D3.8 - ULOOP Framework Design and Implementation Report

Deliverable Number	D3.8
Lead Beneficiary	ULHT
Nature/Dissemination Level	PU
Working Group/Task	WP3
Editor	ULHT (Paulo Mendes)
List of Authors	***ULHT*** (Rute Sofia, Paulo Mendes, Waldir Moreira, Jonnahtan Saltarin, Luis Lopes, Tauseef Jamal), ***HWDU*** (Qing Zhou), ***UniGe*** (Jean-Marc Seigneur, Carlos Ballester), ***UniK*** (Huseyin Haci, Huiling Zhu, Hassan Osman), ***TUB*** (Sebastian Peters, Mürsel Yildiz, Fikret Sivrikaya), ***UniUrb*** (Alessandro Bogliolo)
Date (Project Month and dd.mm.yy)	M38 12.11.2013
QAT Reviewer	Fikret Sivrikaya (TUB)

Executive Summary

This document represents deliverable D3.8 of the EU FP7 IST project ULOOP (*User-centric Wireless Local Loop,* grant Number 257418). It is a follow-up of deliverables D2.3, D3, and D3.a. The latter two deliverables were confidential, and so D3.8 provides to the general public a global perspective of results derived from the work developed in Work Package 3, namely, final specification aspects, conceptual work as well as implementation choices. It also includes pointers to the validation of each of the pieces described.

Table of Contents

List of Figures

List of Equations

List of Tables

List of Definitions

This section summarizes ULOOP definitions, the most relevant ones being addressed with more detail in section 2.1. The ULOOP definitions have been aligned to current European Telecommunications legislation and regulation aspects, as far as it is possible. For such alignment, refer to ULOOP deliverable D2.3 [1]

Acronym	**Meaning**
Application	Computer software design to perform a single or several specific tasks, e.g. a calendar. In ULOOP, it is an instantiation of a user service. For instance, Voice over IP is an example of a user service provided by different applications, e.g. Skype, or Gizmo.
Application Programming Interface (API)	Well-defined specification used in a software program to access services or resources provided by another software application. Establishes the interface between two different applications.
Business incentives	Business incentives relate to micro-generation models based on the guidelines provided by WP2 (Task 2.2., Socio-economic Sustainability).
Community	Set of ULOOP nodes that hold common interests (such as sharing connectivity or resources / peripherals) at some instant in time and space. In other words, the node location exhibits a space and time correlation, which is the basis to establish a robust connectivity model.
Conditional access system	Any technical measure and/or arrangement whereby access to a protected radio or television broadcasting service in intelligible form is made conditional upon subscription or other form of prior individual authorization. Refer to D2.3 for the respective legislation.
Consumer	See service recipient.

End-user	See user.
Exclusive rights	"exclusive rights" [2002/77/EC Art. 2.5] shall mean the rights that are granted by a Member State to one undertaking through any legislative, regulatory or administrative instrument, reserving it the right to provide an electronic communications service or to undertake an electronic communications activity within a given geographical area. Refer to D2.3 for the respective legislation.
Handover	Process of transferring an ongoing communication session between two networks, or two communities, from one or several ULOOP enabled devices to other device(s).
Incentive	A factor (economic or sociological) that motivates a particular action or a preference for a specific choice.
Interconnection	The physical and logical linking of public communications networks used by the same or a different undertaking in order to allow the users of one undertaking to communicate with users of the same or another undertaking, or to access services provided by another undertaking.
Interest	A parameter capable of providing a measure (cost) of the "attention" of a node towards a specific location in a specific time instant. In other words, an interest is a parameter that provides a node with a measure of a specific time and space correlation.
Local Loop	The physical circuit connecting the network termination point to a distribution frame or equivalent facility to the access network.
Network infrastructure	Collection of links and networking nodes that together enable data transmission between (Internet) users. The links connect the nodes together and are built upon an underlying transmission network, which physically pushes the message across the link.
Network Service	A system that is required to support, from a network perspective, user services. For instance, Internet connectivity is a network service.
Network Service	Set of operational network functionality required to sustain user services. Examples of network services in ULOOP are trust management; resource management; identity disambiguation.
Network Termination	Defines the last logical block of the local-loop. It is normally a device controlled by the provider and which connects to a subscriber data or networking equipment.
Operator	Entity that manages a network infrastructure.
Owner	An entity (e.g., end-user, operator, virtual operator) that is to be made responsible for any actions concerning his/her device.
Provider	Entity that provides services to subscribers.
Recipient of a service	"recipient of the service" [2000/31/EC Art.2.d] any natural or legal person who, for professional ends or otherwise, uses an information society service, in particular for the purposes of seeking information or making it accessible;

Resources	A physical or virtual element of a global system. For instance, bandwidth, energy, data, devices, are examples of resources in ULOOP.
Service	Any Information Society service, that is to say, any service normally provided for remuneration, at a distance, by electronic means and at the individual request of a recipient of services. For the purposes of this definition: • "at a distance" means that the service is provided without the parties being simultaneously present, • "by electronic means" means that the service is sent initially and received at its destination by means of electronic equipment for the processing (including digital compression) and storage of data, and entirely transmitted, conveyed and received by wire, by radio, by optical means or by other electromagnetic means, • "at the individual request of a recipient of services" means that the service is provided through the transmission of data on individual request. Refer to D2.2 for further details.
Service Provider	An entity that provides some kind of service to Internet stakeholders (users or providers). Examples are ISPs, ASPs, WISPs, access providers, users, as well as ULOOP communities.
Session	Permanent or transient information exchange between two or more devices and/or users.
Social Trust	Trust which builds upon associations of nodes based on the notion of shared interests, or affinities between owners.
Subscriber	Any natural person or legal entity who or which is party to a contract with the provider of publicly available electronic communications services for the supply of such services.
Technical incentive	Technical incentives in ULOOP relate to natural features of the technology that result in a win-win match when cooperation is applied.
Trust Association	A unidirectional social trust association between two different nodes.
ULOOP Gateway	Role (software functionality) that reflects an operational behavior making a ULOOP node capable of acting as a mediator between ULOOP systems and non-ULOOP systems – the outside world.
ULOOP node	Role (software functionality) that a wireless capable device takes. Concrete examples of nodes can be specific user-equipment, access points, or event some management server.
User	A legal entity or individual using or requesting a publicly available electronic communications service for private or business purposes, without necessarily having subscribed to such service.

List of Acronyms

Acronym	Meaning
AP	Access Point
CAC	Call Admission Control
DT	Dispositional Trust
GUI	Graphical User Interface
GW	Gateway
ISP	Internet Service Provider
LTE	3GPP Long Term Evolution
LTE EPC	LTE Evolved Packet Core
MAC	Media Access Control
OFDM	Orthogonal Frequency Division Multiplexing
OS	Operating System
OSN	Online Social Network
PET	Privacy Enhancing Technology
QoE	Quality of Experience
QoS	Quality of Service
RTC	Request to Connect
SIA	Social Interaction Analysis
SNR	Signal-to-Noise Ratio
UCN	User-Centric Networking
UE	User Equipment
ULOOP	User-centric Wireless Local Loop
Wi-Fi	Wireless Fidelity
WLAN	Wireless Local Area Network
WP	Work Package
SP	Service Provider
NAP	Network Attachment Point

Acknowledgements

Acknowledgement to the significant contribution provided by all partners participating in Work Package 3 of the ULOOP Project.

1. Introduction

This document summarizes the findings of Work Package 3 and is therefore an update to D3a. While D3a was more focused on the software architecture specification and software design of ULOOP, this document starts from the novel concepts that ULOOP explored, covering all aspects of the work developed, including conceptual work that was not covered by implementation, until the proof-of-concept stage that ULOOP currently provides. As such, it considers input from D3, D3a, as well as input already provided in D3.7 and D3.9, the software results of ULOOP.

The document is organized as follows. This section, Section 1, provides an overview of the goals of the project as well as the contribution of the ULOOP vision. Section 2 goes over the functional blocks that ULOOP addresses (Trust Management and Cooperation Incentives; Resource Management; Mobility Aspects), explaining how the concepts were developed and implemented. Section 3 provides a list of the output of the project grouped in deliverables, accepted scientific papers, scientific papers under submission, IPRs and other material.

1.1 WP3 Goals

The Work Package 3 (WP3) was dedicated to the design, implementation and integration of the ULOOP software suit and its building blocks. WP3 covered the investigation of aspects related to cooperation incentives and trust management, resource management, as well as mobility. These topics were addressed based upon the global specification and the full outcome of WP2. The technical use-cases are common and hence each set of functionality is to be added to a specific part of the global framework.

Within WP3 a specific task was also dedicated to the integration of the functionality derived from each functional block. Such task was responsible to integrate the designed technology into the full operation of a ULOOP-enabled node. In WP3, validation was based on tools such as simulators, emulators, but also local test-beds.

1.2 WP3 Contributions to the ULOOP Vision

ULOOP has the vision to assist in the deployment of robust wireless local loops, which today are the de-facto solution to complement broadband access. The project had as motivation the realization that the current infrastructures are underused. Such underuse can be corrected if operators are provided with software that provides answers to two main questions: i) can we track the source of information

without compromising user's anonymity; ii) can we improve the network operation in these scenarios by considering trust-based metrics.

ULOOP provides answers to the aforementioned aspect, by addressing the following aspects:

- **The proliferation of these networks requires a way to give entities a sense of liability.** Instead of addressing the complex paradigm of strong security, by developing strong and complex architectures often based on encapsulation, ULOOP addresses the problem by bringing to the network a social perspective, namely, allowing the exchange of services to occur only within specific circles of trust and based on an individual willingness to trust the world.
- **Both the user and the provider need non-repudiation in place**. What ULOOP considers is the notion of mobile token in the form of a crypto-id, a unique identifier that marks the source of information independently of the device in use.
- **Selfishness of the peers can prevent the ULOOP vision**. As ULOOP relies on a distributed trust scheme (social trust), it was necessary to ensure that the system would bootstrap in adverse environments. Moreover, the trust propagation mechanisms, to be robust enough, require an incentive mechanism, which in ULOOP is provided by the CooperationManager entity.
- **From a wholesale perspective, the ULOOP concept requires rewarding good behavior**. While the CooperationManager assists ULOOP in bootstrapping, the RewardManager takes care of rewarding entities that cooperate.
- **The trust metrics embody social aspects that are relevant to consider in fairness aspects concerning resource management and allocation**. As such, ULOOP integrates an augmented control mechanism which takes into consideration trust levels between different nodes of a graph, to serve requests coming from the different entities in a way that considers the relevance of an entity towards the sharing of resources.
- **Optimal resource sharing requires a different design of the wireless MAC Layer.** When addressing resource management, ULOOP can, as explained, provide fairness that is proportional to the contribution of specific entities to the robustness of the network – the more one shares, the more resources it can get. Still, ULOOP deals with the regular OSI MAC Layer operation, which makes each request be served at an instant in time. Therefore, from a resource management perspective, ULOOP would always face the fact that if a station with a lower trust level had better signal to an access point, this station would get the medium first and therefore other entities with a better trust association would have to wait for a chance to transmit. In ULOOP, what was done was to develop a mechanism that gives the possibility for

an access point to transmit within the same time frame to multiple stations, in a way that is fully backward compatible with MAC802.11 standards.

- **Monitoring.** The User-Centric Networking (UCN) concept is an example of highly dynamic and fully stochastic network environments. Various decision-making mechanisms, which address aforementioned aspects, are the building blocks of the ULOOP functionality. These decisions are taken based on a certain set of critical network indicators, which in turn have an impact on various network parameters. Constructing a measurement plane, ULOOP deals with monitoring aspects of various critical network indicators. With the measurement plane, ULOOP targets optimality in upper layer decision-making mechanisms. Additionally the resource consumption of third party users becomes significant in considering ULOOP enablers such as trust management, optimal resource sharing, selfishness of members and incentive mechanisms. Hence in ULOOP feasible plug & play cooperative software is developed monitoring various network performance indicators and user behaviors.
- **Frequent roaming impact on user-centric anchor points**. In user-centric scenarios it is assumed that the user controls part of the network devices (as is today the case if femtocells are considered). As such, if local anchor points perform delegation of mobility, the overlay of anchor points may appear and disappear in a way that is not adequately controlled. Therefore, ULOOP proposed a mechanism, the Mobility Coordination Function, which takes care of coordinating and providing the users with the best anchor points, both depending on policies of the network and of the user side.
- **Roaming patterns have some statistical similarities that may assist mobility management.** Based on the notion of routine, and on previous work that shows that some properties of roaming can be statistically inferred, ULOOP provides a mobility estimation plugin, which considers information naturally present and available in visited networks to provide a time estimate, as well as a potential target, for a next handover.

2. ULOOP Functional Blocks

This section goes over the ULOOP blocks initial specification, derived from the content provided in D2.3 [1], updated in D3 [2] and D3.a [9], and related to the **full lifespan** of WP3.

For each subsection of Section 2 starts by explaining the goals proposed to be achieved, the conceptual novelty and whether or not it was followed by the consortium and why; the status of readiness of the code, as well as the status of validation of each aspect that was investigated.

In its deliverable, the following innovation blocks of WP3 are considered:

- Trust management and cooperation incentives (developed in task 3.1 of WP3).
- Resource management (developed in task 3.2 of WP3)
- Mobility Aspects (developed in task 3.3 of WP3)
- Interoperability and integration aspects (developed in task 3.4 of WP3)

For the development of these WP3 functional blocks, the following aspects tackled in other work packages were also considered:

- Network-neutrality aspects, with regards to tasks 2.2 and 5.1.
- Social sustainability, tasks 2.2 and 5.3.
- Economic sustainability, task 5.3.

From an architectural, end-to-end perspective, based on the proposed goals, ULOOP developed concepts that could assist a user-friendly deployment of user-centric networks in a way that would leverage new business models, both for the user and for the operator, having as focus existing wireless infrastructures.

Since an early stage, WP3 considered realistic boundaries, e.g. in terms of feasible equipment to be used, feasibility in terms of available time frame, and usefulness. Therefore, in D2.3 [1] the consortium provided clear boundaries to the equipment and software to be considered. Setting early requirements allowed the project to reach a technological readiness early; however, it also implied having to place some boundaries in terms of the concepts to be explored.

The work of specifying each of the ULOOP functional blocks started in the first year and undertook several revisions, while the consortium continued on improving the software available.

In this section we address each of the functional units developed in WP3. As specifications were already provided in D3.a [9], they are directly referenced here. For each section, we provide a high-level illustration of each functional block, of which design was translated into an UML scheme that provides not only details concerning the functionality to be addressed, but also the respective interfacing and communication between blocks, useful for the implementation and integration phases.

Validation aspects are referenced in each section. Then, a table also summarizes the aspects that were validated in the course of the work.

2.1 Trust Management and Cooperation Incentives

In ULOOP, trust management and incentives for cooperation are related to understanding how to define and build circles of trust on-the-fly to provide the user with liability.

Trust management is based on reputation mechanisms able to identify end-user misbehavior and to address social aspects, e.g., the different types of levels of trust users may have in different communities (e.g., family, affiliation). In situations where the created network of trust is not enough to allow resources to be shared, ULOOP devices are able to use a cooperation incentive scheme based on the transfer of credits directly proportional to the amount of shared resources.

Trust Management here is split as follows: i) Identity management; ii) trust setup; ii) trust management iii) cooperation and rewarding.

2.1.1 Trust Setup

Trust setup in ULOOP is a one-time process that a user (owner) executes on one of its devices. This process does not need to be repeated on other devices of the user. After the setup procedure, the trust value may be updated based on a new value for the dispositional trust value, which can be always adjusted in each of the devices owned by the same user as a first step. It is worth mentioning that the trust setup process may be repeated; a user is always free to request a new crypto-id and nickname for each of his/her devices (by ticking the option “yes” when the ULOOP setup asks if this is the first device ever in ULOOP for that given user).

