

# Topology Forming and Optimization Framework for Heterogeneous Wireless Back-Haul Networks supporting Unidirectional Technologies

M.Kretschmer<sup>a,b</sup>, P.Batroff<sup>a</sup>, G.Ghinea<sup>b</sup>

<sup>a</sup>Fraunhofer FOKUS, Schloss Birlinghoven, St. Augustin, Germany

<sup>b</sup>Brunel University, London, United Kingdom

## Abstract

Wireless operators, in developed or emerging regions, must support triple-play service offerings as demanded by the market or mandated by regulatory bodies through so-called Universal Service Obligations (USOs). Since individual operators might face different constraints such as available spectrum licenses, technologies, cost structures or a low energy footprint, the EU FP7 CARrier grade wireless MESH Network (CARMEN) project has developed a carrier-grade heterogeneous multi-radio back-haul architecture which may be deployed to extend, complement or even replace traditional operator equipment. To support offloading of live triple-play content to broadcast-optimized, e.g. DVB-T, overlay cells, this heterogeneous wireless back-haul architecture integrates unidirectional broadcast technologies. In order to manage the physical and logical resources of such a network, a centralized coordinator approach has been chosen, where no routing state is kept at plain WiBACK Nodes (WNs) which merely store QoS-aware MPLS forwarding state. In this paper we present our Unidirectional Technology (UDT)-aware design of the centralized Topology Management Function (TMF), which provides a framework for different topology and spectrum allocation optimization strategies and algorithms to be implemented. Following the validation of the design, we present evaluation results using a hybrid local/centralized topology optimizer showing that our TMF design supports the reliable forming of optimized topologies as well as the timely recovery from node failures.

**Keywords:** QoS, Multi-Radio WNM, MPLS, IEEE 802.21, Traffic Engineering, Unidirectional Links, all-IP network

## 1. Introduction

Research on Wireless Mesh Networks (WMNs) has matured in recent years and Quality of Service (QoS)-support has been widely discussed [18, 36, 1]. However, the applicability of the proposed WMN or Multi-Radio Wireless Mesh Network (MR-WMN) solutions for QoS-sensitive operator back-haul networks providing *triple-play* services is still actively being discussed. Compared to the traditional rather statically planned and configured operator back-haul networks, meshed wireless back-haul networks offer simplified deployment processes due to their flexible self-configuration and self-management characteristics [26]. These enable them to quickly form optimized topologies and to adapt to usage pattern or wireless spectrum availability variations. For example, our heterogeneous Wireless Back-Haul (WiBACK)<sup>1</sup> architecture which is based on the consolidated outcomes of the CARMEN [4] project, supports the proper configuration of heterogeneous technologies, such as packet-switched IEEE 802.11, 802.16 or 802.22 equipment, or broadcast technologies such as DVB-T. Moreover, it also integrates existing technologies such as micro-wave or fiber-optical solutions.

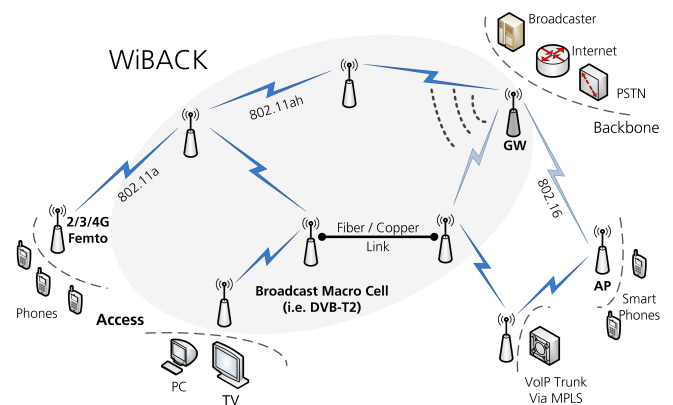


Figure 1: The WiBACK architecture integrates heterogeneous technologies supporting mobile/fixed terminals and trunked payload.

Hence, for each use case, the most suitable technologies may be combined to optimally utilize the available spectrum resources in order to reliably provide back-haul capacity. The selection criteria may be requirements to operate in certain frequency bands, to support low per-node energy footprints, to support rapid temporary deployments or CAPEX/OPEX cost-effectiveness. For example, to address deployment scenarios in rural areas or emerging regions, low-power embedded devices with prop-

<sup>1</sup><http://www.wiback.org>

erly configured IEEE 802.11 radios may be used [13, 23]. Another scenario identified by the CARMEN project is the temporary extension of existing wireless coverage to address high demand periods due to special events such as the London Olympics [8]. This use case assumes an increased demand for live broadcast content, which can introduce a high load on capacity-constrained and especially on collision-sensitive wireless links (Figure 1).

To address this issue, the WiBACK architecture integrates broadcast technologies, such as DVB-T, to enable the network management components to dynamically route such traffic via more efficient broadcast technologies, possibly depending on customer demand, density and distribution. This allows the WiBACK architecture to leverage the existing broadcast infrastructure, exploiting the benefits of the usually longer range of broadcast cells and their higher channel utilization efficiency compared to typical e.g. packet-based IEEE 802 technologies.

The scope of our WiBACK architecture is to provide or extend existing back-haul capacity, which might range from single-hop long distance wireless connectivity to multi-hop connectivity with up to ten hops in urban and rural environments in developed or emerging regions. The interface to external networks at Gateway (GW) or Access Point (AP) nodes can be realized via e.g. regular Internet Protocol (IP), Proxy Mobile IP (PMIP) [12], LAN Emulation (LANE) [24] or Multi Protocol Label Switching (MPLS) trunking, see Figure 1. As depicted in Figure 2, a typical real-world back-hauling pilot scenario implementing our QoS-aware LANE concept has been presented and thoroughly evaluated regarding QoS performance in [23]. Here we roughly assume a 90/10 *best effort*/Voice-over-IP (VoIP) traffic mix and typical back-hauling traffic flows between the AP nodes and the GW. As the wireless technology IEEE 802.11a radios where used.

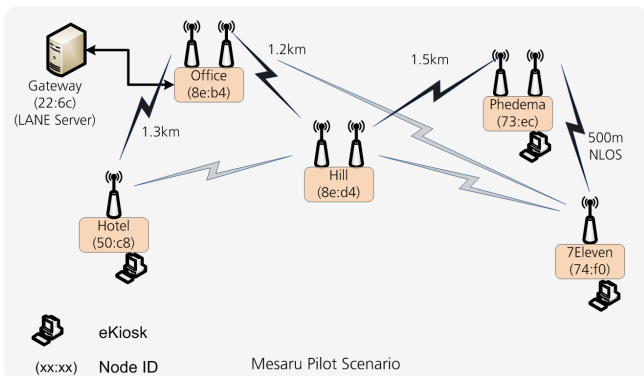


Figure 2: The back-hauling pilot in Maseru, Lesotho consists of five outdoor WiBACK nodes and one indoor node acting as the WiBACK controller and GW node.

Our WiBACK architecture is based on a centrally managed cross-layer concept which builds on a set of IEEE 802.21-inspired command and event services and hardware abstraction as well as technology independent MPLS-based Traffic Engineering (TE) and a model to address po-

tentially shared wireless channel resources. In the WiBACK architecture, MPLS Label-Switched Paths (LSPs) are associated with per-hop resource allocation and referred to as *Pipes*. These are used as aggregates providing resource isolation among traffic classes as well as individual *Pipes* of the same traffic class. Building upon proven TE concepts and protocols, the WiBACK architecture strives to be considered as an alternative for a rather statically configured and over-provisioned operator back-haul network. It must, therefore, meet similarly strict requirements such as guaranteed QoS differentiation, high availability and predictable behavior in high load situations in order to support the provisioning of the triple-play service mix today's customers expect. Thus, to manage such heterogeneous WiBACK networks, a Topology Management Function (TMF) is required to facilitate proper resource descriptions, reliable node discovery as well as association. Furthermore, it must provide a framework for topology forming and maintenance, provide a database of node, interface and link properties as well as available wireless spectrum resources and provide access to *live* link monitoring statistics. This information should allow for different topology optimization strategies to be utilized depending on the intended optimization goals, such as highest reliability, highest capacity or lowest energy consumption. Moreover, to support TE-based capacity management, the TMF must work in close cooperation with the Capacity Management Function (CMF) (see also Figure 4), which is tasked with managing the capacity of the links activated by the TMF by assigning capacity to *Pipes* or 1-to-N multicast Trees based on capacity requests from AP nodes. The details of the CMF are outside the scope of this article, which, accordingly, focuses on the presentation of our TMF design for the WiBACK architecture.

The rest of this article is structured as follows. First, we discuss related work and compare it against the TMF requirements, followed by a summary of relevant background information on the WiBACK architecture. We then present our approach of a centralized TMF followed by a functional validation and evaluation results obtained in real and emulated scenarios. Concluding, we will summarize our contribution and give an outlook on future work.

## 2. Related Work

Topology discovery in WMNs is typically handled at the *Network Layer* by protocols such as Optimised Link State Routing (OLSR), batman [34], Dynamic Source Routing Protocol (DSR) [15] or Ad hoc On-Demand Distance Vector Routing Protocol (AODV) [31] or by, for example, batman-adv [29] or IEEE 802.11s at the *Data Link* layer, with the latter typically relying on a reactive distance ad-hoc vector routing protocol. With the exception of 802.11s, such protocols are unaware of the underlying wireless hardware properties. Having their history in fixed wired routing protocols, such protocols do not support

topology forming via, for example, channel selection or transmit power adaptation, and therefore work under the assumption that the wireless interfaces have already been configured and that connectivity has been established, either manually or via a separate mechanism. Topology discovery, link monitoring and route computation are implemented in one monolithic protocol, while capacity allocations, monitoring or enforcement thereof cannot be supported conceptually. Moreover, hot-standby backup paths [30], which are often used in TE-based networks to support fast fail-overs in cases of link or node failures are not supported.

