D4.2

Interoperability of terrestrial and satellite links: high-level functional specification

Grant Agreement nº: 645047
Project Acronym: SANSA
Project Title: Shared Access Terrestrial-Satellite Backhaul Network enabled by Smart Antennas
Contractual delivery date: 31/01/2017
Actual delivery date: 10/05/2017
Contributing WP 4
Dissemination level: Public
Editors: TAS-E
Contributors: CTTC, AVA, OTE

Abstract:
This Deliverable contains the outcome of Task 4.1 “Interoperability of terrestrial and satellite links”. It provides the functional specification for the main elements involved in the SANSA network architecture, the Hybrid Network Manager (HNM) and the Intelligent Backhaul Node (IBN). Furthermore, a full description of the main interoperability scenarios foreseen is presented. The interactions between the main functionalities present at HNM and IBN level and the description of the simulation framework to simulate a subset of these functionalities will be the focus of the analysis.
Document History

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Editor</th>
<th>Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>29/07/2015</td>
<td>TAS-E</td>
<td>Document creation with proposed ToC</td>
</tr>
<tr>
<td>1.0</td>
<td>28/10/2015</td>
<td></td>
<td>First version with contributions from CTTC, OTE, AVA and TASE.</td>
</tr>
<tr>
<td>1.1</td>
<td>20/11/2015</td>
<td>CTTC</td>
<td>Inclusion of a structured SOA of routing protocols</td>
</tr>
<tr>
<td>1.2</td>
<td>15/02/2016</td>
<td>CTTC</td>
<td>Inclusion of the simulation framework description</td>
</tr>
<tr>
<td>1.3</td>
<td>30/07/2016</td>
<td>TASE</td>
<td>Inclusion of the functional description for the HNM and IBN</td>
</tr>
<tr>
<td>1.4</td>
<td>05/12/2016</td>
<td>CTTC</td>
<td>Final Contributions related to routing and energy saving protocols functional description.</td>
</tr>
<tr>
<td>1.5</td>
<td>01/12/2016</td>
<td>AVA</td>
<td>Final Contributions related to traffic management techniques.</td>
</tr>
<tr>
<td>1.6</td>
<td>19/01/2017</td>
<td>OTE/AVA</td>
<td>Final Contributions related to Network Management and version released for QA.</td>
</tr>
<tr>
<td>1.7</td>
<td>31/01/2017</td>
<td>TASE</td>
<td>Final Version Released.</td>
</tr>
<tr>
<td>1.8</td>
<td>13/04/2017</td>
<td>CTTC</td>
<td>Routing and Energy updates according to EC reviewers.</td>
</tr>
<tr>
<td>1.9</td>
<td>26/04/2017</td>
<td>TASE</td>
<td>Modifications by EC Reviewers adopted.</td>
</tr>
<tr>
<td>1.10</td>
<td>28/04/2017</td>
<td>CTTC</td>
<td>Final edition</td>
</tr>
</tbody>
</table>

Contributors

<table>
<thead>
<tr>
<th>Name</th>
<th>Company</th>
<th>Contributions included</th>
</tr>
</thead>
<tbody>
<tr>
<td>Willie Jose Betancourt Simoza</td>
<td>TASE</td>
<td>Chapter 1, 2, 3 and 8.</td>
</tr>
<tr>
<td>Georgios Ziaragkas</td>
<td>AVA</td>
<td>Chapter 2, 5, 6 and Quality Assurance</td>
</tr>
<tr>
<td>José Núñez, Jorge Baranda, Paolo Dini, Marco Miozzo</td>
<td>CTTC</td>
<td>Chapter 1,2,3,4,5, and Appendix.</td>
</tr>
<tr>
<td>Georgios Agapiou</td>
<td>OTE</td>
<td>Chapter 5 and 7 and Quality assurance</td>
</tr>
<tr>
<td>Diana Mathew</td>
<td>AVA</td>
<td>Quality assurance</td>
</tr>
<tr>
<td>José Núñez, Xavier Artiga, Ana Pérez-Neira</td>
<td>CTTC</td>
<td>Quality assurance</td>
</tr>
</tbody>
</table>
# Table of Contents

List of Figures .................................................................................................................. 6  
List of Tables ..................................................................................................................... 7  
List of Acronyms ............................................................................................................... 8  
Executive Summary .......................................................................................................... 12  
1. Introduction .................................................................................................................. 14  
2. SANSA network functional specification ...................................................................... 16  
   2.1. System Architecture ................................................................................................. 16  
   2.2. HNM Functional Specification ................................................................................ 18  
      2.2.1. Topology Management Module ......................................................................... 20  
      2.2.2. Radio Resources Module .................................................................................. 27  
      2.2.3. Routing Management Module ......................................................................... 28  
      2.2.4. Quality of Service Module .............................................................................. 29  
      2.2.5. Energy Saving Module ...................................................................................... 29  
2.3. IBN Functional Specification .................................................................................... 30  
   2.3.1. Initial Routine (Auto-provisioning) Module ...................................................... 31  
   2.3.2. AAA Operations Module .................................................................................... 31  
   2.3.3. Topology Management Agent ........................................................................... 32  
   2.3.4. Radio Resources Agent ...................................................................................... 32  
   2.3.5. Routing Management Agent ............................................................................ 32  
   2.3.6. Quality of Service Agent ................................................................................... 40  
   2.3.7. Energy Saving agent ............................................................................................ 44  
3. HNM – IBN Functional Interoperability Specification .................................................. 54  
   3.1. Link Switch-Off Scenario ....................................................................................... 54  
   3.2. Terrestrial Beam Activation Scenario .................................................................... 56  
   3.3. Traffic Off-loading Scenario .................................................................................. 57  
   3.4. Resiliency Scenario ............................................................................................... 58
3.5. Terrestrial beam interference Scenario ................................................................. 59

4. Functional Implementation Roadmap ........................................................................ 60
   4.1. Network Simulation Framework ......................................................................... 64

5. Conclusions .............................................................................................................. 71

References .................................................................................................................... 73

Appendix ....................................................................................................................... 78
List of Figures

Figure 2.1. SANSA Environment ........................................................................................................ 17
Figure 2.2. SANSA Elements Module Definition ............................................................................... 18
Figure 2.3. Graphical representation of a SANSA topology ............................................................... 24
Figure 2.4. Hybrid Terrestrial - Satellite backhaul Network under conditions ............................... 35
Figure 2.5. Throughput evolution of different flow attained in the hybrid wireless backhaul .......... 36
Figure 2.6. Normalized throughput evolution with the variation of satellite-terrestrial links connected to the EPC ......................................................................................... 37
Figure 2.7. High-level diagram specifying the two-stage routing algorithm for SANSA. ............... 39
Figure 2.8. Flowchart diagram ........................................................................................................ 43
Figure 2.9. Example of traffic demand and amount of energy harvested ........................................ 49
Figure 2.10. Percentage of throughput gain of QL respect to greedy scheme. ............................... 50
Figure 2.11. Traffic drop rate for QL and greedy. ............................................................................ 51
Figure 2.12. Example of temporal behavior for a network of 3 SCs. ............................................... 52
Figure 2.13. Percentage of energy efficiency improvement of QL respect to greedy. .................. 53
Figure 3.1. Link Switch Off Message Exchange .............................................................................. 55
Figure 3.2 Beam Activation Messages Exchange ............................................................................ 56
Figure 3.3 Traffic Off-loading Scenario ............................................................................................. 57
Figure 3.4 Resilience Scenario .......................................................................................................... 58
Figure 3.5 Interference Scenario ....................................................................................................... 59
Figure 4.1. Ns-3 network simulator stack .......................................................................................... 64
Figure 4.2. Lena Network Topology .................................................................................................. 65
Figure 4.3. Example of hybrid terrestrial satellite-backhaul network proposed by SANSA project . 66
Figure 4.4. LENA extensions to support Hybrid Satellite-Terrestrial mesh backhaul networks .... 67
Figure 4.5. Wireshark capture of data plane traffic traversing a wireless mesh .............................. 68
Figure 4.6. Example of a Hybrid Backhaul Network ....................................................................... 68
Figure 0.1. Main working principle of backpressure routing ......................................................... 88
Figure 0.2. Diagram showing the configuration layout of the SANSA management system .......... 94
Figure 0.3. Configuration management layout .................................................................................. 94
Figure 0.4. Management of the functionality of the elements ........................................................... 94
List of Tables

Table 2-1. Description of objects in a SANSA topology .................................................. 24
Table 2-2. Power model parameter for different types of BS ........................................... 46
Table 4-1. Functional Roadmap ....................................................................................... 60
Table 4-2 Example of topological matrix configuring Figure 4.6 ....................................... 69
Table 4-3. Example of vectors indicating connection with EPC and Satellite nodes of Figure 4.6 ... 70
Table 0-1. Taxonomy of Routing protocols suitable for backhaul networks ....................... 78
Table 0-2. Overview of Available Implementation of route-based routing algorithms ......... 85
# List of Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACM</td>
<td>Adaptive Coded Modulation</td>
</tr>
<tr>
<td>AODV</td>
<td>Ad hoc On-Demand Distance Vector</td>
</tr>
<tr>
<td>API</td>
<td>Application Program Interface</td>
</tr>
<tr>
<td>ARPU</td>
<td>Average revenue per user</td>
</tr>
<tr>
<td>B.A.T.M.A.N</td>
<td>Better Approach To Mobile Adhoc Networking</td>
</tr>
<tr>
<td>BCP</td>
<td>Backpressure Collection Protocol</td>
</tr>
<tr>
<td>CAPEX</td>
<td>Capital Expenditures</td>
</tr>
<tr>
<td>DAG</td>
<td>Directed acyclic graph</td>
</tr>
<tr>
<td>DL</td>
<td>Downlink</td>
</tr>
<tr>
<td>DSR</td>
<td>Dynamic Source Routing</td>
</tr>
<tr>
<td>EH</td>
<td>Energy Harvesting</td>
</tr>
<tr>
<td>EPC</td>
<td>Evolved Packet Core</td>
</tr>
<tr>
<td>ESF</td>
<td>Extended Super Frame</td>
</tr>
<tr>
<td>EsNo</td>
<td>Energy per symbol to Noise power spectral density</td>
</tr>
<tr>
<td>ETX</td>
<td>Expected Transmission Count</td>
</tr>
<tr>
<td>ExOR</td>
<td>Extremely Opportunistic Routing</td>
</tr>
<tr>
<td>GTP-U</td>
<td>GPRS Tunneling Protocol User Plane</td>
</tr>
<tr>
<td>GPSR</td>
<td>Greedy Perimeter Stateless Routing</td>
</tr>
<tr>
<td>HMI</td>
<td>Human Machine Interface</td>
</tr>
<tr>
<td>Abbreviation</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>-------------</td>
</tr>
<tr>
<td>HNM</td>
<td>Hybrid Network Manager</td>
</tr>
<tr>
<td>HWMP</td>
<td>Hybrid Wireless Mesh Protocol</td>
</tr>
<tr>
<td>HeNB</td>
<td>Home eNodeB</td>
</tr>
<tr>
<td>IBN</td>
<td>Intelligent Backhaul Node</td>
</tr>
<tr>
<td>IETF</td>
<td>Internet Engineering Task Force</td>
</tr>
<tr>
<td>LIFO</td>
<td>Last-In First-Out</td>
</tr>
<tr>
<td>LTE</td>
<td>Long Term Evolution</td>
</tr>
<tr>
<td>LQSR</td>
<td>Link Quality Source Routing</td>
</tr>
<tr>
<td>MCS</td>
<td>Mobile Cloud Service</td>
</tr>
<tr>
<td>MNL</td>
<td>Mobile Network Layer</td>
</tr>
<tr>
<td>MNO</td>
<td>Mobile Network Operator</td>
</tr>
<tr>
<td>ModCod</td>
<td>Modulation and Coding</td>
</tr>
<tr>
<td>MP</td>
<td>Mesh Point</td>
</tr>
<tr>
<td>MP-TCP</td>
<td>MultiPath TCP</td>
</tr>
<tr>
<td>MTU</td>
<td>Maximum Transmission Unit</td>
</tr>
<tr>
<td>OFDMA</td>
<td>Orthogonal Frequency Division Multiple Access</td>
</tr>
<tr>
<td>OSS</td>
<td>Operations Systems Support</td>
</tr>
<tr>
<td>PtP</td>
<td>Point to Point</td>
</tr>
<tr>
<td>PSBOL</td>
<td>Path Selection based on Object Length</td>
</tr>
<tr>
<td>QoE</td>
<td>Quality of Experience</td>
</tr>
<tr>
<td>QL</td>
<td>Queue Learning</td>
</tr>
<tr>
<td>RAN</td>
<td>Radio Access Network</td>
</tr>
<tr>
<td>RAT</td>
<td>Radio Access Technology</td>
</tr>
<tr>
<td>Acronym</td>
<td>Definition</td>
</tr>
<tr>
<td>---------</td>
<td>------------</td>
</tr>
<tr>
<td>RB</td>
<td>Radio Base</td>
</tr>
<tr>
<td>RES</td>
<td>Renewable Energies Sources</td>
</tr>
<tr>
<td>RERR</td>
<td>Route Error</td>
</tr>
<tr>
<td>RL</td>
<td>Reinforcement Learning</td>
</tr>
<tr>
<td>RREP</td>
<td>Route Reply</td>
</tr>
<tr>
<td>ROMER</td>
<td>Resilient Opportunistic Mesh Routing</td>
</tr>
<tr>
<td>RREQ</td>
<td>Route Request</td>
</tr>
<tr>
<td>RRM</td>
<td>Radio Resource Management</td>
</tr>
<tr>
<td>RTO</td>
<td>Retransmission Time Out</td>
</tr>
<tr>
<td>RTT</td>
<td>Round Trip Time</td>
</tr>
<tr>
<td>SoA</td>
<td>State of the Art</td>
</tr>
<tr>
<td>SC</td>
<td>Small Cells</td>
</tr>
<tr>
<td>S-GW</td>
<td>Serving Gateway</td>
</tr>
<tr>
<td>SNMP</td>
<td>Simple Network Management Protocol</td>
</tr>
<tr>
<td>SON</td>
<td>Self-Organizing Network</td>
</tr>
<tr>
<td>ST</td>
<td>Satellite Terminal</td>
</tr>
<tr>
<td>TC</td>
<td>Traffic Classification</td>
</tr>
<tr>
<td>TNL</td>
<td>Transport Network Layer</td>
</tr>
<tr>
<td>TORA</td>
<td>Temporally-Ordered Routing algorithm</td>
</tr>
<tr>
<td>TT</td>
<td>Terrestrial Terminal</td>
</tr>
<tr>
<td>TTI</td>
<td>Transmission Time Interval</td>
</tr>
<tr>
<td>WCETT</td>
<td>Weighted Cumulative Expected Transmission Time</td>
</tr>
<tr>
<td>UE</td>
<td>User Equipment</td>
</tr>
<tr>
<td>UL</td>
<td>Uplink</td>
</tr>
<tr>
<td>-----</td>
<td>----------------------</td>
</tr>
<tr>
<td>VM</td>
<td>Virtual Machine</td>
</tr>
<tr>
<td>VSAT</td>
<td>Very Small Aperture Terminal</td>
</tr>
<tr>
<td>WMN</td>
<td>Wireless mesh network</td>
</tr>
<tr>
<td>WSN</td>
<td>Wireless Sensor Networks</td>
</tr>
<tr>
<td>ZRP</td>
<td>Zone Routing Protocol</td>
</tr>
</tbody>
</table>
Executive Summary

This is deliverable D4.2 of the Shared Access Terrestrial-Satellite Backhaul Network enabled by Smart Antennas (SANSA) project (grant agreement no 645047) which documents the outcome of Task 4.1. The objective of this deliverable is to describe the specification of the different building blocks integrated in the SANSA network elements, which are the Hybrid Network Manager (HNM) and the Intelligent Backhaul Node (IBN). Therefore, we specified the main functionalities that should be covered and split between both elements for the proper management of the dynamic and hybrid SANSA network. In particular, we described the Topology management function which provides the capability of adapting the network topology to the traffic demands through the activation/deactivation and reconfiguration of the backhaul links; the Radio Resource management function which is in charge of coordinating the efficient use of all spectrum resources available; the Routing function which, given a topology, decides the most efficient way to propagate the traffic flows through the network; the Quality of Service (QoS) function, in which we included traffic classification and load balancing between terrestrial and satellite segments, in order to exploit the offloading through the satellite; and the Energy saving function that combines the use of energy harvesters in small cells with the capability of switching-off access and backhaul interfaces for improving the network energy efficiency. The topology and radio resource changes are processed and decided by the HNM, which have a global vision of the network. In these cases, the responsibilities of the IBNs are to apply the configuration changes decided by the HNM and to gather information and alerts about the status of the backhaul links and send them to the HNM. In contrast, the routing, QoS and energy efficiency processing run in a distributed fashion in the IBNs following the self-organization principles. In these cases, the HNM just decides global configuration parameters and receives alerts and feedback about the link status.