Trust setup is triggered in any ULOOP node and comprises a series of steps which result in: i) a unique identifier, the crypto-id; ii) a wallet with an initial set of credits; ii) an initial trust value towards any new neighbour, familiar or not – dispositional trust.

2.1.1.1 ULOOP Virtual Identities, the Crypto-Id

The first step towards building trust references in communities, i.e., from a ULOOP node to others, is to be able to uniquely identify owners of ULOOP nodes. Ideally, the recognition must be attack-proof. Hence the end-user must be able to authenticate her/him. However, it also important to protect the privacy of this end-user, so this building block contains both identity management and *privacy-enhancing technologies (PETs).* To fit identity management to the distributed trust system required in the trust management block, identity management should be tackled in the following aspects:

- ***Implementing the appropriate identity management mechanisms that will provide authentication and authorization***. ULOOP will reuse the concept of crypto-identifiers

(crypto-ids) based on asymmetric cryptography. With such crypto-ids, the end-users can prove in a decentralized way and with cryptographic strength that they really own the secret linked to the crypto-id. Concerning privacy, creation and proof of ownership of crypto-ids does not require a centralized identity authority. Thus, end-users in ULOOP will protect their privacy through crypto-ids that they generate themselves and act as their pseudonyms not linked to their real world identity.

- ***Mitigating identity-based attacks on the ULOOP trust metric by means of a novel identity disambiguation scheme***, going beyond the state of the art, which will try to detect whether a ULOOP end-user is a fake end-user or not based, e.g., on real end-user information extracted from available real social networks. The investigation of this aspect was not concluded due to the internal restructuration suffered in 2011 by the partner responsible by this effort, Alcatel-Lucent Bell Labs.
- ***Identity disambiguation based on OSNs***. The information extracted from online social networks will also be used to compile reputation evidence that will be in turn taken into account within the trust metric of the first building block. Such compiled reputation evidence will also be fed back into those social networks to act as reputation-based incentive as mentioned in the second main building block. The investigation of this aspect was not concluded due to the internal restructuration suffered in 2011 by the partner responsible by this effort, Alcatel-Lucent Bell Labs.

The first alternative would be to enforce one unique virtual identity per user in ULOOP. Once in place it is more familiar for the user to manage one virtual identity and to avoid attacks based on the use of different virtual identities per user, for example, preventing voting twice. Moreover, using a unique crypto-ID will avoid a potential complex process of identity disambiguation.

However, such an approach is likely to assume an authentication service that ensures the authenticity of the unique virtual identity to be used per user. This would require some steps for the authentication and verification of the user's identity in the real world. For example, it may require the ownership of one identity smartcard per user combined with a one-time verification of the authenticity of the identification. After the verification of the identification, the crypto-ID generated based on such identification would be used in any country. Worth mentioning here is the fact that most countries are implementing digital identity systems to automate most of the national services. One example is Portugal, were the citizen card has embedded a chip with a one-time generated crypto-ID that is used to authenticate the user in several different services. Some services, such as changing the address information, require an extra secret key that is provided to the user with the card. This process will allow unique crypto-IDs to be generated based on any system that EU counties will decide to implement in the future to identify their citizens electronically.

Figure 1 shows the flowchart to generate a unique crypto-ID based on a set of information provided to the user by an authorized entity (e.g. the personal identification number embedded in a citizen identity card provided by a government to any citizen, or a mobile phone number associated to a unique SIM card). Such personal identification number will be used to generate a unique crypto-ID based on a hash function that is implemented in any ULOOP node or gateway. The local generated crypto-ID will need to be verified by an authorized entity in order to allow the ULOOP node/gateway to gain full access to the ULOOP community. When such verification cannot happen, the ULOOP device gets a minimum trust level in the community, allowing it to use a predefined set of minimum resources.

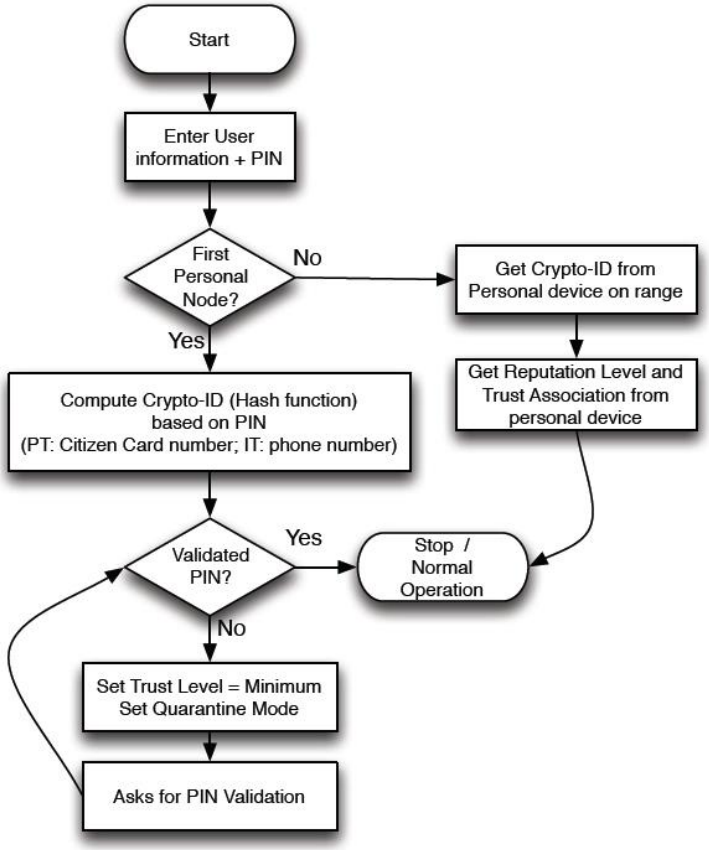

Figure 1: Generation of Unique crypto-ID flow-chart.

In ULOOP, owners (users) are likely to be responsible for more than one active device. One would be a primary device, and the remainder equipment will share the same crypto-ID generated by the first personal device, as well as the reputation level and trust associations associated to the unique crypto-ID. This is possible by using secure in range wireless or wired communications. Synchronizing the reputation levels and trust associations among personal devices will allow the user to always make use of the earned reputation level, trust associations and credits that resulted from the usage of the

unique crypto-ID in another personal device. Synchronization of trust information can be done by using prior-art on file and data synchronization.

The validation of the unique crypto-ID can be done by making use of any opportunity to access the Internet (limited Internet access should be allowed by the minimum trust level). This may create some problem in extreme cases, in which Internet access is not possible for a long time. However, such scenarios are more related to delay-tolerant networks and not to ULOOP, in which it is expected that trust management and cooperation incentives will create the conditions to make Internet access more pervasive than today.

Nevertheless, it is clear that the usage of a unique crypto-ID may limit the usage of ULOOP in fully decentralized environments, namely in the presence of isolated ULOOP networks (without any Internet access whatsoever) and new users (that still need their crypto-IDs to be validated).

2.1.1.2 Dispositional Trust, Bringing the Willingness to Trust into the Picture

User-centric networks such as ULOOP are supported both by static, fully dedicated nodes as well as by nodes provided by end-users on-the-fly. Since some nodes are carried by Internet end-users, their networking composition, surrounding environment and organization can rapidly change. As such, the dispositional trust level on a given node might not be appropriate in all circumstances and should be able to be adapted and changed over time, in order to protect the node's integrity. The process of dispositional trust adaptation might occur in two different cases:

- The node has a dispositional trust level that is inappropriate and leaves it too open to attacks.
- The node joins a different community than the initial one in which the dispositional trust level had been setup.

An untrustworthy node in ULOOP goes through a boot-up procedure where the node may be the first one an owner is responsible for, or one of several nodes. In the former case the owner is prompted to set its Dispositional Trust (DT) level, e.g. being able to select from a list of predefined values, which range from **0 to 100**, being 0 *"paranoid"* which means that a priori the node will not trust anyone, and being 100 *"blind trust"* which means that the node will trust no matter what.

In the second case, the user is presented with two options: i) to clone the dispositional trust level assigned to other devices that are already in ULOOP and that she/he owns, for the usage of unique crypto-IDs in different personal devices: ii) to assign a new DT level for the node being introduced, as explained in the previous paragraph. These two cases are depicted in Figure 2.

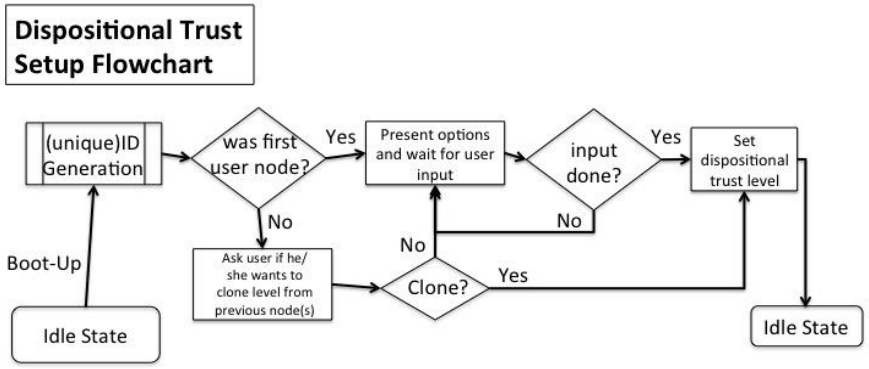

Figure 2: Dispositional trust setup.

As explained previously, the DT level may or may not remain constant throughout all of the node's lifetime.

2.1.2 Trust Management

Trust management is performed in two different phases of ULOOP: i) when connectivity is attempted; ii) during data transmission.

When a node attempts to connect to a wireless network (e.g. via a captive portal), this triggers a request for resources, an aspect that is tackled by the TrustManager entity in ULOOP.

2.1.2.1 Trust Manager

The TrustManager is the main skeleton of Task 3.1. It is in charge of executing the main, and establishing and maintaining the external interfaces (communication via TCP sockets) with the Trust Manager of other ULOOP nodes (requester to requestee and vice versa), as well as the internal interfaces with other operational modules (Resource manager and Mobility manager) within the same node. When first instantiated, TrustManager performs a series of initial setup procedures, such as the virtual crypto-id generation and validation, as well as the dispositional trust setup. After this, and before going to the main operational mode, it starts a set of periodic activities from the reward manager that have to be executed in the background in order to ensure the proper operation and update of the bank account and the wallet of the node.

Finally, the main functionality allows the node to perform its main operation, such as exchanging crypto-ids with other ULOOP nodes in order to start a cooperation process, performing social trust computation of those nodes and carrying out the control of cooperation, fundamental to decide if a service is obtained or allowed from/to another node. As the Trust Manager is expected to run in both a ULOOP node (end-user equipment) and on a ULOOP gateway (e.g. Access Point), we have

developed two different versions of the same specification. For the ULOOP node we have considered in D2.3 [1] the main operating system as Android. As such, the Trust Manager has been developed in Java for Android. From an implementation perspective and to ensure that the code would be available on gateways, we have then ported the code to C. The mapping provided in D3.a [9] reflects this design choice, providing the paths and methods both on Android java and on C.

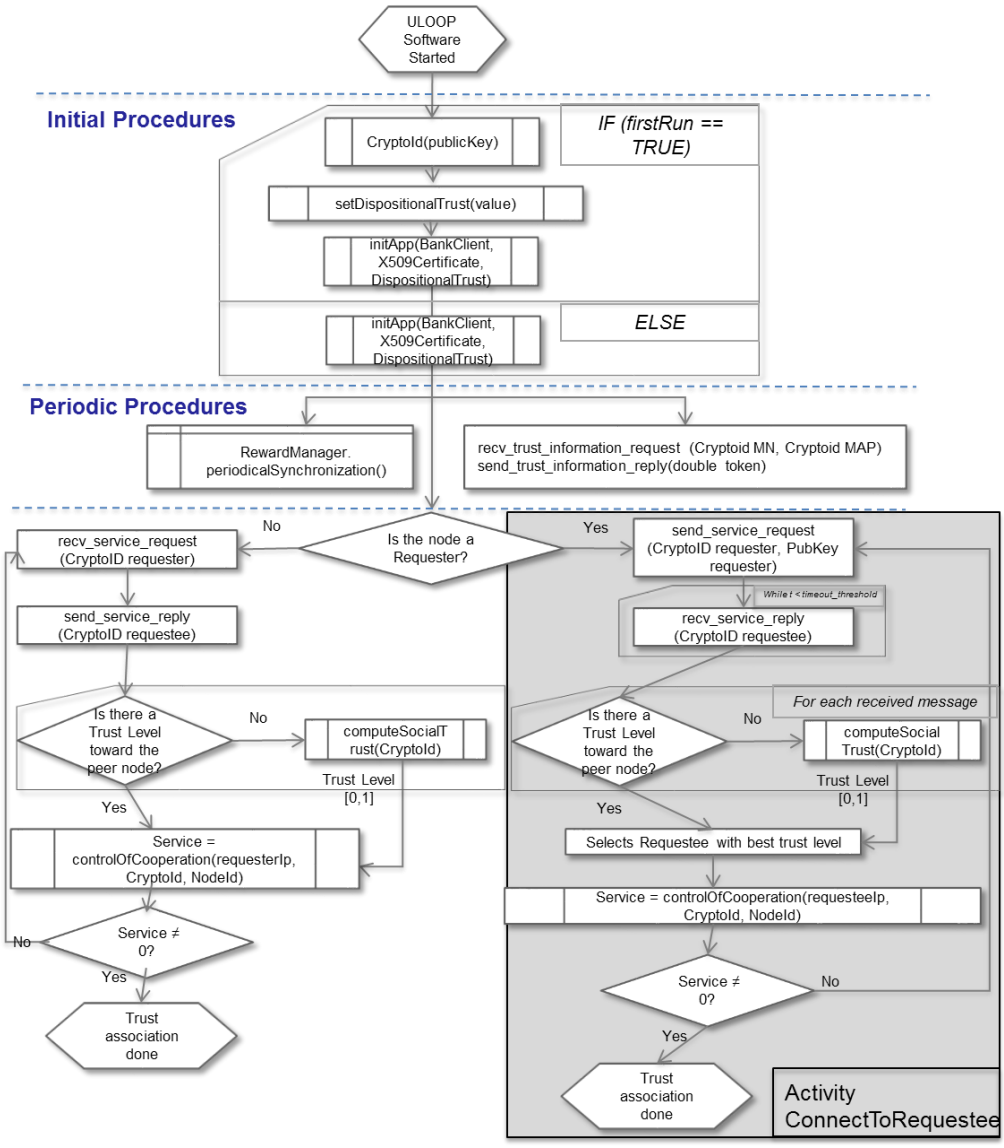

Figure 3: Flow-chart of trust manager.

2.1.2.2 Cooperation Manager

At bootstrap, the Cooperation Manager starts by assigning an initial amount of cooperation credits to the user. This initial amount takes into consideration the node trust level and established minimum and maximum amount of credits thresholds for ULOOP devices. As the Reward Manager handles

credits, the Cooperation Manager informs it of this amount in order to make the Reward Manager aware of how much cooperation credits the device has.

During negotiation, credits are used by the requestee to express the cost of the service/resource he/she provides. The negotiation phase is positively concluded if and only if an agreement is reached both in terms of service level and in terms of credits between requester and requestee.

In ULOOP trust can also be used as a parameter to affect the cost of the negotiated service. Although the ULOOP incentive framework is open to the implementation of any functional relation between cost and trust, a representative example is provided by the piece-wise linear function in Equation 1.

$$C(T) = \begin{cases} C_{\min} + \dfrac{C_{\max} - C_{\min}}{T_{th}}(T - T_{th}) & T < T_{th} \\ C_{\min} & T \geq T_{th} \end{cases}$$

Equation 1: utility function, relation between cost and incentives.

where C is the cost in terms of credits, T is the trust of the requestee on the requester, C_{min} is the minimum reward (cost) asked by the requestee regardless of his/her trust on the requester, C_{max} is the maximum reward asked to serve untrusted users, and T_{th} is the trust threshold above which the minimum cost is applied to the requester.