Traffic Engineering (TE) is concerned with performance optimization of operational networks with the goal to achieve efficient and reliable network operations while simultaneously optimizing network resource utilization [3]. Compared to a typical rather statically configured operator network, a major difference of the WiBACK architecture is the TMF which is tasked to detect and configure WiBACK nodes and their radio interfaces in order to autonomously form and maintain a meshed wireless back-haul network among all participating nodes. The path and capacity computation or path setup-related aspects are mostly covered by the IETF Generalized Multiprotocol Label Switching (GMPLS) protocol suite [6] or the Path Computation Element (PCE)[11] architecture. However, the main TMF tasks, such as physical resource description, identifier management, initial node configuration, dynamic spectrum and channel management as well as wireless topology forming are not addressed, especially in cases where UDTs are to be considered. Conceptually, TE considers links as unidirectional resources, where a bi-directional link is represented as a pair of unidirectional links. Following this concept and contrary to typical *Network Layer* protocols such as AODV, Link Layer Tunneling Mechanism (LLTM) [9], or Bidirectional Routing Abstraction [32], WiBACK can readily describe and utilize UDTs instead of black listing or tunneling around them. UDT support is addressed in detail in subsection 2.1.

Research on MR-WMNs focuses mainly on routing metrics and optimization algorithms for IEEE 802.11-based systems in order to increase the network capacity by, for example, reducing inter-channel interference [35] or by avoiding busy channels and external interferences, see [2], [17] or [27]. Both centralized and decentralized schemes have been discussed and, for example, in [37] the impact of packet loss and queuing delays on QoS-support is considered, while, conceptually, per-path capacity allocations are not supported. In [25] an IEEE 802.11-specific multi-radio *Infrastructure*-mode approach has been implemented which forms a mesh rooted at a gateway node. This approach supports node-local channel assignments to minimize interferences, but does not allow to adjust the Tx-Power nor the coverage class. To the best of our knowledge, no heterogeneous approach to facilitate topology forming and maintenance supporting link optimization for multi-radio multi-channel networks including UDTs has been

proposed.

To decide on a centralized or distributed management approach for the heterogeneous carrier-grade MR-WMN, the CARMEN project has studied two approaches, a distributed approach similar to Open Shortest Path First - Traffic Engineering (OSPF-TE)[16] and a centralized stateful approach comparable to the PCE[11] architecture. The goal was to determine the impact of frequent link state fluctuations as well as QoS reservation changes on the protocol's coherency. Since wireless links are suffering from more volatile conditions compared to traditional wired networks, this may quickly lead to inconsistencies in distributed, i.e. OSPF-TE link state tables, see [7] for a more detailed discussion. Therefore, the centralized network management approach was chosen, where most of the control plane communication is only exchanged between *management nodes* hosting TMF *Masters* and their *Slaves*. Hence, as opposed to distributed WMN protocols, only the tables at the *Master* nodes must be kept up to date, which can be accomplished using the threshold-aware IEEE 802.21 *Event Service* via the *management pipes*. To avoid a single point of failure in the network, *Master* instances may be replicated.

### 2.1. Unidirectional Technologies

In [22], we have discussed in detail how UDTs can be integrated into the WiBACK architecture so that they can be utilized when beneficial to achieve network-wide optimization goals while exploiting the full lower layer information provided by the Abstract Interface (AI), see section 3.1. This aspect concerns the description of radio interfaces, link monitoring, topology management, pipe signaling and, to a lesser degree, path or multicast tree computation.

As a minimum requirement, a WN would have to be equipped with at least one receive-only and at least one transmit-only radio in order to associate with a WiBACK network. In most typical use cases, however, a WN will most likely be equipped with at least one bidirectional radio. Additionally, unidirectional interfaces may be available. To properly support heterogeneous interfaces, including UDTs, WiBACK relies upon the IEEE 802.21 *InterfaceId* which can describe *LinkTypes*. For a technology to be integrated into the WiBACK architecture the technology must be able to send a frame from node *A* to node *B* while node *B* must be able to identify *A* as the sender, an vice-versa in the case of bidirectional technologies. In the case of DVB-T this can be accomplished by relying on Multi Protocol Encapsulation (MPE), while DVB-T2 can be supported via Generic Stream Encapsulation (GSE).

The knowledge about the directionality of the underlying technology can also be exploited by the TMF in order to detect possibly malfunctioning bidirectional links which may occur due to inconsistent radio configuration or physical layer issues. Such inconsistencies have been investigated in-depth [38] for *Ad-Hoc* technologies when

radios are operated within the so-called *Transitional Region*. They may occur due to, e.g. asymmetric transmit power settings, local interferences or a drop in signal strength after channel reassignments due to, i.e. higher cable attenuation or lower antenna gain. *Infrastructure-mode* technologies do not expose this issue since they only operate in fully bidirectional connected mode and manage cell membership internally.

Supporting WNs with only unidirectional interfaces requires substantial protocol support during the topology discovery phase. We have shown in [19] that such WNs can be supported by implementing an *inclusive cycle*-based [5] detection mechanism in order to determine a path, possibly involving neighboring WNs, to establish bidirectional control connectivity with the TMF *Master*. In [5] the authors have shown that pre-configured UDTs can be supported in link state routing protocols as long as an inclusive cycle exists. Following this approach, a TMF *Master* would have to determine *loops* via neighboring WNs, see the bottom part of Figure 6.

Supporting UDTs in addition to bidirectional interfaces, would require the TMF to be able to detect UDT and to properly configure them to form Unidirectional Links (UDLs) or broadcast cells. UDLs would be ignored when setting up the initial management connectivity and would then be made available to CMF for regular capacity allocations.

WiBACK thus provides two major advantages over traditional WMN protocols such as AODV or DSR when addressing UDTs. It can clearly distinguish between unidirectional and bidirectional technologies and can therefore a) seamlessly integrate UDTs into its topology so that their benefits for multicast traffic distribution can be exploited, and b) reliably detect malfunctioning bidirectional technologies avoiding topology fluctuations or unpredictable link behavior.

## 2.2. Requirements

The WiBACK architecture requires a broader approach regarding a suitable Topology Management Function compared to existing solutions, which cover aspects of the TMF functionality, but cannot easily be integrated or extended to fulfill the set of requirements, as listed below:

- Detection, description and configuration of wireless node capabilities
- Detection, description and configuration of heterogeneous wireless interfaces in an abstract way
- Integrated support for Unidirectional Technology
- Protocol support for node discovery, association, link calibration, failure detection and management
- Support for different centralized or decentralized optimization strategies and configurable goals such as high resilience, high throughput or lower energy footprint
- Close interaction with Capacity Management Function and timely maintenance of optimal subset of active links

## 3. WiBACK Architecture

In this section we provide a summary of the WiBACK architecture, highlight some design decisions and introduce the underlying concepts our TMF design is built upon.

### 3.1. IEEE 802.21 beyond Handover Signalling

Support for heterogeneous technologies in the WiBACK architecture is inspired by the mechanisms and functionality specified by the IEEE 802.21 standard. Although IEEE 802.21 actually standardizes protocols and procedures which aim at enabling seamless hand-overs of Mobile Terminals (MTs) between heterogeneous access networks, the basic mechanisms and concepts can be easily extended for other purposes as well [28]. Of particular interest for WiBACK is the concept of media-abstraction used in IEEE 802.21, which hides the technology specific details from the upper layers by providing uniform interfaces in the form of a common set of command, event and information messages. This allows for modularity and a media-independent design of the higher layers, while leveraging the knowledge about the lower layers. IEEE 802.21 also defines a media independent messaging service, which can utilize various transport mechanisms. We have therefore designed the WiBACK architecture leveraging the general IEEE 802.21 concepts, introducing new mesh management related primitives or messaging service extensions where needed [33].

Figure 3 depicts the WiBACK Interface Management Function (IMF) including the Abstract Interface (AI) which extend the IEEE 802.21 Media Independent Handover Function (MIHF) with specific primitives to manage heterogeneous wireless networks. A subset of these primitives has been discussed within the IEEE 802.21b working group and is now included in the IEEE 802.21b standard. Our extensions provide a single Abstract Interface for enabling Mobile Terminal hand-overs on AP nodes as well as forming and managing heterogeneous wireless networks. Technology-specific *MAC Adapters* provide the interface layer between the AI and the specific technology and also provide access to passive per-packet measurement statistics which are updated periodically. IMF user modules located above the AI such as the Topology Management Function (TMF), Capacity Management Function (CMF), Pipe Management Function (PMF), Statistics Function (SF) and the optional Terminal Control Function (TCF) combined implement the WiBACK control plane.