For the cases of the topology management, the routing, the QoS and the energy saving functions, the deliverable not only describes the functional specification of this modules but provides a preliminary description of the algorithms to be implemented. Moreover, in the case of routing and energy efficiency, preliminary simulation results supporting the selected algorithms are shown.

In addition, the deliverable analyses the interaction of the HNM, the IBN and the functionalities therein in a set of interoperability scenarios that are unique in the SANSA project. These scenarios allow demonstrating the capabilities and the logic behind the modules and subsystems already defined and mapped to the HNM and IBN devices.

Finally, the deliverable shows the roadmap that the SANSA project is following, from the Design stage of the aforementioned functionalities, through the simulation stage, to the demonstration
stage. Remarkably, here it is included the description of the simulation framework used for evaluating the developed techniques.
1. Introduction

The aim of SANSA project is to improve capacity, resilience, and coverage of mobile backhaul networks. Current and future traffic demands require not only the appropriate access infrastructure but also the corresponding backhaul infrastructure for not becoming the network bottleneck. This can be achieved thanks to the self-reconfigurable hybrid terrestrial-satellite backhaul transport network proposed within the SANSA project. This solution is based on the network architecture derived in [61], which relies on a centralized management entity called Hybrid Network Manager (HNM) and distributed network elements extending current backhaul nodes called Intelligent Backhaul Nodes (IBNs).

In general, we can assume that it is unlikely that fiber reaches every IBN in the scenarios defined within the SANSA project. High installation costs and uncertain returns in the context of declining average revenue per user (ARPU) reduce fiber availability in dense network deployments in urban areas and in scarce network deployments in low and high populated areas. The deployment of terrestrial wireless multi-hop backhauls along with additional satellite resources becomes an alternative to achieve the objective previously stated. Moreover, as explained in Deliverable 2.2 [60] current and future wireless technologies are a feasible solution to create such kind of networks.

Under such a context and considering that each IBN with its associated transport devices and the satellite are resources of the hybrid transport network, the problem, from a high-level perspective, is how the Mobile Network Operator (MNO) can make the most out of the transport network resources deployed.

In this case, SANSA project conceives the answer to this question by including self-organizing network capabilities (SON) at the transport network layer (TNL). The TNL is in charge of carrying control and data plane traffic to/from the core network (e.g., EPC) to the SCs. From 3GPP standard release 12, self-organization capabilities are envisaged for mobile networks. However, they merely refer to Radio Access Technology (RAT) procedures and not to the intrinsic procedures conducted at the TNL. Additionally, the introduction of SON capabilities at the wireless multi-hop TNL could help reducing OPEX/CAPEX expenditures because optimizing the operation of the TNL would also reduce the human intervention, which is also in line with the MNOs’ objectives in the context of declining ARPU.

The concept of self-organization has multiple dimensions, comprising of self-configuration, self-optimization, and self-healing methods. The self-configuration method is triggered by incidental events of an intentional nature (e.g., when an IBN joins or leaves the network due to energy efficiency mechanisms). Such incidental events of intentional nature are triggered partly by the HNM. Second, self-optimization methods should include a set of procedures to exploit wireless
transport network resources (i.e., IBNs) as much as possible in order to obtain improved network performance or to keep the wireless transport network stable under sudden traffic changes by, for instance, offloading traffic to the satellite connection when possible. Third, self-healing capabilities refer to undertake incidental events of a non-intentional nature, like when there is an unexpected node/link failure in the wireless TNL.

The main goal of this deliverable is to describe the functional specification of the HNM and the IBN which follow the self-organizing principles. The main focus is on the topology management, routing, traffic classification and energy saving functionalities for which the extended specifications include a first elaboration of the algorithms to be developed. In the case of routing and energy saving, these algorithms are supported by preliminary simulation results. In addition, several hybrid satellite-terrestrial backhaul scenarios including situations such as backhaul link failures and terrestrial interference conditions are described. At this point, the main focus is to define how the specified functionalities are capable of interacting amongst each other in order to reach the desired key performance indicators. Finally, the roadmap followed by the SANSA project form the conception of this functionalities to its implementation and demonstration is presented. It includes the description of the simulation framework that will be used to evaluate the network performance.

The deliverable contains most of the work covered in Task 4.1 and is organized as follows:

- Chapter 2 describes the HNM and IBN functional specification including the interoperability scenarios.
- Chapter 3 present some representative HNM-IBN interoperability scenarios.
- Chapter 4 provides the implementation roadmaps for the functionalities defined in Chapter 2.
- Chapter 5 summarizes the main outcomes in the conclusions section.
- The annex describes the analysis of the State-of-the-Art supporting the functionality definition defined in Chapter 2.
2. SANSA network functional specification

In this chapter we first recall the SANSA network architecture and then derive the functional specifications of the two SANSA elements that enable the hybrid network management, namely the Hybrid Network Manager (HNM) and the Intelligent Backhaul Node (IBN).

2.1. System Architecture

The SANSA network architecture derived in [61] is depicted in Figure 2.1. Its main purpose is to enable the dynamic management and self-organization of the hybrid terrestrial-satellite network, allowing an efficient use of all network resource in any traffic conditions. Such a management scheme includes the capability of activating and deactivating terrestrial and satellite links as well as re-pointing terrestrial links, thus adapting the network topology to the traffic demands. In this sense, Figure 2.1 shows the links established in a given instant and the alternative links that can be used for reconfiguring the network. Therefore, the new network elements proposed by SANSA, i.e. the HNM and the IBN aim to provide an overall control over local elements at each backhaul node: Terrestrial Modems (TM), Satellite Modems (SM). Here we consider that the terrestrial modems include also the smart antennas, which are the key component enabling the terrestrial topology reconfiguration.

In short, the IBNs collect all event information (e.g. link status) from the terrestrial and satellite modems. Then the IBNs are able to execute the configuration changes necessary over these elements to make the most efficient use of all network resources. The decision about the configuration changes to be executed can be taken in a distributed fashion in the IBNs themselves, or in a centralized fashion in the HNM.

Therefore, two kinds of traffic are propagated though the network, Control and Data Traffic. The Control Traffic acts at two different levels: first for interactions between the hybrid network manager and the intelligent Backhauling Nodes; and at a second level between the IBN and the satellite and terrestrial modems present in each node. The Data traffic is the real traffic going from User Equipment (UEs) to the Evolved Packet Core (EPC) to reach, for instance, a remote server in the Internet, or other UE connected to the SANSA backhaul. In particular, the trajectory will be established as a function of resource availability (links, bandwidth, etc.). In SANSA we consider an in-band signaling scheme, meaning that the Control Traffic share the same links established for Data traffic, for an efficient use of resources.
In order to provide a flexible system architecture adapted to the functionality introduced here an elaborated in following sections, an approach based on modules and sub-systems has been conceived [62] to complete the system configuration. A high level view of the modular architecture adopted for the HNM and IBN is presented in the Figure 2.2. It is worth remarking here that the HNM and the IBN are designed as pure software elements, with the ability to run on top of commodity hardware or even, the modules composing the HNM and IBN could be hosted by virtual nodes deployed in a flexible virtualized network infrastructure. As for the deployment model used for the HNM and the IBN modules, note that the study of the deployment model used is out of the scope of SANSA. From herein, we will assume a single centralized HNM entity and a several IBN distributed in all the nodes composing the hybrid satellite-terrestrial backhaul. In what follows, we will focus on the functional specification of both the HNM and the IBN.
2.2. **HNM Functional Specification**

The Hybrid Network Manager has been designed to present three management blocks [61][62]:

- **Commissioning Management**: This function manages the initial routine subsystem responsible to support the auto-provisioning process for the IBNs. Additionally, Authentication, Authorization and Accounting (AAA) tasks are covered by this functionality.

- **Configuration Management**: This function is responsible for managing the end to end reconfiguration process and for generating the configuration updates to be loaded at the IBN’s local elements.

- **Event Management**: This function receives and collects the information generated at the IBN level. These events, which have been previously classified and processed by the IBN’s core server, offer a common data model. Once these events reach the event manager module at the HNM, they are processed and locally allocated to start the threshold evaluation process separately in each sub modules.

As presented in Figure 2.2, five main functionalities required for the proper operation of the hybrid network are mapped to five management modules across the Configuration and Events management blocks. Specifically, the Topology management, the Radio resource management, the Routing management, the Quality of Service (QoS) management and the Energy saving
management. The functional specifications of these modules are elaborated in next sections, but the summary of the identified high level HNM functional specifications is shortlisted here:

- The HNM will embed the computational power to run the algorithms responsible to accomplish the configuration management and events processing necessities over the network.
- The HNM will be able to coordinate and populate the configuration changes across different HNM functionalities.
- The HNM will calculate alternative topologies to overcome the shortcomings of the current one (e.g. link failures or link congestion).
- The HNM will provide network topology calculation results to IBNs so they can manage the possible topological changes over Terrestrial and Satellite Terminals.
- The HNM will interact with a spectrum analysis module (defined as Radio environment mapping in [61]) in order to optimize the use of spectrum resources.
- The HNM will be able to determine the best routing algorithm for the IBNs involved in the hybrid SANSA network and will provide high-level network rules to the IBNs so they can route traffic properly.
- The HNM will inform the IBN if any specific traffic may be offloaded through the satellite segment. Indeed, the HNM can manage QoS policies on a centralized manner and will provision the traffic classification procedures to determine whether traffic is backhauled over the terrestrial or the satellite network.
- The HNM can interact with the energy management procedures running at the IBNs in order to arbitrate the decisions taken by the energy agent running in the IBNs. In this way, the energy management procedures aim to maximize the energy savings while the HNM assures that the decisions locally taken at the IBN maintain a hybrid satellite-terrestrial network able to satisfy the traffic demands.
- The HNM can provide an Application Program Interface (API) to the SANSA operator referred to as Human Machine Interface (HMI) so that the SANSA operator can specify its intended high-level policy recommendations.
- The HNM will be able to provide the graphical user interface to the SANSA operator to determine the status of the whole network and configuration capabilities over the elements. In this way, the HNM could also provide recommendations to the SANSA operator and so potentially maintaining a continuous interaction with the SANSA operator.
2.2.1. Topology Management Module

In contrast to current wireless backhauls, SANSA is conceived as a dynamic hybrid network capable of adapting its topology and operation to the traffic conditions. Therefore, a SANSA network should be able to activate and deactivate terrestrial and satellite links, as well as to reconfigure terrestrial links. By reconfiguration we mean here the capability to change the direction to which one antenna on a backhaul node is pointing, so deactivating one link and activating a new one.

The Topology Management module is thus in charge of providing the best possible topology in order to meet the project objectives in terms of improved capacity, resilience to link failures and congestion, and energy efficiency. This is accomplished, for example, by: activating a satellite link and reconfiguring several terrestrial links to reinforce a certain part of the network in order to increase its capacity region, resulting in an increased net capacity and resilience to congestion; activating a backup path (being it terrestrial or satellite) to overcome a link failure; or creating an alternative path for a link switched–off by the energy saving module.

In general, SANSA follows the self-organization principles, so the HNM collects information and alerts from the IBNs regarding to failures, congestion and suggested energy saving procedures and processes them centrally to assure the end-to-end connectivity at network level. However, the changes in the topology can be also forced by an Operator through the Human Machine Interface (HMI) of the HNM. As detailed in next section, the topology processing algorithm based its decisions on KPIs such as the number of changes required, delay and power consumption. In addition, it is assisted by the Radio resource management module in order to assure that the selected topologies are feasible and can be implemented with acceptable levels of interference.

Summarizing, through the Topology Management module the HNM will have the capability of:

- Receiving and processing the status and alerts information collected by the IBN’s regarding to current traffic conditions and generated statistics.
- Processing topology changes forced by an Operator though the HMI.
- Calculating improved topologies taking under consideration KPI’s such as the number of changes with respect to the current topology, delay and power consumption estimations to determine the best topology configuration to be applied.
- Sending the topology configuration updates to the IBNs, so that they can proceed with the link reconfigurations at IBN level.
2.2.1.1. Topology calculation algorithm

The topology calculation algorithm of the Topology Management module in the HNM will have two phases. The initialization phase in which the whole network is modelled, and the calculation phase in which the new topology is calculated.

For the initialization phase, the following SANSA elements must be identified:

- EPC with terrestrial connections and with/without Satellite connection.
- Satellite network.
- IBN with/without Satellite links and with/without Radio Access Network (RAN) including the evolved Node B (eNodeB) stack.

For each of these elements, the next parameters will be necessary for the model:

- Name: The element name.
- Position: The element position (longitude and latitude in degrees).
- Number of Terrestrial Terminals: The number of terrestrial terminals (modems) connected to this element.
- List of terrestrial transmission capabilities (i.e. links):
  - List of Tx Fixed Link Capabilities.
  - List of Tx Dynamic Beam Capabilities.
- Number of Satellite Terminals: The number of satellite terminals (modems) connected to this element.
- List of satellite transmission capabilities (i.e. links):
  - List of Tx Satellite Link Capabilities.

The number of terrestrial/satellite terminals are the maximum number of transmission/reception links with other elements. Eg. <NumberOfTerrestrialTerminals> sets to 2 states a maximum of 2 TxLinks and 2 RxLinks. In turn, each list of transmission capabilities groups transmission capability elements (TxLinkCapability) composed of the following parameters:

- ToNode: The element name to transmit data. Inside a list of transmission capabilities this name shall be unique. It is no possible to have more than one link of the same type - FixedLink, DynamicBeam or SatelliteLink- with the same element node.
- Enabled: If the current link is enabled or not, due to some problems (e.g. link failure) or due to some events (e.g. user event, energy saving).
- Status:
Mandatory_on: If the link is imperatively powered-on. Therefore, this link shall be powered-on on the final topology.

On: If the link is initially powered-on and shall be used on the topology algorithm. Therefore, this link may or may not be on the final topology.

Off: If the link is initially powered-off and shall be used on the topology algorithm. Therefore, this link may or may not be on the final topology.

Unknown: If the link status is not known.

It is worth remarking here that any terrestrial transmission link shall starts and ends in a Terrestrial terminal, and that any satellite transmission link shall starts and ends in a Satellite terminal.

Specifically, the SANSA elements are modelled as follows:

1. **EPC**

   It shall be one and no more than one EPC element (EPC or SatelliteEPC).

   EPC: The EPC transmission capabilities will be:
   - Fixed link (List Of Tx Fixed Link Capabilities).

   Satellite EPC: The Satellite EPC transmission capabilities will be:
   - Fixed link (List Of Tx Fixed Link Capabilities).
   - Satellite link (List Of Tx Satellite Link Capabilities).

   The EPC element shall have at least, one enabled link power-on.

2. **Satellite**

   The Satellite element is a transparent satellite (no regenerative), so only are available the following transmissions:
   - Satellite EPC => Satellite => Satellite IBN.
   - Satellite IBN => Satellite => Satellite EPC.

3. **IBN**

   It shall be at least one IBN element (IBN or SatelliteIBN).

   IBN: with Terrestrial Terminal. Transmission capabilities will be represented by:
   - Fixed link (List Of Tx Fixed Link Capabilities).
   - Dynamic beam (List Of Tx Dynamic Beam Capabilities).
Satellite IBN: with Terrestrial Terminal and Satellite Terminal. Transmission capabilities will be:

- Fixed link (List Of Tx Fixed Link Capabilities).
- Dynamic beam (List Of Tx Dynamic Beam Capabilities).
- Satellite link (List Of Tx Satellite Link Capabilities).

Once the elements identification has been completed at HNM Topology Management module, the initial network topology is provided as an initial input for the algorithm, identifying:

- Physical restrictions (where it is not feasible to establish a link).
- Fixed links.
- Potential (satellite or terrestrial) dynamic links.
- EPC connections.

Next diagram and Table show an example of a graphic representation of the whole SANSA topology:


Figure 2.3. Graphical representation of a SANSA topology

Table 2-1. Description of objects in a SANSA topology

<table>
<thead>
<tr>
<th>Objects</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Grey cloud: EPC</td>
<td>[TT x] states the number of Terrestrial Terminals</td>
</tr>
<tr>
<td></td>
<td>[ST x] states the number of Satellite Terminals</td>
</tr>
</tbody>
</table>
### Link statuses

- **Green lines:** link power-on
- **Black lines:** potential link (currently power off)
- **Red lines:** link not enabled

### Link types

- **Solid line:** fixed link
- **Dashed line:** Dynamic beam
- **Dotted line:** Satellite link

---

**Orange square:** IBN with terrestrial links  
[TT x] states the number of Terrestrial Terminals

**Blue octagon:** IBN with terrestrial and satellite links  
[TT x] states the number of Terrestrial Terminals  
[ST x] states the number of Satellite Terminals
Once the network modelling is complete, the HNM Topology Management module performs the following steps in order to finish the initialization phase:

- Compile the input file that describes the SANSA network
- Creates the baseline topology. This topology is composed of:
  - All the links that are not enable (attribute “Enabled” set to “false”).
  - All the enabled links (attribute “Enabled” set to “true”) that shall be mandatory power-on (attribute “Status” set to “mandatory_on”).
These links shall be present on the final topology.
- Computes all the possible optional TxLinks per each node.