Within a ULOOP community, T_{th} could be imposed to all members, in order to be used as tuning parameters to adjust the behaviour of the community as a whole, while C_{max} and C_{min} could be set by any member according to their need for direct rewards.

On the control of cooperation, if the device is a requester and requires a service, it must compute the amount of credits that will convince the prospective requestee in engaging in cooperation. Additionally, as the tokens are the common language among the different managers, a number of tokens is computed by means of Social Trust Computation and a promise of payment is done by means of Reward Manager. Then, the Cooperation Manager sends (by means of external interface made available to all modules of the Trust Manager) a service request to the potential requestee, which in turn replies specifying whether or not it will engage in cooperation.

In the case the device is a requestee, it receives the service request and evaluates whether the received credits are enough to provide the requested service. Then, a check on the amount of resources is done is order to assess whether the requestee can answer the service request. If so, the requestee (i.e., Reward Manager) accepts the received amount of credits, Social trust Computation updates the trust level, and issues a service reply informing the Cooperation and Trust Managers that requestee is ready to engage in cooperation.

The Cooperation Manager is expected to run in the both ULOOP node and gateway. The role of a device will be set by detecting the conditions around and feeding that data to the respective daemon. For instance, a device may become a gateway because it is connected to the Internet and if it has the required trust level. So, if by some reason the trust level changes, that node may be automatically prevented from becoming a gateway. Two different versions of the Cooperation Manager implementation exist: the first one (requester mode) is done in Java for Android, while the second (requester and requestee mode) is implemented in C.

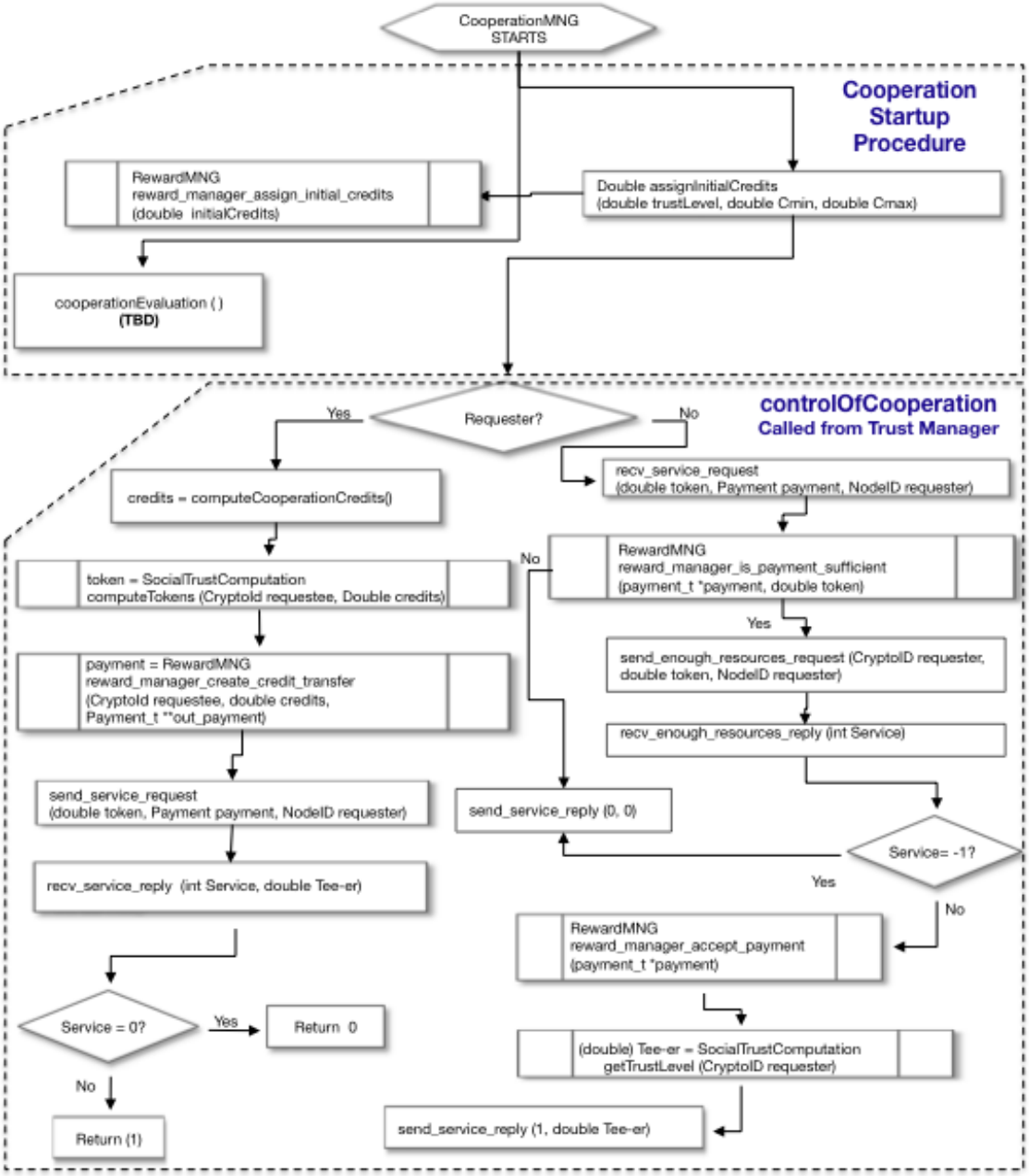

Figure 4: Flow-chart of cooperation manager.

2.1.2.3 Social Trust Modelling

This section describes initial work concerning the trust metrics and trust cost functions that are under development in ULOOP. The computation of trust is provided by a function implemented in ULOOP nodes and gateways. Trust computation is a dynamic cost function that has to be sufficiently strong to provide, based on a local perspective, attack resistance. It comprises therefore the dispositional trust of a node, as well as evidence concerning contacts with other nodes. To explain our function we consider three nodes: node i, the node that is about to compute a trust level towards a node z, and node j representing a node in the same community as node i. Node i has a dispositional trust level $\gamma \in [0,1]$.

In order for node i to compute the trust association cost towards node z (t_{iz}), i takes into consideration recommendations sent by nodes j belonging to the community. Such recommendation may be *direct*, i.e., node j has a direct trust association to node z, t_{jz} or indirect, i.e., node j has an indirect trust association to node z with the association being established through some other node. Direct trust associations are more relevant (have more weight on the trust cost function) than indirect recommendations. Recommendations provide i with a trust cost that nodes in the community have towards a new node.

A *direct recommendation* received by node i represents an answer from a node j in the community, and contains the computed cost of one or several trust associations between j and the target node. An *indirect recommendation* received by node i represents an answer from a node j in the community which contains the computed cost of one or several trust associations between j and the target node, but j is not yet in the trust table of i.

The proposed trust computation function is provided by Equation 2 and therefore provides a cost for the association between a node i and a node j. It considers both direct and indirect recommendation values, as well as the owner's own beliefs - dispositional trust. Moreover, the more stable acquaintances are, the more trusted their recommendations become.

$$t_{iz} = \gamma * \left\lceil \frac{\alpha * \frac{\sum_{j=0}^{k} t_{jz}}{k} + \theta * \frac{\sum_{j=0}^{p} t_{jz}}{p}}{n-1} \right\rceil$$, where

k : number of direct recommendations, $k \leq n$

p : number of indirect recommendations, $p \leq n$

j : node providing trust recommendation, $j \leq n, j \in N \wedge j \neq i \wedge j \neq z$.

z : target node

i : node requesting recommendations

n : total of nodes in the community

Equation 2: Example of a utility function for social trust computation.

A second potential embodiment for the trust value is provided in Equation 3, where the trust cost function computes a trust value T in a ULOOP node p_m according to the following parameters:

$$T_{p_n}(p_m) = f\left(\sum O, \sum R, S_M, M_T, D_T\right)$$

Equation 3: example of a function for social trust computation.

An initial trust value format could be based on a triple (p,u,n) where *p* is the number of interactions with the ULOOP node with a positive outcome, *n* is the number of interactions with negative outcomes and *u* is the number of interactions whose outcome, positive or negative, is still unknown.

Once the trust value is known, it is time to trigger the trust decision-making process within the trusting ULOOP node. An initial approach for the trust policy may be that for any type of request, the trust value should be above a threshold between 0 and 1, for example, the dispositional trust level manually configured by the user, leading to the following condition for considering that the requesting ULOOP node is trustworthy enough for the request:

$$Tp(p_i) \geq D_T$$

One of the main challenges of the final ULOOP trust metric will be to make it attack-resistant such as resistant to the Sybil attack. To be able to bootstrap the ULOOP community, it will also be important to have a good number of users who are generally disposed to trust others. For this reason, the proposed ULOOP framework aims to reward those users who are essential to sustain a high level of cooperation (c.f. Sections 2.1.2.2 and 2.1.2.4).

We provide in Figure 5 the flow chart for trust computation: after boot up (1) the nodes check for their dispositional trust *D* (2) and activate a trust table (3). The trust table is a structure where each row is a tuple with the following structure: <Node Id, trust level, ageing>. When activated, the node provides each of its neighbors with an equal trust level of *D*. In other words, in environments where relations were not yet established, ULOOP nodes trust equally all nodes around. The Social Trust Computation may consider recommendations by neighbors to assist in computing periodically the trust table of each node.

Requests for social trust computation come from the trust manager, cooperation manager and are provided via a look up to the trust table.

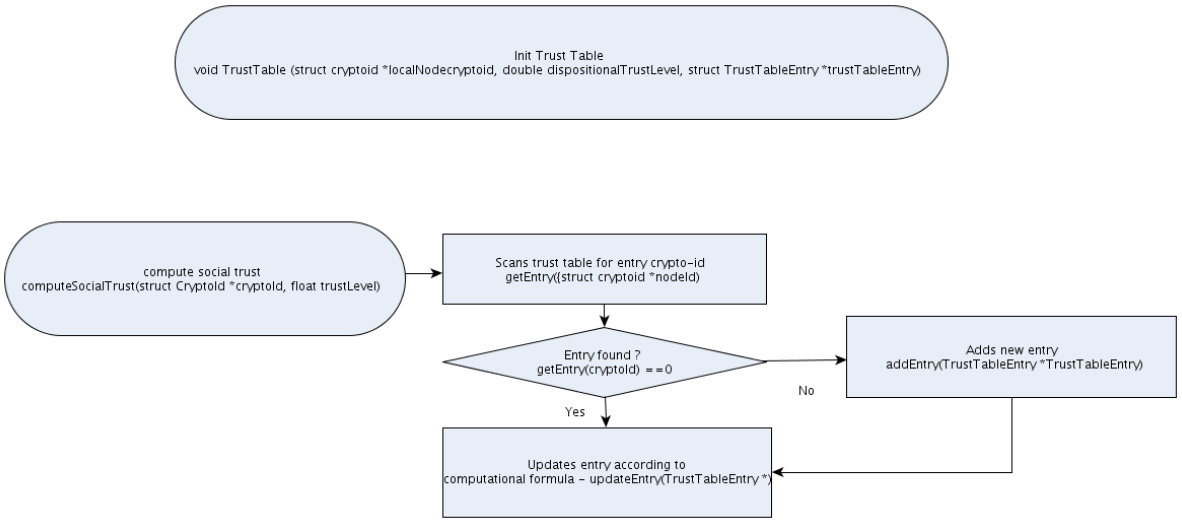

Figure 5: Flow-chart of social trust computation.

2.1.2.4 Reward Manager

The Reward Manager is the ULOOP software module that handles payments and credit transfers, used as additional rewarding incentives for cooperation. Credit transfers revolve around *credit units*, which are a form of virtual currency. The Reward Manager software module has been architected in a way to ensure that the transmission of credits is validated and secure, by preventing the creation of fake credits and the forging or duplicating of payments. The resulting virtual currency model is secure and, while being centralized in nature, allows the nodes to exchange credits when offline.

The Reward Manager is a software module running in each ULOOP node. The module does not require any additional external interfaces and it provides a set of APIs (in the form of function calls) that can be directly used by any other ULOOP module on the same node. Communication between nodes and the central authority managing credit exchanges and ownership (also known as the “Bank”) requires HTTP connectivity. Software in need of exchanging credits *must* use the Reward Manager’s APIs.

The system allows users, uniquely identified and registered with the central authority (Bank), to generate credits when registering into the system for the first time and to exchange such credits between registered users at any time. Each payment is uniquely identified. Payments may be made and exchanged even while disconnected from the Internet, but they must eventually be acknowledged by the Bank in order to be processed.

The Reward Manager is expected to run in both a ULOOP node (end-user equipment) and on a ULOOP gateway (e.g. Access Point). While the main implementation of the specification has been developed in C, the Reward Manager specification was also ported to the Java language, since the main operating system for ULOOP nodes will be Android. The operation of the Reward Manager module is described by the following flowchart.

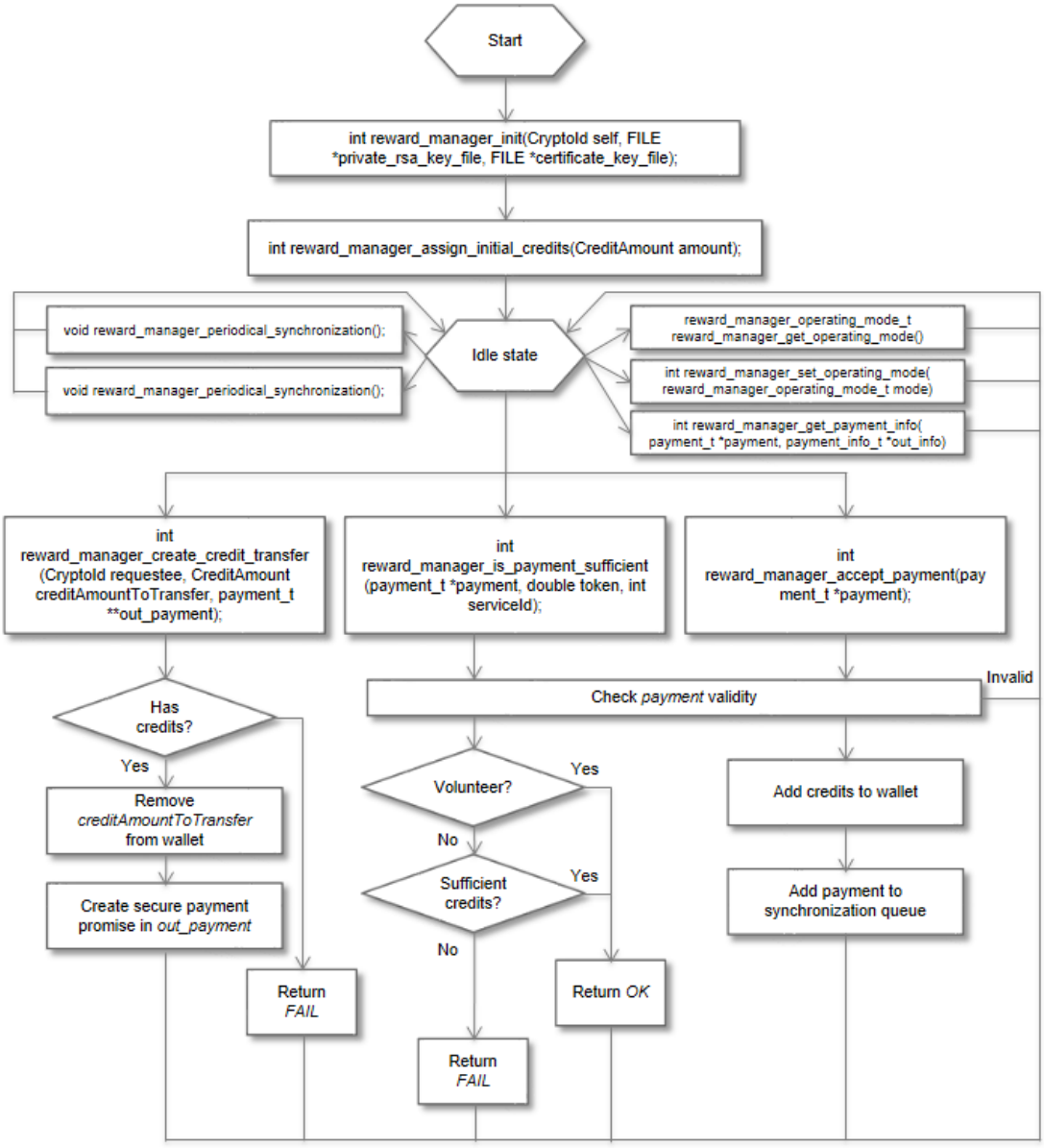

Figure 6: Flow-chart of reward manager

2.1.2.5 Operational stages

In this section we provide a brief description of the operational stages executed in a requester (requesting resources) and in a requestee (providing resources), based on the trust, cooperation and reward manager described before.

