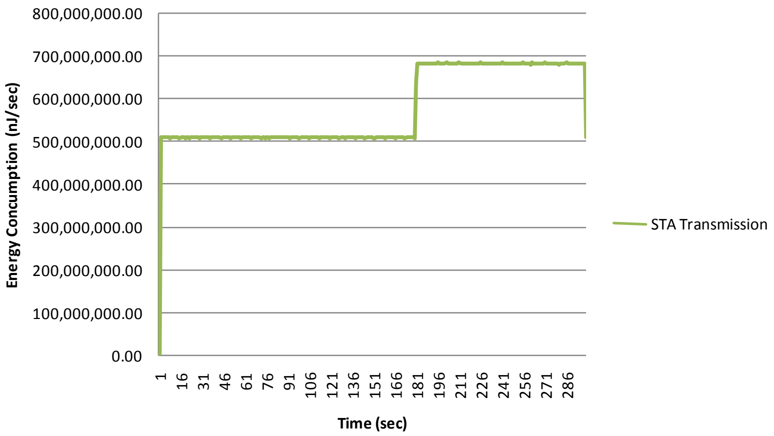


Figure 23: Total energy consumption of stations (STAs) per second (Transmission phase)

After the deactivation of four of the APs, the total energy consumption reduces mainly due to the decrease of the power used for the data reception, especially after the 4th deactivation (Figure 22). On the other hand we observe that there is an increase of the energy that is used for the data transmission from the stations (STAs). The packets that the terminals have sent remain the same after each deactivation action. The longer transmission distance after the handover justifies this increase. This cost is more obvious for stations that have high UL traffic (Figure 23). Hence, investigating the interaction between coverage and capacity optimisation (through dynamic AP switch On/Off) and energy saving we deduce that on the network side both goals are achieved (i.e. resources and power saving). However, on the station side the metric of energy increases, something that is not beneficial for the total network status, triggering further optimisation actions (e.g. RAT selection).

3.4.4 Autonomic Coordination among Inter-Cell Interference Coordination and Capacity and Coverage Optimisation in Downlink LTE Self-Organising Networks

This subsection describes the steps towards the policy-based coordination of two SON algorithms, namely Inter-cell Interference Coordination (ICIC) and Capacity and Coverage Optimisation (CCO), in the context of a downlink LTE SON network. First, a brief analysis on the possible interactions among the two mechanisms is given. Then, the two SON mechanisms are formulated through a mathematical framework, so that they have the same parameters and it is possible to form a common optimisation problem to represent their coordination. Finally, this subsection focuses on the derivation of a multi-objective problem, while other optimisation methods for SON coordination are also proposed and are actually intentions for future work.

3.4.4.1 SON Coordination

SON coordination is triggered by both governance (business objectives and high level policies) and/or context changes (e.g. traffic rise). Governance triggers are in principle proactive, while context triggers can be either proactive or reactive depending on the nature of the change (predicted or happening now).

Three kinds of interactions are derived in this analysis, each one representing a different interaction scenario. The first scenario takes place when the two SON algorithms affect the same set of parameters. For instance, a cell that wishes to increase its capacity applies CCO, deriving an appropriate resource allocation but increasing the interference of neighbour cells, which in turn send an update of RNTP (Relative Narrowband Transmit Power indicator) messages [45] that lead to a concurrent ICIC to the target cell. The common set of parameters consists of the resource allocation (allocation of Physical Resource Blocks (PRBs) with specific power). This scenario is described in detail in this subsection. The second scenario occurs when the two SON mechanisms affect the same metric(s). This scenario differs from the previous one, in the sense that the two SON algorithms may affect different parameters but the same metric, e.g. the cell throughput. As an example, CCO maximises the cell throughput via an appropriate resource allocation, while ICIC increases cell throughput by just optimising the reporting thresholds/periods of RNTP signalling and thus reducing inter-cell interference. Finally, the third scenario happens when one SON algorithm is just activated to compensate for a negative effect of the other one. For instance, ICIC is activated to reduce the increased interference caused by CCO, similarly to the first scenario but without affecting same parameters or metrics. The three scenarios are illustrated in the concept maps of Figure 24.

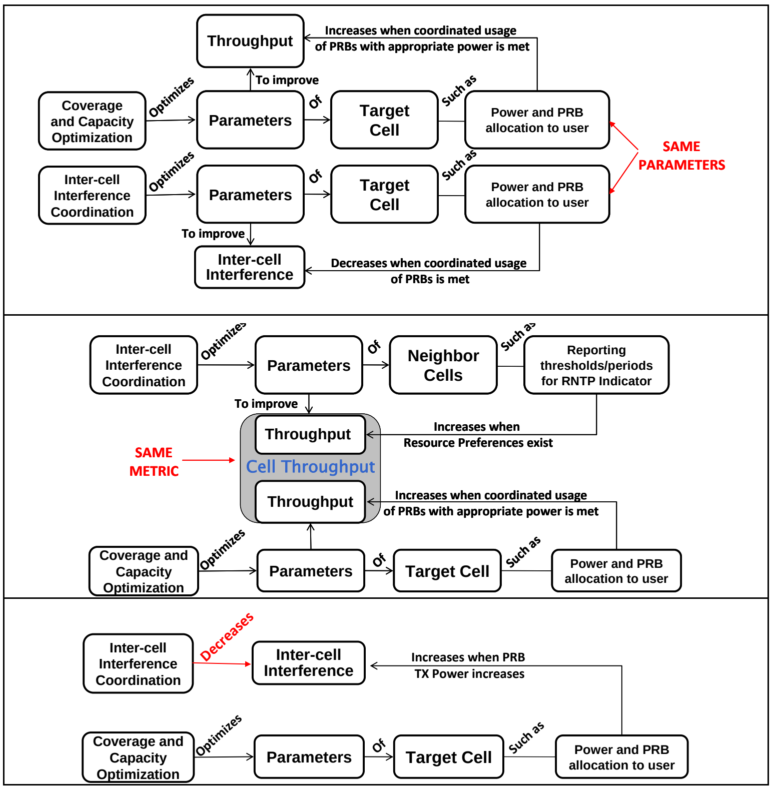


Figure 24: Possible interactions among ICIC and CCO

3.4.4.2 ICIC and CCO Problem Formulations

The more interesting scenario is undoubtedly the first one, since the common set of parameters allows the design of a common optimisation problem. Let us assume that N denotes the number of active users in the target cell and S is the number of total available radio resource elements, i.e. PRBs, in the system. Then, we select $n \in [1, N]$ to represent a user, $s \in [1, S]$ to represent a PRB and $c \in [1, C]$ to represent an interfering cell. Moreover, the SINR of user *n*, who is served from the cell *i* over the PRB *s*, can derive from the following equation:

$$SINR_{n,s} = \frac{g_{n,s,i} K_{n,s} P_{s,i}}{\sum_{c=1}^{C} g_{n,c,i} v_{s,c} P_{s,c} + N_0}$$

where $g_{n,s,i}$ is the channel gain between the user *n* and the base station *i* over the PRB *s*, $P_{s,i}$ is the total transition power level by which BS *i* can transmit PRB *s* and N_0 is the received thermal noise power. Moreover, $v_{s,i}$ is the transmit power coefficient of the PRB *s* from the BS *i*, selected from a discrete set of values $[0 : st : 1]$, where *st* is the step of discrete value selection. Finally, $K_{n,s}$ is the resource and power allocation matrix of the target cell *i*. More specifically, it is a $N \times S$ matrix which determines which PRB is going to be allocated to which user and the power level by which this PRB is going to be transmitted. The logical expression of $K_{n,s}$ matrix is the following:

$$K_{n,s} = \begin{cases} v_{s,i}, \; if \;\; PRB \; s \; is \; assigned \; to \; user \; n \\ 0, \; otherwise \end{cases}.$$

In order to calculate the throughput provided to a user by the system, the information of the combined SINR of the user (i.e. the SINR that derives from the combination of the SINR of the individual PRBs that have been allocated to the user) is required. This piece of information derives from the following equation.

$$SINR_n = \left[\left(\frac{1}{|S_n|} \sum_{x \in S_n} \frac{SINR_{n,x}}{1 + SINR_{n,x}} \right)^{-1} - 1 \right]^{-1},$$

where S_n is the set of PRBs that are assigned to user *n*, and $|S_n|$ the cardinality of the set.

The provided throughput to the user *n* is equal to $Thrpt_n = \Gamma\left(SINR_n\right)$, where Γ is a step function that can be obtained by link level simulations and describes the mapping of channel quality to the expected throughput [46].

The ICIC and CCO problem formulations are depicted in Table 1.

Table 1: ICIC and CCO problem formulations

	ICIC	CCO
Objective function	$\min \sum_{n=1}^{N} \sum_{s=1}^{S} \sum_{c=1}^{C} \boldsymbol{logical}(K_{n,s}) m_{s,c} I_{s,c,n}$	$max\left(CellThrpt\right) = max\left(\sum_{n=1}^{N} \Gamma\left(SINR_n\right) \right)$
Constraints	$\sum_{n=1}^{N} \sum_{s=1}^{S} \boldsymbol{logical}(K_{n,s}) \le S$ $\sum_{n=1}^{N} \boldsymbol{logical}(K_{n,s}) \le 1, \forall s \in [1, S]$ $Thrpt_n \ge r_n, \forall n \in [1, N]$	$\sum_{n=1}^{N} \sum_{s=1}^{S} \boldsymbol{logical}(K_{n,s}) \le S$ $\sum_{n=1}^{N} \boldsymbol{logical}(K_{n,s}) \le 1, \forall s \in [1, S]$ $Thrpt_n \ge r_n, \forall n \in [1, N]$ $SE_n = \frac{Thrpt_n}{S_n} \ge SE_{Thres} \; \forall n \in [1, N]$

The objective of the ICIC optimisation problem is to find the appropriate resource allocation in the target cell, in order to minimise the interference caused at the target cell's users. In order to retrieve only the information of resource (and not power) allocation from $K_{n,s}$, we use in the problem formulation the $\boldsymbol{logical}(K_{n,s})$, where $\boldsymbol{logical}(\;)$ is a function that converts numeric values to logical. The array under investigation in this problem is $K_{n,s}$. The first constraint satisfies that the resource allocation in the target cell will not exceed the available resources of the system, i.e. the number S of total available PRBs. The second constraint is used to represent that each PRB is allocated to only one user of the target cell, while the third constraint guarantees that each user will receive the appropriate resources, in order to satisfy his rate requirements.

The objective of the CCO optimisation problem, is to find the appropriate resource and power allocation in the target cell, in order to maximise the provided cell throughput $CellThrpt$ (capacity optimisation) while it takes into account that all users of the system experience acceptable or default channel quality (coverage optimisation). The variable under investigation is again the $K_{n,s}$ matrix, based on which the $SINR$ is calculated. The first three constraints are the same with those of the previous formulated ICIC problem. The forth constraint, though, satisfies that each user of the system is served with appropriate spectral efficiency (SE_n), which is above a certain threshold (SE_{Thres}), where S_n is the set of PRBs that are assigned to user *n* as above.

3.4.4.3 ICIC and CCO Coordination Optimisation Problem

The SON coordination may be investigated with various approaches, namely multi-objective optimisation, definition of a common objective function, game theory (cooperative and non cooperative games) problems.

The previous formulations allow us to consider the same parameter array, which eases the definition of the SON coordination problem via the previous optimisation tools. In this analysis, we focus on the first approach, that is to say, the multi-objective optimisation. The two SON algorithms are represented as a vector of objectives, e.g.

$$F(x) = [F_1(x), -F_2(x)],$$

where $F_1(x)$, $F_2(x)$ are the objective functions of ICIC and CCO in Table 1, respectively, and x the array $K_{n,s}$. Therefore, the array x represent the parameters, the objective functions F(x) represent the SON algorithms and the output values of $F(x)$ represent the metrics.

Then, the problem is formulated as minimisation (since maximisation is needed for CCO, then - $F_2(x)$ is used) of a vector of objectives $F(x)$ subject to the aggregation of ICIC and CCO constraints in Table 1.

Note that because $F(x)$ is a vector, if any of the components of $F(x)$ are competing, there is no unique solution to this problem. Instead, the concept of non-inferiority [49] (also called Pareto optimality [47] and [48]) must be used to characterise the objectives. A non-inferior solution is one in which an improvement in one objective requires a degradation of another. Since any point in the feasible region Ω that is an inferior point represents a point in which improvement can be attained in all the objectives, it is clear that such a point is of no value. Multi-objective optimisation is, therefore, concerned with the generation and selection of non-inferior solution points. Non-inferior solutions are also called *Pareto optima*. A general goal in multi-objective optimisation is constructing the Pareto optima.