Each IBN has a configured number of Terrestrial Terminals that constrains the number of possible Tx/Rx links. Therefore, given a IBN with 2 Terrestrial Terminal, possible Tx/Rx links are:

- A <= X => B  2 Tx + 0 Rx
- A <= X <=> B  2 Tx + 1 Rx
- A <= X <= B  2 Tx + 2 Rx
- A => X <= B  1 Tx + 2 Rx
- A => X <= B  0 Tx + 2 Rx

It is not possible to have more than one Tx/Rx link with other IBN

In the calculation phase, the HNM Topology Management module will determine a candidate topology using the following procedure:

- Computes all the possible node combinations:
  - Node #1 combination #1 + Node #2 combination #1 + ... + Node #n combination #1
  - Node #1 combination #1 + Node #2 combination #1 + ... + Node #n combination #2
  - ...
  - Node #1 combination #x + Node #2 combination #y + ... + Node #n combination #z
- Per each node combination, creates a candidate topology based on the baseline topology (i.e. required links + optional links combination #x) and checks:
  - If there are IBNs without TxLinks (data is never transmitted), the topology is discarded.
  - If there are IBNs without RxLinks (data is never received), the topology is discarded.
D4.2 Interoperability of terrestrial and satellite links high level functional specification

If there are IBNs that are not reachable from the EPC (data is not received in the EPC), the topology is discarded.
If there are IBNs that do not reach the EPC (data is not transmitted to the EPC), the topology is discarded.
If there are nodes having a satellite link, they cannot connect to other “disconnected” nodes (i.e. not connected to the topology by other means).

• Scores the Candidate topologies calculating a set of KPIs on each of them. The number and weight of KPIs is configurable. KPIs currently developed are:
  o Similarity: This KPI compare the candidate topology with the original one. The more similar the better.
  o Total Power Consumed: This KPI count the number of Terrestrial/Satellite Terminals power-on and multiply them by a configurable coefficient based on the power consumed in each Terminal.
    ▪ Terrestrial Terminal: 50W
    ▪ Satellite Terminal 100W

Other KPIs that are being used are:
  o Minimum Number of Hops from the EPC: This KPI aggregates the minimum number of hops required from the EPC to reach each IBN.
  o Minimum Number of Hops to the EPC: This KPI aggregates the minimum number of hops required from each IBN to reach the EPC.
  o Minimum Delay from the EPC: This KPI aggregates the minimum delay to transmit a packet from the EPC to each IBN.
  o Minimum Delay to the EPC: This KPI aggregates the minimum delay to transmit a packet from each IBN to the EPC.

The top scored candidate topologies are then provided to the Radio Resources module which check their feasibility with regard to the interference levels and carries available. Finally, the Topology Management module selects the best topology according to this analysis.

Once the best topology has been selected, configuration updates are provided to the IBNs in order to locally the topology changes.

2.2.2 Radio Resources Module
In order to meet the project objective relative to a 10x network spectral efficiency increase, SANSA foresees an aggressive frequency reuse between terrestrial links but also between terrestrial and satellite links. In addition, the dynamicity of the SANSA networks not only apply to their topology
but also to their carrier allocation, or use of the spectrum, since any new topology would create
new levels of interference and may require different carrier allocation or other interference
mitigation techniques. Therefore, the HNM have to coordinate the use of available spectrum
resources through the Radio resources module. In this task, the HNM is closely assisted by an
external tool capable of performing interference analysis, of running carrier allocation algorithms
and of considering other interference mitigation techniques. This external module have been
defined as the Radio Environment Mapping (REM) in [61] and may contain the algorithms and
techniques developed within the WP3 of SANSA.

In practice, the Radio Resource module will closely interact with the Topology Management
module, so whenever a topology is selected the corresponding carrier allocation is also provided.
Then this carrier allocation is distributed to the IBNs, which implement the required changes.

In summary, the Radio Resource module of the HNM must be able to:

- Optimize the use of spectrum resources across the terrestrial and satellite segments with
  the assistance of the REM module.
- Closely interact with the Topology Management module in order to determine the carrier
  allocation for the best topology.
- Distribute the selected carrier allocation to all IBNs in the network.

2.2.3. Routing Management Module

Once a network topology is fixed, the routing function is in charge of deciding how to propagate
the traffic across the network in order to maximize throughput and minimize the latency of delay
sensitive flows. According to the self-organizing principles and the functional specifications derived
in Section 2.3.5, the routing calculations will be locally and autonomously conducted in a
distributed manner at the IBNs.

However, the Routing Management module of the HNM behaves here as the central controller. In
particular, this module is in charge of deciding and propagating the configuration parameters of
the routing protocol to the IBNs, or even in charge of deciding the type of routing protocol to be
used. In addition, it will collect network status information and alerts from the IBNs related to the
routing function, and propagate them to the topology Manager and/or to a SANSA Operator
though the Human Machine Interface (HMI). Therefore, the functionalities of the Routing
Management module of the HNM can be summarized as:
The HNM will be responsible for allocating the configuration information related to the routing protocol scheme defined for the IBNs.

The HNM will have a catalogue of the routing protocols supported by the network elements in order to apply the most convenient routing solution.

The HNM will be responsible to send the routing protocol election and configuration to the IBNs.

The HNM will be responsible to process the events and network status information statistics to generate events and alerts at HMI level.

2.2.4. Quality of Service Module
In order to make the most efficient use of all network resources, and especially, to properly exploit the offloading capabilities of the satellite segment, it is vital to define and apply Quality of Service policies. Indeed, not all services or type of traffics can be routed through the satellite because of its long round trip time which directly affects the communication latency. Therefore, SANSA foresees to apply traffic classification techniques with a service operation orientation. As in the case of the routing module, these techniques will be enforced in the IBNs but the Quality of Service module of the HNM will act as a central manager selecting and distributing to the IBNs the selected QoS policies.

Then, the Quality of Service module of the HNM must be able to:

- Allocate the methods to optimize the traffic exchange performance using the Quality of Services policies. Depending on the functionalities that allow to choose between Integrated Services (IntServ) and Differentiated Services (DiffServ) based on the capabilities of the network elements, different ways to apply packet marks could be implemented.

- Centrally manage the QoS module and the IBN will apply the QoS policies selected.

- Process the events and statistics information collected by the IBNs regarding to present status information to the operator across the HMI.

2.2.5. Energy Saving Module
In order to deal with the project objective relative to a 30% improvement of the energy efficiency SANSA foresees to combine the two solutions detailed in Section 2.3.7. On one hand, the possibility of switching-off access and backhaul interfaces during periods with low traffic demands. On the other hand, the use of small cells with energy harvesters such as solar panels. For the proper exploitation of this combination an intelligent learning algorithm will be developed. Like in previous cases, and according to self-organizing principles, this algorithm will run locally at the
IBNs. However, a centralized Energy Saving module at the HNM is still required in order to mediate and coordinate the autonomous decisions taken locally at the IBNs in terms of energy savings (i.e., switching on and off IBN links and/or IBN RAN components).

Such coordination will imply the interaction with other HNM modules such as the Topology Management or the Radio resources module. It must be noted here that the energy saving function running at the IBNs do not have a vision over the whole network. Thus, when the IBN determines a backhaul terrestrial interface to be switched-off, it is needed to invoke the Topology Management module in order to assure that the necessary end-to-end connectivity is maintained or, otherwise, in order to calculate an alternative topology.

Therefore, the Energy saving module of the HNM will be in charge of:

- Determining and distributing to the IBNs the configuration options necessary to keep the energy consumption policies established across the network
- Processing all the “intents” in terms of switch on/off decisions coming from the Energy Saving agents embedded in the IBNs.
- Approving/disapproving the local decisions switch on/off decisions taken at the HNM. This can imply the interaction of the Energy saving modules with other HNM functionalities such as topology calculations or radio resources management or QoS to assure the proper network operation.

### 2.3. IBN Functional Specification

The IBNs represent the local control element present in a distributed topology across the SANSA network that will establish control links with the local elements (e.g. satellite and terrestrial modems) to manage the configuration and operational events.

The next summary shows the high level IBN Functional specifications that will be addressed in the following sections,

- The IBN will be able to apply configuration changes in the radio resources allocation over the satellite and terrestrial modems present in each owner node.
- The IBN will be able to apply configuration changes in the topology configuration over the satellite and terrestrial modems present in each owner node.
- The IBN will be able to manage the traffic at routing level over the satellite and terrestrial modems present in each owner node.
- The IBN will be able to get the congestion state and geolocation information of IBNs at a 1-
hop distance so that it can determine the more proper trajectory to route traffic over the
network.

- The IBN will present control capabilities to optimize the routing protocol select for the
data traffic exchanged between the IBN discovered neighbors, regarding to find the most
suitable next-hop based on the current network conditions related to topology and
congestion.

- The IBN will support the possibility to offload traffic (load balancing) to satellite segment
when required by higher-level management entity, the HNM.

- The IBN will perform traffic classification in order to determine which flows can be routed
though the satellite.

- The IBN will be able to run an intelligent energy saving algorithm determining whether the
access and/or backhaul interfaces can be switched –off for energy saving.

2.3.1. **Initial Routine (Auto-provisioning) Module**
Contains the initial bootstrap routine, this routine will include:

- The self-identification information for the IBN.

- The initial configuration will be composed by the IBN identifier, IP Address, Mac Address
and network broadcast method.

- Authentication information, user and password, Key exchange or security certificates.

This information will be exchanged with the HNM in order to provision the IBN node in the system
and presented to the operator across the HMI.

2.3.2. **AAA Operations Module**
Two kinds of AAA operations will be present in the SANSA environment under the IBN perspective.
First one, related to the communications between the IBN and HNM necessary to apply an
authentication, authorization and accounting process to complete this operation. The second one
is related to communications between the IBN and the local elements present in each node (e.g.
satellite and terrestrial modems). The protocol exchange between the IBN and the different
devices and all the configuration information required will be provided by the HNM depending on
the node identification information collected. In addition, a method to encrypt the communication
path should be recommended but is not mandatory for the AAA exchanges.
2.3.3.  **Topology Management Agent**
As detailed in Section 2.2.1, all the topology management processing is carried out at the HNM. Therefore, the IBN functionalities in this regard, are limited to applying the configuration changes decided by the HNM and to collect information for the HNM, as summarized here:

- The IBN will apply the configuration changes received by the HNM to the local elements (e.g. satellite and terrestrial modems) to reconfigure the radio access resources (Physical links status) activating and deactivating the links.
- The IBN will be responsible to apply the configuration changes of the different elements without causing any changes in the operating configuration. The new configuration is activated when there is a commitment to the new changes.
- The IBN will be responsible to collect all the events related to topology changes received from the elements and propagating this information to the HNM.

The Simple Network Management Protocol (SNMP) protocol will be used by the IBN for message exchange and events/stats gathering from the local modems, hence requiring a proper SNMP plugin in the HNM able to interact with the SNMP agents installed in the IBNs in the implementation phase, which will be conducted in WP5 and demonstrated in WP6.

2.3.4.  **Radio Resources Agent**
As in the case of the Topology Management, the Radio Resources processing is done at the HNM and thus, the IBN only applies the suggested configuration changes and collects information. Therefore, the functionalities of the Radio Resource Agent at the IBN can be summarized as:

- The IBNs will be able to monitor the correct use of spectrum resources by means of frequency monitoring in order to avoid interference between channels and performance reduction.
- The IBN will be responsible to collect statistics and sent to HNM in order to take corrective actions if needed.

2.3.5.  **Routing Management Agent**
In order to exploit the capabilities of the proposed satellite-terrestrial wireless backhaul architecture, SANSA has to provide a self-organized routing protocol capable of meeting the following requirements related to the intrinsic nature of the wireless multi-hop Transport Network Layer (TNL):

- Adaptable to the dynamicity of satellite-terrestrial wireless backhaul deployments.
- Scalable with network parameters (e.g., the number of IBNs and the number of interfaces
in an IBN).
- Decentralization to provide self-organization capabilities.
- Improvement of performance against SoA routing approaches in key performance metrics.

As already anticipated, in order to meet the decentralization and scalability requirements, all the routing processing will run in a distributed fashion on the Routing management agent of the IBN. In what follows, we will provide more specify of each of the aforementioned requirements the routing protocol needs to satisfy.

- **Adaptability to the dynamicity of satellite-terrestrial wireless backhaul deployment:** The increasing requirements in terms of access traffic will increase the density of networks. This increase will lead to deployments with mesh topologies. Mesh topologies offer path redundancy and resiliency, hence decreasing the performance of traditional daisy-chain terrestrial networks, where the network is as robust as one of its links. Redundancy and resiliency are desirable properties present in wireless multi-hop backhauls. Equipment failures, and wireless link variability are some of the common drawbacks of wireless multi-hop backhauls that a redundant topology, like mesh topologies, can potentially mitigate. On the one hand, the wireless backhaul may be subject to traffic dynamics. Therefore, maintaining all Small Cells (SCs) in active state when traffic conditions are light is unnecessarily resource consuming. A possibility is to power off SCs during light operation conditions (e.g., during the night), thus ending up with an appropriate percentage of nodes powered off. Even though these mechanisms can potentially support high energy efficiency gains; they also substantially alter the wireless backhaul topology. On the other hand, these SC deployments may suffer from node and link failures due to vandalism to ambient conditions, or obstacles. Finally, another factor which can modify the topology of the network proposed within SANSA project is the change in the pointing capabilities of IBNs, which could be equipped with smart antenna devices. Because of the high degree of adaptability attached to the dynamics posed by the proposed hybrid terrestrial-satellite backhaul transport network, it is of primal importance and maybe the most important requirement to design mechanisms at the TNL to leverage redundancy and resiliency to a variety of topological models. The range of path redundancy may vary depending on the deployment. Nevertheless, the IBN requires optimization at the TNL in order to manage the available routes offered by redundant topologies (mesh) or hybrid topologies (satellite component) appropriately and to make an efficient use of such resources. The resulting solution will be evaluated on a wide range of simulation deployments. The routing solution will be extensively subject to evaluation. We will conduct studies in urban SC deployments where a high densification of cells is present. We will also consider rural deployments that will include more sparse deployment of terrestrial cells and a satellite will be used to, for instance, offloading traffic from the SCs. Finally, we will study the solution in
the context of HetNets deployments where macro BSs are combined with SC deployment. Results obtained from simulation will confirm the suitability of this approach to conduct routing in hybrid satellite-terrestrial backhaul scenarios.

- **Scalability with Network Parameters**: The routing protocol has to provide support for scalability. In the access network segment, dense deployments are expected to fulfill capacity requirements in urban scenarios. The move towards capacity-oriented deployments has given a starring role to Small Cells (SCs), as increasing frequency re-use by decreasing cell radii has historically been the most effective way to increase capacity at the spectrum level. From a SANSA perspective, this may entail massive deployments of IBNs, with a variable number of heterogeneous interfaces per IBN. To exploit such a benefit, the TNL running on top of these interfaces must scale with the size of the backhaul, the number and nature of wireless interfaces equipped at each IBN, the number of aggregation gateways either towards the Evolved Packet Core (EPC) or to the satellite segment between others. The proposed solution will have as requirement to also scale with the number of traffic flows injected in the network.

- **Decentralization to provide self-organization capabilities**: Usually, real equipment faces implementation constraints to pass from analytical research in the field of routing protocols to real world implementations. There are assumptions, like network control centralization that are very difficult to ensure or to have availability (at least for short scale decisions) for all kind of deployments. Hence, the routing protocol would require operating in a decentralized manner, avoiding the excessive use of the wireless channel to transmit routing control messages. In this way, the reduced control information will render more resources to be used for data traffic.

- **Improvement of performance against SoA routing approaches in key performance metrics**: The routing protocol is expected to allow coping with the increasing demands of data traffic to run high-throughput applications. Therefore, adding high throughput to the set of requirements of the intended routing protocol becomes very important. At a first thought, whilst wireless mesh topologies can provide extensive connectivity and cost reduction, they may do so at the cost of losing capacity. However, the wireless technologies envisaged for the use in the SANSA network (high directivity and pointing capabilities) decreases the impact of such first thought. Nevertheless, throughput is an always-increasing demand, and so, a main requirement for wireless network operators. In light of these phenomena, one may think on high-throughput oriented wireless routing protocols in order to propose an appropriate TNL mechanism that satisfies the ever-growing capacity demand. In addition to throughput, we also include load-balancing capabilities to make most out of the deployed resources to make an efficient use of the satellite network. However, throughput is not the only requirement; latency and packet
delivery ratio are also a network metric of primal importance for mobile network operators. Therefore, the comparison of our solution with SoA TNL routing approaches in terms of the aforementioned key performance metrics will determine the success of the TNL routing strategy proposed for the SANSA hybrid backhaul network.