2.1.2.5.1 Requesting Resources

A requester goes through four stages: i) bootup; ii) requestee discovery; iii) data transfer; iv) dispositional trust adjustment. The bootup phase is present in any ULOOP node, be it a requestee or a requester, since it aims to establish the initial set of conditions for participation in a ULOOP community.

From a requester perspective, this implies generation of its virtual identity. Based on this virtual identity the requester initiates the creation of a set of trust parameters (c.f. Dispositional trust) that will influence the way the requester is willing to cooperate with other ULOOP nodes. Since the ULOOP trust environment may not be enough as an incentive for cooperation, the bootup phase ends up with the assignment of a set of credits that the requester may use to access shared resources.

While in idle mode, the requester sends wireless beacons in order to detect the local presence of ULOOP gateways (potential requestees). As a response to sent beacons, the requester may get a set of tuples providing indications about neighbour gateways, such as dispositional trust and resource threshold values. Based on the collected information the requester will try to establish trust associations with one of the responsive gateways (e.g. the one showing the best set of dispositional trust and resource thresholds), after which the requester will perform a MAC layer attachment with such ULOOP gateway.

After the attachment with one gateway, the requester will use the established association to send data to the gateway, taking advantage of a set of resources shared by the latter. As soon as the requester has data to send, it must first check if the association with the gateway needs to be re-established. This may be needed since the cooperation scheme among gateways may lead to the handover of the requester to a different gateway (from the set of cooperative ones) from the one the requester has initially set the association to.

Data transmission starts after the reception of a *Clear to Send (CTS)* by the gateway. Data transmission will be coordinated by the Resource Management block in what concerns the request to admit the transmission, and the data transmission itself. As soon as the Resource Management scheme acknowledges the end of the transmission, the negotiated resources against the provided resources will be compared. This evaluation will be important to adjust the trust levels between the requestee and the requester and their incentives for further cooperation.

In case the current trust level is not enough for the requestee to be motivated to cooperate, the requester will send an explicit request for cooperation. This request starts the cooperation incentive scheme, which includes the negotiation of the number of credits that the requester should transfer to the requestee at the end of the transmission, in case the requester has enough credits for the requested resources. While the requester will return to an idle mode after each transmission, at any moment the requester can adjust its dispositional trust based on output of any operation.

2.1.2.5.2 Providing Resources

As previously mentioned, a requestee in ULOOP is always a gateway that may offer resources to a ULOOP node or another gateway. The functionality of a requester has four major blocks: i) bootup; ii) cooperation request process; iii) data reception; iv) monetization and dispositional trust adjustment.

After bootup and while in idle mode, the requestee will receive wireless beacons that requesters send to detect the local presence of gateways. As a response, the requester will send a tuple providing indications about its dispositional trust and resource thresholds. This information will allow any requester to select the best gateway to establish a trust association with, after which the requestee will perform a MAC layer attachment with such requester.

Data transmission starts with the reception of a request to send from the requester, and will proceed only if the requestee has incentives to cooperate with such requester based on the established trust association only. After the transmission of a clear to send by the requestee, data transmission is coordinated by the Resource Management in what concerns the decision to admit the transmission, and the execution of the data transmission itself. As soon as the Resource Management scheme acknowledges the end of the transmission, the same will be evaluated. This evaluation is important to adjust the trust levels between the requestee and the requester and their incentives for further cooperation.

In case the requestee is not motivated to cooperate only based on the trust association with the requester, the latter has to send an explicit request for cooperation. After the reception of such request, the requestee triggers the cooperation incentive scheme, which includes the negotiation of the number of credits that the requester should transfer to the requestee at the end of the transmission, in case the requester has enough credits for the requested resources.

While the requestee returns to an idle mode after each transmission, at any moment it can adjust its dispositional trust based on output of any operation, and may decide to invoke the monetization process based on the earned credits.

2.1.3 Technical Readiness Level and Validation Aspects

Aspect	Technical readiness level/Validation	Reference(s)
Trust Setup	PoC[1]	[1][2][11][10][3][5]
	Papers	[23][25][26]
Crypto-id creation	PoC	[1][2][11][10][3][5]
Crypto-id validation	PoC	[1][2][11][10][3][5]
	Papers	Na
Dispositional trust	PoC	[1][2][11][10][3][5]
	Papers	[36]
Cooperation incentives – credit assignment	PoC	[1][2][11][10][3][5]
	Papers	[12][14][21][50]
Cooperation Incentives – evaluation	PoC	Not implemented
Rewarding scheme	PoC	[1][2][11][10][3][5]
	Papers	[54]
Tokens	PoC	[1][2][11][10][3][5]
	Papers	[53][41]
Social trust computation	Partially implemented / proof of concept	[1][2][11][10][3][5]
	Papers	[49][39]

2.2 Resource Management

The Resource Manager (RM) is the main skeleton of Task 3.2 at the requester side (station) and the requestee side (gateway). When the RM sub-block starts, the main function checks if the node is a

[1] PoC: Proof-of-Concept.

requester or requestee. If it's a requestee, it will be waiting for a trigger from trust management block or mobility block. If the request is from trust manager, this means there is a new request for resources from a node and this will trigger other functions in resource management. In fact, this triggers a request to CAC, which will prioritize the request and sends a request to Elastic Spectrum Management (ESM) for resources. ESM provides a feedback (0 or 1) to CAC and CAC forwards this to RM, which provides a feedback to trust management.

If the request is from mobility management, RM forwards this request to M&M, which provides a feedback to RM and RM provides this information back to mobility. If the node is a requester (station), this will initialize ESM and M&M at the requester side.

The code of the sub-blocks in resource management has been written in C programming language. The MAC layer of IEEE802.11g standard is written and developed in C language. Since ESM is a part of the MAC layer, ESM concept has been developed in C. Regarding the other sub-blocks in T3.2, the code is written in C because of the memory issues on devices, specifically the access points. The code in C has much less memory requirements compared to other programming language options (e.g. Java).

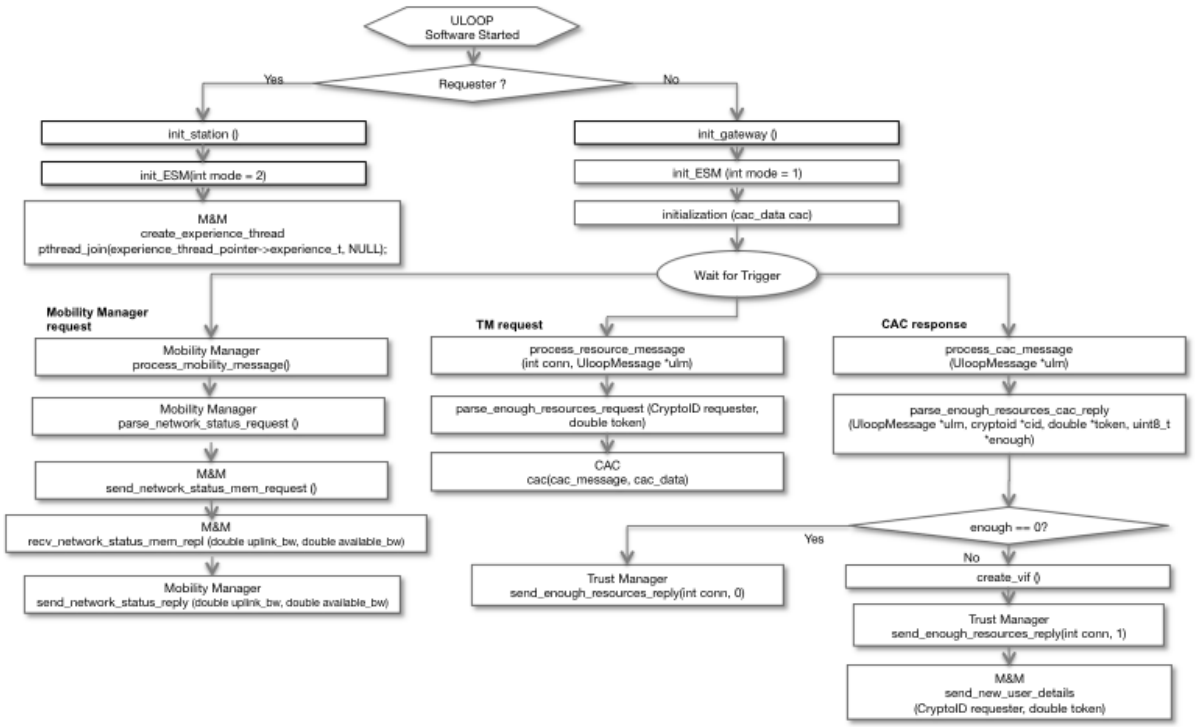

Figure 7: Flow-chart of resource manager.

In ULOOP, resource management relates to cooperative aspects and resource allocation, which are to be addressed from an OSI Layer 2 and Layer 3 perspective. The resource management operation takes place for nodes that have credits and are trusted in the community.

The resource management operation itself starts when a Gateway gets a request for resources. This request is mainly from trust management block on a ULOOP gateway. If the resources are available in the Gateway, the resource management block provides a positive feedback to trust management and the new node can then join the network. The resource management block also provides updates about the channel to the mobility block.

2.2.1 Call Admission Control based on Trust

Call Admission Control (CAC) is responsible for checking if there are available resources, on the gateway, to accept or deny a request from RM. The CAC is only enabled on the requestee side (Gateway), which means that all CAC code is done in C and implemented in an OpenWRT device. After the Resource Manager initializes the CAC function, the CAC stays in an idle state until a Resource Manager calls CAC or when the thread, scheduled to run before, wakes up to check the priority queue (pqueue).

When the Resource Manager calls CAC, CAC handles the incoming request, prioritizes it, and puts it on the pqueue. After this, CAC schedules the thread to run. When the thread wakes up, it checks if the pqueue is empty or not. If not, it enqueues the request with highest priority and then checks if the gateway can accept it or not. This is done by sending a request to ESM. ESM will provide a reply, 0 if not accepted or 1 for accepted request. After that, CAC will forward this reply to RM.

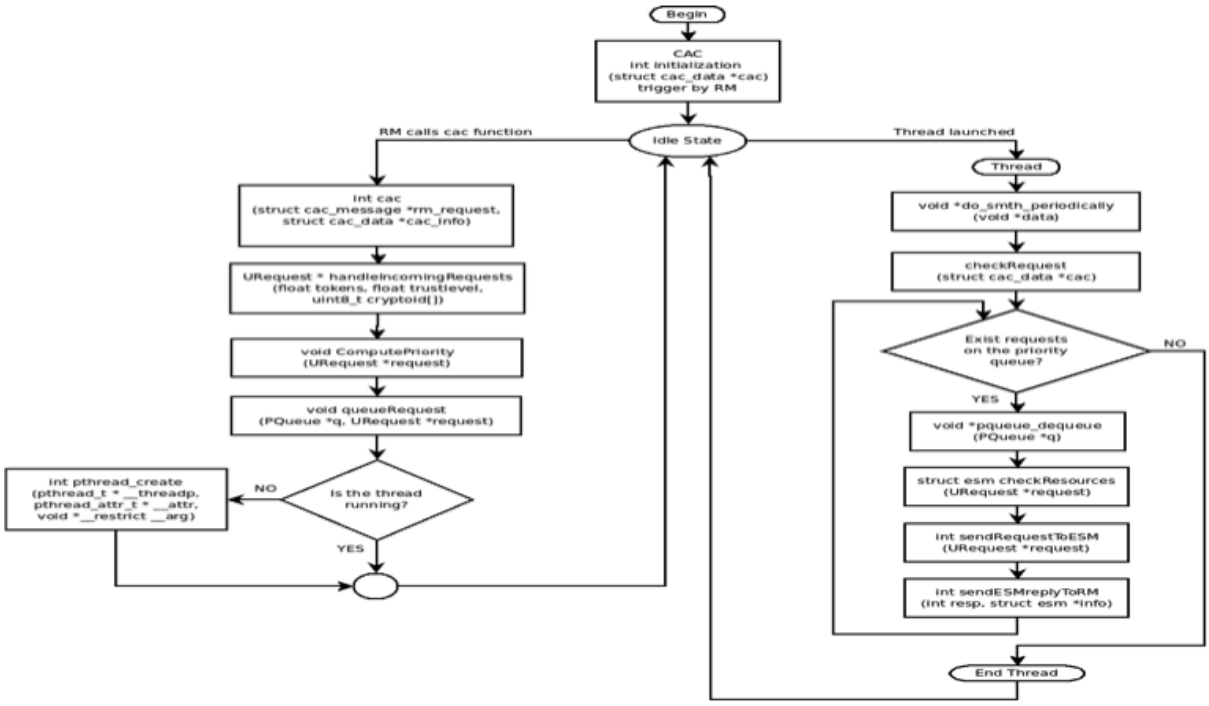

Figure 8: Flow-chart of call admission control.

2.2.2 Resource Allocation

ULOOP is envisioned to be applicable to dense area networks, which face, among other problems, the issue of interference. Moreover, in these environments, spectrum abounds and is underused. In ULOOP and in addition to augmented call admission control and self-organizing mechanisms to elect and to select gateways, a key aspect to be developed relates to considering mechanisms that allow the MAC layer to become more elastic in multi-user environments. Our intention is to provide such design without having to change the IEEE 802.11 standards – just by working with MAC frame format, and with the interpretation of such frames by ULOOP nodes.

Our work follows the recent trend concerning frequency assignment and sub-division which argues that the channel width of nodes should be adaptive. After reviewing the state-of-the-art we identify two major persistent drawbacks in this research trend: the coordination complexity intrinsic to the per-node channel width adaptation and the periodic computation of NP-hard problems. In this work we start studying an alternative way of arranging wireless channel assignments, based on *Orthogonal Frequency Division Multiplexing (OFDM)*. OFDM is supported by IEEE 802.11a/g/n standards, which are the ones that are dealt with in ULOOP.

To achieve this purpose - which we name *elastic spectrum management* - we have developed a new mechanism that employs adaptive multi-user access, modulation, error coding and power allocation techniques to judge the tradeoff between costs vs. performance gain. Such mechanism has been validated, showing good results in terms of network performance (throughput and end-to-end delay).

2.2.2.1 Elastic Spectrum Management on the Requestee

The Elastic Spectrum Management (ESM) is initialized by Resource Management, indicating which mode the ESM must work, as a gateway or as a station. For the gateway mode, the ESM, coded in C and implemented on OpenWRT devices, is responsible to assign a number of bits to a station depending on the number of tokens. Each station connected to the gateway has its own number of bits. This number will be used to create a superframe, containing multiple parts of different payloads of each station. A station with a higher number of tokens, compared to the others stations, has the right to write more data inside the superframe. A station with the least tokens may get to write less data or even nothing inside the payload, in case the 6 slots, each corresponding to 8 bits, have already been assigned to others stations with higher number of tokens. After the slots have all been assigned to the stations, the ESM will create the superframe and send it to the driver for transmission.