3.5 Results

The benign interaction of SON functionalities is of primary importance for operators as otherwise they cannot trust the stability of the self-managed network. In this section we provided an overall analysis of potentially conflicting interactions for the 3GPP standard. We propose that a separation in time is a suitable cooperation strategy to avoid conflicts among functionalities. The overall view was complemented by analyses of individual interactions that have not been touched by state-of-the-art yet, e.g. the HetNet analysis. The integration of the solutions in the UMF requires the translation of high-level policies set by the operator to more specific technology-dependent policies for the given functionalities.

The work described in this section does not focus so much on specific methods, but rather on the overall picture of the integration of various approaches and methods and how this can be achieved in a benign and harmonised way. The primary requirement of operators towards a self-managed network with a number of different functionalities is certainly the stable interaction of these functionalities and the avoidance of oscillations of parameters that are controlled by multiple functionalities. All these were addressed in this section.

3.6 Discussion and Future Work

This section goes beyond state-of-the-art in the following sense: First, we provide a comprehensive visualisation of interaction issues appearing in LTE systems. Secondly, we analyse an interaction issue occurring in an LTE-Advanced HetNet system; the underlying dilemma of how to distribute power and resources between macro and pico base stations is vital for a successful deployment of networks with nodes with heterogeneous form factors. More specifically, it could be shown that the amount of resources on the backhaul link (i.e. between pico base station and macro base station, which would serve as the link between pico base station and core network) can be successfully equalised to the access links (i.e. between pico base station and terminal). Furthermore, we were able to show that energy savings can successfully obtained in the absence of a harmful impact on coverage/capacity optimisation in WiFi systems. Last but not least a mathematical framework for the interaction between coverage/capacity optimisation and interference coordination was derived for LTE systems.

Future work will further elaborate on SON coordination in LTE and LTE-Advanced networks. In this context stochastic approximation algorithms seem to be a promising direction. As a bracket around the individual coordination contributions, governance and policies, together with appropriate weights for individual components of utility functions, will be considered as well as the specific embodiment with respect to the UMF.

4 Cooperation Strategies for Network Stability

4.1 Introduction

Dimensions, dynamicity and complexity of today's networks are growing continuously. Understanding and controlling/managing the network behaviour to meet technical and business objectives is becoming increasingly more complicated and challenging. This is likely to exacerbate in the future, when it is expected that the networks will become highly dynamic and pervasive, capable of interconnecting large numbers of interconnected real and virtual resources (e.g., routers, switches, transport nodes, servers, etc), users' devices (e.g., smart phones, etc) and machines (e.g. sensors, smart things, etc).

Introduction of self-* features and control loops to tame expected level of dynamicity and complexity will imply the dependence of the network's global characteristics (e.g., connectivity and average delay rate) on some local parameters (e.g., congestion control) and as such the risk of instabilities. Network stabilisation can be defined as an adaptive feature of network which, given certain objectives and constraints, should be able to converge, within given time requirements, to stable desired state(s), characterised by target performance levels.

Overall, instability in communication networks may have primary effects both jeopardising the network performance and compromising an optimised use of resources, so network self-stabilisation is an important feature. As an example, instability of end-to-end communication paths may be dependent both on the underlying transport network, as well as the higher level components specific to flow control and dynamic routing. Also the arguments for introducing advanced flow admission control are essentially derived from the observation that the network otherwise behaves in an inefficient and potentially unstable manner. Even with resources over-provisioning, a network without an efficient flow admission control mechanism may have instability regions that can even lead to congestion collapse in certain configurations. Another example is the instability which is characteristic of any dynamically adaptive routing system. Routing instability, which can be (informally) defined as the quick change of network reachability and topology information, has a number of possible origins, including problems with connections, router failures, high levels of congestion, software configuration errors, transient physical and data link problems, and software bugs.

This section presents the work undertaken by Task Force (TF) 3.3F. The main causes of configuration instabilities are identified and the basic principles of the proposed solutions for handling such situations are outlined. Issues related to context dissemination and to reliable routing of traffic in the presence of multiple optimisation objectives are investigated. A study on the stability of dynamic games of control loops is also provided.

4.2 Related Work

Given that networks can experience unexpected traffic demand as well as link failures, appropriate mechanisms should be in place to react to such events so that network instability is avoided and traffic can be reliably shipped from source to destination. Absence of such mechanisms can lead to high packet loss rates and subsequently quality degradation of supported services. Furthermore, the algorithms employed by these mechanisms should converge to a safe network configuration avoiding oscillations.

The main body of work in the area of network stability concerns routing protocols. The work in [62][63] [64] and [65] investigates issues related to the stability of BGP. In [62] and [63] the authors consider the fundamental problem of verifying the convergence of BGP configurations and address this issue by devising restrictions on BGP policies. The solution proposed by [63] however imposes restrictions on both the autonomy of ASes (Autonomous Systems) and the expressiveness of policies, whereas [62] preserves the autonomy of ASes. Determining potential routing oscillations in BGP configurations is also the subject of the work in [64]. The proposed algebraic methodology does not have significant restrictions on configuration flexibility and is able to provide the reasons behind detected oscillations. As suggested by the work in [65], the BGP convergence problem can be addressed either dynamically or statically. The dynamic solution is a mechanism suppress or prevent BGP oscillations due to policy conflicts at run time, whereas a static solution relies on programs to analyse routing policies, before deployment, for conflicts that could lead to protocol divergence. It is argued that if additional information is carried in BGP messages the dynamic approach can overcome persistent route flapping issues and be more practical than the static approach which is rather complex. Routing instabilities have also been studied in the context of wireless mesh networks, which occur as a result of the quality of wireless links due to external interference. The authors of [66] identify the shortcomings of some routing protocols that do not evaluate route quality frequently at run time and subsequently fail to adapt to route quality variations. They present a measurement-based characterisation of routing stability and identify that simple stabilisation techniques, such as hysteresis thresholds, can significantly reduce unwanted route flapping.

Network configuration instabilities that can arise as result of policy conflicts have been investigated in [67]. Various configuration inconsistencies defined by policies implementing control loops in the domain of Quality of Service (QoS) have been identified and classified into static and dynamic conflicts. Special rules are used for the detection and resolution of inconsistencies in policy specifications by employing formal reasoning techniques. QoS policy conflicts have also been addressed in [70] with policies used to define the treatment of a traffic flows. Network stability when enforcing policies is also addressed in [71], where policies and their constraints are modelled using finite state transducers. A process has been devised to derive predictions of policy enforcement consequences, which can detect configuration flip-flops that can cause instabilities. Further description of previous work related to policy conflict analysis can be found in Section 9.3.1.

4.3 Relation with UMF

Figure 25 provides an overall view of the mapping of TF3.3F to UMF functional blocks (FBs). The key issues tackled by this task force and associated solutions correspond to the functionality provided by the Cooperation FB (CO_FB) given that orchestration of multiple control loops is essential when addressing potential configuration instabilities. For effective decision making and resolution of possible instabilities orchestration mechanisms take input from the Information and Knowledge Building FB (IKB_FB) and also from the Profiles and Models FB (PM_FB) regarding the network state (e.g. congested links) and the network model (e.g. routers, links, paths), respectively. The Governance FB (GOV_FB) provides the high-level goals to guide the process when optimising multiple objectives. The results of the formal verification process described in Section 4.4.2 are used to populate the Profiles and Models FB (PM_FB) in the form of static knowledge for network planning purposes. The IKB_FB receives input from the Monitoring FB (MON_FB) regarding the run-time state of the network.

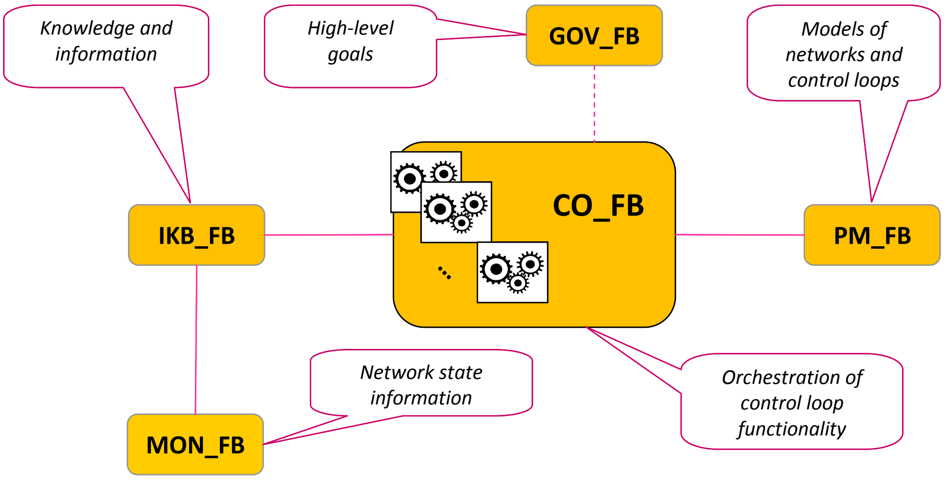


Figure 25: Relation with UMF functional blocks

4.4 Methods, Tools, and Models

4.4.1 Control Loop Stability through Orchestration

The complexity in managing current and emerging networks has led researchers into investigating novel paradigms for automating many of the management tasks. However, the introduction of self-management functionality to achieve this automation, in the form of control loops (CLs), can have destabilising effects in the network operation given the distributed nature of control and also possible interactions between multiple control loops with different objectives. This section identifies various causes of configuration instabilities in fixed networks and provides an overview of the proposed approach for achieving stability while optimising multiple objectives.

4.4.1.1 Configuration Inconsistencies

Although closed loop control can realise self-management functionality and can thus automate a large part of the network management process it can also lead to configuration instabilities, which can have catastrophic effects on the network performance and the quality of associated services. This is mainly attributed to the distributed nature of control and also to potential interactions between multiple control loops that realise different management functionality. Below, we identify and provide examples of such inconsistencies when developing closed loop management solutions.

Given the disadvantages of centralised management – e.g. single point of failure, scalability issues, and lag in the central manager reactions – distributed approaches have gained increasing interest. Based on this paradigm, control over network (re-)configuration actions is granted to more than one network entity. An example of this scheme, applied to online resource management, is demonstrated in Figure 26 where multiple network ingress nodes (*I1-I3*) are able to change the splitting ratios of incoming traffic flows (defined as traffic between source-destination pairs) over multiple paths towards a destination so that various objectives are achieved, such as load balancing and energy conservation. Let us first consider a single type of control loop – one that manages traffic distribution for the purpose of balancing the network load. Suppose that link *C1-E1* is highly utilised and a re-configuration is required to shift some traffic from this link to a less loaded one. Since traffic emanating from all ingress nodes contribute to the load on *C1-E1*, i.e. flows *f1-f3*, and all nodes can reach destination *E1* through alternative paths, adjusting the splitting ratio at any of the three ingresses can potentially solve the problem: (a) *I1* sends more traffic over link *I1-E1*, (b) *I2* sends more traffic over links *I2-C2-E1*, and (c) *I3* sends more traffic over links *I3-C2-E1*. If, however, ingress nodes act independently their configurations can be inconsistent relative to one another since traffic can accumulate in other parts of the network, overloading for example link *C2-E1*, while *C1-E1* becomes underutilised. Thus, configuration inconsistencies that arise due to the distributed nature of decision making entities (DMEs) optimising a specific objective should be handled.