2.3.5.1. Preliminary Simulations Results

The aim of this section is to further motivate the use of dynamic routing and load balancing strategies. The preliminary network deployment consists of a 2x3 network of IBN as the one depicted in Figure 2.4. We use an ideal shortest path (in number of hops) routing protocol to forward traffic through the hybrid backhaul network. This evaluation also allows us to test and validate the ns-3 [21] simulation framework, which is also detailed in Section 4.1.

![Figure 2.4. Hybrid Terrestrial - Satellite backhaul Network under conditions](image)

- Case Study 1: Satellite Offloading, flow in the middle

This experiment illustrates the performance of a TCP flow when changing its path to arrive to the EPC. The Flow1 generated by a UE attached to IBN#1 arrives to the EPC using the terrestrial resources. At instant $t$, a new flow from this UE (Flow2) enters in the network and the HNM notifies the IBN, at which the UE is attached, to change the path of Flow1 from terrestrial to satellite backhaul, while Flow2 reaches the EPC using terrestrial backhaul. In such scenario, Flow1 will use the terrestrial backhaul to arrive to an IBN equipped with a satellite terminal.
Figure 2.5 shows the evolution of the throughput experienced by these flows when using NewReno TCP variant, the default TCP variant in Ns-3. We can notice how the satellite backhaul degrades the performance of Flow1 when routed over the satellite resource, while Flow2, routed over the terrestrial network, can achieve the injected throughput. The sudden increase in round trip time (RTT) of Flow1 (around 500ms) causes Retransmission Time Outs (RTO), which lead to an abrupt decrease of the attained throughput. An important observation is that the connectivity of Flow1 is not lost while switching to the satellite backhaul. In this sense, we can conclude that TCP flows without strict requirements of throughput and delay can be seamlessly transported through the satellite backhaul. This test shows the flexibility of the proposed routing scheme to use either terrestrial or satellite resources. This behavior could bring significant benefits to other competing TCP traffic flows in the network with stricter requirements, especially on latency and throughput, because they could find less congested terrestrial backhaul resources.

Figure 2.5. Throughput evolution of different flow attained in the hybrid wireless backhaul

- Case Study 2: Satellite Offloading: Increasing the number of terrestrial and satellite GWs

In this case study, there are several UEs generating the same amount of traffic are attached to each IBN in the network under evaluation. Half of the traffic arriving to the IBN is mapped to reach the EPC using the satellite backhaul and the other half of the traffic reaches the EPC using only the terrestrial resources. The aim of this experiment is to see the impact on the network performance in terms of throughput when adding new satellite terminals progressively (from zero to five) and
when changing the number of IBN nodes connected to the EPC (from one to two). Notice that in this experiment, an IBN connected to the EPC cannot be equipped with a satellite terminal. Traffic and link rates in the terrestrial and satellite backhaul network are dimensioned to achieve network saturation conditions, hence studying network performance when there is scarcity of resources.

The general trend is that the attained throughput will grow with the number of satellite links, but this is not always true as depicted in Figure 2.6. In the figure, we can observe that throughput gains are marginal when a third satellite link is added in the case of a single terrestrial GW to the EPC.

![Figure 2.6. Normalized throughput evolution with the variation of satellite-terrestrial links connected to the EPC.](image)

This misuse of satellite resources is due to 1) the static traffic management policy and 2) the deployment of satellite link in a non-congested zone. Furthermore, Figure 2.6 reveals that more resources could even translate into performance degradation. This is the case of introducing a single satellite terminal in the network. Both the satellite backhaul network and the allocation of terrestrial resources to reach the satellite backhaul get congested due to this static traffic management policy.

In summary, two conclusions can be extracted from these preliminary simulations:

- First, additional resources may be carefully planned for an efficient exploitation so its deployment brings significant improvements to compensate the additional CAPEX
- Second, rather than static traffic allocation techniques, dynamic routing, traffic management and load balancing strategies are required to exploit these additional deployed resources.

### 2.3.5.2. SANSA Routing Algorithm: High-level Specification

After analyzing the State-of-the-art of routing techniques that can be found in the Section A of the Appendix, and in order to attain the previous requirements, it is decided that the SANSA backhaul routing algorithm will rely on a practical combination of backpressure and geographic routing. The roots of this solution come from the work specified by authors from [56] in the context of single-radio wireless mesh backhauls based on WiFi. It is important to note that the routing approach in [56] will yield inefficiencies in SANSA hybrid satellite-terrestrial scenarios and a different routing model is needed due to the different nature of the backhaul studied in SANSA. The routing approach in [56] only applies for dense terrestrial Point to MultiPoint (PMP) networks that do not need to cope with packet reordering (i.e., no TCP traffic injection). In what follows we will specify the main challenges to overcome in SANSA network and we will provide a high-level specification of the building blocks needed to devise an efficient routing solution for SANSA.

First, SANSA multi-radio (with PtP or PMP backhaul links) setups bring some challenges. First, and due to their multi-radio nature, IBNs must appropriately handle head-of-line (HoL) blocking, by which packets that cannot be transmitted through a certain interface (e.g., medium busy) block other packets queued behind them, when, in fact, these other packets could have been transmitted through another free interface. Previous proposals based on the backpressure concept assume a single radio per each backhaul (or IBN). This leads to an inefficient use of wireless backhaul resources, and consequently, to the degradation of the backhaul network performance. Thus, SANSA requires of a solution that entails a per-interface queue and so a queue management system.

And second, previous proposal exploiting backpressure and geographic routing take decisions on a per-packet basis of UDP traffic whereas especially due to the inclusion of high latency satellite links and TCP traffic (requiring in-order packet reception), the SANSA routing algorithm requires to take routing decisions using longer time-scales. To overcome the packet reordering problem, without losing intrinsic characteristic of backpressure and the capability of circumvent congested paths, we apply the concept of per-flow path selection strategy to the backpressure routing combined with geographic routing strategy. Even if this strategy shares the name with the per-
flow queuing system presented in the original backpressure proposal, our proposal does not apply to the queuing system (that continues to be per-interface) but instead on the way the routing decisions are made. Through identifying a flow as an origin to destination packet stream of a transport layer connection between two end-hosts, each node maintains per-active flow state information, or in other words it maps the packets of a flow to its pre-assigned path, calculated the first time the node sees the flow. Nevertheless, a new flow has the flexibility to route dynamically to any of the available paths, and so is able to circumvent congested routes, without actually causing packet reordering at the destination.

![Figure 2.7. High-level diagram specifying the two-stage routing algorithm for SANSA.](image)

The building blocks providing the high-level specification are detailed in Figure 2.7. In particular, the SANSA routing approach will be comprised of two different building blocks to conduct routing decisions:

- **Per interface queueing management system**: This system (see Figure 2.7) will be in charge of accommodating packets in each one of the terrestrial backhaul queues. Note that a Traffic Classification agent will previously have filtered traffic selected to be routed over the satellite network and traffic that can be routed over the terrestrial backhaul network. The policy used to accommodate packets in each of the terrestrial backhaul interfaces to avoid is a novelty with respect to SoA. It is important to note that the discovery of 1-hop IBN neighbours will be conducted by means of the exchange of ‘HELLO’ packets to know the surrounding network conditions, both congestion and geolocation information.
• **Next-hop computation**: At each transmission opportunity, the actual terrestrial next-hop for the head packet at each FIFO L3 queue is determined based on current network conditions. At each per-backhaul interface transmission opportunity, the packet head of the interface is scheduled to be forwarded. If the packet head corresponds to a traffic flow \( f \) with no route entry present in the routing table, the next-hop is determined by the computation of a weight metric amongst the current node and all its neighboring nodes. The link obtaining the maximum weight is the one scheduled to be transmitted. To take the actual decision backpressure and geolocation information will be used. In essence, the computation of this weight metric will depend on the combination of two routing components, namely, backpressure and geographic routing, combined by a parameter which trades-off the importance of both components aiming to forward packets following the less congested path heading to destination. Subsequent packets belonging to the same traffic flow \( f \) will use the same route to reach the EPC or to reach the destination UE. Thus, geolocation and backpressure information will merely be used for the first packet belonging to a traffic flow. This is also a novel approach to carry LTE traffic in comparison with SoA. For load balancing amongst the the satellite backhaul and the terrestrial backhaul see Section 2.3.6.

2.3.6. **Quality of Service Agent**

As mentioned in Section 2.2.4, in order to exploit the offloading though the satellite without impacting delay sensitive traffic, the hybrid IBN with both ST and TT will have capabilities to classify traffic depending on QoS, acting over network resources. A summary of traffic classification techniques can be found in Section C of the Appendix.

In the terrestrial network and in the context of QoS control, traffic classification is performed at the level of Ethernet service packets by the IBN. In particular, the service packets are classified into different flows, and then they are sent to the corresponding flow queues for rate management or priority queues for further processing. Traffic classification of service packets can be performed with respect to:

- TCP/UDP port number.
- VLAN (outer VLAN) identifier.
- IEEE 802.1p - provide QoS at Media Access Control (MAC) layer.
- VLAN and 802.1p.
- Differentiated Services Code Point (DSCP) – coarse-grained mechanism employing a field in the IPv4 and IPv6 header; provides QoS at the Network layer.

Therefore, the IBN will be responsible to provide the mechanisms that guarantee proper flow of data for applications in order to maximize the Quality of Experience (QoE) for users. It will
implement techniques as ‘passing through packet’. The following process describe the functionality:

1. The packet is examined to decide whether it is an incoming or an outgoing packet. In case of incoming packets, the process ends without marking the packet as it is not beneficial to mark incoming packets.

2. In case of outgoing packets, packet size is examined to see if the current packet size is already the size of the Maximum Transmission Unit (MTU). The process continues only with those packets that are smaller than the MTU.

3. The process continues only with TCP or UDP packets.

4. The five-tuple identifier of the packet, it is checked whether there is already available information about which application the flow belongs to.

5. When all information is prepared for the marking of the packet, there is a final chance to decide whether the packet should be marked or not. The packet marking can be done for all of the packets in the flow, randomly selected packets of the flow or only the first packet of the flow. Additionally, it is also possible to switch off the marking for specific applications.

The marking is done by extending the original IP packet with an option field. The Router Alert option field is selected because the existence of this field is transparent for both the routers on the path and also for the receiver host. If one uses any other option field, it should be carefully checked whether the marking conforms to the security policy of the given network, otherwise the marking could be removed by an edge router in the border of the access network. In the option field, the first two characters of the corresponding executable file name are added, thus increasing the size of the packet by 4 bytes. The packet size field in the IP header is also increased by 4 bytes and the header checksum is recalculated. A flowchart describing the previous process is also shown in Figure 2.8.

The IBN will benefit from the Traffic classification in order to provide load balancing amongst the terrestrial and satellite backhaul network. Load balancing here is a special term of load distribution which tries to maintain balance across processing elements [54]. The most suitable technique is the policy based routing since it is the simplest solution as it will not add excessive complexity to the IBN when it comes to the link selection and will be able to select the most suitable link based on the application and its needs when it comes to latency, bandwidth and jitter. Latency sensitive applications can be sent through terrestrial backhauls, whereas latency tolerant yet bandwidth hungry applications (e.g. video) can be sent through satellite backhaul. The backhaul network
(terrestrial or satellite) selection mechanism will be linked with the QoS mechanism described before and a fully functional component will be presented in deliverable D4.3.

The set of functionalities that the Quality of Service Agent embedded in the IBN must fulfill are:

- Include “passing through packet” traffic classification techniques oriented to service operation.
- Traffic load balancing by determining the backhaul network of choice (satellite or terrestrial) for each traffic flow.
- Include a link selection mechanism (satellite link vs terrestrial-link) based on policy based routing.
- To connect the link selection mechanism (satellite link vs terrestrial link) with the traffic classification technique.
Figure 2.8. Flowchart diagram
2.3.7. Energy Saving agent

The goal of the Energy Saving Function (ESF) in SANSA is to control the operational states of the terrestrial access and backhaul in order to reduce the grid energy consumption of the terrestrial network, while satisfying the traffic demands. In the SANSA vision, energy harvesters (i.e., photovoltaic panels) are used to power IBNs, mostly their terrestrial interfaces. The ESF is in charge of controlling the energy inflow and spending based on traffic demands and enables interface sleep mode.

Energy saving in cellular networks is becoming a key requirement for network operators to reduce their operating expenditure (OPEX) and to mitigate the footprint of Information and Communication Technologies (ICT) on the environment. Costs and greenhouse gases emissions of ICT grew in the last few years due to the escalation of traffic demand from mobile devices such as smartphones and tablets. The annual growth rate in global mobile data traffic has been about 81% and cloud-based and Internet of Things services are expected to further aggravate this trend. In fact, it is expected that the fifth generation (5G) of cellular networks will support 1,000 times more capacity per unit area than 4G. Base stations are considered the most energy hungry elements in a mobile network since they need more than 70% of the whole network energy to operate. For this reason, SANSA is focusing on the access and backhaul segments.

The reference power consumption model for BSs has been introduced by the EU EARTH project [36]. It captures the power consumption of different BS components: radio frequency circuitry, baseband unit, power amplifier, antenna feeder, DC-DC power supply, main supply and active cooling. The BS power consumption is a linear model as below:

\[ P_b = \Delta_p P_{tb} + P_f \]

Where \( P_f \) is a fixed term that includes the power consumption of the BS when idle (i.e. consumption at power supply, cooling, RF circuitry and backhaul) and \( \Delta_p P_b \) is the component proportional to the traffic load. The slope \( \Delta_p \) is different based on the BS type: its value increases with the coverage and max transmission power of the BS. It is worth mentioning here that the slope of the power consumption curve is almost flat for femto and pico BSs.

The energy saving problem is well described by the throughput/energy trade-off, i.e., to satisfy the (entire) traffic demand, by draining the minimum (zero) amount of energy from the power grid. Current scientific literature can be classified into two main approaches: minimizing the energy consumption of the network elements and introducing energy harvesting hardware within the network elements. A detailed state-of-the-art of these two paradigms is reported in the Appendix of this deliverable.
BS sleep mode is considered the most promising energy saving technique, based on the literature review. In our previous study [57] we theoretically proved that, by enabling these methods, a heterogeneous dense network can reach up to 10% of energy savings, depending on the scenario and the system capacity. However, the SANSA objective is to reach an improvement of the 30% in energy efficiency. This is the reason why SANSA considers also the introduction of energy harvesters to supply BSs. It must be remarked that this solution is not only interesting from a pure research view, but also from the real market perspective. In fact, renewably powered BSs have been recently deployed by operators and vendors to reduce their energy costs: examples are the Docomo Eco-Tower [58] and Nokia Future Cell technology [59].

The introduction of the energy harvesters increases the CApital EXpenditure (CAPEX) due to the additional hardware. Small sizes are then required to maintain the initial cost at minimum. For this reason, the main requirement of the SANSA ESF is to control the BS duty cycle, based on the traffic demand and the available harvested energy. ESF is designed to be an intelligent control agent placed within the IBN and programmed in software. ESF is in charge of:

- On-line, cognitive and energy-aware network resource optimization;
- Automatic selection of interface sleep/wake up modes (duty cycle);
- Dynamically adapting to varying traffic conditions and radiation patterns, following the self-organizing network (SON) paradigm;
- Communicating the sleep/wake-up information to the HNM for the proper management of the terrestrial backhaul resources.

In the following section, we will present a preliminary design of the ESF, based on the definition of an energy-aware reinforcement learning agent, which minimizes the energy drained from the power grid by the terrestrial segment.

### 2.3.7.1. Terrestrial Network Energy Optimization Modeling for simulations

A proper management of the SCs and its TNL calls for a lightweight and scalable architecture such as the SON one. This paradigm is of paramount importance, especially considering that SCs will be operated in an uncoordinated fashion. Previous research work demonstrates that sleep strategies (or switch ON-OFF) are a valuable means to reach these goals [43]. However, when using energy harvesting (EH) as unique power supply, we also need to consider the erratic and intermittent nature of renewable energy sources, which entails some additional complexity.
Therefore, for the ESF we propose to model the SC network by means of a multi-agent system where each agent makes autonomous decisions, according to the Decentralized SON (D-SON) paradigm. In this context, we propose a distributed on-line solution based on a multi-agent Reinforcement Learning (RL) algorithm, known as distributed Q-learning. Through RL, each agent (SC and IBN) independently learns a proper radio resource management (RRM) policy, so as to jointly maximize the system performance in terms of throughput, drop rate and energy consumption, while adapting to the dynamic conditions of the environment, in terms of energy inflow and traffic demand.

**System Model**

We consider a two-tier HetNet composed of heterogeneous LTE BSs, which includes one macro Base Station (BS) and N SCs. The macro BS is connected to the power grid and provides baseline coverage to the whole cell. The SCs are deployed in a hotspot manner to increase the capacity where needed (e.g., shopping hall, city center, etc.). Also, these SCs are solely powered through solar-harvested energy and are controlled in a distributed fashion by means of Q-learning agents, as we detail in the next section. The SCs are connected to the core of the network with the wireless backhaul TNL.