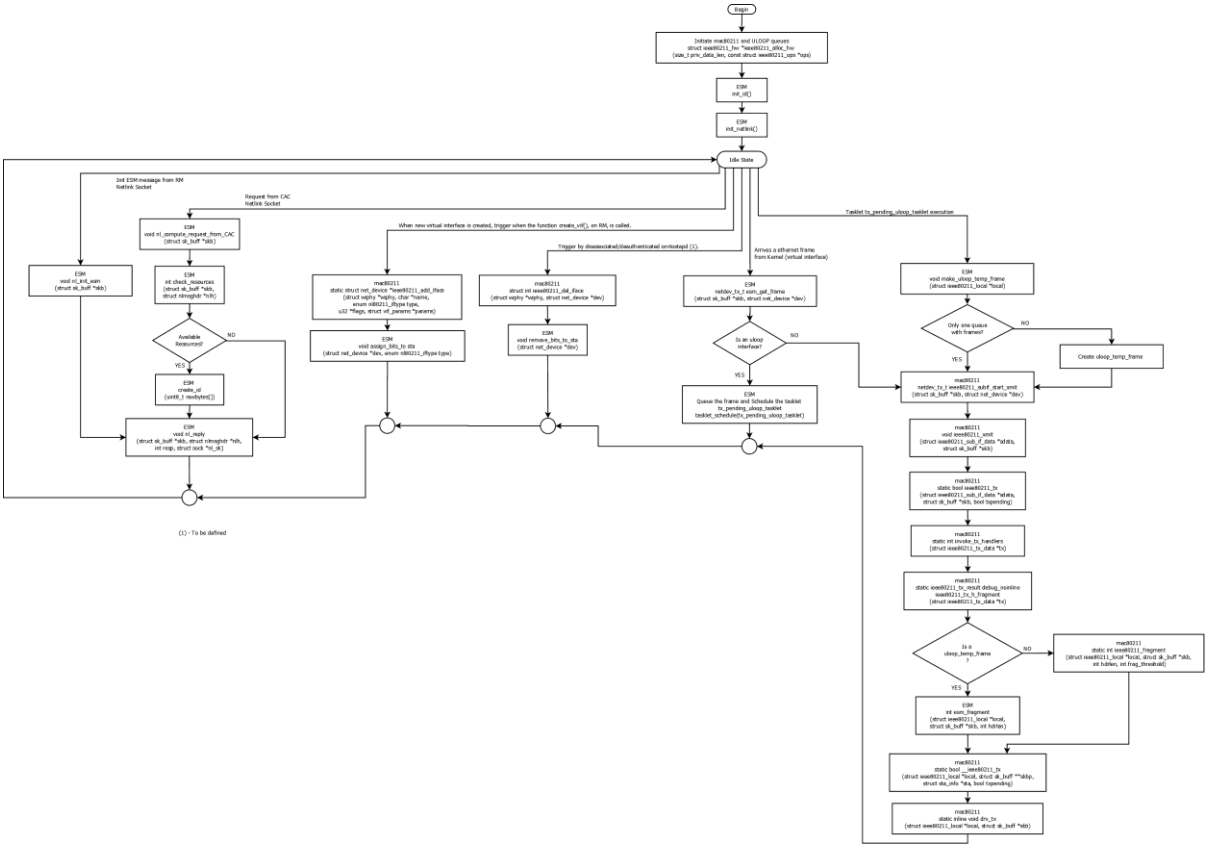

Figure 9: Flowchat of elastic spectrum management control on the requestee.

2.2.2.2 Elastic Spectrum Management on the Requester

The MAC layer of IEEE802.11g standard is written in C language. Since ESM is a part of the MAC layer, ESM concept has been developed and written in C programming language.

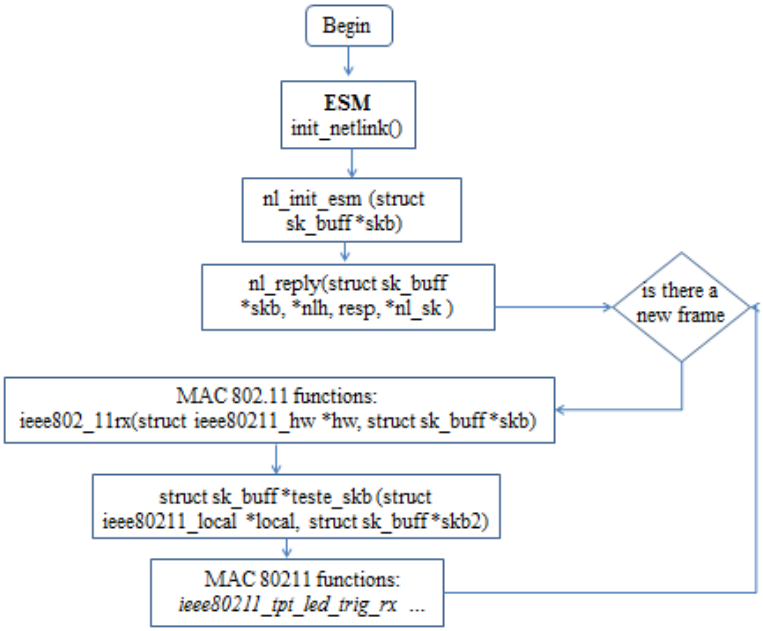

Figure 10: Flowchat of elastic spectrum management control on the requester.

2.2.3 Cooperative Relaying

This section describes the general functionality of a node running RelaySpot [18][19][24], the cooperative relaying solution devised in the ULOOP project. With RelaySpot a ULOOP node can operate as a source, potential relay or as a destination for each flow. RelaySpot can be used when the direct link between source and destination exists (proactive mode), or when the direct link fails (reactive mode).

Figure 11 illustrates the proactive operation of a RelaySpot node, based on an example with one source, two potential relays and one destination (the numbers before the messages refer to the order in which the frames are sent).

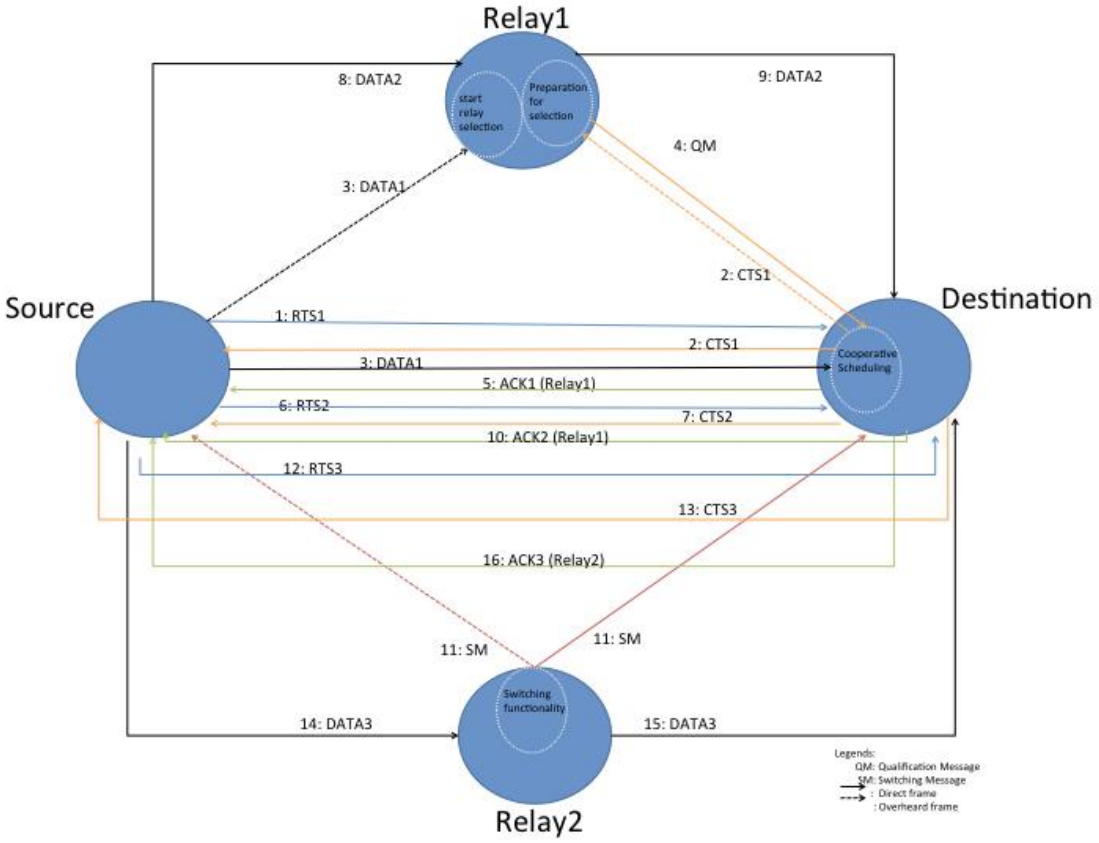

Figure 11: RelaySpot proactive functionality.

As shown in Figure 11, the operation starts as in a normal 802.11 network with the source starting an RTS/CTS procedure with the destination in order to gain access to the wireless medium. In the process, Relay2 (the potential relay present in the vicinity) overhears the CTS message and estimates the quality of the direct link. When the source transmits the data frame, this is overheard by Relay2, which activates the opportunistic relay selection mechanism in this node. As a result Relay2 transmits a Qualification Message (QM) to the destination aiming to notifying it of Relay2 availability to relay data from the source. Based on the information received from Relay2, the destination acknowledges the reception of the data frame and notifies the source that subsequent data frames should be sent via Relay2, since this offers better quality transmission. As a result, the source sends the next data frame through Relay2, after gaining again access to the wireless medium by executing the RTS/CTS operation with the destination. The reception of this message is acknowledged by the destination, informing the source that frames should keep being sent via Relay2. This acknowledgment message is overheard by a new potential relay in the vicinity (Relay3), which, after comparing its own cooperation factor with the one from Relay2, notifies the source and destination that it is a better relay than Relay2. As a consequence, the next time that the source gains access to the wireless medium (through an RTS/CTS procedure) it will send the data frame through Relay3.

Figure 12 illustrates the reactive operation of a RelaySpot node, based on an example with one source, two potential relays and one destination (the numbers before the messages refer to the order in which the frames are sent). In this scenario, the direct link has enough quality for the exchange of RTS/CTS frames, but not enough bandwidth for the transmission of the data frame. This means that the transmission between source and destination may end up without acknowledgment, meaning that the data frame was not delivered successfully. In such situation, the potential relays start opportunistic relay selection process, after detecting a missing acknowledgement to an overheard data frame. As a consequence, the first relay to gain access to the wireless medium (the one with best selection factor) will resend the overheard data frame to the destination. In Figure 12, Relay2 activates the opportunistic relay selection; as a result it retransmits the data frame.

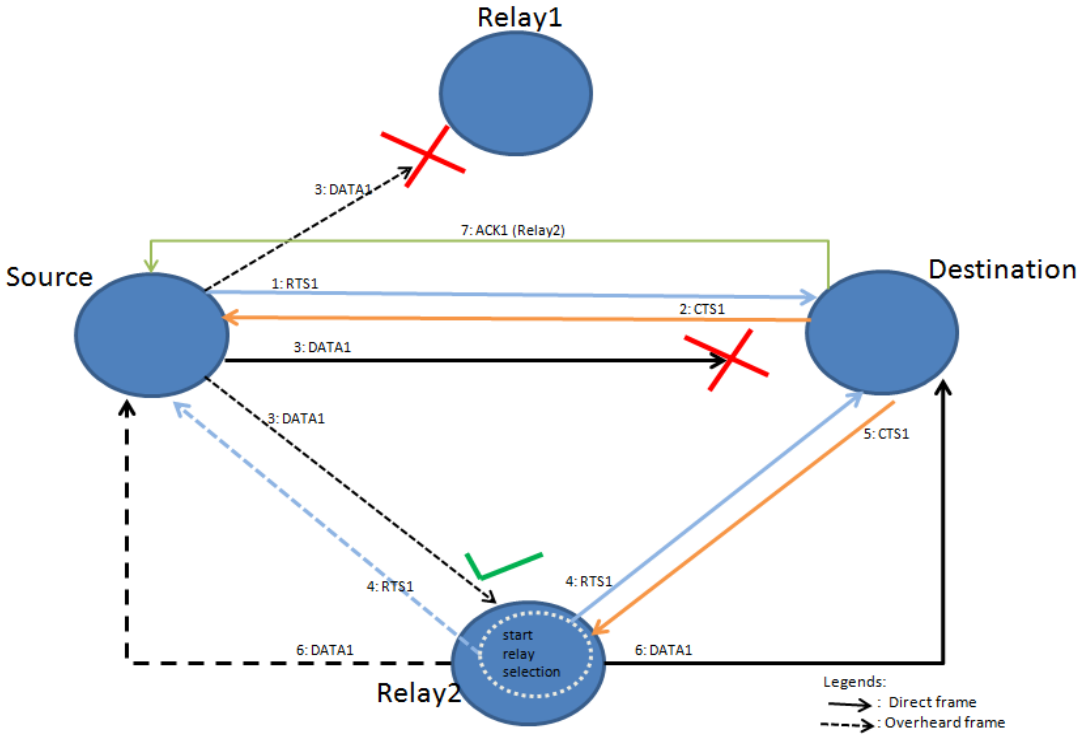

Figure 12: RelaySpot reactive functionality.

In what concerns the execution of the three mechanisms described in Section 3.1 and mentioned in the description of the generic functionality of RelaySpot, Figure 13, Figure 14, and Figure 15 provide a description of the sequence of operations related to the selection, scheduling and switching mechanisms.

Figure 13 provides a flowchart describing the opportunistic relay selection operation in general for one flow (RelaySpot starts parallel processes to handle each active flow). This opportunistic relay selection is performed on the relay node. The relay performs some background computation to estimate how good it is to help an active flow, namely by computing its selection factor. In the

presence of an active flow, it starts preparation for relay selection by checking its eligibility (previously computed value in background).

After overhearing the RTS/CTS exchange related to an active flow, the potential relay starts operating in a proactive mode, if *need for relaying* is indicated within CTS frame. In proactive mode, if the relay is eligible to improve the performance of the active flow (i.e., cooperation factor better than direct link), it starts the Contention Window (CW) based on the computed selection factor, in order to become a relay. After the expiration of the CW, and if ACK is not overheard for that flow, the relay performs relaying action by sending a Qualification Message (QM) to the destination as shown in Figure 11 (for an active link).

The relaying action is different for reactive relaying (broken link). In this case after overhearing a CTS frame with no indication for relaying, the potential relay does not overhear the acknowledgement of a previously overheard data frame. In this case, the potential relay sends the overheard data to the destination (instead of QM) if it satisfies some relaying conditions, and it does not overhear another retransmission after the expiration of the CW. Figure 13 also shows that opportunistic relay selection is activated due to the lack of ACK in case of reactive mode.

The role of the relaying scheduler at the destination is to select the best relay among opportunistic relays based on received QMs. Figure 14 illustrates the scheduling operation, which starts a parallel scheduling process for each active flow, if there is the need for relaying. For each active flow the destination checks the need for relaying the flow if the quality of the direct transmission is below a preconfigured threshold. If there is a need for relaying, this information is communicated in the CTS, allowing potential relays to pay attention to this specific flow. After this the destination starts a reception window as soon as the data frame is received in order to collect the QMs from potential relays. At the end of the reception window, the destination sends an acknowledgment message to the source with the identification of the relay or relays (if diversity is configured to a value higher than 1) selected to help this flow, if there are one or more relays that can improve the quality of the direct link.