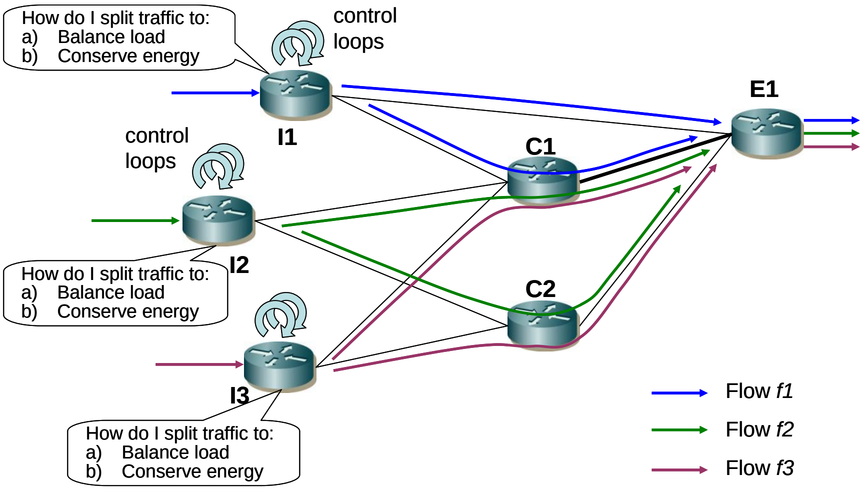


Figure 26: Example re-configuration decisions in distributed settings

To automate the majority of network management tasks multiple control loops are required, each realising different management functionality. A simple example is depicted in Figure 26 where, in addition to the load balancing (LB) control loop, an energy efficiency (EngE) loop aims to control the traffic rates on links so that energy consumption is minimised. This is based on the fact that sending data at low rates on router interfaces requires less energy compared to high rates, where switching off some interfaces altogether being also a possibility. It is evident that these two control loops have contradictive objectives: the LB loop aims to optimise resource utilisation by evenly spreading traffic to as many network links as possible, whereas the EngE loop aims to optimise energy consumption by routing traffic using the fewest possible links and/or by minimising the sending rates on certain links. As such, if DMEs treat these optimisation objectives separately the resulting configuration actions can be inconsistent with each other. For example, in Figure 26, a decision by the EngE loop of node *I3* to stop sending traffic over path *I3-C2-E1*, so that link *I3-C2* can be switched off (all traffic thus routed via path *I3-C1-E1*) can result in violating the objective of the LB loop if link *C1-E1* becomes over utilised; in such case the LB loop reacts by shifting traffic load back to path *I3-C2-E1*. Independent optimisation of management objectives can thus lead to configuration instability through oscillations whereby two competing CLs perform inconsistent actions in response to each other's decisions.

4.4.1.2 Control Loop Orchestration and Resolution of Inconsistencies

Based on the identified causes of inconsistencies that can lead to configuration instabilities we describe here an approach by which these can be handled. The approach is based on the principles introduced in Section 2.4.3 regarding the formation of an in-network overlay (INO) among DMEs to orchestrate the re-configuration process. This is depicted in Figure 27 where network ingress nodes (*I1-I3*), responsible for configuring traffic splitting ratios, communicate to decide on the most appropriate configuration taking into account: (a) the distributed nature of control, and (b) the requirement of optimising several objectives concurrently.

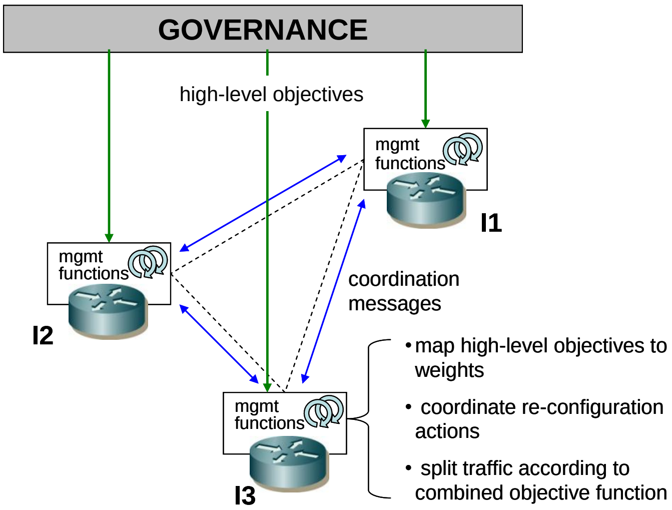


Figure 27: Orchestrating distributed re-configuration through coordination

In the proposed scheme re-configurations are performed in an iterative manner with only one network node being able to adapt the splitting ratios at a time. This prevents concurrent adaptation actions across multiple nodes that could lead to inconsistencies, with the re-configuration responsibility essentially being 'locked' to a specific DME at each iteration. Selecting the most appropriate node to carry out a re-configuration is one of the main challenges. As explained in Section 2.4.3.3 we have effectively used selection rules when optimising the single objective of load balancing, which consider the contribution of each ingress node to the load of the most utilised link in the network. Similar rules could be used concerning other objectives, such as energy efficiency, but if evaluated separately different nodes could be selected for optimising different objectives, which can lead to instabilities. As such, we propose a scheme by which, at the start of a re-configuration cycle, each DME computes the *combined gain* (numerical value that quantifies the effectiveness of a reconfiguration action) associated with its local solution, taking into account all objectives, and communicates this information to other DMEs through coordination massages. The node with the highest gain is subsequently selected to perform the re-configuration – in the case of multiple nodes with the same gain one can be randomly selected.

To avoid instabilities introduced by configuration oscillations the objectives of all control loops should be jointly considered by each DME as a single optimisation function. As such, the orchestration of multiple control loops can be formulated as a multi-objective optimisation problem:

$$U_{DME}(x) = \min \sum_{n=1}^{N} w_n U_n(x) \quad \text{, where} \quad \sum_{n=1}^{N} w_n = 1$$

Based on the above, each DME can compute a local traffic splitting ratio *x*, by minimising the sum of the weighted utility functions U_n corresponding to individual objectives. The weights designate the relative importance of each objective and are derived, i.e. mapped, from the high-level goals defined by Governance. For example, maximising the *profit* would increase the weight associated with the EngE utility function so that energy consumption is minimised, whereas maximising *service satisfaction* would increase the weight of the LB utility function so that packet loss is kept low [58]. Given that re-configuration is an iterative process, individual DMEs update and exchange the gain associated with their local solution at each iteration. This value represents the effectiveness of a re-configuration taking into account the weights associated with individual objectives. For example, an adaptation by a DME *I3* which can offload 30% of the traffic on the most utilised link and at the same time reduce energy consumption by 10%, would be preferred to the solution calculated by DME *I2* which can achieve 30% and 8% reduction in utilisation and energy consumption, respectively, if there is more bias on the energy efficiency. The re-configuration process terminates when the high-level goals are achieved or if it reaches the maximum number of permitted iterations.

4.4.2 Cooperation of Nodes for the Needs of Context Dissemination

One of the most fundamental and "low-level" forms of cooperation in any kind of computer networks is the cooperation between routers (or nodes) for routing purposes; that is for finding the most appropriate path according to some predefined metrics from a given source to a given destination. To do so each router runs an instance of a routing process relying, in many cases, only on partial network information provided to it by its neighbouring nodes rather than network-wide information.

This "in parallel" operation which relies only on partial information can lead to instabilities and problematic situations. Problematic situations in general can be either deadlocks or livelocks. Deadlock is a condition where a process stalls; meaning it reaches a state from which there is no exit action. When it comes to routing this would mean the condition where a packet reaches a router/node and is not forwarded any further because the routing process has reached a state which was not taken into account in its behavioural specification. As a result, there is no appropriate exit action to be taken since the routing process was never expected to enter this state and therefore no action to be taken was defined. Livelock is a condition from where a process can exit; however every exit action will eventually lead the process back to the same condition. With respect to routing this would refer to the existence of loops. That is why there exists an exit action for every state of the routing process at every router; however, one packet after leaving one router will always keep being directed back to this same router and will never reach its destination.

In this section we will show how formal verification (model checking) can be applied in this context, to find problems in routing that may result as a consequence of "in parallel" operations that are performed based on partial rather than global network information. As an example case study we will use a routing protocol designed for wireless sensor networks (WSNs) named Adaptive Load Balanced Algorithm, Rainbow version (ALBA-R) [59]. In the context of UniverSelf such a protocol could be of interest with respect to the dissemination of context between context providers (CPs) and context clients (CCs) of the context management infrastructure (CMI) [60]. In this specific case the CPs would be individual WSN nodes sensing some information from the environment and wrapping it into packets, whereas the CCs would be the sinks where the individual sensor readings should be delivered and processed e.g. for feeding an inference reasoner.

With respect to the UMF itself, the outcomes of this formal verification process can be used to populate the respective Profiles and Models Functional Block which include static knowledge in order to create a sort of look-up scheme to be used during the network planning process. That is, when a network operator would need to employ this protocol for communication between CPs and CCs, by consulting this look-up scheme they would know beforehand whether it would work properly for their topology and also obtain performance related metrics, related for example with the time needed for every packet to reach the sink.

Before proceeding with the description of the protocol itself and its modelling we will very briefly introduce the underlying concepts and ideas of formal verification.

4.4.2.1 Model Checking Concepts

Formal verification provides a systematic way to assess the correctness of protocols, processes and systems. The main difference compared to simulation is that instead of only examining a limited area of the operational space of the system under consideration, it can be used to examine the whole state space of possible operations and conditions under which the system may operate. This means that all possible combinations of inputs and actions can be taken into account and, therefore, all possible outputs can be derived and evaluated. One could regard the outcome of formal verification as the outcome of an infinitely large number of simulation runs. This means that contrary to simulations, formal verification methods are capable of capturing conditions and operations that may otherwise remain unnoticed, even after a very large number of simulation runs.

In general there are two approaches in formal verification [61]. The first one is to try to prove the correctness of a system and derive its properties through a sequence of theorems; this is called theorem proving. This process however is very cumbersome in practice. The second approach is called model checking. In this approach the behaviour of the system under consideration is modelled using the description language of the model checker and then the model checker examines all possible system evolutions based on the model. The main limitation of model checking is the state explosion problem; that is as the size of the system and the parameters under consideration increase so do the number of states and transitions between states.

In model checking, four types of models are commonly used, depending on the characteristics of the system to be modelled and analysed; these are Discrete-time Markov Chains (DTMCs), Markov Decision Processes (MDPs), Continuous-time Markov Chains (CTMCs), and Continuous-time Markov Decision Processes (CTMDPs). In the first two, all transitions take place in discrete (time) steps whereas in the latter two, time is modelled in a continuous manner. It is worth noting though that “dense” time representation is possible also through the former two models.

4.4.2.2 ALBA-R Specification

In the ALBA-R protocol, every node that needs to transmit a packet (either a packet itself generated or a packet from another node for which it is acting as a relay) it can do so either towards the nodes closer to the sink than it or to nodes further to the sink than it. Every time such a node needs to transmit a packet it broadcasts a control packet that indicates its “colour” and its distance from the sink (Geographic Priority Index - GPI) and asks from eligible relays to respond with their GPI and their congestion status (Queue Priority Index - QPI). It is worth mentioning that for energy saving reasons nodes follow an ON-OFF duty cycle (set to 10% in [59]) which means that it is not always the case that eligible relays within the transmission range of another node will respond. To account for this every node can make a number of broadcast reattempts up to N whenever it is in a state where it looks for eligible relays.

The selection of the direction for transmission depends on the “colour” C of the node. Initially all of the nodes of the topology are considered to be yellow, C_0, and they try to route packets towards the sink; that is to nodes with lower GPI than them (positive advancement). Eligible relays for yellow nodes are only other yellow nodes. In case a node realises that it cannot send to a yellow node, even after all reattempts are over, it back offs, changes colour to C_1, which means red and tries to find eligible relays in the reverse direction of the sink (negative advancement). Eligible relays in the case where the node has become red can only be red or yellow nodes. Whenever yellow nodes are reached, then normal operations are performed until the packet is finally delivered to the sink through positive advancement.

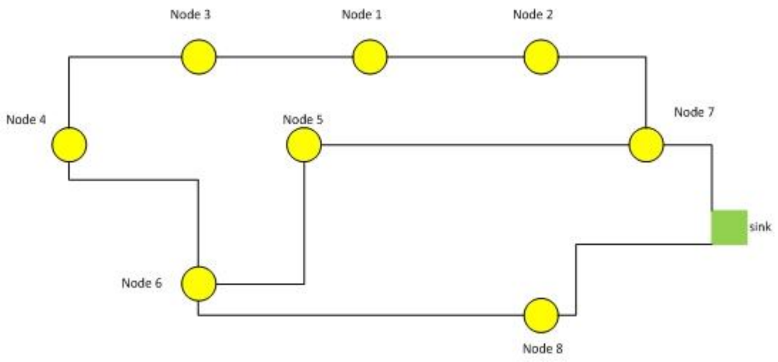
Not always, however, yellow or red nodes can be found so the node is forced again after the number of reattempts to change its colour to C_2, which means blue. According to its new colour the node tries to find a best relay offering positive advancement and nodes which have blue or red colour are candidate relays towards the sink. So, any packet generated by a blue node will travel through blue nodes towards the sink until it reaches a red node. Then red nodes will advance the packet in the reverse direction to red nodes until they reach a yellow node and then normal operations will be performed until the packet is delivered to the sink.

This process can be generalised to any number of nodes the main principles being that: a) when a node has an even number colour it can only look for relays offering positive advancement whereas if it has an odd number colour it can only look for relays offering negative advancement, b) eligible relays for a node with colour C_N are only nodes with colour C_N or C_{N-1} colour with nodes with colour C_{N-1} having priority over nodes with colour C_N and, d) among eligible nodes based on colour criteria the nodes with lower QPI or with lower GPI -in case of nodes with the same QPI- have priority to be selected as relays.