#### 3.1.1. Abstract Interface

The AI extends the IEEE 802.21 media abstraction concept with primitives to manage heterogeneous network interfaces, see Table 1. The functionality provided by such primitives covers key functional blocks and has been chosen coarse enough so that the complete or part of this functionality can be mapped onto functionality provided by the underlying hardware, while additional functionality can be provided by the technology-specific MAC Adapters. The

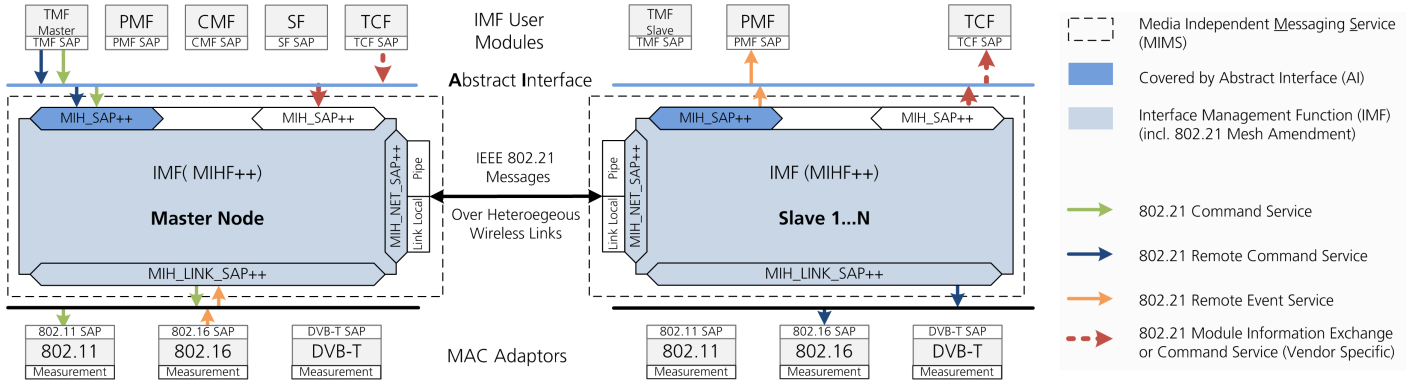


Figure 3: IMf extends the IEEE 802.21 MIHF with an Abstract Interface and Module-to-Module Communication

goal of the WiBACK AI is to allow IMf User modules, such as the TMf, to manage the underlying technology according to a network wide optimization criterion, while leveraging technology specific optimization on a per-link level. For example, the TMf might specify the channel in terms of center frequency and bandwidth as well as the maximum TxPower level for a given link, while the underlying technology might choose any technology-specific features to provide an optimal link within the boundaries set forth by the TMf. This allows the TMf to perform abstract channel scanning, radio planning or optimization as well as channel assignment while leveraging the full knowledge of the technology about how to best perform within the given link budget.

Primitive Name	Description
AI_RadioGetProperties	Returns interface properties
AI_RadioJoinCell	Instructs interface to join a cell
AI_RadioLeaveCell	Instructs interface to leave a cell
AI_LinkCalibrate	Instructs interface to calibrate a link
AI_RadioSetupBeacon	Parameterizes the WiBACK beacon
AI_RadioBeaconScan	Triggers a beacon scan
AI_RadioChannelScan	Triggers a channel scan
AI_LinkDown	Indicates a LINK_DOWN
AI_PipeDown	Indicates a PIPE_DOWN

Table 1: AI primitives used by the TMf to manage heterogeneous wireless interfaces

### 3.2. Control Plane and Signalling

In the centralized WiBACK architecture, reliable and resilient connectivity to the *Master* nodes is essential, but WN-to-WN communication is not required for management purposes. Dedicated LSPs, so-called *Management Pipes* are configured to establish this management connectivity. Optionally, they may be protected with backup LSPs as specified with the MPLS Fast Reroute (FRR) extension to increase the resilience to intermediate node or link failures [21], or to cover periods of channel reconfigurations on the primary LSP. Such a fail-over can be signaled via IEEE 802.21 event messages which are sent directly to the Point of Local Repair (PLR).

Since the TMf is a logically centralized entity, only one *Master* instance is required to manage an entire WiBACK network or an area thereof. However, the design allows for running replicated instances in, for example, hot-standby mode, to avoid a single-point-of-failure in the network. A major advantage of the centralized approach is that it allows the TMf to consider network-wide optimization goals when assigning the available radio spectrum, thus optimally utilizing the available WNs, their capabilities and the current wireless spectrum resources. Additionally, the majority of WNs merely acts as relatively simple MPLS forwarding nodes with an integrated LSP and link monitoring component. Considering rural environments without access to a power grid, this allows for low-power hardware designs which can easily be powered via alternative sources such as solar and wind.

The Pipe Management Function (PMf) implements Resource ReSerVation Protocol - Traffic Engineering (RSVP-TE)-style LSP setup and tear down signalling via source-routed IMf messages and, in addition, allows for the resource allocations of existing *Pipes* to be altered [20]. Both regular downstream-assigned and upstream-assigned multicast LSPs are supported [10].

### 3.3. Passive Monitoring

The WiBACK architecture relies mainly on passive receiver-side monitoring [21] to monitor active wireless links and *Pipes*. This information can be queried from the respective *MAC Adaptor* or be delivered via the IEEE 802.21 event service, either periodically or when a specified threshold is crossed. Based on this information, events such as LINK\_DOWN, PIPE\_DOWN, NEW\_NEIGHBOR or PIPE\_QOS\_VIOLATION can be generated.

WiBACK *Pipes* may be configured to generate heartbeats on their ingress nodes which allow the intermediate and especially the egress nodes to determine stale *Pipes*. To avoid unnecessary traffic, heartbeat frames are only generated if no payload has been sent within a *Pipe*'s heartbeat interval. Heartbeats are by default enabled for *Management Pipes* and the TMf relies on the PIPE\_DOWN event to indicate broken management *Pipes* between the *Master* and its *Slaves*.

### 3.4. Data Plane and Capacity Management

Leveraging proven protocols where applicable, the WiBACK architecture clearly separates the GMPLS and PCE-inspired capacity management task, performed by the CMF, from topology and physical resource management, handled by the TMF. Therefore, a WiBACK network is managed on two time scales. On a slower time scale, the centralized TMF manages nodes, radio interfaces and spectrum resources in order to establish an optimal set of *logical links* among the participating WNs following one or multiple optimization goals. At a faster time scale the CMF assigns the capacity available on those *logical links* in the form of *Pipes* to resource requests for user payload among *Edge Nodes*, see Figure 4. The CMF independently monitors its *Pipes* to quickly react on quality degradation either pro-actively via MPLS FRR [30] backups or reactively by setting up alternative *Pipes*.

### 3.5. Summary

The heterogeneous WiBACK architecture follows a novel concept for MR-WMN management by combining IEEE 802.21-inspired hardware abstraction and TE-based capacity management to provide a carrier-grade meshed backhaul solution. For the TMF to support dynamic network self-configuration, optimization and fast recovery from node or links failures, we have argued why a centralized master/slave-based TMF approach is required which performs topology forming and maintenance, provides mechanisms to support radio configuration as well as link optimization and maintains a close interaction with the TE-based CMF.

## 4. Topology Management Function integrating Unidirectional Technology

The centralized TMF is designed as an IMF user module implementing a master/slave model and introduces a set of primitives to facilitate the communication among the TMF entities. TMF *Masters* are located at *management nodes* while TMF *Slaves* are instantiated at each WN (Figure 4).

A TMF *Master* manages all possible physical links among nodes within its administrative domain, providing a framework to utilize a variety of optimization strategies depending on the specific use case. For this initial TMF validation, we have implemented a local optimization mechanism to form an interference-free meshed topology of point-to-point links out of all possible physical links among multi-radio WNs, see section 4.8. This subset, the so-called *logical links* may include 1-to-N broadcast cells and is exposed to the CMF which only operates on this subset.

In order to allow the CMF to perform constraint based path computation, the TMF *Master* describes the logical links with attributes, such as nominal bandwidth and link latency, as determined by the `AILink.Calibrate` primitive, see section 4.5. When topology changes occur, the

TMF *Master* checks if the current logical configuration or any of its management connections are affected. In such a case it will inform CMF that the affected links will be taken off-line and then attempt to reconfigure the available radio interfaces of the affected WNs, by either issuing an `AILink.Calibrate` request or by assigning a different channel or a new peer.

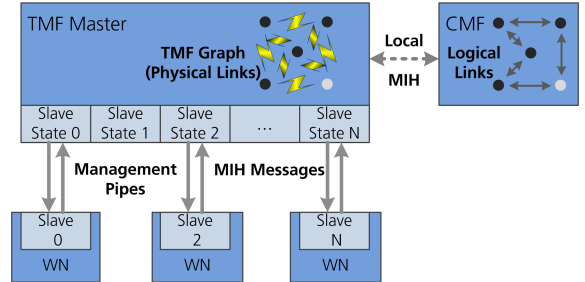


Figure 4: TMF uses a Master/Slave model and communicates via MIH messages through so-called *Management Pipes*. The optimal subset of assigned *logical links* is made available to CMF

In the following subsections we first introduce the key functional components of the TMF, followed by the description of the ring-based master/slave approach, our initial optimization scheme and a protocol scalability analysis.

### 4.1. Identifiers

The WiBACK architecture uses different identifiers for its resources, such as a `NodeId`, an `InterfaceId`, a `LinkId` and a `PipeId`. The first three identifiers have been taken from the IEEE 802.21 standard, with the `NodeId` being the equivalent of the `MIHFid`. `InterfaceIds` are generated from the hardware address of each interface and a `LinkId` uniquely identifies a link between two interfaces. `NodeId`s and `InterfaceIds` are considered to be unique across a WiBACK network and TMF *Masters* will verify this upon an association of a *Slave* and reject *Slaves* with non-unique identifiers.