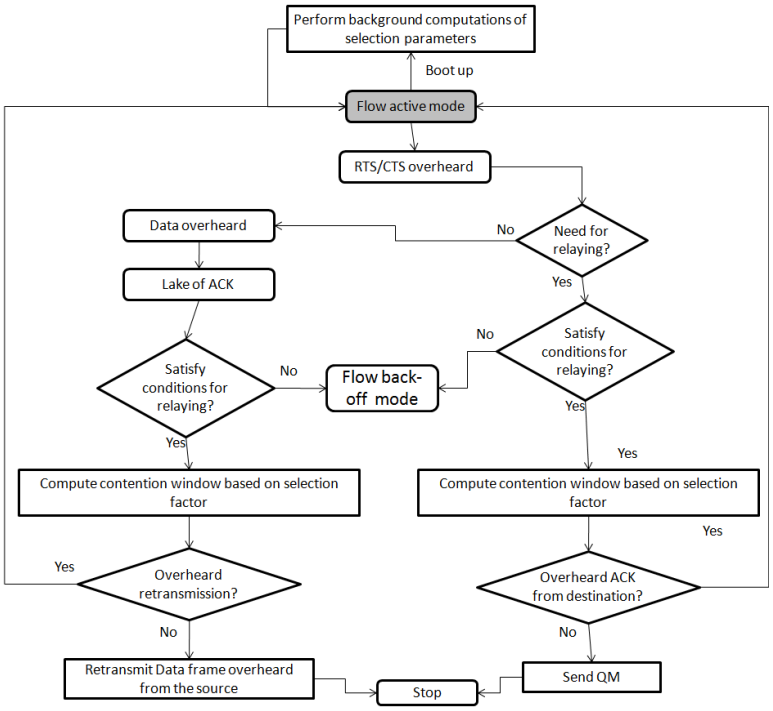

Figure 13: Opportunistic relay selection at relays.

If there is no need to relaying, the destination enters in a normal procedure without relaying, although the data frames that it can get would be relayed by a relay operating in reactive mode, and not directly from the source. In this case the acknowledgement is sent to the source, as in a normal 802.11 procedure.

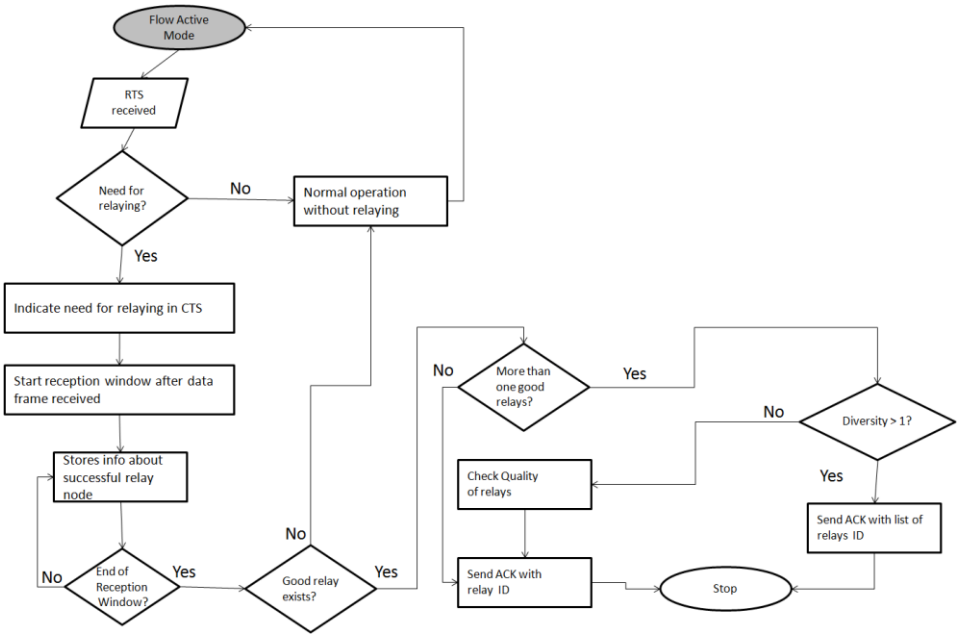

Figure 14: Cooperative relay scheduling at destination.

Switching between relays is performed when there is a potential relay that can improve the quality of the source – destination communication to a value higher than the one provided by the current relay. This situation can occur as a consequent of a bad estimation of the best relay by scheduler, or when a new relay comes to the vicinity of the source – destination link.

As shown in Figure 15, after overhearing the acknowledgement sent by the destination to a relayed communication, a potential relay checks if it satisfies the conditions for relay switching: as explained in Section 5.1.3, this happens if it has a cooperation factor higher than current relay. If so, the potential relay sends a Switching Message (SM) to the destination after the expiration of its content windows (this message is overheard by the source).

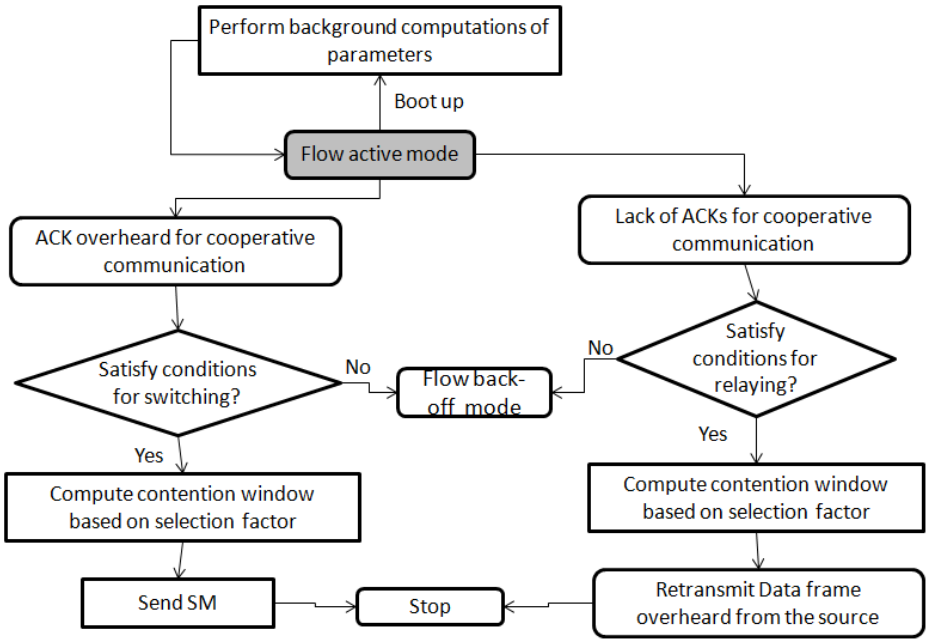

Figure 15: Relay switching operation at relays.

Relay switching is also used to keep data being relayed in the presence of a failed relay. If a potential relay detects that the cooperative transmission via a relay failed, that node tries to retransmit the failed data frame, leading to relay switching (implicitly).

2.2.4 Cooperative Load-Balancing

Due to the dynamic behaviour of ULOOP, nodes willing to share resources are more prone to be exposed to interference due to associations of other nodes. One of the aspects that is required to consider based on a self-organizing behaviour that is inherent to ULOOP gateways is to assist in preventing excessive resource consumption, i.e., by performing network load optimization. Part of this

mechanism relates to being able to shift in an optimal way stations across different gateways and also to be able to adequately perform load-balancing among gateways.

The aggregation of resource utilization, QoS and QoE measurements in a semantic form is the reasoning mechanism of the decision-making engines having the responsibility of load balancing trigger. Resource consumption monitoring that works in a passive way provides ULOOP gateways with the ability of classifying its clients according to bandwidth usage. The gateway arranges the client Id's with respect to their bandwidth consumption and marks the most consuming stations as "resource hungry". With this categorization, gateways can be aware of nodes that are less beneficial to the system, and if required assist their handover to other gateways while balancing load in the network in a fairer way. The flow-chart associated to this idea and currently undergoing investigation is provided by Figure 16, where ULOOP Gateway 1 corresponds to the source gateway, and ULOOP Gateway 2 corresponds to the destination gateway.

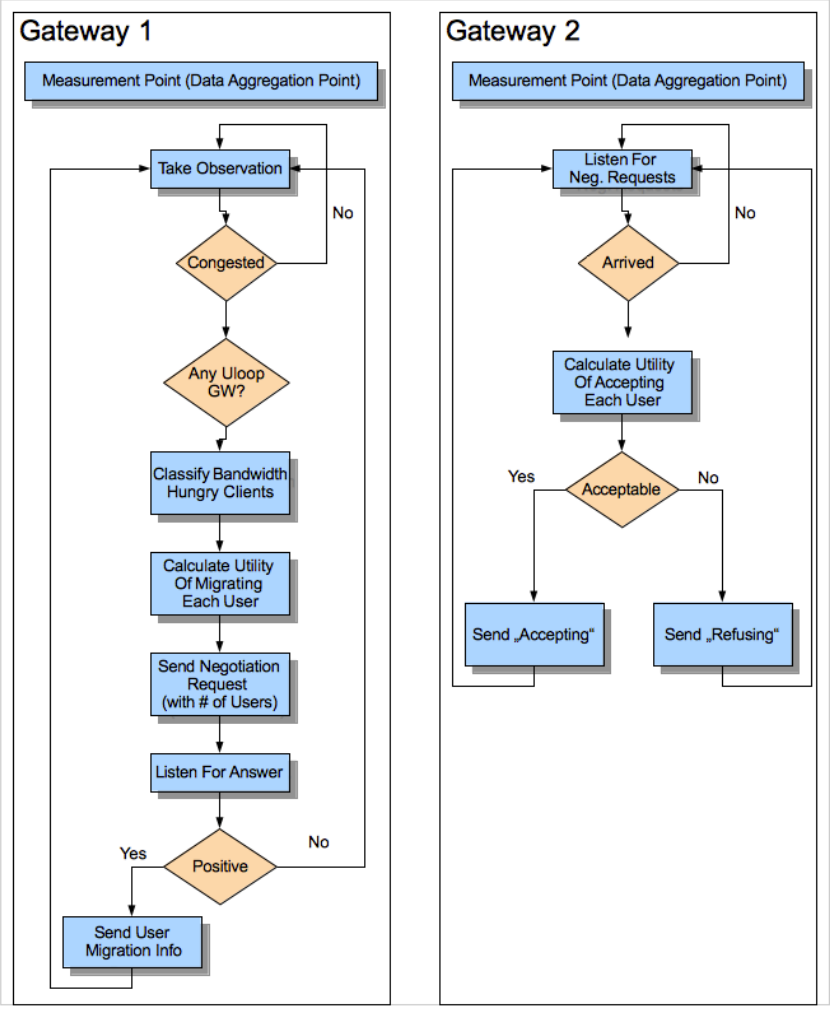

Figure 16: Load-balancing flow-chart.

2.2.5 Monitor and Measurement

The monitor and measurement (M&M) component is a multi-thread functionality. The M&M forms a distinct plane of the specific triggers for the upper level decision making engines in block functionalities. This unit runs both on the nodes and GWs in a collaborative manner, which is responsible for the following activities and functionalities:

- Providing an active measurement-taking platform for the nodes in extracting network performance related measurements.
- A network performance related measurement plane for upper level decision making units such as ESM, Mobility Management and potential Load balancing activities.
- Monitoring traffic behaviours and bandwidth utilization of the nodes with the motivation of resource consumption tracking.
- Providing a suitable architecture for the organization of measurement-related requests from the upper level block functionalities.

In Figure 17, we illustrate the detailed flow chart of the M&M and provide corresponding functionalities.

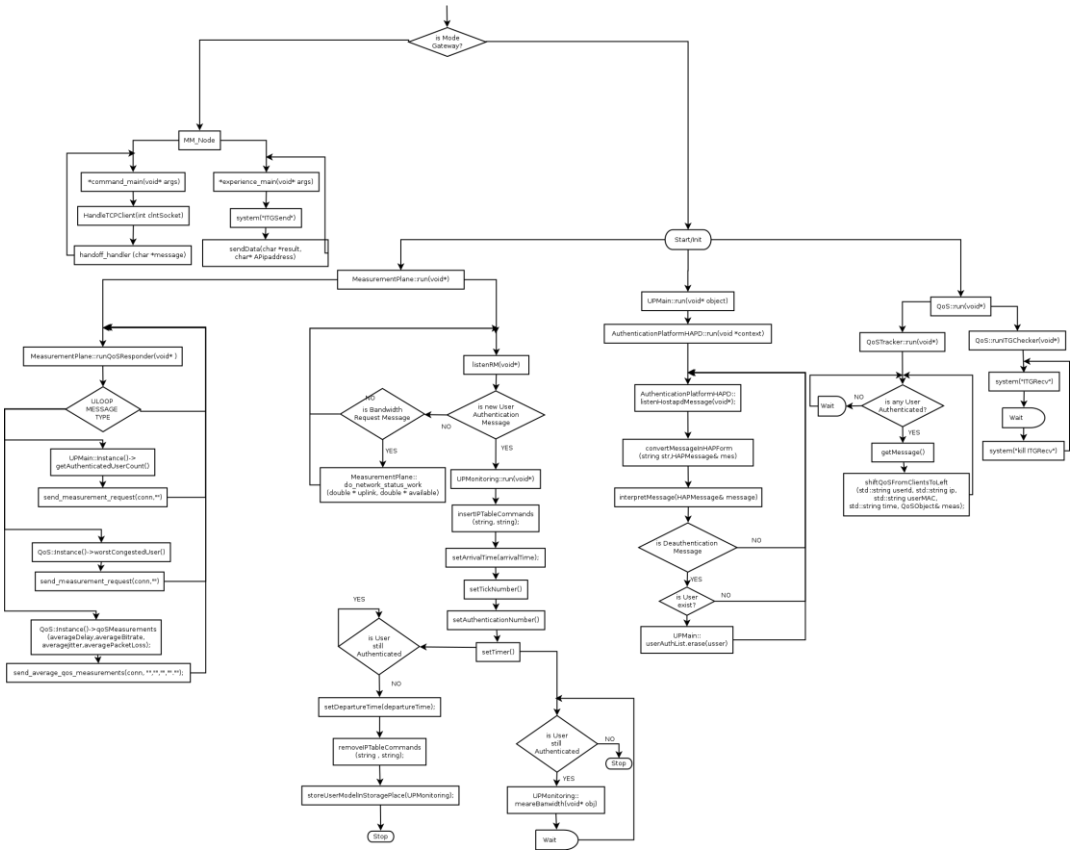

Figure 17: Flowchat of monitor and measurement component

2.2.6 Technical Readiness and Validation Aspects

Aspect	Technical readiness level	Reference(s)
Call admission control	PoC	[1][2][11][10][4][7]
	Papers	[53]
Resource allocation	PoC	[2][11][10][4][7]
	Papers	[13][16][17][34][47][48][53][57][58][59]
	IPRs	[62]
Cooperative load-balancing	ULOOP PoC add-on	[2][4][7]
	Papers	[55][56][35][60]
Monitor and measurement	PoC	[7] [11][10]
	Papers	Na
Cooperative relaying	Simulation module, OMNET++	[7]
	Papers	[18][19][24][32][33][1][40][44][42]
	IPRs	[63]

2.3 Mobility Aspects

UCNs are based on the notion of users carrying (or owning) low-cost and limited capacity portable devices which are cooperative in nature and which extend the network in a user-centric way, not necessarily implying the support for networking services such as multi-hop routing. For instance, in UCNs transmission may simply be relayed based on simple mechanisms already existing in end-user devices.