4.4.2.3 ALBA-R Modelling and Verification

For our case we assumed the minimum time unit is 10msec, which corresponds to the time needed for a control packet to be broadcasted and a reply for it to be received and all other time intervals were expressed as multiples of this basic time unit. More specifically the ON period of a node was set to 3 time units and the OFF period of a node was set to 27 time units so that we had a 10% duty cycle. We considered a sparse topology of 8 nodes and one sink with GPIs as shown in Figure 28 and that node #1 wants to transmit a packet to the sink. Since only one packet needs to be transmitted, the QPI functionality did not have to be included in the model.

For simplicity reasons we assumed that 6 nodes can start from any time instance within their OFF period whereas the remaining 2 can start from the beginning of their OFF periods but their transitions afterwards, corresponding to packet transitions and time progression within the duty cycles, are synchronised over the 10msec basic unit; this allowed us to build the model as a DTMC. It is worth noting that these assumptions with respect to the initial conditions of the nodes led to 387,420,489 initial states. This means that the results of model checking would correspond to this number of distinct simulation runs.



Node ID	***GPI value***
1	30
2	20
3	40
4	60
5	20
6	30
7	10
8	10

Figure 28: Considered topology and GPI values

As the results of model checking showed, in order to guarantee 100% delivery from node #1 to the sink, the number of transmission reattempts *N* should be equal or greater than 30. For number of reattempts lower than 30 there always exists the probability of deadlocks; this probability is 50% for *N*=10 and 25% for *N*=15.

It was also possible to see how the time needed for a packet delivery varies with the number of reattempts. Figure 29 shows the probability of delivery to the sink versus time as a function of *N*. As one can see for *N*=15 (blue line) the packet is delivered within 400msec with a probability of 50% whereas for *N*=4 (yellow line) the packet is delivered within 400msec with a probability of 43%. Similar statistics can also be derived for a number of properties, e.g. with what probability a packet stalls at node #3 or follows a specific route.

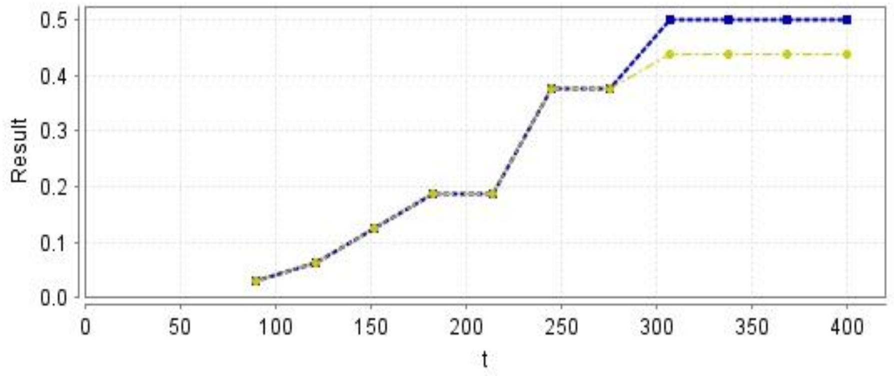


Figure 29: Probability of packet delivery to sink versus time as a function of *N*

4.4.3 Stability of Dynamic Games of Control Loops

Loosely speaking a network is said to be stable near a given state if one can construct a Lyapunov function (scalar function) that identifies the regions of the state space over which such functions decrease along some smooth trajectories near the solution. For example in mechanical systems a Lyapunov function is considered as an energy minimisation term, or overall it can be considered as a cost-minimisation term or an error-minimisation term. On the other hand, it should be noted that an inadequate Lyapunov function may cause the excess of false-positive warnings, a risk that cannot be avoided. Moreover even if we obtain an asymptotic stability, it is not clear what may happen during transients (typical of control feedbacks).

The introduction of self-* features and automatic control loops to tame expected level of dynamicity and complexity of future networks, will imply a reverse side of the coin which is the dependence of some network's global characteristics on certain local parameters and as such the risk of instabilities. This is basically due to the nonlinear interactions unavoidably introduced as a consequence of exploitation of self-* features and automatic control loops (interaction of multiple control loops can bring chaotic behaviours).

As such, in future networks we will witness (at least at the edges, where the density of node will be very high) a sort of dynamic game of controllers. In the usual formulation of game theory there is an equilibrium state that can arise: loosely speaking this equilibrium state is in some sense analogous to thermal equilibrium and reflects the static nature of the game itself.

If the game is allowed instead to be dynamic, with the rules able to change due to the states of the controllers, then there could also be dynamic equilibrium, analogous to a non-equilibrium steady state. Such a game could be described again by using dynamical systems theory. Under learning, chaotic dynamics can arise, and the game may fail to converge to Nash equilibrium. Understanding these dynamics is essential.

At this level of complexity it might be advisable moving the stability analysis in the network phase-space. A phase space can represent the network behaviour in terms of trajectories changing over time. For example the structural stability of a complex dynamic network involves an analysis on how domains of attractions, particularly at the boundaries, are modified by alteration in the value of the network parameters.

Indeed, an interesting perspective is understanding the relationship between global network states vs. local networks states and how they influence each another. Actually each (global or local) network state is characterised by its associated data. As a consequence, the adoption of data mining and knowledge extraction techniques will be valuable approach for the design, management and control of these network states.

Given the number of nodes expected in future networks, a phase space analysis might appear quite challenging. Some simplification should be adopted.

A proposal is adopting that the basic units are not the control-loops, but small “attractor (sub-) networks” of control-loops i.e. non-linear networks (modules) whose behaviour is dominated by their attractor states that may be built in or acquired through learning. Adopting this metaphor, this is like saying that interactions can be approximated as interactions between attractor states (i.e. connection strengths are replaced by state interaction matrices).

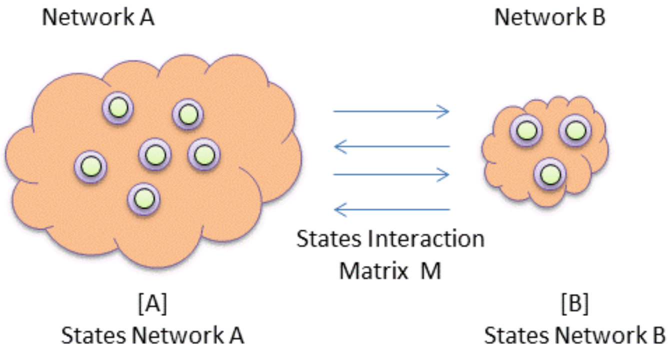


Figure 30: States interaction matrix

Ensuring network stability implies validating the existence (in the said phase space) of stable attractor states (with the related levels of performance) and bringing the overall network dynamics to target states.

A given strategy of interactions (for instance of sub-networks) has a functional dependence on certain attractors of the game dynamics: norms for cooperation and competition can be defined as strategies that can appear as specific attractors of the network game dynamics.

4.5 Results

The work presented in this section describes the first steps towards the development of solutions for handling the critical issue of network instabilities. Driven by the problems defined in the project Use Cases, our analysis revealed possible causes of these instabilities, which are mainly attributed to inconsistent routing/resource management decisions that can be taken in a distributed fashion. Our initial study suggests that a cooperative approach is required by which decision making entities orchestrate the re-configuration processes through the exchange of control messages to coordinate their actions. The use of a model checking technique has also been investigated for the purpose of verifying the stability of a routing configuration, the effectiveness of which has been demonstrated through a practical example.

The analysis and the mapping of the self-management functionalities to be developed to UMF functional blocks will enable the derivation of the requirements for their effective embodiment into network functions.

4.6 Discussion and Future Work

Network stability is of prime importance to network operators given that their main objective is to transport traffic in a reliable manner with sustainable levels of quality. This section identifies possible causes that can compromise the integrity of networks and proposes ways in which this issue can be potentially addressed. Given the distributed nature of most routing algorithms and control loops in general we believe a cooperative approach is necessary, which will provide the means to harmonise the decisions of various entities in the network. The model checking approach presented can be generalised to consider the interactions between any processes rather than routing processes alone. Using a similar modelling approach one can check the joint operation of a number of interacting processes and deduce with a very high degree of confidence about the correctness of this joint operation and also reason about sequence of operations that may lead to problematic situations. This way when networking scenarios that rely on these interacting processes arise, a network operator can be certain about what to expect and know in advance whether problematic situations can or cannot be ruled out.

In terms of orchestration of control loops future work will involve the formulation of several objectives into utility functions based on the model of network entities. A method for calculating local reconfiguration gain as well as algorithms for the intelligent and dynamic selection of re-configuration entities will be developed. Mapping of high-level Governance goals and their effect on optimisation functions will also be investigated and applied in practical use case problems.

In a network (of a certain level of complexity) the dynamic games of multiple control loops may result in chaotic behaviours. The challenge is modelling the related dynamics and mastering this complexity handling a limited number of parameters. Future work will involve modelling the behaviour of a (set of) control loop(s) and their interactions as nonlinear dynamic systems. As in many other nonlinear systems that change with time, this behaviour is dominated by a relatively small number of "attractors", which correspond to activity patterns (i.e. eventually sets of data). The degree of influence that the state of one set of control loops would have on the state of other ones would be represented by a "multi-dimensional matrix" coupling attractor states of networks. This toy model of an ensemble of interacting control loops will be developed and will be validated through simulations.

5 Conclusion

This document describes the work carried out in the Task 3.4 of Work Package (WP) 3 on “Network Empowerment”. The presented work addresses the key issue of cooperation among decision making entities so that high-level management objectives can be achieved without compromising the desired overall network behaviour and stability. A set of approaches are proposed that tackle this research problem in the context of different networking technologies and aim at providing the means by which decentralised optimisation mechanisms can safely operate together to generate reliable network configurations.

While previous WP3 deliverables D3.1, D3.2/D3.3 focused on optimisation issues, network behaviour and knowledge building, the cooperation approaches proposed in this deliverable extend the efforts of WP3 towards empowering the network with additional self-management capabilities. The technical challenges addressed here are based on practical problems identified in the project’s use cases. Furthermore, the functionality of the various cooperation mechanisms proposed has been mapped to relevant UMF components. Results in four technical areas were elaborated and analysed in this report.

Section 2 investigates solutions for dynamic inter-cell interference coordination and collaborative load balancing in cellular networks, cooperative traffic engineering between P2P and core network segments, and dynamic resource management in fixed networks. Initial implementations of these solutions have been evaluated through experimentation, and key performance factors as well as benefits have been identified. Future work will include further tuning of the approaches under various conditions to avoid possible oscillations and to minimise convergence times, and the development of mechanisms for the effective orchestration of resource management functionality.

Section 3 presents the work on the interactions of self-management functionality in self-organising networks (SON). Different types of interactions in wireless environments (LTE, WiFi) have been identified and a methodology has been proposed to handle different management objectives. A mathematical framework for the interaction between coverage/capacity optimisation and interference coordination has been specified. Based on this, we were able to show that energy savings can successfully obtained in the absence of a harmful impact on wireless environments. Future work will involve the testing of different stochastic approximation algorithms for the control parameters, and the convergence of the SON coordination algorithm. Governance and policies, together with appropriate weights for individual components of utility functions, will also be considered.

Section 4 is the second of the core technical sections and discusses the critical issue of IP network stability. Various causes that can lead to compromising the integrity of networks are identified and a potential solution based on the concept of orchestration has been proposed. This involves the coordination of multiple decisions making entities in the network and the harmonisation of re-configuration decisions to avoid inconsistencies and destabilising management actions. Extensions of this work will address issues related to the formulation of multiple objectives into utility functions, the dynamic selection of re-configuration entities, and experimental evaluations.

Cooperation strategies as presented in this deliverable represent the artefacts and design for the UMF functional blocks in the emerging UMF specification. Further refinements to the proposed solutions will be performed during the second part of the project’s lifecycle. These refinements will also ensure the tighter integration of partner’s work within Task T3.4, tighter integration with the other WP3 tasks, as well as the streamlining with the forthcoming UMF releases and use cases’ specifications progressing towards a more unified vision of self-management cooperation strategies. Future work will also assess the suitability of the various methods used to solve the identified problems as these represent the current working assumption.

6 References

[1]. M. Salem, Adbulkareem, H. Yanikameroglu and D. Falconer, "Opportunities and challenges in OFDMA-based cellular relay networks: A radio resource management perspective," *IEEE Trans. on Vehicular Technology*, vol. 59, no. 5, pp. 2496-2510, June 2010.

[2]. J. Liu, D. Wang, J. Pang, and G. Shen, "Inter-cell interference coordination based on soft frequency reuse for relay enhanced cellular network," in *PIMRC*, Istanbul, Sep 2010, pp. 2304-2308.