Parameter	Type	Description
<code>InterfaceId</code>	<code>INTERFACE.ID</code>	MIH InterfaceId
<code>State</code>	<code>ENUMERATION</code>	UP, DOWN, ...
<code>Channels</code>	<code>LIST(CHANNEL)</code>	Frequency, Bandwidth
<code>TxPower</code>	<code>LIST(TXPOWER)</code>	TxPower Levels
<code>Directionality</code>	<code>ENUMERATION</code>	Rx, Tx, Duplex

Table 2: TMF describes Interfaces using the `TMFInterface` object

During the WN bootstrap period, the `NodeId` is derived from the `InterfaceIds` of the local Interfaces by hashing the link layer addresses. The `InterfaceId` contains a `LinkType` field which indicates the underlying technology as well as an `AddressFamily` field characterizing the hardware address type. Hence, potentially any kind of hardware address can be supported while the `LinkType` can be used to indicate hardware-specific features. In the WiBACK context, the `LinkType` is used to determine if a physical

Parameter	Type	Description
NodeId	MIHF_ID	Node ID
State	ENUMERATION	Discovered, Operational
NetworkId	UNSIGNED	ID of WiBACK network
Distance	UNSIGNED	Hop Distance to Master
TimeStamp	TIMESTAMP	Node creation time
Coordinates	COORDINATES	GPS Coordinates
Interfaces	LIST(TMFIInterface)	Local Interfaces

Table 3: TMF describes Nodes using the *TMFNode* object

Parameter	Type	Description
LinkId	MIHLINK_ID	MIH LinkId
State	ENUMERATION	Discovered, Operational
Properties	LINK_PROPERTIES	Bandwidth, Latency

Table 4: TMF describes Links using the *TMFLink* object

link is bidirectional or unidirectional. For example, *Ethernet*, *IEEE\_80211* or *IEEE\_80216* are considered to provide bidirectional connectivity while *DVB* or *ATSC* are considered to provide only unidirectional connectivity.

An MIH LinkId consists of an InterfaceId describing the source and a *LinkAddress* describing the destination which must be of the same *LinkType*. Hence a LinkId also indicates if its underlying technology provides bidirectional or unidirectional connectivity. Since WiBACK considers links as unidirectional resources, physical connectivity provided by a bidirectional technology is represented by a pair of LinkIds. During operation, TMF will verify that such a bidirectional connectivity for bidirectional technologies exists, otherwise the affected link pair will be marked as *FLAKY* and not considered for traffic forwarding until it has been reconfigured and bidirectional connectivity could be verified.

WiBACK introduces a *PipeId* as an additional identifier for its management and data *Pipes*. It consists of the ingress NodeId and an integer descriptor generated at the ingress node, thus the *PipeId* is a network-wide unique identifier of a pipe. A pipe consist of a primary LSP, but may have additional backup LSPs or refer to CMF-organized multicast 1-to-N trees.

The TMF maintains the topology representation at *Master* nodes in a *TMFGraph*, storing *TMFNodes* as vertices which may hold multiple *TMFInterfaces* while *TMFLinks* are represented as edges, see Tables 2, 3 and 4. This information is made available via internal and external interfaces to allow different spectrum or topology optimization algorithms to be deployed.

#### 4.2. WiBACK Beacon

The WiBACK beacon is used to periodically broadcast information about a WN's state within the broadcast domain of its Tx-capable interfaces. Similar mechanism are used, for example, by IEEE 802.11 or DVB which broadcast information about their cell using beacon frames or information tables. Hence, to avoid the overhead of additional periodic transmissions, the respective *MAC Adaptor* will, where possible, extend such management

frames, with WiBACK specific information, the so-called WiBACK beacon. If such mechanisms do not exist, the WiBACK beacon may be sent as a regular link-local multicast or broadcast frame. The specifics are implemented in the respective *MAC Adaptor* and the TMF may control the WiBACK beacon content via the *AI\_RadioSetupBeacon* primitive (Table 5):

Parameter	Type	Description
NetworkId	UNSIGNED	ID of WiBACK network
MasterId	MIHF_ID	TMF master ID
MasterTStamp	TIMESTAMP	TMF Master time stamp
SenderId	MIHF_ID	ID of sendign node
Coordinates	COORDINATES	GPS Coordinates
Distance	UNSIGNED	Hop distance to Master

Table 5: Parameters of the *AI\_RadioSetupBeacon* primitive

WiBACK beacons are sent by associated WNs to provide initial information for freshly joining WNs. For example, if such a WN receives beacons from multiple associated WN it may use the hop distance and the signal quality the beacon was received at, as a criterion to choose the most suitable WN in order to associate with a WiBACK network. WiBACK beacons are also interpreted by the neighboring WN's monitoring component which are using them as a heartbeat signal and to differentiate between WNs and interfering non-WiBACK transmitters. The *NetworkId* is used to separate multiple WiBACK networks. A TMF *Master* only accepts *Slaves* configured with its own *NetworkId*.

#### 4.3. BeaconScan Procedure

The *BeaconScan* procedure aims at detecting all possible physical connectivity among the scanning WN and its neighboring WNs in order to allow the TMF to choose the optimal links among all possible options. Typically, the *BeaconScan* procedure is executed by associating TMF *Slaves* on all or a selection of their Rx-capable interfaces. Alternatively, it may be triggered by the *Master* to obtain an up-to-date view of the physical connectivity among neighboring WNs and the scanning WN.

This set of neighboring WNs serves two purposes. It may be sorted by signal quality and hop distance of the sending node in order to determine the most suitable proxy WN to associate with a *Master*. During the association procedure this set is also shared with the *Master* which examines it in order to determine all possible physical connectivity of the associating WNs. Based on this information, the *Master* may determine an optimal alternative link or WN for the on-going association.

Especially in black-out scenarios, multiple unassociated neighboring WNs may be executing the *BeaconScan* procedure simultaneously, however, without coordination. Since unassociated WNs do not send regular WiBACK beacons, they would not be able to discover each other. Therefore the *BeaconScan* procedure is split into two stages. The first stage performs a so-called *wellknown-channel-scan* where the channel is determined independently for

each technology and is chosen as the lowest supported channel in the current regulatory domain. Alternatively, it may be pre-configured. In this mode, the beacon scanner periodically broadcasts WiBACK beacons with the *MasterId* set to *None* while at the same time interpreting beacons received from neighboring WN. The second stage performs a passive *all-channel-scan* for WiBACK beacons on all channels supported by the respective interfaces. Those two scanning procedures are alternated. To ensure that the independently running WNs overlap during the *well-known-channel-scan* period, this period must be longer than the *all-channel-scan* period (Figure 5).

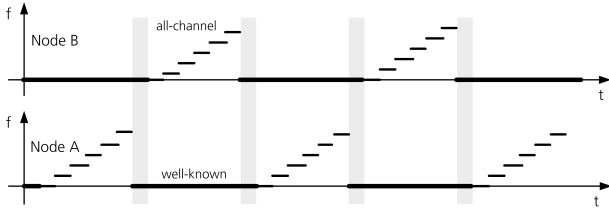


Figure 5: TMF slaves execute the *BeaconScan* procedure to detect all possible physical connectivity to neighboring WNs by alternating between the *well-known-channel* and the *all-channel* scan.

#### 4.4. Channel Scanning and Assignment

The *ChannelScan* procedure is executed by WNs in order to assess the current channel utilization on a specific or all rx-capable interfaces. This information is typically used as input for a local or network-wide radio planning or optimization function.

In local optimization mode, a WN may attempt to assign the least utilized channels while maintaining a configurable channel separation among node-local interfaces to minimize inter-radio cross-talk.

In network-wide optimization mode, the *Master* might poll some or all its *Slaves* and perform channel (re-)assignments following administered optimization goals. To push such optimization results back into the network, the order of the reconfigurations should be carefully considered since channel reassignments may require link calibrations which may cause considerable connectivity interruptions.

Hence, the involved TMF instances must be aware that channel reconfigurations may fail due to, for example, communication issues on the new channel. After a timeout they should revert back to the old configuration. As a last resort, if the *Slave's* management pipes are affected, that *Slave* might be required to begin a new association procedure, which might affect other WNs connected via this failed *Slave*.

#### 4.5. Link Calibration

The *AllLinkCalibrate* primitive may be used to trigger the calibration of an interface for a link to a neighboring WN. Typically, the *Master* will trigger a link calibration upon establishment of a new link or in an attempt to repair

a link if, for example, the monitoring subsystem reports a drop of the link quality below a configured threshold.

This primitive is defined sufficiently coarse to allow a mapping onto already existing mechanisms of certain wireless technologies, such as, IEEE 802.16. For IEEE 802.11, the *MAC Adaptor* may support, for example, the calibration of the Modulation and Coding Scheme (MCS) configuration, the actual TxPower level as well as the coverage class to optimize the throughput on long distance links. The required messaging is handled directly among the *MAC Adaptors* via media specific communication.

Parameter	Type	Description
LinkId	MIHLINKID	Link to be calibrated
MaxTxPower	UNSIGNED	Max. TxPower level
Distance	UNSIGNED	Distance, if known
Mode	ENUMERATION	Capacity, Robustness

Table 6: Parameters of the *AllLinkCalibrate* primitive

The radio planning or optimization function may have determined a maximum TxPower level to be used for the given LinkId. If supported by the underlying technology, the respective *MAC Adaptor* is free to assign any TxPower level up to the specified maximum in order to optimally configure the given link. The distance may optionally be specified, if known. Otherwise the *MAC Adaptor* should, where required, determine the proper distance or range setting via technology means. The Mode parameter determines if the calibration should be rather aggressive and prefer capacity over robustness or vice versa.

Once the link calibration procedure has completed, the *MAC Adaptor* computes the logical parameters for this link in terms of bandwidth in kbps and typical latency in milliseconds, taking into account the current MAC scheduler type as well as its parameterization. This information is then stored in the *TMFLink* object and pushed to the CMF as a basis for its resource allocations and book-keeping. Capacity management is outside of scope of this paper.