From radio access network perspective, LTE is based on Orthogonal Frequency Division Multiple Acces (OFDMA). The total transmission bandwidth $B$ is divided into $R$ resource blocks (RBs) of 1 msec each (referred to as TTI). Each SC $i$ has a set $U_i$ of associated users, which depend on its geographical location and on the distribution of the users. As for the BS power consumption, we adopt the model previously presented. The values of $\Delta_p$ and $P_f$ for macro and small BSs are reported in Table 2-2. Regarding the type of SC, we consider medium scale factor “metro cells”, such as the Alcatel-Lucent 9764 Metro Cell Outdoor featuring a maximum transmission power of 38 dBm.

<table>
<thead>
<tr>
<th>BS Type</th>
<th>$\Delta_p$ [w]</th>
<th>$P_f$ [w]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Macro</td>
<td>600</td>
<td>750</td>
</tr>
<tr>
<td>Small</td>
<td>39</td>
<td>105.6</td>
</tr>
</tbody>
</table>

The (time-varying) BS capacity (in terms of number of resource blocks allocated to the users) is defined based on [47]. This includes the simulation of the wireless channels and the selection of the modulation and coding scheme (MCS) for each user, based on the particular channel conditions and on the (dynamically computed) system interference. For the SC management, we
assume a slotted time model with a slot duration of 1 hour. This time granularity is deemed appropriate to track variations in the system load and in the EH process.

**Algorithm**

We consider a network setup of $N$ distributed agents (the SCs and their correspondent IBNs), which can be modeled by means of a multi-agent system, as it fulfills the following conditions: (1) the intelligent Radio Resource Management (RRM) decisions are made by multiple intelligent and uncoordinated agents; (2) the agents partially observe the overall scenario; and (3) their inputs to the intelligent decision process differ from agent to agent, since they come from spatially distributed sources of information. In particular, the inputs to the RRM algorithm depend on the SC's particular location and on the geographical distribution of the users (i.e., the load). The objective of the algorithm is for each agent to learn, through real-time interactions with the environment, an energy management policy by means of a Q-learning approach. The decision making process of each agent is defined by a Markov Decision Process with state vector $\mathbf{x}_t = \{x_1^t, x_2^t, ..., x_N^t\}$, where $x_i^t$ is the state associated with SC $i$ (described below), at time $t$. Based on $x_i^t$, each agent $i$ independently chooses an action $a_i^t$ from an action set $A$. As a result of the execution of this action, the environment returns an agent dependent reward $r_i^t$, which allows the local update of a Q-value, $Q(x_i^t, a_i^t)$, indicating the appropriateness of selecting action $a_i^t$ in state $x_i^t$. The Q-value is computed according to the rule:

$$Q(x_i^t, a_i^t) \leftarrow Q(x_i^t, a_i^t) + \alpha \left[ r_i^t + \gamma \max_{a'} Q(x_{i+1}^t, a') - Q(x_i^t, a_i^t) \right]$$

where $\alpha$ is the learning rate, $\gamma$ is the discount factor, $x_{i+1}^t$ is the next state for agent $i$ and $a'$ is the associated optimal action. For more details on Reinforcement Learning (RL) and Q-learning the reader is referred to [48].

In detail, we have defined the state, action set and reward function for the ESF of the $N$ agents below.

The local state $x_i^t$ is defined by $x_i^t = \{S_i^t, B_i^t, L_i^t\}$, where $S_i^t$ is the state of the renewable energy source based on the incoming amount harvested energy (e.g., day and night), $B_i^t \in (0^+, B_{max}]$ is the normalized battery energy level, $L_i^t$ is the normalized load for SC $i$ in slot $t$, which depends on the number of users served by this SC.

The set of possible actions $A$ consists of the two actions of switching ON and OFF the SC and its IBN. We have not considered the option of modulating the load $P_f$ between 0 and 1, due to the energy profile of SCs. In fact, the $\Delta_p$ parameter, showed in Table 2-2, for the SCs is usually small and therefore the parameter $P_f$ has a marginal impact on their energy consumption. When a SC is
switched OFF, the associated users have to connect to the macro BS. However, in case the macro BS is not able to provide them with service, they will be dropped, until the next time slot, when a variation of system state may lead to different RRM decisions. By switching off the SC, also its IBN is switched off, which implies that the link in the TNL will not be available.

Finally, the reward function is defined as follow:

$$r_t^i = \begin{cases} 0 & B_t^i < B_{th} \text{ or } D_t > D_{th} \\ kT_t^i & B_t^i \geq B_{th} \text{ and } D_t \leq D_{th} \text{ and SC } i \text{ is ON} \\ 1/B_t^i & B_t^i \geq B_{th} \text{ and } D_t \leq D_{th} \text{ and SC } i \text{ is OFF} \end{cases}$$

where $T_t^i$ is the normalized throughput of SC $i$ in slot $t$, $D_t$ is the instantaneous system drop rate, defined as the ratio between the total amount of traffic dropped and the traffic demand in the entire network (accounting for macro and small BSs). $D_{th}$ is the maximum tolerable drop rate and $B_{th}$ is a threshold on the battery level. The rationale behind this reward function is the following:

The condition in the first line implies a zero reward when the battery level falls below $B_{th}$ ($B_t^i < B_{th}$) or the system drop rate is above $D_{th}$ ($D_t > D_{th}$). This implies a higher reward and incentivizes the SC to turn itself OFF to save energy. When $B_t^i < B_{th}$, this is necessary to promote the energetic self-sustainability of the SC, whereas when $D_t^i > D_{th}$, the system performance is deemed sufficient. Thus, the SC can be switched OFF and offload the macro BS at a later time. In the second and third line of the function, the reward is proportional to the throughput when the SC is turned ON and is instead proportional to the inverse of the energy buffer level when the SC is OFF. Note that the SC, after a learning phase, will choose to remain ON (and offload the macro BS) when the reward in the second line is higher, i.e., when $kT_t^i < 1/B_t^i$. Note that $1/B_t^i$ may dominate over $kT_t^i$ in case battery level and throughput are both low. In this case, the SC switches OFF to save energy. The constant $k$ is used to balance the impact of the two terms (throughput vs energy saving).

**Simulation Scenario and Preliminary Results**

Simulation will be carried out by means of an octave system level simulator designed around the model presented previously. We start evaluating the algorithm in a simple scenario with a random deployment of SC where all of them have a direct connection with the EPC Node in order to preliminary investigate the algorithm viability and highlight its strength and weak aspects. On this matter, we considered $N$ SCs operating within a square macro cell area with a side of 1 km ($N$ is varied as a free parameter). We will consider a transmission power of 38 dBm for SCs, which translate into a coverage radius of 50m. 120 users (UEs) are uniformly placed within the coverage area of each SC. The number of UEs has been selected so that the SCs are congested during the
traffic peaks. The traffic of these users follows an urban profile (i.e., traffic peaks are concentrated around working hours). For what concerns the distribution of traffic among users, we adopt the model in [47], configuring 20% of the UEs as heavy users (their data volume is 900 MB/h), while the remaining UEs are ordinary users (112.5 MB/h). For the renewable energy sources we consider the Panasonic N235B solar modules, which have single cell efficiencies of about 21%, delivering about 186W/m². For SCs, an array of 16×16 (4.48 m²) solar cells has been chosen.

The battery size of the small cell is 2 kWh (panel and battery sizes have been chosen so that SC batteries can be replenished in a full winter day). Harvested energy traces have been obtained using the SolarStat tool [47], considering the city of Los Angeles as the deployment location. These traces have been translated into a Markov process with 12 energy states that, as shown in [47], provide an excellent approximation of the harvested energy process, and so are used for this purpose in our simulations. Figure 2.9 shows typical profiles for the traffic demand and the harvested energy across two subsequent days. Interestingly, we see that the maxima in the energy inflow and in the traffic demand are not aligned. This means that some optimization actions that could be taken are e.g., saving energy resources and use them when the next traffic peak occurs.

![Figure 2.9. Example of traffic demand and amount of energy harvested](image)

The decentralized Q-learning algorithm proposed is independently implemented by each SC. The learning rate is set to $\alpha=0.5$ and the discount factor to $\gamma=0.9$ for all SCs, according to our simulation analysis. The constant $k$ of the reward function is set to 10 as this provides a good tradeoff for the
considered system parameters. The Q-learning algorithm also implements exploration features, i.e., random states are visited by the learning agents with probability $\epsilon=0.1$.

In the following plots, we refer to “QL” as our Q-learning solution. We compare QL against a greedy scheme (“greedy” in the figures) where SCs are put into a sleep mode (OFF) when their battery level $B_t$ drops below $B_{th}$, and they are switched ON when $B_t \geq B_{th}$. The battery threshold $B_{th}$ is set to 30% of the battery capacity. The threshold on the instantaneous traffic drop rate is set to $D_{th}=0.05$. Simulations are run for 420 consecutive days, where 60 of them are used for the training phase, while the results from the remaining 360 days are used to evaluate the behaviour of QL and the greedy approach. In the following plots, we treat separately the winter and the summer months, as the energy harvesting statistics are very different in these two cases. Specifically, we consider as winter the months of January, February, October, November and December, while the remaining months are classified as summer.

In Figure 2.10, we show the system throughput gain provided by QL with respect to the greedy scheme. It can be observed that the QL approach offers improvements of up to 14%, during the winter months.

![Percentage of throughput gain of QL respect to greedy scheme.](image)

QL is able to outperform the greedy scheme since it considerably drops less traffic, as depicted in Figure 2.11.
To better understand this behavior, we plot in Figure 2.12 an example of the temporal system behavior for a network of 3 SCs and a macro BS for the last week of December. Here, from top to bottom we show temporal traces concerning traffic demand and instantaneous harvested energy (in the same plot), battery level, policy adopted at the SCs (y-label “Action”) and normalized load at the macro BS (y-label “Macro Load”). From these results various observations can be made. First, the policy adopted by QL tends to save energy during the night, and this makes it possible to offload more the macro BS during the day, as it can be seen in the bottom plot of Figure 2.12 in correspondence of the points marked with “(a)”. Also, the impact of our reward function can be appreciated in correspondence of label “(b)”. Here, the QL keeps the SCs ON, as the traffic demand is high, and in this case sleeping would cause congestion at the macro BS. We remark that QL is capable of doing this as it proactively saves some of the harvested energy when the energy inflow is abundant. In contrast, the greedy scheme shows a more aggressive behavior and, as a result, it has no residual energy to compensate for an upsurge in the traffic load.
Figure 2.12. Example of temporal behavior for a network of 3 SCs.

As depicted in Figure 2.13, this translates in a higher system energy efficiency (EE), which is defined as $EE = \frac{E_S}{T_S}$, where $E_S$ is the total energy drained by the macro BS from the power grid and $T_S$ is the system throughput. As we can see, QL offers a higher EE than the greedy scheme. However, the EE diminishes for an increasing number of SCs because the macro BS has to serve a higher number of UEs when the SCs are switched OFF. Finally, when we look at the total amount of energy spent by the system, it is proportional to the served traffic (which is higher for the QL option), so that it approximately amounts to 7.5 MWh in a year for a greedy solution, while it varies from 7.5 (with 3 SCs) to 8.3 MWh (with 10 SCs) when QL is adopted. As a final remark, it is worth mentioning that the same system implemented without energy harvesting capabilities (i.e., where the SCs are grid-connected) would consume from 9.6 (3 SCs) to 17 MWh (10 SCs) in a year, which implies an increment of more than 50% in terms of used energy.

According to these encouraging preliminary results, we will proceed investigating the behavior of the system with a more in depth evaluation considering a tighter integration with the TNL and more realistic scenarios, e.g., like the one in Figure 2.4.
Figure 2.13. Percentage of energy efficiency improvement of QL respect to greedy.
3. HNM – IBN Functional Interoperability Specification

This section describes the functionality data flow to be covered based on the interoperability dependencies between modules at IBNs and HNM and the overall system network behaviour. Several scenarios will be presented to show some specific conditions and how the system is able to manage them, in order to provide a clear picture about how the specifications can be achieved.

3.1. Link Switch-Off Scenario

In this scenario the Energy Saving module at the IBN switches off the terrestrial link, triggering an event at the HNM. The HNM Topology Management subsystem updates its information files and then informs the network about the new topology. Messages sequence is as follows:

- The energy saving module sends an intent request to the HNM determining to switch off a terrestrial beam.
- The HNM by exploiting the topology management modules determines the implications at the backhaul topology level of switching off a terrestrial beam, as demanded by the Energy Saving module installed in the IBN.
- In this case, the energy saving module at the HNM approves the aforementioned request to switch-off one terrestrial beam, based on the intent request received from the IBN Energy saving module.
- The topology management module determines the proper new topology required to satisfy the backhaul demands in terms of traffic as a consequence of switching off the terrestrial beam.
- Once the feasibility of the new topology is assessed by the HNM, it is distributed to the IBNs.
- The IBNs will be responsible to apply these configuration changes in the network elements and confirm to the HNM the successful change application. Figure 3.1 describes the flows between the elements involved.
Figure 3.1. Link Switch Off Message Exchange
3.2. Terrestrial Beam Activation Scenario

In this interoperability scenario, the SANSA Operator decides to enable a new terrestrial link at an IBN. The Topology Management Module asks the Frequency Manager which resources are available for this node. Based on the provided information a command is then sent to the terrestrial terminal to reconfigure its smart antenna to establish a new link with some other terrestrial terminal. The sequence description is depicted in Figure 3.2 and detailed below:

- New topology is calculated.
- Frequency plan is checked by the HNM, specifically by the Radio Resources manager.
- A new topology is generated and mapped on to configuration parameters / commands. Then this information is sent to the IBN topology agents.
- The new configuration is applied and new terrestrial beam is activated.

![Figure 3.2 Beam Activation Messages Exchange](image_url)
3.3. Traffic Off-loading Scenario

In this scenario the IBN detects that the available bandwidth is low. The IBN reports this status to the HNM which, after consulting the Topology Manager Module, decides to activate the satellite resources. Therefore, it configures the satellite hub and the satellite terminals suitable for the use of this new available resource. The possible sequence, depicted in Figure 3.3, is the following:

- The IBN sends statistics to the HNM about link status, additionally QoS agent detect congestion in one of the links (terrestrial link is saturated) and sends information to the HNM.
- HNM decides that a satellite traffic off-load is needed. A new satellite link is configured through the Hub.
- The new unicast/multicast link is established at the hybrid node.

![Figure 3.3 Traffic Off-loading Scenario](image)
3.4. Resiliency Scenario
This interoperability scenario shows the case of a satellite link supporting the failure of a terrestrial link. When the Events Manager at the HNM detects the failure it configures the satellite hub and the satellite terminal to use the new satellite link. The possible message sequence, depicted in Figure 3.4, is the following:

- A terrestrial link in the IBN fails.
- Link failure is communicated to the HNM (or detected by monitoring).
- A new terrestrial beam is configured and provisioned.
- New link is established.
- New topology is distributed.

![Figure 3.4 Resilience Scenario](image-url)
3.5. Terrestrial beam interference Scenario

In this scenario the Radio Resources agent at the IBN detects interference between channels. This situation is reported to the Radio resources manager at the HNM which triggers the corresponding algorithm. After consulting the database at the configuration Module it commands the terrestrial terminal to change its frequency. The message sequence is shown in Figure 3.5 and detailed below:

- Radio Resources monitoring readings from all IBNs are received by HNM through of Radio Resources Manager.
- Radio Resources Manager at Event Management module in HNM detects the interference.
- A new frequency plan is generated.
- Interferer Terrestrial terminal antenna/modem is reconfigured by the HNM through of the IBN.

![Figure 3.5 Interference Scenario](image-url)
4. Functional Implementation Roadmap

Next table summarizes the features evolution from Architecture/Design Stage as part of Deliverables 4.1 and 4.2, across the Simulation Stage covered in Deliverable 4.3 until demonstration stage as part of the Work Package 6 outputs.

<table>
<thead>
<tr>
<th>Table 4-1. Functional Roadmap</th>
</tr>
</thead>
<tbody>
<tr>
<td>AAA – Authentication</td>
</tr>
<tr>
<td>Routing</td>
</tr>
<tr>
<td>Topology Management</td>
</tr>
<tr>
<td>Energy Saving</td>
</tr>
<tr>
<td>Traffic Offloading</td>
</tr>
<tr>
<td>Link Redundancy</td>
</tr>
<tr>
<td>Traffic Classification</td>
</tr>
<tr>
<td>Events Gathering</td>
</tr>
<tr>
<td>Modular Architecture</td>
</tr>
<tr>
<td>HNM – IBN Integration</td>
</tr>
</tbody>
</table>

For the sake of completeness, the design stage considered almost all the key functionalities required for a final product. However, not all these functionalities are object of research inside the project, hence are not developed at simulation and demonstration stages. This is the case of AAA authentication, which will remain at design stage to provide a valid functional approach to incorporate an extra value at solution stage. However, current SoA with multiple solutions present in the market with proprietary approaches made the added value less differential for next stages of the SANSA project. Similarly, the traffic classification module embedded in IBNs will be presented as a concept with the main building blocks and the algorithms behind them along with the interfaces. The focus will be to select and implement algorithms based on a number of different criteria (scenario, traffic, type of deployment etc.) and not on the comparison of different techniques. Therefore, we will not focus on simulations but on the full functional description of
the component and the justification of our choices based on the characteristics of each of the algorithms used to provide traffic classification.