[3]. V. Ayyadurai, K. Moessner and R. Tafazolli, "Multihop Celular Network optimization using genetic algorithms," in *CNSM*, France, 2011, p. accepted for publication.

[4]. R. Alimi et al, "ALTO Protocol", IETF draft, draft-ietf-alto-protocol-05.txt, July 2010.

[5]. UniverSelf D2.1 deliverable "UMF Specifications Release 1".

[6]. C. Wang, N. Wang, M. Howarth and G. Pavlou, "An Empirical Study of the Interactions between ALTO-assisted P2P Overlays and ISP Networks", IEEE LCN 2011, Workshop on Wireless and Internet Services.

[7]. The GEANT Network Topology, http://www.geant.net

[8]. N. Wang, K.-H. Ho, G. Pavlou, "Adaptive Multi-Topology IGP Based Traffic Engineering with Near-Optimal Network Performance," in Proceedings of IFIP-TC6 Networking Conference (Networking), Singapore, May 2008.

[9]. I. Gojmerac, P. Reichl, L. Jansen, "Towards Low-complexity Internet traffic Engineering: The Adaptive Multi-Path Algorithm," Computer Networks, 52 (15), pp. 2894-2907, 2008.

[10]. S. Fischer, N. Kammenhuber, A. Feldmann, "Replex: Dynamic Traffic Engineering based on Wardrop Routing Policies," in Proceedings of ACM CoNEXT conference (CoNEXT '06), Lisboa, Portugal, pp. 1–12, December 2006.

[11]. P. Psenak et al., "Multi-Topology (MT) Routing in OSPF," RFC 4915, June 2007.

[12]. J. Wang, Y. Yang, L. Xiao, K. Nahrstedt, "Edge-based Traffic Engineering for OSPF Networks," Computer Networks, 48 (4), pp. 605-625, July 2005.

[13]. The Abilene topology and traffic matrices dataset available at http://www.cs.utexas.edu/~yzhang /research/AbileneTM

[14]. The TOTEM toolbox available at http://totem.run.montefiore.ulg.ac.be/

[15]. B. Fortz, M. Thorup, "Internet Traffic Engineering by Optimizing OSPF Weights," in Proceedings of IEEE INFOCOM Conference, Tel-Aviv, Israel, 2000.

[16]. B. Fortz et al., "Traffic Engineering with Traditional IP Routing Protocols," IEEE Communications Magazine, 40(10), pp. 118–24, October 2002.

[17]. A. Sridharan, R. Guerin, C. Diot, "Achieving Near-Optimal Traffic Engineering Solutions for Current OSPF/IS-IS Networks," in Proceedings of IEEE INFOCOM Conference, San Francisco, CA, 2003.

[18]. D. Xu, M. Chiang, J. Rexford, "Link-state Routing with Hop-by-hop Forwarding can Achieve Optimal Traffic Engineering, " IEEE/ACM Transactions on Networking.

[19]. Wang et al., "An Overview of Routing Optimization for Internet Traffic Engineering," IEEE Communications Surveys, 10(1), 2008.

[20]. A. Elwalid, C. Jin, S. Low, I. Widjaja, "MATE: MPLS Adaptive Traffic Engineering," in Proceedings of IEEE INFOCOM Conference, 2001.

[21]. S. Kandula, D. Katabi, B. Davie, and A. Charny, "Walking the Tightrope: Responsive yet Stable Traffic Engineering", In Proceedings of ACM SIGCOMM, 2005.

[22]. A. Kvalbein, O. Lysne, "How Can Multi-topology Routing be Used for Intradomain Traffic Engineering," in Proceeding of IEEE SIGCOMM Workshop, 2007.

[23]. J. Lessmann, Z. Yousaf, P. Mannersalo, E. Patouni, S. Klein, P. Loureiro ja G. Athanasiou, ”A collaborative end-to-end resource management framework for cellular networks," *IEEE Wireless communications Magazine (submitted),* 2011.

[24]. Z. Boufidis, E. Patouni, N. Alonistioti, P. Magdalinos, D. Bourse ja K. El-Khazen, ”Exploiting the Reconfiguration Management Plane for Protocol Adaptation and Quality of Service Negotiation in B3G Systems,” tekijä: *Wireless World Research Forum (WWRF)*, Paris, France, 2005.

[25]. L. Qiu et al, “On Selfish Routing in Internet-Like Environments” Proc. ACM SIGCOMM 2003

[26]. Y. Liu et al, “On the Interaction Between Overlay Routing and Traffic Engineering”, Proc. IEEE INFOCOM 2005

[27]. W. Jiang et al, “Cooperative Content Distribution and Traffic Engineering in an ISP Network”, Proc. ACM SIGMETRICS 2009

[28]. D. DiPalantino et al, “Traffic Engineering vs. Content Distribution: A Game Theoretic Perspective”, Proc. IEEE INFOCOM 2009

[29]. S. Klein, I. Karla and E. Kuehn, "Potential of intra-LTE, intra-frequency load balancing," in *IWSON*, Budapest, Hungary, 2011

[30]. 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.

[31]. R. Bolla, R. Bruschi, A. Cianfrani and M. Listanti, "Enabling backbone networks to sleep," *IEEE Network Magazine,* vol. 25, no. 2, pp. 26-31, 2011.

[32]. F. Yu and V. Krishnamurthy, "Efficient radio resource management in integrated WLAN/CDMA mobile networks," *Springer Telecommun. Sys.,* vol. 30, no. 1-3, pp. 177-192, 2005.

[33]. W. Song, W. Zhuang and Y. Cheng, "Load balancing for cellular/WLAN integrated network," *IEEE Network,* vol. 21, no. 1, pp. 27-33, 2007.

[34]. W. Song and W. Zhuang, "Multi-service load sharing for resource management in the cellular/WLAN integrated network," *IEEE Transactions on Wireless Communications,* vol. 8, no. 2, pp. 725-735, 2009.

[35]. T. Taleb, Y. Aoul and S. Schmid, "Geographical location and load based gateway selection for optimal traffic offload in mobile networks," *Networking,* vol. 1, pp. 331-342, 2011.

[36]. T. Wauters, J. Coppens, B. Dhoedt and P. Demeester, "Load balancing through efficient distributed content placement," in *EuroNGI Conference on Next Generation Internet Networks - Traffic Engineering*, Rome, Italy, 2005.

[37]. 3GPP TR 36.902, v9.3.1, “Evolved Universal Terrestrial Radio Access Network (E-UTRAN); Self-configuring and self-optimizing network (SON) use cases and solutions”

[38]. FP7 Socrates deliverable D2.4 “Framework for the development of self-organising networks”

[39]. M. Gruber, S. Borst, and E. Kuehn, “Stable interaction of self-optimization processes in wireless networks”

[40]. Univerself Technical Report D3.1, “Identification of suitable classes of methods for parameter optimization”, September 2011

[41]. A. Kousaridas, N. Alonistioti, A. Mihailovic, “Dynamic compartment formation for coverage optimization of cognitive wireless networks”, IEEE Personal, Indoor and Mobile Radio Communication (PIMRC), 2010, pp. 2255-2260

[42]. OPNET Modeler, www.opnet.com

[43]. T. Schank and D. Wagner, “Approximating clustering coefficient and transitivity”, J. Graph Algorithms and Applications 9 (2005), 265-275

[44]. V. S. Borkar, “Stochastic Approximation, a Dynamic Systems Viewpoint”, Cambridge University Press, and Hindustan Book Agency, 2008

[45]. P.Vlacheas, E.Thomatos, K.Tsagkaris, P.Demestichas, “Autonomic Downlink Inter-Cell Interference Coordination in LTE Self-Organizing Networks”, in Proc. 7th International Conference on Network and Service Management (CNSM) 2011, Poster Sessions, 24 – 28 October 2011, Paris, France

[46]. F.-T. Chen and G.-L. Tao, “A Novel MCS Selection Criterion for Supporting AMC in LTE System”, International Conference on Computer Application and System Modeling (ICCASM 2010), 2010

[47]. Y. Censor, “Pareto Optimality in Multiobjective Problems”, Appl. Math. Optimiz., Vol. 4, pp 41–59, 1977

[48]. N.O. Da Cunha and E. Polak, “Constrained Minimization Under Vector-Valued Criteria in Finite Dimensional Spaces”, J. Math. Anal. Appl., Vol. 19, pp 103–124, 1967

[49]. L. A. Zadeh, “Optimality and Nonscalar-Valued Performance Criteria”, IEEE Trans. Automat. Contr., Vol. AC-8, p. 1, 1963

[50]. L. Lamport, “Proving the correctness of multiprocess programs”, IEEE Transactions on Software Engineering 1977; 3 : 125-143.

[51]. S. Islam, M. Squalli, S. Khan, “Modeling and formal verification of DHCP using SPIN”, International Journal of Computer Science and Applications 2006; 3 : 145-159.

[52]. M. Duflot, M. Kwiatkowska, G. Norman, D. Parker, C. Peyeronnet, C. Picaronny, J. Sproston, “Practical applications of probabilistic model checking to communication protocols”, Chapter in FMICS Handbook on Industrial Critical Systems, IEEE Computer Society Press, 2010. To appear.

[53]. M Duflot, M. Kwiatkowska, G. Norman, D. Parker, “A formal analysis of bluetooth device discovery”, International Journal on Software Tools for Technology Transfer 2006; 8; 621-632.

[54]. M. Kwiatkowska, G. Norman, D. Parker, “Analysis of a gossip protocol in PRISM”, ACM SIGMETRICS Performance Evaluation Review 2008; 36; 17-22.

[55]. The PRISM manual, available at the PRISM website, http://www.prismmodelchecker.org/

[56]. M. Huth, M. Ryan, “Logic in Computer Science: Modeling and Reasoning about Systems”, Cambridge University Press 2004.

[57]. J. Katoen, I. Zapreev, E. Hahn, H. Hermanns, D. Jansen, “The ins and outs of the probabilistic model checker MRMC”, Proceedings of the 6th International Conference on the Quantitative Evaluation of Systems 2009; 167-176.

[58]. J. Loyola, M. Charalambides, I. Aib, J. Serrat, G. Pavlou, R. Boutaba, “Business-driven Management of Differentiated Services,” Proceedings of 12th IEEE/IFIP Network Operations and Management Symposium (NOMS), Osaka, Japan, April 2010.

[59]. S. Basagni, M. Nati and C. Petrioli, “Localisation Error-Resilient Geographic Routing for Wireless Sensor Networks”, IEEE GLOBECOM 2008.

[60]. UniverSelf D2.1 deliverable “UMF Specifications Release 1”.

[61]. M. Huth and M. Ryan, “Logic in Computer Science: Modeling and Reasoning about Systems,” Cambridge University Press 2004.

[62]. M. Chiesa, L. Cittadini, G. Di Battista, S. Vissicchio, “Local transit policies and the complexity of BGP Stability Testing,” in Proceedings of IEEE INFOCOM Conference, Shanghai, China, April 2011.

[63]. L. Gao, J. Rexford, “Stable internet routing without global coordination,” in Proceedings of ACM SIGMETRICS, Santa Clara, California, June 2000.

[64]. A. Flavel, M. Roughan, N. Bean, A. Shaikh, “Where’s Waldo? Practical searches for stability in iBGP” in Proceedings of IEEE International Conference on Network Protocols (ICNP), Orlando, FL, October 2008.

[65]. T.G. Griffin, G. Wilfong, “An Analysis of BGP Convergence Properties,” in Proceedings of ACM SIGCOMM, Cambridge, Massachusetts, August 1999.

[66]. K. Ramachandran, I. Sheriff, E. Belding, K. Almeroth, “Routing Stability in Static Wireless Mesh Networks,” in Proceedings of Passive and Active Measurement Conference, Louvain-la-neuve, Belgium, April 2007.

[67]. M. Charalambides, G. Pavlou, P. Flegkas, J.R. Loyola, A.K. Bandara, E.C. Lupu, A. Russo, N. Dulay, M. Sloman, “Policy Conflict Analysis for DiffServ Quality of Service Management” IEEE TNSM, Vol. 6, No. 1, March 2009.

[68]. M. Charalambides, P. Flegkas, G. Pavlou, A. Bandara, E. Lupu, A. Russo, N. Dulay, M. Sloman, J. Loyola, “Policy Conflict Analysis for Quality of Service Management,” in Proceedings of 6th IEEE Workshop on Policies for Distributed Systems and Networks (Policy 2005), Stockholm, Sweden, June 2005.