These emerging architectures therefore represent networks where the nodes that integrate the network are in fact end-user devices which may have additional storage capability and which may or may not sustain networking services. Such nodes, being carried by end-users exhibit a highly dynamic behaviour. Nodes move frequently following social patterns and based on their carriers interests; inter-contact exchange is the basis for the definition of connectivity models as well as data transmission. The network is also expected to frequently change (and even to experience frequent partitions) due to the fact that such nodes, being portable, are limited in terms of energy resources.

From a mobility perspective UCNs therefore exhibit a highly dynamic behaviour where the selection of the “best” mobility anchor points requires the pursuit of two main aspects: adequate selection and redundancy. This has to be achieved by always weighting user expectations and the support each user is willing to give as well as the network support (access sharing) each user can in fact provide to its counter-peers in the network.

Mobility anchor point location and selection optimization is therefore a crucial requirement of UCNs. Mobility anchor points may be part of the SP equipment, of the NAP equipment (edge node) or in fact be part of the equipment of the MP and this can increase heavily a UCN complexity. Table 1 illustrates Mobility management assumptions and requirements for UCNs.

Table 1: Mobility management assumptions and requirements for UCNs

Assumptions	MP is a key target in terms of network management (and hence of mobility management) VO is simply a coordinator of authentication Users (and carried devices) roam frequently – devices carried or owned by humans Node movement follows human movement patterns
Requirements	Mobility anchor point redundancy Optimal mobility anchor point selection Flexible mobility management architecture (most likely, decentralized)

Therefore, based on the aforementioned requirements, ULOOP has addressed how to assist the network and the user in terms of mobility, by allowing devices to infer future roaming behaviour, based on a selection optimization that simply relies on data available to devices, and which concerns visited networks. Concrete examples of network parameters include, but are not limited to:

- Number of visits performed over a specific period of time, e.g. 24h. The counter is increased if MN1 gets attached to a specific AP.
- Average duration of one visit. The average duration starts when MN1 attaches to a MAP, and stops when MN1 gets detached from that MAP.
- Visited network attractiveness. In ULOOP this corresponds to the trust level that a node has towards a specific AP of a network that is regularly visited.
- Number of times MN1 has been accepted on that visited network.
- Time gap since the last visit to a specific visited network, represented as time elapsed.

For each visited network, MN1 then computes (locally, seamlessly, and periodically) a cost (a ranking parameter) based on a specific formula that relies on the collected network parameters. That ranking parameter is also stored in the listing of visited networks of MN1. As proof of concept, ULOOP has worked these concepts and integrated them into the end-user background application MTracker, described in section 2.3.4.

2.3.1 Mobility Anchor Point

Mobility Anchor Point (MAP) is developed in ULOOP to be extended and adapted in the mobility anchor function of an existing mobility management solution. It interacts with resource management to get the resource information in the gateway and registers its context and sends keep alive message to the Mobility Coordination Function (MCF). It is selected by the MCF to provide the mobility anchor function in user-centric environment.

The PMIP protocol in RFC 5213 is selected in ULOOP project as the existing mobility management solution, and OPMIP open source project is selected as the potential PMIP implementation. The OPMIP open source is coded in C++, to integrate the MAP into the LMA function of the OPMIP implementation, the MAP code in OpenWRT is provided in C++.

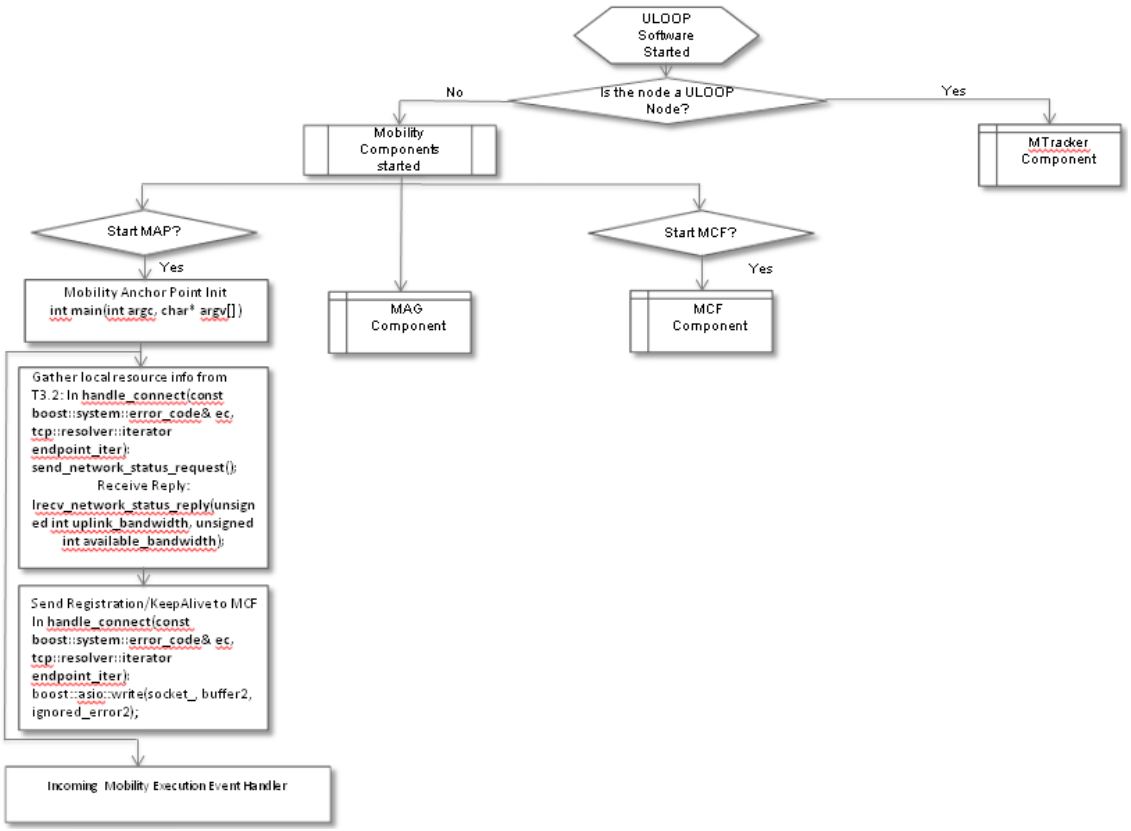

Figure 18: Flow-chart of mobility anchor point.

2.3.2 Mobility Access Gateway

Mobility Access Gateway (MAG) is developed in ULOOP to be extended and adapted in the access gateway function of an existing mobility management solution. It requests the gateway provided mobility anchor function for the ULOOP node from the Mobility Coordination Function (MCF), and interworking with mobility tracker and trust management to select the suitable mobility anchor function for the ULOOP node.

The PMIP protocol in RFC 5213 is selected in ULOOP project as the existing mobility management solution, and OPMIP open source project is selected as the potential PMIP implementation. The OPMIP open source is coded in C++, to integrate the MAG into the MAG function of the OPMIP implementation, the MAG code in OpenWRT is provided in C++.

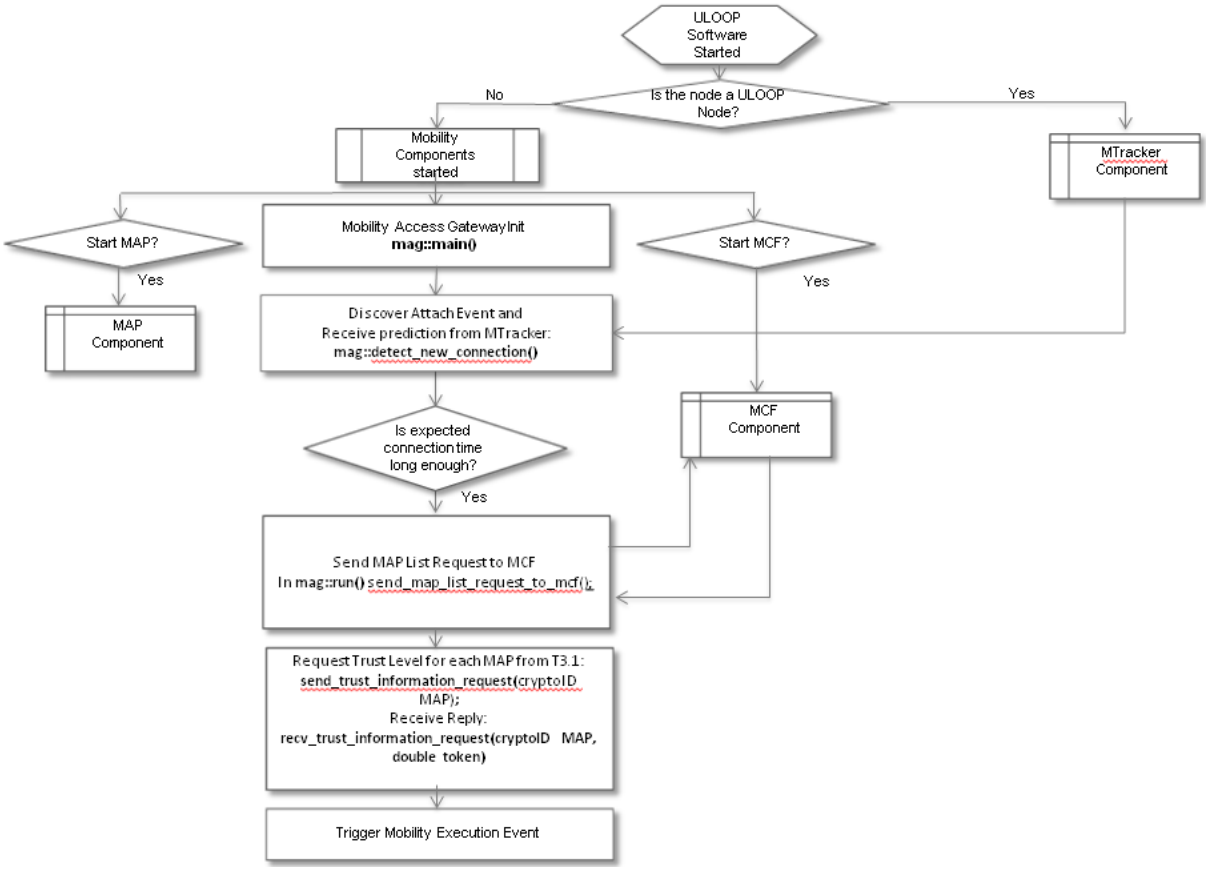

Figure 19: Flow-chart of mobility access gateway.

2.3.3 Mobility Coordination Function

The Mobility Coordination Function (MCF) is responsible for maintaining currently active, registered Mobility Anchor Points (MAPs). Based on the number of currently known active MAPs it is responsible to perform a MAP selection decision for the ULOOP node upon receiving MAP request from the MAG, which are then enforced on the data path.

The MAP, MAG and MCF are integrated and provided as single OpenWrt Application in ULOOP. The MAP and MAG codes are provided in C++, to integrate MCF with MAP and MAG, the MCF code in OpenWRT is provided in C++.

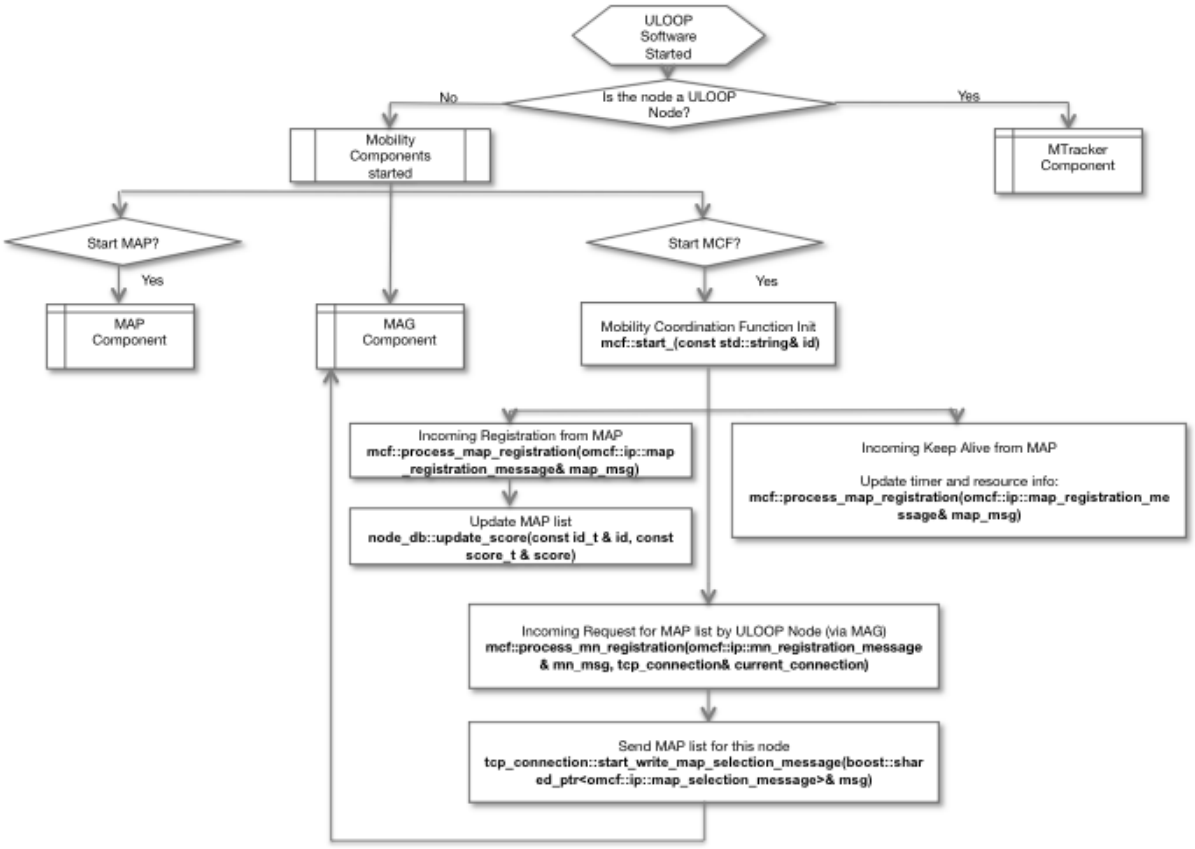

Figure 20: Flow-chart of mobility coordination function

2.3.4 Mobility Tracker

The Mobility Tracker (or MTracker) is an application that passively tracks anonymous properties of a user's roaming behaviour, and ranks each visited network based on a specific algorithm which takes into consideration aspects such as number of visits to a given access point and the average duration of such visits. The MTracker application then tries to predict in how much time the node will change the network connection, and which will be the next network. MTracker is currently available in Android, as well as in C#.

Within the user side, the MTracker collects information concerning visited networks, periodically computing a ranking to each visited network. Then, again periodically, it emits a message to potential anchor points or, in the case of ULOOP, to the MCF. This is done by having a MTracker plugin on the gateway, aspect which facilitates the future development of MTracker.