The testing of the developed techniques on a real backhaul network is out of the scope of the project and the small dimensions of the proof-of-concept may limit its evaluation capabilities. Therefore, the simulation stage will complement the proof-of-concept by including extensive evaluations results, which will be provided in deliverable D4.3. In this way, it is the simulation stage the main outcome of WP4. Specifically, it will include: (i) the evaluation of the proposed routing against SoA techniques across a different set of scenarios (spanning from urban dense mesh scenarios formed by small cells to rural terrestrial backhauls complemented with the use of satellite network); (ii) the evaluation of the performance centralized topology management algorithms enabling dynamic topologies (i.e., with HNM) against fixed topologies (i.e., without HNM) under equivalent traffic patterns and routing algorithms; and (iii) the joint evaluation of energy saving algorithms and routing algorithms embedded in the IBN against fixed SoA networks connected to the grid. These simulations, specially (i) and (ii), will consider cases of link redundancy for increased resilience and traffic offloading through the satellite. In terms of traffic classification, simulations that include satellite components will use fixed policies to discriminate amongst satellite traffic and terrestrial traffic.

Note that the ns-3 [21] will be used as simulation framework in all the simulations. The details around the simulation framework used for the aforementioned evaluations is detailed in next section. Finally, the HNM-IBN integration, their modular architecture and the events gathering will be somehow considered and offline emulated in the simulations. For example, in these simulations the Routing agent in the IBN may detect congestion and inform the Topology Management module of the IBN through an offline file exchange. In the same way, the Topology Manager may inform the IBN about a new calculated topology through another offline file exchange.

Once we established the design foundations for the SANSA elements in WP4, for the implementation in WP5 and demonstration stage in WP6, the focus will be on the implementation of the modular architecture of the HNM, as well as their integration, and the feature for gathering events from the IBNs. A subset of the features described in this deliverable will be implemented in the IBN. The implementation will be mainly focused on the topology management features of the HNM applied to a hybrid satellite-terrestrial backhaul, which is one the main features of SANSA when compared to current fixed wireless backhaul (satellite or terrestrial). In particular, this topology management module will be combined with a simplified flavor of routing, given the small dimensions of the proof-of-concept, in order to demonstrate improved capacity and resilience to failures and congestion. The offloading through the satellite and the link redundancy will also play a key role in the demonstration.
More specifically, regarding to accomplish with the HNM and IBN implementation, we can distinguish the HNM functional modules in: commissioning, configuration, event handling and graphic user interfaces. Next diagram shows the functionalities mapping between modules and features at HNM:

The main two activities that need to be addressed are: the interfaces definition and the translation from algorithms to software.

In the case of the interfaces definition, new diagram describe the iBN exposed interfaces, where:

- IF1 and IF2 IPv4 data traffic and control traffic interfaces.
- IF4 and IF5 data and control interfaces embedded.
- IF3 and IF6 Control interface between Channel emulator Terrestrial modems and IBN.
- IF7 and IF8 data traffic external interfaces.

Regarding to start the developments with absences of the rest of elements involving in SANSA network, development team can use virtualized elements and emulators to generate traffic patterns and messages according with the expectations.

GNS3 a network emulator also provides Virtual Machine (VM) allocation capabilities and was the perfect container for IBN development allocation and emulate the rest of the elements present in SANSA environment as shows next.
As illustrated by the diagram, elements such as eNBs, which are embedded in the IBNs, and UEs will be emulated using Traffic Generators tools embedded in Virtual Machines.

Terrestrial and Satellite Modems can be emulated using software image from commercial elements with similar capabilities as routers with control message under SNMP protocol.

Once these elements are putting in place, IBNs deployed in the real testbed can consume data control information from the elements to route data traffic between them, hence validating the topology algorithms implementation embedded in the HNM.

It is important to note that a unified yet simple interface will be developed between HNM and IBN, enabling the control messages exchange amongst the HNM and the IBNs. Note also that data traffic is only managed by IBNs and it is not expected to be forwarded to the HNM.

Regarding to proceed with the translation from algorithms to machine language, parameter definition from topology algorithm have been modulated in a SW using tags to mark all the necessary information, this pseudo language enabled the simulation stage using software, decreasing the implementation times.

A general vision about the activities that will be cover during implementation phases are described next.

During this tasks some transversal activities are being developed as to prepare an appealing web interface that HNM will present to the SANSA operator.
4.1. Network Simulation Framework

The simulation framework used to develop and test the performance of the designed routing protocol is Ns-3 network simulator [21]. Ns-3 is a discrete-event network simulator, targeted primarily for research and educational use. Ns-3 is a free software, licensed under the GNU GPLv2 license, and is publicly available for research, development, and use.

As a network simulator, ns-3 contains a collection of simulation models (wireless and wired network technologies, protocols, applications, topology, mobility, propagation models etc.) as depicted in Figure 4.1. This collection of models allows simulating complete communication systems which contain detailed modeling of layers from 2 (MAC) to 7 (application) and simplified modeling of layer 1 (PHY).

![Ns-3 network simulator stack](image)

**Figure 4.1. Ns-3 network simulator stack**

Ns-3 is written in C++ and counts with a highly optimized simulator core with a modular architecture where adding new models/modules is easy. Ns-3 presents alignment with real-world interfaces (especially Linux) and makes emphasis on real world integration thanks to its emulation mode.

The final goal of the Ns-3 project is to develop a preferred, open simulation environment for networking research: it should be aligned with the simulation needs of modern networking
research and should encourage community contribution, peer review, and validation of the software. The Ns-3 project is a well maintained project which produces a new stable version of the simulator (new models developed, documented, validated, and maintained) every three months. Additionally, the Ns-3 framework counts with the LENA (LTE-EPC Network Simulator) module [35], developed at CTTC.

LENA module consists of an accurate model of the LTE/EPC protocol stack useful for modeling LTE access and the core network. A brief overview of this module is provided in the following subsection. The LENA module is an open source product-oriented LTE/EPC Network Simulator that allows LTE small/macro cell vendors to design and test Self Organized Network (SON) algorithms and solutions. It has been designed around an industrial application program interface (API), the Small Cell Forum MAC Scheduler Interface Specification and provides accurate models of the LTE/EPC protocol stack and specific channel and PHY layer models for LTE macro and small cells, Figure 4.2 shows a high level view of the environment used.

LENA is based on the popular Ns-3 network simulator for internet systems. The development of LENA is open to the community in order to foster early adoption and contributions by industrial and academic partners. Target applications for LENA include the design and performance evaluation of DL & UL Schedulers, Radio Resource Management Algorithms, Inter-cell Interference
Coordination solutions, Load Balancing and Mobility Management, Heterogeneous Network (HetNets) solutions, End-to-end Quality of Experience (QoE) provisioning, Multi-RAT network solutions and Cognitive LTE systems.

The characteristics of the Ns-3 LENA module make it very useful in the context of the SANSA project for modeling the access and the core network of LTE networks. The LENA module originally did not support the inclusion of a backhaul infrastructure, which is more complex than a single wired cable between an eNodeB (eNB) and the core network. However, the LENA module has been extended to simulate hybrid backhaul networks, thus, increasing the functionality of the module. This subsection presents the details of the extensions implemented.

The hybrid terrestrial-satellite backhaul network adds two main elements to the LENA architecture to connect the access and the core network segments as depicted in Figure 4.3.

Figure 4.3. Example of hybrid terrestrial satellite-backhaul network proposed by SANSA project

1) Intelligent Backhaul Node (IBN): IBN is each of the nodes forming the terrestrial wireless mesh backhaul. These nodes embed a HeNB (or eNB) and are in charge of forwarding the traffic towards/from the EPC to the corresponding user equipment (UE). In particular, each IBN can be equipped with multiple interfaces to emulate wireless terrestrial links (i.e., microwave or mmWave) or a single satellite link. Terrestrial links can be switched on/off due to the consideration of energy efficient algorithms developed within the SANSA project (see Section 2.3.7 for further details). Additionally, these links emulate the behavior of smart antennas. Hence,
backhaul topology is not only modified due to energy criteria but also due to a change in the radiation pattern of the antenna interface equipment (beamforming).

2) **Satellite:** In this framework, a satellite is seen as a network element offering a reliable and almost ubiquitous communication channel for the terrestrial backhaul nodes. Besides the evident benefits for rural or remote network deployments, satellite terminals (ST) enable data off-loading from the terrestrial network, which in turn results in an overall capacity increase and an improvement in the network resilience in front of link failures. As depicted in Figure 4.3, the role of the satellite is to directly connect the terrestrial IBNs equipped with a ST with the EPC in a reliable manner. In this framework, the satellite link has been modeled as a point-to-point (PtP) link with a certain transmission rate, bit error rate and a propagation delay corresponding to a GEO satellite (in the order of 250ms). It is important to note that not all IBNs may endow a ST to not incur in an excessive increase of CAPEX expenditures.

**Connecting the Transport Network Layer (TNL) to the Mobile Network Layer (MNL)**

As mentioned before, the LENA simulator model does not support the inclusion of a complex backhaul infrastructure between each defined HeNB and the core network. LENA extension consists of a flexible API in the form of a new class called HybridMeshEPCHelper, extending the EPCHelper class to enable the interconnection of the access and the core network segment through a custom hybrid satellite-terrestrial mesh backhaul network. The most important method defined in this new class is AddHybridMeshBackhaul, which is dedicated to build and configure the hybrid backhaul network according to the configuration files provided by the HNM.

![Figure 4.4. LENA extensions to support Hybrid Satellite-Terrestrial mesh backhaul networks](image-url)
In Figure 4.4 we illustrate the sequence diagram of the configuration of the hybrid mobile network: the main program calls the `AddHybridMeshBackhaul` method of the `HybridMeshEPCHelper` class which configures the terrestrial backhaul network and then sets the indicated satellite backhaul connections. Once the backhaul is configured, the access and the core network parts are connected between them through the backhaul segment.

**Figure 4.5. Wireshark capture of data plane traffic traversing a wireless mesh**

**Figure 4.6. Example of a Hybrid Backhaul Network**

Figure 4.5 shows a Wireshark capture of the S1 interface in an IBN which forms part of the hybrid backhaul. These traces belong to the data plane traffic of a UE. Thanks to the extension previously
described, allowing complex backhaul configurations, this UE may not be necessarily attached to the HeNB embedded in the IBN where the trace has been captured, hence showing the multi-hop capabilities of the presented extension. Note that prior to the exchange of data plane traffic, the HeNB embedded in the IBN requires to trigger 3GPP signaling procedures. In Figure 4.5, we can observe how user plane LTE traffic generated by UEs is tunneled over the hybrid satellite-terrestrial mesh backhaul by means of GPRS Tunneling Protocol User Plane tunnels (GTP-U). This protocol tunnels data between the eNodeB and the S-GW located at the EPC node.

**Simulator Input parameters**

Currently, at the time of this description, the SANSA simulation framework requires the following input parameters to run a simulation. As we can see, the implemented simulator API allows a high degree of re-configurability.

- Node positions (in x, y coordinates) to place them in a surface.
- Topological matrix indicating how IBN nodes are connected between them. Notation to indicate the different link state can be the following:
  - 0: no connection between nodes.
  - 1: nodes connected between them and the connection is fixed (no beamforming).
  - 2: nodes that can be connected between them but, currently, the link is not active (because the node is pointing to other node)
  - 3: nodes that can be connected between them and, currently, the link is active
- Topological matrix indicating the link rate of the connected nodes
- Boolean vector informing of the nodes with direct link to the EPC, referred to as EPC vector, which will be fixed during all the simulation).
- Boolean vector informing of the nodes which counts with satellite link (satellite vector, the same in all the simulation).
- Traffic generation capabilities: type of traffic (UDP, TCP), rate, traffic direction (UL, DL), number of UEs per IBN, flow start/end times.

The following example provides the topological matrix (Table 4-2 and Table 4-3) and the Boolean vectors required to configure the network depicted in Figure 4.6.

<table>
<thead>
<tr>
<th>Node Id</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>3</td>
<td>0</td>
<td>0</td>
<td>2</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 4-2 Example of topological matrix configuring Figure 4.6
<table>
<thead>
<tr>
<th></th>
<th>0</th>
<th>0</th>
<th>1</th>
<th>0</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 4-3. Example of vectors indicating connection with EPC and Satellite nodes of Figure 4.6

<table>
<thead>
<tr>
<th>EPC Vector</th>
<th>0</th>
<th>0</th>
<th>1</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Satellite Vector</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

**Simulator Output parameters**

The simulations for the designed routing protocol will allow a high number of tracing characteristics to extract its KPI metrics (throughput, latency, PDR) such as:

- Per node/Per interface wireshark pcap format traces including information of each correct transmitted and received packet.
- Per node/per interface periodical information (the periodicity can be selected):
  - Per interface queue length, dropped packets, transmitted packets, time of the report.
- Per packet/flow information:
  - Source/Destination UE, Source/Destination eNodeB/EPC, time of origin, time of arrival, traversed number of hops, packet length, port destination of the packet and if it corresponds to a “terrestrial” or a “satellite” flow.
5. Conclusions

This deliverable provides the specifications of the functionalities required to be endowed in the HNM and IBN, which present the main novelties at the network layer in SANSA.

As for the functional specification of the HNM detailed in Section 2.2, it is of primal importance to highlight the introduction of topology recalculation algorithms to exploit the unique re-configurability offered by the SANSA IBN, hence building optimized topologies to improve KPIs such as throughput, latency, and packet delivery ratio. As for the IBN detailed in Section 2.3, one of the main novelties is the adoption of the decentralized routing algorithm suggested in Section 2.3.5, which allows the balance of traffic under redundant terrestrial backhauling situations. It is important to note that the balancing of traffic amongst the satellite and the terrestrial backhaul will be conducted by the traffic classification module. Traffic classification amongst the satellite and the terrestrial network to provide QoS, presented in Section 2.3.6, aims provide the best traffic classification policy scheme approach to be followed in SANSA, to achieve this a deep analysis over the SoA have been done taking as reference the last convergence tendencies across IP interfaces. This decision provides compatibility for the application with an End to End view. The capabilities to manage DiffServ or IntServ configurations over the elements in the environment is considered the main goal for SANSA. Finally, the energy savings agent will run in each IBN on a distributed way. We have presented a clear vision about the issues associated with the energy consumption and the necessities to find effective and feasible solutions in order to optimize the energy consumption. To achieve this challenge, the definition of energy saving functions has been completed, by providing a high-level specification of such module as well some preliminary simulation. The application of energy saving policies suggested offers a clear optimization of around a 50% in our simulations scenarios. These savings are directly appointed to reduce the operator OPEX and increase the margins in the business operations.

To complement the specification of each entity in Section 2, in Section 3 we specified some representative interoperability scenarios between the HNM and the IBNs in order to clarify the kind of interactions that need to be established between the HNM and the IBN (e.g., switching-off a terrestrial link, offloading). Section 4 details the stages at which each functionality described in Section 2 not only at the simulation layer in WP4 but also in terms of implementation and demonstration in WP5 and WP6, respectively. Then, we provide a description of simulation framework based on ns-3. The ns-3 LENA simulator module have been key to determine the best system model option and obtain traffic simulation results closest to the real network conditions. Preliminary simulations were conducted with the ultimate goal of validating the chosen simulator for SANSA hybrid satellite-terrestrial backhauls.
References


[20] Open source implementation of HWMP, Available at: https://github.com/cozybit/open80211s


http://dx.doi.org/10.1109/SURV.2012.062612.00109


D4.2: Interoperability of terrestrial and satellite links high level functional specification

Date: 5/10/17


D4.2: Interoperability of terrestrial and satellite links high level functional specification

Date: 5/10/17


Appendix

A. SoA of Backhaul Routing Algorithms

The aim of the SANSA project is to create and design techniques and technologies to achieve a reconfigurable hybrid mobile backhaul network where terrestrial (dense) deployment of macro cells and small cells can integrate seamlessly with the additional resources provided by a satellite backhaul network to increase the performance of the mobile backhaul network. Among the techniques developed within SANSA project, an efficient routing algorithm plays an important role to achieve the mentioned objective. Due to the static and non-power constrained capabilities of the backhauling network nodes, we can identify the proposed mobile backhaul network as a Wireless Mesh Network (WMN). A great number of routing protocols and variations exists for WMN. This section reviews some of the relevant routing protocols suitable for this kind of networks. In the literature we can find several references ([1], [2], [3], and [4]) where routing protocols for WMNs are classified according to different taxonomies, for instance, route discovery method, protocol management, routing metric importance, etc. The following text provides a classification of some of the most relevant routing protocols for WMN from a route-based vs node-based approach perspective.