[69]. M. Charalambides, P. Flegkas, G. Pavlou, J. Loyola, A. Bandara, E. Lupu, A. Russo, N. Dulay, M. Sloman, “Dynamic Policy Analysis and Conflict Resolution for DiffServ Quality of Service Management,” in Proceedings of IEEE/IFIP Network Operations and Management Symposium (NOMS 2006), Vancouver, Canada, April 2006.

[70]. T. Samak, E. Al-Shaer, H. Li, “QoS Policy Modeling and Conflict Analysis,” in Proceedings of 9th IEEE Workshop on Policies for Distributed Systems and Networks, Palisades, New York, 2008.

[71]. J. Baliosian, A. Devitt, “Forecasting Unstable Policy Enforcement,” in Proceedings of the International Conference on Systems and Networks Communications (ICSNC), Tahiti, French Polynesia, October 2006.

[72]. E. Al-Shaer, H. Hamed, “Modeling and Management of Firewall Policies,” IEEE Transactions on Network and Service Management, vol. 1(1), 2004.

[73]. L. Yuan, et al., “FIREMAN: A Toolkit for Firewall Modeling and Analysis,” in Proceedings of IEEE Symposium on Security and Privacy, Oakland, California, 2006.

[74]. S. Ferraresi, et al., “Automatic Conflict Analysis and Resolution of Traffic Filtering Policy for Firewall and Security Gateway,” in Proceedings of International Conference on Communications, Glasgow, Scotland, 2007.

[75]. S. Reiff-Margainiec, K. J. Turner, “Feature Interaction in Policies,” Computer Networks, 45(5), pp. 569-584, 2004.

[76]. L. Blair, K. Turner, “Handling Policy Conflicts in Call Control,” in Proceedings of 8th International Conference on Feature Interaction, pp. 39-57, IOS Press, Amsterdam, 2005.

7 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.

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.

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.

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 optimisation with a model would be the optimisation 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 optimisation problems where solutions can be found in a straightforward way without getting trapped in local optima. For the class of non-convex optimisation 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 automaticity capabilities, easier use of management tools and empowering the network with inbuilt cognition and intelligence. Additional benefits include reduction and optimisation in the amount of external management interactions, which is key to the minimisation 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-optimisation – Selection and adjusting best (network and/or service parameters or behaviours from some set of available alternatives and/or minimise or maximise a utility function by systematically choosing the values of the parameters from within an allowed set in an autonomous way. Self-Optimisation 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-Optimisation is one of the key pillars of the Self-Organising Networks (SON) management paradigm.

Traffic engineering – Concerned with the design and provisioning of communication networks in order to provide the required quality to contracted services, while at the same time optimising the usage of resources.

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.

UMF – A framework and architecture for the management of future networks and services being developed by the UniverSelf project.

8 Abbreviations

3G	3rd Generation Mobile Telecommunications
3GPP	3rd Generation Partnership Project
AATE	Application-Agnostic Traffic Engineering
ALBA	Adaptive Load Balanced Algorithm
ALTO	Application Layer Traffic Optimisation
AP	Access Point
AS	Autonomous System
BGP	Border Gateway Protocol
BS	Base Station
CC	Context Client
CDF	Cumulative Distribution Function
CDN	Content Distribution Network
CL	Control Loop
CO_FB	Cooperation FB
CP	Context Provider
CQI	Channel Quality Information
CRM	Collaborative Resource Manager
CSI	Channel State Information
CTMC	Continuous-Time Markov Chain
CTMDP	Continuous-Time Markov Decision Process
CUR	Capacity Usage Ratio
DACoRM	Decentralised Adaptive Coordinated Resource Management
DiffServ	Differentiated Services
DIT	Demand Information Table
DME	Decision Making Entity
DTMC	Discrete-Time Markov Chain
EngE	Energy Efficiency
EPC	Evolved Packet Core
ePDG	Evolved Packet Data Gateway
FB	Functional Block
GGSN	Gateway GPRS Support Node
GOV_FB	Governance FB
GPI	Geographic Priority Index
GSM	Global System for Mobile Communications
GUI	Graphical User Interface
HetNet	Heterogeneous Network (in the context of this deliverable: a network consisting of nodes with different/heterogeneous form factors)
ICIC	Inter-Cell Interference Coordination
IETF	Internet Engineering Task Force
IKB_FB	Information and Knowledge Building FB

INO	In-Network Overlay
IGW	Internet Gateway
ISI	Inter Symbol Interference
KPI	Key Performance Indicator
LAN	Local Area Network
LB	Load Balancing
LIT	Link Information Table
LSP	Label Switch Path
LTE	Long Term Evolution
MDP	Markov Decision Process
MIMO	Multiple Inputs Multiple Outputs
MME	Mobility Management Entity
MON_FB	Monitoring FB
MPLS	Multi-Protocol Label Switching
MTR	Multi-Topology Routing
OF	Overlapping Factor
OFDM	Orthogonal Frequency Division Modulation
PGW	Packet Gateway
PoP	Point of Presence
PRB	Physical Resource Block
P2P	Peer-to-Peer
PCRF	Policy and Charging Rules Function
PDM_FB	Policy Derivation & Management FB
PHB	Per Hop Behaviour
PM_FB	Profiles and Models FB
QoE	Quality of Experience
QoS	Quality of Service
RAT	Radio Access Technology
RMA	Resource Management Agent
RN	Relay Node
RNC	Radio Network Controller
RNTP	Relative Narrowband Transmit Power
RRC	Radio Resource Control
RRM	Radio Resource Management
S-D	Source-Destination
SEA_FB	Solution Evaluation and Assessment FB
SGSN	Serving GPRS Support Node
SGW	Service Gateway
SINR	Signal-to-Interference and Noise Ratio
SLA	Service Level Agreement
SON	Self-Organising Network
SPF	Shortest Path First
SSE_FB	Solution Selection and Elaboration FB

TB	Transport Block
TE	Traffic Engineering
TM	Traffic Matrix
TTI	Transmission Time Interval
U_DC	UMF intra-Domain Controller
U_FC	UMF inter-domain controller/Federated Controller
UC	Use Case
UE	User Equipment
UMF	Unified Management Framework
UMTS	Universal Mobile Telecommunications System
WLAN	Wireless Local Area Network
WSN	Wireless Sensor Network

9 Appendix

9.1 Traffic Engineering

9.1.1 State of the Art

In this section some state of the art analysis on methodologies related to the work presented in Section 2 is given.

9.1.1.1 Inter-Cell Interference Coordination

Inter-cell interference coordination (ICIC) is inherent in any multi-cellular networks mainly due to inter-cell and intra-cell resource reuse. Clustering, cell sectorisation and static/dynamic fractional frequency reuse are common techniques used to control ICIC in conventional cellular networks. The aggressive reuse suggests that entire system resources will be made available in each cell, whereas intra-cell spatial reuse can be applied to further improve resource utilisation. On the network level, dynamic inter-cell coordination could be the candidate for addressing interference problems. Dynamic inter-cell coordination can be employed by exchanging vital interference information between eNodeBs over the backbone network connection to achieve the interference avoidance gain and improve user throughput and fairness. In [1], general ways of dynamic ICIC given to discretise resource allocation with transmit power levels with steering coefficients with directional antennas and transmission rates are provided. In [2], ICIC in relay networks with semi-static frequency reuse in RNs with weighted scheduling schemes are proposed to improve cell-edge throughput and cell-centre throughput. In [3], dynamic resource allocation scheme for single cell multi-hop relay networks are given to improve the total cell throughput but without considering ICIC.

The considered ICIC approaches use semi-static ICIC which is giving comparatively average cell mean throughput and user throughput with less fairness for cell edge users with no adaptive tuning mechanisms. Due to the high increase in mobile traffic demands with different services with variable bandwidth requirements, tuning mechanisms are needed. The proposed dynamic ICIC in multi-hop cellular networks allows for more self-adaptive frequency reuse schemes for intra-cell and inter-cell coordination.

9.1.1.2 Interactions between TE and Overlay Network Operations

In the literature, a number of works [25][26][27][28] have investigated the interaction between TE and overlay network operations. We can classify these works into two categories.

The first category focuses on the interactions between network-layer routing configurations decided by TE and logical overlay routing on top [25][26]. In this scenario, TE and the overlay respectively adjust their own *routing* strategies in turn, based on each other's decisions. Compared with this type of interaction, the key difference from our work is as follows: we focus on the P2P overlay side which only considers how to select the best partner peers (i.e. the other endpoint of individual P2P connection sessions), rather than considering routing in the overlay.

Some other works [27][28] focuses on CDN (Content Distribution Network) –like paradigms, and considers the interaction between network-layer routing decisions made by TE and application-layer content server selections. Our work differs from this category in the following three features. Firstly, in P2P overlay networks, peers, as both content producers and consumers, have highly dynamic join/departure patterns, while in the CDNs of [27][28] content servers are statically provisioned in the network for providing content delivery services. Secondly, we consider *symmetric* content exchange patterns: in P2P overlays a peer not only requests data from, but also provides content to other peers; this differs from the previous studies in which a specific set of clients only download data from a number of dedicated content servers. Finally, in P2P overlays each peer needs to simultaneously fetch chunks of content from a set of partners, while in conventional CDNs a client typically requests content from one specific server at a time.

9.1.1.3 Intra-domain Traffic Engineering

Current practices for intra-domain TE rely on off-line approaches, where a central management system is responsible for computing routing configurations, especially tuning link weights based on the estimation of the traffic demand. The goal of these approaches is to find a routing configuration that optimises the network performance over long timescales, e.g. weekly or monthly. Off-line TE schemes have been extensively investigated both in the context of MPLS-based TE by using MPLS paths and in the context of IP-based TE by determining heuristics to tune the link weights that optimise some objective function given a set of traffic matrices [15][16][17][18]. As such, off-line approaches may be sub-optimal in the face of unexpected traffic demand.

In contrast to these off-line schemes, online TE approaches do not rely on the knowledge of any traffic matrix to configure the routing or the link weights. Instead, they dynamically adapt the settings in short timescales in order to rapidly respond to traffic dynamics [19]. These schemes do not rely on any knowledge of future demands to configure the settings but instead use monitored real-time information from the network. In order to satisfy the future traffic demands, online TE approaches aim at adaptively distributing the traffic load as evenly as possible onto the network according to the changing traffic conditions.

There have been some proposals for both online MPLS-based TE such as [20][21] and online IP-based TE such as [8][9][10]. These approaches focus on dynamically adjusting the volume of traffic (represented by splitting ratio) sent across several available paths between each S-D pair in the network according to real-time traffic information.

Multi-topology routing can be used (MTR) [11] as the underlying network routing protocol to provide a set of available routes between each pair of edge nodes, e.g. [12][22]. MTR is a standardised extension to the common IGP routing protocols OSPF and IS-IS, that aims at determining several independent virtual IP topologies based on a single network topology, each having its own independent routing configurations, especially its own link weight settings. Based on these link weight settings, the Shortest Path First (SPF) algorithm can be applied independently between each source-destination (S-D) pair in each topology. Thus, it is possible to compute a set of paths between each pair of end points in the network with each path being related to one virtual topology. More precisely, the traffic demand between any pair S-D is virtually split into n independent sets at ingress nodes and each traffic set is assigned to one of the n topologies and routed according to that topology's configuration. Results in [8][12][22] show that only a small number of topologies (typically between 3 and 5) is enough to offer pretty good path diversity.

Dynamic adaptation of traffic splitting ratios was initially proposed by two MPLS-based TE solutions, MATE [20] and TeXCP [21], where ingress routers use periodical information from the network to adjust the ratios. Unlike MATE and TeXCP where re-configurations are performed at ingress nodes only, in AMP [9] and REPLEX [10], all the nodes in the network are responsible for dynamically splitting the traffic between the different available next hops, based on information received from upstream routers.

Unlike the above distributed approaches, the authors in [8] use a central controller that has a global knowledge of the network state to perform the re-configurations. The advantage of such a centralised decision-making process is that the consistency between different re-configuration actions is guaranteed. However, using a centralised approach is less scalable than a solution where decisions are taken by nodes themselves inside the network, since at each re-configuration period the central controller needs to gather information from all the links and nodes in the network, which incurs a significant communication overhead.

9.1.1.4 Collaborative End-to-End Load Balancing for Cellular Networks

An overview of a sample mobile network topology is presented in Figure 31. It is divided into different network domains, from the service to the user domain. For each domain, the right side of the figure lists possible load optimisation actions available within that specific domain. On the left side, one can find the context data, which must be collected in each domain to be able to make intelligent overall resource management decisions. The depicted topology is only an example for a mobile network comprehensive enough to be used for illustration purposes. It encompasses a basic 3G and LTE core infrastructure and several 3G, LTE and WLAN base stations.