In most cases, UDTs can be assumed to be statically configured and per-link calibrations would not be possible. However, with the availability of e.g. DVB-S2 or DVB-T2, receiver specific optimizations are supported in Advanced Coding and Modulation (ACM) mode. Since typically a return channel is required during the calibration phase, link calibration of UDTs cannot readily be supported by the *AllLinkCalibrate* primitive. Hence, to facilitate UDT link calibrations, a specific IMF user module could be provided to orchestrate the calibration messaging, or a dedicated pipe could be configured to provide a return channel and therefore temporarily provide direct bidirectional connectivity among the two affected interfaces and their *MAC Adaptors*.

#### 4.6. Ring-based Master/Slave Approach

TMF is executed as a continuous process aiming at forming and maintaining a meshed network of multi-radio



nodes according to its configured optimization goal. Management nodes executing a TMF *Master* instance are either assumed to be designated manually as a result of off-line network planning or may be elected at run-time. The criterion to qualify as a TMF *master* node may be the availability of sufficient resources to execute the TMF functionality or the presence of back-bone connectivity (i.e. Fiber, DSL), since in many practical, and especially rural use cases, a node with back-bone connectivity can be assumed to be operated more reliably. In the cases of network partitioning, other leader election mechanisms may be used to provide connectivity within the network partitions.

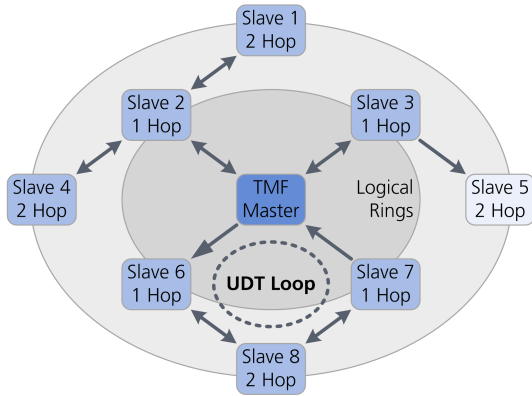


Figure 6: The *Master* node discovers new WNs by forming logical rings based on the hop distance.

The initial discovery of new WNs is performed by forming logical rings around a *Master*, see Figure 6. The first ring consists of WNs with direct radio associations. WNs belonging to the next outer rings aim at associating with nodes on the inner rings. WNs discover associated neighbors by passively scanning all available channels on their Rx-capable interfaces for WiBACK beacons. If the BeaconScan procedure reports a *Master* or already associated WNs the associating WN will attempt to associate through them with the *Master*. Especially after larger network outages, careful consideration is required to avoid storms of association requests, since depending on node distribution and connectivity, the number of potentially newly discovered WNs may increase exponentially with the hop distance from the *Master*. Larger bursts of association requests may overload the *Master* which might perform rather complex topology optimizations taking into account the new connectivity options provided by the associating WN. Links in the vicinity of the *Master* may experience additional load bursts since an association requires increased signalling to establish *Pipes* and to update neighbor information.

To distribute the association requests, we have applied an exponentially increasing randomized back-off timer depending on the hop distance  $d$  of the associating WN. To limit the growth of the maximum back-off time for the maximum WiBACK-supported 10-hop use case, we are us-

ing a *tamed* exponential back-off function:

$$MaxBackoff(d) = \frac{2^d}{(d+1)^2} * C \quad (1)$$

The constant  $C$  determines the range of the random back-off timer with smaller value of  $C$  yielding a tighter timing and thus shorter discovery times but possibly introducing an overload at *Master* nodes. Thus, the value of  $C$  should be adjusted to match the average node density, available CPU resources at the master node and complexity of the topology optimization algorithms. As a future work item we plan to determine optimal values of  $C$  for typical scenarios and to allow for an auto-adjustment by propagating the *Master* load via its WiBACK beacons.

#### 4.7. Association Procedure

The following subsections describe the association procedure from the *Master's* (section 4.7.1) and the *Slave's* (section 4.7.2) point of view.

##### 4.7.1. TMF Master procedures

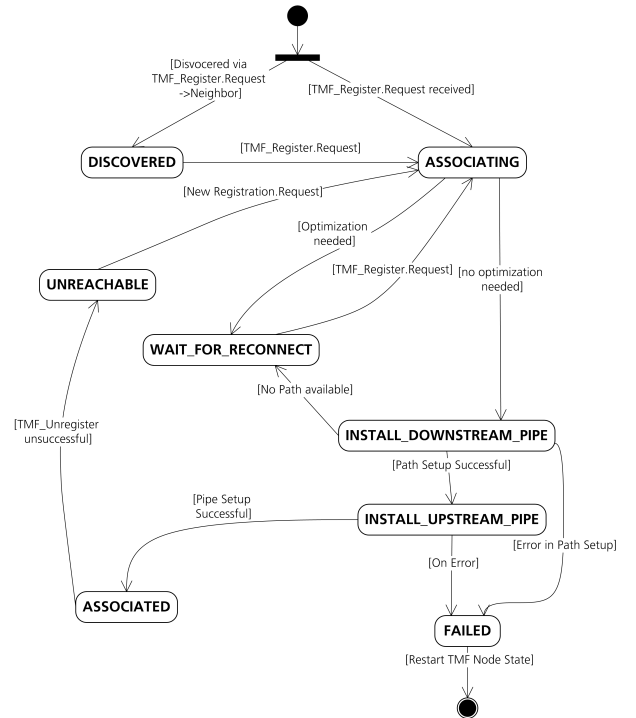


Figure 7: Each *Slave* node is managed via its own *SlaveState* object and an associated state machine at the *Master*

First, a *Master* performs a channel scan on all available Rx-capable interfaces in order to determine the least utilized channels and then determines and configures the optimal channels for its radio interfaces. Finally, it sets up the WiBACK beacon which in turn starts the topology discovery process.

To manage its *Slaves*, the *Master* maintains a *SlaveState* object for each discovered WN which keeps track of

the WN’s state. New *SlaveStates* are either directly created upon the reception of a *TMF\_LinkRegister* association request from the new WN or indirectly by parsing the neighbor information contained in the *TMF\_LinkRegister* request. The state machine of the *SlaveState* object is depicted in Figure 7. A *TMF\_LinkRegister* request is either sent directly to a *Master*, or, if no direct radio association exists, to an already associated WN. This WN will then act as a proxy and forward the request message to the *Master* through its own management pipe. On the first hop, this initial request is sent via IMF *path vector* routing [20] in order to specify the specific link to be used for the association.

Upon reception of a *TMF\_LinkRegister* request, the *Master* will set the links used for the association to *ASSIGNED* state so they can be considered for management pipe computation. Since this link was chosen by the *Slave* solely based on its local knowledge, the master may invoke an optimization algorithm to check for possibly better connectivity options for the associating WN, see section 4.8. If such options are available, the *Slave’s* state is set to *WAIT\_FOR\_RECONNECT* and the association request is rejected. In case of registration rejection depending on the optimization algorithm and its strategy, a set of either black-listed or white-listed links is attached to the reject message to indicate to the *Slave* which links to avoid or to choose for future association attempts. Even in white-listed mode, the *Master* should return multiple options, where available, since the *Slave* may not be able to establish bidirectional connectivity on the specified links due to physical layer issues.

The paths of the management pipes may be computed by applying a Dijkstra search on the topology graph where the metric may consider, for example, hop distance, signal quality, low interference or more complex metrics. The TMF first ensures that it can compute a downstream and an upstream path and then triggers the Pipe Management Function (PMF) to push the downstream pipe state into the network. Upon successful completion, the TMF triggers the PMF to push the upstream pipe state from the associating WN back to the *Master*. Once the pipes have been successfully set up, the new WN is marked as *ASSOCIATED* in the TMF master’s topology representation and a *TMF\_Link\_Register* response is returned indicating a successful association. If an error occurs during the path setup the association process is aborted and the *Slave’s* state is set to *UNREACHABLE*. Such *Slaves* may attempt a new association, possibly via a different link.

#### 4.7.2. TMF Slave procedures

During their initialization phase, *Slaves* at unassociated WNs first detect their own capabilities in terms of number, type and properties of available radio interfaces by calling the `AI_RadioGetProperties` primitive, see Figure 8. Then they continuously execute the BeaconScan procedure on all receive-capable interfaces in order to discover potential *Masters* or associated WNs. Upon each

completed BeaconScan the *Slave* evaluates the collected information by first filtering out Neighbors with mismatching NetworkIds. If, at least, one beacon from an associated WN has been detected, the *Slave* will start the *Association Manager* passing it the filtered set of beacons of associated WNs.

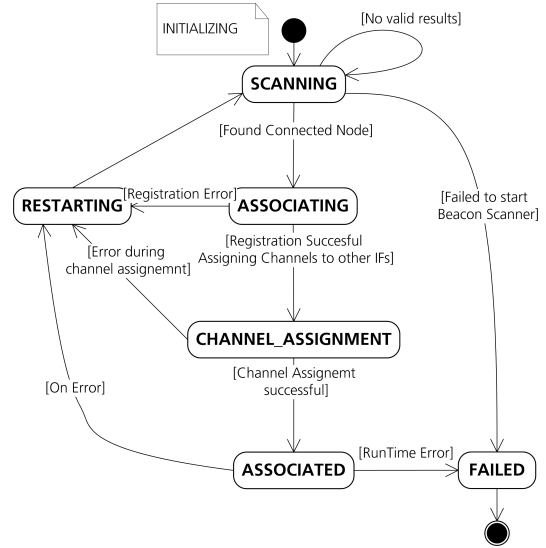


Figure 8: TMF state machine

The *Association Manager* may sort this set according to its local optimization goal, such as bidirectional links first, best signal quality or shortest hop distance and will then begin with the association procedure starting with the highest rated neighbor by sending a *TMF\_Link\_Register* request towards the *Master* or an associated WN which then acts as proxy for this association procedure. This request includes a set of neighbors gathered during the BeaconScan period to allow the *Master* to determine the optimal link for the association. If UDLs are detected during the association process, the *Master* may make them available to CMF. As described in [19] they can optionally be used for management traffic, but for most typical use cases this adds extra complexity and stability issues especially under non-optimal link conditions.