In route-based or path centric approach, source node knows or starts the procedure to discover the end-to-end route before it starts sending the first packet of the data stream. However, in the node-based approach, the route is discovered while the packet traverses the network. Table 0-1 summarizes the content of this section.

Table 0-1. Taxonomy of Routing protocols suitable for backhaul networks

<table>
<thead>
<tr>
<th>Routing approach classification</th>
<th>Routing protocol examples</th>
</tr>
</thead>
<tbody>
<tr>
<td>Route-based</td>
<td>Proactive protocols: OLSR, DSDV, B.A.T.M.A.N.</td>
</tr>
<tr>
<td></td>
<td>Reactive protocols: AODV, TORA, LQSR.</td>
</tr>
<tr>
<td></td>
<td>Hybrid protocols: ZRP, HWMP.</td>
</tr>
<tr>
<td>Node-Based</td>
<td>Opportunistic Routing: ExOR, ROMER.</td>
</tr>
<tr>
<td></td>
<td>Geographic Routing: GPRS.</td>
</tr>
</tbody>
</table>
A.1. Route-based/Path centric routing algorithms

As mentioned before, this subsection deals with routing protocols for WMN where routes are known or determined before sending the first packet of the data stream. In each forwarding hop, the node checks its routing table to determine the next hop for the packet to relay. According to the procedure to perform the route discovery operation, route-based routing protocols can be further classified into proactive, reactive and hybrid.

In proactive routing protocols, also known as table-driven protocols, nodes maintain routing information of the entire network even before needed, that is, before the data stream starts. This can be achieved with the continuous and periodical exchange of control routing messages to obtain information about the network topology. The main advantage of this strategy is that nodes are able to forward a packet to the next hop node towards destination immediately after receiving it making it suitable for time-sensitive applications. However, the generation of the routing tables requires the continuous exchange of messages between network nodes resulting in a big wastage of bandwidth resources, which is even worse in the wireless scenario. Additionally, the node maintains information about routes that potentially may not be used. Hence, this strategy is not suitable for large-scale networks. Representative examples of proactive protocols are OLSR (Optimized Link State Routing Protocol), DSDV (Destination Sequenced Distance Vector Routing Protocol) and B.A.T.M.A.N (Better Approach To Mobile Adhoc Networking).

Reactive routing protocols are those protocols that only calculate the route when needed, that is, when the source node wants to transmit a packet, the route discovery process starts and finds the route towards the destination. Hence, there is no routing activity at the network if network nodes do not require communications thus reducing bandwidth overhead in the network and the need of maintaining unused routes. However, with this approach, forwarding latency increases due to the route finding process and excessive network flooding can cause network clogging. Within reactive routing approaches, AODV (Ad hoc On-Demand Distance Vector), TORA (Temporally-Ordered Routing algorithm) and LQSR (Link Quality Source Routing Algorithm) protocols will be presented in the following subsections.

Finally, hybrid routing approaches attempt to combine the advantages of proactive and reactive strategies. Initially, the routing is established with some proactively prospected routes for the nearby and frequently used routes and then the demand for far away nodes is served through reactive flooding. Examples of hybrid routing protocols are ZRP (Zone Routing Protocol) and HWMP (Hybrid Wireless Mesh Protocol).

A.2. Proactive routing protocols

- OLSR (Optimized Link State Routing Protocol) [5]: OLSR is a proactive routing protocol defined
in the IETF’s experimental draft RFC3626 based on the traditional concept of link-state routing algorithm. In order to keep the routing information up to date and disseminate link state information throughout the network, OLSR exchanges its topology information with other nodes in the network regularly. This exchange is done by means of two types of control packets, the hello and topology control (TC).

Hello packets are employed by nodes to find out 2-hop neighbor information and so these packets are not retransmitted to the entire network. Once each node knows information about its vicinity, it starts sending TC packets, including its neighbor and the state of the links established between them. This helps other nodes to build the network topology. The improvement introduced by OLSR is that in order to reduce the number of link-state updates, OLSR uses a special mechanism of multipoint relays (MPR). This mechanism consists of the election of neighboring nodes which will be in charge to retransmit the TC packets of a node. Only the MPRs selected from the surrounding nodes are allowed to forward control traffic (the remaining nodes can read these packets but are not allowed to retransmit them). This mechanism provides an efficient method for flooding the control traffic by reducing the number of transmissions required. OLSR also supports a way to advertise networks that are reachable through a node, using the host network association message.

The original definition of OLSR does not include any provisions for sensing link quality; it simply assumes that a link is bi-modal (either working if a number of hello packets have been received, or failed, which is not necessarily the case on wireless networks, where links often exhibit intermediate rates of packet loss. Implementations such as the open source OLSRd [6] (commonly used on linux-based mesh routers) have been extended (as of v. 0.4.8) with link quality sensing. A new version of OLSR, OLSRv2, has been published by the IETF in April 2014 [7]. It maintains many of the key features of the original OLSR, including MPR selection and dissemination. Key differences are the flexibility and modular design using shared components common to next generation IETF Mobile AdHoc Networks (MANET) protocols: packet format and neighborhood discovery protocol. In [6], an open source implementation of OLSRv2 can be found.

- **DSDV (Destination Sequenced Distance Vector Routing Protocol)** [8]: DSDV is an example of proactive unicast routing protocol which is based on traditional Bellman Ford algorithm to compute shortest paths from a single source to all of the other network destinations. The main contribution of DSDV routing protocol was to solve the routing loop problem.

In DSDV, every node maintains a routing table with entries to all possible destination in the network, number of hops (hopCount) to each destination, the timestamp of the last update received for that node, and the last known sequence number, which is used to avoid loops. The sequence numbers are generally even if a link is present and odd when not. The number is generated by the destination, and the emitter needs to send out the next update with this
number. The routing updates are either time driven or event driven (topology change detected). In DSDV, two types of updates are possible. The first one is full dump, used infrequently. Another update approach is incremental update, which is used more frequently. This approach contains only those entries whose metric have been changed since the last sent update. The updates are accepted based on the metric for a particular node. The first factor determining the acceptance of an update is the sequence number. It has to accept the update if the sequence number of the update message is higher irrespective of the metric. If the update with same sequence number is received, then the update with less metric (hopCount) is given precedence. Stale entries are those entries that have not been updated for a while. Such entries as well as the routes using those nodes as next hops are deleted. Whenever the topology of the network changes, a new sequence number is necessary before the network re-converges. An implementation of this protocol can be found in the ns-3 network simulator [21].

- **B.A.T.M.A.N (Better Approach To Mobile Adhoc Networking)** [10]: This proactive routing protocol is under development by the “Freifunk Community” and was born out of a response to the shortcoming of OLSR routing protocol when the network grew very large (hundreds of nodes). The B.A.T.M.A.N project was started as a result of the belief that a routing algorithm for a large static mesh network needs to be developed from first principles. The novelty of B.A.T.M.A.N resides in the decentralization of the knowledge about the routes, that is, single nodes do not have routing tables for the entire network. Instead, each node determines one single next-hop forwarding node for each destination in the mesh. Thus, a very fast and efficient routing scheme is developed, creating a network of collective intelligence, and allowing low CPU and, consequently, less battery consumption for each node.

In B.A.T.M.A.N, all nodes periodically broadcast hello packets, also known as OriGinator Message (OGM), to its neighbors. Each OGM consists of an originator address, sending node address and a unique sequence number. On receiving an OGM, each neighbor changes the sending address to its own address and re-broadcast the message. On receiving its own message, the originator does a bidirectional link check to verify that the detected link can be used in both directions. The sequence number is used to check the currency of the message. Thus, the BATMAN mesh network is flooded with OGM messages until every node has received each of them at least once, or until they got lost due to packet loss occurred in the communication links, or until their TTL value has expired. The number of OGM messages received from a given node via each link-local neighbor is used to estimate the quality of a route. To be able to find the best route to a particular end node, B.A.T.M.A.N counts the OGM messages received from each node in the network and logs which link-local neighbor relayed the message. Hence, B.A.T.M.A.N is able to determine the best link-local route towards every other node in the network. B.A.T.M.A.N project is still ongoing and new releases of the open source implementation of this protocol can be found at [10].
A.3. Reactive routing protocols

- **AODV (Ad-Hoc On-Demand Distance Vector)** [11]: AODV is a reactive routing protocol described in RFC 3561, which offers quick adaptation to dynamic link conditions, low processing and memory overhead (reduction of stale routes), and low network resource use (reduction of the need for route maintenance). The philosophy in AODV, like all reactive protocols, is that topology information is only transmitted by nodes on-demand. The AODV protocol is based on the DSDV routing algorithm, and like this protocol, uses destination sequence numbers to ensure loop freedom. The destination sequence number is created by the destination to be included along with any route information it sends to the requesting station that allows the source to select the most recent and fresh route.

In AODV, different packet types are used for the route discovery, forwarding and route maintenance. AODV uses the two basic packet types for route discovery, the Route Request (RREQ) and the Route Reply (RREP). When a node wishes to transmit traffic to a host to which it has no route, it will generate a RREQ message that will be flooded in a limited way to other nodes. This causes control traffic overhead to be dynamic and it will result in an initial delay when initiating such communication. A route is considered found when the RREQ message reaches either the destination itself, or an intermediate node with a valid route entry for the destination. A RREP message is then generated back to the node generating the RREQ message. For as long as a route exists between two endpoints, AODV remains passive. When the route becomes invalid or lost, AODV will again issue a request. When a link break in an active route is detected, the broken link is invalidated and a Route Error (RERR) message is typically transmitted back to the source to notify other nodes that the loss of that link has occurred. To detect a link break each node transmits HELLO packets periodically with their identifier.

AODV is the routing protocol used in ZigBee protocol. An open source implementation of such protocol can be found at [12]. This implementation has been developed mainly for use in the APE testbed ([http://apetestbed.sourceforge.net](http://apetestbed.sourceforge.net)). Another implementation can be found with the ns-3 network simulator.

In [13], we can find a well-known extension of the AODV protocol to the multi-radio scenario (MR-AODV), which is the likely scenario to find in SANSA networks. In this protocol, the channel assignment is decoupled from the routing operation. The network throughput can be improved when the different radios in a node are configured to different non-interfering channels. MR-AODV defines a new metric called iAWARE that aids in finding paths that are better in terms of reduced inter-flow and intra-flow interference. This metric, together with
new support for multi-radio nodes, constitute the enhancements to the AODV routing protocol.

- **TORA (Temporally-Ordered Routing Algorithm)** [14]: This reactive routing protocol attempts to achieve a high degree of scalability based on the concept of link reversal. It finds multiple routes from a source node to a destination node without using the shortest path solution, which is usual for the routing protocols covered in this section. The main feature of TORA is that its control algorithm attempts to suppress, to the greatest extent possible, the generation of far-reaching control message propagation. The control messages are localized to a very small set of nodes near the occurrence of a topological change. To achieve this, the nodes maintain routing information about adjacent nodes. The protocol has three basic functions: Route creation, Route maintenance and Route erasure.

  TORA builds and maintains a directed acyclic graph (DAG) rooted at a destination. No two nodes may have the same height. Data packets may flow from nodes with higher heights to nodes with lower heights. Therefore, information can be thought of as a fluid that may only flow downhill. By maintaining a set of totally ordered heights at all times, TORA achieves loop-free multipath routing, as information cannot 'flow uphill' and so cross back on itself. The protocol reacts only when all routes to the destination are lost. In the event of network partition, the protocol is able to detect the partition and erase all invalid routes.

- **LQSR (Link Quality Source Routing)** [15]: This reactive routing protocol is based on other reactive routing protocol called Dynamic Source Routing (DSR) [16], which uses source routing (requires accumulating the address of each device between the source and destination during route discovery) instead of relying on the routing table at each intermediate device. However, LQSR has two major differences from DSR. One is that LQSR is implemented as layer 2.5 protocol instead of as network layer protocol. The other is that LQSR protocol employs single-hop link parameters instead of end-to-end path parameters to determine the routes. In the process of setting a connection path, the protocol describes individual links by the quality metric, and then sends back the information to the node that initiates the setting up of the path. Depending on the mobility pattern of network elements, these quality metrics are chosen, changing the performance of LQSR protocol. For stationary nodes as the one in SANSA, the routing metric Expected Transmission count (ETX) achieves the best performance. The ETX is defined as the number of expected transmissions of a packet necessary for it to be received without error at its destination. Though the protocol has many advantages, it is still necessary to develop more appropriate routing metrics that would take into account the specificity of a wireless mesh network.

  There is a well-known extension of the LQSR protocol which takes into consideration the use of the multi-radio architecture in WMN referred to as MR-LQSR [17]. In MR-LQSR, if a node
has multiple radios, they are tuned to different non-interfering channels. The channel assignment is determined by some outside agency and changes relatively infrequently. MR-LQSR extends LQSR protocol with the use of the Weighted Cumulative ETT (WCETT) metric that take into account both quality parameters of the link and the minimum number of hops. This protocol makes it possible to achieve the expected equilibrium (balance) between the delay and the throughput by selecting channels of best quality with a diversity of radio channels taken into account.

### A.4. Hybrid routing protocols

- **ZRP (Zone Routing Protocol)** [18]: In ZRP, each node knows the topology of the network within its routing zone and nodes are updated about topological changes only within their defined routing zone, hence reducing control overhead. When starting a flow, if packet’s destination is located in the same zone as the origin, the proactive protocol using the already stored routing information is used to deliver the packet. On the other hand, if the packets head towards a destination outside the packet’s originating zone, the reactive part of the protocol takes over to check each successive zone in the route to see if the destination belongs to that zone. Once a zone is confirmed to have the destination node, the proactive protocol is used to deliver the packet through the successive zones. ZRP reduces the control overhead, because proactive protocol just discovers zone routes and reduces the delay for routing within a zone that would be caused by the route-discovery processes of reactive routing protocols.

- **HWMP (Hybrid Wireless Mesh Protocol)** [19]: defined in IEEE 802.11s, HWMP is a mesh routing protocol combining the flexibility of on-demand (reactive) routing based on AODV with proactive topology tree extensions. The combination of both elements enables optimal and efficient path selection in a wide variety of mesh networks. HWMP uses a common set of protocol primitives, generation and processing rules from AODV routing protocol adapted for Layer-2 address-based routing and link metric awareness. AODV forms the basis for finding requested routes within a mesh network while additional primitives are used to proactively set up a distance-vector tree rooted at a single root mesh point (MP). The root role that enables building of topology tree is a configurable option of a MP. HWMP protocol is hybrid in the sense that it supports two kinds of path selection protocols, supporting two modes of operation depending on the configuration. These modes are not exclusive, reactive and proactive tree building mode may be used concurrently. The open source implementation of HWMP has been integrated to Linux kernel by Cozybit. Inc. and can be found at [20]. Table 0-2 summarize the route-based algorithms.
Table 0-2. Overview of Available Implementation of route-based routing algorithms

<table>
<thead>
<tr>
<th>Protocols</th>
<th>Open Source Implementation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Proactive protocols</td>
<td></td>
</tr>
<tr>
<td>OLSR</td>
<td>In [6].</td>
</tr>
<tr>
<td>BATMAN</td>
<td>In [10].</td>
</tr>
<tr>
<td>Reactive protocols</td>
<td></td>
</tr>
<tr>
<td>AODV</td>
<td>In [12].</td>
</tr>
<tr>
<td>Hybrid protocols</td>
<td></td>
</tr>
<tr>
<td>HWMP</td>
<td>In [20].</td>
</tr>
</tbody>
</table>

**A.5. Node centric routing protocols**

This subsection deals with routing protocols where the route is discovered on-the-fly based on the links status while the packet traverses the network. Hence, these protocols constitute a reactive hop-by-hop routing approach where no discovery for the full route should be completely done. The following subsections provide some examples of suitable protocols for WMN belonging to different kinds of node centric routing protocol families.

**A.6. Opportunistic routing protocols**

This kind of routing is based on the following principle: when a node wants to transmit a data packet, it directly broadcasts the data packet. Afterwards, the routing protocol decides on-the-fly which of the potential receivers of the broadcasted packet may forward the data packet, and thus become the next-hop. Hence, each packet (or a batch of them) may potentially follow a different path. The potential receivers of the packets need to work in a coordinated way in order to minimize forwarding of duplicated packets, which is one of the main disadvantages of this routing approach.

Additionally, in the context of SANSA, it is worth mentioning that routing protocols falling in this family are a complement because as they exploit the broadcast medium, they are not compatible with PtP links.

- **ExOR (Extremely Opportunistic Routing)** [22]: In ExOR, the starting radio, the source,
broadcasts a batch of packets. At each potential forwarder, the protocol computes the shortest path to the intended destination. The shortest path is estimated by summing up the link costs associated to the path calculated using the Dijkstra algorithm. The link costs are computed using the ETX metric. Thus, each node has all the ETX values to reach the destination and it calculates the minimum ETX value. For coordination purposes, a forwarder priority list (determined by the cost to the destination) is sent in each data packet to schedule the order of forwarding attempts by the next-hop set. As a result, a node only forwards a data packet if all higher priority WMRs failed to do so. Previously open source ExOR implementation was available in 2005, but is no longer obtainable.