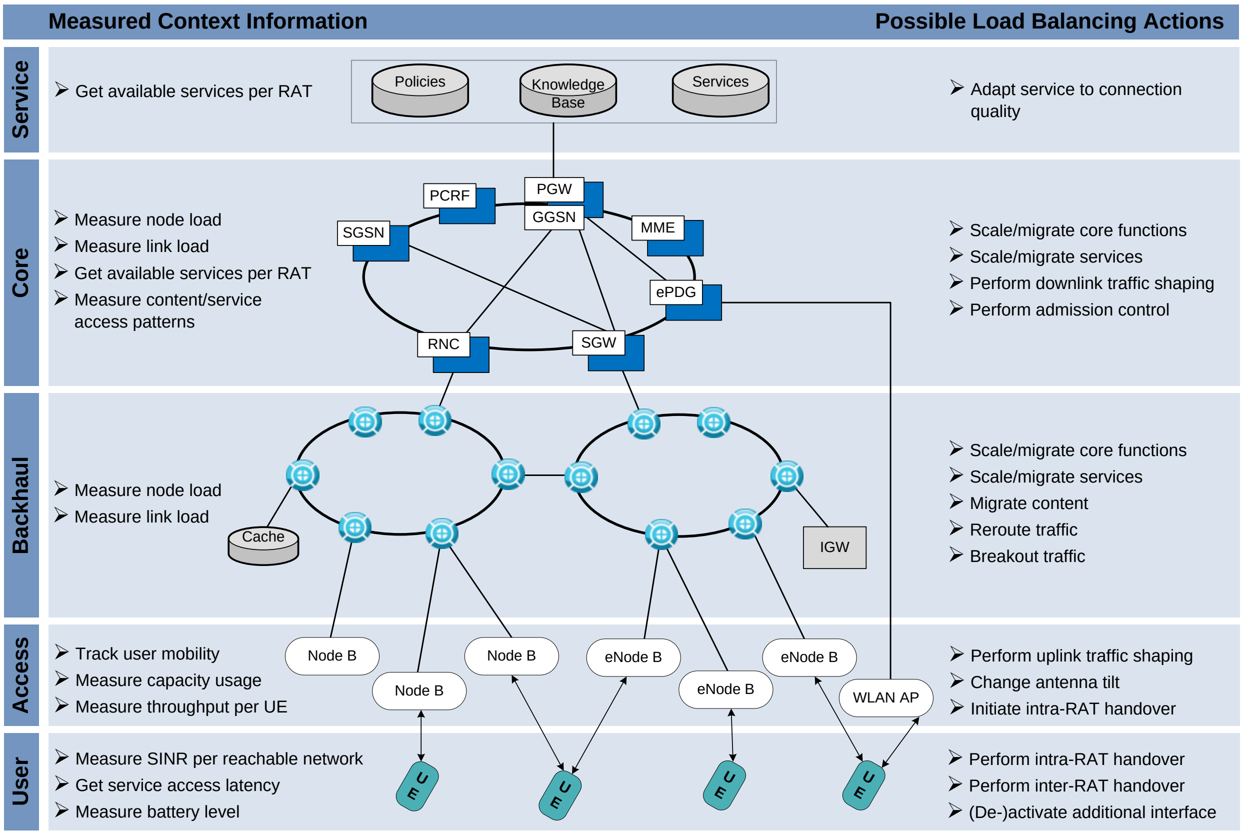


Figure 31: End-to-end resource management view

On the user side, the most obvious load balancing options are intra-RAT (Radio Access Technology) and inter-RAT handover [29]. E.g., if an LTE coverage cell is overloaded, the user device can change to a different eNodeB (often resulting in poorer but still acceptable wireless connectivity for that device) or it may handover to a 3G or even to a WLAN base station. Assuming all RATs depicted in Figure 31 belong to the same operator, the 3G and LTE backhaul network might have certain interconnection points, which means that the old and new user plane paths might actually merge somewhere in the backhaul. As opposed to this, WLAN access is connected to the mobile core network via a dedicated gateway (ePDG). This is another example showing that load balancing actions in the different network domains are not independent. Another option on the user side in congested scenarios is to enable multi-homing, i.e., to offload parts of the traffic to a different RAT.

On the access front, means of load balancing are network-initiated handover (either within the same RAT [30] or between different RATs) and antenna tilt adaptation. Besides, base stations normally also perform admission control and traffic shaping for uplink traffic [31][32]. This allows not only blocking excess traffic per user, but also can be used to reduce the amount of traffic that enters the backhaul network in cases of backhaul congestion [33][34].

Since the backhaul network's single purpose is to transport traffic between the base stations and the core network, load balancing options in this domain have traditionally be limited to rerouting of traffic. Typically, centralised path provisioning is employed in operator networks, which is why rerouting normally translates into reconfiguration of Label Switched Paths (LSPs) or Virtual LANs.

Recently, load optimisation based on traffic breakout [35] has received great attention. Conceptually, this refers to the scenario where traffic between user devices and the Internet is not routed through the core network to the operator's Packet Data Network Gateway (PGW), but rather to backhaul routers which have a direct connection to an Internet gateway (IGW) from an Internet Service Provider. The advantage is that this can save a lot of resources in both backhaul and core, since traffic is leaving the network early. Technically, the term "breakout" refers to the fact that such traffic is normally routed along pre-established tunnels between eNodeB and PGW and must therefore break out of this tunnel in order to go via the IGW.

Another option for load optimisation in the backhaul domain is opened up by introducing content caches [36]. The idea here is to cache highly popular content closer to the operator network. While currently, content is normally delivered by content providers' over-the-top of the mobile network, mobile operators are starting to deploy content caches in their data centers, which are located behind and accessed via the PGW. However, one can clearly make out the trend of putting caches yet closer to the access, i.e., into the backhaul network. This not only decreases the content access latency but also reduces the amount of traffic that needs to be served through the core network. As such, content caching is clearly a load optimisation measure. By using breakout mechanisms, caches can be locally accessed. If content is not available in a certain cache, the request can even be redirected to other backhaul caches. In essence, the operator's caches in the backhaul network thus form a mobile content delivery network.

In addition, other entities such as services (e.g. remote desktop, mobile games, etc.) or even core network functions (PGW, SGW, MME) can also be hosted in the backhaul network and migrated based on load conditions. This obviously requires more general-purpose types of nodes in the transport network instead of plain switches or dedicated-purpose machines such as classical gateway nodes. Yet if traffic intense services are located closer to the access, load can be reduced in the transport network and access latency be improved. Load-aware instantiation (and migration) of core network functions on general-purpose nodes in core and backhaul is certainly the most radical concept, but after all only a natural consequence of the previous concepts of content and service migration. All these concepts are applicable as load optimisation strategies to backhaul as well as core network domain (cf. Figure 31). Similar to the access side, the core network may also perform traffic shaping and admission control.

Finally, load management could also be supported in the service domain. This is primarily achievable by adapting service delivery to the quality of the end-to-end connectivity. If the transport network is congested or the wireless access network connectivity is poor, an option would be, e.g., to select a more appropriate, less bandwidth-intense codec.

9.1.2 Simulation/Implementation Details

9.1.2.1 Dynamic Inter-Cell Interference Coordination in Multi-hop Cellular Network: Problem Formulation and Simulation Setup Details

9.1.2.1.1 Problem Formulation

The utility maximisation problem can be formulated as follows:
Maximise:

eq. 1: $$\sum_{i}\left[\sum_{k=1}^{K}\sum_{n=1}^{N}u_{k,n}^{i}\,\rho_{k,n}^{i}\right];u_{k,n}^{i}=s_{k}^{i}\,r_{k,n}^{i}$$

Subject to:

eq. 2: $$I_{n}^{i}=\sum_{k=1}^{K}\rho_{k,n}^{i}=\begin{cases}0; & chunk\,n\,is\,restricted\ \ in\ i\\ 1; & otherwise;\end{cases}$$

eq. 3: $$\rho_{k,n}^{i}\in\{0,1\};\forall\{k,n\}$$

eq. 4: $$s_{k}^{i}=\frac{\overline{R^{(i)}}}{(R_{m}^{i}+\delta)}$$

eq. 5: $$\gamma_{k,n}^{i}=\frac{P_{c}\,H_{k,n}^{(i,i)}}{P_{c}\sum_{j=1}^{J}H_{k,n}^{(i,j)}.I_{n}^{(j)}+P_{c}\sum_{l=1}^{L}H_{k,n}^{(i,l)}.I_{n}^{(l)}+P_{N}}$$

In eq. 1, $u_{k,n}^{(i)}$ and $r_{m,n}^{i}$ the achievable utility and the rate (in bps/Hz) on block n, respectively, seen by UE m in sector i; $s_{k}^{(i)}$ is the UE traffic demand factor that indicates the service status of UE k, eq. 2 is an indicator to show whether chunk n is restricted or not, eq. 3 is an indicator showing allocation of chunk n to UE k, eq. 4

gives the service status of UE with R_m as the time average throughput achieved by UE k and $\overline{R}$ as the average throughput across all UEs, eq. 5 gives SINR experienced by UE k on chunk n from sector RNs and eNodeBs, where P_c is the transmit power applied on each derived from equal power distribution from RNs and eNodeBs such that $P_c = P_t/N$, P_t is the total transmit power per sector RN and eNodeBs and N is the number of available chunks; P_N is thermal noise power over the chunk bandwidth; $H_{k,n}^{(i,j)}$ and $H_{k,n}^{(i,l)}$ the link gains to the first-tier dominant and other non-dominant interferer sectors, respectively; I_n^j I_n^l take the value of 0 or 1 depending on whether or not the nth chunk is restricted in non-dominant sector j and dominant sector l, respectively.

9.1.2.1.2 Simulation Details

The system parameters used in the simulations are summarised in Table 2.

Table 2: Simulation parameters

Parameter	Value
Frequency	2.14 GHz
Bandwidth	5 MHz (25 chunks)
Thermal noise density	-174 dBm/Hz
Receiver noise figure	9 dB
nTX x nRX antennas	2 x 2
TX mode	OLSM (Open Loop Spatial Multiplexing)
Simulation length	100 TTIs
Number of simulations	200 per scenario
Inter eNodeB distance	500 m
Minimum Coupling loss	70 dB
Macroscopic path loss	$128.1 + 37.6 \log_{10}(R)$
Shadow fading	Lognormal, space-correlated, μ=0, σ = 10 (dB)
Shadow fading correlation	Inter-site: 0.5, Intra-site: 1
eNodeB TX power	43 dBm
RN TX power	30 dBm
Number of RNs per cell	3
Number of UEs per sector and centre	5x4 = 20 UEs per cell
Microscale fading	PedB uncorrelated, time-correlated
UEs position	Homogeneous. UEs located in RN sector and eNodeB sector, 5 UEs/sector
UE speed	5 KM/h
BS Antenna pattern	$A(\theta) = -\min[12(\theta/\theta3_{dB)})^{2,}20]$
BS Antenna gain	15 DBi
Scheduler	Proportional fair
Subcarrier averaging algorithm	MIESM (Mutual Information Effective Signal to Interference and Noise Mapping
CQI delay	3 TTIs
Traffic model	Full Buffer

9.1.2.2 Case Study for Collaborative End-to-End Load Balancing

Figure 32 presents a case study for collaborative load balancing in access and backhaul network domains. The trigger event is the detection of traffic overload in the access domain sent by the respective RMA. Following the retrieval and communication of the contextual information to the Decision Making Module, the QoE/QoS Management Module translates QoE to QoS constraints. This information is then conveyed to the RMA of the access domain, which is responsible for specifying, implementing and evaluating the appropriate load balancing decision. We assume that in this case it is decided to apply a handover of a user to a neighbouring BS, due to resources saturation. However, such an action fails since the resources of the neighbouring BS are also saturated. This situation is reported to the Decision Making module, which then accordingly computes the appropriate load balancing strategy.