Next, the *Slave’s* additional radios will be configured. The configuration may either be determined and administered locally by e.g. choosing the least utilized channels or remotely by *Master* enforcing a configuration following its network-wide optimization goals. Eventually, the WN enables the transmission of WiBACK beacons and all of its transmit-capable interfaces allowing other WNs to associate with the WiBACK network.

Associations may fail if either the request or response messages are lost e.g. due to link stability issues or because the *Master* has rejected the request for e.g. administrative reasons, due to pipe setup failures or due to the availability of alternative and more suitable connectivity options. In cases of packet loss or pipe setup errors, the *Slave* should proceed with the next possible link. If the *Master* has

rejected the request and provided a set of either black-listed or white-listed links, the *Slave* should start a new association attempt either avoiding the black-listed links or preferring the white-listed links.

In the unlikely event that all attempts avoiding black-listed or using white-listed links fail due to communication errors, the client may force an association to the WiBACK network by setting the *ForceAssociation* flag in the request message. This allows the WN to still join the network, and further optimizations can take place once the WN is associated.

#### 4.8. Local Optimization

To validate our TMF framework we have implemented an optimization algorithm which aims at forming point-to-point links among adjacent multi-radio WNs whenever possible. This optimization is performed during the association procedure of a WN by determining the optimal link for the association. Hence, our algorithm performs a *local* optimization, only affecting the associating node, but taking into account the centralized network-wide topology knowledge regarding already associated nodes and their connectivity options. Since the *local* optimization can leverage network-wide knowledge, it can compute a set of optimal candidate links and therefore uses white-listing to indicate its decisions to the *Slave*. This set of candidate links is ordered according to the *Master's* preference. In high connectivity scenarios multiple WNs might compete for a white-listed link, therefore an association might be rejected and an updated list of white-listed links is returned. The parameter  $R$  controls how often such a rejection is accepted until the WN forces an association, possible by-passing the optimization and tolerating a sub-optimal connection.

This network-wide optimization mechanism affects only the currently associating WN and enables the network to form point-to-point links where available without requiring reconfigurations of established links. Thus, for this initial evaluation, this mechanism avoids issues such as interruptions of established links or even oscillations of the optimization algorithm which might occur when larger parts of the topology are reconfigured due to additional connectivity options introduced by each (re-)associating WN.

#### 4.9. Failure Recovery

Once a *Slave* has associated with a *Master* both sides start monitoring the management pipes for possible connectivity issues by subscribing to `AI_PipeDown` indications on their respective egress pipe. Upon an `AI_PipeDown` event, the WN considers the association broken and closes both pipes from its respective side. This ensures that even in the case of a unidirectional link failure, the peer node will detect the broken association, as well. A *Master* will mark the affected *Slave* as `UNREACHABLE` and attempt to send a `TMF_LinkUnRegister` message towards the *Slave* in order to perform a proper association tear down. The

affected *Slave*, in turn, will attempt to re-associate with the WiBACK network possibly via alternative links or alternative neighboring WNs.

Any WN connected via the failed WN may experience a loss of connectivity either due to the original cause, for example, management pipes on the same broken link, or due to the failed WN switching back in association mode thus dropping all active pipes. Possible backup paths might keep the other WNs connected.

#### 4.10. Analysis

In order to analyze the performance of the TMF's topology discovery phase we consider an error-free case of a black-out scenario, where all WNs are restarted at roughly the same time. Due to the ring-based approach the duration should mainly depend on the number of logical rings formed during the discovery phase. Assuming a fixed duration for a neighborhood scan  $t_{Scan}$ , the lower bound of the discovery time for a  $d$ -hop topology can be expressed as follows, with  $t_{backoffMin}$  denoting an optional minimum back-off time adhered to by *Slaves* before attempting an association.

$$t_{DiscMin}(d) = d \cdot (t_{Scan} + t_{backoffMin}) \quad (2)$$

To determine the upper bound, the exponential back-off function (1) as well as the possible impact of the optimization algorithm must be considered. Hence, the upper bound of the discovery time can be expressed as shown in equation 3, where the factor  $R$  covers the possibility of a maximum of  $R$  rejections during the association procedure due to optimizations which would result in repeated association attempts subject to a new back-off timer period.

$$t_{DiscMax}(d) = d \cdot t_{Scan} + R \cdot \sum_{i=1}^d \frac{2^i}{(i+1)^2} \cdot C * 1s \quad (3)$$

Since the purpose is to estimate performance of the basic TMF functionality, the above formulas do not consider the computational overhead introduced by possibly complex topology optimization or link calibration algorithms. Accordingly, the `AI_LinkCalibrate` primitive was implemented to immediately return the logical link parameters based on the current interface configuration. The signaling overhead of the `TMF_LinkRegister` primitive itself and the PMF signalling required to set up the management pipes is omitted since it is typically in the order of tens of milliseconds while the TMF timing is in the order of seconds.

## 5. Validation and Implementation

TMF has been implemented as a WiBACK IMF module. The topology representation is based on the C++ boost graph library as a *directed multi graph*. The boost graph library allows for graphs to be easily serialized into

*dot* or *GraphML* format in order to visualize the discovered topologies as well as their properties.

Our WiBACK software is built upon our SENF-based low latency NetEMU framework [14] which also provides a real-time network emulator component. This allows us to evaluate the same binary code on emulated or real embedded nodes. For emulated interfaces random packet loss and a fixed link latency can be introduced and the transmission range of an interface is emulated considering the frequency-dependent free space loss as well as a varying transitional zone. The packet delivery latencies introduced by the Linux scheduler and the emulation overhead have been shown to be less than 1ms [20], which is a typical latency for loaded IEEE 802.11 links and therefore should not impact TMF-related measurements where timeouts or back-off timers are rather in the order of seconds.

The unidirectional connectivity provided by the DVB-T cell was emulated at the MPE *layer* with a fixed QPSK<sub>3/4</sub> modulation and coding configuration and a relatively low latency of 10ms. This latency figure has been determined in our outdoor testbed where, from a previous project, we operate a DVB-T transmitter to evaluate back-haul connectivity to a remote farm using the RFC3077 [9] LLTM approach with the return channel being realized via a DSL connection. We’re currently working the technical, but mainly regulatory issues to include the DVB-T cell in our WiBACK testbed to verify the below emulation results in a real network.

### 5.1. Functional Validation

We validated the basic TMF functionality, the discovery and optimized forming of a topology consisting of six emulated WNs where node 2 was equipped with a DVB-T transmitter and the nodes 3, 5 and 6 were equipped with DVB-T receivers. Bidirectional connectivity was provided by IEEE 802.11a interfaces. For all interfaces, omnidirectional antennas were assumed. The TxPower settings were fixed, but partially asymmetric, and have been chosen to exhibit *FLAKY* links. As depicted in Figure 9, the *Master* located at node 1 has successfully discovered and associated all *Slaves* and formed a broadcast cell among the detected DVB-T interfaces. The discovery process took 62 seconds to complete and 89 possible links have been discovered of which 22 were marked as *FLAKY*, among them the potential link d1:37 ↔ ff:c1. The affected radios itself have been reconfigured to establish connectivity with nodes 5 and 3 respectively. The center frequency for the DVB-T cell was fixed at 714MHz, while the frequencies for the IEEE 802.11a links have been assigned by the TMF, maintaining, at least, the configured minimum 60Mhz separation between the center frequencies of node-local interfaces. These results confirm our earlier evaluations on real Linux-based multi-radio nodes [23].

### 5.2. Scalability Evaluation

In order to evaluate the scalability of our ring-based TMF design we considered black-out scenarios for differ-

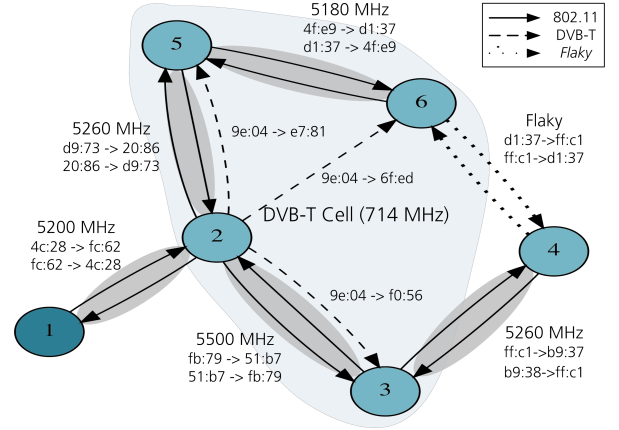


Figure 9: The *dot* output of the graph maintained at the TMF master node 1 shows the discovered and optimized topology including the DVB-T cell and *FLAKY* links.

ent topologies. As the benchmark topology we defined the *chain* topology which consists of 11-nodes geographically arranged as a linear chain with the first and the last node being equipped with one radio each while the intermediate nodes were equipped with two radios. The transmission range was chosen so that each node could communicate with its one and two hop neighbors. As additional topologies we generated two 100-node topologies with dense and sparse inter-node connectivity, with the *sparse* topology resembling a typical WiBACK scenario most closely.