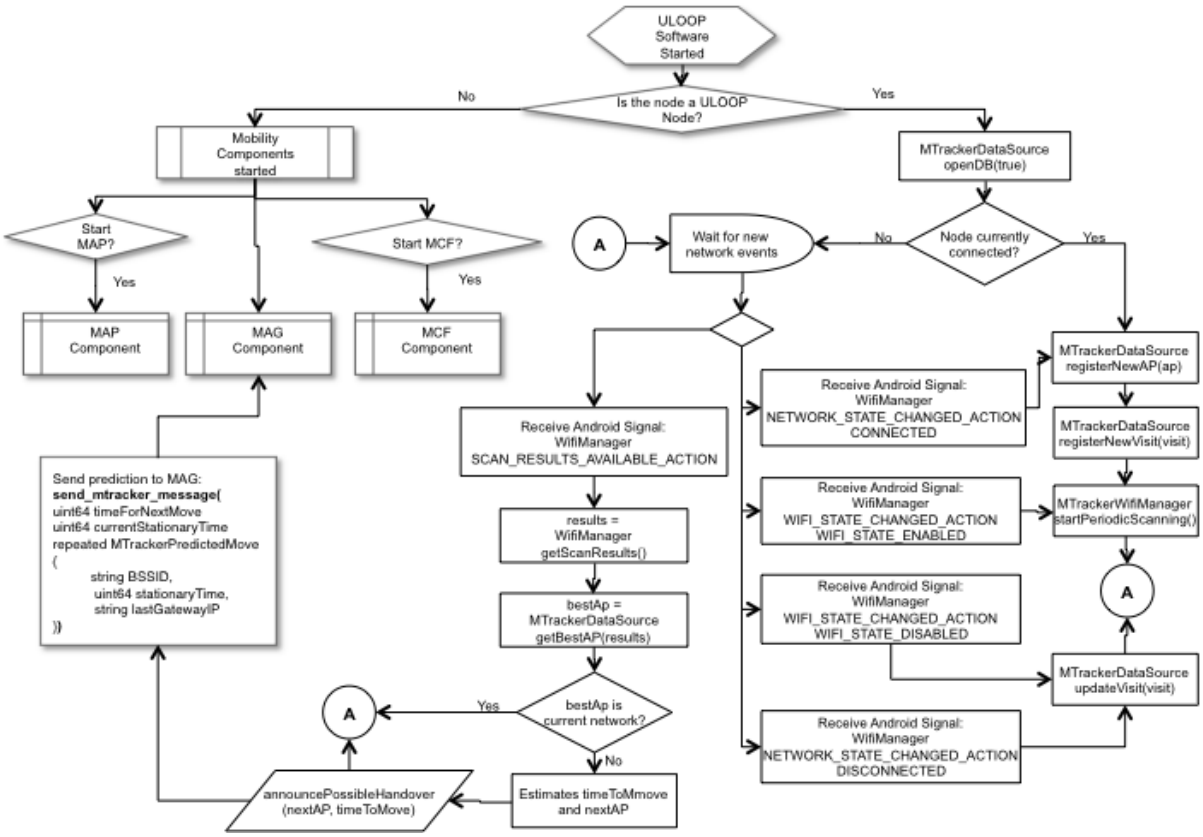

Figure 21: Flow-chart of mobility tracker.

2.3.5 Technical Readiness and Validation Aspects

Aspect	Technical readiness level	Reference(s)
Mobility Tracker	PoC	[6][1][2][11][9]
	IPR	[61]
	Papers	[51][43][52][45]
MCF	PoC	[10][11]

3. References

3.1 Deliverables

[1] Rute Sofia (editor). “D2.3: Initial Guidelines and Global Framework”. September 2011. ULOOP European project deliverable (gr. Nr 257418)

[2] Rute Sofia (editor). “D3: ULOOP High-level specification – Initial design, first six months”. December 2011. ULOOP European project deliverable (gr. Nr 257418).

[3] Carlos Ballester (editor). “D3.1: Cooperation Incentives and Trust Management Pre-Prototype Software”, June 2012. ULOOP European project deliverable (gr. Nr 257418)

[4] Huiling Zhu (editor). “D3.2: Resource Management Pre-prototype Software Support Report”, June 2012. ULOOP European project deliverable (gr. Nr 257418)

[5] Sebastian Peters (editor). “D3.3: Mobility Aspects Pre-prototype Software Support Report”, June 2012. ULOOP European project deliverable (gr. Nr 257418)

[6] Carlos Ballester (editor). “D3.4: Cooperation Incentives and Trust Management Specification and Refined Software Report”, October 2012. ULOOP European project deliverable (gr. Nr 257418)

[7] Temitope Alade, Huiling Zhu (editors). “D3.5: Resource Management Specification and Refined Software Report”, October 2012. ULOOP European project deliverable (gr. Nr 257418)

[8] Qing Zhou (editor). “D3.6: ULOOP Mobility Aspects Specification and Refined Software”, October 2012. ULOOP European project deliverable (gr. Nr 257418)

[9] Paulo Mendes (editor). “D3a: ULOOP Low-level Specification - Specification of ULOOP functionality based on flow-chart and link to code available on the ULOOP SVN server”. May 2013. ULOOP European project deliverable (gr. Nr 257418)

[10] Alfredo Matos (editor). “D3.7: ULOOP software pre-release suite”. August 2013. ULOOP European project deliverable (gr. Nr 257418).

[11] Alfredo Matos (editor). “D3.9: ULOOP Software Suite”. October 2013. ULOOP European project deliverable (gr. Nr 257418).

3.2 Scientific Papers, Accepted

[12] A. Aldini, A. Bogliolo. Model Checking of Trust-Based User-Centric Cooperative Networks. W-PIN2012, April 2012.

[13] H. Haci. Novel Scheduling for a Mixture of Real-time and Non-real-time Traffic. IEEE GLobecom 2012, Best paper award. September 2012.

[14] A. Aldini, Trading Cooperation Incentives and Performance in UCNs. Quasa ESORICS 2012. July 2012.

[15] M. Yildiz. Cooperation Incentives Based Load Balancing in UCN: A Probabilistic Approach. IEEE Globecom 2012. July 2012.

[16] H. Zhu and J. Wang, “Chunk-based resource allocation in OFDMA systems - part I: chunk allocation,” *Communications, IEEE Transactions on*, vol. 57, no. 9, pp. 2734-2744, 2009.

[17] H. Haci, H. Zhu and J. Wang, “Resource Allocation in User-Centric Wireless Networks”, VTC-Spring, 2012.

[18] T. Jamal, Paulo Mendes, André Zuquete, "RelaySpot: A Framework for Opportunistic Cooperative Relaying", ACCESS conference, Luxembourg, June 19-24, 2011.

[19] T. Jamal, Paulo Mendes, André Zúquete "Interference-Aware Opportunistic Relay Selection", in Proc. of ACM CoNext (student workshop), Tokyo, Japan, December 2011.

[20] Andréa Ribeiro, Rute C. Sofia and André Zúquete, Improving Mobile Networks based on Social Mobility modeling, IEEE International Conference on Network Protocols, 2011.

[21] Paulo Mendes, Waldir Moreira Junior, Christian da Silva Pereira, Tauseef Jamal, Alessadro Bogliolo, Huseyin Haci and huiling zhu, Cooperative Networking in User-centric Wireless Networks, ULOOP White Paper 05, 2012.

[22] Alessadro Bogliolo, Saverio Delpriori, Lorenz Klopfenstein, Alessandro Aldini, Jean-Marc Seigneur and Waldir Moreira Junior, Crediting Aspects in ULOOP, ULOOP White Paper 09, 2012.

[23] Carlos Ballester, Jean-Marc Seigneur, Rute C. Sofia, Christian da Silva Pereira, Waldir Moreira Junior and Alessadro Bogliolo, Trust Management in ULOOP, White Paper 07, 2012.

[24] Tauseef Jamal, Paulo Mendes and André Zúquete, Wireless Cooperative Relaying Based on Opportunistic Relay Selection, IARIA International Journal On Advances in Networks and Services (invited paper), 2012

[25] Carlos Ballester, Jean-Marc Seigneur, Paolo di Francesco, Valentin Moreno, Rute C. Sofia, Alessadro Bogliolo, Nuno martins and Waldir Moreira Junior, A User-centric Approach to Trust Management in Wi-Fi Networks, IEEE INFOCOM - Demos Track, 2013. (**Internal reference UNIGE-2**).

[26] Jean-Marc Seigneur, Carlos Ballester Lafuente, Alfredo Matos, Secure User-Friendly Wi-Fi Access Point Joining. IEEE WCNC 2013. (**Internal reference UNIGE-3**)

[27] Aldini, A. Bogliolo. Model Checking of Trust-Based User-Centric Cooperative Networks. AFIN 2012, July 2012.

[28] Carlos Ballester Lafuentea, Jean-Marc Seigneur, Waldir Moreira, Paulo Mendes b, Linas Maknavicius. A Survey on Trust and Cooperation Incentives for Wireless User-centric Environments. IADIS e-Society 2012.

[29] A. Bogliolo, P. Polidori, A. Aldini, W. Moreira, P. Mendes, M. Yildiz, C. Ballester, J.-M. Seigneur . Virtual Currency and Reputation-based Cooperation Incentives. IWCMC2012.

[30] Xavier Titi, Jean-Marc Seigneur, Carlos Ballester. Trust Management for Selecting Trustworthy Access Points. International Journal of Computer Science Issues, Volume 8, Issue 2, 2011.

[31] Xavier Titi, Jean-Marc Seigneur, Carlos Ballester. Boosting Trustworthy Hotspot QoE Rating with Implicit Hotspot QoS Evidence. IADIS e-Society 2011, March 2011, Avila, Spain.

[32] T. Jamal, P. Mendes. RelaySpot: A Framework for Opportunistic Cooperative Relaying. Proc of International Conference on Access Networks, June 2011, Luxembourg.

[33] T. Jamal, P. Mendes. Relay Selection Approaches for Wireless Cooperative Networks. IEEE WiMob Workshop on Cooperative Mobile Protocols and Application 2010 Communications, Vol 6, Vol6:No 1 (2011)(56-57).

[34] H. Haci, H. Zhu, J. Wang . Resource allocation in user-centric networks. Vt2012, May 2012.

[35] M. Yildiz. Mobility Behavior Modeling in UCN. IEEE WCNC2013. (**Internal reference TUB-4**).

[36] C. Ballester, Jean-Marc Seigneur. Dispositional Trust Self-Adaptation in User-Centric Networks. In Proc. of AINA, March 2013. (**Internal reference UNIGE-1**).

[37] A. Aldini. Formal Approach to Design and Automatic Verification of Cooperation-Based Networks. IARIA International Journal On Advances in Internet Technology, June 2013. (**Internal reference UNIURB-1**).

[38] A. Aldini. A. Bogliolo. Modeling and Verification of Cooperation Incentive Mechanisms in User-Centric Wireless Communications. Book chapter in Danda B. Rawat, Bhed B. Bista and Gongjun Yan (Editors), "Security, Privacy, Trust, and Resource Management in Mobile and Wireless Communications" IGI Global, 2013. (**Internal reference UNIURB-2**).

[39] R. Sofia, L. Lopes. Trust as a fairness parameter for QoE in Wireless Networks. Springer LNCS ULOOP Book, 2014. (**Internal reference ULHT-14**).

[40] T. Jamal, P. Mendes. Cooperative relaying in user-centric networks. Springer LNCS ULOOP Book, 2014. (**Internal reference ULHT-4**).

[41] R. Sofia. Bringing the Home Network to the Core. Springer LNCS ULOOP Book, 2014. (**Internal reference ULHT-12**).

[42] P. Mendes, W. Moreira, T. Jamal, H. Haci, H.Zhu. Cooperative networking in user-centric networks. Springer LNCS ULOOP Book, 2014. (**Internal reference ULHT-15**).

[43] R. Sofia. User-centric networking: the ULOOP perspective. Chapter in Springer LNCS WiNeMo (Wireless Networks Moving Objects) Book, accepted. (**Internal reference ULHT-8**).

[44] T. Jamal, P. Mendes. Cooperative Relaying for Dynamic Networks. Chapter in Springer LNCS WiNeMo (Wireless Networks Moving Objects) Book, accepted. (**Internal reference ULHT-3**).

[45] A. Ribeiro, R. Sofia. Social Mobility Modeling on Heterogeneous Wireless Networks. Chapter in Springer LNCS WiNeMo (Wireless Networks Moving Objects) Book, accepted. (**Internal reference ULHT-16**).

[46] R. Sofia, P. Mendes, A. Bogliolo, H. Zhu, F. Sivrikaya. “User-centric Networking and Services”. IEEE Communications Feature Topic Proposal. (**Internal reference ULHT-19**).

3.3 Scientific Papers, Under Submission

[47] H. Haci, Huiling Zhu, Rute Sofia, Paulo Mendes. Stochastic Performance Analysis of Proportional Fair Scheduling. Under preparation (**internal reference UNIK-1**).

[48] H. Haci, H. Zhu, R. Sofia, P. Mendes, T. Jamal. A Survey of Resource Management in User-centric Networks. Submitted to IEEE Surveys and Tutorials, 2013. (**Internal reference UNIK-2**).

[49] R. Sofia, C. Ballester, Jean-Marc Seigneur. Social Trust as a Boost Mechanism for User-centric Networks. Submitted to IEEE Wireless Communications, Consumer and Networking Series Special Issue. (**Internal reference ULHT-1**).

[50] P. Mendes, W. Moreira, R. Sofia, L. Lopes. Motivating Cooperative Behavior in Wireless User-Centric Networks. Under submission to IEEE Communications Magazine, Consumer Communication and Networking Series. (**Internal reference ULHT-2**).

[1] T. Jamal, P. Mendes. Cooperative Relaying for Dynamic Wireless Networks. Under submission to IEEE Transaction of Wireless Communications. (**Internal reference ULHT-6**).

[51] R. Sofia. Predicting Mobile Social Behavior in User-centric Wireless Environments. Submitted to Elsevier PMC Journal. (**Internal reference ULHT-7**).

[52] R. Sofia, J. Saltarin. Mtracker, a Mobility estimation tool. Under preparation for Elsevier JNC. (**Internal reference ULHT-10**).

[53] R. Sofia, L. Lopes, H. Zhu, A. Bogliolo, M. Yildiz. A trust-based scheduling mechanism. Under preparation. (**Internal reference ULHT-11**).

[54] L. Klopfeinstein, S. Delpriori, A. Aldini, A. Bogliolo,. Designing a trust-based virtual currency system for user-centric networks: the ULOOP case. Under preparation. (**Internal reference UNIURB-3**).

[55] M. Yildiz. An Autonomous Load Balancing Framework for UCN. Under preparation for IEEE Communications Magazine. (**Internal reference TUB-1**).

[56] M. Yildiz. AP selection in UCN: A network Performance History Based Approach. Under preparation for IEEE Transactions on Wireless Communications. (**Internal reference TUB-3**).

[57] L. Lopes. R. Sofia. H. Osman. A Proposal for Elastic Spectrum Management in Wireless Local Area Networks. Under preparation for INFOCOM 2014, Demo Track. (**Internal reference ULHT-17**)

[58] L. Lopes. R. Sofia, H. Osman, H. Haci, H. Zhu. A proposal for Dynamic Frequency Sharing in Wireless Local Area networks. Under preparation to IEEE Communications magazine, Special Issue: the Future of Wi-Fi. (**Internal reference ULHT-18**)

[59] R. Sofia, H. Zhu, H. Haci, A. Bogliolo. A Virtual Currency for Direct Exchange of Resources based on Trust and Credits. Under Submission (**Internal reference ULHT-9**).

[60] M. Yildiz. A Survey on Load Balancing (or Congestion Avoidance) in Wireless Networks. Under submission to Elsevier Computer Communications. (**Internal reference TUB-2**).

3.4 IPRs

[61] R. Sofia. “Method and Apparatus for Ranking Visited Networks” (EP 13186562.9). August 2013.

[62] H. Osman, H. Zhu, H. Haci, L. Lopes, R. Sofia. “Method and Apparatus for communication in a wireless network” (EP 13191667.8), August 2013.

[63] T. Jamal, P. Mendes. “Cooperative Relaying for Dynamic Networks” (EP13182366.8). August 2013

3.5 Other Material

[64] Mirco Musolesi and Cecilia Mascolo. 2007. Designing mobility models based on social network theory. *SIGMOBILE Mob. Comput. Commun. Rev.* 11, 3 (July 2007), 59-70. DOI=10.1145/1317425.1317433 http://doi.acm.org/10.1145/1317425.1317433.

[65] Chiara Boldrini, Andrea Passarella, HCMM: Modelling spatial and temporal properties of human mobility driven by users' social relationships, Computer Communications, Volume 33, Issue 9, 1 June 2010, Pages 1056-1074, ISSN 0140-3664, DOI: 10.1016/j.comcom.2010.01.013.