- **ROMER (Resilient Opportunistic Mesh Routing)** [23]: The key idea of ROMER is that each packet carries a credit which is initially set by the source and is reduced as the packet traverses the network. As in ExOR, each node also computes a path cost for forwarding a packet from itself to the intended destination. In ROMER, a data packet may be duplicated when traversing the WMN. This may happen because potential next-hops may forward data packets if the credit of the packet is high enough.

  The credit associated to each data packet is decremented at each forwarding step according to the WMR credit cost, which means that more credits are consumed as the packet moves away from the shortest path to the destination; hence data packets are not forwarded through these paths.

**A.7. Geographic routing protocols**

Geographic routing is an attractive solution when position information is available at all nodes in the network [24]. This kind of routing is known as location-based routing approach, contrary to the previous schemes presented in previous sections, which are topology-based approaches.

In geographic routing, packets are routed to the destination based on the geographic position of the destination. Routing decisions are made on a hop-by-hop basis based on the network state instead of trying to construct and then follow a single static link path from source to destination. When a node receives a packet, it will inspect its neighbor table and select the most appropriate next-hop based on some geographic criteria. Hence, geographic routing eliminates the need for expensive topology discovery/maintenance procedures (in terms of both memory and communications). Nodes are less sensitive to path or link breakages, as the path in the conventional sense of the word does not exist. In the case of SANSA scenario, where forwarding nodes are likely to be stationary, the network can rely on GPS techniques to acquire the location of the nodes or on the use of virtual coordinates.
There are many different variants of geographic routing protocols. Recently, they are gaining research interest in the field of WMN due to its scalable and stateless capabilities. An exhaustive survey of this kind of protocols can be found in [24]. This subsection ends with the description of one of the most known geographic routing protocol, the Greedy Perimeter Stateless Routing (GPSR) protocol.

- **GPSR (Greedy Perimeter Stateless Routing) [13]**: GPSR is a geographic routing protocol which combines the two earliest geographic routing strategies: greedy forwarding and face routing. The aim of greedy forwarding is to forward the packet to the neighbor located closest to the destination at each hop. However, this approach has one big drawback: when a node is unable to find a closer neighbor it must drop the packet to prevent the existence of loops. On the other hand, the main advantage of face routing is that it guarantees delivery. In face routing, faces on a planar graph are traversed using the ‘right hand rule’ technique (sometimes left hand rule instead) in which the algorithm keeps track of all the times it crosses the line connecting the source to destination.

GPSR combines these two techniques in the following way: it makes greedy forwarding decisions using only information about a node’s immediate neighbors in the network topology. When a packet reaches a region where greedy forwarding is impossible, the algorithm recovers by routing around the perimeter of the region using face routing. An ns-3 implementation of this protocol is provided by the authors of [26].

### A.8. Backpressure routing protocols

Recently, backpressure routing algorithms are receiving much attention due to their provable throughput performance guarantees, robustness to stochastic network conditions and, most importantly, their ability to achieve the desired performance without requiring any statistical knowledge of the underlying randomness in the network. These characteristics make this protocol a suitable one to be studied within the context of the SANSA project.

Tassiulas and Ephremides developed the roots of dynamic backpressure routing for multi-hop wireless networks in [27]. In essence, it is a centralized policy working on a time-slotted basis which routes traffic in a multi-hop network attaining throughput optimality of any traffic workload capable of being served by minimizing the Lyapunov drift in the network, i.e., minimizing the sum of the queue backlogs in the network from one timeslot to the following one.
Although this protocol shows optimality in terms of throughput, it suffers several drawbacks to make it practical. Besides the high routing complexity due to the need of centralized information, the work in [27] presents high queue complexity (per-flow queuing system) and increased end-to-end latencies derived from the last-packet problem, whereby packets belonging to a flow may get excessively delayed in queues due to the lack of subsequent packet arrivals of the same flow. This happens because decisions are based on sending the packet to the neighbor generating the highest queue backlog differential for a given flow. Therefore, if there is little traffic, no queue backlog differential is generated and packets are not forwarded.

Based on [27], several modifications to the backpressure algorithm have been proposed focusing on decreasing the complexity of queue structures or decreasing the attained latency. The shadow queue concept in [28] reduces the queue complexity of the original backpressure framework by maintaining a counter per destination instead of a queue per flow. Authors from [29] tackle latency problems associated with the original backpressure algorithm by increasing its number of queues. Their scheme proposes per-hop queues in each node. That is, each node maintains a separate queue per packet that has to be delivered to each destination within a certain number of hops. Even though these proposals alleviate to some extent the original backpressure shortcomings, they still require per-flow or per-destination information, making its practical implementation difficult.

However, there are some examples of real world implementations of backpressure like systems using the work of [27] as reference. One such example is the work in [30], where the authors present XPRESS, a cross-layer backpressure stack implementation. XPRESS operates IEEE 802.11 on a slotted manner, uses centralized routing tables, a queue per flow at every node, and uses backpressure to take scheduling and routing decisions as explained in [27].
Another example is the one presented in [31], where the authors present an implementation of Enhanced BP (EBP), an incremental work that reduces end-to-end delay without sacrificing throughput optimality and which is compatible with the decentralized operation of WiFi networks. This approach also maintains a queue per flow at every node and requires topology knowledge to create shortest path information tables.

In [32], Neely et al. build upon the work of Tassiulas and Ephremides a solution for general power control problem for multi-beam one-hop satellite links. In addition to this, Neely et al. has made several novel contributions that has laid the foundations for many future publications by providing joint power allocation and throughput optimality in multi-hop networks while supporting links having generalized inter-link interference and time varying channel capacity.

This result generalizes the results of Tassiulas and Ephremides. Neely et al. defines a concept of network capacity, different from the information-theoretic concept of capacity. They bridge the existing gap between network capacity, throughput optimality, and network optimization. Their work applies to multi-hop wireless networks with general ergodic arrival and channel state processes, and does not require the knowledge of the parameters of these processes.

One of the novel contributions is the introduction of a tuning parameter $V$, which defines the Lyapunov-drift-plus penalty approach. This allows maintaining (theoretically) throughput optimality while coming arbitrarily close to the optimal (minimum) time average power consumption per node. This work constituted one of the first applications of the Lyapunov drift concept for the joint purpose of utility optimization and throughput optimality.

Moeller et al. used Neely’s Lyapunov drift-plus-penalty theoretical framework to implement the Backpressure Collection Protocol (BCP) [34], which is a practical backpressure routing approach for Wireless Sensor Networks (WSNs) dealing with many-to-one low-volume traffic scenarios. This traffic pattern simplifies the management of queues at each node. In fact, since there is only a single destination, nodes in this implementation only count with a single queue. One of the most remarkable features of this implementation is the use of last-in first-out (LIFO) data queues. In particular, they empirically show that by using LIFO queues, the end-to-end delay experienced by packets decreases. However, this improvement has a drawback that some data packets are trapped at queues and hence, they are never delivered to the destination.

After this review on different alternatives of routing in a WMN, we conclude that there are some interesting ideas in combining backpressure and geographic routing approaches, which could be of use to design a routing protocol suitable for the requirements.
B. SoA of Energy Efficiency Algorithms

Addressing the challenge of energy saving in mobile networks is a new, still unexplored, cross-layer and cross-disciplinary area, which requires several paradigm shifts such as energy-aware mobile architectures and protocols, efficient BS re-design, smart grids, cognitive radio, cooperative relaying and heterogeneous network deployment. In the following sub-sections, we focus on reviewing the literature on cognitive algorithms for heterogeneous deployments to compare such proposals with SANSA ESF. First we introduce proposals for increasing the energy efficiency of the network and then we concentrate on the introduction of energy harvesters.

B.1. Energy efficiency

Heterogeneous deployment permits to satisfy the traffic volume in peak hours, while keeping power consumption low due to the small base stations. However, traffic demand experiences significant temporal and spatial fluctuations throughout the day. This leads to the overdimensioning of the network during the low load periods. In these low traffic conditions some cells can become underutilized or even redundant. During the periods of low user activity, the traffic demand can be served by a fewer number of cells, without compromising the Quality of Service (QoS) provided to the users. The identification of the set of BSs to be slept is not a trivial task and is influenced by the behavior of multiple variables, related to the coverage provided by the multiple interfering small cells, load factor of each node in absolute terms and with respect to the neighbors, etc. A random and inappropriate way of enabling sleep mode can deteriorate the performance of the system since the BSs which remain active need to serve some extra traffic.

Most of the literature in this field covers BS sleep mode enablers in macro cellular scenarios, together with cell zooming [37], [38], [39]. These proposals study the possibility to enable sleep mode in underutilized macro cells and extending the coverage of their neighboring cells to cover the holes left by the BSs in sleep mode to meet required QoS. Heterogeneous deployments are on the contrary studied by [40], where a stochastic game approach is introduced to enable sleep mode in a decentralized fashion, while at the same time trying to satisfy traffic demands. Finally, multiple criteria decision analysis and fuzzy logic has been introduced in [41]and [42].

B.2. Energy Harvesting

Thanks to the reduced energy demand of the small cells (SCs), the use of energy harvesters, like renewable energies sources (RESs), to power them has become more attractive in the research community. HetNets powered by RES can help in reducing the impact of ICT in carbon emissions by saving energy in the SC tier and allowing the adoption of energy saving mechanisms in the
macro BS (e.g., sleep mode). Like in a symbiotic process, HetNets can in parallel move toward a more energy efficient network paradigm and help in solving the problem of the huge energy demand.

Research work presented in the previous section demonstrates that sleep strategies are a valuable means to reach energy savings. However, in the case of Energy Harvesting (EH) in HetNets, the erratic and intermittent nature of renewable energy sources has to be considered, which entails some additional complexity.

The increasing interest in EH cellular networks is testified by the rich literature on this topic [43]. In [44], the authors present a design based on stochastic geometry for the management of k-tier HetNets powered by Renewable Energy Sources (RESs). Their model controls the fraction of time that each tier can be kept on, according to its energy reserve. Similarly, in [45], the authors propose an algorithm to control the BS power consumption as a function of the energy reserve and the expected amount of renewable energy that will be stored. However, neither of these two works considers the temporal variations in traffic and in harvested energy processes, which is fundamental for a realistic model of the scenario. In [46], the authors focus on off-grid mesh networks of EH BSs. First, the problem of dimensioning the renewable energy “add-on” (solar panel and battery) is solved by considering typical daily traffic and harvested energy profiles for different cities. Then, an optimization approach, for two-tier networks, is proposed based on the sleep modes of SCs. However, the proposed optimization approach is based on historical data, and is consequently unable to adapt to the dynamic conditions of the system, in terms of harvested energy or traffic demand, as would be desirable in a realistic setting. From the preliminary results achieved, energy harvesting communication networks can potentially lead to a huge decrease in the consumption from the power grid, Opex and greenhouse gases footprint, at the cost of a small increase of Capex for the installation of energy harvesters.

C. SoA of Traffic Classification Algorithms

Traffic Classification implies the process of associating traffic flows with the applications or application types that generated them in order to prioritize, protect, or prevent certain traffic [51].

Quality of Service (QoS) can be defined as the totality of characteristics of a telecommunications service that bear on its ability to satisfy stated and implied needs of the user of the service [Ref: Definitions of terms related to quality of service”, Recommendation ITU-T E.800, ITU-T, 2008]. In the field of computer networking, Quality of Service (QoS) refers to the capability of a network to provide better service to selected network traffic over various technologies. The primary goal of QoS is to provide priority including dedicated bandwidth, controlled jitter and latency (required by
some real-time and interactive traffic), and improved loss characteristics, making sure that providing priority for one or more flows does not make other flows fail[52].

Review of traffic classification methods:

**C.1. Port based methods**

A simple identification method based on the ports used by the application. This is the most common method for data classification. Port numbers and service names are used to distinguish between different services that run over transport protocols such as TCP, UDP, DCCP, and SCTP. It is based on associating a well-known port number to a given traffic type, e.g., web traffic is associated with TCP port 80.

**C.2. DSCP**

DSCP value is a packet header value that can be used to indicate the level of service requested for traffic such as high priority or best effort delivery. In other words, DSCP is a marking that shall be notified by the transport level to the IP level notifying the content which is transported. The different IP nodes need to replicate this mark in the different IP headers even if the rest of the IP header content is modified for routing or other purposes.

DiffServ (Differentiated Services) is a computer networking architecture that is used to classify and manage traffic on IP networks. By classifying traffic streams, it is able to provide Quality of Service (QoS) on network services. DiffServ uses a 6-bit traffic identifier in the IP packet header known as the DS field. The value of this field is known as the Differentiated Services Code Point (DSCP) value. The routers on the network use this field to identify the traffic class to treat them accordingly. The DiffServ RFC recommends a set of traffic class, but theoretically there can be up to 64 different classes of traffic.

**C.3. Deep packet inspection**

A payload-based method which classifies traffic by analyzing the headers and the payload of packets. It poses formidable privacy challenges along with technological and economic challenges.

**C.4. TCP objects (BATS algorithms)**

The QoS-aware link selection algorithm developed as part of the BATS project is based on the assumption that TCP traffic can be split into TCP objects, where an object is typically an Application Protocol Data Unit (APDU), e.g. an HTML request. Long-objects benefit from high bandwidth links and reduce the total time to receive the object and are routed via satellite whereas short-objects are routed via terrestrial links in order to optimize latency. The discovery of
object boundaries is based on an algorithm that computes the Inter-arrival times of the packets and differentiates between the receptions of current or new objects. Based on this process, the size of the objects is updated on-the-fly with the reception of each TCP segment. Segments are initially considered part of a small object and transmitted over the lowest RTT link. Once the object size exceeds the threshold, segments start to be transmitted over the highest bandwidth link [53].

C.5. **Signature based classification**
This classification method is based on predefined byte signatures to identify the particular traffic types, e.g., web traffic contains the string ‘GET’. The common feature of the Signature based methods is that in addition to the packet header, they also need access to the payload of the packets.

C.6. **Statistics based classification method**
In statistics based classification method, some statistical feature of the trace is captured and used to classify the network traffic.

C.7. **Information theory based classification**
The information theoretic approach can group the hosts into typical behaviors e.g., servers, attackers. The main idea is to look at the variability or randomness of the set of values that appear in the five-tuple of the flow identifiers, which belong to a particular source or destination IP address, source or destination port.

C.8. **Combined classification method**
This classification method combines the advantages of different approaches, in order to provide a high level of classification completeness and accuracy. This classification method is based on a complex decision mechanism, in order to provide an appropriate identification mode for each different application type. As a consequence, the ratio of the unclassified traffic becomes significantly lower. Further, the reliability of the classification improves due to the joint decision of various methods.

C.9. **Layer 2 traffic classification methods**
Apart from the methods already presented, there are also methods that use layer 2 information to implement QoS such as:

- VLAN (outer VLAN).
- IEEE 802.1p: provide QoS at Media Access Control (MAC) layer.
- VLAN and 802.1p.
D. Load Balancing Techniques

A brief review of load balancing methods at different layers is presented below.

- **Policy Based Routing**

  Normally an IP routing decision is made based solely on the destination address according to the routers’ routing table. If it is necessary, e.g. due to administrative or performance reasons, that the next hop is selected based on some other criteria, such as the packet size and/or the payload protocol, policies might be defined by the network administrator which are taken into account by the routing decision.

- **Weighted Round Robin.**

  As the name suggests, the basic Round-Robin path selection scheme distributes traffic units in a round robin manner on all available paths without considering any external information or using information gained from the packet headers. This might lead to unbalanced (over/under) utilization of paths if the paths are heterogeneous or if the traffic is not split in packets. The Weighted Round-Robin method addresses this issue by adding a static weighting factor to each connection which expresses bandwidth difference relative to other paths. Based on the weight, less or more traffic units are sent over the corresponding connection. However, weighted round robin can lead to over- or under-utilization of paths as well since it cannot react to varying size of traffic units [54].

- **Multipath TCP (MP-TCP) and Path Selection based on Object Length (PSBOL).**

  The IETF has defined the MPTCP to exploit the situation where there are multiple paths and/or multiple IP Addresses between the source and destination hosts. MP-TCP enables the hosts to use multiple, but not necessarily disjoint, paths and IP addresses while providing the same interface to both the application and the network layer as regular TCP and, thus, does not require any modifications either in the application or in the IP protocol. By using multiple paths in parallel, MP-TCP aims at increasing throughout, reliability and resilience [54].

  In BATS, a novel QoS-aware Link Selection algorithm named ‘Path Selection based on Object Length (PSBOL)’ that takes into account the traffic requirement and the real-time status of each of the links is defined. In the initial implementation, the link selection is performed by analyzing the TCP-object size. An object is a sequence of TCP segments belonging to the same flow, i.e. same source and destination IP and same source and destination port, which are sent within a certain time frame [55].