For the above scenarios we first evaluated the TMF topology forming duration per logical-ring, which should be in between the bounds expressed by the terms (2) and (3) defined in the previous section. Since the TMF relies on the IMF and its *AckService* for reliable messaging and as well as PMF for path setup signalling, the obtained results reflect the performance of the combined modules. For the below measurements the following parameterization has been used:  $C = 6s$ ,  $R = 2$ ,  $t_{backoffMin} = 500ms$ ,  $t_{scan} = 5s$  based on a WiBACK beacon interval of 250ms and eight available IEEE 802.11a channels. The *PIPE\_DOWN* detection threshold was set to 5s. All measurements were run 50 times and the averages are shown.

It can be observed in Figure 10 that the topology forming time is within the theoretical bounds for all scenarios and we expect that the number of WNs to be joined per ring has no significant impact on the per-ring discovery times as long as the *Master* node or the links in its vicinity are not overloaded. For the *chain* scenario the *local* optimization formed point-to-point links for all hops, while for the *sparse* and *dense* scenarios about 5% of all *ASSIGNED* links could not be optimized either due to timing issues among competing WNs within the allowed rejection count  $R$  or due to limited connectivity options.

To evaluate the topology forming phase under varying link error conditions, we have chosen the *sparse* scenario as an example for a WiBACK scenario. Figure 12 depicts the results for one, five and ten percent of per-link packet

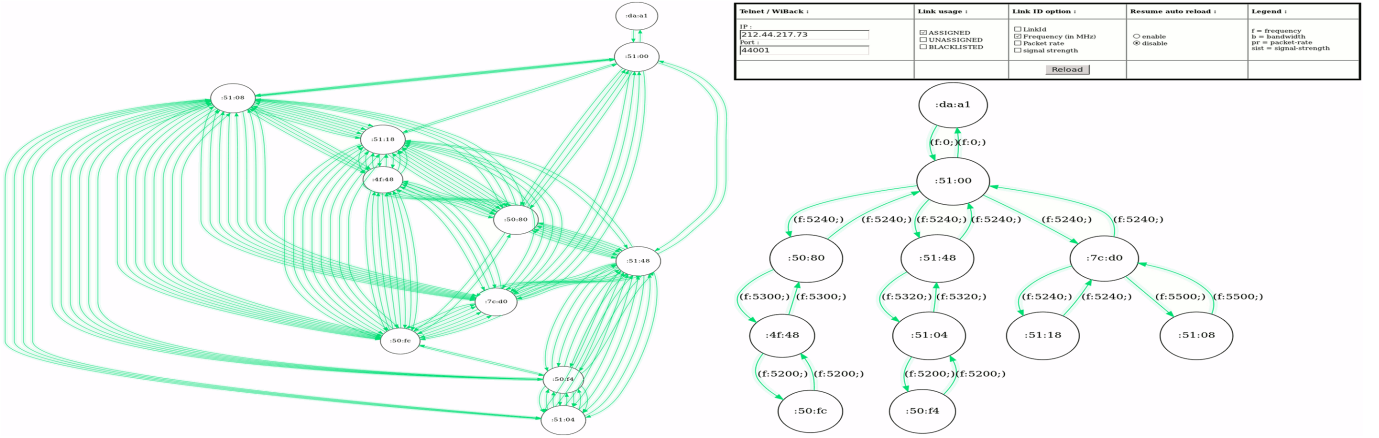


Figure 11: Live snapshots of our 11-node two-radio outdoor testbed showing all possible links (left) and the subset of assigned links (right)

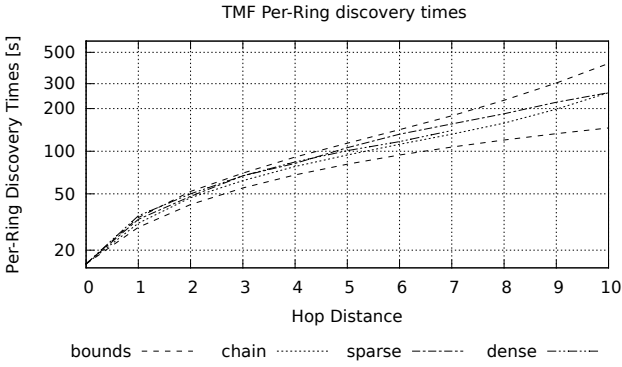


Figure 10: Per-ring topology forming times for the *chain*, *dense* and *sparse* scenarios with up to ten hops

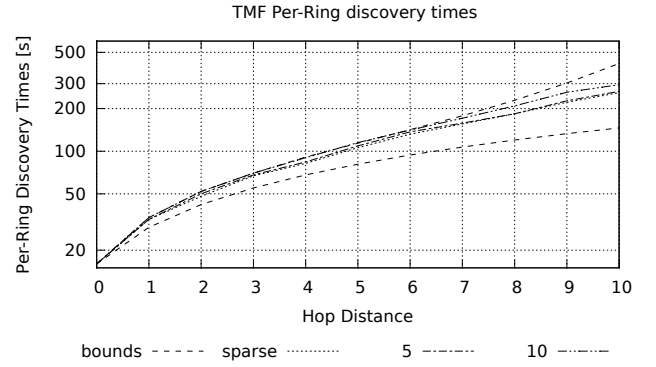


Figure 12: Per-ring topology forming times for the *sparse* scenario under 0% (sparse), 5% and 10% per-link error conditions

loss, which may result in up to 10%, 50% or 100% of end-to-end packet over 10 hops. It can be observed that the WiBACK control plane is relatively robust against these rather high loss figures with the total discovery times only increasing moderately, which shows that the *AckService* as well as the PMF can cope well with the *mersenne twister*-based loss distribution introduced by the emulation. We suspect that larger burst losses, i.e. short link outages, would have a stronger impact. Investigating the resilience of the WiBACK control plane is ongoing work.

Next, we have evaluated the recovery times for the three scenarios in cases of node failures, where the complete topology has been discovered and then the hop distance  $d$  of the failed node from the *Master* was varied from one to ten. Figure 13 depicts the results for the *chain* scenario which yielded constant results. It can be observed that for failures of the first hop node, the failure detection and recovery time is roughly identical to a complete topology discovery time determined above. For nodes farther away from the *Master*, the discovery time decreases. Hence, the recovery procedure also performs within deterministic bounds. The results for the *dense* and *sparse* scenarios varied significantly due to the randomness of the

formed topologies and the number of nodes affected by a node failure and have therefore been omitted. It should be noted that the maximum discovery time was still bound.

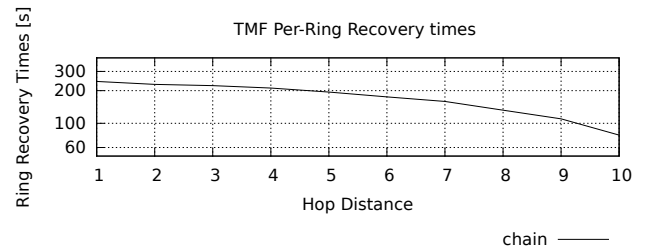


Figure 13: Recovery times after failure of a  $d$ -hop node for the benchmark *chain* scenario

### 5.3. Outdoor Testbed Results

In order to evaluate our TMF approach on real hardware we have deployed the same code on an 11-node outdoor testbed. The testbed is located on the roof of an office building with an area of roughly 20m x 50m in size. Ten nodes are actual low-power outdoor mesh nodes with two IEEE 802.11a radios and an omni-directional antenna

each. The *Master* is hosted on a dedicated server and is connected to one of the mesh nodes via a 100Mbps Ethernet link. The roof is cluttered with air conditioning units, satellite dishes and other equipment. Hence, not all nodes have line-of-sight connectivity and reflections, etc. are to be expected.

Figure 11 depicts snapshots of the *TMFGraph* visualization tool. On the left, all possible physical links are shown, while on the right only the assigned links are shown together with the chosen center frequency. The two links with a center frequency of 0 MHz refer to the Ethernet cable between the *Master* the WN. It can be observed, that, due to the current link conditions, the TMF has mostly configured independent point-to-point links while maintaining a node-local channel separation of, at least, 60 MHz.

## 6. Conclusion and Future Work

We have presented our centralized Topology Management Function (TMF) framework which builds upon the IEEE 802.21 standard to discover, form and maintain a heterogeneous multi-hop WiBACK network. Extensive evaluation results show that the TMF framework supports the forming and maintaining of topologies of WiBACK Nodes with up to ten hops, even under sub-optimal link conditions. Ongoing work focusses on the evaluation of QoS performance, especially compared to traditional WMN protocols. Since such comparisons depend highly on the investigated topologies (i.e. node distribution, number of interfaces, spectrum availability) and traffic patterns (i.e. *best effort*/VoIP ratio and any-to-any vs. AP-to-GW flows), we're currently working on defining realistic benchmark scenarios and Key Performance Indicators (KPIs) to quantify the results.

Building upon the IEEE 802.21 InterfaceId and MIH-LinkId allows the TMF to properly describe and handle UDTs and to differentiate between UDTs and *FLAKY* bidirectional connectivity. Hence, our TMF seamlessly integrates UDTs such as DVB into the WiBACK architecture and therefore supports the convergence towards an *all-IP* back-haul network.

With the basic TMF functionality in place, we plan to focus on an effective network-wide channel assignment strategy to determine the optimal channel configurations while trying to maximize the networks resilience, the overall capacity or the network's energy efficiency.

Other directions include the use of MPLS backup paths to reduce the impact of link or node failures, the investigation of different replication schemes for *Master* entities and the study of least disruptive on-line node and interface reconfiguration schemes.