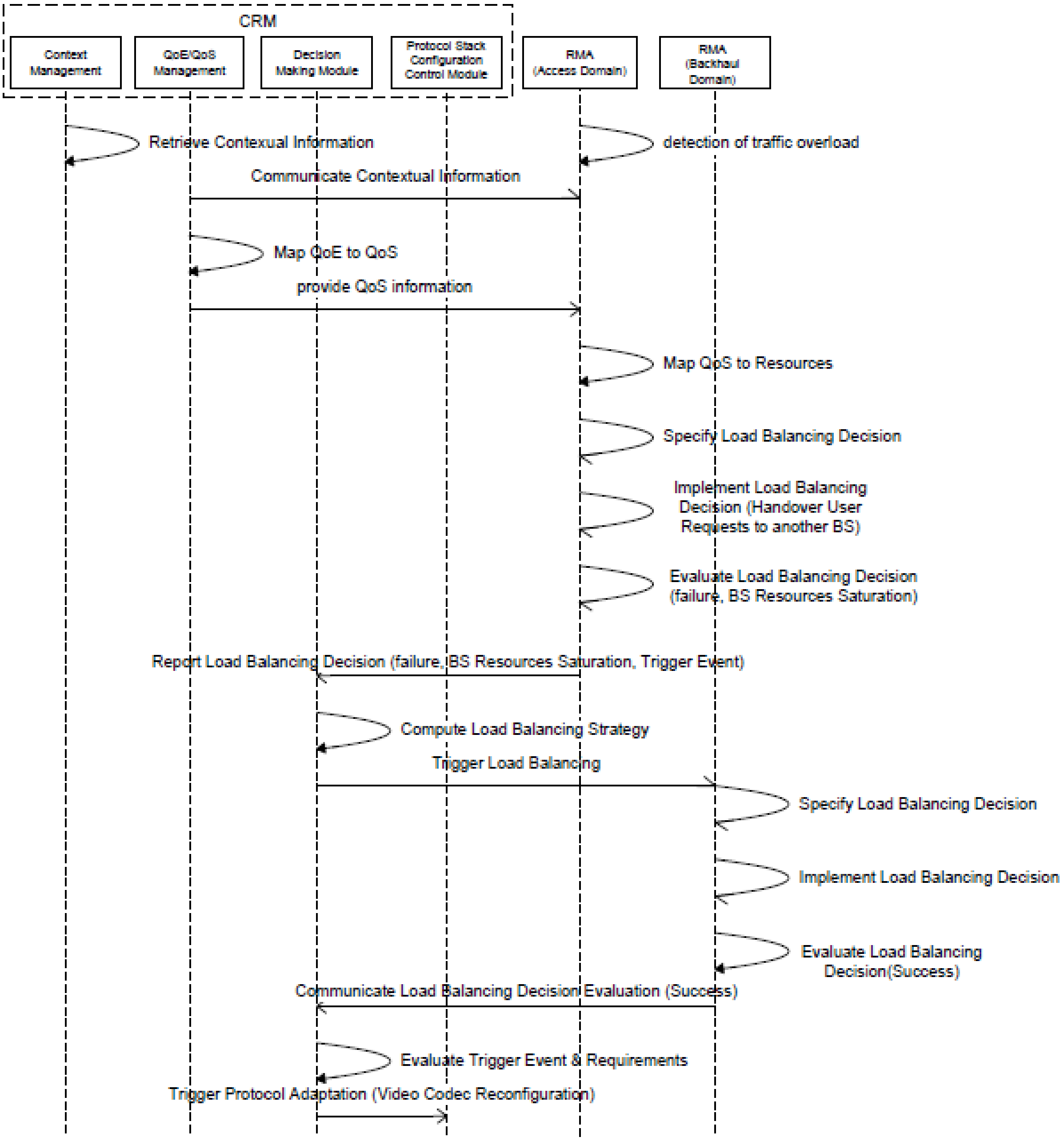


Figure 32: Signalling diagram for collaborative load balancing in access and backhaul network domains

To this end, the Decision Making module triggers the RMA responsible for the backhaul domain to select and realise the required load balancing actions. The outcome of the decision is rerouting to backhaul caches. Such decision is evaluated to be successful. Next, the Decision Making Module checks if the initial QoE/QoS requirements are met. Since the requirements were not fulfilled, the Decision Making Module triggers a protocol adaptation procedure that is coordinated by the Protocol Stack Configuration Control Module. This module is responsible to control the required video codec adaptation. This example illustrates the value of the collaborative end-to-end resource management procedure by showcasing two major challenges that need to be addressed:

1. *Which network domain should handle the load balancing action*: in this work we consider a simple hierarchical approach; first we investigate whether load balancing in the access domain provides a solution. If not, respective procedures in the backhaul domain are considered.
2. *Which type of action should take place*: this is related to specifying the exact load balancing action, e.g., selection of the Base Station (BS) for the user reallocation.

9.2 SON Interaction – State of the Art

Figure 33 shows the overall interaction system as introduced in the European project Socrates:

“A (SON-) controller determines the actual network performance by interpreting […] measurements […] and error reports. […] The (SON-) controller decides whether the current network performance exceeds certain thresholds and the SON functionality needs to be activated in order to counteract the actual network performance degradation. Since the SON functionalities of the use cases are not able to counteract every kind of network performance degradation it is useful to identify the SON functionalities (use cases) to be activated at a certain network state. The (SON-) controller identifies these different states and distinguishes between different triggers that are fulfilled. After the (SON-) controller decided which trigger is fulfilled the SON functionalities will be activated, i.e. the influence of the (SON-) controller ends.” [38]

“The self-organisation layer includes all SON processes initiated by the triggers. [...] Operator policies are taken into account in the coordination layer. This layer includes the conflict handling [...] and will avoid self-organisation loops. The (SON-) coordinator will also be able to monitor the success of the SON functionalities and give feedback to the SON processes. The new parameter settings found by the SON processes will only be applied to the network if no conflicts are detected.” [38]

The Socrates project has also provided trigger strategies for a number of concrete instances that can also be found in [38].

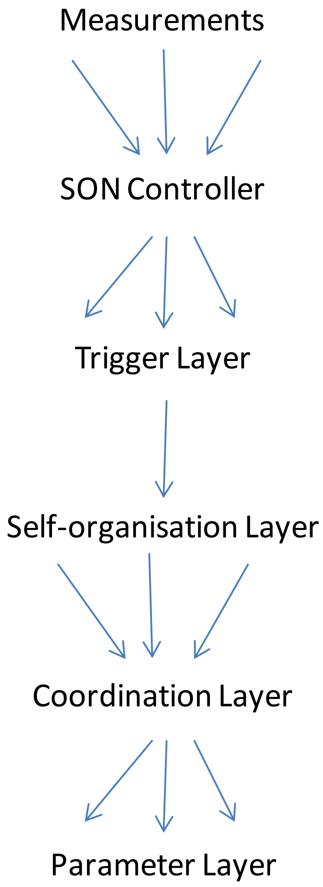


Figure 33 : Socrates overall interaction system [38]

9.3 Network Stability – State of the Art

9.3.1 Policy Conflict Analysis

Policy-based management provides the ability to re-configure networks so that high-level goals are achieved and facilitates flexibility and adaptability in that the policies can be changed without changing the implementation. However, inconsistencies may arise in the policy specification (conflicts) that can have catastrophic effects on the network operation and supported services.

There has been considerable work addressing the issue of policy conflicts, which focuses on different applications domains. The authors of [72], [73], and [74] have focused on techniques for analysing firewall policies for networks. All possible firewall rule relations are formally defined and are used to identify and classify policy conflicts. Their resolution is based on the relative ordering of rules in a filtering policy and a degree of automation is proposed for some conflict types by removing or re-ordering rules. Another application domain is that of telecommunications. In [75] the authors identify the analogy between policy conflicts and feature interactions and they identify different approaches that could potentially be used to detect and resolve conflicts. This work is extended in [76] where specific resolution processes are proposed to handle call control policy conflicts both in centralised and distributed settings.

One of the most popular application domains for policies has been QoS management. The authors of [67] identify and classify policy inconsistencies in this domain based on their characteristics, which are used to describe the reasons and the conditions under which a conflict will arise. They mainly distinguish between conflicts that can be detected statically through off-line analysis at policy specification-time [68], and those that can only be determined dynamically during system execution when policies are enforced [69]. In this work static conflicts are detected through analysis initiated manually by the system administrator; conflicts represent inconsistencies between policies and are typically resolved by amending the policies. Run-time conflicts are detected by a process that monitors policy enforcement and detects inconsistent situations in the system's execution. Conflict resolution is achieved automatically by enforcing special rules. Unlike other resolution approaches, this does not involve identifying which of the conflicting policies will prevail based on their relative priority, but provides separate resolution rules that handle potential inconsistencies. These rules are effectively obligation policies which are pre-specified by the administrator and their triggering events are conflict occurrences rather than network events. Also, they are generic enough with only few required per conflict type to cater for multiple occurrences of the same inconsistency. The overall approach is based on the use of a logic formalism (event calculus) for which seamless and efficient mapping mechanisms are provided. Its use allows for advanced reasoning methods and provides the means to not only identify a conflict but also generate an explanation as to how that conflict occurred.

Policy conflicts in the domain of QoS management are also the subject of the work in [70]. Here, the authors identify conflicts among resource management policies at the router level, which define the treatment of a traffic flow on network nodes by setting parameter values for BW allocation, queue size, drop method, and priority for the various Per-Hop Behaviours (PHBs). Inconsistencies among these policies are classified according to the scope in which they occur: intra-PHB conflicts arise within the flow properties at a specific node and inter-PHB conflicts occur between policy definitions across different nodes.

9.3.2 Formal Verification

Formal verification methods, and the corresponding front-end tools, provide a systematic way to assess the correctness of protocols, processes and systems. Their main difference compared to simulation methods and tools is that instead of only examining a limited area of the operational space of the system under consideration, they can be used to examine the whole state space of possible operations and conditions under which the system may operate. This means that all possible combinations of inputs and actions can be taken into account and, therefore, all possible outputs can be derived and evaluated.

One could regard the outcome of formal verification methods as the outcome of an infinitely large number of simulation runs. This means that contrary to simulations, formal verification methods are capable of capturing conditions and operations that may otherwise remain unnoticed, even after a very large number of simulation runs. Traditionally, correctness means guaranteeing two properties: *liveness*, that is some desired properties will be satisfied eventually, and *safety*, that is some undesired properties will never occur [50]. Two notable counterexamples of such properties are the existence of deadlock and livelock states. A deadlock state is a state where a deterministic loop occurs, which doesn't allow the system to leave that state; in other words the system stalls. A livelock state is a state that allows the system to exit from; however, all possible exit actions will eventually lead the system back to this very same state; in other words the system will not "progress" any further.

Numerous examples of applying formal verification tools in that context can be found in the literature (e.g. [51] and [52]). Works such as [53] and [54] suggest that formal verification methods can be used not only to assess the correctness -as defined above- of protocols, processes and systems, but to additionally derive performance related bounds, such as time to converge or time to reach a desired state. Contrary to simulations, which can derive performance bounds with a limited degree of confidence, by using formal verification tools for performance evaluation, it is possible to derive in many cases the absolute worst, best, and average performance bounds.

Probabilistic model checking, which is one of the main methods of formal verification, is suitable for modelling and analysing systems (the term system can refer to protocols as well as to processes) that exhibit probabilistic behaviour [55]. It involves the construction of a probabilistic model describing the system to be analysed, typically in the form of a state-transition system where states of this model represent the ways in which the system can evolve, associated with likelihood probabilities for their occurrence. In probabilistic model checking, four types of probabilistic models are commonly used, depending on the characteristics of the system to be modelled and analysed; these are Discrete-time Markov Chains (DTMCs), Markov Decision Processes (MDPs), Continuous-time Markov Chains (CTMCs), and Continuous-time Markov Decision Processes (CTMDPs) ([56], [57]). According to the probabilistic model deployed, appropriate temporal logics, to reason about the validity of properties -as expressed through formulas during system evolution- are used.

In DTMCs, all transitions can take place in discrete (time) steps and the associated probabilities describe the likelihood of moving from that given state to any other possible state in the subsequent step. Since the behaviour of a DTMC is fully probabilistic, the likelihood of a particular event occurring can be quantified over all the possible system evolutions (the term path is commonly used to refer to distinct system evolutions in time).

MDPs extend DTMCs to model non-deterministic behaviour; that is behaviour where the transition probabilities cannot be clearly defined. For example, probabilities for transitions triggered by external factors at random instances or incurred due to poor/unknown behaviour specification; being therefore difficult to model using a unique probability distribution. To overcome this, in MDPs each state is associated with a set of probability distributions and a transition between states occurs in two steps [54]: first, there is a non-deterministic choice between available distributions in the current state, and then, the next state is selected at random according to the chosen distribution. Contrary to DTMCs, when MDPs are used one can reason about the minimum and maximum (*absolute* and *expected/weighted*) likelihood of an event occurring over all the resolutions of non-determinism but not for the "exact" average probability over all the possible paths.

In CTMCs, time is modelled not in discrete steps, but rather in a continuous manner. Therefore, CTMCs offer a much "denser" notion of time compared to DTMCs and MDPs. In CTMCs, transitions are associated with rates rather than probabilities [54]. These rates represent parameters of negative exponential distributions and give the delay until the transition is enabled. CTMDPs, which constitute an area of active research interest themselves at the moment, extend CTMCs to take into account non-deterministic behaviour, the way MDPs extend DTMCs for the same reason.