## 7. Acknowledgment

This work has been funded by the Federal Ministry of Education and Research of the Federal Republic of Ger-

many (Förderkennzeichen 01 BU 1116, SolarMesh - Energieeffizientes, autonomes großflächiges Sprach- und Datenfunknetz mit flacher IP- Architektur). The authors alone are responsible for the content of this paper.

## References

- [1] Akyildiz, I. F., & Wang, X. (2009). *Wireless Mesh Networks*. (1st ed.). John Wiley & Sons Ltd.
- [2] Athanasiou, G., Broustis, I., Korakis, T., & Tassioulas, L. (2009). Routing-aware channel selection in multi-radio mesh networks. In *Communications, 2009. ICC '09. IEEE International Conference on* (pp. 1–6).
- [3] Awduche, D., Chiu, A., Elwalid, A., Widjaja, I., & Xiao, X. (2002). Overview and Principles of Internet Traffic Engineering. RFC 3272 (Informational). Updated by RFC 5462.
- [4] Banchs, A., Bayer, N., Chieng, D., de la Oliva, A., Gloss, B., Kretschmer, M., Murphy, S., Natkaniec, M., & Zdarsky, F. (2008). Carmen: Delivering carrier grade services over wireless mesh networks. In *Proc. IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications PIMRC 2008* (pp. 1–6).
- [5] Bao, L., & Garcia-Luna-Aceves, J. J. (1999). Link-state routing in networks with unidirectional links. In *Proc. Eight International Conference on Computer Communications and Networks* (pp. 358–363).
- [6] Berger, L. (2003). Generalized Multi-Protocol Label Switching (GMPLS) Signaling Functional Description. RFC 3471 (Proposed Standard). Updated by RFCs 4201, 4328, 4872.
- [7] Bernardos, C. J., Fitzpatrick, J., Kuo, F.-C., Kretschmer, M., Lessmann, J., Niephaus, C., de la Oliva, A., Robitzsch, S., & Zdarsky, F. (2009). Carrier-grade wireless mesh networks: D3.4-unicast and multicast routing specification and analysis. <http://www.ict-carmen.eu/wp-uploads/2009/D3.4.pdf>.
- [8] CARMEN-Consortium (). Ratified architecture deliverable.
- [9] Duros, E., Dabbous, W., Izumiya, H., Fujii, N., & Zhang, Y. (2001). A Link-Layer Tunneling Mechanism for Unidirectional Links. RFC 3077 (Proposed Standard).
- [10] Eckert, T., Rosen, E., Aggarwal, R., & Rekhter, Y. (2008). MPLS Multicast Encapsulations. RFC 5332 (Proposed Standard).
- [11] Farrel, A., Vasseur, J.-P., & Ash, J. (2006). A Path Computation Element (PCE)-Based Architecture. RFC 4655 (Informational).
- [12] Gundavelli, S., Leung, K., Devarapalli, V., Chowdhury, K., & Patil, B. (2008). Proxy Mobile IPv6. RFC 5213 (Proposed Standard).
- [13] Henkel, D., Englaender, S., Kretschmer, M., & Niephaus, C. (2011). Connecting the unconnected - economical constraints and technical requirements towards a Back-Haul network for rural areas. In *IEEE Globecom 2011 Workshop on Rural Communications-Technologies, Applications, Strategies and Policies (RuralComm 2011) (GC'11 Workshop - RuralComm)*. Houston, Texas, USA.
- [14] Horstmann, T., Kretschmer, M., Niephaus, C., Mödeker, J., & Sauer, S. (2011). Development framework for prototyping heterogeneous Multi-Radio wireless networks. In *ICCCN 2011 Workshop on Wireless Mesh and Ad Hoc Networks (WiMAN 2011)*. Maui, Hawaii, USA.
- [15] Johnson, D., Hu, Y., & Maltz, D. (2007). The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4. RFC 4728 (Experimental).
- [16] Katz, D., Kompella, K., & Yeung, D. (2003). Traffic Engineering (TE) Extensions to OSPF Version 2. RFC 3630 (Proposed Standard). Updated by RFC 4203.
- [17] Kim, W., Kessler, A., Di Felice, M., & Gerla, M. (2010). Urban-x: Towards distributed channel assignment in cognitive multi-radio mesh networks. In *Wireless Days (WD), 2010 IFIP* (pp. 1–5).

- [18] Kone, V., Das, S., Zhao, B. Y., & Zheng, H. (2007). Quorum - quality of service routing in wireless mesh networks. In *ICST Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness (QShine)*.
- [19] Kretschmer, M., Batroff, P., Niephaus, C., & Ghinea, G. (2011). Topology discovery and maintenance for heterogeneous wireless Back-Haul networks supporting unidirectional technologies. In *17th Asia-Pacific Conference on Communications (APCC 2011)*. Kota Kinabalu, Sabah, Malaysia.
- [20] Kretschmer, M., & Ghinea, G. (2010). An IEEE 802.21-based approach for seamless wireless mobile integration using QoS-aware paths supporting unidirectional links. In *IEEE Globecom 2010 Workshop on Seamless Wireless Mobility (SWM 2010)*. Miami, Florida, USA.
- [21] Kretschmer, M., Niephaus, C., & Ghinea, G. (2009). QoS-aware flow monitoring and event creation in heterogeneous MPLS-based wireless mesh networks supporting unidirectional links. In *9th IEEE Malaysia International Conference on Communications 2009*. Kuala Lumpur, Malaysia.
- [22] Kretschmer, M., Niephaus, C., & Ghinea, G. (2011, Accepted for Publication). Wireless multi-access environments and quality of service provisioning: Solutions and application. chapter Heterogeneous Meshed Wireless Back-haul Network Integrating Unidirectional Technologies. Information Science Reference, USA/UK.
- [23] Kretschmer, M., Niephaus, C., Henkel, D., & Ghinea, G. (2011). QoS-aware wireless back-haul network for rural areas with support for broadcast services in practice. In *Fifth IEEE International Workshop on Enabling Technologies and Standards for Wireless Mesh Networking (IEEE MeshTech 2011)*. Valencia, Spain.
- [24] Laubach, M., & Halpern, J. (1998). Classical IP and ARP over ATM. RFC 2225 (Proposed Standard). Updated by RFC 5494.
- [25] Maurina, S., Fitzpatrick, J., Trifan, L., & Murphy, L. (2010). An Enhanced Bridged-Based Multi-Hop Wireless Network Implementation. In *Wireless Internet Conference (WICON), 2010 The 5th Annual ICST* (pp. 1–9).
- [26] Mihailovic, A., Chochliouros, I., Kousaridas, A., Nguengang, G., Polychronopoulos, C., Borgel, J., Israel, M., Conan, V., Belesioti, M., Sfakianakis, E., Agapiou, G., Aghvami, H., & Alonistioti, N. (2009). Architectural principles for synergy of self-management and future internet evolution. In *Proceedings of Future Network & MobileSummit 2009 Conference* (p. 8).
- [27] Mirchandani, V., Prodan, A., & Marce, O. (2008). A non-tpc based enhanced topology control process for multi-radio wireless mesh networks. In *Sensor Technologies and Applications, 2008. SENSORCOMM '08. Second International Conference on* (pp. 758–763).
- [28] la Oliva, A. d., Soto, I., Banchs, A., Lessmann, J., Niephaus, C., & Melia, T. (2011). Ieee 802.21: Media independence beyond handover. *Comput. Stand. Interfaces*, 33, 556–564.
- [29] OpenMesh.org (2011). B.a.t.m.a.n. advanced.
- [30] Pan, P., Swallow, G., & Atlas, A. (2005). Fast Reroute Extensions to RSVP-TE for LSP Tunnels. RFC 4090 (Proposed Standard).
- [31] Perkins, C., Belding-Royer, E., & Das, S. (2003). Ad hoc On-Demand Distance Vector (AODV) Routing. RFC 3561 (Experimental).
- [32] Ramasubramanian, V., & Mosse, D. (2008). Bra: A bidirectional routing abstraction for asymmetric mobile ad hoc networks. *IEEE/ACM TON* 16, 116–129.
- [33] Serrano, P., Patras, P., Perez-Costa, X., Gloss, B., & Chieng, D. (2009). A MAC Layer Abstraction for Heterogeneous Carrier Grade Mesh Networks. In *ICT-MobileSummit '09*. Santander, Spain.
- [34] Sridhar, K., Casetti, C., & Chiasserini, C.-F. (2009). A localized and distributed channel assignment scheme for wireless mesh networks. In *Local Computer Networks, 2009. LCN 2009. IEEE 34th Conference on* (pp. 45–52).
- [35] Subramanian, A. P., Gupta, H., Das, S. R., & Cao, J. (2008). Minimum Interference Channel Assignment in Multiradio Wireless Mesh Networks. *IEEE Transactions on Mobile Computing*, 7, 1459–1473.
- [36] Wushi, D., Niansheng, C., & Qiang, S. (2008). A qos routing protocol supporting unidirectional links in manets. In *Proc. IEEE Asia-Pacific Services Computing Conference APSCC '08* (pp. 1401–1405).
- [37] Zhou, Y., Yun, M., Kim, T., Arora, A., & Choi, H.-A. (2009). RI-based queue management for qos support in multi-channel multi-radio mesh networks. In *Network Computing and Applications, 2009. NCA 2009. Eighth IEEE International Symposium on* (pp. 306–309).
- [38] Zuniga, M., & Krishnamachari, B. (2004). Analyzing the transitional region in low power wireless links. In *In First IEEE International Conference on Sensor and Ad hoc Communications and Networks (SECON)* (pp. 517–526).